On the number of types in sparse graphs

Michał Pilipczuk

based on joint work with Sebastian Siebertz and Szymon Toruńczyk

Institute of Informatics, University of Warsaw, Poland

ANR DeLTA meeting

Bordeaux, December 5^{th}, 2018

Part 1: Sparsity

Sparsity

- **Q**: What does it mean that a graph is **sparse**?

Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...

Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...

Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...
- **Attempt 1.** Edge density bounded by a constant:

$$\mathrm{density}(G) := \frac{|E(G)|}{|V(G)|} \leqslant c.$$

Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...
- **Attempt 1.** Edge density bounded by a constant:

$$\mathrm{density}(G) := \frac{|E(G)|}{|V(G)|} \leqslant c.$$

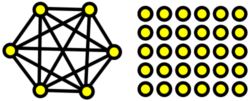
- **Note**: density(G) is half of the average degree.

Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...
- **Attempt 1.** Edge density bounded by a constant:

$$\text{density}(G) := \frac{|E(G)|}{|V(G)|} \leqslant c.$$

- **Note**: density(G) is half of the average degree.
- **Problem**: Take a clique of size n plus $n^2 - n$ isolated vertices.

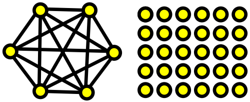


Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...
- **Attempt 1.** Edge density bounded by a constant:

$$\mathrm{density}(G) := \frac{|E(G)|}{|V(G)|} \leqslant c.$$

- **Note**: density(G) is half of the average degree.
- **Problem**: Take a clique of size n plus $n^2 - n$ isolated vertices.
  - This has density $< \frac{1}{2}$.

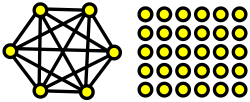


Sparsity

- **Q**: What does it mean that a graph is **sparse**?
  - **Positive ex**: Bounded max degree, planar, bounded treewidth, ...
  - **Negative ex**: Cliques, bicliques, ...
- **Attempt 1.** Edge density bounded by a constant:

$$\mathrm{density}(G) := \frac{|E(G)|}{|V(G)|} \leqslant c.$$

- **Note**: density(G) is half of the average degree.
- **Problem**: Take a clique of size n plus $n^2 - n$ isolated vertices.
  - This has density $< \frac{1}{2}$.
  - **Issue**: Although the density is small, contains a dense substructure.



Sparsity

- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

Sparsity

- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.

Sparsity

- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

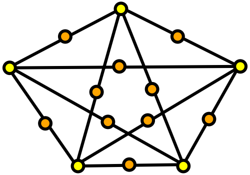
- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.
  - A graph G is d-degenerate if $V(G)$ can be ordered so that every vertex has $\leqslant d$ smaller neighbors.

Sparsity

- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

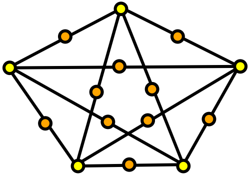
- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.
  - A graph G is d-degenerate if $V(G)$ can be ordered so that every vertex has $\leqslant d$ smaller neighbors.
- **Problem**: Take a clique K_n with each edge subdivided once.



- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

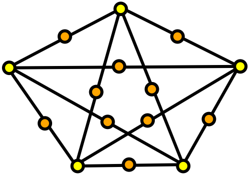
- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.
  - A graph G is d-degenerate if $V(G)$ can be ordered so that every vertex has $\leqslant d$ smaller neighbors.
- **Problem**: Take a clique K_n with each edge subdivided once.
  - In every subgraph of this graph, the number of edges is at most twice the number of vertices.



- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.
  - A graph G is d-degenerate if $V(G)$ can be ordered so that every vertex has $\leqslant d$ smaller neighbors.
- **Problem**: Take a clique K_n with each edge subdivided once.
  - In every subgraph of this graph, the number of edges is at most twice the number of vertices.
  - **Issue**: We see a dense structure "at depth" 1.

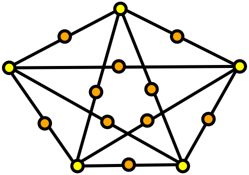


Sparsity

- **Attempt 2.** Every subgraph of G has bounded edge density:

$$\max_{H \subseteq G} \{\text{density}(H)\} \leqslant c.$$

- **Remark**: Essentially equivalent to **degeneracy** or **arboricity**.
  - A graph G is d-degenerate if $V(G)$ can be ordered so that every vertex has $\leqslant d$ smaller neighbors.
- **Problem**: Take a clique K_n with each edge subdivided once.
  - In every subgraph of this graph, the number of edges is at most twice the number of vertices.
  - **Issue**: We see a dense structure "at depth" 1.
  - If we are looking for a structurally robust notion of sparsity, morally this example should be dense.

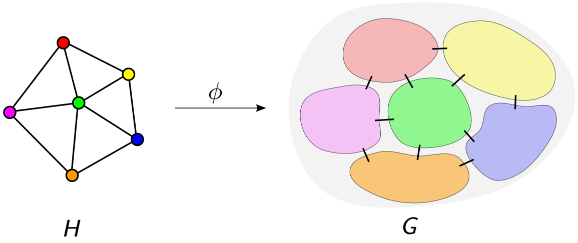


Shallow minors

- **Need**: notion of embedding that looks at constant "depth".

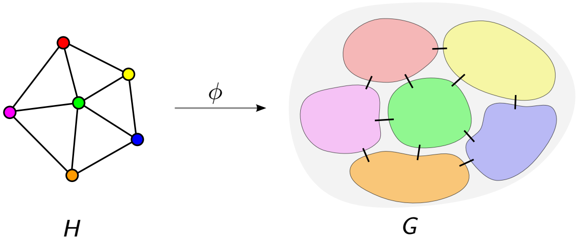
Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.



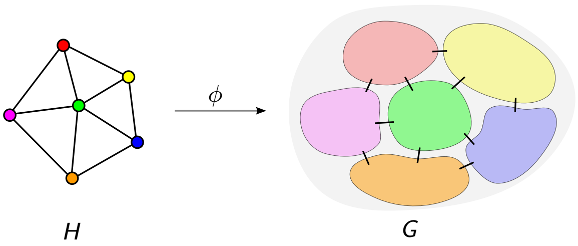
Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.
  - Model ϕ maps vertices $u \in V(H)$ to pairwise disjoint connected subgraphs $\phi(u)$ of G, called **branch sets**.



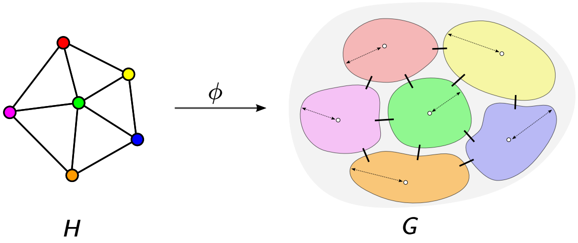
Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.
  - Model ϕ maps vertices $u \in V(H)$ to pairwise disjoint connected subgraphs $\phi(u)$ of G, called **branch sets**.
  - If $uv \in E(H)$, then there should be an edge between $\phi(u)$ and $\phi(v)$.



