Between FO and MSO

Michał Pilipczuk
University of Warsaw

3rd Workshop on Logic, Graphs, and Algorithms,
LoGAlg 2025

Vienna, Austria November 19th, 2025

European Research Council
Established by the European Commission

Supported by ERC project BOBR,
ref. no. 948057.

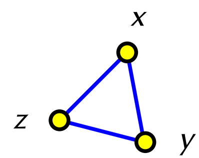
FO and MSO

FO and MSO

First-order logic FO: quantify over single vertices, verify adjacency.

$$\mathrm{triangle}(x, y, z) = \mathrm{adj}(x, y) \wedge \mathrm{adj}(y, z) \wedge \mathrm{adj}(z, x).$$

$$\mathrm{triangleFree} = \forall x. \forall y. \forall z.\ \neg \mathrm{triangle}(x, y, z).$$

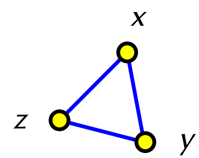


FO and MSO

First-order logic FO: quantify over single vertices, verify adjacency.

$$\mathrm{triangle}(x, y, z) = \mathrm{adj}(x, y) \wedge \mathrm{adj}(y, z) \wedge \mathrm{adj}(z, x).$$

$$\mathrm{triangleFree} = \forall x.\forall y.\forall z.\ \neg\mathrm{triangle}(x, y, z).$$



Monadic second-order logic MSO:

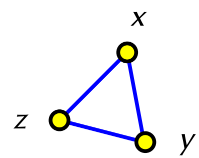
- MSO_1: quantification over subsets of $V(G)$ (3-colorability)
- MSO_2: quantification over subsets of $V(G)$ and $E(G)$ (hamiltonicity)

FO and MSO

First-order logic FO: quantify over single vertices, verify adjacency.

$$\text{triangle}(x, y, z) = \text{adj}(x, y) \wedge \text{adj}(y, z) \wedge \text{adj}(z, x).$$

$$\text{triangleFree} = \forall x.\forall y.\forall z.\ \neg\text{triangle}(x, y, z).$$



Monadic second-order logic MSO:

– MSO_1: quantification over subsets of $V(G)$ (3-colorability)

– MSO_2: quantification over subsets of $V(G)$ and $E(G)$ (hamiltonicity)

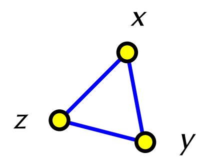
Relational structures: MSO allows quant. over subsets of the universe.

FO and MSO

First-order logic FO: quantify over single vertices, verify adjacency.

$$\text{triangle}(x, y, z) = \text{adj}(x, y) \wedge \text{adj}(y, z) \wedge \text{adj}(z, x).$$

$$\text{triangleFree} = \forall x. \forall y. \forall z.\ \neg \text{triangle}(x, y, z).$$



Monadic second-order logic MSO:

– MSO_1: quantification over subsets of $V(G)$ (3-colorability)

– MSO_2: quantification over subsets of $V(G)$ and $E(G)$ (hamiltonicity)

Relational structures: MSO allows quant. over subsets of the universe.

Two different **encodings** of graphs:

MSO_1: universe = $V(G)$, relation $\text{adj}(u, v)$.

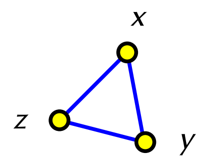
MSO_2: universe = $V(G) \uplus E(G)$ marked with V, E, relation $\text{inc}(u, e)$.

FO and MSO

First-order logic FO: quantify over single vertices, verify adjacency.

$$\text{triangle}(x, y, z) = \text{adj}(x, y) \wedge \text{adj}(y, z) \wedge \text{adj}(z, x).$$

$$\text{triangleFree} = \forall x.\forall y.\forall z.\ \neg\text{triangle}(x, y, z).$$

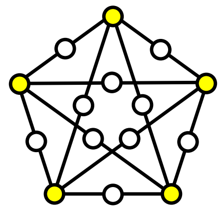


Monadic second-order logic MSO:

- MSO_1: quantification over subsets of $V(G)$ (3-colorability)
- MSO_2: quantification over subsets of $V(G)$ and $E(G)$ (hamiltonicity)

Relational structures: MSO allows quant. over subsets of the universe.

Two different **encodings** of graphs:



MSO_1: universe = $V(G)$, relation $\text{adj}(u, v)$.

MSO_2: universe = $V(G) \uplus E(G)$ marked with V, E, relation $\text{inc}(u, e)$.

MSO_2 boils down to MSO_1 on the **1-subdivision** of G.

Correspondence

Correspondence

Want: Understand the correspondence

Logic $\mathcal{L}$ $\longleftrightarrow$ Graph classes $\mathscr{C}$ **tamed** wrt $\mathcal{L}$

Correspondence

Want: Understand the correspondence

Logic $\mathcal{L}$ $\longleftrightarrow$ Graph classes $\mathscr{C}$ **tamed** wrt $\mathcal{L}$

Proposition: $\mathcal{L}$ is **tamed** on $\mathscr{C}$ $\Leftrightarrow$

Problems expressible in $\mathcal{L}$ can be solved **efficiently** on $\mathscr{C}$.

Correspondence

Want: Understand the correspondence

Logic $\mathcal{L}$ $\longleftrightarrow$ Graph classes $\mathscr{C}$ **tamed** wrt $\mathcal{L}$

Proposition: $\mathcal{L}$ is **tamed** on $\mathscr{C}$ $\Leftrightarrow$

Problems expressible in $\mathcal{L}$ can be solved **efficiently** on $\mathscr{C}$.

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Correspondence

Want: Understand the correspondence

Logic $\mathcal{L}$ $\longleftrightarrow$ Graph classes $\mathscr{C}$ **tamed** wrt $\mathcal{L}$

Proposition: $\mathcal{L}$ is **tamed** on $\mathscr{C}$ $\Leftrightarrow$

Problems expressible in $\mathcal{L}$ can be solved **efficiently** on $\mathscr{C}$.

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Note: This is a **meta-problem**.

Correspondence

Want: Understand the correspondence

Logic $\mathcal{L}$ $\longleftrightarrow$ Graph classes $\mathscr{C}$ **tamed** wrt $\mathcal{L}$

Proposition: $\mathcal{L}$ is **tamed** on $\mathscr{C}$ $\Leftrightarrow$

Problems expressible in $\mathcal{L}$ can be solved **efficiently** on $\mathscr{C}$.

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Note: This is a **meta-problem**.

Note: This is a **parameterized** problem, with φ being the parameter.

Complexity of model-checking

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Complexity of model-checking

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Notions of tractability:

- XP: running time $|G|^{f(\varphi)} = |G|^{\mathcal{O}_\varphi(1)}$. (slice-wise polynomial)
- FPT: running time $f(\varphi) \cdot |G|^c = \mathcal{O}_\varphi(|G|^c)$, for a constant c. (fixed-parameter tractable)

Complexity of model-checking

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Notions of tractability:

- XP: running time $|G|^{f(\varphi)} = |G|^{\mathcal{O}_\varphi(1)}$. (slice-wise polynomial)
- FPT: running time $f(\varphi) \cdot |G|^c = \mathcal{O}_\varphi(|G|^c)$, for a constant c. (fixed-parameter tractable)

General graphs:

- FO: brute-force runtime $|G|^{\mathcal{O}(\|\varphi\|)}$, but $f(\varphi) \cdot |G|^c$ unlikely. (AW[$\star$]-hard)
- MSO_1: 3-coloring planar graphs already NP-hard. $\rightsquigarrow$ No XP algorithm.

Complexity of model-checking

Model-checking problem for $\mathcal{L}$

Given a graph G and a sentence $\varphi \in \mathcal{L}$, is φ true in G?

Notions of tractability:

- XP: running time $|G|^{f(\varphi)} = |G|^{\mathcal{O}_\varphi(1)}$. (slice-wise polynomial)
- FPT: running time $f(\varphi) \cdot |G|^c = \mathcal{O}_\varphi(|G|^c)$, for a constant c. (fixed-parameter tractable)

General graphs:

- FO: brute-force runtime $|G|^{\mathcal{O}(\|\varphi\|)}$, but $f(\varphi) \cdot |G|^c$ unlikely. (AW[$\star$]-hard)
- MSO_1: 3-coloring planar graphs already NP-hard. $\rightsquigarrow$ No XP algorithm.

Q: On what graph classes is model-checking FO / MSO_1 / MSO_2 FPT?

Fixed-parameter model-checking

Fixed-parameter model-checking

Theorem [Courcelle '90]

If $\mathscr{C}$ has **bounded treewidth**,

then model-checking MSO_2 on $\mathscr{C}$ in time $\mathcal{O}_{\varphi}(|G|)$.

Fixed-parameter model-checking

Theorem [Courcelle '90]

If $\mathscr{C}$ has **bounded treewidth**,

then model-checking MSO_2 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [Courcelle, Makowsky, Rotics '00]

If $\mathscr{C}$ has **bounded cliquewidth**,

then model-checking MSO_1 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^2)$.

Fixed-parameter model-checking

Theorem [Courcelle '90]

If $\mathscr{C}$ has **bounded treewidth**,

then model-checking MSO_2 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [Courcelle, Makowsky, Rotics '00]

If $\mathscr{C}$ has **bounded cliquewidth**,

then model-checking MSO_1 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^2)$.

Idea: **Treewidth** and **cliquewidth** are notions of **tree-likeness** $\rightsquigarrow$

Run a bottom-up dynamic programming / tree automaton.

Fixed-parameter model-checking

Theorem [Courcelle '90]

If $\mathscr{C}$ has **bounded treewidth**,

then model-checking MSO_2 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [Courcelle, Makowsky, Rotics '00]

If $\mathscr{C}$ has **bounded cliquewidth**,

then model-checking MSO_1 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^2)$.

Idea: **Treewidth** and **cliquewidth** are notions of **tree-likeness** $\rightsquigarrow$

Run a bottom-up dynamic programming / tree automaton.

Theorem [Grohe, Kreutzer, Siebertz '14]

If $\mathscr{C}$ is **nowhere dense**,

then model-checking FO on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^{1+\varepsilon})$, for any $\varepsilon > 0$.

Fixed-parameter model-checking

Theorem [Courcelle '90]

If $\mathscr{C}$ has **bounded treewidth**,

then model-checking MSO_2 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [Courcelle, Makowsky, Rotics '00]

If $\mathscr{C}$ has **bounded cliquewidth**,

then model-checking MSO_1 on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^2)$.

Idea: **Treewidth** and **cliquewidth** are notions of **tree-likeness** $\rightsquigarrow$

Run a bottom-up dynamic programming / tree automaton.

Theorem [Grohe, Kreutzer, Siebertz '14]

If $\mathscr{C}$ is **nowhere dense**,

then model-checking FO on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|^{1+\varepsilon})$, for any $\varepsilon > 0$.

Idea: **Nowhere denseness** defined by exclusion of **local** obstructions $\rightsquigarrow$

Exploit **local decompositions** and **locality** of FO.

Correspondence for MSO

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are
the **limit of tractability** of MSO_1/MSO_2.

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathrm{MSO}_1/\mathrm{MSO}_2$.

Do we really understand this?

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are the **limit of tractability** of $\mathsf{MSO}_1/\mathsf{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices}\colon\ G \text{ a graph}\}$

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathsf{MSO}_1/\mathsf{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices} \colon G \text{ a graph}\}$ $\Rightarrow$ need some closure

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathsf{MSO}_1/\mathsf{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices}: G \text{ a graph}\}$ $\Rightarrow$ need some closure

Obs: $\mathscr{C}$ is **minor-closed** and of unbounded treewidth $\Rightarrow$

$\mathscr{C}$ contains all planar graphs $\Rightarrow$

model-checking MSO_1 is para-NP-hard on $\mathscr{C}$.

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathrm{MSO}_1/\mathrm{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices}\colon G \text{ a graph}\}$ $\Rightarrow$ need some closure

Obs: $\mathscr{C}$ is **minor-closed** and of unbounded treewidth $\Rightarrow$

$\mathscr{C}$ contains all planar graphs $\Rightarrow$

model-checking MSO_1 is para-NP-hard on $\mathscr{C}$.

Theorem [Kreutzer, Tazari '10]

If $\mathscr{C}$ is **coloring-closed** and **treewidth** in $\mathscr{C}$ is **not poly-log bounded**,

then model-checking MSO_2 on $\mathscr{C}$ is not in XP (under assumptions).

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathsf{MSO}_1/\mathsf{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices}\colon\ G \text{ a graph}\}$ $\quad\Rightarrow$ need some closure

Obs: $\mathscr{C}$ is **minor-closed** and of unbounded treewidth $\Rightarrow$

$\mathscr{C}$ contains all planar graphs $\Rightarrow$

model-checking MSO_1 is para-NP-hard on $\mathscr{C}$.

Theorem [Kreutzer, Tazari '10]

If $\mathscr{C}$ is **coloring-closed** and **treewidth** in $\mathscr{C}$ is **not poly-log bounded**,

then model-checking MSO_2 on $\mathscr{C}$ is not in XP (under assumptions).

Q: Do we know such a theorem for cliquewidth?

Correspondence for MSO

Folklore: Bounded **treewidth**/**cliquewidth** are

the **limit of tractability** of $\mathrm{MSO}_1/\mathrm{MSO}_2$.

Do we really understand this?

Ex: $\mathscr{C} = \{G + 2^{2^{|G|}} \text{ isolated vertices}: G \text{ a graph}\}$ $\Rightarrow$ need some closure

Obs: $\mathscr{C}$ is **minor-closed** and of unbounded treewidth $\Rightarrow$

$\mathscr{C}$ contains all planar graphs $\Rightarrow$

model-checking MSO_1 is para-NP-hard on $\mathscr{C}$.

Theorem [Kreutzer, Tazari '10]

If $\mathscr{C}$ is **coloring-closed** and **treewidth** in $\mathscr{C}$ is **not poly-log bounded**,

then model-checking MSO_2 on $\mathscr{C}$ is not in XP (under assumptions).

Q: Do we know such a theorem for cliquewidth?

Theorem

If $\mathscr{C}$ is **subgraph-closed** and **not nowhere dense**,

then model-checking FO on $\mathscr{C}$ is as hard as on general graphs.

Encoding graphs in graphs

Proposition: Understand **tameness** through **expressive power**, instead of **algorithmic tractability**.

Encoding graphs in graphs

Proposition: Understand **tameness** through **expressive power**, instead of **algorithmic tractability**.

Need: Notion of encoding graphs in graphs in logic.

Encoding graphs in graphs

Proposition: Understand **tameness** through **expressive power**, instead of **algorithmic tractability**.

Need: Notion of encoding graphs in graphs in logic.

$\rightsquigarrow$ **transductions** and **transducibility**.

Encoding graphs in graphs

Proposition: Understand **tameness** through **expressive power**, instead of **algorithmic tractability**.

Need: Notion of encoding graphs in graphs in logic.

$\rightsquigarrow$ **transductions** and **transducibility**.

$\mathscr{D}$ is **$\mathcal{L}$-transducible** from $\mathscr{C}$ $\Leftrightarrow$
every $H \in \mathscr{D}$ can be **encoded** in some vertex-colored $G \in \mathscr{C}$ using a fixed formula $\varphi(x, y) \in \mathcal{L}$.

Encoding graphs in graphs

Proposition: Understand **tameness** through **expressive power**,
instead of **algorithmic tractability**.

Need: Notion of encoding graphs in graphs in logic.

$\rightsquigarrow$ **transductions** and **transducibility**.

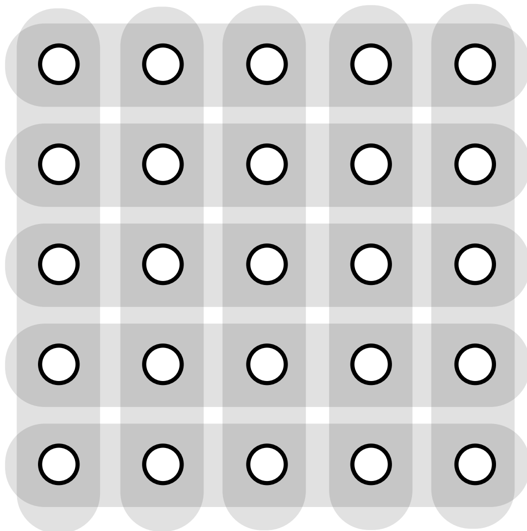
$\mathscr{D}$ is **$\mathcal{L}$-transducible** from $\mathscr{C}$ $\Leftrightarrow$
every $H \in \mathscr{D}$ can be **encoded** in some vertex-colored $G \in \mathscr{C}$
using a fixed formula $\varphi(x, y) \in \mathcal{L}$.

Formally: There is a 1-dimensional $\mathcal{L}$-interpretation
from (colorings of $\mathscr{C}$) onto $\mathscr{D}$.

Transductions: Example

Transductions: Example

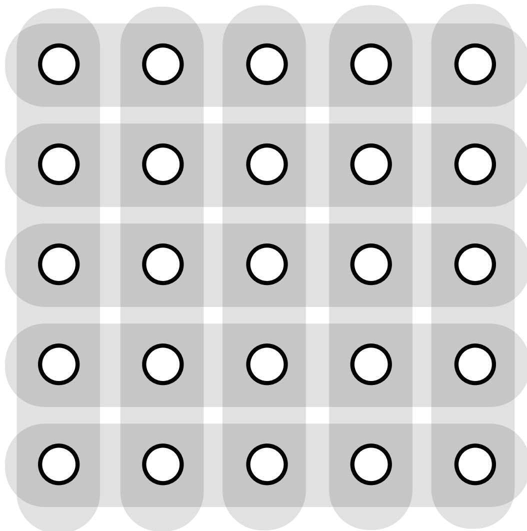
$\mathscr{C} := \{\textit{rook graphs}\} =$ grids with rows and columns made into cliques.



