Dynamic programming on bipartite tree decompositions

Lars Jaffke[1], Laure Morelle[2], Ignasi Sau[2], Dimitrios M. Thilikos[2]

JGA 2023

[1]Department of Informatics, University of **Bergen, Norway**
[2]CNRS, LIRMM, Université de **Montpellier, France**

Odd-minors

Let H and G be two graphs.

Odd-minors

Let H and G be two graphs.

H is a **minor** of G if H can be obtained from a subgraph of G by contracting edges.

Odd-minors

Let H and G be two graphs.

H is a **minor** of G if H can be obtained from a subgraph of G by contracting edges.

H is an **odd-minor** of G if H can be obtained from a subgraph of G by contracting an edge cut.

Edge cut E': $\exists$ partition (A, B) of $V(G)$ s.t. $E' = E(A, B)$.

Odd-minors

Let H and G be two graphs.

H is a **minor** of G if H can be obtained from a subgraph of G by contracting edges.

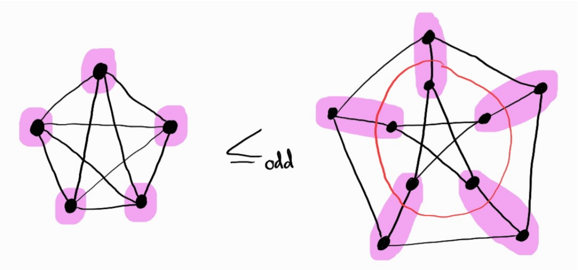
H is an **odd-minor** of G if H can be obtained from a subgraph of G by contracting an edge cut.

Edge cut E': $\exists$ partition (A, B) of $V(G)$ s.t. $E' = E(A, B)$.

odd-minor $\Rightarrow$ minor.

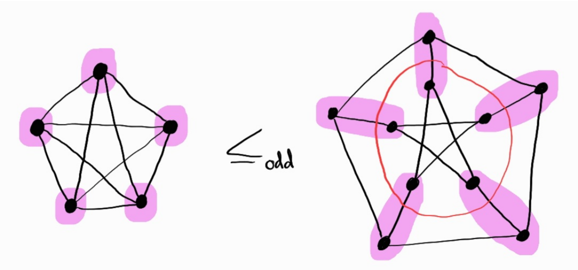
Odd-minors

Examples



Odd-minors

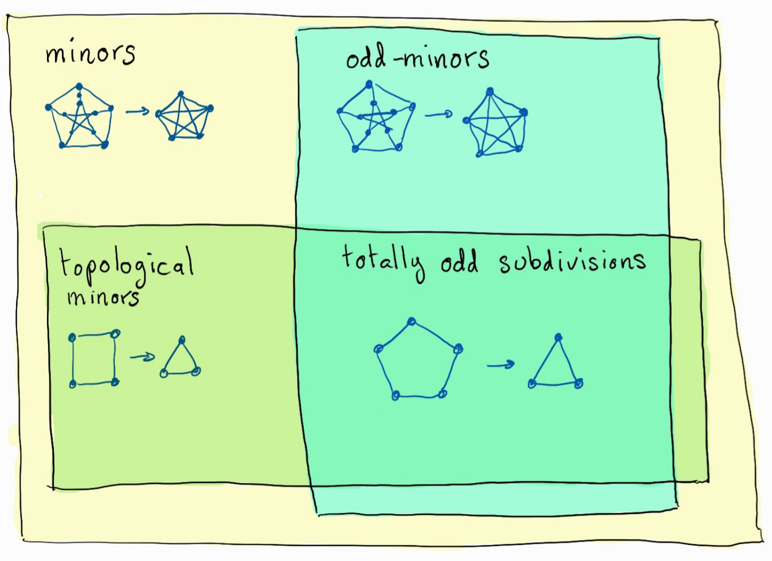
Examples



→ odd-minor = "minor preserving the parity of cycles"