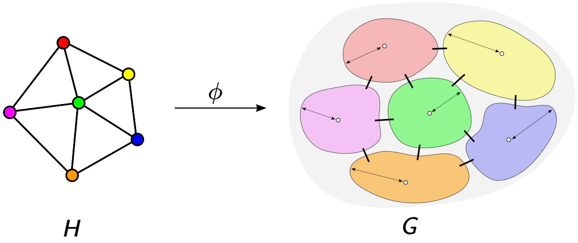
Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.
  - Model ϕ maps vertices $u \in V(H)$ to pairwise disjoint connected subgraphs $\phi(u)$ of G, called **branch sets**.
  - If $uv \in E(H)$, then there should be an edge between $\phi(u)$ and $\phi(v)$.
- Graph H is a **depth-d minor** of G if H has a minor model in G where each branch set has radius at most d.



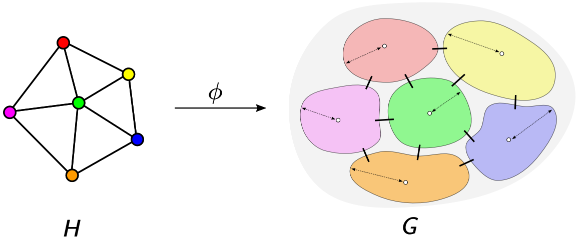
Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.
  - Model ϕ maps vertices $u \in V(H)$ to pairwise disjoint connected subgraphs $\phi(u)$ of G, called **branch sets**.
  - If $uv \in E(H)$, then there should be an edge between $\phi(u)$ and $\phi(v)$.
- Graph H is a **depth-d minor** of G if H has a minor model in G where each branch set has radius at most d.
  - **Note**: Depth-0 minors are exactly subgraphs.



Shallow minors

- **Need**: notion of embedding that looks at constant "depth".
- Graph H is a **minor** of G if there is a **minor model** ϕ of H in G.
  - Model ϕ maps vertices $u \in V(H)$ to pairwise disjoint connected subgraphs $\phi(u)$ of G, called **branch sets**.
  - If $uv \in E(H)$, then there should be an edge between $\phi(u)$ and $\phi(v)$.
- Graph H is a **depth-d minor** of G if H has a minor model in G where each branch set has radius at most d.
  - **Note**: Depth-0 minors are exactly subgraphs.
- **Idea**: Replace **subgraphs** with **shallow minors** in the definition.



Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Bounded expansion

A class of graphs $\mathcal{C}$ has **bounded expansion** if for every $d \in \mathbb{N}$ there exists $c \in \mathbb{N}$ such that $\text{density}(H) \leqslant c$ for all $H \in \mathcal{C}\nabla d$.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Bounded expansion

A class of graphs $\mathcal{C}$ has **bounded expansion** if for every $d \in \mathbb{N}$ there exists $c \in \mathbb{N}$ such that density$(H) \leqslant c$ for all $H \in \mathcal{C}\nabla d$.

Nowhere dense

A class of graphs $\mathcal{C}$ is **nowhere dense** if for every $d \in \mathbb{N}$ there exists $t \in \mathbb{N}$ such that $K_t \notin \mathcal{C}\nabla d$.

Equivalently, $\mathcal{C}\nabla d \neq \text{Graphs}$ for all $d \in \mathbb{N}$.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Bounded expansion

A class of graphs $\mathcal{C}$ has **bounded expansion** if for every $d \in \mathbb{N}$ there exists $c \in \mathbb{N}$ such that density$(H) \leqslant c$ for all $H \in \mathcal{C}\nabla d$.

Nowhere dense

A class of graphs $\mathcal{C}$ is **nowhere dense** if for every $d \in \mathbb{N}$ there exists $t \in \mathbb{N}$ such that $K_t \notin \mathcal{C}\nabla d$.

Equivalently, $\mathcal{C}\nabla d \neq \mathrm{Graphs}$ for all $d \in \mathbb{N}$.

- **Intuition**: At every constant depth we see a sparse class, but the parameters can deteriorate with increasing depth.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Bounded expansion

A class of graphs $\mathcal{C}$ has **bounded expansion** if for every $d \in \mathbb{N}$ there exists $c \in \mathbb{N}$ such that density$(H) \leqslant c$ for all $H \in \mathcal{C}\nabla d$.

Nowhere dense

A class of graphs $\mathcal{C}$ is **nowhere dense** if for every $d \in \mathbb{N}$ there exists $t \in \mathbb{N}$ such that $K_t \notin \mathcal{C}\nabla d$.

Equivalently, $\mathcal{C}\nabla d \neq \mathrm{Graphs}$ for all $d \in \mathbb{N}$.

- **Intuition**: At every constant depth we see a sparse class, but the parameters can deteriorate with increasing depth.
- **Note**: Nowhere dense classes are also sparse.

Sparsity, formally

- **Note**: Sparsity is a property of a graph class, not of a single graph.
- **Notation**: $\mathcal{C}\nabla d = \{\text{depth-}d \text{ minors of graphs from } \mathcal{C}\}$.
  - **Ex**: $\mathcal{C}\nabla 0$ is the closure of $\mathcal{C}$ under subgraphs.

Bounded expansion

A class of graphs $\mathcal{C}$ has **bounded expansion** if for every $d \in \mathbb{N}$ there exists $c \in \mathbb{N}$ such that density$(H) \leqslant c$ for all $H \in \mathcal{C}\nabla d$.

Nowhere dense

A class of graphs $\mathcal{C}$ is **nowhere dense** if for every $d \in \mathbb{N}$ there exists $t \in \mathbb{N}$ such that $K_t \notin \mathcal{C}\nabla d$.

Equivalently, $\mathcal{C}\nabla d \neq \text{Graphs}$ for all $d \in \mathbb{N}$.

- **Intuition**: At every constant depth we see a sparse class, but the parameters can deteriorate with increasing depth.
- **Note**: Nowhere dense classes are also sparse.
  - If $H \in \mathcal{C}\nabla d$, then H has $\mathcal{O}_{\varepsilon,d}(|V(H)|^{1+\varepsilon})$ edges, for any $\varepsilon > 0$.