Transductions: Example

$\mathscr{C} :=$ {*rook graphs*} = grids with rows and columns made into cliques.

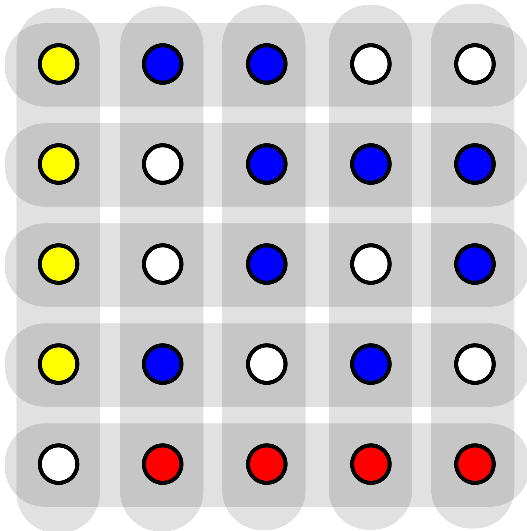
Claim: {all bipartite graphs} is FO-transducible from $\mathscr{C}$.



Transductions: Example

$\mathscr{C} :=$ {*rook graphs*} = grids with rows and columns made into cliques.

Claim: {all bipartite graphs} is FO-transducible from $\mathscr{C}$.

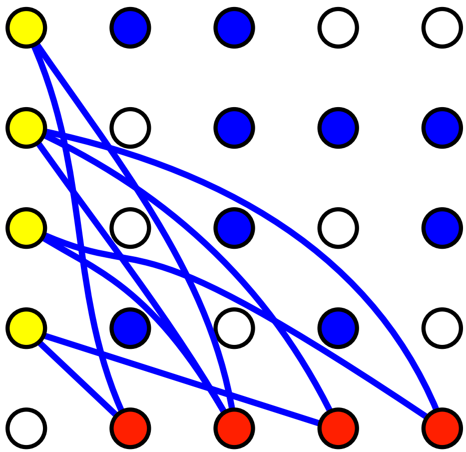


First row & column using **red** & **yellow**, the adjacency matrix using **blue**.

Transductions: Example

$\mathscr{C} :=$ {*rook graphs*} = grids with rows and columns made into cliques.

Claim: {all bipartite graphs} is FO-transducible from $\mathscr{C}$.



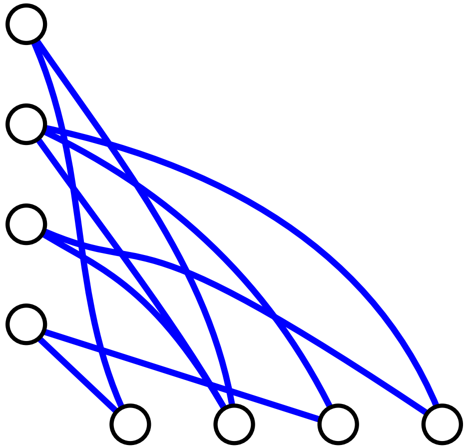
First row & column using **red** & **yellow**, the adjacency matrix using **blue**.

$\varphi(x, y) =$ **red**(x) and **yellow**(y) and x, y have a common **blue** neighbor.

Transductions: Example

$\mathscr{C} :=$ {*rook graphs*} $=$ grids with rows and columns made into cliques.

Claim: {all bipartite graphs} is FO-transducible from $\mathscr{C}$.



First row & column using **red** & **yellow**, the adjacency matrix using **blue**.

$\varphi(x, y) =$ **red**(x) and **yellow**(y) and x, y have a common **blue** neighbor.

Drop all **blue** and **white** vertices.

Transductions, formally

Transductions, formally

$\mathcal{L}$-**transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Transductions, formally

$\mathcal{L}$-transduction: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

Transductions, formally

$\mathcal{L}$**-transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

– **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$

Transductions, formally

$\mathcal{L}$-**transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$

Transductions, formally

$\mathcal{L}$-**transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $H \subseteq_{\mathsf{i}} \varphi(\widehat{G})$

Transductions, formally

$\mathcal{L}$-transduction: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $H \subseteq_{\mathrm{i}} \varphi(\widehat{G})$

Note: A transduction is a **nondeterministic** mechanism.

Transductions, formally

$\mathcal{L}$**-transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $H \subseteq_{\mathsf{i}} \varphi(\widehat{G})$

Note: A transduction is a **nondeterministic** mechanism.

$$\mathsf{T}(G) := \text{all possible outputs of } \mathsf{T} \text{ on } G.$$

Transductions, formally

$\mathcal{L}$**-transduction**: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $H \subseteq_{\mathrm{i}} \varphi(\widehat{G})$

Note: A transduction is a **nondeterministic** mechanism.

$$\mathsf{T}(G) := \text{all possible outputs of } \mathsf{T} \text{ on } G.$$

For a class $\mathscr{C}$ and a transduction T, define

$$\mathsf{T}(\mathscr{C}) := \bigcup_{G \in \mathscr{C}} \mathsf{T}(G)$$

Transductions, formally

$\mathcal{L}$-**transduction**: a set unary predicates C, a formula $\varphi(x,y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $\quad G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u,v)\}$. $\quad \widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $\quad H \subseteq_{\mathsf{i}} \varphi(\widehat{G})$

Note: A transduction is a **nondeterministic** mechanism.

$$\mathsf{T}(G) \coloneqq \text{all possible outputs of } \mathsf{T} \text{ on } G.$$

For a class $\mathscr{C}$ and a transduction T, define

$$\mathsf{T}(\mathscr{C}) \coloneqq \bigcup_{G \in \mathscr{C}} \mathsf{T}(G)$$

Call $\mathscr{D}$ $\mathcal{L}$-**transducible** from $\mathscr{C}$ if there is a $\mathcal{L}$-transduction T such that

$$\mathscr{D} \subseteq \mathsf{T}(\mathscr{C}).$$

Transductions, formally

$\mathcal{L}$-transduction: a set unary predicates C, a formula $\varphi(x, y) \in \mathcal{L}$.

Applying $\mathsf{T} = (C, \varphi)$ to G:

- **Color** vertices of G using C. $G \rightsquigarrow \widehat{G}$
- **Replace** the adjacency relation with $\{uv \mid G \models \varphi(u, v)\}$. $\widehat{G} \rightsquigarrow \varphi(\widehat{G})$
- Output any **induced subgraph** of $\varphi(\widehat{G})$ and drop colors. $H \subseteq_{\mathsf{i}} \varphi(\widehat{G})$

Note: A transduction is a **nondeterministic** mechanism.

$$\mathsf{T}(G) := \text{all possible outputs of } \mathsf{T} \text{ on } G.$$

For a class $\mathscr{C}$ and a transduction T, define

$$\mathsf{T}(\mathscr{C}) := \bigcup_{G \in \mathscr{C}} \mathsf{T}(G)$$

Call $\mathscr{D}$ **$\mathcal{L}$-transducible** from $\mathscr{C}$ if there is a $\mathcal{L}$-transduction T such that

$$\mathscr{D} \subseteq \mathsf{T}(\mathscr{C}).$$

Obs: $\mathcal{L}$-transductions closed under composition (for reasonable $\mathcal{L}$) $\Rightarrow$

$\mathcal{L}$-transducibility is a **quasi-order** on graph classes.

Monadic dependence

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: $\{\text{rook graphs}\}$ is **not** mon dependent wrt FO.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: {rook graphs} is **not** mon dependent wrt FO.

Theorem

$\mathscr{C}$ is **mon dependent** wrt MSO_2 $\Leftrightarrow$ $\mathscr{C}$ has **bounded treewidth**

$\mathscr{C}$ is **mon dependent** wrt MSO_1 $\Leftrightarrow$[1] $\mathscr{C}$ has **bounded cliquewidth**

[1]: Counting mod 2 needed.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: {rook graphs} is **not** mon dependent wrt FO.

Theorem

$\mathscr{C}$ is **mon dependent** wrt MSO_2 $\Leftrightarrow$ $\mathscr{C}$ has **bounded treewidth**

$\mathscr{C}$ is **mon dependent** wrt MSO_1 $\Leftrightarrow$[1] $\mathscr{C}$ has **bounded cliquewidth**

Note: MSO_2 transduction = Starting with the incidence encoding.

[1]: Counting mod 2 needed.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: {rook graphs} is **not** mon dependent wrt FO.

Theorem

$\mathscr{C}$ is **mon dependent** wrt MSO_2 $\Leftrightarrow$ $\mathscr{C}$ has **bounded treewidth**

$\mathscr{C}$ is **mon dependent** wrt MSO_1 $\Leftrightarrow$[1] $\mathscr{C}$ has **bounded cliquewidth**

Note: MSO_2 transduction = Starting with the incidence encoding.

Proof sketch:

[1]: Counting mod 2 needed.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is } \textbf{not } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: $\{\text{rook graphs}\}$ is **not** mon dependent wrt FO.

Theorem

$\mathscr{C}$ is **mon dependent** wrt MSO_2 $\Leftrightarrow$ $\mathscr{C}$ has **bounded treewidth**

$\mathscr{C}$ is **mon dependent** wrt MSO_1 $\Leftrightarrow$[1] $\mathscr{C}$ has **bounded cliquewidth**

Note: MSO_2 transduction = Starting with the incidence encoding.

Proof sketch:

$\Rightarrow$: **Grids** MSO-transduce all graphs, use **Grid Theorems** for **tw**/**cw**.

[1]: Counting mod 2 needed.

Monadic dependence

Def: $\mathscr{C}$ is **monadically dependent** wrt $\mathcal{L}$ if

$$\{\text{all graphs}\} \text{ is \textbf{not} } \mathcal{L}\text{-transducible from } \mathscr{C}.$$

Intuition: Weakest possible restriction, cannot $\mathcal{L}$-encode all graphs in $\mathscr{C}$.

Ex: $\{$rook graphs$\}$ is **not** mon dependent wrt FO.

Theorem

$\mathscr{C}$ is **mon dependent** wrt MSO_2 $\Leftrightarrow$ $\mathscr{C}$ has **bounded treewidth**

$\mathscr{C}$ is **mon dependent** wrt MSO_1 $\Leftrightarrow$[1] $\mathscr{C}$ has **bounded cliquewidth**

Note: MSO_2 transduction = Starting with the incidence encoding.

Proof sketch:

$\Rightarrow$: **Grids** MSO-transduce all graphs, use **Grid Theorems** for **tw**/**cw**.

$\Leftarrow$: MSO transductions preserve bounded **cliquewidth**. $\square$

[1]: Counting mod 2 needed.

Monadic dependence and complexity

Monadic dependence and complexity

Theorem [Adler and Adler '14]

Suppose $\mathscr{C}$ is **subgraph-closed**. Then

$\mathscr{C}$ is **nowhere dense** $\Leftrightarrow$ $\mathscr{C}$ is **mon dependent** wrt FO.

Monadic dependence and complexity

Theorem [Adler and Adler '14]

Suppose $\mathscr{C}$ is **subgraph-closed**. Then

$\mathscr{C}$ is **nowhere dense** $\Leftrightarrow$ $\mathscr{C}$ is **mon dependent** wrt FO.

Conjecture

If $\mathscr{C}$ is **mon dependent** wrt FO, then FO model-checking on $\mathscr{C}$ is FPT.

Monadic dependence and complexity

Theorem [Adler and Adler '14]

Suppose $\mathscr{C}$ is **subgraph-closed**. Then

$\mathscr{C}$ is **nowhere dense** $\Leftrightarrow$ $\mathscr{C}$ is **mon dependent** wrt FO.

Conjecture

If $\mathscr{C}$ is **mon dependent** wrt FO, then FO model-checking on $\mathscr{C}$ is FPT.

Note: Proved for **mon stable** classes.

[Gajarský, Mählmann, McCarty, Ohlmann, P, Przybyszewski, Siebertz, Sokołowski, Toruńczyk '23]
[Dreier, Mählmann, Siebertz '23]
[Dreier, Eleftheriadis, Mählmann, McCarty, P, Toruńczyk '24]

Monadic dependence and complexity

Theorem [Adler and Adler '14]

Suppose $\mathscr{C}$ is **subgraph-closed**. Then

$\mathscr{C}$ is **nowhere dense** $\Leftrightarrow$ $\mathscr{C}$ is **mon dependent** wrt FO.

Conjecture

If $\mathscr{C}$ is **mon dependent** wrt FO, then FO model-checking on $\mathscr{C}$ is FPT.

Note: Proved for **mon stable** classes.

[Gajarský, Mählmann, McCarty, Ohlmann, P, Przybyszewski, Siebertz, Sokołowski, Toruńczyk '23]
[Dreier, Mählmann, Siebertz '23]
[Dreier, Eleftheriadis, Mählmann, McCarty, P, Toruńczyk '24]

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Meta-conjecture

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

– MSO_2 (bnd treewidth)

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

- MSO_2 (bnd treewidth)
- MSO_1 (bnd cliquewidth)

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

- MSO_2 (bnd treewidth)
- MSO_1 (bnd cliquewidth)
- FO on ordered graphs (bnd twin-width) [Bonnet et al. '22]

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

- MSO_2 (bnd treewidth)
- MSO_1 (bnd cliquewidth)
- FO on ordered graphs (bnd twin-width) [Bonnet et al. '22]
- FO on tournaments (bnd twin-width) [Geniet, Thomassé '23]

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

- MSO_2 (bnd treewidth)
- MSO_1 (bnd cliquewidth)
- FO on ordered graphs (bnd twin-width) [Bonnet et al. '22]
- FO on tournaments (bnd twin-width) [Geniet, Thomassé '23]

Reverse implication fuzzy: closure assumptions allowing coding.

Meta-conjecture

Meta-conjecture

Let $\mathcal{L}$ be a reasonable logic. Then **model-checking** $\mathcal{L}$ is FPT on every graph class $\mathscr{C}$ that is **mon dependent** wrt $\mathcal{L}$.

Conjectured for FO, **works** for:

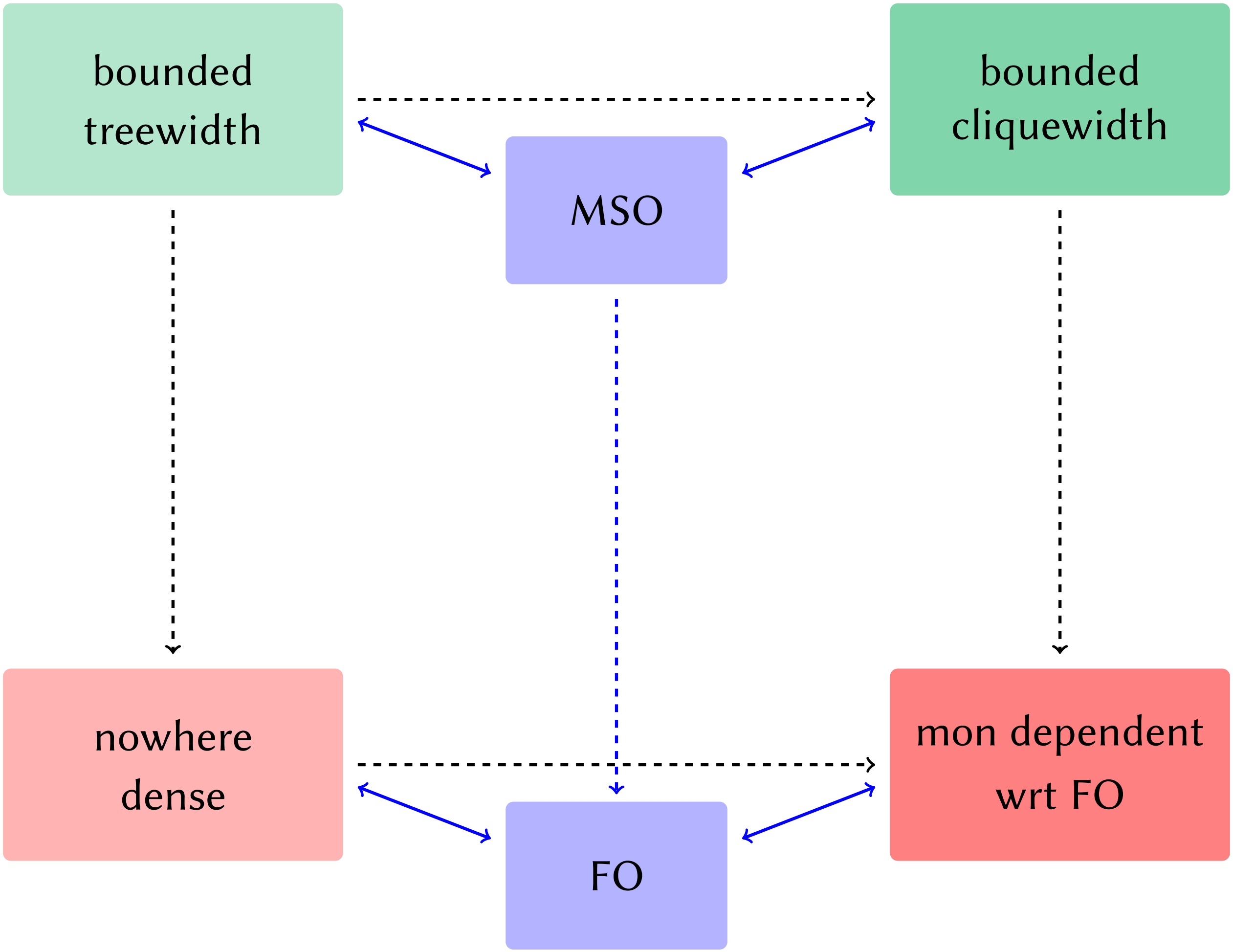
- MSO_2 (bnd treewidth)
- MSO_1 (bnd cliquewidth)
- FO on ordered graphs (bnd twin-width) [Bonnet et al. '22]
- FO on tournaments (bnd twin-width) [Geniet, Thomassé '23]

Reverse implication fuzzy: closure assumptions allowing coding.

