Grid Induced Minor Theorem for Graphs of Small Degree

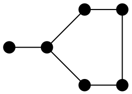
Tuukka Korhonen

UNIVERSITY OF BERGEN

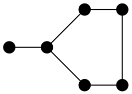
CSGT 2022

Prague, July 27, 2022

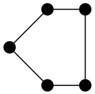
Graph containment



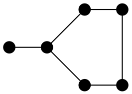
Graph containment



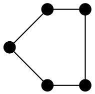
1. Induced subgraph
   - vertex deletions



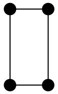
Graph containment



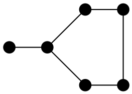
1. Induced subgraph
   - vertex deletions



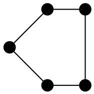
2. Induced minor
   - vertex deletions
   - edge contractions



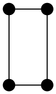
Graph containment



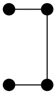
1. Induced subgraph
   - vertex deletions



2. Induced minor
   - vertex deletions
   - edge contractions



3. Minor
   - vertex deletions
   - edge contractions
   - edge deletions



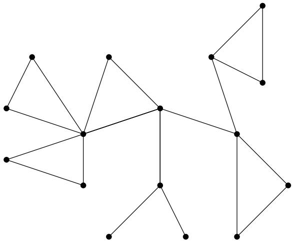
Graph classses defined by containment

- For a graph H, we can define graph classes by excluding H
  - H-minor-free graphs
  - H-induced-minor-free graphs

Graph classses defined by containment

- For a graph H, we can define graph classes by excluding H
  - H-minor-free graphs
  - H-induced-minor-free graphs

- Example: C_4-minor-free graphs

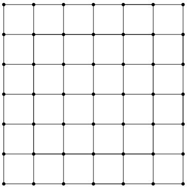


- Every biconnected component is a triangle
- Chordal and treewidth ≤ 2

Grid minor theorem

Robertson & Seymour, Graph Minors V. ('86):

- If H is planar, then H-minor-free graphs have bounded treewidth

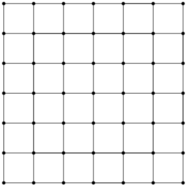


Grid minor theorem

Robertson & Seymour, Graph Minors V. ('86):

- If H is planar, then H-minor-free graphs have bounded treewidth

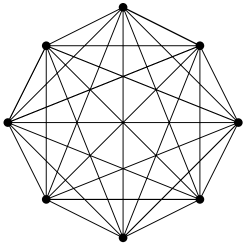
$\Leftrightarrow$ There is a function f so that if a graph has treewidth $\geq f(k)$, then it contains a $k \times k$-grid as a minor



Grid minor theorem for induced minors?

Grid minor theorem for induced minors?

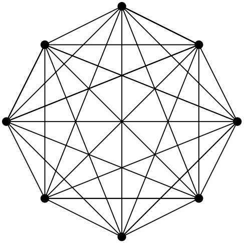
- Complete graph contains only complete graphs as induced minors



Grid minor theorem for induced minors?

- Complete graph contains only complete graphs as induced minors

⇒ C_4-induced-minor-free graphs have unbounded treewidth

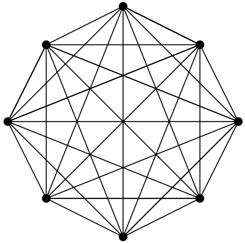


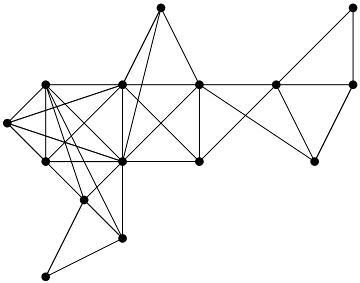
Grid minor theorem for induced minors?

- Complete graph contains only complete graphs as induced minors

⇒ C_4-induced-minor-free graphs have unbounded treewidth

- C_4-induced-minor-free graphs ⇔ chordal graphs





When does grid minor theorem work for induced minors?

When does grid minor theorem work for induced minors?

Theorem (Fomin, Golovach, and Thilikos, 2011)

For any graph H there is a constant C_H so that any H-minor-free graph with treewidth $\geq C_H \cdot k$ contains a $k \times k$-grid as an induced minor.

When does grid minor theorem work for induced minors?

