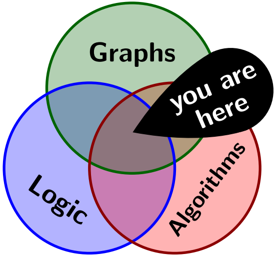
Monadically Stable and Monadically Dependent Graph Classes

Characterizations and Algorithmic Meta-Theorems

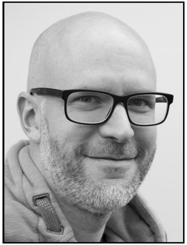
Nikolas Mählmann
previously: University of Bremen, now: University of Warsaw

CSL 2026, Ackermann Award



Acknowledgements

For their great mentorship and collaboration I want to especially thank:



Sebastian Siebertz

Jan Dreier

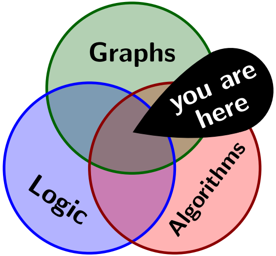
Szymon Toruńczyk

Michał Pilipczuk

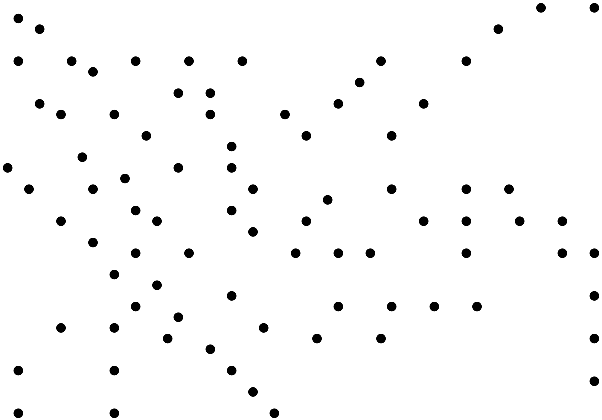
Monadically Stable and Monadically Dependent Graph Classes

Characterizations and Algorithmic Meta-Theorems

Nikolas Mählmann
previously: University of Bremen, now: University of Warsaw
CSL 2026, Ackermann Award

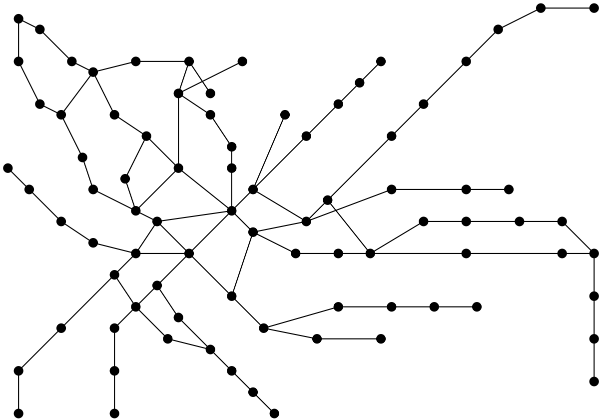


Graphs



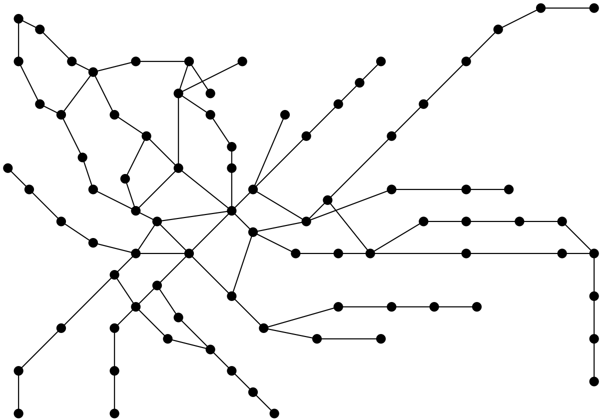
A *graph* consists of *vertices* connected by *edges*.

Graphs



A *graph* consists of *vertices* connected by *edges*.

Graphs

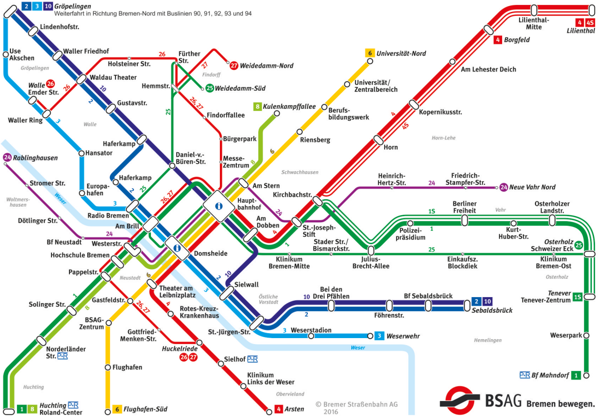


A *graph* consists of *vertices* connected by *edges*.

Graphs are an effective way to model real systems:

- road networks
- power grids
- computer networks
- circuits
- molecules

Graphs

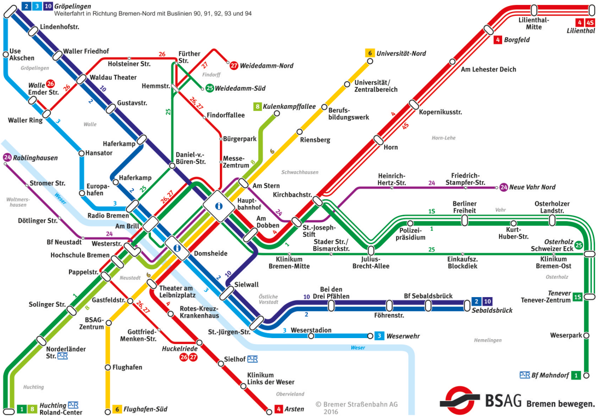


A *graph* consists of *vertices* connected by *edges*.

Graphs are an effective way to model real systems:

- road networks
- power grids
- computer networks
- circuits
- molecules

Graphs

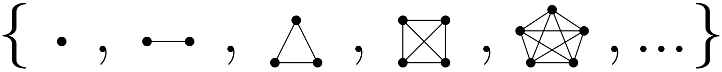


A *graph* consists of *vertices* connected by *edges*.

Graphs are an effective way to model real systems:

- road networks
- power grids
- computer networks
- circuits
- molecules

A *graph class* is a (usually infinite) set of graphs. Example:



The FO Model Checking Problem

Problem: Given a graph G and an FO sentence φ, decide whether

$$G \models \varphi.$$

Example: G contains a dominating set of size k iff.

$$G \models \exists x_1 \ldots \exists x_k \forall y : \bigvee_{i \in [k]} \big(y = x_i \vee \mathrm{Edge}(y, x_i)\big).$$

The FO Model Checking Problem

Problem: Given a graph G and an FO sentence φ, decide whether

$$G \models \varphi.$$

Example: G contains a dominating set of size k iff.

$$G \models \exists x_1 \ldots \exists x_k \forall y : \bigvee_{i \in [k]} \big(y = x_i \vee \mathrm{Edge}(y, x_i)\big).$$

Further expressible problems: Independent Set, Subgraph Isomorphism, Independent Red-Blue Distance-7 Dominating Set, ...

The FO Model Checking Problem

Problem: Given a graph G and an FO sentence φ, decide whether

$$G \models \varphi.$$

Example: G contains a dominating set of size k iff.

$$G \models \exists x_1 \dots \exists x_k \forall y : \bigvee_{i \in [k]} \big(y = x_i \vee \mathrm{Edge}(y, x_i)\big).$$

Further expressible problems: Independent Set, Subgraph Isomorphism, Independent Red-Blue Distance-7 Dominating Set, ...

Runtime: For the class of *all graphs* the best possible running time is $n^{\mathcal{O}(|\varphi|)}$.

($n = \#$vertices G; assuming ETH)

The FO Model Checking Problem

Problem: Given a graph G and an FO sentence φ, decide whether

$$G \models \varphi.$$

Example: G contains a dominating set of size k iff.

$$G \models \exists x_1 \ldots \exists x_k \forall y : \bigvee_{i \in [k]} \big(y = x_i \vee \mathrm{Edge}(y, x_i)\big).$$

Further expressible problems: Independent Set, Subgraph Isomorphism, Independent Red-Blue Distance-7 Dominating Set, ...

Runtime: For the class of *all graphs* the best possible running time is $n^{\mathcal{O}(|\varphi|)}$.

($n = \#$vertices G; assuming ETH)

Question: On which classes is model checking fpt, i.e., solvable in time $f(|\varphi|) \cdot n^c$?

Nowhere Dense Classes of Graphs

For sparse graph classes, we know the exact limits of tractability.

Theorem [Grohe, Kreutzer, Siebertz, 2014]

Let $\mathcal{C}$ be a *monotone* graph class.

- If $\mathcal{C}$ is *nowhere dense*, then model checking is fixed-parameter tractable on $\mathcal{C}$.
- Otherwise model checking is AW[$*$]-hard on $\mathcal{C}$.

Nowhere denseness generalizes: bounded degree, bounded tree-width, planarity, excluding a minor, ...

Monotone and Hereditary Graph Classes

The class of all cliques is not nowhere dense, but model checking is trivial there.

Monotone and Hereditary Graph Classes

The class of all cliques is not nowhere dense, but model checking is trivial there.

Cliques are not *monotone*: closed under taking subgraphs.
(i.e., deleting vertices and edges)

Monotone and Hereditary Graph Classes

The class of all cliques is not nowhere dense, but model checking is trivial there.

Cliques are not *monotone*: closed under taking subgraphs.
(i.e., deleting vertices and edges)

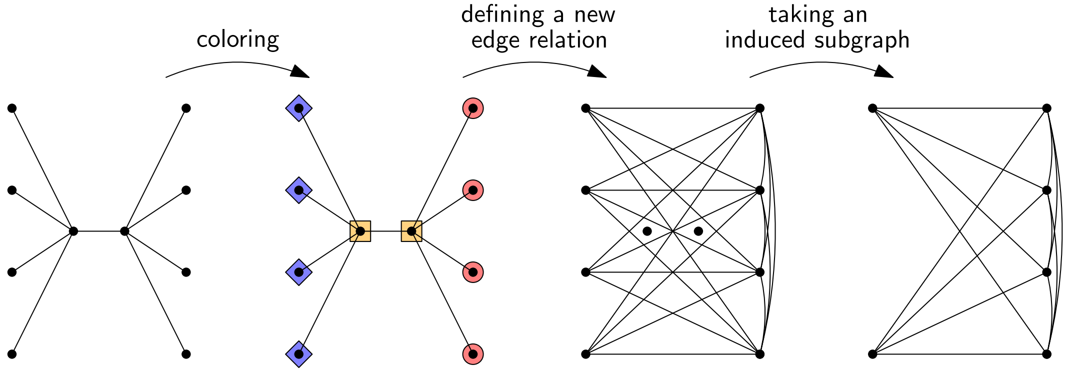
But cliques are *hereditary*: closed under taking induced subgraphs.
(i.e., deleting vertices)

To go beyond sparse classes, we need to shift from monotone to hereditary classes.

Transductions

Transductions are graph transformations defined by FO logic.

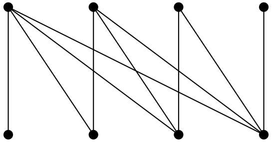
Example: $\varphi(x, y) = (\mathrm{dist}(x, y) = 3) \vee (\mathrm{Red}(x) \wedge \mathrm{Red}(y))$



Monadic Stability and Monadic Dependence

Definition

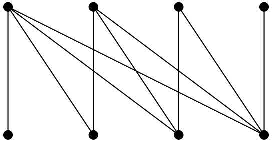
A class is *monadically stable*, if it does not transduce the class of all half graphs.



Monadic Stability and Monadic Dependence

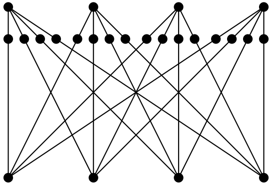
Definition

A class is *monadically stable*, if it does not transduce the class of all half graphs.

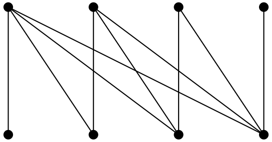


Definition

A class is *monadically dependent*, if it does not transduce the class of all graphs.

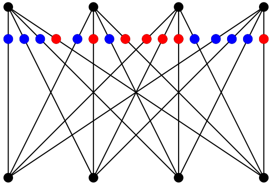


Monadic Stability and Monadic Dependence



Definition

A class is *monadically stable*, if it does not transduce the class of all half graphs.