Theorem [Dreier, Mählmann, Toruńczyk '24]

If $\mathscr{C}$ is **hereditary** and **not mon dependent**, then model-checking FO on $\mathscr{C}$ is AW[$\star$]-hard.

Big picture



bounded treewidth
bounded cliquewidth
MSO
nowhere dense
mon dependent wrt FO
FO

Between FO and MSO

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with FO $\subseteq \mathcal{L} \subseteq$ MSO.

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with FO $\subseteq \mathcal{L} \subseteq$ MSO.

$$\text{Logic } \mathcal{L} \qquad \longleftrightarrow \qquad \mathcal{P}_{\mathcal{L}} \coloneqq \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with $\mathrm{FO} \subseteq \mathcal{L} \subseteq \mathrm{MSO}$.

Logic $\mathcal{L}$ $\longleftrightarrow$ $\mathcal{P}_{\mathcal{L}} := \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$

$$\mathrm{MSO} \supseteq \mathcal{L} \supseteq \mathrm{FO}$$

$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$

$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with $\mathsf{FO} \subseteq \mathcal{L} \subseteq \mathsf{MSO}$.

$$\text{Logic } \mathcal{L} \qquad \longleftrightarrow \qquad \mathcal{P}_{\mathcal{L}} \coloneqq \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

$$\mathsf{MSO} \supseteq \mathcal{L} \supseteq \mathsf{FO}$$

$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$

$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Motivation:

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with $\mathrm{FO} \subseteq \mathcal{L} \subseteq \mathrm{MSO}$.

$$\text{Logic } \mathcal{L} \quad \longleftrightarrow \quad \mathcal{P}_{\mathcal{L}} \coloneqq \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

$$\mathrm{MSO} \supseteq \mathcal{L} \supseteq \mathrm{FO}$$

$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$

$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Motivation:

– This is just a **natural idea**.

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with $\mathsf{FO} \subseteq \mathcal{L} \subseteq \mathsf{MSO}$.

$$\text{Logic } \mathcal{L} \qquad \longleftrightarrow \qquad \mathcal{P}_{\mathcal{L}} \coloneqq \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

$$\mathsf{MSO} \supseteq \mathcal{L} \supseteq \mathsf{FO}$$
$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$
$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Motivation:

– This is just a **natural idea**.

– **Utility:** If $\mathcal{P}$ is a natural property of graph classes (e.g. minor-free), then $\mathcal{L}$ with $\mathcal{P} = \mathcal{P}_{\mathcal{L}}$ may capture the toolbox of $\mathcal{P}$.

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with FO $\subseteq \mathcal{L} \subseteq$ MSO.

$$\text{Logic } \mathcal{L} \qquad \longleftrightarrow \qquad \mathcal{P}_{\mathcal{L}} := \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

$$\text{MSO} \supseteq \mathcal{L} \supseteq \text{FO}$$

$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$

$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Motivation:

- This is just a **natural idea**.
- **Utility:** If $\mathcal{P}$ is a natural property of graph classes (e.g. minor-free), then $\mathcal{L}$ with $\mathcal{P} = \mathcal{P}_{\mathcal{L}}$ may capture the toolbox of $\mathcal{P}$.
- **Exploration:** If $\mathcal{L}$ is a natural logic, then $\mathcal{P}_{\mathcal{L}}$ is a natural property.

Between FO and MSO

Goal: Explore logics $\mathcal{L}$ with FO $\subseteq \mathcal{L} \subseteq$ MSO.

$$\text{Logic } \mathcal{L} \quad \longleftrightarrow \quad \mathcal{P}_{\mathcal{L}} := \{\mathscr{C} : \mathscr{C} \text{ is mon dependent wrt } \mathcal{L}\}$$

$$\text{MSO} \supseteq \mathcal{L} \supseteq \text{FO}$$

$$\text{bnd cliquewidth} \subseteq \mathcal{P}_{\mathcal{L}} \subseteq \text{mon dependent wrt FO}$$

$$\text{bnd treewidth} \subseteq (\mathcal{P}_{\mathcal{L}})_{\downarrow} \subseteq \text{nowhere dense}$$

Motivation:

– This is just a **natural idea**.

– **Utility:** If $\mathcal{P}$ is a natural property of graph classes (e.g. minor-free), then $\mathcal{L}$ with $\mathcal{P} = \mathcal{P}_{\mathcal{L}}$ may capture the toolbox of $\mathcal{P}$.

– **Exploration:** If $\mathcal{L}$ is a natural logic, then $\mathcal{P}_{\mathcal{L}}$ is a natural property.

Now: A thread of results exemplifying this.

Logic FO + conn

Logic FO + conn

Idea: There is no FO formula $\varphi(x, y)$ such that

$$G \models \varphi(u, v) \quad \Leftrightarrow \quad u, v \text{ are in the same connected component.}$$

Let us add this to the logic FO!

Logic FO + conn

Idea: There is no FO formula $\varphi(x, y)$ such that

$$G \models \varphi(u, v) \quad \Leftrightarrow \quad u, v \text{ are in the same connected component.}$$

Let us add this to the logic FO!

Adding just the $\text{conn}(x, y)$ predicate is not very revealing.

Logic FO + conn

Idea: There is no FO formula $\varphi(x, y)$ such that

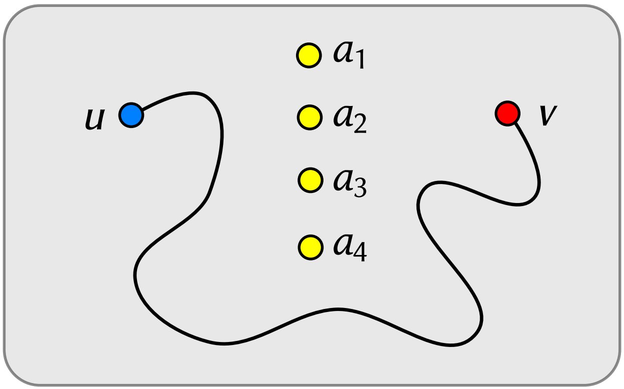
$$G \models \varphi(u, v) \quad \Leftrightarrow \quad u, v \text{ are in the same connected component.}$$

Let us add this to the logic FO!

Adding just the $\mathrm{conn}(x, y)$ predicate is not very revealing.

Def: For every $k \in \mathbb{N}$, we add the predicate $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ with

$$G \models \mathrm{conn}(u, v, a_1, \ldots, a_k) \quad \Leftrightarrow \quad u, v \text{ are connected in } G - \{a_1, \ldots, a_k\}.$$



Logic FO + conn

Idea: There is no FO formula $\varphi(x, y)$ such that

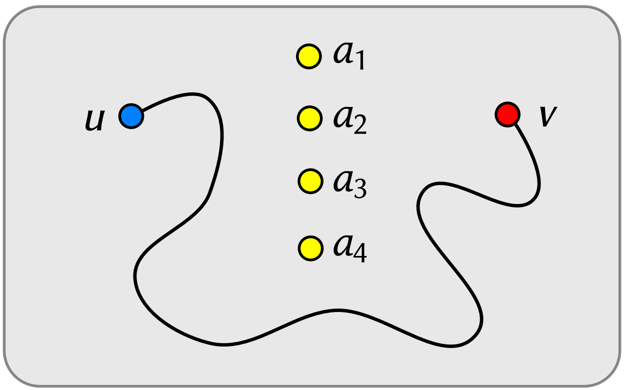
$$G \models \varphi(u, v) \quad \Leftrightarrow \quad u, v \text{ are in the same connected component.}$$

Let us add this to the logic FO!

Adding just the $\mathrm{conn}(x, y)$ predicate is not very revealing.

Def: For every $k \in \mathbb{N}$, we add the predicate $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ with

$$G \models \mathrm{conn}(u, v, a_1, \ldots, a_k) \quad \Leftrightarrow \quad u, v \text{ are connected in } G - \{a_1, \ldots, a_k\}.$$



FO + conn: FO plus conn_k for each $k \in \mathbb{N}$.

Logic FO + conn

Idea: There is no FO formula $\varphi(x, y)$ such that

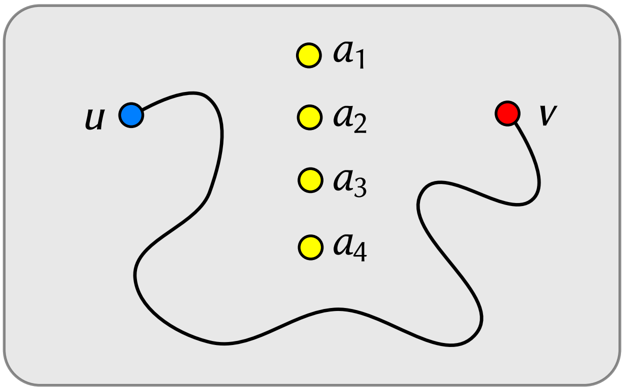
$$G \models \varphi(u, v) \quad \Leftrightarrow \quad u, v \text{ are in the same connected component.}$$

Let us add this to the logic FO!

Adding just the $\mathrm{conn}(x, y)$ predicate is not very revealing.

Def: For every $k \in \mathbb{N}$, we add the predicate $\mathrm{conn}_k(x, y, z_1, \ldots, z_k)$ with

$$G \models \mathrm{conn}(u, v, a_1, \ldots, a_k) \quad \Leftrightarrow \quad u, v \text{ are connected in } G - \{a_1, \ldots, a_k\}.$$



FO + conn: FO plus conn_k for each $k \in \mathbb{N}$.

Idea: Speak about small-size separators.

Logic FO + conn: **motivation**

Logic FO + conn: **motivation**

Introduced independently in 2021 by

Bojańczyk and by **Schirrmacher, Siebertz, and Vigny**.

Logic FO + conn: **motivation**

Introduced independently in 2021 by

Bojańczyk and by **Schirrmacher, Siebertz, and Vigny**.

Motivation of **Bojańczyk**:

– Graph languages definable in FO + conn =

Star-free expressions in the treewidth algebra.

– Schützenberger's Theorem for graphs of bounded pathwidth.

Logic FO + conn: motivation

Introduced independently in 2021 by

Bojańczyk and by **Schirrmacher, Siebertz, and Vigny**.

Motivation of **Bojańczyk**:

– Graph languages definable in FO + conn =

Star-free expressions in the treewidth algebra.

– Schützenberger's Theorem for graphs of bounded pathwidth.

– FO + conn is a natural graph analogue of FO on words with order.

Logic FO + conn: motivation

Introduced independently in 2021 by

Bojańczyk and by **Schirrmacher, Siebertz, and Vigny**.

Motivation of **Bojańczyk**:

- Graph languages definable in FO + conn =

  Star-free expressions in the treewidth algebra.
- Schützenberger's Theorem for graphs of bounded pathwidth.
- FO + conn is a natural graph analogue of FO on words with order.

Motivation of **Schirrmacher et al.**:

- Expressive power.

Logic $\mathsf{FO} + \mathsf{conn}$: **motivation**

Introduced independently in 2021 by

Bojańczyk and by **Schirrmacher, Siebertz, and Vigny**.

Motivation of **Bojańczyk**:

– Graph languages definable in $\mathsf{FO} + \mathsf{conn}$ =

Star-free expressions in the treewidth algebra.

– Schützenberger's Theorem for graphs of bounded pathwidth.

– $\mathsf{FO} + \mathsf{conn}$ is a natural graph analogue of FO on words with order.

Motivation of **Schirrmacher et al.**:

– Expressive power.

Subgraph-closed classes **tamed** wrt $\mathsf{FO} + \mathsf{conn}$ =

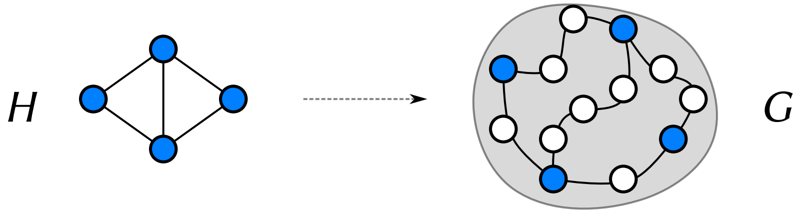
Classes excluding a **topological minor**

Tameness of FO + conn

Tameness of FO + conn

Def: *G* contains *H* as a **top minor** if

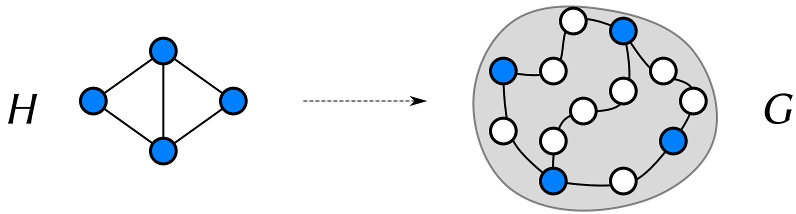
G contains a **subdivision** of *H* as a subgraph.



Tameness of FO + conn

Def: G contains H as a **top minor** if

G contains a **subdivision** of H as a subgraph.



Theorem [P, Schirrmacher,Siebertz, Toruńczyk, Vigny '22]

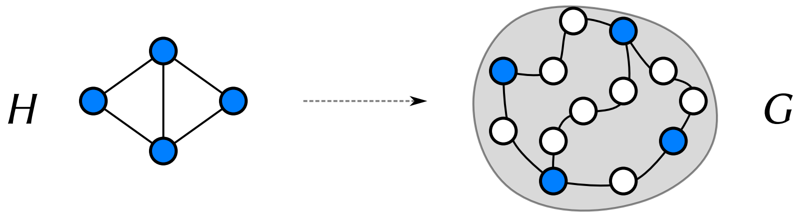
Suppose $\mathscr{C}$ can excludes some fixed graph H as a topological minor.

Then model-checking FO + conn on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Tameness of FO + conn

Def: G contains H as a **top minor** if

G contains a **subdivision** of H as a subgraph.



Theorem [P, Schirrmacher,Siebertz, Toruńczyk, Vigny '22]

Suppose $\mathscr{C}$ can excludes some fixed graph H as a topological minor.

Then model-checking FO + conn on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [follows from the same work]

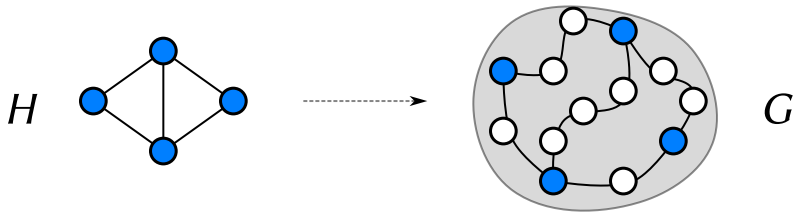
Suppose $\mathscr{C}$ is a **subgraph-closed** graph class. Then:

$\mathscr{C}$ is **mon dependent** wrt FO + conn $\Leftrightarrow$ $\mathscr{C}$ excludes some **top minor** H.

Tameness of FO + conn

Def: G contains H as a **top minor** if

G contains a **subdivision** of H as a subgraph.



Theorem [P, Schirrmacher,Siebertz, Toruńczyk, Vigny '22]

Suppose $\mathscr{C}$ can excludes some fixed graph H as a topological minor.

Then model-checking FO + conn on $\mathscr{C}$ in time $\mathcal{O}_\varphi(|G|)$.

Theorem [follows from the same work]

Suppose $\mathscr{C}$ is a **subgraph-closed** graph class. Then:

$\mathscr{C}$ is **mon dependent** wrt FO + conn $\Leftrightarrow$ $\mathscr{C}$ excludes some **top minor** H.

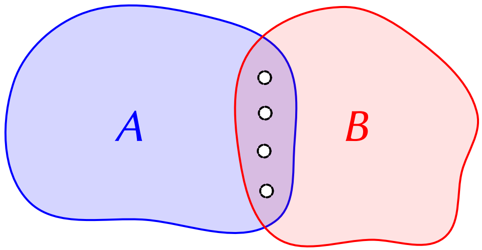
Note: $\Rightarrow$ is easy, because FO + conn can shorten long paths.

Unbreakability

Unbreakability

Separation of G is (A, B) with $A \cup B = V(G)$ and $E(A - B, B - A) = \emptyset$.

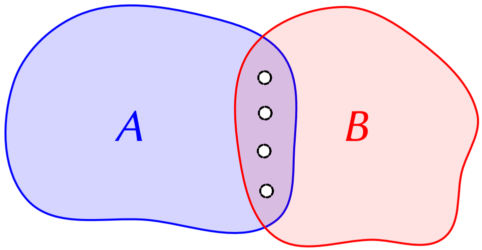
Order of (A, B) is $|A \cap B|$.



Unbreakability

Separation of G is (A, B) with $A \cup B = V(G)$ and $E(A - B, B - A) = \emptyset$.

Order of (A, B) is $|A \cap B|$.



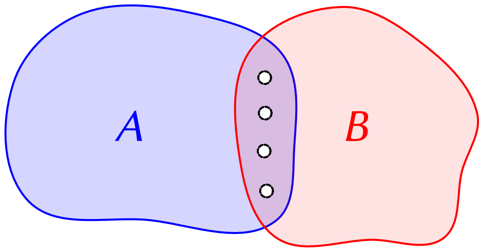
$S \subseteq V(G)$ is (q, k)**-unbreakable** if for every separation (A, B) of order $\leqslant k$,

$$|A \cap S| \leqslant q \qquad \text{or} \qquad |B \cap S| \leqslant q.$$

Unbreakability

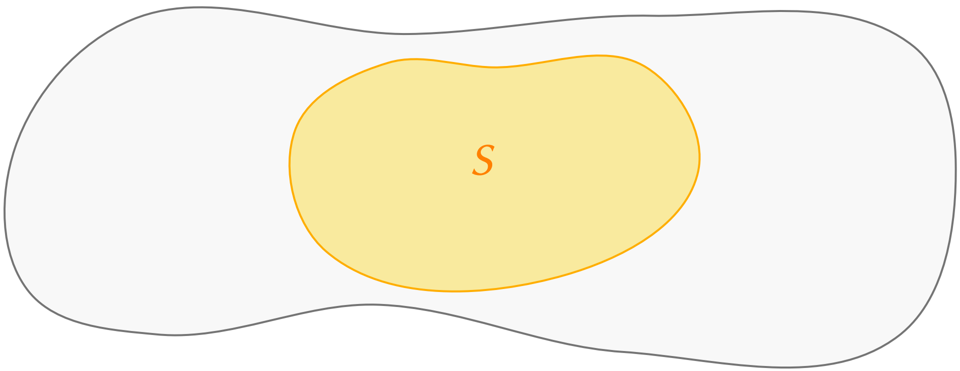
Separation of G is (A, B) with $A \cup B = V(G)$ and $E(A - B, B - A) = \emptyset$.