Hierarchy of sparsity

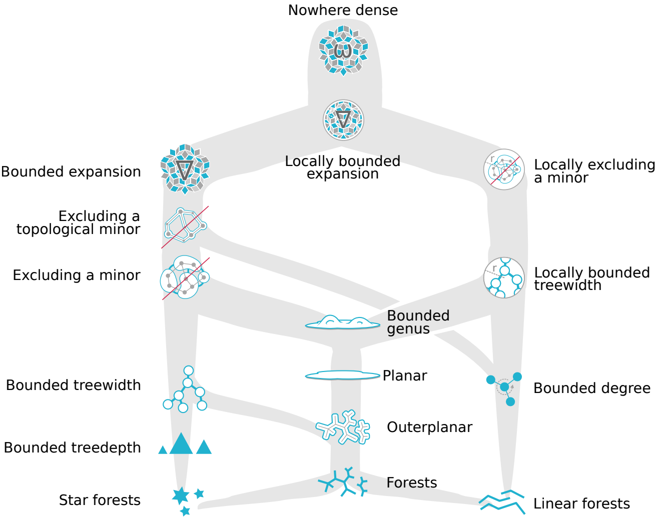


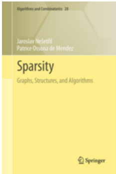
Figure by Felix Reidl

Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.

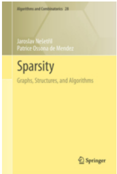
Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.



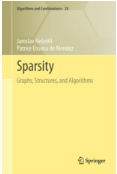
Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes: https://www.mimuw.edu.pl/~mp248287/sparsity/

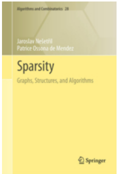


Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes:
    `https://www.mimuw.edu.pl/~mp248287/sparsity/`
- Many concepts appeared already much earlier.

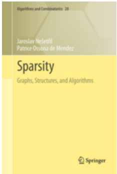


- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes: `https://www.mimuw.edu.pl/~mp248287/sparsity/`
- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness: Podewski and Ziegler in 1976.



Theory of sparsity

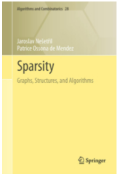
- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes: `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness: Podewski and Ziegler in 1976.
- **In summary**:

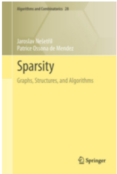
Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes:
    `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness:
    Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.

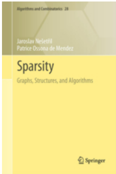
- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes:
    `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness:
    Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.
    - **This talk**: *neighborhood complexity*

Theory of sparsity

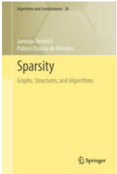
- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes:
    `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness:
    Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.
    - **This talk**: *neighborhood complexity*
    - **Friday**: *uniform quasi-wideness*

Theory of sparsity

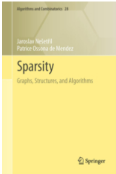
- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes:
    `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness:
    Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.
    - **This talk**: *neighborhood complexity*
    - **Friday**: *uniform quasi-wideness*
  - Each characterization yields a different viewpoint and a tool.

Theory of sparsity

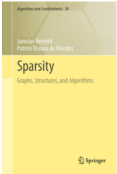
- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes: `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness: Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.
    - **This talk**: *neighborhood complexity*
    - **Friday**: *uniform quasi-wideness*
  - Each characterization yields a different viewpoint and a tool.
  - Applications in **combinatorics**, **algorithms**, and **logic**.

Theory of sparsity

- Developed by Nešetřil and Ossona de Mendez since 2005.
  - Monograph *Sparsity* presents the field as of 2012.
  - More recent lecture notes: `https://www.mimuw.edu.pl/~mp248287/sparsity/`



- Many concepts appeared already much earlier.
  - Earliest definition of nowhere denseness: Podewski and Ziegler in 1976.
- **In summary**:
  - Bounded expansion and nowhere denseness are fundamental concepts that have multiple equivalent characterizations.
    - **This talk**: *neighborhood complexity*
    - **Friday**: *uniform quasi-wideness*
  - Each characterization yields a different viewpoint and a tool.
  - Applications in **combinatorics**, **algorithms**, and **logic**.
  - Nowhere denseness delimits tractability for many basic problems.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

- **Trivial**: an $\mathcal{O}(n^{\|\varphi\|})$-time algorithm.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

- **Trivial**: an $\mathcal{O}(n^{\|\varphi\|})$-time algorithm.
- **Goal**: runtime $f(\|\varphi\|) \cdot n^c$ for a constant c and some function f.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

- **Trivial**: an $\mathcal{O}(n^{\|\varphi\|})$-time algorithm.
- **Goal**: runtime $f(\|\varphi\|) \cdot n^c$ for a constant c and some function f.
  - Called **fixed-parameter tractable**, or **FPT**, parameterized by $\|\varphi\|$.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

- **Trivial**: an $\mathcal{O}(n^{\|\varphi\|})$-time algorithm.
- **Goal**: runtime $f(\|\varphi\|) \cdot n^c$ for a constant c and some function f.
  - Called **fixed-parameter tractable**, or **FPT**, parameterized by $\|\varphi\|$.
- No such algorithm on general structures, unless $\mathrm{FPT} = \mathrm{AW}[\star]$.

Model checking on sparse graphs

- Our definition of sparsity is based on **local** contractions, so we should study **local** problems in this framework.
- What (meta-)class of problems is famously local?

FO model-checking

Input	Relational structure $\mathbb{M}$, FO sentence φ
Question	Does $\mathbb{M} \models \varphi$?

- **Trivial**: an $\mathcal{O}(n^{\|\varphi\|})$-time algorithm.
- **Goal**: runtime $f(\|\varphi\|) \cdot n^c$ for a constant c and some function f.
  - Called **fixed-parameter tractable**, or **FPT**, parameterized by $\|\varphi\|$.
- No such algorithm on general structures, unless $\mathrm{FPT} = \mathrm{AW}[\star]$.
- FPT algorithms for structures whose Gaifman graphs have bounded degree, are planar, H-minor-free, ...

FO model-checking dichotomy

Theorem [Grohe et al., Dvořák et al.]

Let $\mathcal{C}$ be a monotone graph class (closed under taking subgraphs). Then:

- If $\mathcal{C}$ is nowhere dense, then FO model-checking can be done in time $f(\varphi) \cdot n^{1+\varepsilon}$ on structures with Gaifman graphs from $\mathcal{C}$, for any $\varepsilon > 0$.
- If $\mathcal{C}$ is somewhere dense, then FO model-checking is AW[$\star$]-complete on structures with Gaifman graphs from $\mathcal{C}$.

FO model-checking dichotomy

Theorem [Grohe et al., Dvořák et al.]

Let $\mathcal{C}$ be a monotone graph class (closed under taking subgraphs). Then:

- If $\mathcal{C}$ is nowhere dense, then FO model-checking can be done in time $f(\varphi) \cdot n^{1+\varepsilon}$ on structures with Gaifman graphs from $\mathcal{C}$, for any $\varepsilon > 0$.
- If $\mathcal{C}$ is somewhere dense, then FO model-checking is AW[$\star$]-complete on structures with Gaifman graphs from $\mathcal{C}$.