Theorem (Fomin, Golovach, and Thilikos, 2011)

For any graph H there is a constant C_H so that any H-minor-free graph with treewidth $\geq C_H \cdot k$ contains a $k \times k$-grid as an induced minor.

Theorem (K. 2022)

There is a function $f(k, d) = \mathcal{O}(k^{10} + 2^{d^5})$ so that any graph with maximum degree d and treewidth $\geq f(k, d)$ contains a $k \times k$-grid as an induced minor.

When does grid minor theorem work for induced minors?

Theorem (Fomin, Golovach, and Thilikos, 2011)

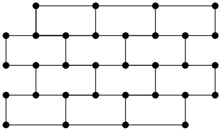
For any graph H there is a constant C_H so that any H-minor-free graph with treewidth $\geq C_H \cdot k$ contains a $k \times k$-grid as an induced minor.

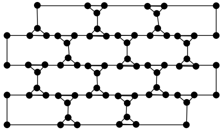
Theorem (K. 2022)

There is a function $f(k, d) = \mathcal{O}(k^{10} + 2^{d^5})$ so that any graph with maximum degree d and treewidth $\geq f(k, d)$ contains a $k \times k$-grid as an induced minor.

Corollary (Induced subgraph version)

There is a function $f(k, d)$ so that any graph with maximum degree d and treewidth $\geq f(k, d)$ contains a $k \times k$-wall or the line graph of a $k \times k$-wall as an induced subgraph.





Conjectured by Aboulker, Adler, Kim, Sintiari, and Trotignon, 2021

Algorithmic motivation/application

Open problem

Is there a quasipolynomial time algorithm for maximum independent set on H-induced-minor-free graphs when H is planar?

Recall: Quasipolynomial time is $2^{\text{polylog} n}$ time

Algorithmic motivation/application

Open problem

Is there a quasipolynomial time algorithm for maximum independent set on H-induced-minor-free graphs when H is planar?

Recall: Quasipolynomial time is $2^{\text{polylog} n}$ time

Solved special cases:

- $H = P_t$ [Gartland and Lokshtanov, 2020]
- $H = C_t$ [Gartland, Lokshtanov, Pilipczuk, Pilipczuk, Rzazewski, 2021]
- $H = W_4$, K_5^-, or $K_{2,q}$ [Dallard, Milanic, Storgel, 2022]

Algorithmic motivation/application

Open problem

Is there a quasipolynomial time algorithm for maximum independent set on H-induced-minor-free graphs when H is planar?

Recall: Quasipolynomial time is $2^{\text{polylog} n}$ time

Solved special cases:

- $H = P_t$ [Gartland and Lokshtanov, 2020]
- $H = C_t$ [Gartland, Lokshtanov, Pilipczuk, Pilipczuk, Rzazewski, 2021]
- $H = W_4$, K_5^-, or $K_{2,q}$ [Dallard, Milanic, Storgel, 2022]

Important open special case:

- even-hole-free graphs

Algorithmic motivation/application

Open problem

Is there a quasipolynomial time algorithm for maximum independent set on H-induced-minor-free graphs when H is planar?

Recall: Quasipolynomial time is $2^{\text{polylog} n}$ time

Solved special cases:

- $H = P_t$ [Gartland and Lokshtanov, 2020]
- $H = C_t$ [Gartland, Lokshtanov, Pilipczuk, Pilipczuk, Rzazewski, 2021]
- $H = W_4$, K_5^-, or $K_{2,q}$ [Dallard, Milanic, Storgel, 2022]

Important open special case:

- even-hole-free graphs

Our theorem directly gives

$\Rightarrow$ Linear-time algorithm when input graphs have bounded degree

Algorithmic motivation/application

Open problem

Is there a quasipolynomial time algorithm for maximum independent set on H-induced-minor-free graphs when H is planar?