Order of (A, B) is $|A \cap B|$.



$S \subseteq V(G)$ is **(q, k)-unbreakable** if for every separation (A, B) of order $\leqslant k$,

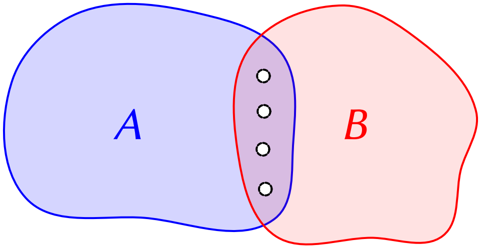
$$|A \cap S| \leqslant q \qquad \text{or} \qquad |B \cap S| \leqslant q.$$



Unbreakability

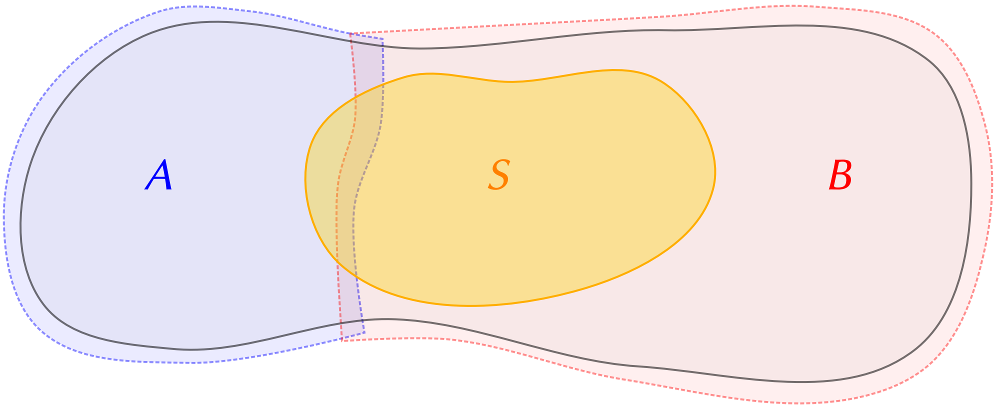
Separation of G is (A, B) with $A \cup B = V(G)$ and $E(A - B, B - A) = \emptyset$.

Order of (A, B) is $|A \cap B|$.



$S \subseteq V(G)$ is **(q, k)-unbreakable** if for every separation (A, B) of order $\leqslant k$,

$$|A \cap S| \leqslant q \qquad \text{or} \qquad |B \cap S| \leqslant q.$$



Unbreakability and FO + conn

Unbreakability and FO + conn

Obs: If G is (q,k)-**unbreakable** for some $q, k \in \mathbb{N}$, (i.e. $S = V(G)$)

then $\mathrm{FO} + \mathrm{conn}_{\leqslant k}$ can be reduced to plain FO on G.

Unbreakability and FO + conn

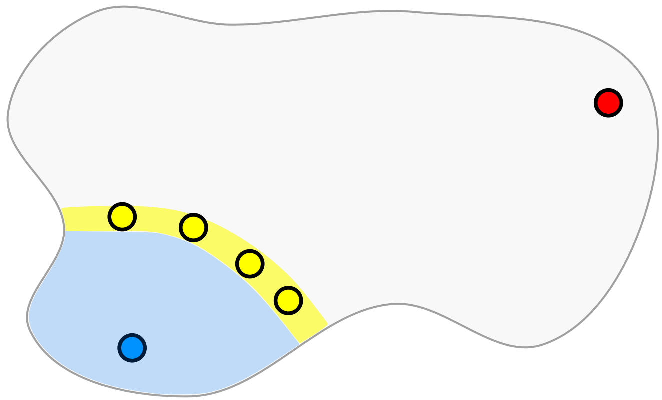
Obs: If G is (q,k)-**unbreakable** for some $q, k \in \mathbb{N}$, (i.e. $S = V(G)$)

then $\mathsf{FO} + \mathsf{conn}_{\leqslant k}$ can be reduced to plain FO on G.

Proof:

If u, v disconnected in $G - \{a_1, \ldots, a_k\}$,

then $\leqslant q$ vrtcs reachable from u, or $\leqslant q$ vrtcs reachable from v.



Unbreakability and FO + conn

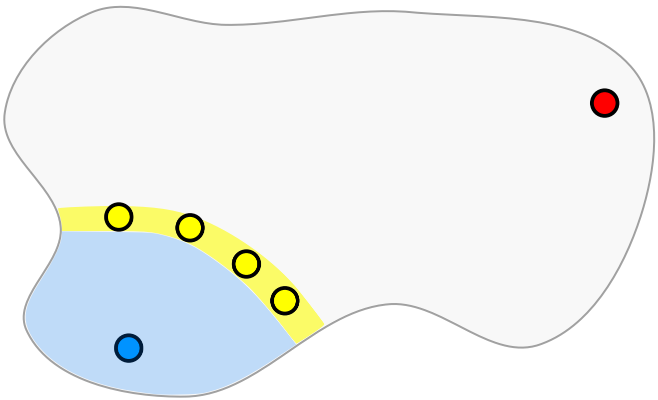
Obs: If G is (q, k)-**unbreakable** for some $q, k \in \mathbb{N}$, (i.e. $S = V(G)$)

then $\mathrm{FO} + \mathrm{conn}_{\leqslant k}$ can be reduced to plain FO on G.

Proof:

If u, v disconnected in $G - \{a_1, \ldots, a_k\}$,

then $\leqslant q$ vrtcs reachable from u, or $\leqslant q$ vrtcs reachable from v.



Hence, $\neg\mathrm{conn}_k(u, v, a_1, \ldots, a_k)$ is equivalent to:

There is X with $|X| \leqslant q$, $|X \cap \{u, v\}| = 1$, and $N(X) \subseteq \{a_1, \ldots, a_k\}$.

Unbreakable decomposition

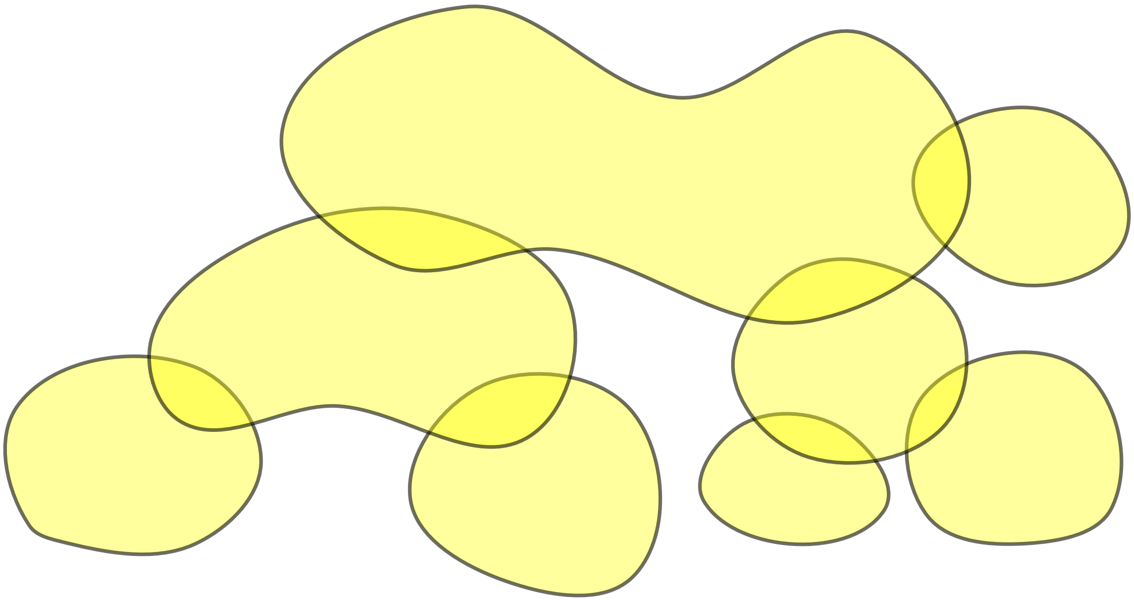
Theorem

[Cygan, Lokshtanov, Pilipczuk, P, Saurabh '14]
[Cygan, Komosa, Lokshtanov, Pilipczuk, P, Saurabh, Wahlström '18]
[Korhonen '24]

For every $k \in \mathbb{N}$, every graph G has a tree decomposition with

adhesion $\leqslant k$ and (k, k)-**unbreakable bags**.

Moreover, such a tree decomposition can be found in time $\mathcal{O}_k(\|G\|)$.



Model-checking $\mathrm{FO} + \mathrm{conn}$

Model-checking $\mathsf{FO} + \mathsf{conn}$

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

Model-checking FO + conn

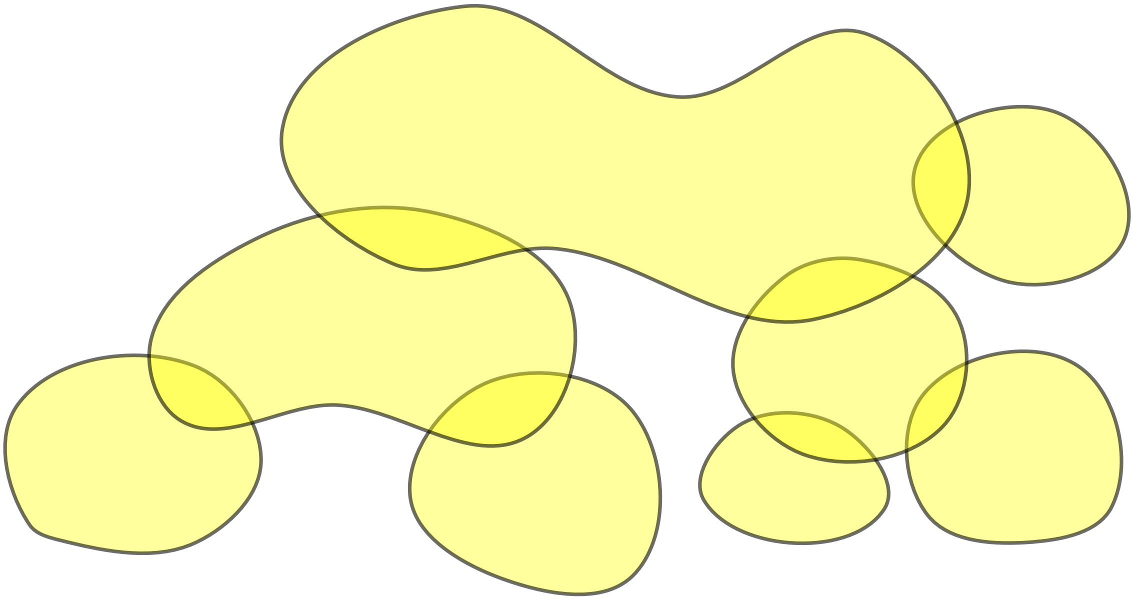
Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \qquad p \coloneqq \mathrm{qr}(\varphi).$$

Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

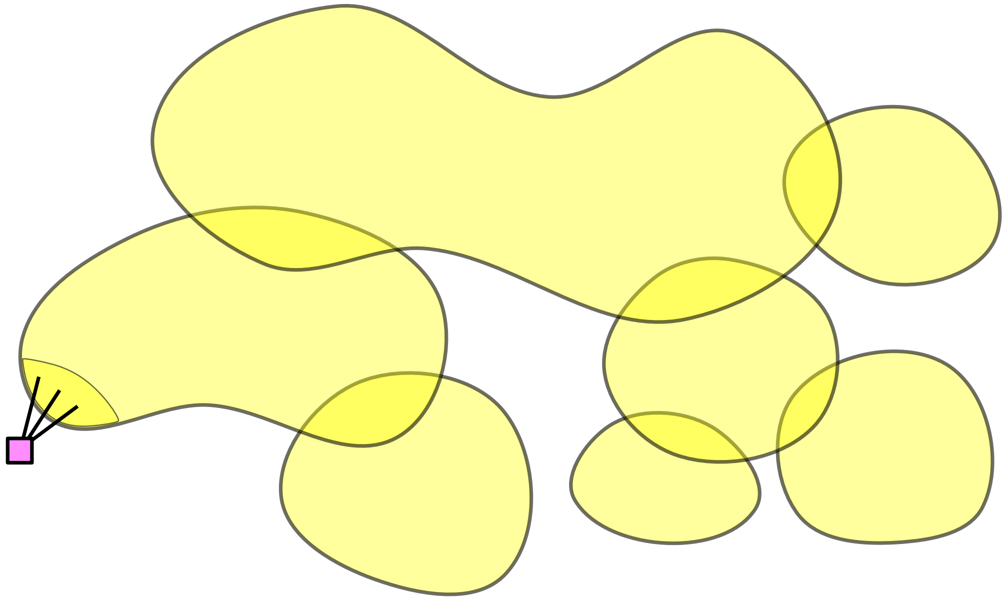
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \qquad p := \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

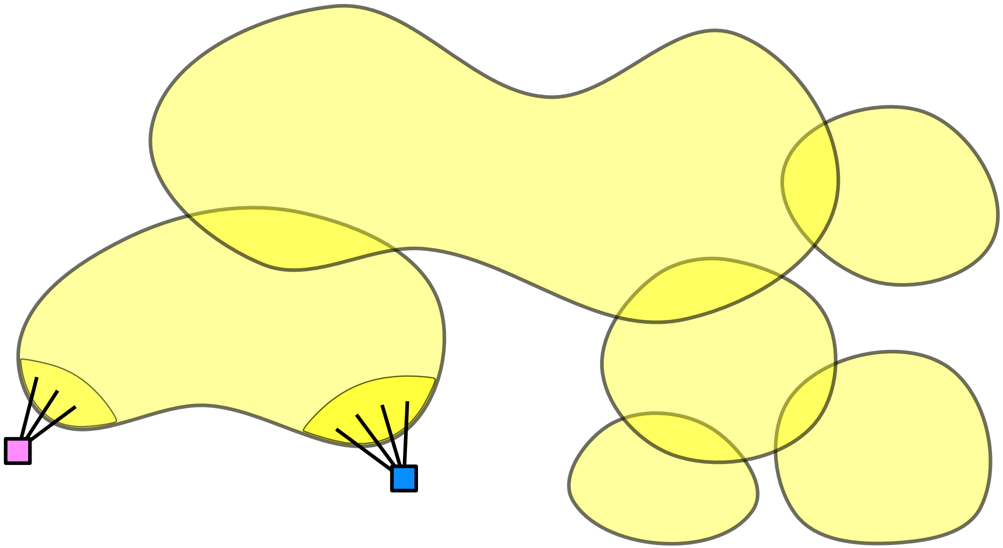
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}, \qquad p := \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

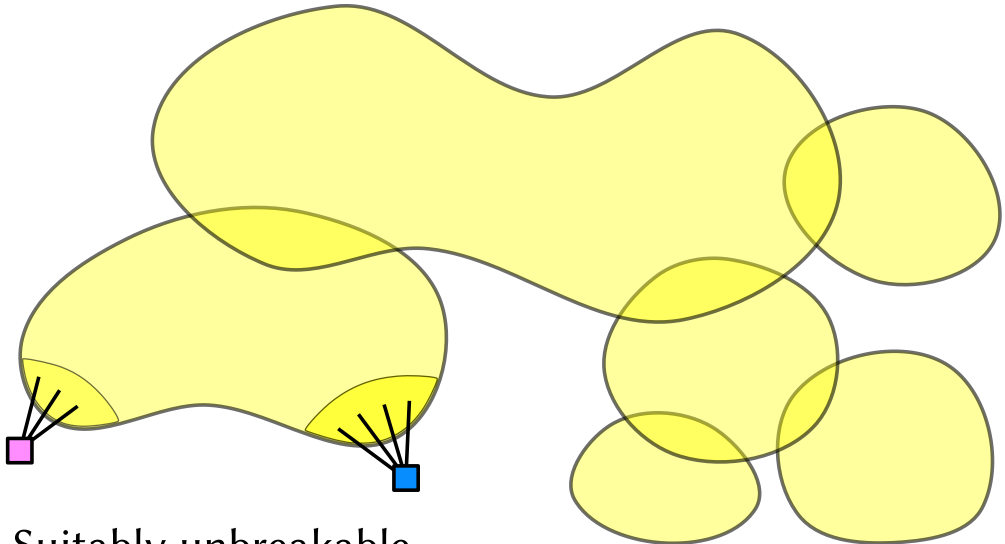
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}, \qquad p \coloneqq \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

G is K_t-top-minor-free, $\varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}$, $p := \mathrm{qr}(\varphi)$.