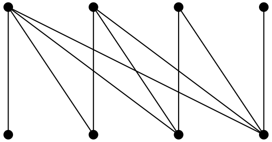
Definition

A class is *monadically dependent*, if it does not transduce the class of all graphs.

Monadic Stability and Monadic Dependence

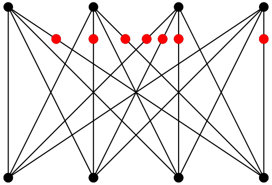
Definition

A class is *monadically stable*, if it does not transduce the class of all half graphs.



Definition

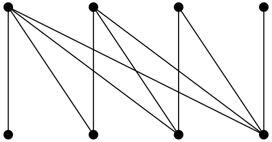
A class is *monadically dependent*, if it does not transduce the class of all graphs.



Monadic Stability and Monadic Dependence

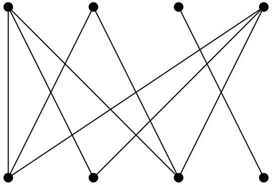
Definition

A class is *monadically stable*, if it does not transduce the class of all half graphs.



Definition

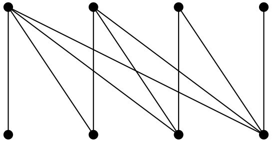
A class is *monadically dependent*, if it does not transduce the class of all graphs.



Monadic Stability and Monadic Dependence

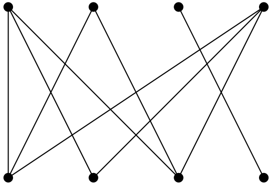
Definition

A class is *monadically stable*, if it does not transduce the class of all half graphs.



Definition

A class is *monadically dependent*, if it does not transduce the class of all graphs.



nowhere dense $\subsetneq$ monadically stable $\subsetneq$ monadically dependent

Algorithmic Results

The Model Checking Conjecture

Let $\mathcal{C}$ be a hereditary graph class.

- $\mathcal{C}$ is monadically dependent $\Rightarrow$ model checking is efficient on $\mathcal{C}$.
- $\mathcal{C}$ is monadically independent $\Rightarrow$ model checking is hard on $\mathcal{C}$.

Algorithmic Results

The Model Checking Conjecture

Let $\mathcal{C}$ be a hereditary graph class.

- $\mathcal{C}$ is monadically dependent $\Rightarrow$ model checking is efficient on $\mathcal{C}$.
- $\mathcal{C}$ is monadically independent $\Rightarrow$ model checking is hard on $\mathcal{C}$.

We show:

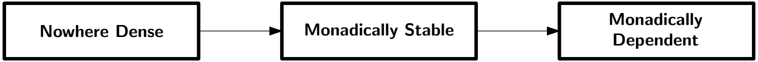
Theorem [Dreier, **NM**, Siebertz, 2023], [Dreier, Eleftheriadis, **NM**, McCarty, Pilipczuk, Toruńczyk, 2024]

$\mathcal{C}$ is monadically stable $\Rightarrow$ model checking is solvable in time $f(|\varphi|) \cdot n^{6.0001}$ on $\mathcal{C}$.

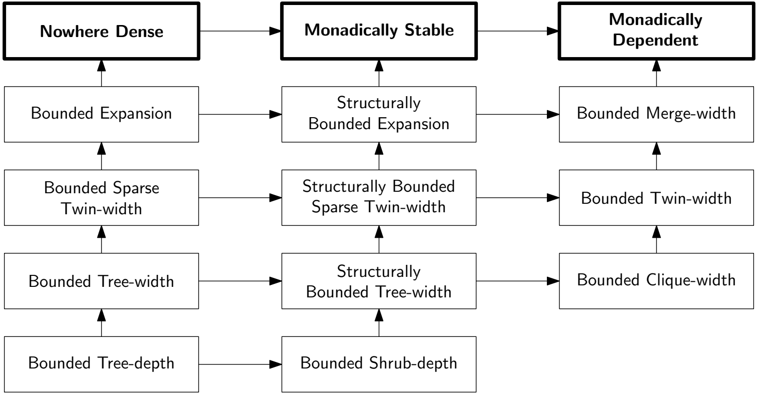
Theorem [Dreier, **NM**, Toruńczyk, 2024]

$\mathcal{C}$ is hereditary and monadically independent $\Rightarrow$ model checking is $\mathrm{AW}[*]$-hard on $\mathcal{C}$.

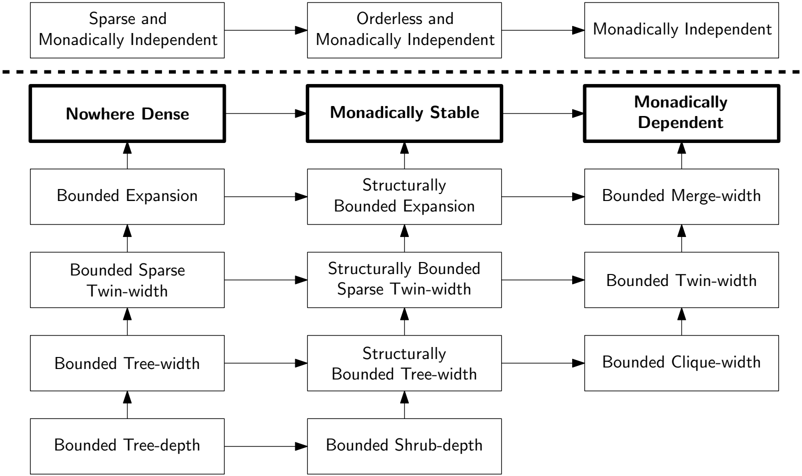
Model Checking in Hereditary Graph Classes



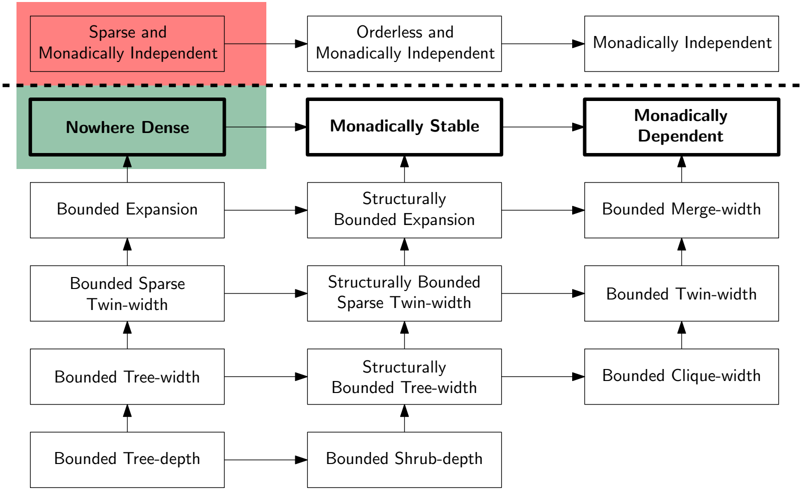
Model Checking in Hereditary Graph Classes



Model Checking in Hereditary Graph Classes

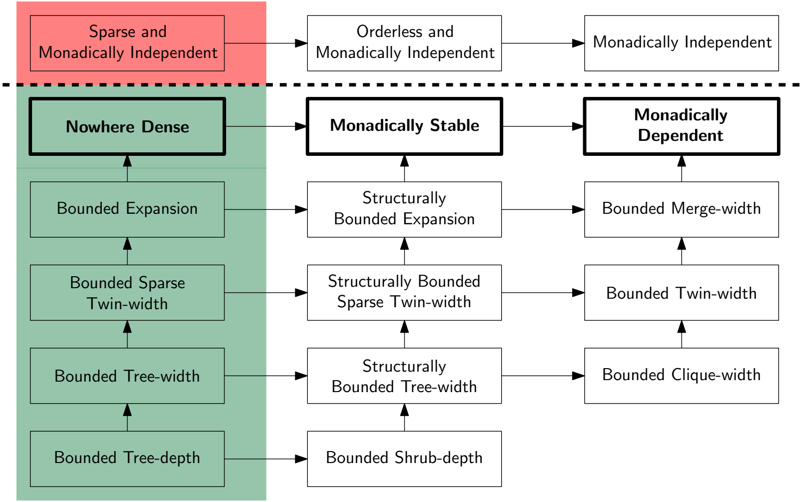


Model Checking in Hereditary Graph Classes



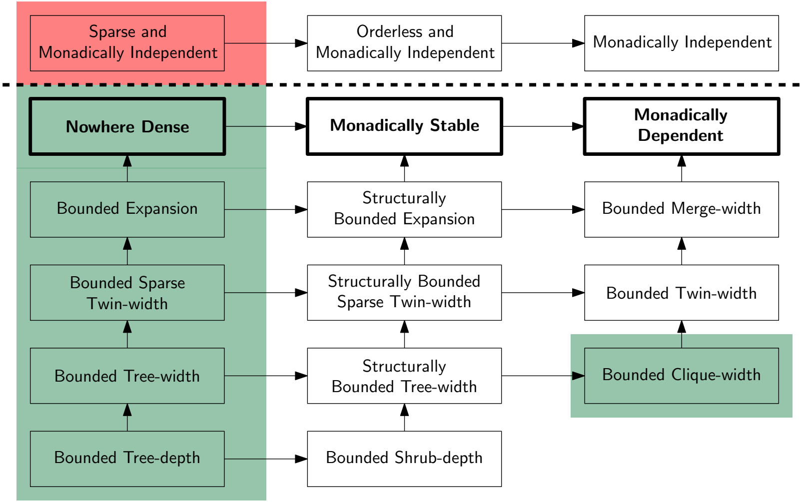
Sources: nowhere dense [Grohe, Kreutzer, Siebertz, 2014]

Model Checking in Hereditary Graph Classes



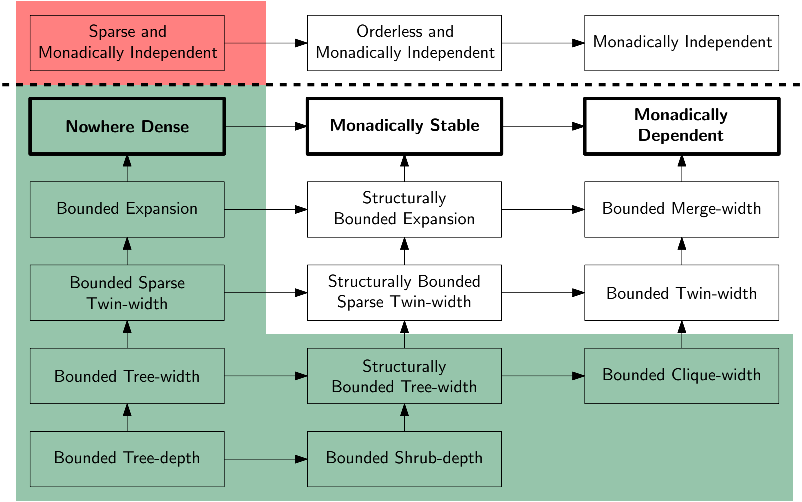
Sources: nowhere dense [Grohe, Kreutzer, Siebertz, 2014]

Model Checking in Hereditary Graph Classes



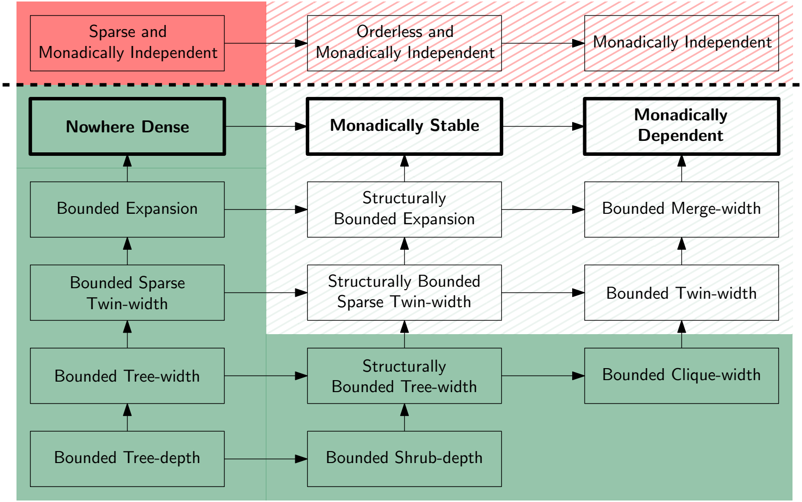
Sources: clique-width [Courcelle, Makowsky, Rotics, 2000]