- Nowhere denseness **exactly** characterizes monotone classes where FO model-checking is tractable from the parameterized viewpoint.

FO model-checking dichotomy

Theorem [Grohe et al., Dvořák et al.]

Let $\mathcal{C}$ be a monotone graph class (closed under taking subgraphs). Then:

- If $\mathcal{C}$ is nowhere dense, then FO model-checking can be done in time $f(\varphi) \cdot n^{1+\varepsilon}$ on structures with Gaifman graphs from $\mathcal{C}$, for any $\varepsilon > 0$.
- If $\mathcal{C}$ is somewhere dense, then FO model-checking is AW[$\star$]-complete on structures with Gaifman graphs from $\mathcal{C}$.

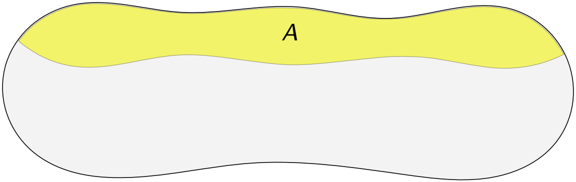
- Nowhere denseness **exactly** characterizes monotone classes where FO model-checking is tractable from the parameterized viewpoint.
- Provides a natural barrier for locality-based methods.

Neighborhood complexity

- Initially introduced for kernelization algorithms on sparse graphs.

Neighborhood complexity

- Initially introduced for kernelization algorithms on sparse graphs.
- Suppose we have a graph G and a subset of vertices $A \subseteq V(G)$.

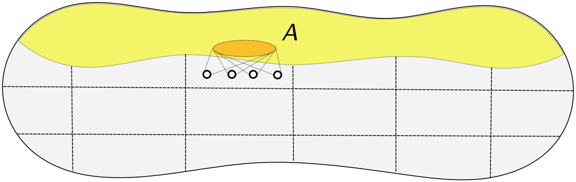


Neighborhood complexity

- Initially introduced for kernelization algorithms on sparse graphs.
- Suppose we have a graph G and a subset of vertices $A \subseteq V(G)$.
- For fixed $r \in \mathbb{N}$, define the following equivalence relation on $V(G)$:

$$u \sim_r v \qquad \Leftrightarrow \qquad B_r(u) \cap A = B_r(v) \cap A,$$

where $B_r(u) = \{w : \mathrm{dist}(u, w) \leqslant r\}$.

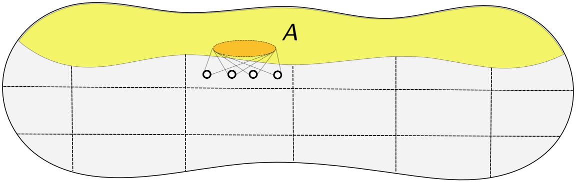


Neighborhood complexity

- Initially introduced for kernelization algorithms on sparse graphs.
- Suppose we have a graph G and a subset of vertices $A \subseteq V(G)$.
- For fixed $r \in \mathbb{N}$, define the following equivalence relation on $V(G)$:

$$u \sim_r v \qquad \Leftrightarrow \qquad B_r(u) \cap A = B_r(v) \cap A,$$

  where $B_r(u) = \{w\colon \mathrm{dist}(u, w) \leqslant r\}$.
- How many equivalence classes may $\sim_r$ have?

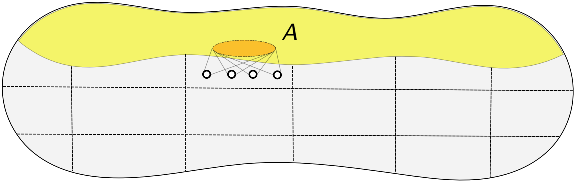


Neighborhood complexity

- Initially introduced for kernelization algorithms on sparse graphs.
- Suppose we have a graph G and a subset of vertices $A \subseteq V(G)$.
- For fixed $r \in \mathbb{N}$, define the following equivalence relation on $V(G)$:

$$u \sim_r v \qquad \Leftrightarrow \qquad B_r(u) \cap A = B_r(v) \cap A,$$

  where $B_r(u) = \{w : \mathrm{dist}(u, w) \leqslant r\}$.
- How many equivalence classes may $\sim_r$ have?
  - In general, even $2^{|A|}$.

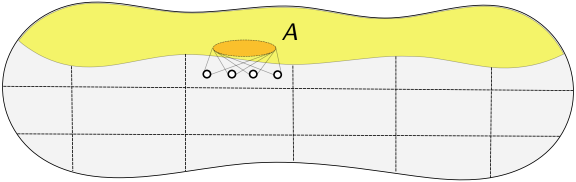


Neighborhood complexity

Theorem

Let $\mathcal{C}$ be a class of graphs, $G \in \mathcal{C}$, $A \subseteq V(G)$, and $r \in \mathbb{N}$.

- If $\mathcal{C}$ has bounded expansion, then $\mathrm{index}(\sim_r) \leqslant c|A|$ for some constant c depending only on $\mathcal{C}$ and r.
- If $\mathcal{C}$ is nowhere dense, then $\mathrm{index}(\sim_r) \leqslant c|A|^{1+\varepsilon}$ for any $\varepsilon > 0$ and some constant c depending on $\mathcal{C}, r, \varepsilon$.



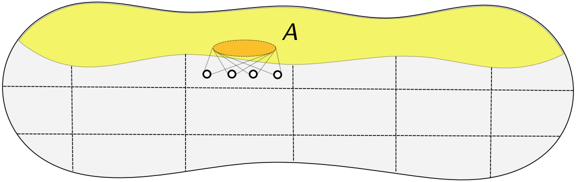
Neighborhood complexity

Theorem

Let $\mathcal{C}$ be a class of graphs, $G \in \mathcal{C}$, $A \subseteq V(G)$, and $r \in \mathbb{N}$.

- If $\mathcal{C}$ has bounded expansion, then $\mathrm{index}(\sim_r) \leqslant c|A|$ for some constant c depending only on $\mathcal{C}$ and r.
- If $\mathcal{C}$ is nowhere dense, then $\mathrm{index}(\sim_r) \leqslant c|A|^{1+\varepsilon}$ for any $\varepsilon > 0$ and some constant c depending on $\mathcal{C}, r, \varepsilon$.

- If $\mathcal{C}$ is somewhere dense and monotone, then for some r there is an example from $\mathcal{C}$ yielding $2^{|A|}$ classes.