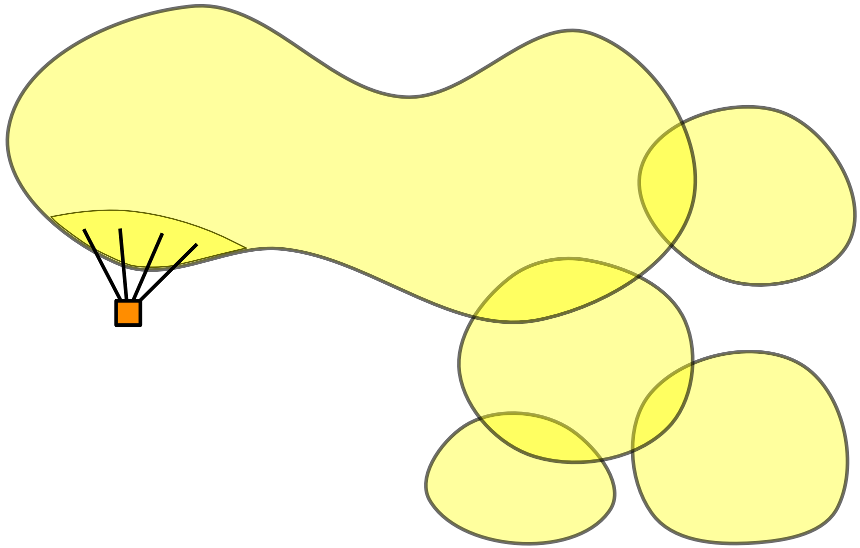


Suitably unbreakable
K_h-top-minor-free for $h = h(t, k)$.

Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

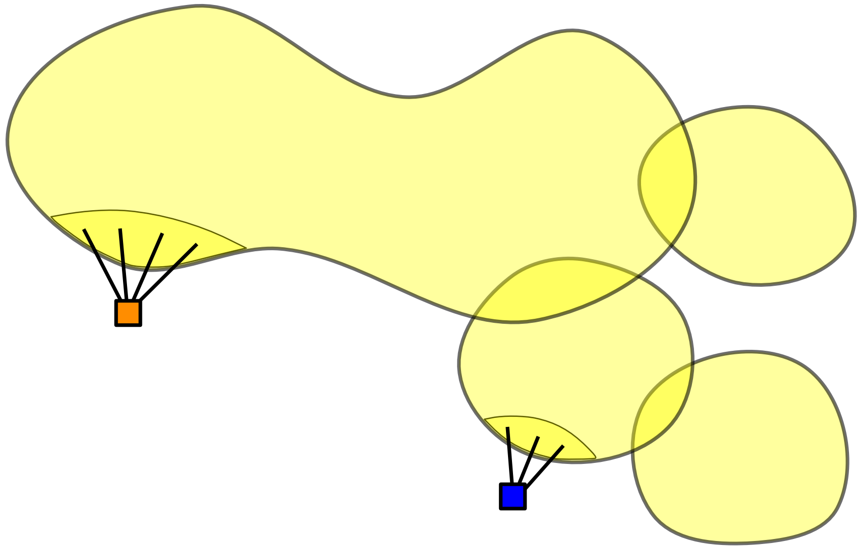
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \qquad p := \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

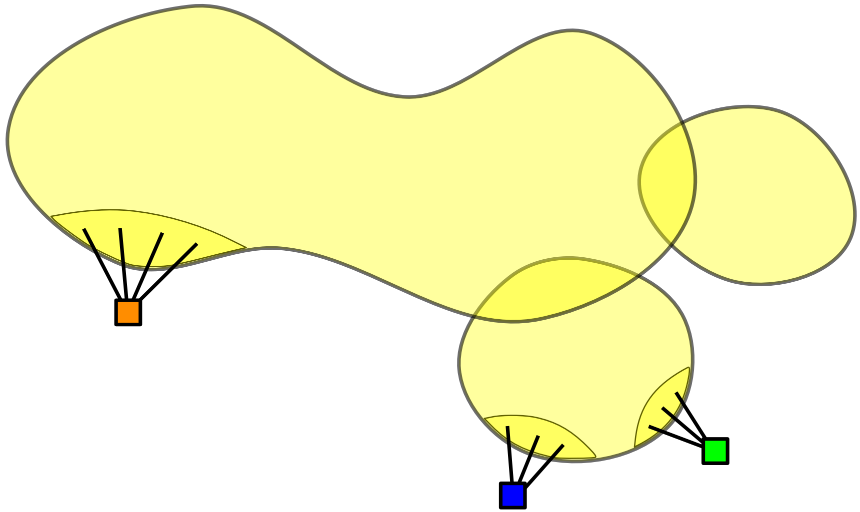
G is K_t-top-minor-free, $\quad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}$, $\quad p \coloneqq \mathrm{qr}(\varphi)$.



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

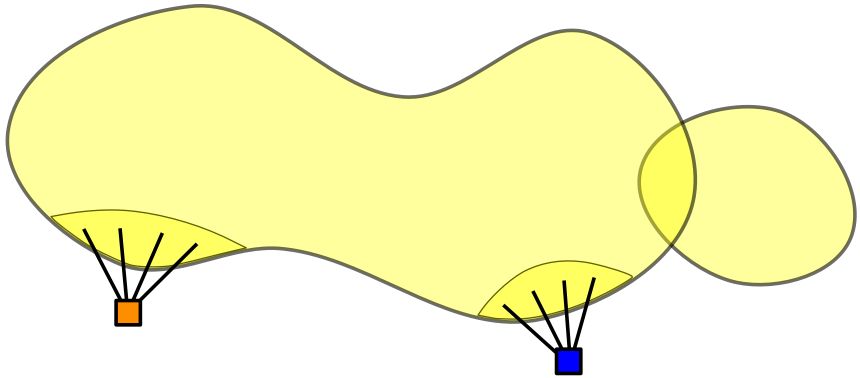
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}, \qquad p \coloneqq \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

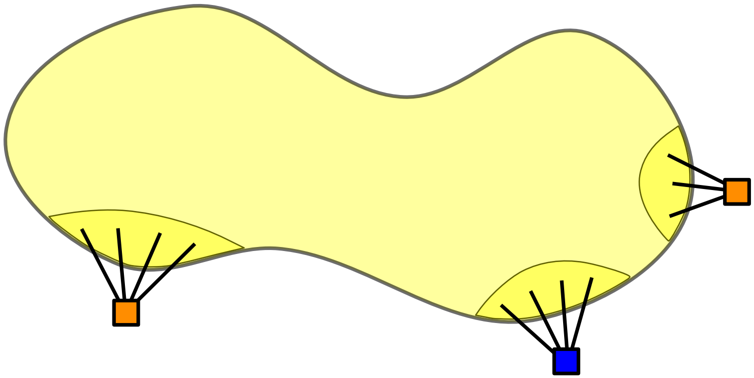
$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}, \qquad p := \mathrm{qr}(\varphi).$$



Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathrm{FO} + \mathrm{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

G is K_t-top-minor-free, $\qquad \varphi \in \mathrm{FO} + \mathrm{conn}_{\leqslant k}, \qquad p \coloneqq \mathrm{qr}(\varphi)$.



Model-checking FO + conn

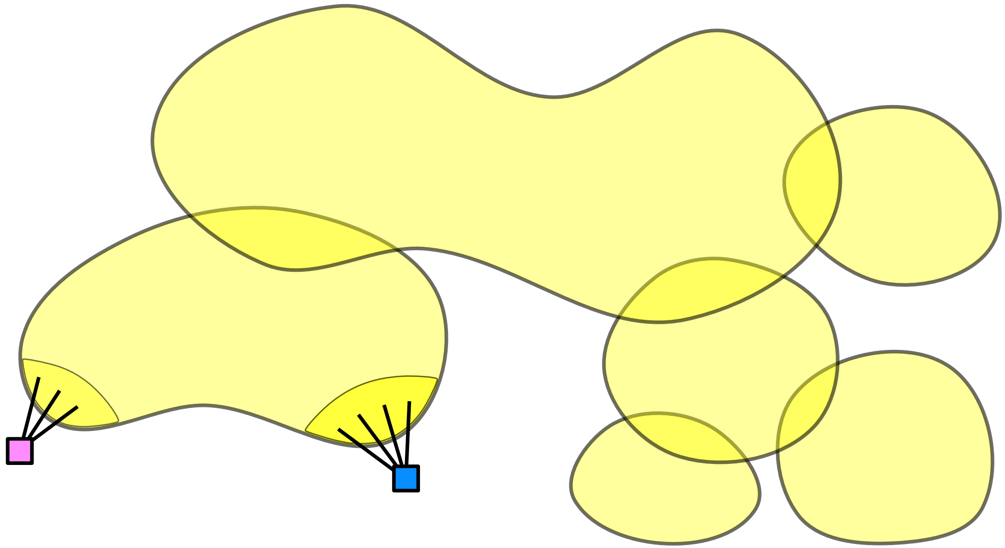
Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \qquad p \coloneqq \mathrm{qr}(\varphi).$$

Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

G is K_t-top-minor-free, $\quad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \quad p \coloneqq \mathrm{qr}(\varphi)$.

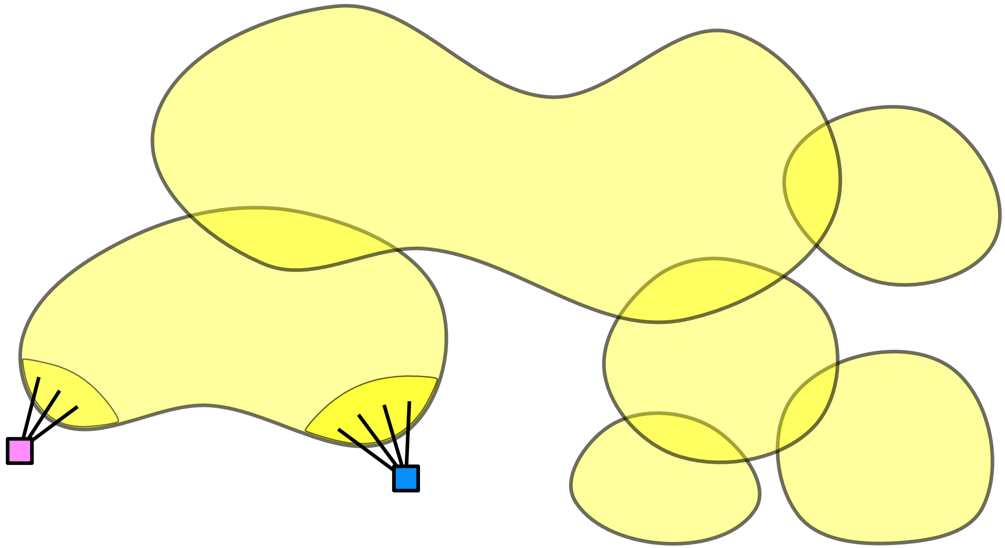


Top-minor-free: Stable under stellations and efficient FO model-checking.

Model-checking FO + conn

Given: $G \in \mathscr{C}$ and $\varphi \in \mathsf{FO} + \mathsf{conn}$. Let $t, k, p \in \mathbb{N}$ be such that:

$$G \text{ is } K_t\text{-top-minor-free}, \qquad \varphi \in \mathsf{FO} + \mathsf{conn}_{\leqslant k}, \qquad p := \mathrm{qr}(\varphi).$$

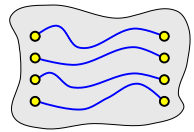


Top-minor-free: Stable under stellations and efficient FO model-checking.

General scheme for **local-to-global** lifts in **linear time**.

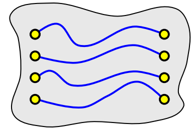
Beyond FO + conn

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

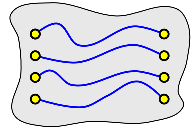
Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

Beyond FO + conn



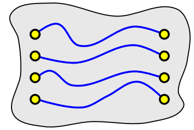
FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes

[Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]

Beyond FO + conn

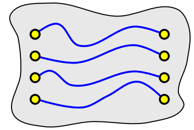


FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Beyond FO + conn



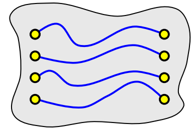
FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

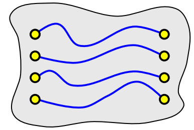
MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

Proposition: FO + conn is meaningful only on sparse graphs.

Beyond FO + conn



$\mathsf{FO}+\mathsf{dp}$: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

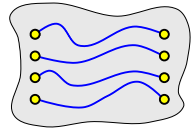
- $\mathsf{FO}+\mathsf{dp}$ tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- $\mathsf{MSO}/\mathsf{tw}+\mathsf{dp}$ tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

Proposition: FO + conn is meaningful only on sparse graphs.

What is the dense analogue of $\mathsf{FO}+\mathsf{conn}$?

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

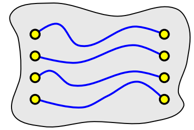
Very useful encapsulation of techniques of Graph Minors.

Proposition: FO + conn is meaningful only on sparse graphs.

What is the dense analogue of FO + conn?

What is the dense analogue of **top-minor-free**?

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

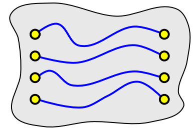
Proposition: FO + conn is meaningful only on sparse graphs.

What is the dense analogue of FO + conn?

What is the dense analogue of **top-minor-free**?

Idea: FO + conn allows quantification over **vertex cuts** of small order.

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

Proposition: FO + conn is meaningful only on sparse graphs.

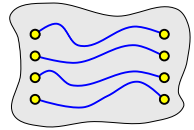
What is the dense analogue of FO + conn?

What is the dense analogue of **top-minor-free**?

Idea: FO + conn allows quantification over **vertex cuts** of small order.

Maybe allow quantification over **dense cuts** of small order?

Beyond FO + conn



FO + dp: FO with **disjoint paths** predicates.

MSO/tw: FO with quantification over sets of small **bidimensionality**.

- FO + dp tractable on **top-minor-free** classes [Schirrmacher, Siebertz, Stamoulis, Thilikos, Vigny '23]
- MSO/tw + dp tractable on **minor-free** classes [Sau, Stamoulis, Thilikos '24]

Very useful encapsulation of techniques of Graph Minors.

Proposition: FO + conn is meaningful only on sparse graphs.

What is the dense analogue of FO + conn?

What is the dense analogue of **top-minor-free**?

Idea: FO + conn allows quantification over **vertex cuts** of small order.

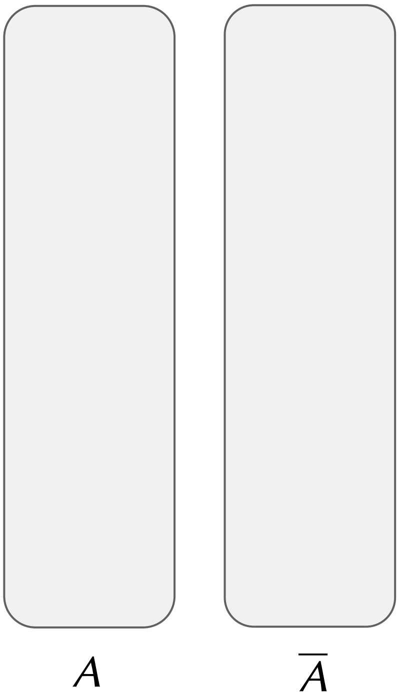
Maybe allow quantification over **dense cuts** of small order?

Quantification over sets of small **cutrank**.

Cutrank

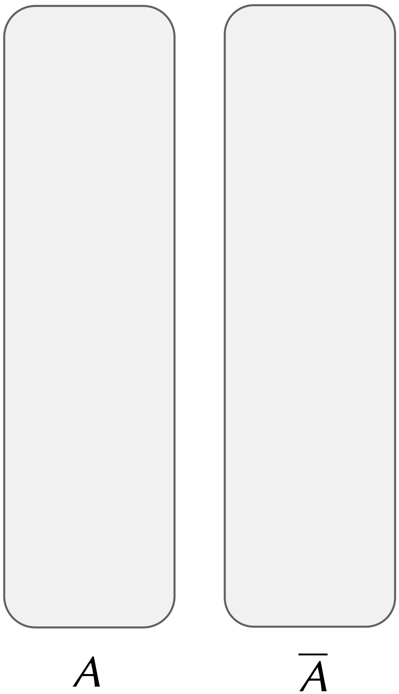
Cutrank

Suppose $(A, \overline{A})$ is a partition of the vertex set of G.



Cutrank

Suppose $(A, \overline{A})$ is a partition of the vertex set of G.

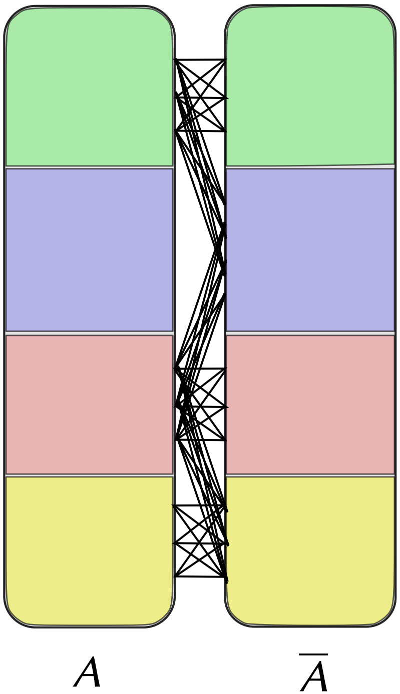


$\mathrm{cutrank}(A) := $ rank over $\mathbb{F}_2$ of $\mathrm{Adj}[A, \overline{A}]$.

$$A \quad \underset{\overline{A}}{\begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}}$$

Cutrank

Suppose $(A, \overline{A})$ is a partition of the vertex set of G.



$\text{cutrank}(A) := \text{rank over } \mathbb{F}_2 \text{ of } \text{Adj}[A, \overline{A}]$.