Model Checking in Hereditary Graph Classes

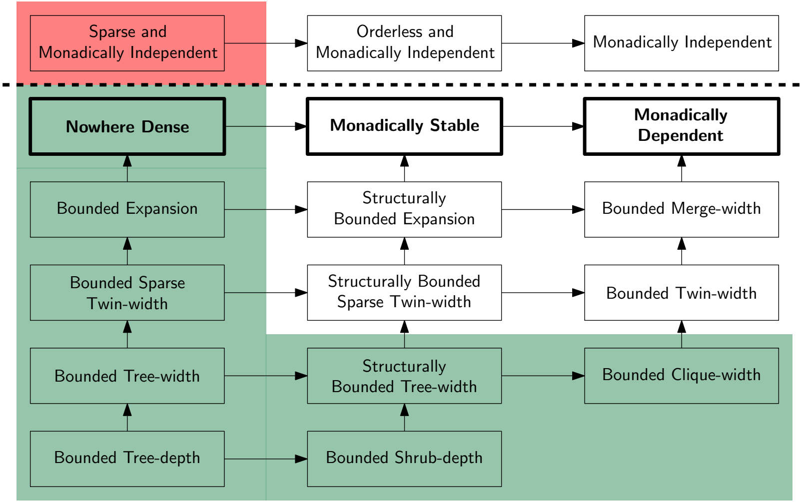


Sources: clique-width [Courcelle, Makowsky, Rotics, 2000]

Conjectured Tractability Limits

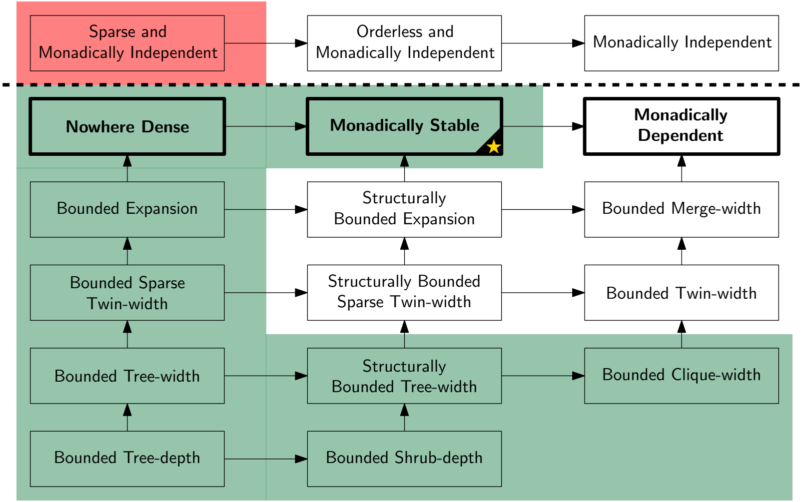


Our Results ★



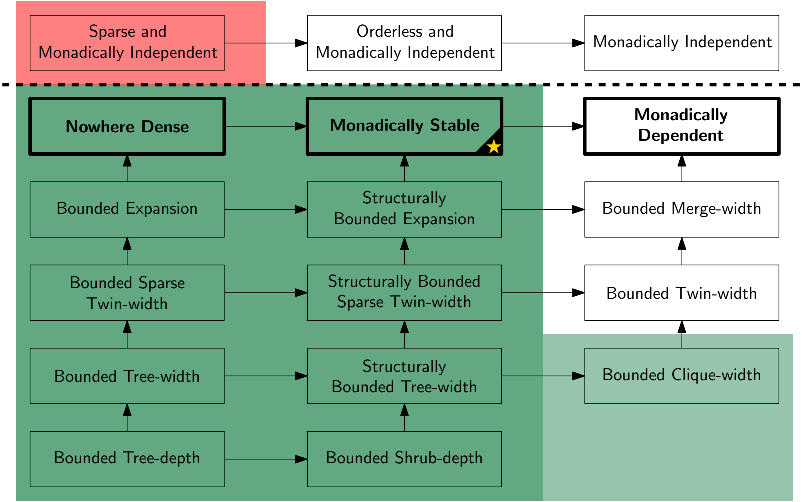
Sparse and Monadically Independent
Orderless and Monadically Independent
Monadically Independent
Nowhere Dense
Monadically Stable
Monadically Dependent
Bounded Expansion
Structurally Bounded Expansion
Bounded Merge-width
Bounded Sparse Twin-width
Structurally Bounded Sparse Twin-width
Bounded Twin-width
Bounded Tree-width
Structurally Bounded Tree-width
Bounded Clique-width
Bounded Tree-depth
Bounded Shrub-depth

Our Results ★



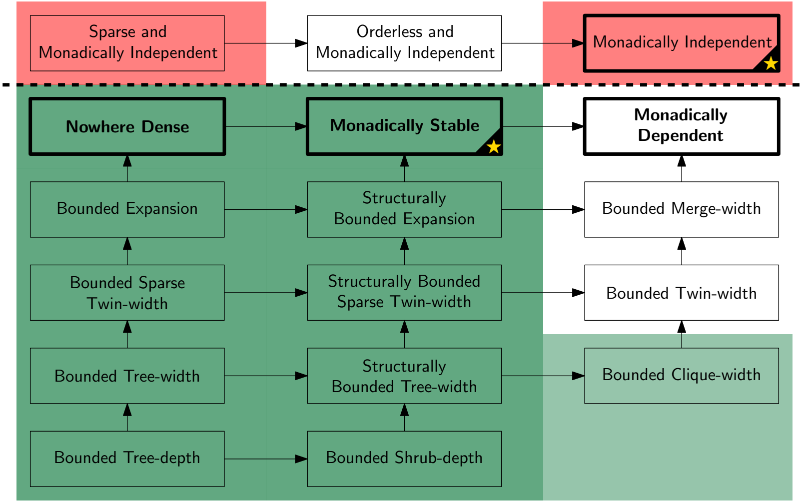
Sources: stable [Dreier, **NM**, Siebertz, 2023], [Dreier, Eleftheriadis, **NM**, McCarty, Pilipczuk, Toruńczyk, 2024]

Our Results ★



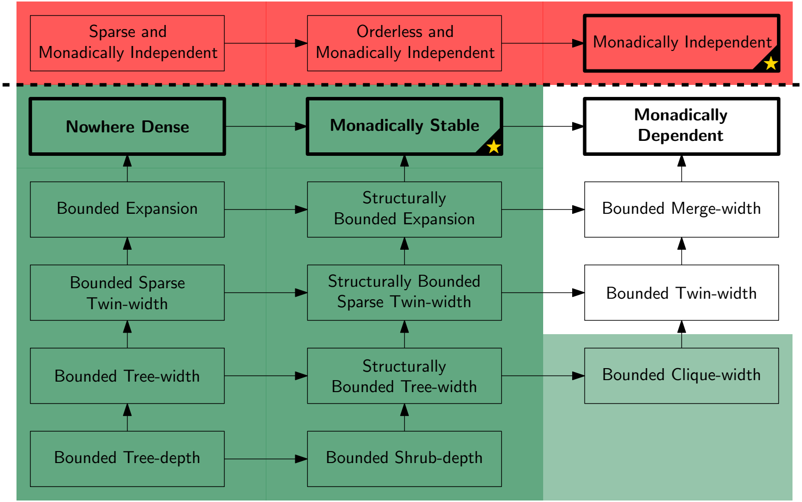
Sources: stable [Dreier, **NM**, Siebertz, 2023], [Dreier, Eleftheriadis, **NM**, McCarty, Pilipczuk, Toruńczyk, 2024]

Our Results ★



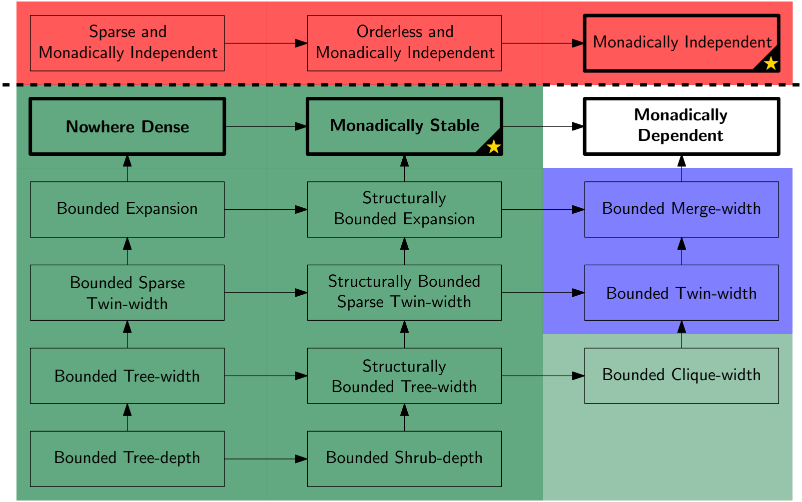
Sources: monadically independent [Dreier, **NM**, Toruńczyk, 2024]

Our Results ★



Sources: monadically independent [Dreier, **NM**, Toruńczyk, 2024]

Conditional Results



Sources: twin-width [Bonnet, Kim, Thomassé, Watrigant, 2020], merge-width [Dreier, Toruńczyk, 2025]

Combinatorial Results

Monadic stability and dependence are defined through **logic**.

Combinatorial Results

Monadic stability and dependence are defined through **logic**.

Main part of the thesis: **combinatorial** characterizations:

Combinatorial Results

Monadic stability and dependence are defined through **logic**.

Main part of the thesis: **combinatorial** characterizations:

1. **Ramsey properties**:

		Flatness	Breakability
dist-r	Flip-	**monadic stability**	**monadic dependence**
	Deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	Flip-	**shrub-depth**	**clique-width**
	Deletion-	**tree-depth**	**tree-width**

Combinatorial Results

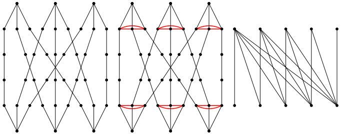
Monadic stability and dependence are defined through **logic**.

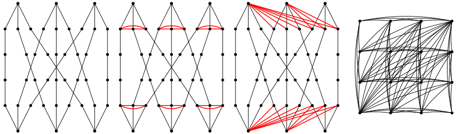
Main part of the thesis: **combinatorial** characterizations:

1. **Ramsey properties**:

		Flatness	Breakability
dist-r	Flip-	**monadic stability**	**monadic dependence**
	Deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	Flip-	**shrub-depth**	**clique-width**
	Deletion-	**tree-depth**	**tree-width**

2. **Forbidden induced subgraphs**:





Combinatorial Results

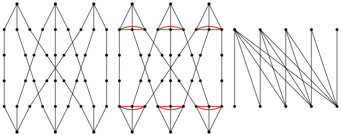
Monadic stability and dependence are defined through **logic**.