Odd-minors

Hierarchy



Odd-minors

Some motivations

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.
2. Odd Hadwiger's conjecture: If a graph excludes K_t as an odd-minor, then it is $(t-1)$-colorable. [Gerards, Seymour, '93]

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.
2. Odd Hadwiger's conjecture: If a graph excludes K_t as an odd-minor, then it is $(t-1)$-colorable. [Gerards, Seymour, '93]
3. MaxCut is solvable in polynomial time on K_5-odd-minor-free graphs. [Grötschel, Pulleyblank, '81] & [Guenin, '01]

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.
2. Odd Hadwiger's conjecture: If a graph excludes K_t as an odd-minor, then it is $(t-1)$-colorable. [Gerards, Seymour, '93]
3. MaxCut is solvable in polynomial time on K_5-odd-minor-free graphs. [Grötschel, Pulleyblank, '81] & [Guenin, '01]
4. Odd-minors-free graph classes seem well-suited to do parameterized complexity.

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.
2. Odd Hadwiger's conjecture: If a graph excludes K_t as an odd-minor, then it is $(t-1)$-colorable. [Gerards, Seymour, '93]
3. MaxCut is solvable in polynomial time on K_5-odd-minor-free graphs. [Grötschel, Pulleyblank, '81] & [Guenin, '01]
4. Odd-minors-free graph classes seem well-suited to do parameterized complexity.
5. Odd-minors are not well-studied yet.

Odd-minors

Some motivations

1. Odd-minor-free graphs generalize both minor-free graphs and bipartite graphs.
2. Odd Hadwiger's conjecture: If a graph excludes K_t as an odd-minor, then it is $(t-1)$-colorable. [Gerards, Seymour, '93]
3. MAXCUT is solvable in polynomial time on K_5-odd-minor-free graphs. [Grötschel, Pulleyblank, '81] & [Guenin, '01]
4. Odd-minors-free graph classes seem well-suited to do parameterized complexity.
5. Odd-minors are not well-studied yet.

Our objective: make a first step towards a sound algorithmic and structural theory for odd-minors.

An analog of treewidth for odd-minors?

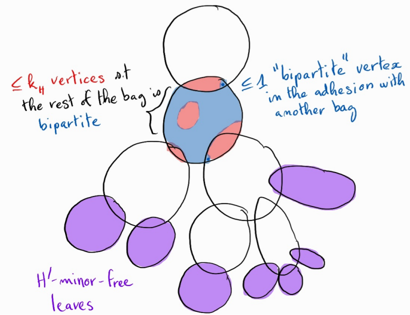
Tazari's decomposition [Tazari, '12], adapted from [Demaine, Hajiaghayi, Kawarabayashi, '10]

An analog of treewidth for odd-minors?

Tazari's decomposition [Tazari, '12], adapted from [Demaine, Hajiaghayi, Kawarabayashi, '10]

Tazari's decomposition: Let G be an H-odd-minor-free graph. Then there is a rooted tree decomposition of G s.t:

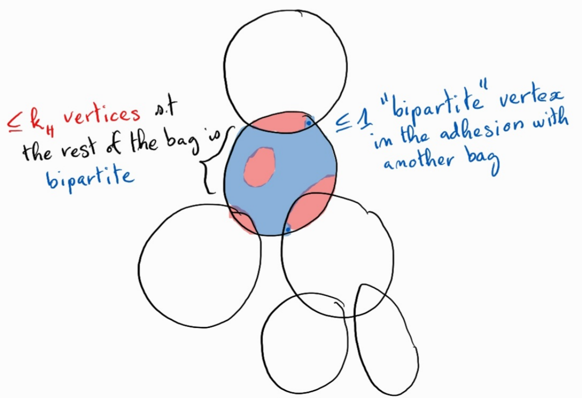
- the leaf-bags are H'-minor-free,
- each internal bag induces a bipartite graph along with at most k_H apex vertices, and
- there is at most one "bipartite" vertex in the adhesion of an internal bag and another bag.