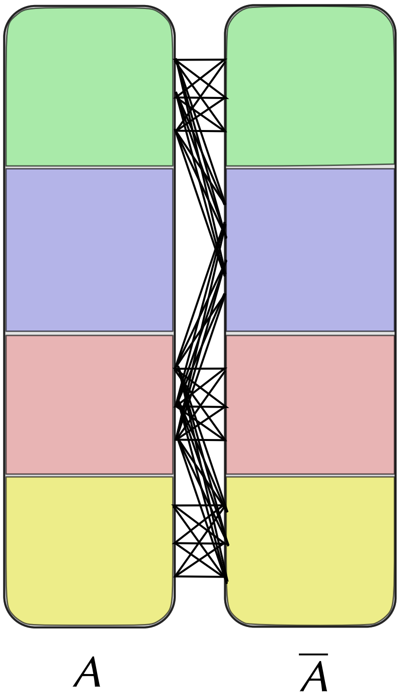
$$A \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

$$\overline{A}$$

$$\text{diversity}(A) := \#(\text{types in } A \text{ over } \overline{A}) + \#(\text{types in } \overline{A} \text{ over } A)$$

Cutrank

Suppose $(A, \overline{A})$ is a partition of the vertex set of G.



$$\mathrm{cutrank}(A) := \text{rank over } \mathbb{F}_2 \text{ of } \mathrm{Adj}[A, \overline{A}].$$

$$A \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$
$$\overline{A}$$

$$\mathrm{diversity}(A) := \#(\text{types in } A \text{ over } \overline{A}) + \#(\text{types in } \overline{A} \text{ over } A)$$

$$\mathrm{cutrank}(A) \leqslant \mathrm{diversity}(A) \leqslant 2 \cdot 2^{\mathrm{cutrank}(A)}$$

Low rank MSO

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Cliquewidth: hierarchical decomposition by cuts of **diversity** $\leqslant k$.

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Cliquewidth: hierarchical decomposition by cuts of **diversity** $\leqslant k$.

Note: Cutrank and diversity are **symmetric**, cutrank is **submodular**.

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Cliquewidth: hierarchical decomposition by cuts of **diversity** $\leqslant k$.

Note: Cutrank and diversity are **symmetric**, cutrank is **submodular**.

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X\colon \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Cliquewidth: hierarchical decomposition by cuts of **diversity** $\leqslant k$.

Note: Cutrank and diversity are **symmetric**, cutrank is **submodular**.

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X:\ \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Obs: Bounding **diversity** yields the same logic. $\exists_{X:\ \mathrm{diversity}(X) \leqslant r}\ \psi$

Low rank MSO

Treewidth: hierarchical decomposition by **separations** of order $\leqslant k$.

Rankwidth: hierarchical decomposition by cuts of **cutrank** $\leqslant k$.

Cliquewidth: hierarchical decomposition by cuts of **diversity** $\leqslant k$.

Note: Cutrank and diversity are **symmetric**, cutrank is **submodular**.

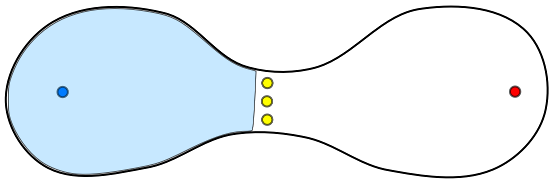
Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X\colon \mathrm{cutrank}(X)\leqslant r}\ \psi$$

Obs: Bounding **diversity** yields the same logic. $\exists_{X\colon \mathrm{diversity}(X)\leqslant r}\ \psi$

Obs: FO + conn $\subseteq$ low rank MSO.



$\mathrm{cutrank}(\mathrm{Reach}(u, G - \{a_1, \ldots, a_k\})) \leqslant k$

Low rank MSO: complexity

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X\colon \mathsf{cutrank}(X)\leqslant r}\ \psi$$

Low rank MSO: complexity

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X:\ \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Q: Relation to $\mathsf{FO} + \mathsf{conn}$? Do the two logics coincide on sparse graphs?

Low rank MSO: complexity

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X:\ \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Q: Relation to FO + conn? Do the two logics coincide on sparse graphs?

Q: Model-checking FO + conn trivially in XP. What about low rank MSO?

Low rank MSO: **complexity**

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X:\ \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Q: Relation to FO + conn? Do the two logics coincide on sparse graphs?

Q: Model-checking FO + conn trivially in XP. What about low rank MSO?

Q: FO-like logics equivalent to low rank MSO?

Low rank MSO: **complexity**

Low rank MSO

MSO_1 where every set quantification is guarded by a bound on **cutrank**.

$$\varphi = \exists_{X:\ \mathrm{cutrank}(X) \leqslant r}\ \psi$$

Q: Relation to FO + conn? Do the two logics coincide on sparse graphs?

Q: Model-checking FO + conn trivially in XP. What about low rank MSO?

Q: FO-like logics equivalent to low rank MSO?

joint work with

M. Bojańczyk, W. Przybyszewski, M. Sokołowski, and G. Stamoulis

Low rank MSO: results

Low rank MSO: results

Theorem

Suppose $\mathscr{C}$ is **weakly sparse** (excludes $K_{t,t}$ as a subgraph for some $t \in \mathbb{N}$).

Then FO + conn and low rank MSO coincide on $\mathscr{C}$.

Low rank MSO: results

Theorem

Suppose $\mathscr{C}$ is **weakly sparse** (excludes $K_{t,t}$ as a subgraph for some $t \in \mathbb{N}$).

Then FO + conn and low rank MSO coincide on $\mathscr{C}$.

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with logic FO + flipconn.

This equivalence does **not** hold on all graphs.

Low rank MSO: results

Theorem

Suppose $\mathscr{C}$ is **weakly sparse** (excludes $K_{t,t}$ as a subgraph for some $t \in \mathbb{N}$).

Then FO + conn and low rank MSO coincide on $\mathscr{C}$.

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with logic FO + flipconn.

This equivalence does **not** hold on all graphs.

Theorem

Low rank MSO coincides with FO + flipreach on **all graphs**.

Low rank MSO: results

Theorem

Suppose $\mathscr{C}$ is **weakly sparse** (excludes $K_{t,t}$ as a subgraph for some $t \in \mathbb{N}$).

Then FO + conn and low rank MSO coincide on $\mathscr{C}$.

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with logic FO + flipconn.

This equivalence does **not** hold on all graphs.

Theorem

Low rank MSO coincides with FO + flipreach on **all graphs**.

Corollary

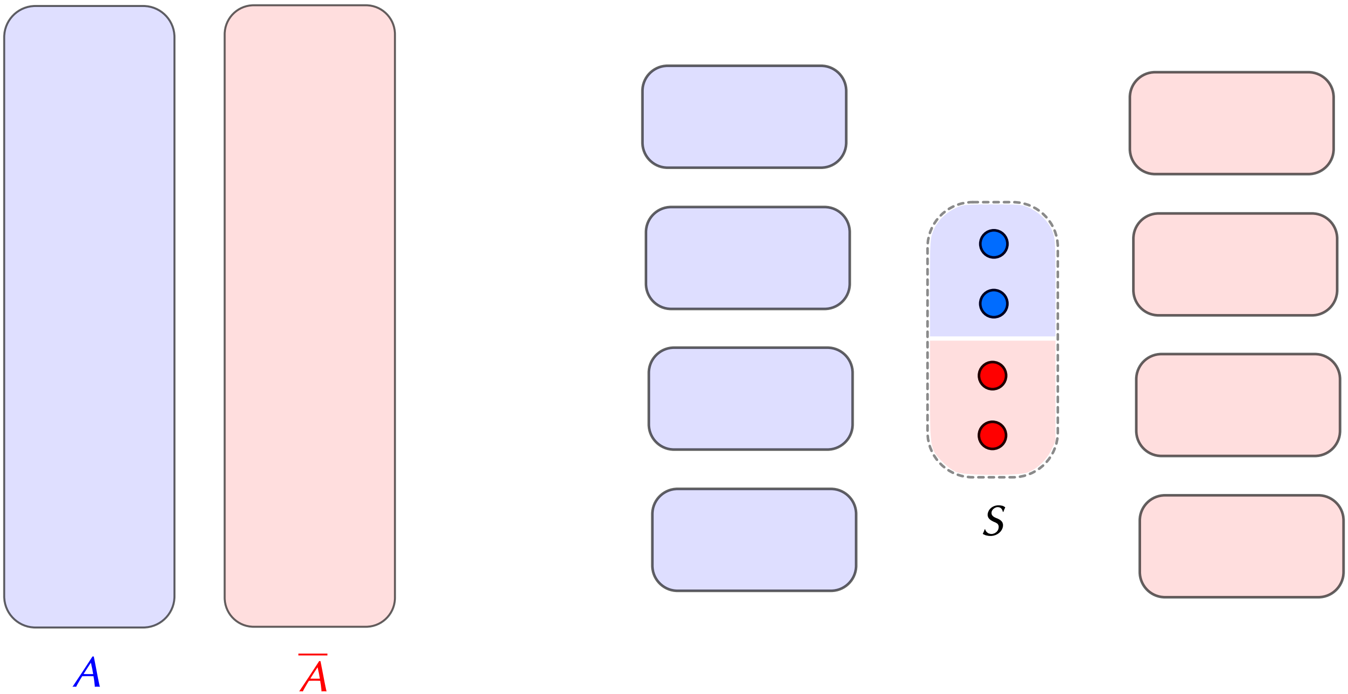
Model-checking low rank MSO is in XP.

Low rank MSO on weakly sparse classes

Low rank MSO on weakly sparse classes

Lemma

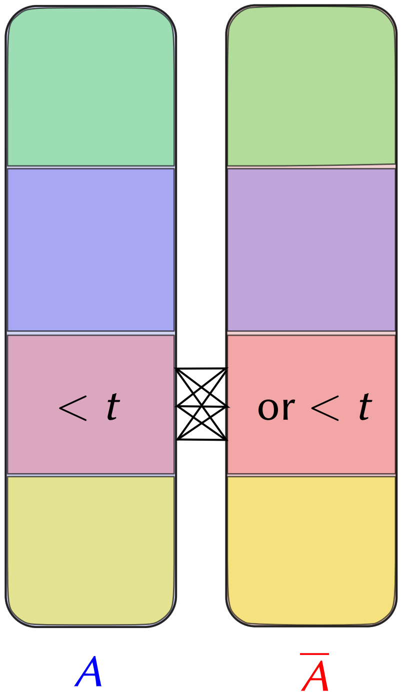
Let G be $K_{t,t}$-free and diversity$(A) \leqslant k$. Then there is S with $|S| \leqslant kt$ s.t. every component of $G - S$ is entirely in A or entirely in $\overline{A}$.

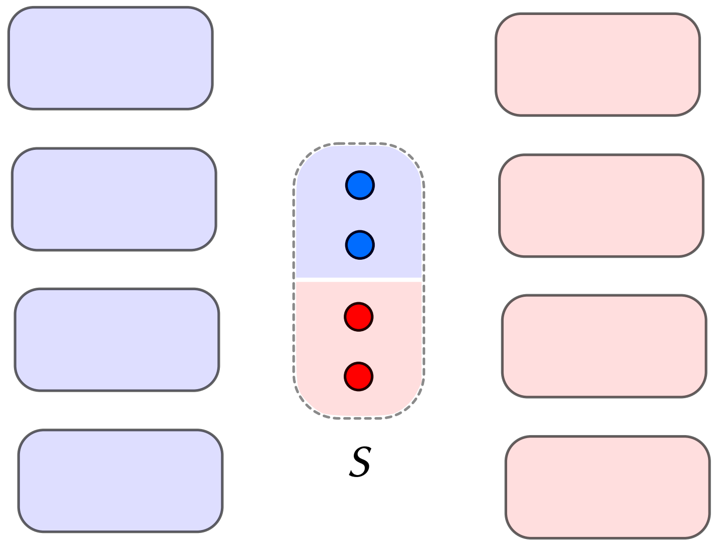


Low rank MSO on weakly sparse classes

Lemma

Let G be $K_{t,t}$-free and diversity$(A) \leqslant k$. Then there is S with $|S| \leqslant kt$ s.t. every component of $G - S$ is entirely in A or entirely in $\overline{A}$.

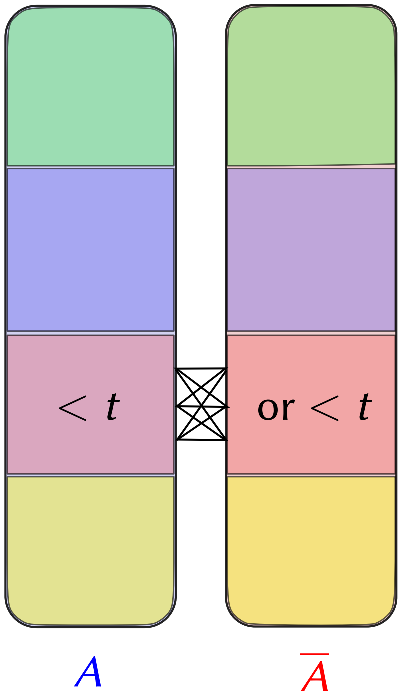


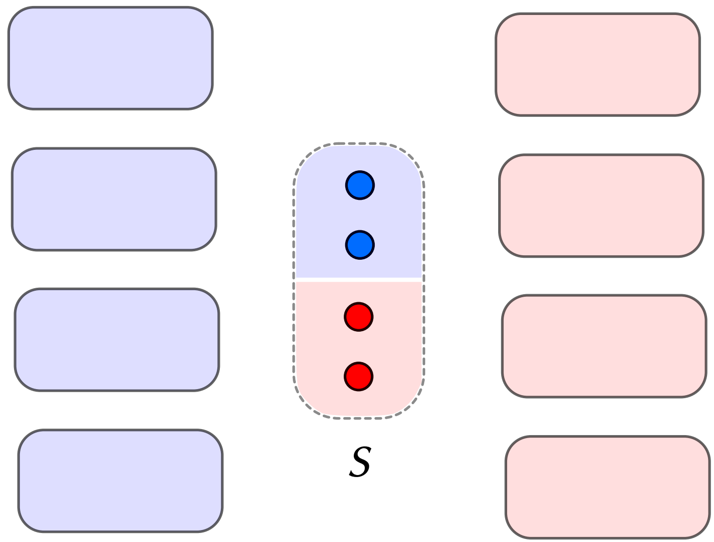


Low rank MSO on weakly sparse classes

Lemma

Let G be $K_{t,t}$-free and diversity$(A) \leqslant k$. Then there is S with $|S| \leqslant kt$ s.t. every component of $G - S$ is entirely in A or entirely in $\overline{A}$.





$S :=$ Union of types of size $< t$.

Low rank MSO on weakly sparse classes

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

Low rank MSO on weakly sparse classes

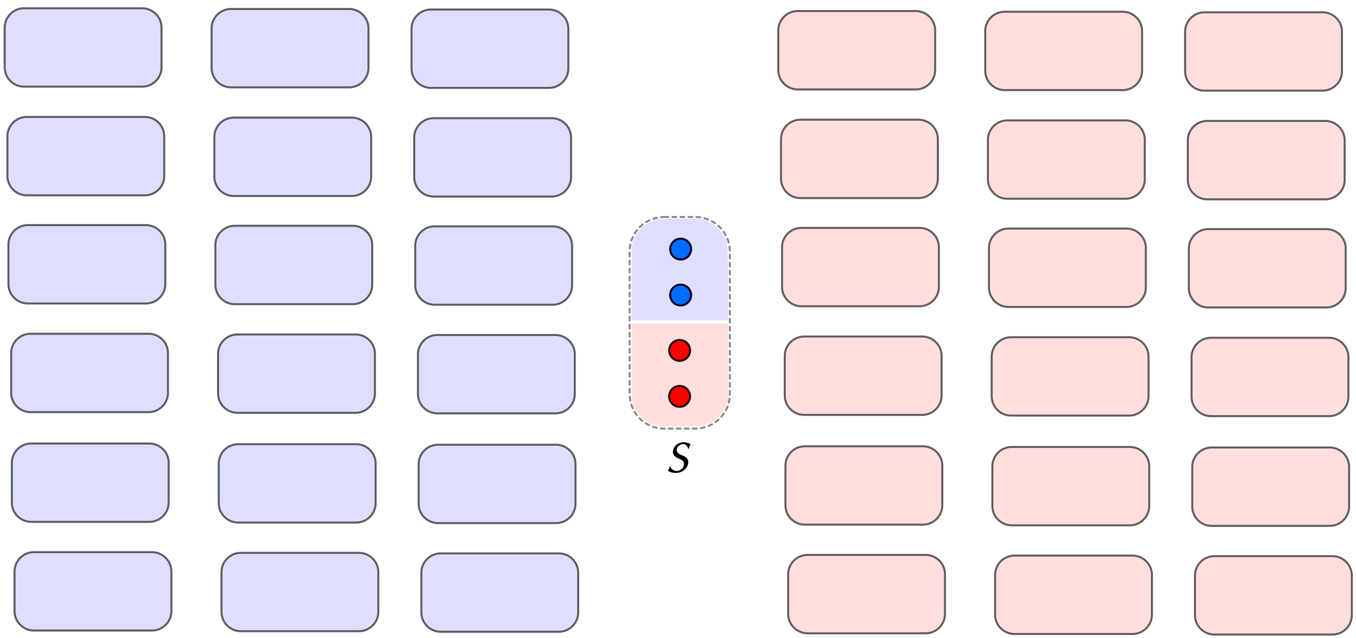
Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A\colon \mathsf{diversity}(A) \leqslant k}\, \psi \quad \text{where} \quad \psi \in \mathsf{FO} + \mathsf{conn} \text{ by induction.}$$