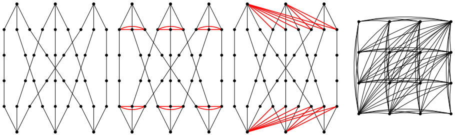
Main part of the thesis: **combinatorial** characterizations:

1. **Ramsey properties**:

		Flatness	Breakability
dist-r	Flip-	**monadic stability**	**monadic dependence**
	Deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	Flip-	**shrub-depth**	**clique-width**
	Deletion-	**tree-depth**	**tree-width**

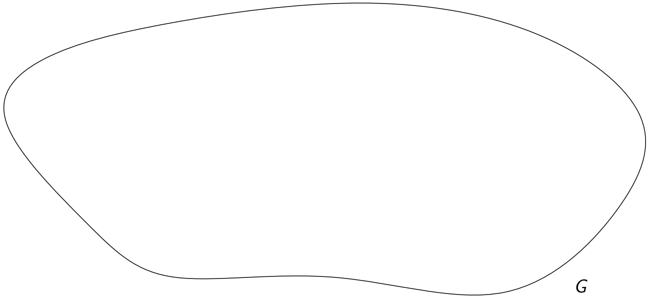
2. **Forbidden induced subgraphs**:



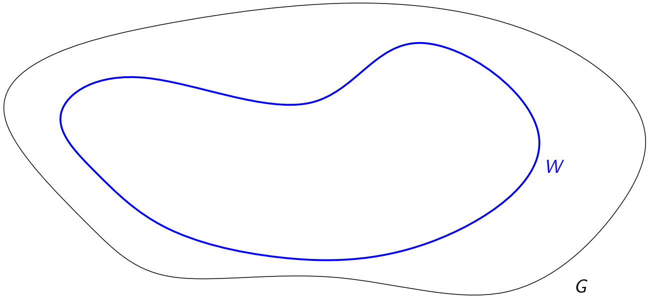


3. **The flipper game** (only for monadic stability)

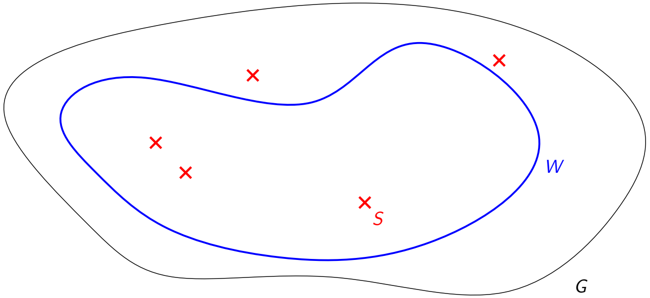
Characterizing Nowhere Denseness: Uniform Quasi-Wideness



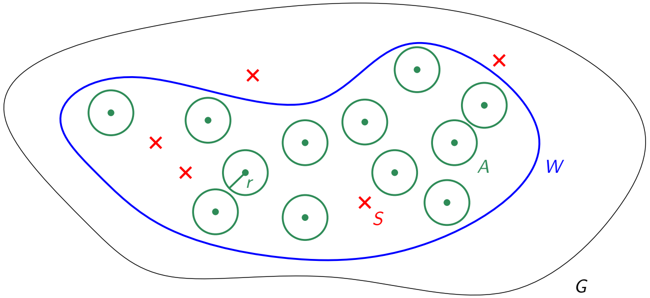
Characterizing Nowhere Denseness: Uniform Quasi-Wideness



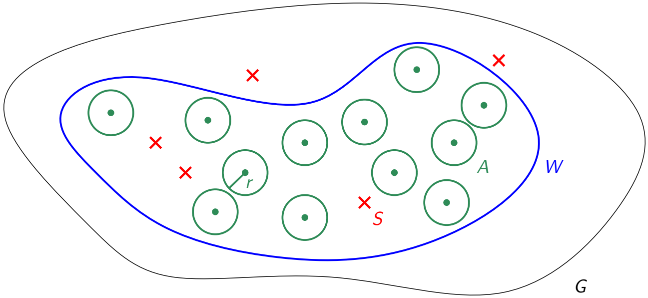
Characterizing Nowhere Denseness: Uniform Quasi-Wideness



Characterizing Nowhere Denseness: Uniform Quasi-Wideness



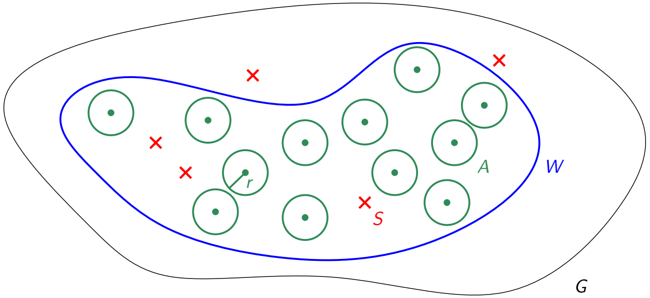
Characterizing Nowhere Denseness: Uniform Quasi-Wideness



Uniform Quasi-Wideness (slightly informal)

A class $\mathcal{C}$ is *uniformly quasi-wide* if for every radius r, in every large set W we find a still large set A that is r-independent after removing a set S of constantly many vertices.

Characterizing Nowhere Denseness: Uniform Quasi-Wideness



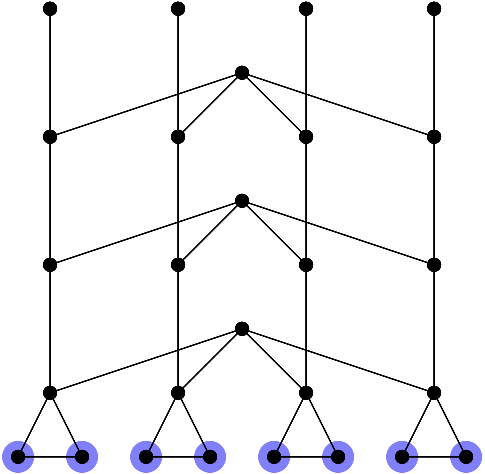
Uniform Quasi-Wideness (slightly informal)

A class $\mathcal{C}$ is *uniformly quasi-wide* if for every radius r, in every large set W we find a still large set A that is r-independent after removing a set S of constantly many vertices.

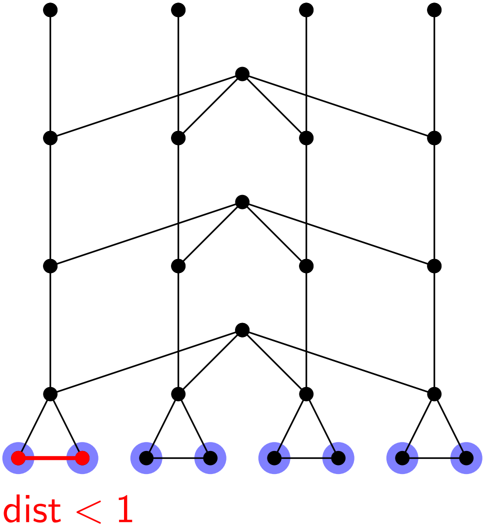
Theorem [Něsetřil, Ossona de Mendez, 2011]

A class $\mathcal{C}$ is uniformly quasi-wide if and only if it is nowhere dense.

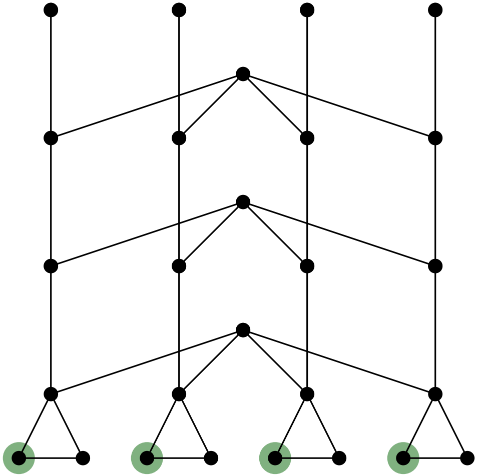
Uniform Quasi-Wideness: Example



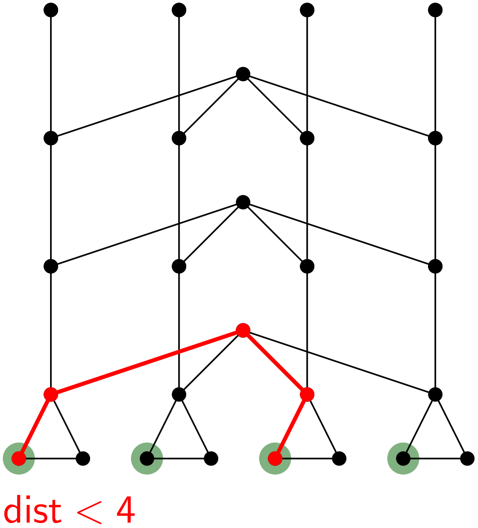
Uniform Quasi-Wideness: Example



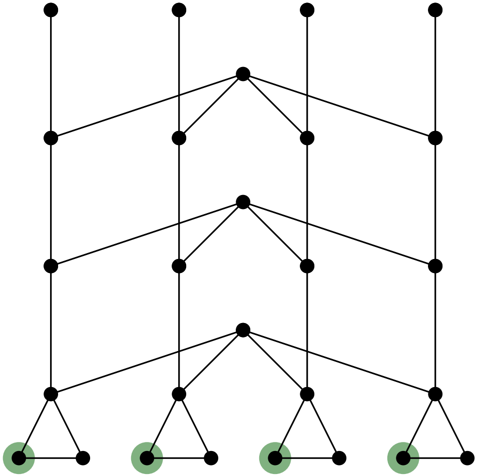
Uniform Quasi-Wideness: Example



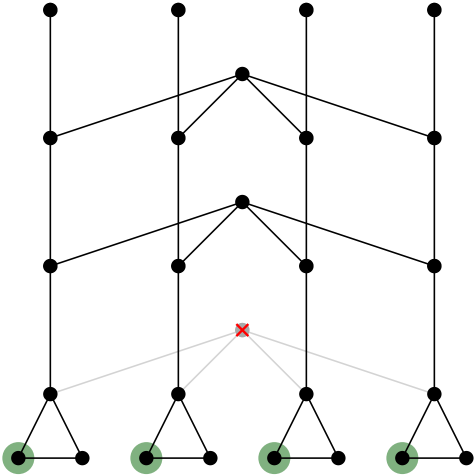
Uniform Quasi-Wideness: Example



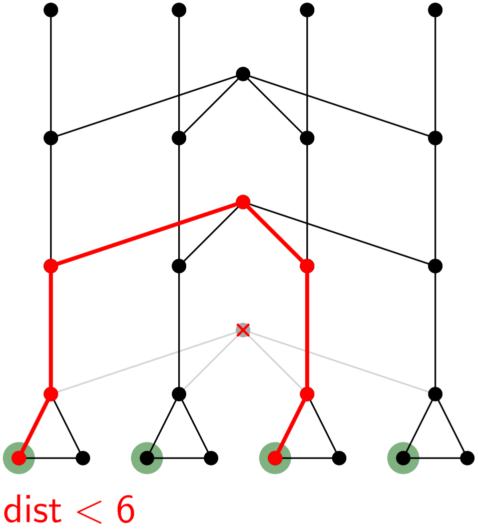
Uniform Quasi-Wideness: Example



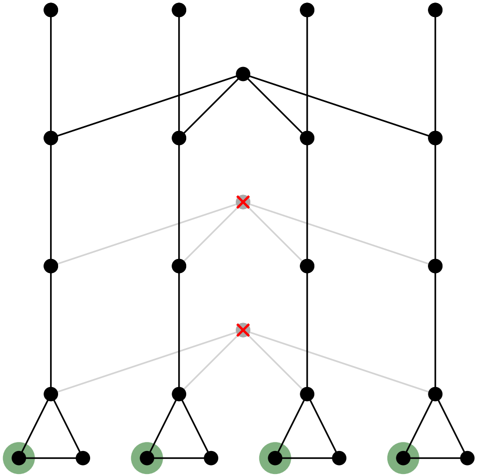
Uniform Quasi-Wideness: Example



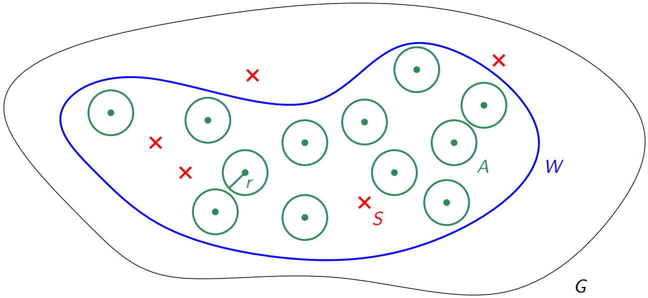
Uniform Quasi-Wideness: Example



Uniform Quasi-Wideness: Example



Characterizing Nowhere Denseness: Uniform Quasi-Wideness



Uniform Quasi-Wideness (slightly informal)

A class $\mathcal{C}$ is *uniformly quasi-wide* if for every radius r, in every large set W we find a still large set A that is r-independent after removing a set S of constantly many vertices.

Theorem [Něsetřil, Ossona de Mendez, 2011]

A class $\mathcal{C}$ is uniformly quasi-wide if and only if it is nowhere dense.

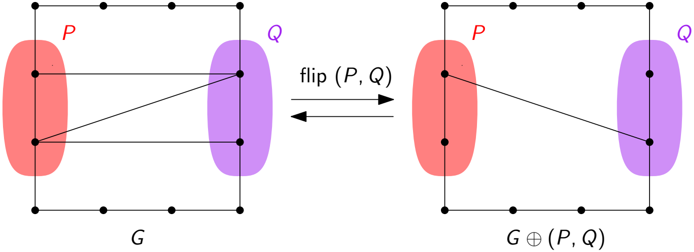
Towards Dense Graphs

Question: Is there a similar characterization for monadic stability/dependence?