An analog of treewidth for odd-minors?

Bipartite tree decomposition

Bipartite tree decomposition: Tazari's decomposition without the H'-minor-free leaves



Bipartite tree decomposition

Formal definition

Bipartite tree decomposition of G: triple (T, α, β) s.t.

- T is a tree, $\alpha, \beta : V(T) \to 2^{V(G)}$ (apex and bipartite),
- $(T, \alpha \cup \beta)$ is a tree decomposition of G
- for $t \in V(T)$, $G[\beta(t)]$ is bipartite
- for $tt' \in E(T)$, $|\beta(t) \cap (\alpha(t') \cup \beta(t'))| \leq 1$ (and vice versa)

Width of (T, α, β): $\max_{t \in V(T)} |\alpha(t)|$.

Bipartite treewidth (btw) of G: minimum width of a bipartite tree decompositions of G.

Bipartite tree decomposition

Formal definition

Bipartite tree decomposition of G: triple (T, α, β) s.t.

- T is a tree, $\alpha, \beta : V(T) \to 2^{V(G)}$ (apex and bipartite),
- $(T, \alpha \cup \beta)$ is a tree decomposition of G
- for $t \in V(T)$, $G[\beta(t)]$ is bipartite
- for $tt' \in E(T)$, $|\beta(t) \cap (\alpha(t') \cup \beta(t'))| \leq 1$ (and vice versa)

Width of (T, α, β): $\max_{t \in V(T)} |\alpha(t)|$.

Bipartite treewidth (btw) of G: minimum width of a bipartite tree decompositions of G.

$\to$ odd-minor-closed

Bipartite tree decomposition

Formal definition

Bipartite tree decomposition of G: triple (T, α, β) s.t.

- T is a tree, $\alpha, \beta : V(T) \to 2^{V(G)}$ (apex and bipartite),
- $(T, \alpha \cup \beta)$ is a tree decomposition of G
- for $t \in V(T)$, $G[\beta(t)]$ is bipartite
- for $tt' \in E(T)$, $|\beta(t) \cap (\alpha(t') \cup \beta(t'))| \leq 1$ (and vice versa)

Width of (T, α, β): $\max_{t \in V(T)} |\alpha(t)|$.

Bipartite treewidth (btw) of G: minimum width of a bipartite tree decompositions of G.

$\to$ odd-minor-closed
$\to$ if replace 1 by $q \geq 2$: not odd-minor-closed

Bipartite tree decomposition

Formal definition

Bipartite tree decomposition of G: triple (T, α, β) s.t.

- T is a tree, $\alpha, \beta : V(T) \to 2^{V(G)}$ (apex and bipartite),
- $(T, \alpha \cup \beta)$ is a tree decomposition of G
- for $t \in V(T)$, $G[\beta(t)]$ is bipartite
- for $tt' \in E(T)$, $|\beta(t) \cap (\alpha(t') \cup \beta(t'))| \leq 1$ (and vice versa)

Width of (T, α, β): $\max_{t \in V(T)} |\alpha(t)|$.

Bipartite treewidth (btw) of G: minimum width of a bipartite tree decompositions of G.

$\to$ odd-minor-closed
$\to$ if replace 1 by $q \geq 2$: not odd-minor-closed
$\to$ if replace 1 by 0: $\mathcal{H}$-treewidth for $\mathcal{H}$ = bipartite graphs.

Bipartite tree decomposition

Not new

Bipartite tree decomposition

Not new

Bipartite treewidth was already implicitly used in previous papers like [Kawarabayashi, Reed, '10] to solve Odd Cycle Transversal.

Bipartite tree decomposition

Not new

Bipartite treewidth was already implicitly used in previous papers like [Kawarabayashi, Reed, '10] to solve ODD CYCLE TRANSVERSAL.

