A combinatorial game for monadic stability

Rose McCarty

UNIVERSITY OF WARSAW

PRINCETON UNIVERSITY

August 21, 2022

With Gajarský, Mählmann, Ohlmann, Ossona de Mendez, Pilipczuk, Przybyszewski, Siebertz, Sokołowski, and Toruńczyk.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.

Theorem

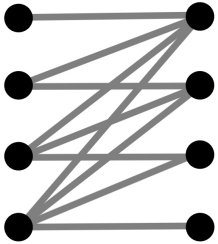
A class of graphs is **monadically stable** *if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.*

Comes from model theory; Baldwin and Shelah.

Theorem

A class of graphs is **monadically stable** *if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.*

Comes from model theory; Baldwin and Shelah. Idea is to use first-order logic to "exclude":



half-graph

Theorem

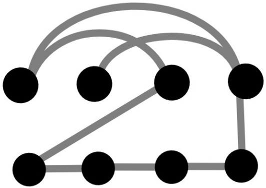
A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector.

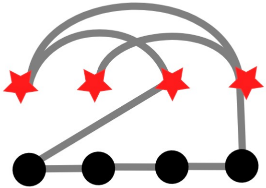


To **flip,** select $X \subseteq V(G)$ and replace $G[X]$ by its complement.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector.

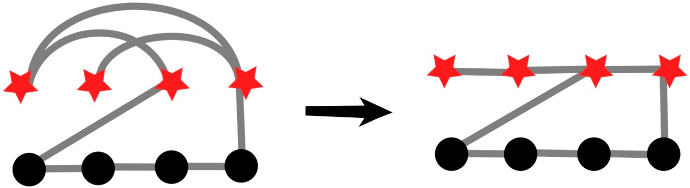


To **flip,** select $X \subseteq V(G)$ and replace $G[X]$ by its complement.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector.



To **flip,** select $X \subseteq V(G)$ and replace $G[X]$ by its complement.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

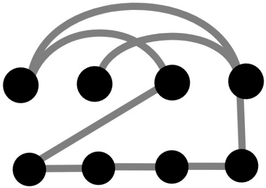
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



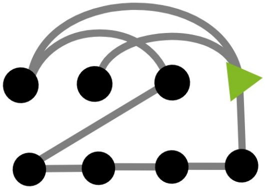
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



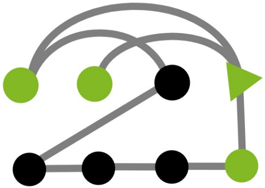
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



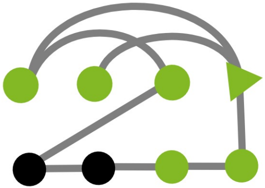
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:

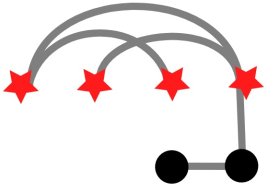
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



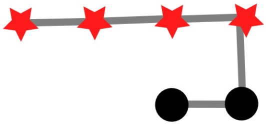
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



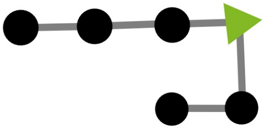
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



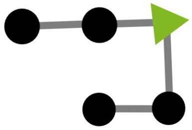
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



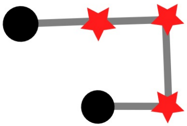
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



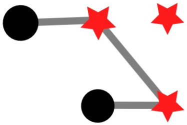
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



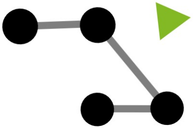
Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:



Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Example with $r = 2$:

Theorem

A class of graphs is monadically stable if and only if Flipper wins the **radius-r flipper game** *for each $r \in \mathbb{N}$.*

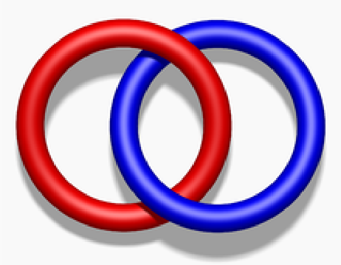
Related to Dreier, Mählmann, Siebertz, and Toruńczyk. Two player game: Flipper and Connector. In each round:

1) If $|V(G)| = 1$ then Flipper wins in that round.
2) Else, Connector picks a vertex v and we restrict to $B_r(v)$.
3) Then Flipper performs a flip.