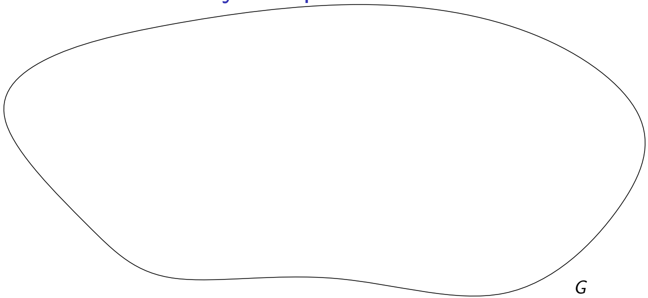
Towards Dense Graphs

Question: Is there a similar characterization for monadic stability/dependence?

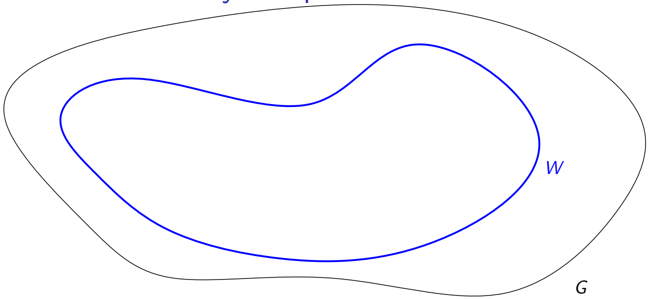
Denote by $G \oplus (P, Q)$ the graph obtained from G by complementing edges between pairs of vertices from $P \times Q$.



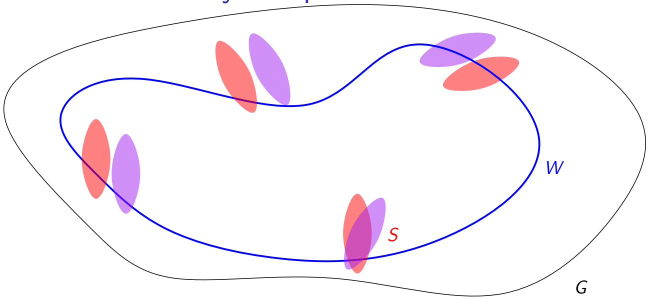
Characterizing Monadic Stability: Flip-Flatness



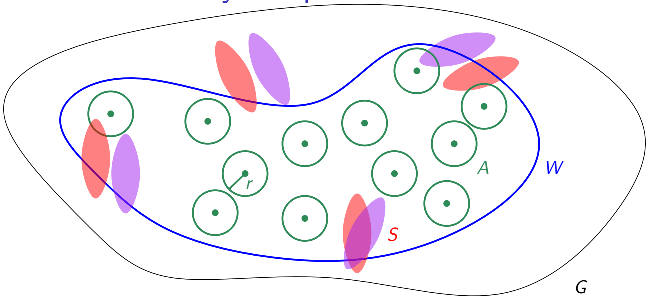
Characterizing Monadic Stability: Flip-Flatness



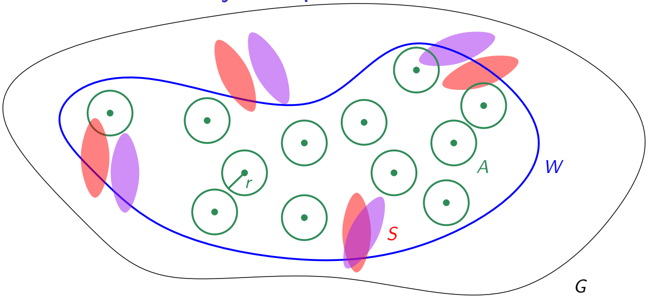
Characterizing Monadic Stability: Flip-Flatness



Characterizing Monadic Stability: Flip-Flatness



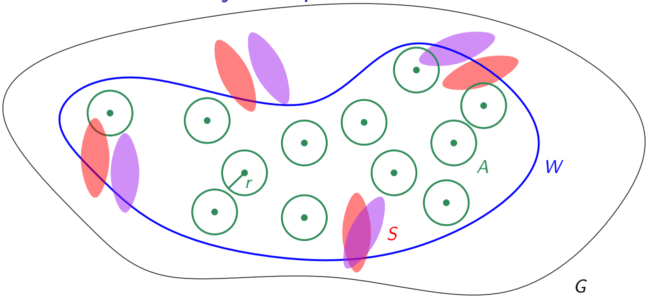
Characterizing Monadic Stability: Flip-Flatness



Flip-Flatness (slightly informal)

A class $\mathcal{C}$ is *flip-flat* if for every radius r, in every large set W we find a still large set A that is r-independent after performing a set S of constantly many flips.

Characterizing Monadic Stability: Flip-Flatness



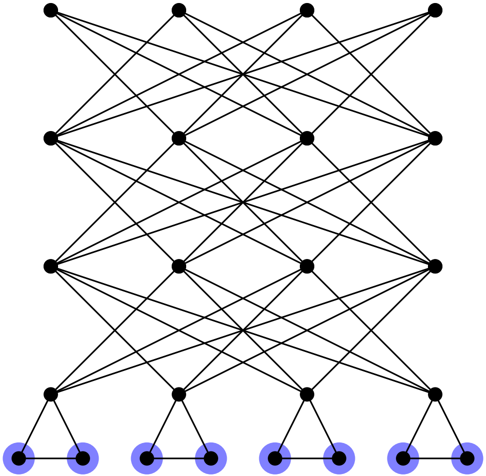
Flip-Flatness (slightly informal)

A class $\mathcal{C}$ is *flip-flat* if for every radius r, in every large set W we find a still large set A that is r-independent after performing a set S of constantly many flips.

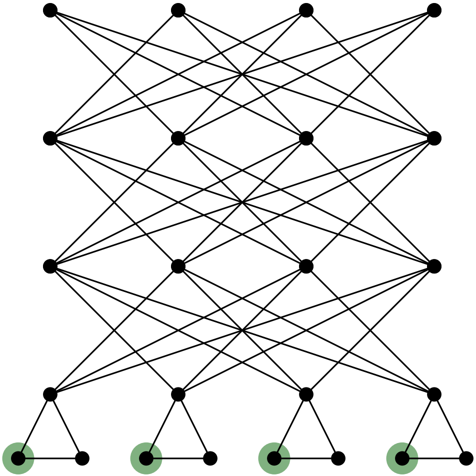
Theorem [Dreier, **NM**, Siebertz, Toruńczyk, 2023]

A class $\mathcal{C}$ is flip-flat if and only if it is monadically stable.

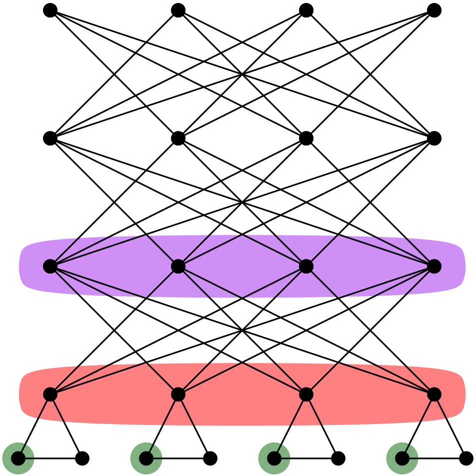
Flip-Flatness: Example



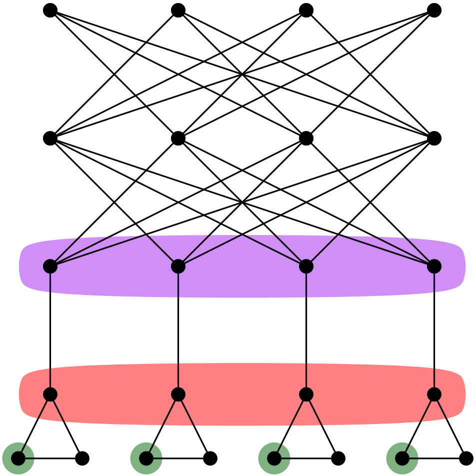
Flip-Flatness: Example



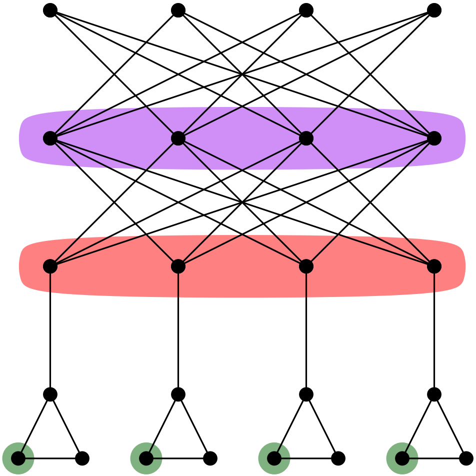
Flip-Flatness: Example



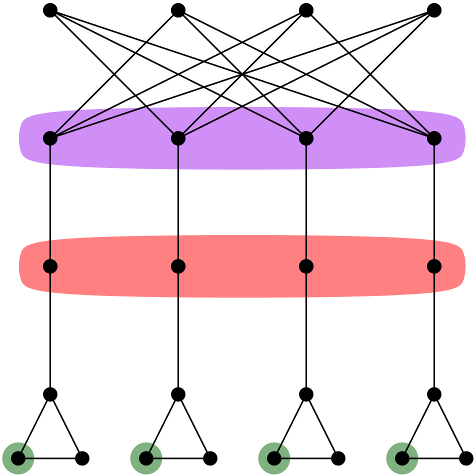
Flip-Flatness: Example



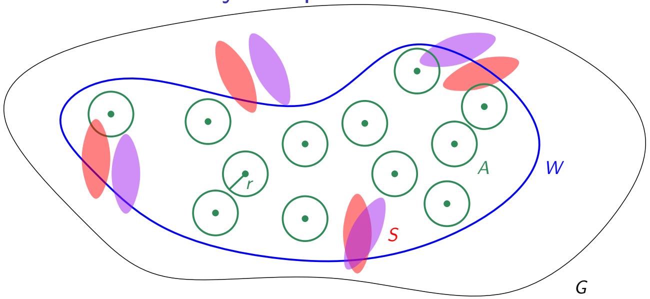
Flip-Flatness: Example



Flip-Flatness: Example



Characterizing Monadic Stability: Flip-Flatness



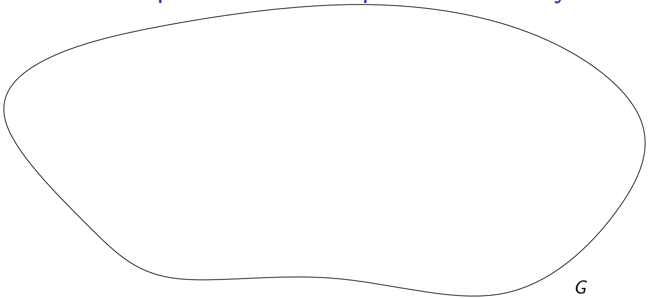
Flip-Flatness (slightly informal)

A class $\mathcal{C}$ is *flip-flat* if for every radius r, in every large set W we find a still large set A that is r-independent after performing a set S of constantly many flips.

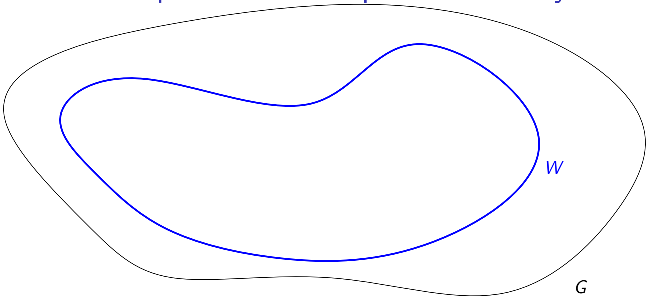
Theorem [Dreier, **NM**, Siebertz, Toruńczyk, 2023]

A class $\mathcal{C}$ is flip-flat if and only if it is monadically stable.