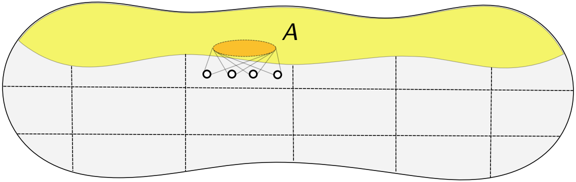
Neighborhood complexity

Theorem

Let $\mathcal{C}$ be a class of graphs, $G \in \mathcal{C}$, $A \subseteq V(G)$, and $r \in \mathbb{N}$.

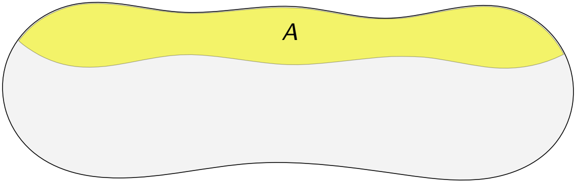
- If $\mathcal{C}$ has bounded expansion, then $\mathrm{index}(\sim_r) \leqslant c|A|$ for some constant c depending only on $\mathcal{C}$ and r.
- If $\mathcal{C}$ is nowhere dense, then $\mathrm{index}(\sim_r) \leqslant c|A|^{1+\varepsilon}$ for any $\varepsilon > 0$ and some constant c depending on $\mathcal{C}, r, \varepsilon$.

- If $\mathcal{C}$ is somewhere dense and monotone, then for some r there is an example from $\mathcal{C}$ yielding $2^{|A|}$ classes.
- **Now**: Proof for $r = 1$ and bounded expansion classes.



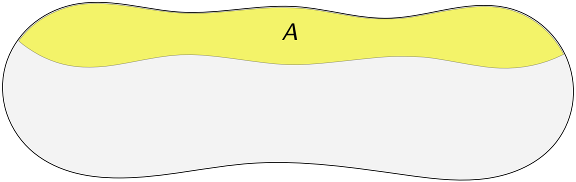
Proof for $r = 1$

- **Procedure**:



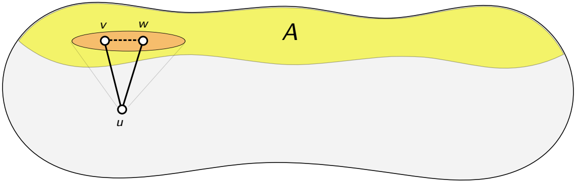
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.



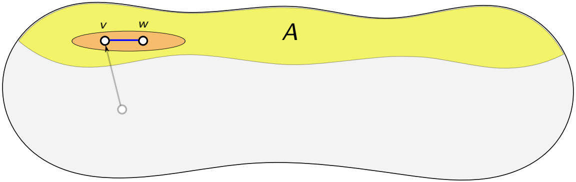
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.



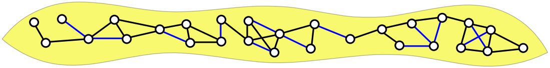
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.

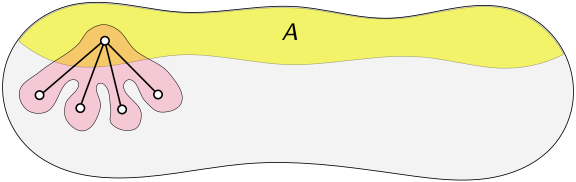


Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.
  - Finally, remove all vertices outside A.

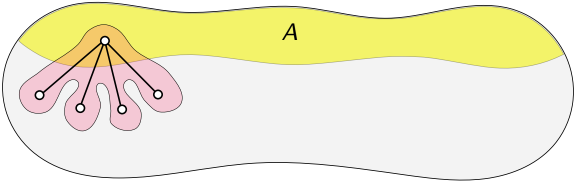


- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.
  - Finally, remove all vertices outside A.
- **Obs**: The obtained H is a depth-1 minor of G, thus H has $\leqslant d|A|$ edges, where d is the bound on edge density in $\mathcal{C}\nabla 1$.



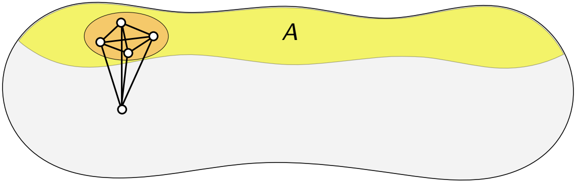
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.
  - Finally, remove all vertices outside A.
- **Obs**: The obtained H is a depth-1 minor of G, thus H has $\leqslant d|A|$ edges, where d is the bound on edge density in $\mathcal{C}\nabla 1$.
  - **Cor**: At most $d|A|$ vertices got contracted.



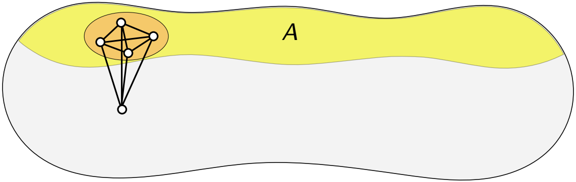
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.
  - Finally, remove all vertices outside A.
- **Obs**: The obtained H is a depth-1 minor of G, thus H has $\leqslant d|A|$ edges, where d is the bound on edge density in $\mathcal{C}\nabla 1$.
  - **Cor**: At most $d|A|$ vertices got contracted.
- After contractions, every vertex outside of A sees a clique in A.



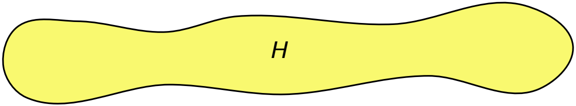
Proof for $r = 1$

- **Procedure**:
  - Start with $H = G$.
  - As long as there exists $u \notin A$ with $N(u) \cap A$ having a nonedge vw, contract u onto v.
  - Finally, remove all vertices outside A.
- **Obs**: The obtained H is a depth-1 minor of G, thus H has $\leqslant d|A|$ edges, where d is the bound on edge density in $\mathcal{C}\triangledown 1$.
  - **Cor**: At most $d|A|$ vertices got contracted.
- After contractions, every vertex outside of A sees a clique in A.
- **Need**: The number of cliques in H is linear in $|A|$.



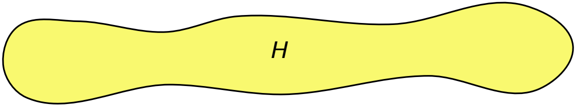
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.



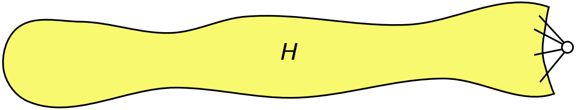
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.



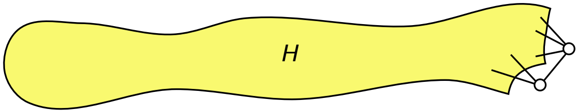
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.



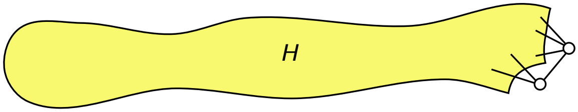
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.