Flipper **wins the game** on a class $\mathcal{C}$ if there exists $t \in \mathbb{N}$ so that Flipper wins in $\leq t$ rounds on each $G \in \mathcal{C}$.

Theorem (Robertson, Seymour, Thomas)

A class of graphs is **linklessly embeddable** *if and only if it contains no minor in the Petersen family.*

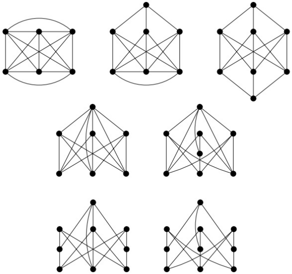


2. R. McCarty and R. Thomas. *The extremal function for bipartite linklessly embeddable graphs*. Combinatorica, Volume 39, pp. 1081–1104, 2019.

1. R. McCarty. *The extremal function and Colin de Verdière graph parameter.* Electron. J. Combin., 25(2):P2.32, 2018.

Theorem (Robertson, Seymour, Thomas)

A class of graphs is linklessly embeddable if and only if it contains no minor in the **Petersen family**.



2. R. McCarty and R. Thomas. *The extremal function for bipartite linklessly embeddable graphs*. Combinatorica, Volume 39, pp. 1081–1104, 2019.

1. R. McCarty. *The extremal function and Colin de Verdière graph parameter.* Electron. J. Combin., 25(2):P2.32, 2018.

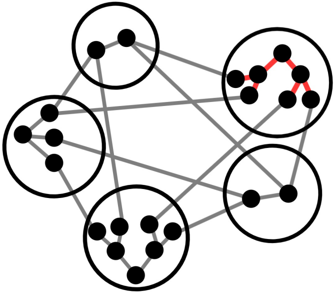
Theorem (Dvořák, Král', Thomas)

First-order model-checking is fixed-parameter tractable on any class of bounded expansion.

Theorem (Dvořák, Král', Thomas)

First-order model-checking is fixed-parameter tractable on any class of **bounded expansion**.

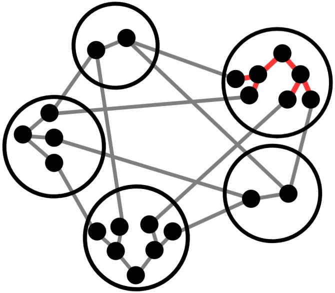
For each $r \in \mathbb{N}$, there exists $t \in \mathbb{N}$ so that no graph with average degree $> t$ is an r-shallow minor of any graph in $\mathcal{C}$.



Theorem (Grohe, Kreutzer, Siebertz)

First-order model-checking is fixed-parameter tractable on any class which is **nowhere dense**.

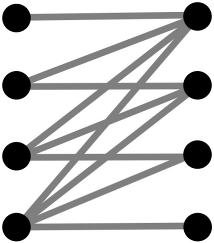
For each $r \in \mathbb{N}$, there exists $t \in \mathbb{N}$ so that K_t is not an r-shallow minor of any graph in $\mathcal{C}$.



Conjecture (folklore; see Gajarský, Pilipczuk, Toruńczyk)

First-order model-checking is fixed-parameter tractable on any class which is **monadically stable***.*

Recall that we use first-order logic to “exclude”:



half-graph

Theorem (Comes from Dvořák 2018)

A class is nowhere dense if and only if it is monadically stable and **forbids a $K_{t,t}$-subgraph** *for some $t \in \mathbb{N}$.*

Theorem (Comes from Dvořák 2018)

A class is nowhere dense if and only if it is monadically stable and **forbids a $K_{t,t}$-subgraph** *for some $t \in \mathbb{N}$.*