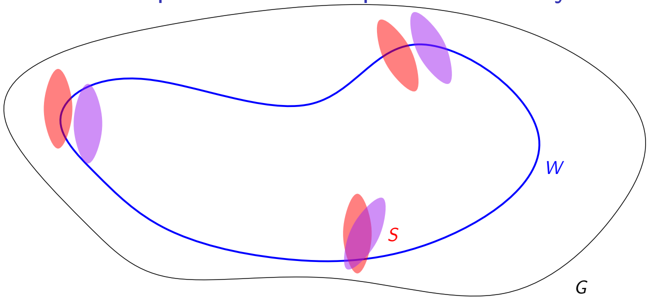
Characterizing Monadic Dependence: Flip-Breakability



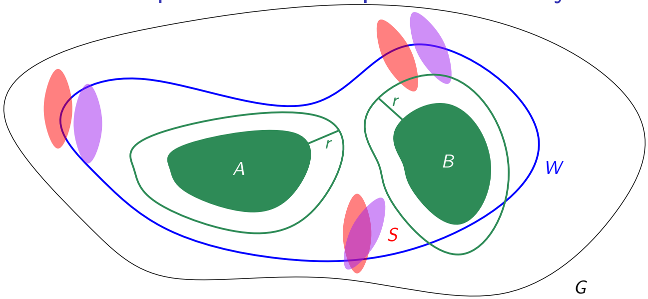
Characterizing Monadic Dependence: Flip-Breakability



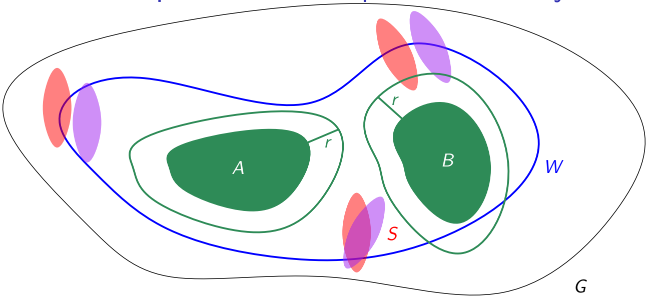
Characterizing Monadic Dependence: Flip-Breakability



Characterizing Monadic Dependence: Flip-Breakability



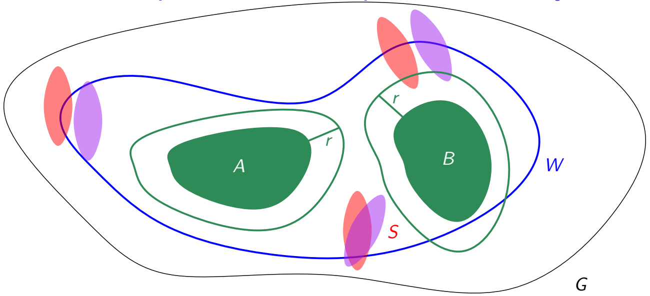
Characterizing Monadic Dependence: Flip-Breakability



Flip-Breakability (slightly informal)

A class $\mathcal{C}$ is *flip-breakable* if for every radius r, in every large set W we find two large sets A and B that are at distance greater than $2r$ from each other after performing a set S of constantly many flips.

Characterizing Monadic Dependence: Flip-Breakability



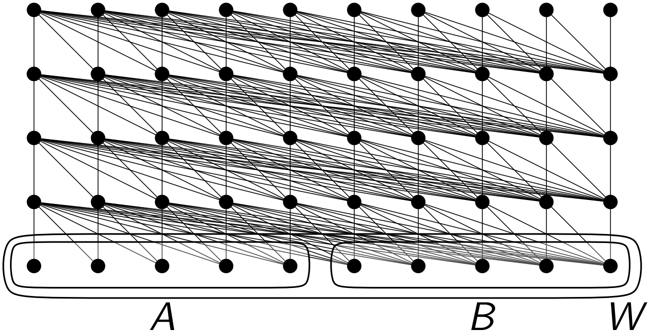
Flip-Breakability (slightly informal)

A class $\mathcal{C}$ is *flip-breakable* if for every radius r, in every large set W we find two large sets A and B that are at distance greater than $2r$ from each other after performing a set S of constantly many flips.

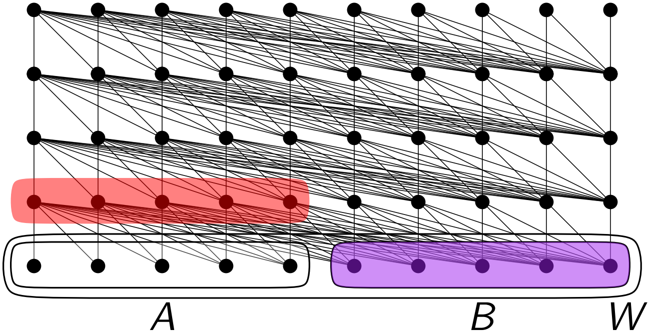
Theorem [Dreier, **NM**, Toruńczyk, 2024]

A class $\mathcal{C}$ is flip-breakable if and only if it is monadically dependent.

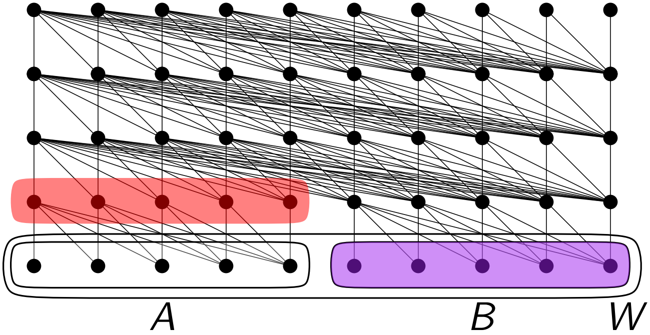
Flip-Breakability: Example



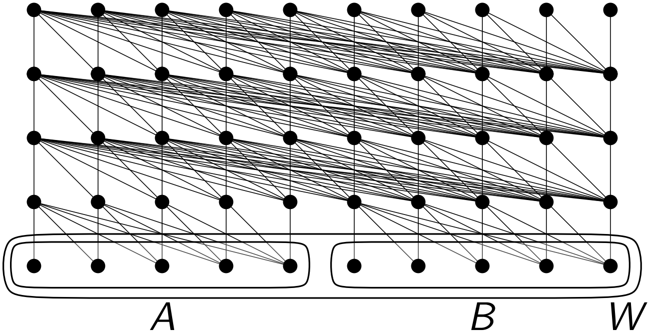
Flip-Breakability: Example



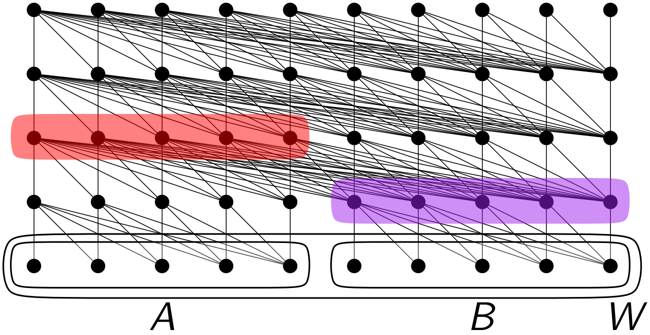
Flip-Breakability: Example



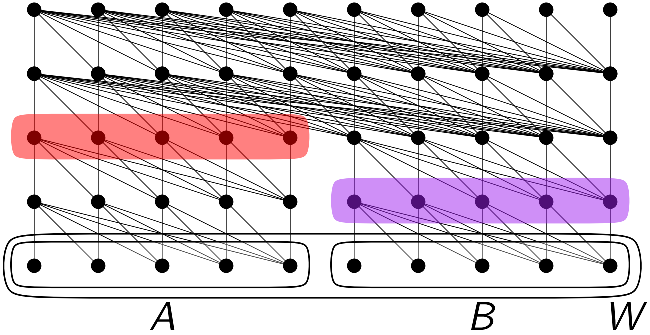
Flip-Breakability: Example



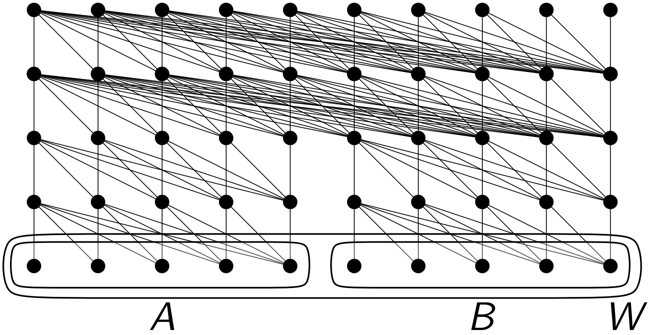
Flip-Breakability: Example



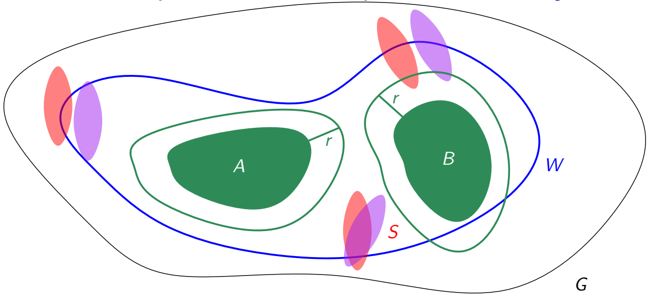
Flip-Breakability: Example



Flip-Breakability: Example



Characterizing Monadic Dependence: Flip-Breakability



Flip-Breakability (slightly informal)

A class $\mathcal{C}$ is *flip-breakable* if for every radius r, in every large set W we find two large sets A and B that are at distance greater than $2r$ from each other after performing a set S of constantly many flips.

Theorem [Dreier, **NM**, Toruńczyk, 2024]

A class $\mathcal{C}$ is flip-breakable if and only if it is monadically dependent.

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

		flatness	breakability
dist-r	flip-	**monadic stability**	**monadic dependence**
	deletion-	**nowhere denseness**	
dist-∞	flip-		
	deletion-		

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

		flatness	breakability
dist-r	flip-	**monadic stability**	**monadic dependence**
	deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	flip-		
	deletion-		

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

		flatness	breakability
dist-r	flip-	**monadic stability**	**monadic dependence**
	deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	flip-	**bd. shrub-depth**	**bd. clique-width**
	deletion-		

Variants of Flip-Breakability

1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

		flatness	breakability
dist-r	flip-	**monadic stability**	**monadic dependence**
	deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	flip-	**bd. shrub-depth**	**bd. clique-width**
	deletion-	**bd. tree-depth**	**bd. tree-width**

Variants of Flip-Breakability

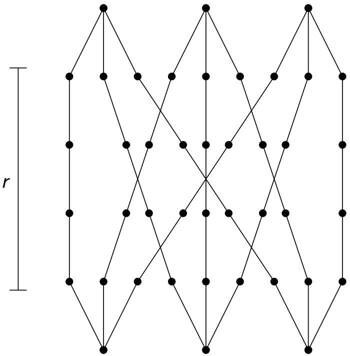
1. We modify a graph using either flips or vertex deletions.
2. We demand our resulting set is either flat or broken.

   flat: pairwise separated; broken: separated into two large sets
3. Separation means either distance-r or distance-∞.

		flatness	breakability
dist-r	flip-	**monadic stability**	**monadic dependence**
	deletion-	**nowhere denseness**	**nowhere denseness**
dist-∞	flip-	**bd. shrub-depth**	**bd. clique-width**
	deletion-	**bd. tree-depth**	**bd. tree-width**

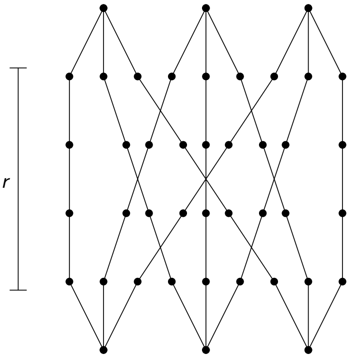
Ramsey-theoretic characterization ✓ next up: forbidden induced subgraphs