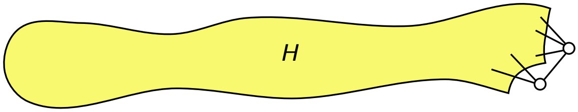
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.



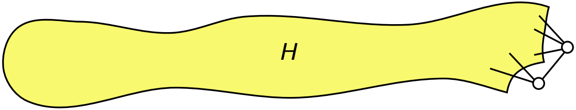
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.
- **Cor**: H contains at most $2^{2d} \cdot |A|$ cliques.



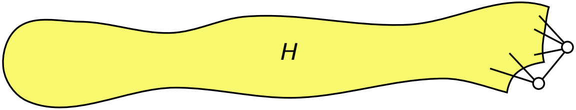
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.
- **Cor**: H contains at most $2^{2d} \cdot |A|$ cliques.
- **Conclusion**: $\sim_1$ has at most $(2^{2d} + d + 1) \cdot |A|$ classes.



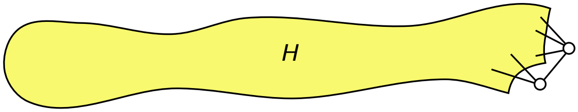
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.
- **Cor**: H contains at most $2^{2d} \cdot |A|$ cliques.
- **Conclusion**: $\sim_1$ has at most $(2^{2d} + d + 1) \cdot |A|$ classes.
  - $\leqslant |A|$ for vertices of A;



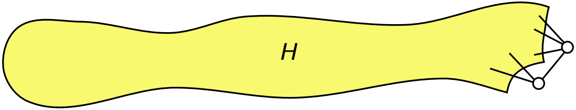
Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.
- **Cor**: H contains at most $2^{2d} \cdot |A|$ cliques.
- **Conclusion**: $\sim_1$ has at most $(2^{2d} + d + 1) \cdot |A|$ classes.
  - $\leqslant |A|$ for vertices of A;
  - $\leqslant d|A|$ for vertices contracted onto A;



Proof for $r = 1$

- **Obs**: H has a vertex of degree $\leqslant 2d$, and the same holds for every subgraph of H.
- **Procedure**: Iteratively remove a vertex of degree $\leqslant 2d$.
- **Obs**: Deleting every vertex removes at most 2^{2d} cliques.
- **Cor**: H contains at most $2^{2d} \cdot |A|$ cliques.
- **Conclusion**: $\sim_1$ has at most $(2^{2d} + d + 1) \cdot |A|$ classes.
  - $\leqslant |A|$ for vertices of A;
  - $\leqslant d|A|$ for vertices contracted onto A;
  - $\leqslant 2^{2d}|A|$ for vertices seeing a clique after contractions.



Part 2: Stability and types

Stability

- Let $\delta(\bar{x}, \bar{y}) \in \mathsf{FO}(\Sigma)$ and let $\mathbb{M}$ be a Σ-structure.

Stability

- Let $\delta(\bar{x},\bar{y}) \in \mathsf{FO}(\Sigma)$ and let $\mathbb{M}$ be a Σ-structure.

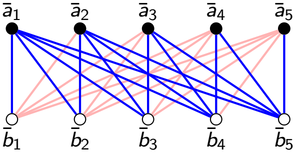
δ-ladder

A δ-**ladder** of length k in $\mathbb{M}$ is pair of tuple sequences

$$\bar{a}_1,\ldots,\bar{a}_k \in \mathbb{M}^{\bar{x}} \qquad \text{and} \qquad \bar{b}_1,\ldots,\bar{b}_k \in \mathbb{M}^{\bar{y}}$$

such that for all $i,j \in \{1,\ldots,k\}$ we have

$$\mathbb{M} \models \varphi(\bar{a}_i,\bar{b}_j) \qquad \text{if and only if} \qquad i \leqslant j.$$



Stability

- Let $\delta(\bar{x}, \bar{y}) \in \mathrm{FO}(\Sigma)$ and let $\mathbb{M}$ be a Σ-structure.

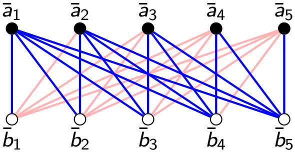
δ-ladder

A δ-**ladder** of length k in $\mathbb{M}$ is pair of tuple sequences

$$\bar{a}_1, \ldots, \bar{a}_k \in \mathbb{M}^{\bar{x}} \qquad \text{and} \qquad \bar{b}_1, \ldots, \bar{b}_k \in \mathbb{M}^{\bar{y}}$$

such that for all $i, j \in \{1, \ldots, k\}$ we have

$$\mathbb{M} \models \varphi(\bar{a}_i, \bar{b}_j) \qquad \text{if and only if} \qquad i \leqslant j.$$



- **Def**: a class of structures $\mathcal{C}$ is **stable** if for every formula δ there is a finite bound on the length of δ-ladders in structures from $\mathcal{C}$.

Stability

- Let $\delta(\bar{x}, \bar{y}) \in \mathrm{FO}(\Sigma)$ and let $\mathbb{M}$ be a Σ-structure.

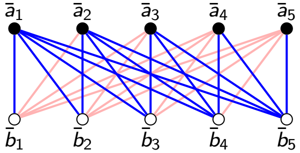
δ-ladder

A δ-**ladder** of length k in $\mathbb{M}$ is pair of tuple sequences

$$\bar{a}_1, \ldots, \bar{a}_k \in \mathbb{M}^{\bar{x}} \qquad \text{and} \qquad \bar{b}_1, \ldots, \bar{b}_k \in \mathbb{M}^{\bar{y}}$$

such that for all $i, j \in \{1, \ldots, k\}$ we have

$$\mathbb{M} \models \varphi(\bar{a}_i, \bar{b}_j) \qquad \text{if and only if} \qquad i \leqslant j.$$



- **Def**: a class of structures $\mathcal{C}$ is **stable** if for every formula δ there is a finite bound on the length of δ-ladders in structures from $\mathcal{C}$.
- **Intuition**: One cannot define arbitrary long linear orders.

Stability theory

- **Goal**: Study complexity of theories (classes of models defined by sets of sentences) and find delimiting conditions for structural properties.

- **Goal**: Study complexity of theories (classes of models defined by sets of sentences) and find delimiting conditions for structural properties.
- **Stability** is one of key conditions.

Stability theory

- **Goal**: Study complexity of theories (classes of models defined by sets of sentences) and find delimiting conditions for structural properties.
- **Stability** is one of key conditions.
- Pioneered by Morley and Shelah.

Stability theory

- **Goal**: Study complexity of theories (classes of models defined by sets of sentences) and find delimiting conditions for structural properties.
- **Stability** is one of key conditions.
- Pioneered by Morley and Shelah.
- **Motivation**: Question about the number of **types**.