.... and other people are currently working on it like [Campbell, Gollin, Hendrey, Wiederrecht, '23$^+$].

Bipartite tree decomposition

Not new

Bipartite treewidth was already implicitly used in previous papers like [Kawarabayashi, Reed, '10] to solve Odd Cycle Transversal.

.... and other people are currently working on it like [Campbell, Gollin, Hendrey, Wiederrecht, '23$^+$].

In particular, they show the following:

Theorem [Campbell, Gollin, Hendrey, Wiederrecht, '23$^+$]: There is an algorithm that, given a graph G with $\mathrm{btw}(G) \leq k$, outputs a bipartite tree decomposition of G of width at most $f(k)$ in time $g(k) \cdot n^4 \log n$.

Bipartite tree decomposition

Not new

Bipartite treewidth was already implicitly used in previous papers like [Kawarabayashi, Reed, '10] to solve ODD CYCLE TRANSVERSAL.

.... and other people are currently working on it like [Campbell, Gollin, Hendrey, Wiederrecht, '23$^+$].

In particular, they show the following:

Theorem [Campbell, Gollin, Hendrey, Wiederrecht, '23$^+$]: There is an algorithm that, given a graph G with $\mathrm{btw}(G) \leq k$, outputs a bipartite tree decomposition of G of width at most $f(k)$ in time $g(k) \cdot n^4 \log n$.

What is the complexity of problems parameterized by btw?

Problems parameterized by btw

Results

H-SUBGRAPH-COVER: find a set $S \subseteq V(G)$ of minimum size s.t. $G \setminus S$ does not contains H as a subgraph.

Problem	Complexity	Constraints on H/Running time
H-MINOR-COVER	para-NP-complete, $k = 0$	$P_3 \subseteq H$
H(-INDUCED)-SUBGRAPH /ODD-MINOR-COVER		$H \in \mathcal{B}$, $P_3 \subseteq H$
H-MINOR-PACKING		H 2-cc, $\|V(H)\| \geq 3$
H(-INDUCED)-SUBGRAPH-PACKING /ODD-MINOR/SCATTERED-PACKING		$H \in \mathcal{B}$ 2-cc, $\|V(H)\| \geq 3$
3-COLORING	para-NP-complete, $k = 3$	
K_t-SUBGRAPH-COVER	FPT	$\mathcal{O}(2^k \cdot (k^t \cdot (n+m) + m\sqrt{n}))$
WEIGHTED INDEPENDENT SET		$\mathcal{O}(2^k \cdot (k \cdot (k+n) + n \cdot m))$
ODD CYCLE TRANSVERSAL		$\mathcal{O}(3^k \cdot k \cdot n \cdot (m + k^2))$
MAXIMUM WEIGHTED CUT		$\mathcal{O}(2^k \cdot (k \cdot (k+n) + n^{\mathcal{O}(1)}))$
H(-INDUCED)-SUBGRAPH /ODD-MINOR/SCATTERED-PACKING	XP	$H \notin \mathcal{B}$ 2-cc $n^{\mathcal{O}(k)}$

$\mathcal{B}$ = bipartite, $P_3 \subseteq H$= P_3 is a subgraph of H, 2-cc = 2-connected

k = bipartite treewidth

Dynamic programming on bipartite tree decompositions

Difficulties

There are two main difficulties:

Dynamic programming on bipartite tree decompositions

Difficulties

There are two main difficulties:

1. the size of a bag is unbounded

Dynamic programming on bipartite tree decompositions

Difficulties

There are two main difficulties:

1. the size of a bag is unbounded
2. the number of children of a bag is unbounded: there is no notion of "nice" bipartite tree decomposition

Dynamic programming on bipartite tree decompositions

Difficulties

There are two main difficulties:

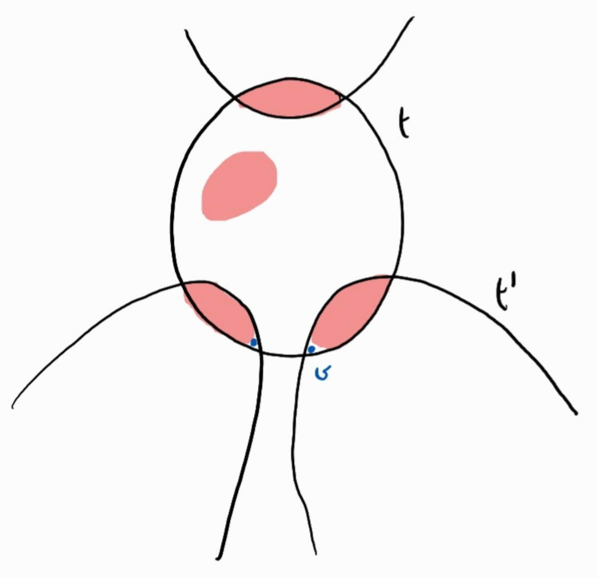
1. the size of a bag is unbounded
2. the number of children of a bag is unbounded: there is no notion of "nice" bipartite tree decomposition

Main idea: guess what happens on each "apex" vertex of the bag, and reduce each child to an equivalent smaller instance.

Sketch for WEIGHTED INDEPENDENT SET

Dynamic programming

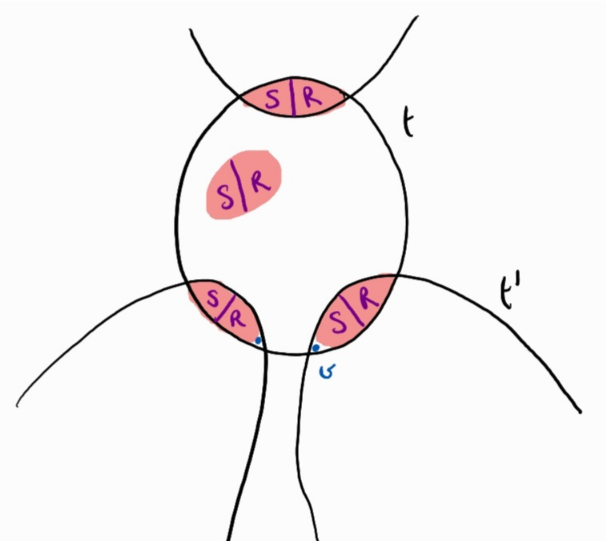
Rooted bipartite tree decomposition with "apex" vertices of bag t



Sketch for WEIGHTED INDEPENDENT SET

Dynamic programming

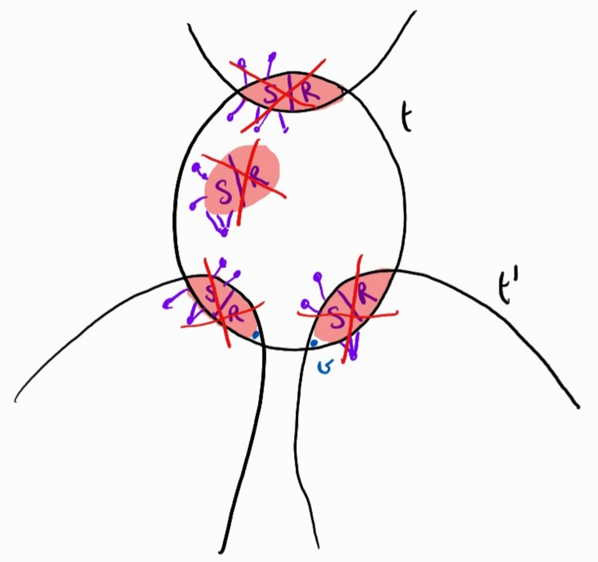
Guess the apex vertices the belong to the solution (S) and those that don't (R).



Sketch for WEIGHTED INDEPENDENT SET

Dynamic programming

Remove R, S, and the neighborhood of S.