Characterizing Monadic Dependence by Forbidden Induced Subgraphs

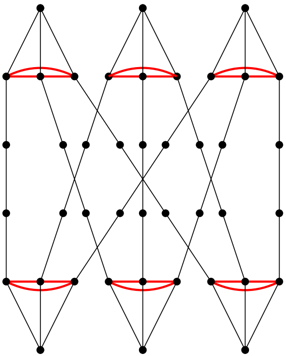


star r-crossing
$= r$-subdivided biclique

Characterizing Monadic Dependence by Forbidden Induced Subgraphs

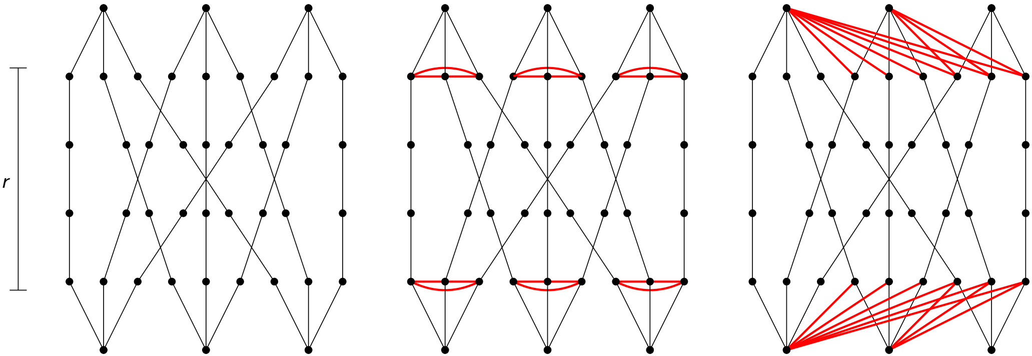


star r-crossing
$= r$-subdivided biclique



clique r-crossing

Characterizing Monadic Dependence by Forbidden Induced Subgraphs

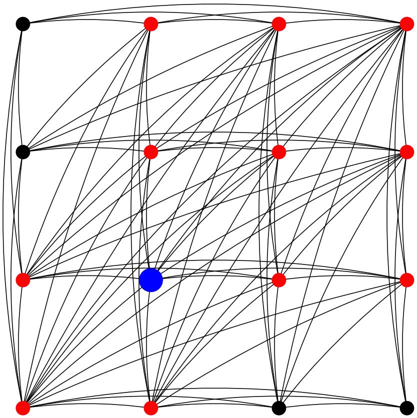


star r-crossing
$= r$-subdivided biclique

clique r-crossing

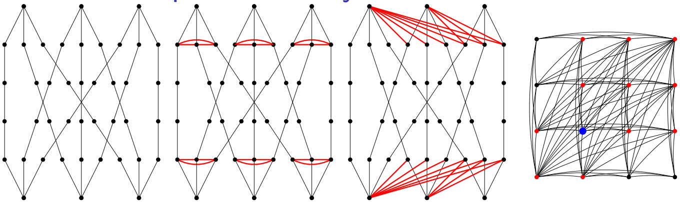
half-graph r-crossing

Characterizing Monadic Dependence by Forbidden Induced Subgraphs



comparability grid

Characterizing Monadic Dependence by Forbidden Induced Subgraphs

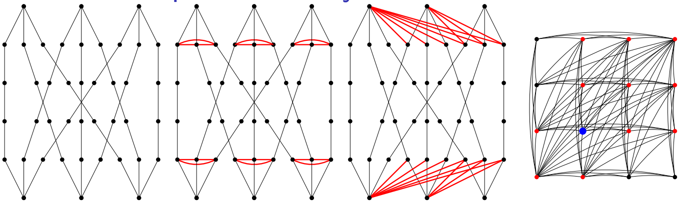


Theorem [Dreier, **NM**, Toruńczyk, 2024]

Let $\mathcal{C}$ be a graph class. Then $\mathcal{C}$ is monadically dependent if and only if for every $r \geq 1$ there exists $k \in \mathbb{N}$ such $\mathcal{C}$ excludes as induced subgraphs

- all layerwise flipped star r-crossings of order k, and
- all layerwise flipped clique r-crossings of order k, and
- all layerwise flipped half-graph r-crossings of order k, and
- the comparability grid of order k.

Characterizing Monadic Dependence by Forbidden Induced Subgraphs



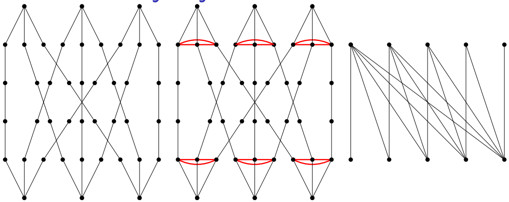
Theorem [Dreier, **NM**,Toruńczyk, 2024]

Let $\mathcal{C}$ be a graph class. Then $\mathcal{C}$ is monadically dependent if and only if for every $r \geq 1$ there exists $k \in \mathbb{N}$ such $\mathcal{C}$ excludes as induced subgraphs

- all layerwise flipped star r-crossings of order k, and
- all layerwise flipped clique r-crossings of order k, and
- all layerwise flipped half-graph r-crossings of order k, and
- the comparability grid of order k.

$\Rightarrow$ Model checking is hard on every hereditary, monadically independent graph class.

Characterizing Monadic Stability by Forbidden Induced Subgraphs

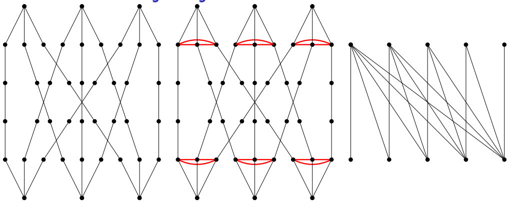


Theorem [Dreier, Eleftheriadis, **NM**, McCarty, Pilipczuk, Toruńczyk, 2024]

Let $\mathcal{C}$ be a graph class. Then $\mathcal{C}$ is monadically stable if and only if for every $r \geq 1$ there exists $k \in \mathbb{N}$ such $\mathcal{C}$ excludes as induced subgraphs

- all layerwise flipped star r-crossings of order k, and
- all layerwise flipped clique r-crossings of order k, and
- all semi-induced halfgraphs of order k

Characterizing Monadic Stability by Forbidden Induced Subgraphs



Theorem [Dreier, Eleftheriadis, **NM**, McCarty, Pilipczuk, Toruńczyk, 2024]

Let $\mathcal{C}$ be a graph class. Then $\mathcal{C}$ is monadically stable if and only if for every $r \geq 1$ there exists $k \in \mathbb{N}$ such $\mathcal{C}$ excludes as induced subgraphs

- all layerwise flipped star r-crossings of order k, and
- all layerwise flipped clique r-crossings of order k, and
- all semi-induced halfgraphs of order k

Characterizations: ramsey-theoretic ✓ forbidden induced subgraphs ✓

Next up: a game characterization for monadic stability

The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

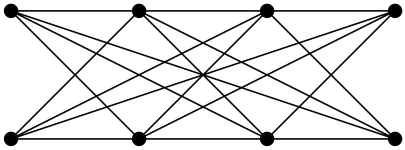
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



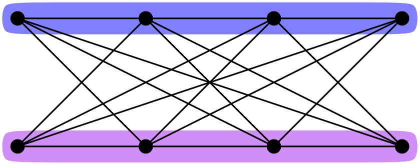
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



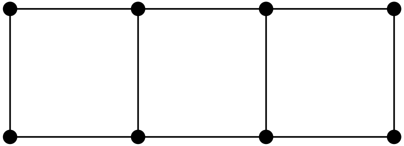
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



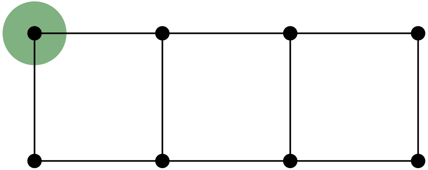
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



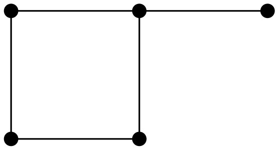
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



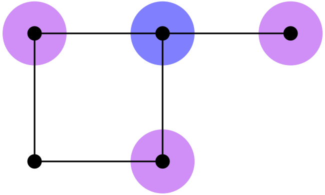
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



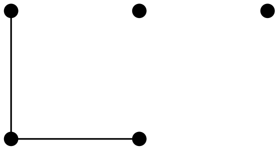
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



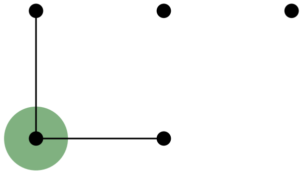
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



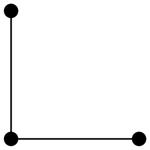
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



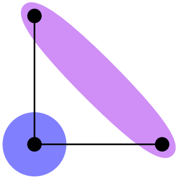
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



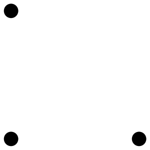
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



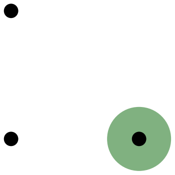
The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:



The Flipper Game

The radius-r Flipper game is played on a graph G_1. In round i

1. Flipper chooses a flip F
2. Localizer chooses G_{i+1} as a radius-r ball in $G_i \oplus F$.

Flipper wins once G_i has size 1.

Example play of the radius-2 Flipper game:

The Flipper Game in Monadically Stable Classes

Theorem [Gajarský, **NM**, McCarty, Ohlmann, Pilipczuk, Przybyszewski, Siebertz, Sokołowski, Toruńczyk]

A graph class $\mathcal{C}$ is monadically stable $\Leftrightarrow$

$\forall r \exists \ell$ such that Flipper wins the radius-r game on all graphs from $\mathcal{C}$ in ℓ rounds.

Proof builds on flip-flatness. Flippers moves are computable in time $\mathcal{O}_{\mathcal{C},r}(n^2)$.

The Flipper Game in Monadically Stable Classes

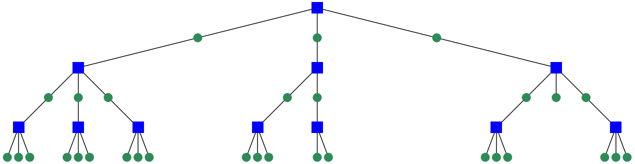
Theorem [Gajarský, **NM**, McCarty, Ohlmann, Pilipczuk, Przybyszewski, Siebertz, Sokołowski, Toruńczyk]

A graph class $\mathcal{C}$ is monadically stable $\Leftrightarrow$

$\forall r \exists \ell$ such that Flipper wins the radius-r game on all graphs from $\mathcal{C}$ in ℓ rounds.

Proof builds on flip-flatness. Flippers moves are computable in time $\mathcal{O}_{\mathcal{C},r}(n^2)$.

The game tree is a **bounded depth** decomposition of a graph into r-neighborhoods.



The Flipper Game in Monadically Stable Classes

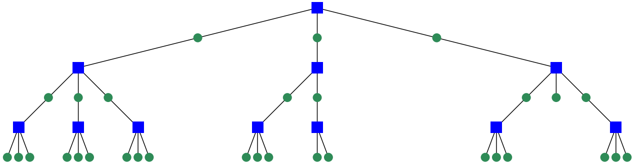
Theorem [Gajarský, **NM**, McCarty, Ohlmann, Pilipczuk, Przybyszewski, Siebertz, Sokołowski, Toruńczyk]

A graph class $\mathcal{C}$ is monadically stable $\Leftrightarrow$

$\forall r \exists \ell$ such that Flipper wins the radius-r game on all graphs from $\mathcal{C}$ in ℓ rounds.

Proof builds on flip-flatness. Flippers moves are computable in time $\mathcal{O}_{\mathcal{C},r}(n^2)$.

The game tree is a **bounded depth** decomposition of a graph into r-neighborhoods.

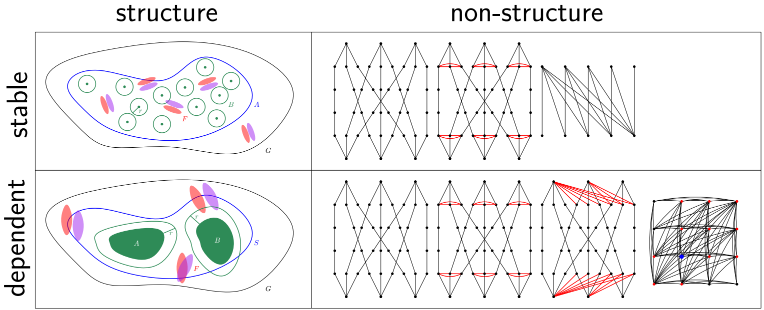


The decomposition can be further compressed by clustering neighborhoods.

Dynamic programming on the compressed tree gives **fpt model checking**.

Summary

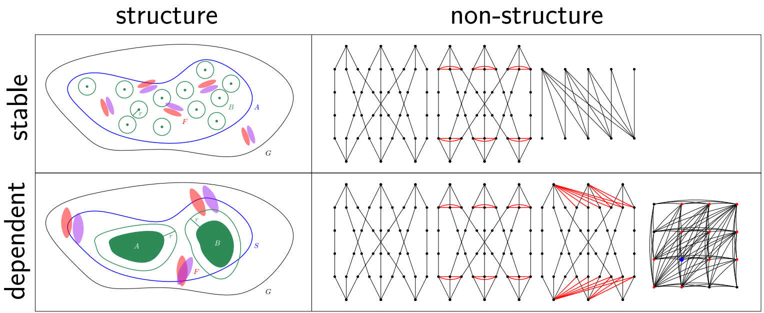
We have initiated the development of a combinatorial structure theory for monadically stable and dependent graph classes:



Algorithmic applications: model checking is fpt on every monadically stable class, but $\mathrm{AW}[*]$-hard on every hereditary, monadically independent class.