Recall: Quasipolynomial time is $2^{\text{polylog} n}$ time

Solved special cases:

- $H = P_t$ [Gartland and Lokshtanov, 2020]
- $H = C_t$ [Gartland, Lokshtanov, Pilipczuk, Pilipczuk, Rzazewski, 2021]
- $H = W_4$, K_5^-, or $K_{2,q}$ [Dallard, Milanic, Storgel, 2022]

Important open special case:

- even-hole-free graphs

Our theorem directly gives

$\Rightarrow$ Linear-time algorithm when input graphs have bounded degree

$\Rightarrow$ $\mathcal{O}(2^{n/\log^{1/6} n})$ time algorithm in general

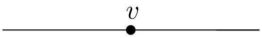
Proof

Sparsifiable vertices

A vertex v of G is *sparsifiable* if it satisfies one of the following:

Sparsifiable vertices

A vertex v of G is *sparsifiable* if it satisfies one of the following:

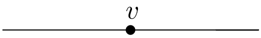


1. v has degree ≤ 2

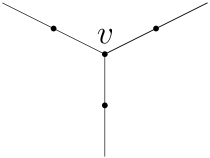
Sparsifiable vertices

A vertex v of G is *sparsifiable* if it satisfies one of the following:

1. v has degree ≤ 2



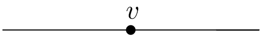
2. v has degree 3 and all neighbors of v have degree ≤ 2



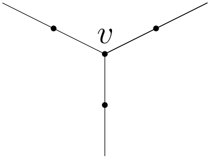
Sparsifiable vertices

A vertex *v* of *G* is *sparsifiable* if it satisfies one of the following:

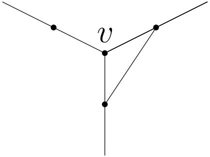
1. *v* has degree ≤ 2



2. *v* has degree 3 and all neighbors of *v* have degree ≤ 2



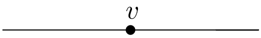
3. *v* has degree 3, one of its neighbors has degree ≤ 2, and two others form a triangle with *v*



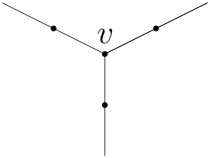
Sparsifiable vertices

A vertex v of G is *sparsifiable* if it satisfies one of the following:

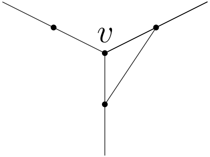
1. v has degree ≤ 2



2. v has degree 3 and all neighbors of v have degree ≤ 2



3. v has degree 3, one of its neighbors has degree ≤ 2, and two others form a triangle with v

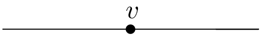


Lemma

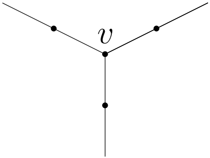
Let G, H be graphs so that every vertex of G is sparsifiable and H has minimum degree ≥ 3. Then G contains H as an induced minor if G contains H as a minor.

Sparsifiable vertices

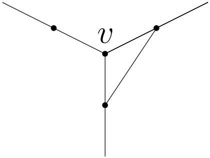
A vertex *v* of *G* is *sparsifiable* if it satisfies one of the following:



1. *v* has degree ≤ 2



2. *v* has degree 3 and all neighbors of *v* have degree ≤ 2



3. *v* has degree 3, one of its neighbors has degree ≤ 2, and two others form a triangle with *v*

Lemma

Let G, H be graphs so that every vertex of G is sparsifiable and H has minimum degree ≥ 3. Then G contains H as an induced minor if G contains H as a minor.

Goal: Make every vertex of G sparsifiable while maintaining high treewidth

Iterative sparsification

Let G be a graph and d the maximum degree of G

Iterative sparsification

Let G be a graph and d the maximum degree of G

1. Partition $V(G)$ into d^5 distance-5 independent sets $I_1, I_2, \ldots, I_{d^5}$

Iterative sparsification

Let G be a graph and d the maximum degree of G

1. Partition $V(G)$ into d^5 distance-5 independent sets $I_1, I_2, \ldots, I_{d^5}$
2. Do d^5 iterations. In iteration i, delete vertices so that all vertices of I_i are either made sparsifiable or deleted, and the treewidth of G stays high

Iterative sparsification

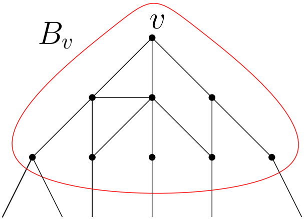
Let G be a graph and d the maximum degree of G

1. Partition $V(G)$ into d^5 distance-5 independent sets $I_1, I_2, \ldots, I_{d^5}$
2. Do d^5 iterations. In iteration i, delete vertices so that all vertices of I_i are either made sparsifiable or deleted, and the treewidth of G stays high