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

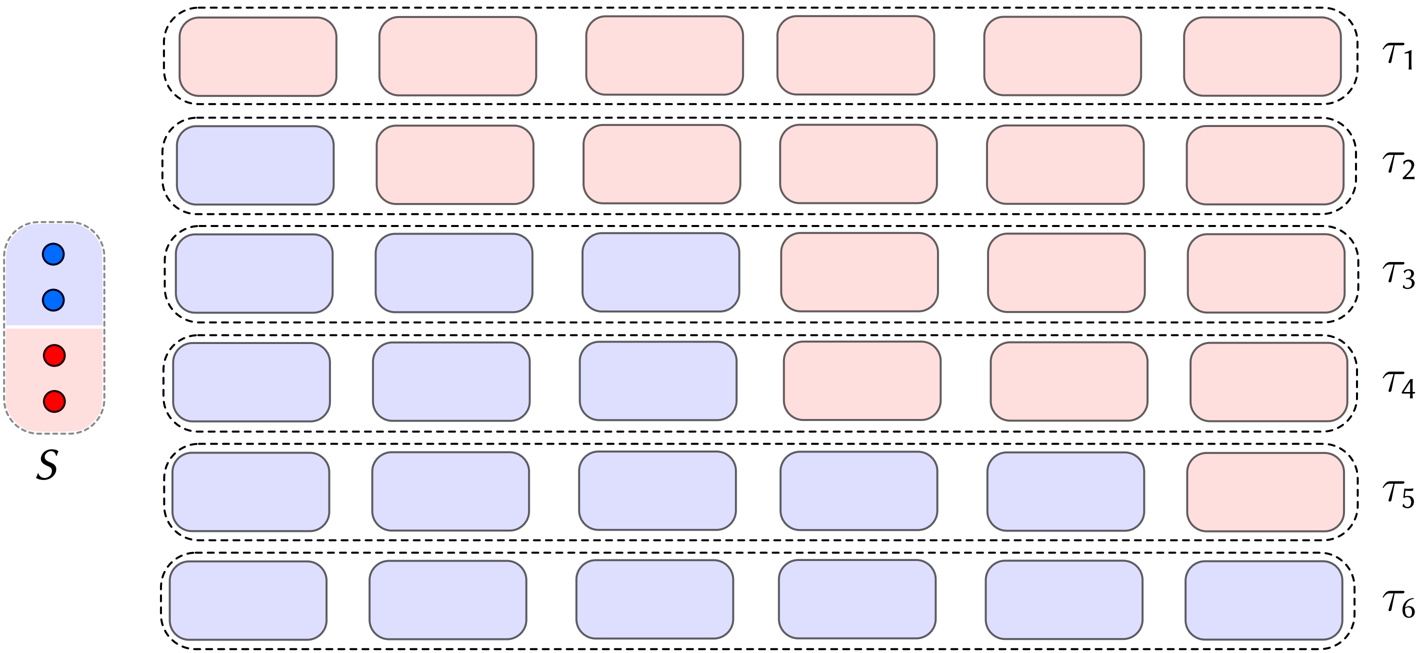
$$\varphi = \exists_{A:\ \mathrm{diversity}(A) \leqslant k}\ \psi \quad \text{where} \quad \psi \in \mathrm{FO} + \mathrm{conn}\ \text{by induction.}$$



Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A:\ \mathsf{diversity}(A) \leqslant k}\ \psi \quad \text{where} \quad \psi \in \mathsf{FO} + \mathsf{conn} \text{ by induction.}$$

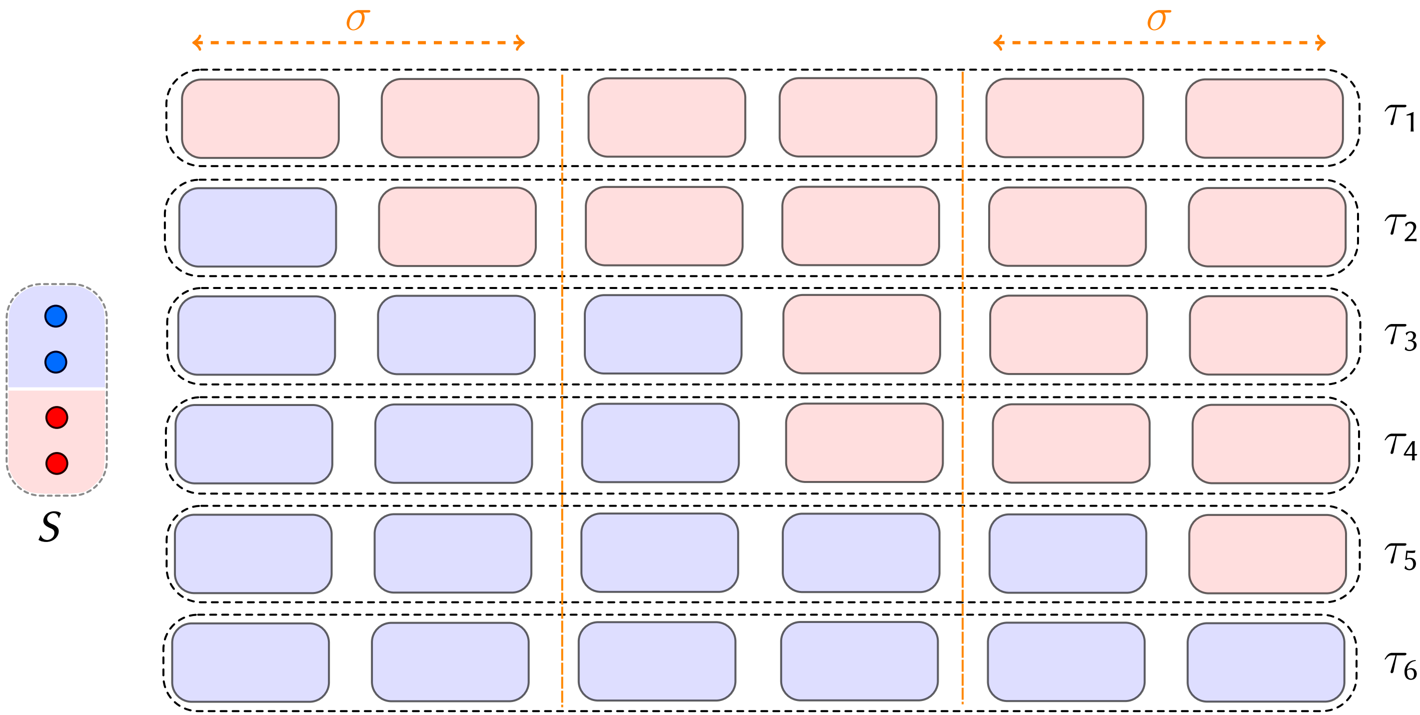


Classify comps wrt high enough FO + conn type.

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A:\ \mathrm{diversity}(A)\leqslant k}\ \psi \quad \text{where} \quad \psi \in \mathrm{FO} + \mathrm{conn} \text{ by induction.}$$

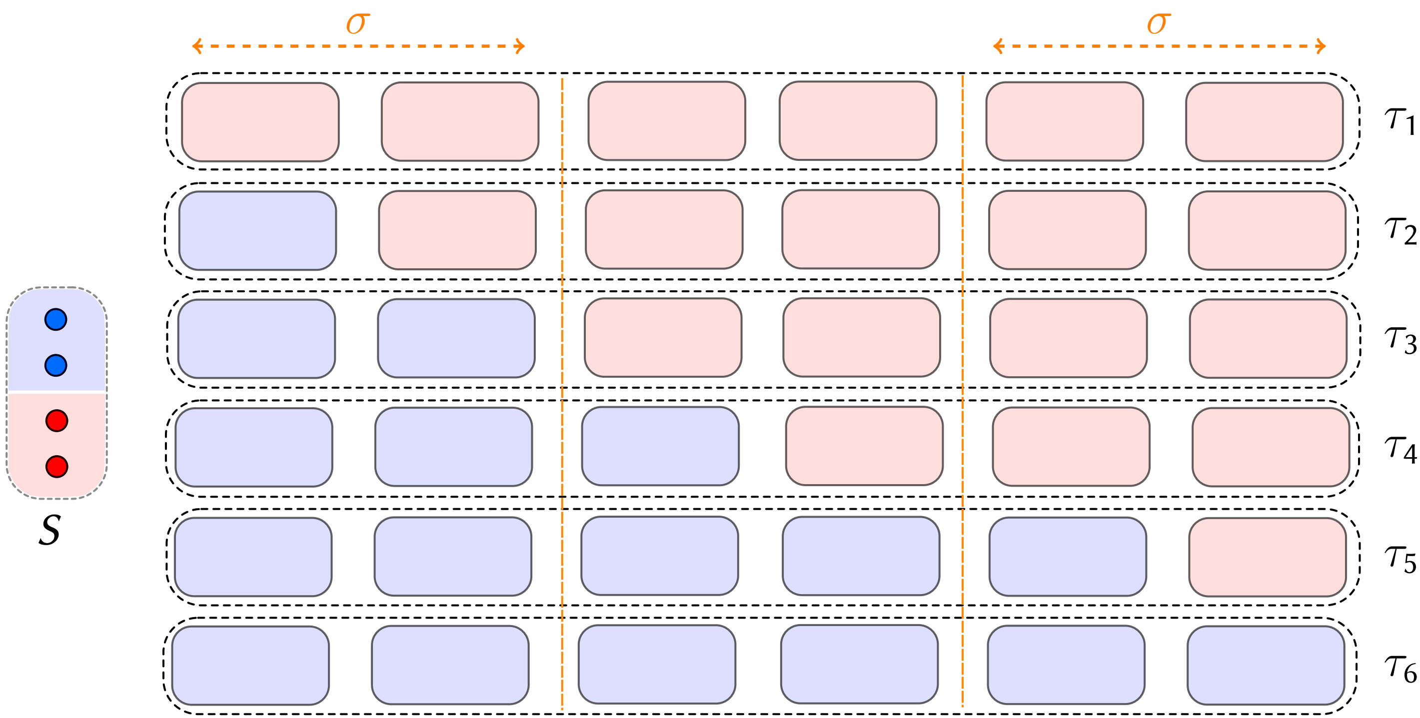


Obs: Can move comps between A and $\overline{A}$ within a type, subject to a margin σ.

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$\varphi = \exists_{A:\ \mathrm{diversity}(A)\leqslant k}\ \psi$ where $\psi \in$ FO + conn by induction.

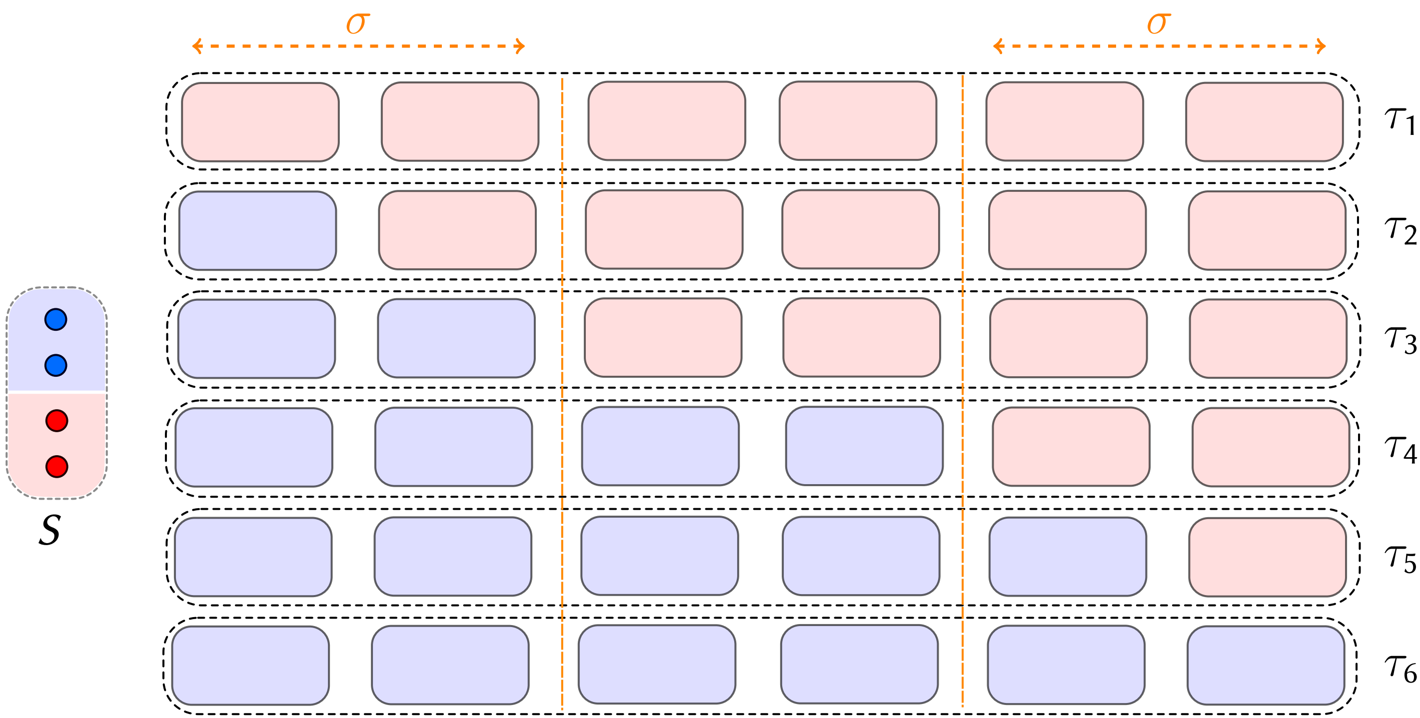


Obs: Can move comps between A and $\overline{A}$ within a type, subject to a margin σ.

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A:\ \mathrm{diversity}(A)\leqslant k}\ \psi \quad \text{where} \quad \psi \in \mathrm{FO} + \mathrm{conn}\ \text{by induction.}$$

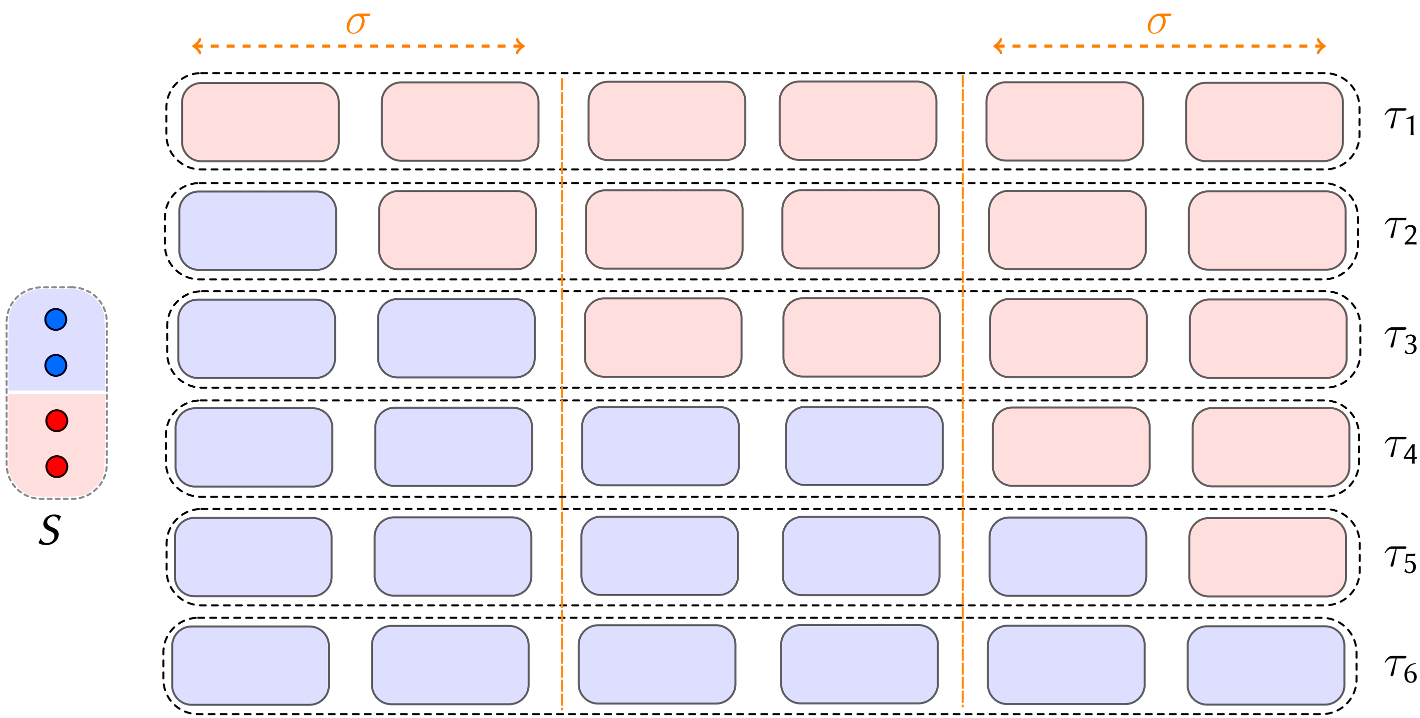


Obs: Can move comps between A and $\overline{A}$ within a type, subject to a margin σ.