So our main theorem

Theorem

A class of graphs is **monadically stable** *if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.*

Theorem (Comes from Dvořák 2018)

A class is nowhere dense if and only if it is monadically stable and **forbids a $K_{t,t}$-subgraph** *for some $t \in \mathbb{N}$.*

So our main theorem

Theorem

A class of graphs is **monadically stable** *if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.*

is like

Theorem (Grohe, Kreutzer, Siebertz)

A class of graphs is **nowhere dense** *if and only if Splitter wins the radius-r splitter game for each $r \in \mathbb{N}$.*

Theorem (Comes from Dvořák 2018)

A class is nowhere dense if and only if it is monadically stable and **forbids a $K_{t,t}$-subgraph** *for some $t \in \mathbb{N}$.*

So our main theorem

Theorem

A class of graphs is **monadically stable** *if and only if Flipper wins the radius-r flipper game for each $r \in \mathbb{N}$.*

is like

Theorem (Grohe, Kreutzer, Siebertz)

A class of graphs is **nowhere dense** *if and only if Splitter wins the radius-r splitter game for each $r \in \mathbb{N}$.*

but for classes that are allowed to be “dense”.

Formulas in first-order logic are obtained as follows.

Formulas in first-order logic are obtained as follows.

1) $x \mathrm{adj} y$ and $x = y$ are formulas.

Formulas in first-order logic are obtained as follows.

1) $x \mathrm{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, $\ldots$.

Formulas in first-order logic are obtained as follows.

1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

Formulas in first-order logic are obtained as follows.

1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

$$\phi := \exists x \forall y (y = x) \vee (y \operatorname{adj} x)$$

says that there is a dominating vertex.

Formulas in first-order logic are obtained as follows.

1) $x \text{ adj } y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

$$\phi(x) := \forall y (y = x) \vee (y \text{ adj } x)$$

Formulas in first-order logic are obtained as follows.

1) $x \mathrm{adj}\, y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, $\ldots$.
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

$$\phi(x) := \forall y (y = x) \vee (y \mathrm{adj}\, x)$$

For $v \in V(G)$, $\phi(v)$ says that v is a dominating vertex.

Formulas in first-order logic are obtained as follows.

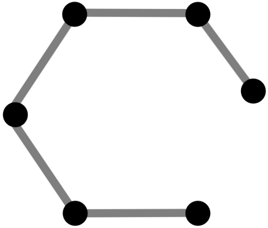
1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, $\ldots$.
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

A **sentence** is a formula ϕ with no free variables.

Formulas in first-order logic are obtained as follows.

1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, $\ldots$.
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

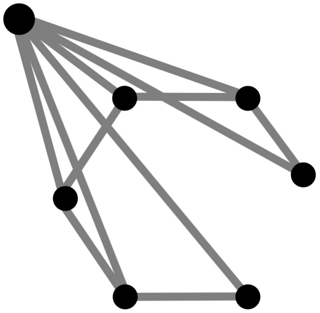
A **sentence** is a formula ϕ with no free variables.



Formulas in first-order logic are obtained as follows.

1) $x \text{ adj } y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

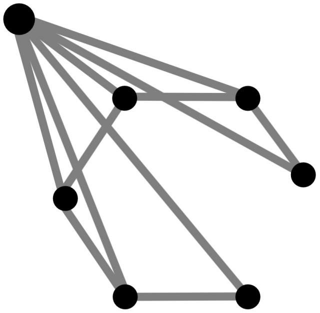
A **sentence** is a formula ϕ with no free variables.



Formulas in first-order logic are obtained as follows.

1) x adj y and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x\phi$ and $\forall x\phi$.

A **sentence** is a formula ϕ with no free variables. If ϕ is true for G, we say G **models** ϕ or write $G \models \phi$.



Formulas in first-order logic are obtained as follows.

1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

A **sentence** is a formula ϕ with no free variables. If ϕ is true for G, we say G **models** ϕ or write $G \models \phi$.