Summary

We have initiated the development of a combinatorial structure theory for monadically stable and dependent graph classes:



Algorithmic applications: model checking is fpt on every monadically stable class, but $\mathrm{AW}[*]$-hard on every hereditary, monadically independent class.

Thanks! Vielen Dank! Dziękuję bardzo! Merci!

Backup slides

Publications 1/2

1. *Indiscernibles and Flatness in Monadically Stable and Monadically NIP Classes*
   joint work with Jan Dreier, Sebastian Siebertz, Szymon Toruńczyk
   presented at ICALP 2023
2. *Flipper Games for Monadically Stable Graph Classes*
   joint work with Jakub Gajarský, Rose McCarty, Pierre Ohlmann, Michał Pilipczuk, Wojciech Przybyszewski, Sebastian Siebertz, Marek Sokołowski, Szymon Toruńczyk
   presented at ICALP 2023
3. *First-Order Model Checking on Structurally Sparse Graph Classes*
   joint work with Jan Dreier, Sebastian Siebertz
   presented at STOC 2023

Publications 2/2

4. *First-Order Model Checking on Monadically Stable Graph Classes*
   joint work with Jan Dreier, Ioannis Eleftheriadis, Rose McCarty, Michał Pilipczuk, Szymon Toruńczyk
   accepted at FOCS 2024

5. *Flip-Breakability: A Combinatorial Dichotomy for Monadically Dependent Graph Classes*
   joint work with Jan Dreier, Szymon Toruńczyk
   presented at STOC 2024

Flip-Flatness

Theorem

A graph class $\mathcal{C}$ is *flip-flat* if for every radius $r \in \mathbb{N}$ there exists a function $N_r : \mathbb{N} \to \mathbb{N}$ and a constant $k_r \in \mathbb{N}$ such that for all $m \in \mathbb{N}$, $G \in \mathcal{C}$ and $W \subseteq V(G)$ with $|W| \geq N_r(m)$ there exist a subset $A \subset W$ with $|A| \geq m$ and a k_r-flip H of G such that for every two distinct vertices $u, v \in A$:

$$\mathrm{dist}_H(u, v) > r.$$

Flip-Breakability

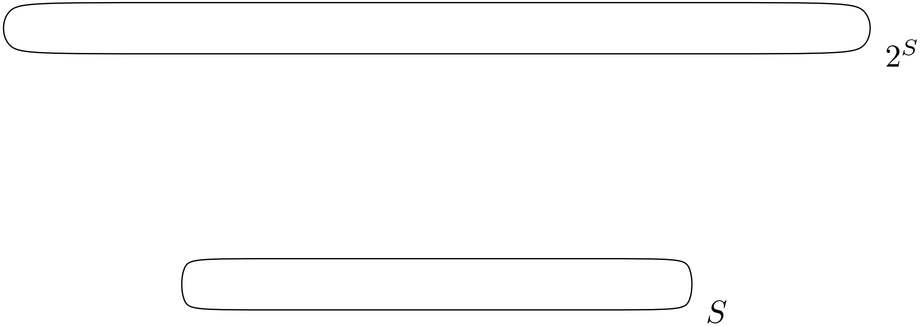
Theorem

A graph class $\mathcal{C}$ is *flip-breakable* if for every radius $r \in \mathbb{N}$ there exists a function $N_r : \mathbb{N} \to \mathbb{N}$ and a constant $k_r \in \mathbb{N}$ such that for all $m \in \mathbb{N}$, $G \in \mathcal{C}$ and $W \subseteq V(G)$ with $|W| \geq N_r(m)$ there exist subsets $A, B \subset W$ with $|A|, |B| \geq m$ and a k_r-flip H of G such that:

$$\mathrm{dist}_H(A, B) > r.$$

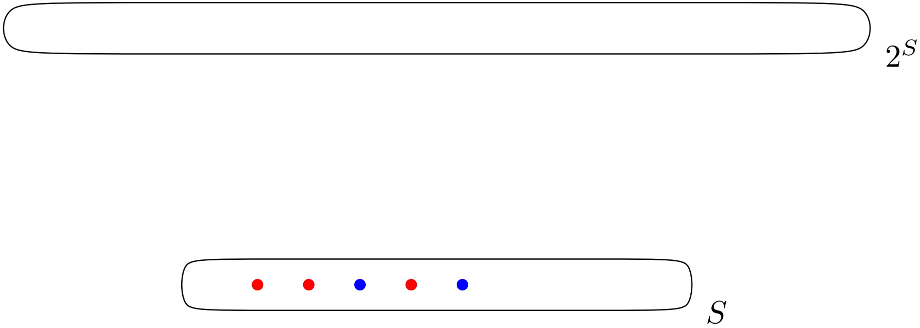
Flip-Breakability ⇒ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



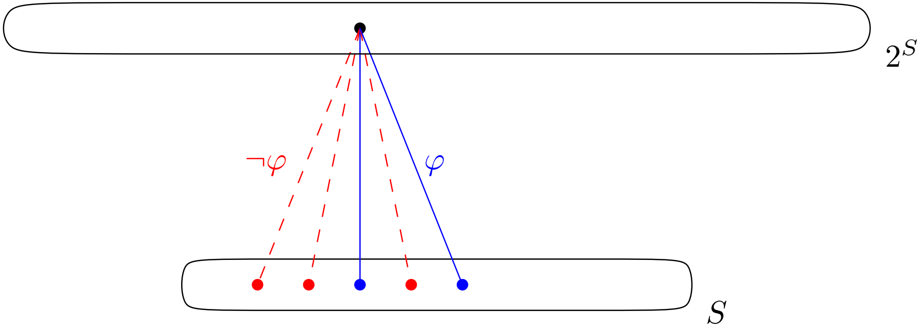
Flip-Breakability ⇒ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



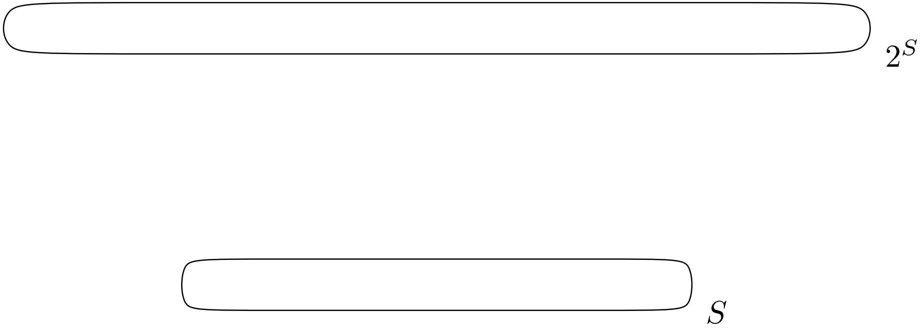
Flip-Breakability ⇒ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



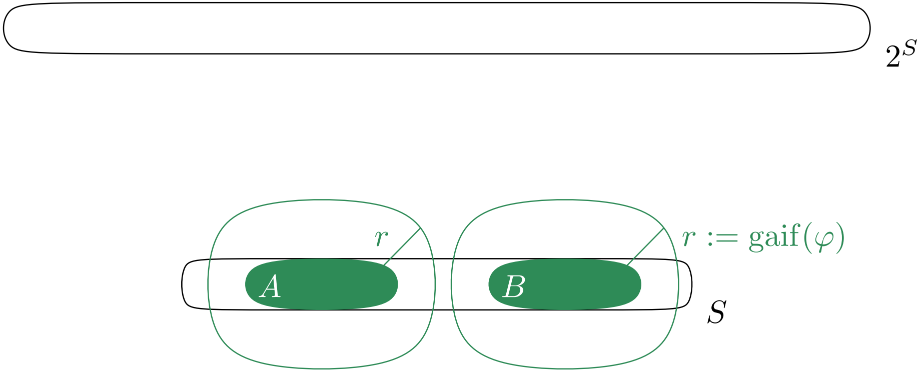
Flip-Breakability $\Rightarrow$ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



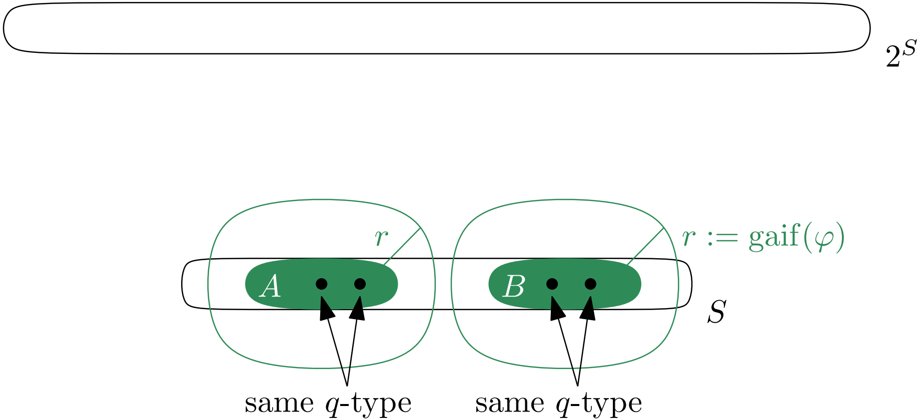
Flip-Breakability $\Rightarrow$ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



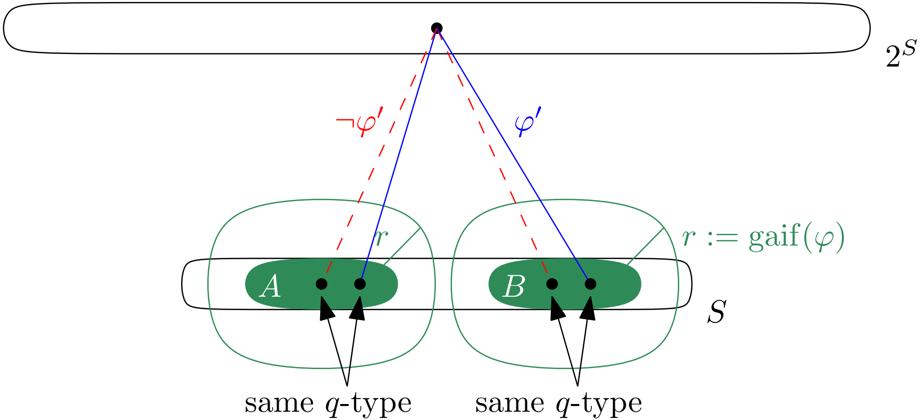
Flip-Breakability ⇒ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



Flip-Breakability ⇒ Monadic Dependence

Assume towards a contradiction a class $\mathcal{C}$ is not monadically dependent but flip-breakable.



Stability and Dependence in Model Theory

On a class $\mathcal{C}$, a formula $\varphi(\bar{x}, \bar{y})$ has

- the *order property* if for every $k \in \mathbb{N}$ there are $G \in \mathcal{C}$ and two sequences $(\bar{a}_i)_{i\in[k]}$, $(\bar{b}_j)_{j\in[k]}$ of tuples in G, such that for all $i, j \in [k]$: $G \models \varphi(\bar{a}_i, \bar{b}_j) \Leftrightarrow i \leq j$.
- the *independence property* if for every $k \in \mathbb{N}$ there are $G \in \mathcal{C}$, a size k set $A \subseteq V(G)^{|\bar{x}|}$ and a sequence $(\bar{b}_J)_{J\subseteq A}$ of tuples in G such that for all $\bar{a} \in A, J \subseteq A$

$$G \models \varphi(\bar{a}, \bar{b}_J) \quad \Leftrightarrow \quad \bar{a} \in J.$$

A graph class is *stable* if it does not have the order property.
It is *monadically stable* if the class of colored graphs from $\mathcal{C}$ is stable.

A graph class is *dependent* if it does not have the independence property.
It is *monadically dependent* if the class of colored graphs from $\mathcal{C}$ is dependent.

Approximation Algorithms

Distance-r dominating set:

- constant factor approximation in bounded expansion classes [Dvořák 2013]
- $O(d \cdot \log(d \cdot OPT))$ approximation of the distance-1 case on graphs with VC dimension $\leq d$ [Brönnimann, Goodrich, 1995]

Distance-r independent set:

- constant factor approximation in bounded expansion classes [Dvořák 2013]
- n^{ε} approximation in nowhere dense classes [Dvořák 2019]
- n^{ε} approximation in bounded twin-width classes [Bergé et al. 2022]