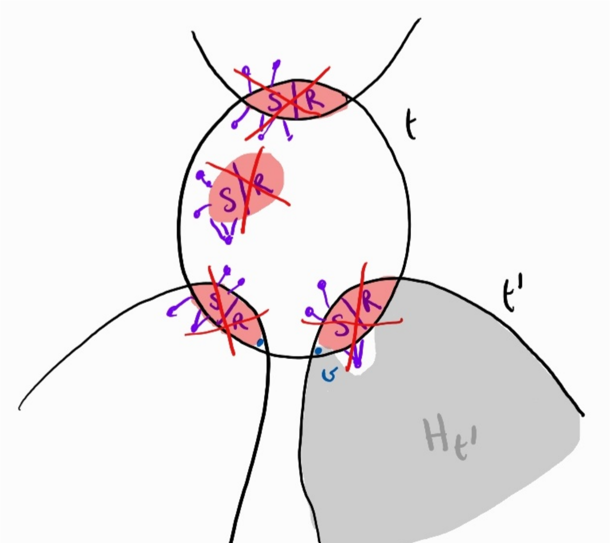


Sketch for Weighted Independent Set

Dynamic programming

w_v = weight of an optimal solution on $H_{t'}$ containing v.

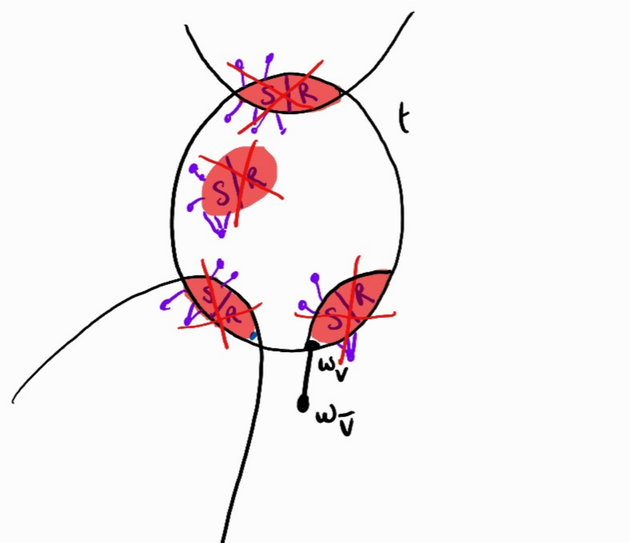
$w_{\bar{v}}$ = weight of an optimal solution on $H_{t'}$ not containing v.



Sketch for WEIGHTED INDEPENDENT SET

Dynamic programming

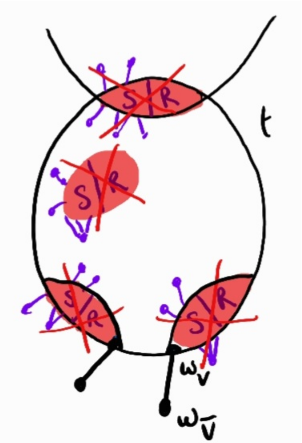
w_v = weight of the optimal solution on $H_{t'}$ containing v.
$w_{\bar{v}}$ = weight of the optimal solution on $H_{t'}$ not containing v.



Sketch for WEIGHTED INDEPENDENT SET

Dynamic programming

Solve the problem on the new bipartite bag t.