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A:\ \mathrm{diversity}(A) \leqslant k}\ \psi \quad \text{where} \quad \psi \in \mathrm{FO} + \mathrm{conn} \text{ by induction.}$$

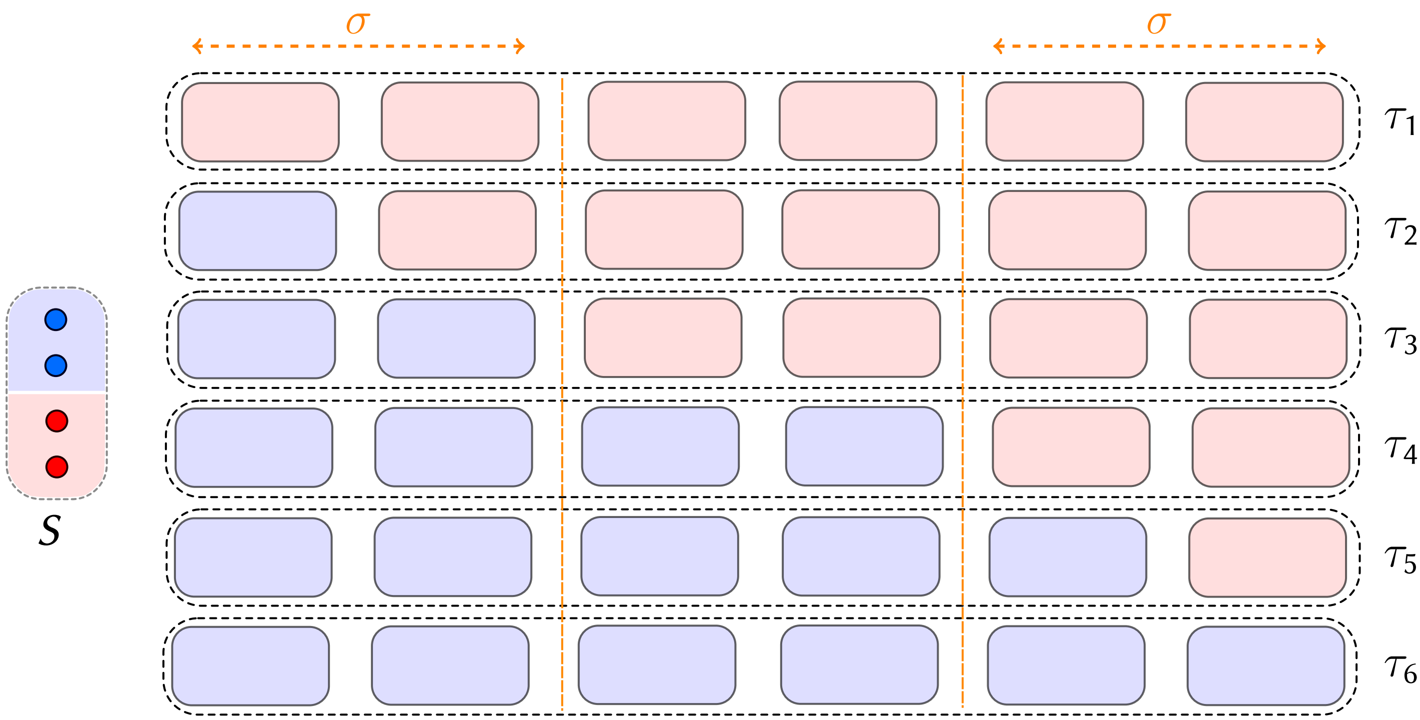


Ergo: May assume that for each τ, A takes or co-takes $\leqslant \sigma$ components.

Low rank MSO on weakly sparse classes

Idea: Inductively rewrite a low rank MSO formula to FO + conn.

$$\varphi = \exists_{A\colon \text{diversity}(A)\leqslant k}\ \psi \quad \text{where} \quad \psi \in \text{FO} + \text{conn by induction.}$$



Ergo: May assume that for each τ, A takes or co-takes $\leqslant \sigma$ components.

Such A can be quantified over in FO + conn.

Flip-connectivity logic

Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Idea: Replace connectivity in $G - \{a_1, \ldots, a_k\}$

by connectivity in a $(a_1, \ldots, a_k)$-definable flip of G.

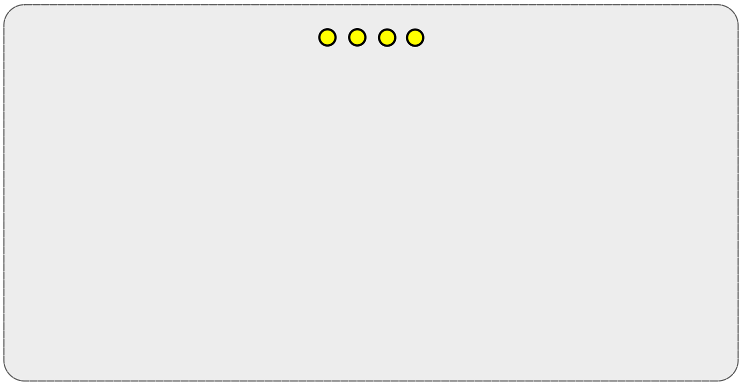
Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Idea: Replace connectivity in $G - \{a_1, \ldots, a_k\}$

by connectivity in a $(a_1, \ldots, a_k)$-definable flip of G.



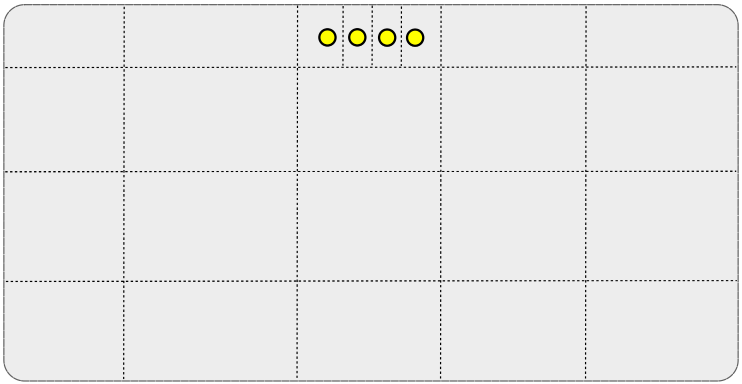
Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Idea: Replace connectivity in $G - \{a_1, \ldots, a_k\}$

by connectivity in a $(a_1, \ldots, a_k)$-definable flip of G.



Types := neighborhood types over $\bar{a} = (a_1, \ldots, a_k)$

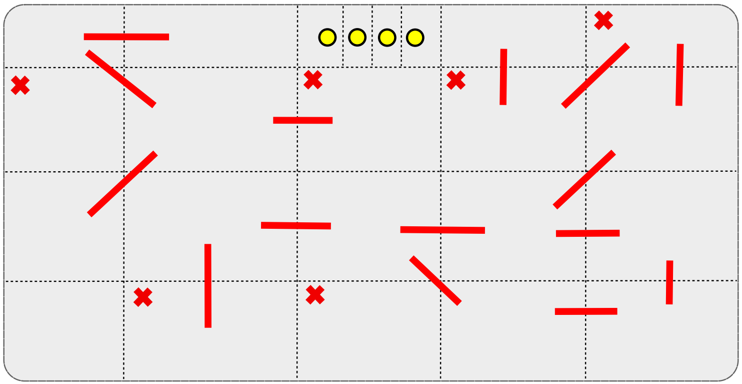
Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Idea: Replace connectivity in $G - \{a_1, \ldots, a_k\}$

by connectivity in a $(a_1, \ldots, a_k)$-definable flip of G.



Types $:=$ neighborhood types over $\bar{a} = (a_1, \ldots, a_k)$

Symmetric $\Pi \subseteq$ Types $\times$ Types $\rightsquigarrow$ flip $G \oplus \Pi$

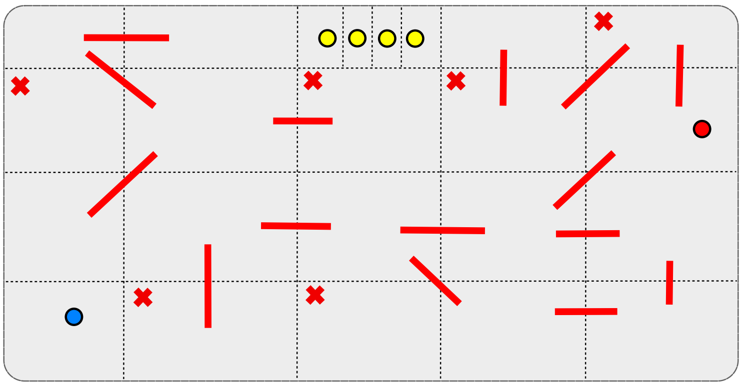
Flip-connectivity logic

Experience shows that lifting from **sparse** to **dense** is facilitated by:

vertex deletion $\rightsquigarrow$ **flip** / **subset complementation** / **perturbation**.

Idea: Replace connectivity in $G - \{a_1, \ldots, a_k\}$

by connectivity in a $(a_1, \ldots, a_k)$-definable flip of G.



Types := neighborhood types over $\bar{a} = (a_1, \ldots, a_k)$

Symmetric $\Pi \subseteq$ Types $\times$ Types $\rightsquigarrow$ flip $G \oplus \Pi$

$\mathrm{flipconn}_\Pi(u, v, a_1, \ldots, a_k)$:= Are u, v connected in $G \oplus \Pi$?

Flip-connectivity and low rank MSO

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with $\mathsf{FO} + \mathsf{flipconn}$.

Flip-connectivity and low rank MSO

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with FO + flipconn.

Lemma [Bonnet et al. '22]

Suppose G has **VC dimension** d and $\mathrm{cutrank}(A) \leq k$.

Then there is a flip $G \oplus \Pi$ definable from $f(k, d)$ vertices such that

every component of $G \oplus \Pi$ is entirely in A or entirely in $\overline{A}$.

Flip-connectivity and low rank MSO

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with FO + flipconn.

Lemma [Bonnet et al. '22]

Suppose G has **VC dimension** d and $\mathrm{cutrank}(A) \leq k$.

Then there is a flip $G \oplus \Pi$ definable from $f(k, d)$ vertices such that

every component of $G \oplus \Pi$ is entirely in A or entirely in $\overline{A}$.

Having this, apply the same strategy as for weakly sparse classes.

Flip-connectivity and low rank MSO

Theorem

Suppose $\mathscr{C}$ has **bounded VC dimension**.

Then low rank MSO coincides on $\mathscr{C}$ with FO + flipconn.

Lemma [Bonnet et al. '22]

Suppose G has **VC dimension** d and $\mathrm{cutrank}(A) \leq k$.

Then there is a flip $G \oplus \Pi$ definable from $f(k,d)$ vertices such that

every component of $G \oplus \Pi$ is entirely in A or entirely in $\overline{A}$.

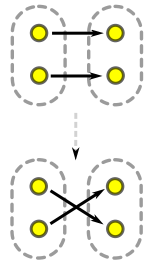
Having this, apply the same strategy as for weakly sparse classes.

Note: **Lemma** fails without the bound on **VC dimension**,

and so does the **equivalence**.

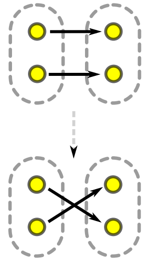
Flip-reachability

Flip-reachability



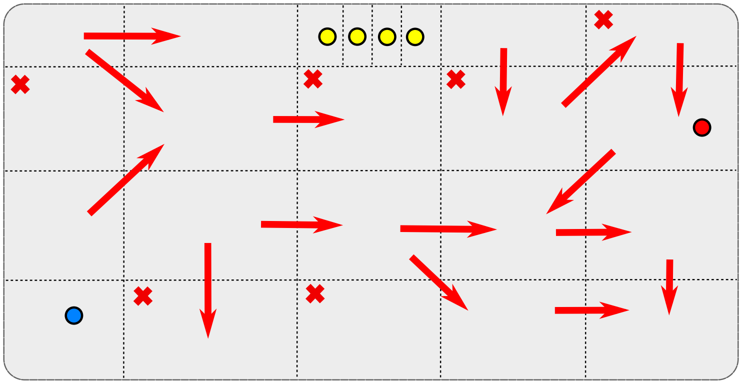
Idea: Work in directed graphs and apply **directed flips**.

Flip-reachability



Idea: Work in directed graphs and apply **directed flips**.

Definable directed flips defined analogously:

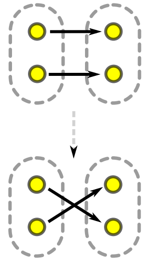


$$\text{Types} := \text{neighborhood types over } \bar{a} = (a_1, \ldots, a_k)$$

$$\Pi \subseteq \text{Types} \times \text{Types} \quad \rightsquigarrow \quad \text{flip } G \oplus \Pi$$

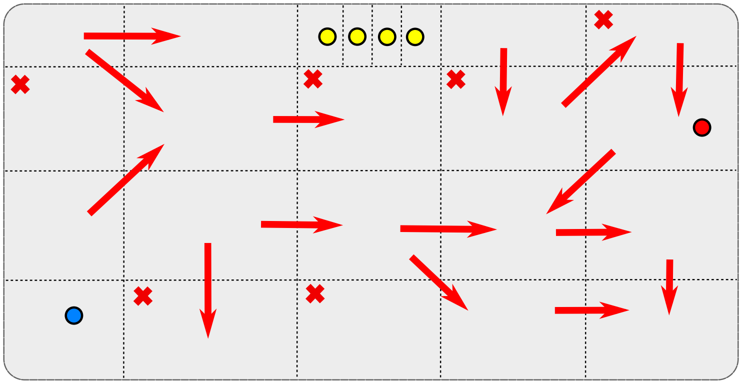
$$\text{flipconn}_\Pi(u, v, a_1, \ldots, a_k) := \text{Is } v \textbf{ reachable} \text{ from } u \text{ in } G \oplus \Pi?$$

Flip-reachability



Idea: Work in directed graphs and apply **directed flips**.

Definable directed flips defined analogously:



$$\text{Types} \coloneqq \text{neighborhood types over } \bar{a} = (a_1, \ldots, a_k)$$

$$\Pi \subseteq \text{Types} \times \text{Types} \quad \rightsquigarrow \quad \text{flip } G \oplus \Pi$$

$$\text{flipconn}_\Pi(u, v, a_1, \ldots, a_k) \coloneqq \text{Is } v \textbf{ reachable} \text{ from } u \text{ in } G \oplus \Pi?$$

Theorem

Low rank MSO coincides with FO + flipreach on **all undirected graphs**.

Compact representation of cuts

Compact representation of cuts

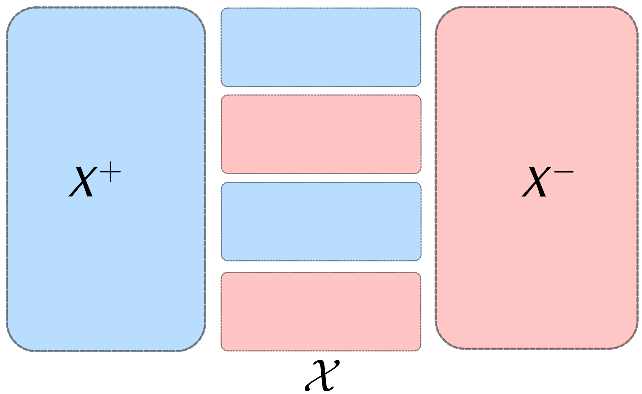
Lemma

Let G be graph, $k \in \mathbb{N}$, and $\mathcal{S}_k = \{A \subseteq V(G) \mid \text{cutrank}(A) \leqslant k\}$. Then there is a family $\mathcal{F}$ of $(X^+, X^-, \mathcal{X})$ with $\{X^+, X^-\} \cup \mathcal{X}$ a partition of $V(G)$ so that

$$\mathcal{S}_k = \bigcup_{(X^+,X^-,\mathcal{X})\in\mathcal{F}} \text{Span}(X^+, X^-, \mathcal{X}).$$

Moreover, $|\mathcal{F}| \leqslant |G|^{f(k)}$ and

every $(X^+, X^-, \mathcal{X}) \in \mathcal{F}$ is $(\text{FO} + \text{flipreach})$-definable from an $f(k)$-tuple.



Compact representation of cuts

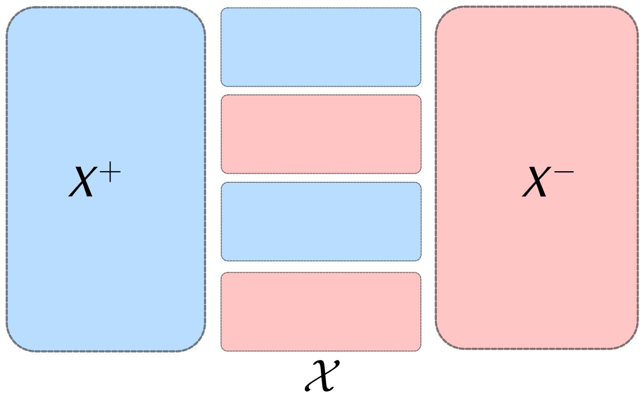
Lemma

Let G be graph, $k \in \mathbb{N}$, and $\mathcal{S}_k = \{A \subseteq V(G) \mid \mathrm{cutrank}(A) \leqslant k\}$. Then there is a family $\mathcal{F}$ of $(X^+, X^-, \mathcal{X})$ with $\{X^+, X^-\} \cup \mathcal{X}$ a partition of $V(G)$ so that

$$\mathcal{S}_k = \bigcup_{(X^+,X^-,\mathcal{X}) \in \mathcal{F}} \mathrm{Span}(X^+, X^-, \mathcal{X}).$$

Moreover, $|\mathcal{F}| \leqslant |G|^{f(k)}$ and

every $(X^+, X^-, \mathcal{X}) \in \mathcal{F}$ is $(\mathrm{FO} + \mathrm{flipreach})$-definable from an $f(k)$-tuple.



Compact XP representation of sets of cutrank $\leqslant k$.

Conclusions

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

– Quantification over sets A with bounded $\lambda(A)$.

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

- Quantification over sets A with bounded $\lambda(A)$.
- What is low rank MSO on matroids?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

– Quantification over sets A with bounded $\lambda(A)$.

– What is low rank MSO on matroids?

– Compact XP representation of sets with small λ?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

– Quantification over sets A with bounded $\lambda(A)$.

– What is low rank MSO on matroids?

– Compact XP representation of sets with small λ?

Is model-checking low rank MSO in XP on directed graphs?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

- Quantification over sets A with bounded $\lambda(A)$.
- What is low rank MSO on matroids?
- Compact XP representation of sets with small λ?

Is model-checking low rank MSO in XP on directed graphs?

Is model-checking low rank MSO in XL?

Conclusions

Conjecture

Model-checking low rank MSO is FPT

on every graph class $\mathscr{C}$ that is **mon dependent** wrt low rank MSO.

Excuse to study **mon dependent** wrt low rank MSO

as a dense analogue of **top-minor-free**.

Unbreakable decomposition for cutrank?

Low rank MSO on **matroids**?

– Quantification over sets A with bounded $\lambda(A)$.

– What is low rank MSO on matroids?

– Compact XP representation of sets with small λ?

Is model-checking low rank MSO in XP on directed graphs?

Is model-checking low rank MSO in XL?

Thanks for your attention!