Naive algorithm for k-vertex-dominating set: $\mathcal{O}(n^k)$.

Formulas in first-order logic are obtained as follows.

1) $x \text{ adj } y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

A **sentence** is a formula ϕ with no free variables. If ϕ is true for G, we say G **models** ϕ or write $G \models \phi$.

Naive algorithm for k-vertex-dominating set: $\mathcal{O}(n^k)$.

Naive algorithm for determining if $G \models \phi$: $\mathcal{O}(n^{|\phi|})$.

Formulas in first-order logic are obtained as follows.

1) $x \operatorname{adj} y$ and $x = y$ are formulas.
2) If ϕ and ψ are formulas, then so are $\neg\phi$, $\phi \wedge \psi$, ...
3) If ϕ is a formula, then so is $\exists x \phi$ and $\forall x \phi$.

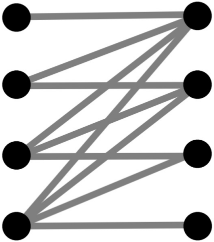
A **sentence** is a formula ϕ with no free variables. If ϕ is true for G, we say G **models** ϕ or write $G \models \phi$.

Naive algorithm for k-vertex-dominating set: $\mathcal{O}(n^k)$.

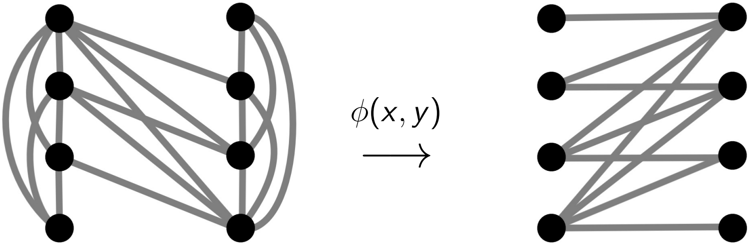
Naive algorithm for determining if $G \models \phi$: $\mathcal{O}(n^{|\phi|})$.

FO model-checking is FPT on $\mathcal{C}$ if there exists $f : \mathbb{N} \to \mathbb{N}$ and $c \in \mathbb{R}$ so that the problem can be solved in time $f(|\phi|)n^c$.

Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic.



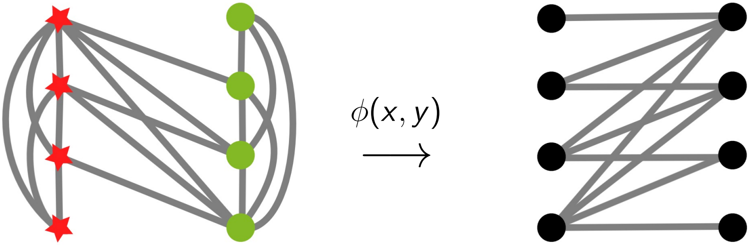
Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic. To exclude something from $\mathcal{C}$:



$G \in \mathcal{C}$

$$\phi(x, y) := \neg x \text{ adj } y$$

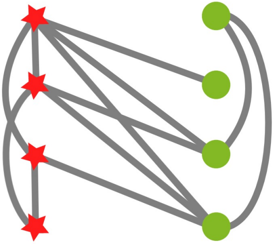
Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic. To exclude something from $\mathcal{C}$:



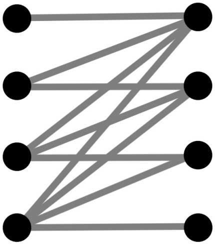
$G \in \mathcal{C}$

$$\phi(x, y) := (\neg x \text{ adj } y) \wedge (c(x) \neq c(y))$$

Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic. To exclude something from $\mathcal{C}$:



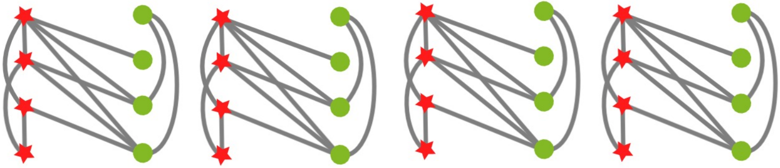
$$\xrightarrow{\phi(x, y)}$$



$G \in \mathcal{C}$

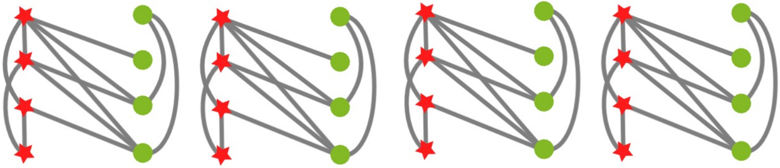
$$\phi(x, y) := (\neg x \text{ adj } y) \wedge (c(x) \neq c(y))$$

Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic. To exclude something from $\mathcal{C}$:



$$\phi(x, y) := (\neg x \text{ adj } y) \wedge (c(x) \neq c(y)) \wedge (\neg x \text{ copy } y)$$

Recall that a class is **monadically stable** if half-graphs are "excluded via" first-order logic. To exclude something from $\mathcal{C}$:



For fixed $\phi(x, y)$, the resulting **transduction** of $\mathcal{C}$ is the class of all induced subgraphs of graphs which can be obtained this way.

$$\phi(x, y) := (\neg x \text{ adj } y) \wedge (c(x) \neq c(y)) \wedge (\neg x \text{ copy } y)$$

forbidden minor

$\subseteq$

bounded expansion

(shallow minors have bounded average degree)

$\subseteq$

nowhere dense

(have forbidden shallow minors)

$\subseteq$

monadically stable

(exclude half-graph using first-order logic)

$\subseteq$

monadically dependent

(exclude *anything* using first-order logic)

forbidden minor

$\subseteq$

bounded expansion

(shallow minors have bounded average degree)

$\subseteq$

nowhere dense

(have forbidden shallow minors)

$\subseteq$

monadically stable

(exclude half-graph using first-order logic)

$\subseteq$

monadically dependent

(exclude *anything* using first-order logic)

forbidden minor

$\subseteq$

bounded expansion

(shallow minors have bounded average degree)

$\subseteq$

nowhere dense

(have forbidden shallow minors)

$\subseteq$

monadically stable

(exclude half-graph using first-order logic)

$\subseteq$

monadically dependent

(exclude *anything* using first-order logic)

forbidden minor

$\subseteq$

bounded expansion

(shallow minors have bounded average degree)

$\subseteq$

nowhere dense

(have forbidden shallow minors)

$\subseteq$

monadically stable

(exclude half-graph using first-order logic)

$\subseteq$

monadically dependent

(exclude *anything* using first-order logic)

forbidden minor

$\subseteq$

bounded expansion

(shallow minors have bounded average degree)

$\subseteq$

nowhere dense

(have forbidden shallow minors)

$\subseteq$

monadically stable

(exclude half-graph using first-order logic)

$\subseteq$

monadically dependent

(exclude *anything* using first-order logic)

Conjecture (folklore; see Gajarský, Pilipczuk, Toruńczyk)

A class $\mathcal{C}$ which is closed under vertex-deletion is **monadically dependent** *if and only if FO model-checking is FPT on $\mathcal{C}$.*

Conjecture (folklore; see Gajarský, Pilipczuk, Toruńczyk)

A class $\mathcal{C}$ which is closed under vertex-deletion is **monadically dependent** *if and only if FO model-checking is FPT on $\mathcal{C}$.*

Conjecture (Gajarský, Pilipczuk, Toruńczyk)

A class has unbounded clique-width if and only if it has an FO-transduction which contains a subdivision of each wall.

Conjecture (folklore; see Gajarský, Pilipczuk, Toruńczyk)

A class $\mathcal{C}$ which is closed under vertex-deletion is **monadically dependent** *if and only if FO model-checking is FPT on $\mathcal{C}$.*

Conjecture (Gajarský, Pilipczuk, Toruńczyk)

A class has unbounded clique-width if and only if it has an FO-transduction which contains a subdivision of each wall.

Thank you!