$\Rightarrow$ Finish by using the lemma of the previous slide with the grid minor theorem

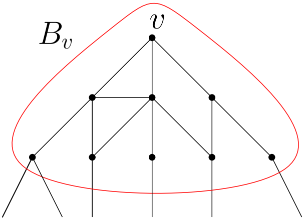
Making vertices sparsifiable

For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



Making vertices sparsifiable

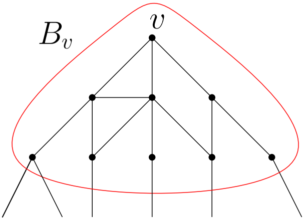
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent

Making vertices sparsifiable

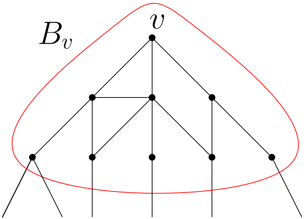
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex

Making vertices sparsifiable

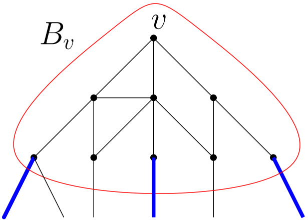
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex
- G' has large treewidth, G' has subgraph G'' with maximum degree 3 and large treewidth

Making vertices sparsifiable

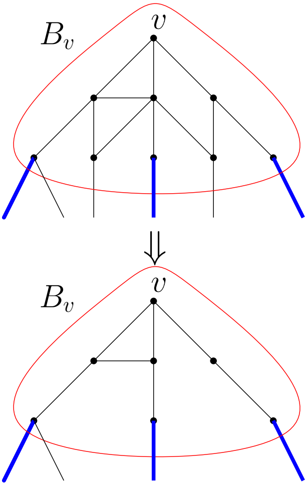
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex
- G' has large treewidth, G' has subgraph G'' with maximum degree 3 and large treewidth
- G'' is a minor of G, with a minor model given by the balls B_v

Making vertices sparsifiable

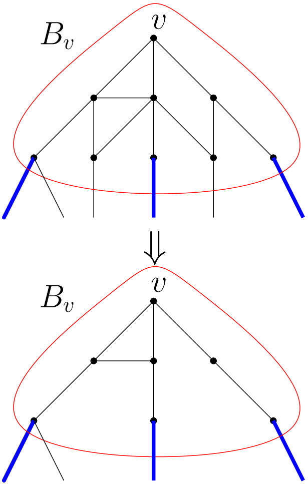
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex
- G' has large treewidth, G' has subgraph G'' with maximum degree 3 and large treewidth
- G'' is a minor of G, with a minor model given by the balls B_v
- delete vertices from B_v so that v is either deleted or made sparsifiable while preserving the minor model

Making vertices sparsifiable

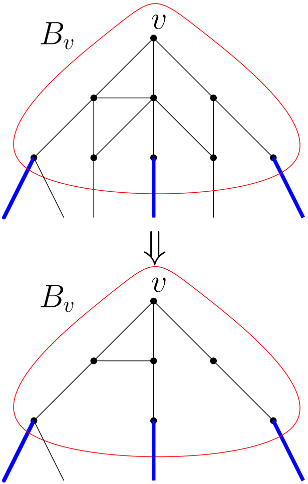
For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex
- G' has large treewidth, G' has subgraph G'' with maximum degree 3 and large treewidth
- G'' is a minor of G, with a minor model given by the balls B_v
- delete vertices from B_v so that v is either deleted or made sparsifiable while preserving the minor model(case analysis with 4 cases)

Making vertices sparsifiable

For each vertex $v \in I_i$, let B_v be the distance-2 ball around v



- the balls B_v do not overlap because I_i is distance-5 independent
- let G' be the graph obtained by contracting each ball B_v into one vertex
- G' has large treewidth, G' has subgraph G'' with maximum degree 3 and large treewidth
- G'' is a minor of G, with a minor model given by the balls B_v
- delete vertices from B_v so that v is either deleted or made sparsifiable while preserving the minor model(case analysis with 4 cases)
- all vertices in I_i become sparsifiable while preserving G'' as a minor

Thank you!

Thank you!