Types in model theory

- Let Σ be a signature, let $\mathbb{M}$ be a Σ-structure, and let $A \subseteq \mathbb{M}$.

Types in model theory

- Let Σ be a signature, let $\mathbb{M}$ be a Σ-structure, and let $A \subseteq \mathbb{M}$.

δ-type

For an FO formula $\delta(\bar{x}, \bar{y})$ and $\bar{a} \in \mathbb{M}^{\bar{x}}$, the δ-**type** of $\bar{a}$ on A is

$$\mathrm{tp}_\delta(\bar{a}/A) = \left\{ \bar{b} \in A^{\bar{y}} : \mathbb{M} \models \delta(\bar{a}, \bar{b}) \right\}.$$

Types in model theory

- Let Σ be a signature, let $\mathbb{M}$ be a Σ-structure, and let $A \subseteq \mathbb{M}$.

δ-type

For an FO formula $\delta(\bar{x}, \bar{y})$ and $\bar{a} \in \mathbb{M}^{\bar{x}}$, the δ-**type** of $\bar{a}$ on A is

$$\mathrm{tp}_\delta(\bar{a}/A) = \left\{\bar{b} \in A^{\bar{y}} : \mathbb{M} \models \delta(\bar{a}, \bar{b})\right\}.$$

- **Example**: If Σ is the signature of graphs and $\delta(x, y) = \text{"}\mathrm{dist}(x, y) \leqslant r\text{"}$, then $\mathrm{tp}_\delta(a/A) = B_r(a) \cap A$.

Types in model theory

- Let Σ be a signature, let $\mathbb{M}$ be a Σ-structure, and let $A \subseteq \mathbb{M}$.

δ-type

For an FO formula $\delta(\bar{x}, \bar{y})$ and $\bar{a} \in \mathbb{M}^{\bar{x}}$, the δ-**type** of $\bar{a}$ on A is

$$\mathrm{tp}_\delta(\bar{a}/A) = \left\{ \bar{b} \in A^{\bar{y}} : \mathbb{M} \models \delta(\bar{a}, \bar{b}) \right\}.$$

- **Example**: If Σ is the signature of graphs and $\delta(x, y) = $"$\mathrm{dist}(x, y) \leqslant r$", then $\mathrm{tp}_\delta(a/A) = B_r(a) \cap A$.
- For a set of formulas Δ we put

$$\mathrm{tp}_\Delta(\bar{a}/A) = (\mathrm{tp}_\delta(\bar{a}/A))_{\delta \in \Delta} \qquad \text{and} \qquad \mathrm{tp}(\bar{a}/A) = \mathrm{tp}_{\mathsf{FO}(\Sigma)}(\bar{a}/A).$$

Types in model theory

- Let Σ be a signature, let $\mathbb{M}$ be a Σ-structure, and let $A \subseteq \mathbb{M}$.

δ-type

For an FO formula $\delta(\bar{x}, \bar{y})$ and $\bar{a} \in \mathbb{M}^{\bar{x}}$, the δ-**type** of $\bar{a}$ on A is

$$\mathrm{tp}_\delta(\bar{a}/A) = \left\{ \bar{b} \in A^{\bar{y}} : \mathbb{M} \models \delta(\bar{a}, \bar{b}) \right\}.$$

- **Example**: If Σ is the signature of graphs and $\delta(x,y) = \text{"}\mathrm{dist}(x,y) \leqslant r\text{"}$, then $\mathrm{tp}_\delta(a/A) = B_r(a) \cap A$.
- For a set of formulas Δ we put

$$\mathrm{tp}_\Delta(\bar{a}/A) = (\mathrm{tp}_\delta(\bar{a}/A))_{\delta \in \Delta} \qquad \text{and} \qquad \mathrm{tp}(\bar{a}/A) = \mathrm{tp}_{\mathsf{FO}(\Sigma)}(\bar{a}/A).$$

Theorem

A theory $\mathbb{T}$ is stable if and only if for some infinite cardinal κ, for every model $\mathbb{M}$ of $\mathbb{T}$ and set $A \subseteq \mathbb{M}$ with $|A| \geqslant \kappa$, the number of types over A has the same cardinality as A.

Part 3: Sparsity and types

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.
- Neighborhood complexity $\longleftrightarrow$ Types for distance formulas

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.
- Neighborhood complexity $\longleftrightarrow$ Types for distance formulas
- What about types w.r.t arbitrary formulas?

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.
- Neighborhood complexity $\longleftrightarrow$ Types for distance formulas
- What about types w.r.t arbitrary formulas?

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.
- Neighborhood complexity $\longleftrightarrow$ Types for distance formulas
- What about types w.r.t arbitrary formulas?

Theorem [P, Siebertz, Toruńczyk]

Let $\mathcal{C}$ be a class of finite Σ-structures, $\mathbb{M} \in \mathcal{C}$, $A \subseteq \mathbb{M}$, and $\delta(\bar{x}, \bar{y}) \in \mathrm{FO}$.

- If $\mathcal{C}$ has bounded expansion, then the number of δ-types over A is at most $c|A|^{|\bar{x}|}$, where c depends on $\mathcal{C}$ and δ.
- If $\mathcal{C}$ is nowhere dense, then for any $\varepsilon > 0$ the number of δ-types over A is at most $c|A|^{|\bar{x}|+\varepsilon}$, where c depends on $\mathcal{C}, \delta, \varepsilon$.

Sparsity and stability

Theorem [Adler and Adler, after Podewski and Ziegler]

Let $\mathcal{C}$ be a monotone class of finite graphs.
Then $\mathcal{C}$ is stable if and only if $\mathcal{C}$ is nowhere dense.

- **Idea**: Transfer tools and ideas between sparsity and stability.
- Neighborhood complexity $\longleftrightarrow$ Types for distance formulas
- What about types w.r.t arbitrary formulas?

Theorem [P, Siebertz, Toruńczyk]

Let $\mathcal{C}$ be a class of finite Σ-structures, $\mathbb{M} \in \mathcal{C}$, $A \subseteq \mathbb{M}$, and $\delta(\bar{x}, \bar{y}) \in \mathrm{FO}$.

- If $\mathcal{C}$ has bounded expansion, then the number of δ-types over A is at most $c|A|^{|\bar{x}|}$, where c depends on $\mathcal{C}$ and δ.
- If $\mathcal{C}$ is nowhere dense, then for any $\varepsilon > 0$ the number of δ-types over A is at most $c|A|^{|\bar{x}|+\varepsilon}$, where c depends on $\mathcal{C}, \delta, \varepsilon$.

- We now sketch the proof for graph classes of bounded expansion and $|\bar{x}|, |\bar{y}| = 1$.