Results

Problem	Complexity	Constraints on H/Running time
H-Minor-Cover H(-Induced)-Subgraph /Odd-Minor-Cover H-Minor-Packing H(-Induced)-Subgraph-Packing /Odd-Minor/Scattered-Packing	para-NP-complete, $k = 0$	$P_3 \subseteq H$ $H \in \mathcal{B}$, $P_3 \subseteq H$ H 2-cc, $\lvert V(H)\rvert \geq 3$ $H \in \mathcal{B}$ 2-cc, $\lvert V(H)\rvert \geq 3$
3-Coloring	para-NP-complete, $k = 3$	
K_t-Subgraph-Cover Weighted Independent Set Odd Cycle Transversal Maximum Weighted Cut	FPT	$\mathcal{O}(2^k \cdot (k^t \cdot (n+m) + m\sqrt{n}))$ $\mathcal{O}(2^k \cdot (k \cdot (k+n) + n \cdot m))$ $\mathcal{O}(3^k \cdot k \cdot n \cdot (m + k^2))$ $\mathcal{O}(2^k \cdot (k \cdot (k+n) + n^{\mathcal{O}(1)}))$
H(-Induced)-Subgraph /Odd-Minor/Scattered-Packing	XP	$H \notin \mathcal{B}$ 2-cc $n^{\mathcal{O}(k)}$

$\mathcal{B}$ = bipartite, $P_3 \subseteq H$= P_3 is a subgraph of H, 2-cc = 2-connected

k = bipartite treewidth

Results

Problem	Complexity	Constraints on H/Running time
H-Minor-Cover H(-Induced)-Subgraph /Odd-Minor-Cover H-Minor-Packing H(-Induced)-Subgraph-Packing /Odd-Minor/Scattered-Packing	para-NP-complete, $k = 0$	$P_3 \subseteq H$ $H \in \mathcal{B}$, $P_3 \subseteq H$ H 2-cc, $\|V(H)\| \geq 3$ $H \in \mathcal{B}$ 2-cc, $\|V(H)\| \geq 3$
3-Coloring	para-NP-complete, $k = 3$	
K_t-Subgraph-Cover Weighted Independent Set Odd Cycle Transversal Maximum Weighted Cut	FPT	$\mathcal{O}(2^k \cdot (k^t \cdot (n+m) + m\sqrt{n}))$ $\mathcal{O}(2^k \cdot (k \cdot (k+n) + n \cdot m))$ $\mathcal{O}(3^k \cdot k \cdot n \cdot (m + k^2))$ $\mathcal{O}(2^k \cdot (k \cdot (k+n) + n^{\mathcal{O}(1)}))$
H(-Induced)-Subgraph /Odd-Minor/Scattered-Packing	XP	$H \notin \mathcal{B}$ 2-cc $n^{\mathcal{O}(k)}$

$\mathcal{B}$ = bipartite, $P_3 \subseteq H$= P_3 is a subgraph of H, 2-cc = 2-connected

k = bipartite treewidth

Open questions

- H-Subgraph/Odd-Minor-Packing FPT when H is 2-connected non-bipartite?

Open questions

- H-Subgraph/Odd-Minor-Packing FPT when H is 2-connected non-bipartite? XP when H is not bipartite and not 2-connected? First open case: $H =$ paw.

Open questions

- H-Subgraph/Odd-Minor-Packing FPT when H is 2-connected non-bipartite? XP when H is not bipartite and not 2-connected? First open case: $H =$ paw.
- H-Subgraph-Cover when H is not bipartite and is not a clique? First open cases: $H = C_5$, or $H =$ paw.

Open questions

- H-Subgraph/Odd-Minor-Packing FPT when H is 2-connected non-bipartite? XP when H is not bipartite and not 2-connected? First open case: $H =$ paw.
- H-Subgraph-Cover when H is not bipartite and is not a clique? First open cases: $H = C_5$, or $H =$ paw.
- Can we use bipartite treewidth to deal with problems related to odd-minors? Currently working on it.

Open questions

- H-Subgraph/Odd-Minor-Packing FPT when H is 2-connected non-bipartite? XP when H is not bipartite and not 2-connected? First open case: $H =$ paw.
- H-Subgraph-Cover when H is not bipartite and is not a clique? First open cases: $H = C_5$, or $H =$ paw.
- Can we use bipartite treewidth to deal with problems related to odd-minors? Currently working on it.

Thank you!