Closing A

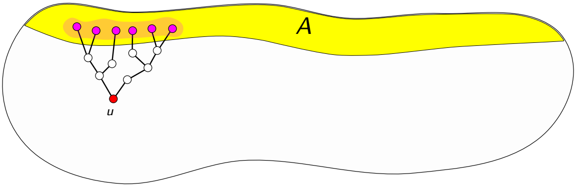
- Let r be the locality radius of δ.

Closing A

- Let r be the locality radius of δ.
- **Step 1**: Close A so that vertices $u \notin A$ have small r-projections.

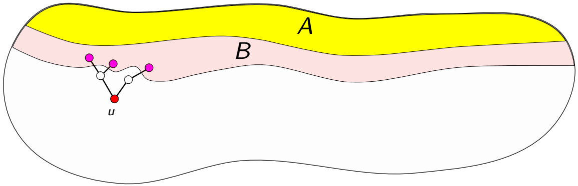
Closing A

- Let r be the locality radius of δ.
- **Step 1**: Close A so that vertices $u \notin A$ have small r-projections.
- r-projection of u onto A:
  vertices of A reachable from u by an A-avoiding paths of length $\leqslant r$.



Closing A

- Let r be the locality radius of δ.
- **Step 1**: Close A so that vertices $u \notin A$ have small r-projections.
- r-projection of u onto A:
  vertices of A reachable from u by an A-avoiding paths of length $\leqslant r$.
- **Fact**: There exists $B \supseteq A$ such that
  - $|B| \leqslant c|A|$ and
  - every r-projection of $u \notin B$ onto B has size $\leqslant c$.

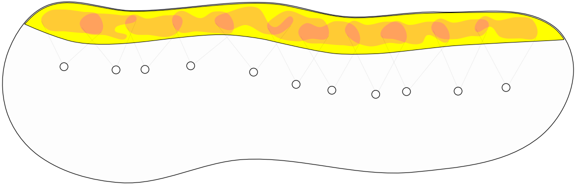


Classifying by projections

- **Cor**: We may assume that r-projections onto A are of constant size.

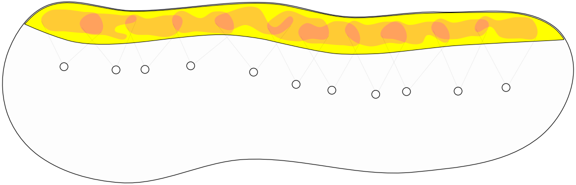
Classifying by projections

- **Cor**: We may assume that r-projections onto A are of constant size.
- **Fact**: The number of different such r-projections is $\leqslant c|A|$.



Classifying by projections

- **Cor**: We may assume that r-projections onto A are of constant size.
- **Fact**: The number of different such r-projections is $\leqslant c|A|$.
- **Ergo**: It suffices do bound the number of types for each possible distance-r projection by a constant.

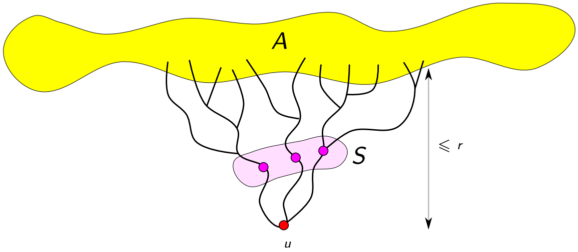


Locality

Feferman-Vaught Lemma

For every formula $\delta(x, y) \in \mathsf{FO}(\Sigma)$ there exists $r \in \mathbb{N}$ and a finite set of formulas Δ such that the following holds.
Let G be a graph, u be a vertex, and A and S be subsets of vertices such that every path of length $\leqslant r$ from u to A passes through S. Then

$$\mathrm{tp}_\delta(u/A) \qquad \text{is determined by} \qquad \mathrm{tp}_\Delta(u/S).$$



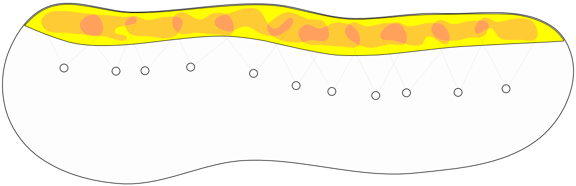
Locality

Feferman-Vaught Lemma

For every formula $\delta(x, y) \in \mathsf{FO}(\Sigma)$ there exists $r \in \mathbb{N}$ and a finite set of formulas Δ such that the following holds.
Let G be a graph, u be a vertex, and A and S be subsets of vertices such that every path of length $\leqslant r$ from u to A passes through S. Then

$$\mathrm{tp}_\delta(u/A) \qquad \text{is determined by} \qquad \mathrm{tp}_\Delta(u/S).$$

- If S is the r-projection of u on A, then S r-separates u from A.

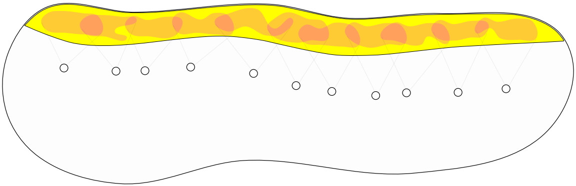


Feferman-Vaught Lemma

For every formula $\delta(x, y) \in \mathrm{FO}(\Sigma)$ there exists $r \in \mathbb{N}$ and a finite set of formulas Δ such that the following holds.
Let G be a graph, u be a vertex, and A and S be subsets of vertices such that every path of length $\leqslant r$ from u to A passes through S. Then

$$\mathrm{tp}_\delta(u/A) \qquad \text{is determined by} \qquad \mathrm{tp}_\Delta(u/S).$$

- If S is the r-projection of u on A, then S r-separates u from A.
- The number of Δ-types on S of constant size is constant.



Conclusions

- **Research programme**

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.
- **Stability**: Wealth of structural insight and abstract techniques accumulated over the years.

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.
- **Stability**: Wealth of structural insight and abstract techniques accumulated over the years.
- **Transfer** techniques, ideas, and questions between the two.

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.
- **Stability**: Wealth of structural insight and abstract techniques accumulated over the years.
- **Transfer** techniques, ideas, and questions between the two.
  - So far, successful transfer from stability to sparsity.

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.
- **Stability**: Wealth of structural insight and abstract techniques accumulated over the years.
- **Transfer** techniques, ideas, and questions between the two.
  - So far, successful transfer from stability to sparsity.
  - **Goal**: Results on stable graph classes, beyond sparsity.

Conclusions

- **Research programme**
- **Sparsity**: Wealth of tools and problems with combinatorial and algorithmic motivation.
- **Stability**: Wealth of structural insight and abstract techniques accumulated over the years.
- **Transfer** techniques, ideas, and questions between the two.
  - So far, successful transfer from stability to sparsity.
  - **Goal**: Results on stable graph classes, beyond sparsity.
- **Thank you for your attention!**