Model Checking on Sparse Graphs

Stephan Kreutzer

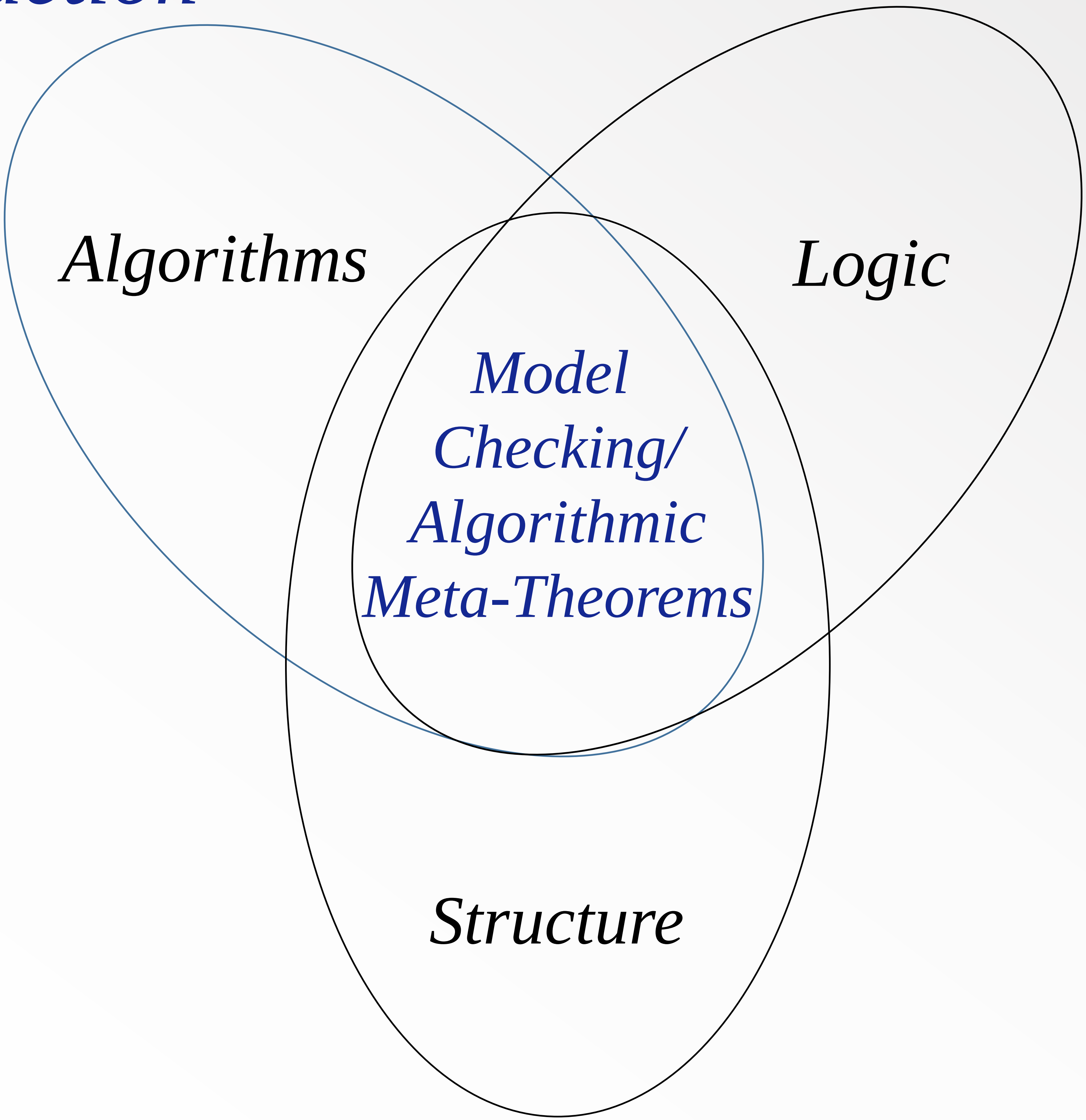
Technical University Berlin

Algorithms, Logic and Structure

Warwick

12.12.2016

Introduction

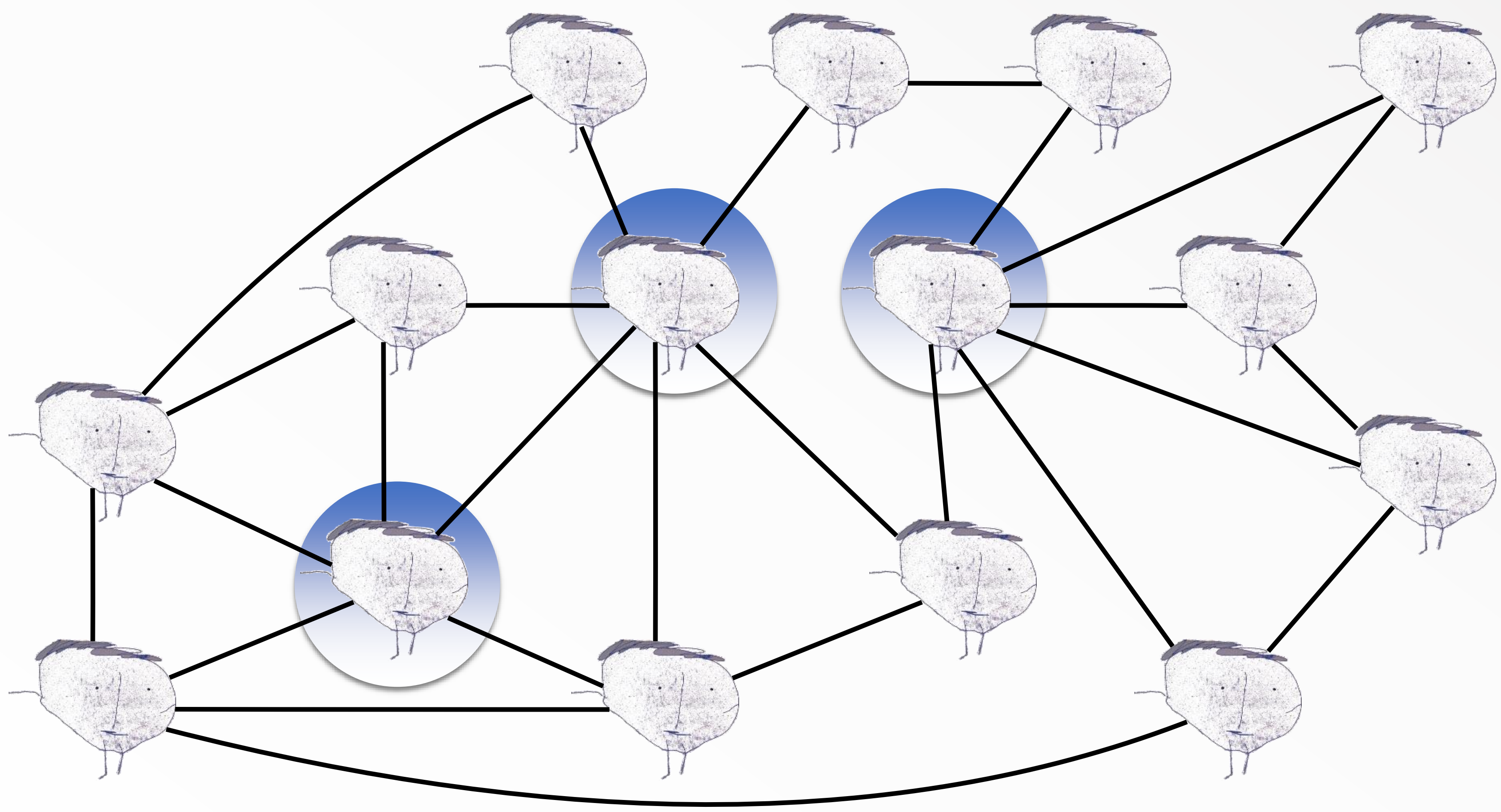


Introduction

Dominating Set

Input: Graph G, number k

Problem: Find a set $S \subseteq V(G)$ with $|S| \leq k$ such that for all $v \in V(G) \setminus S$ there is a $u \in S$ with $\{u, v\} \in E(G)$.

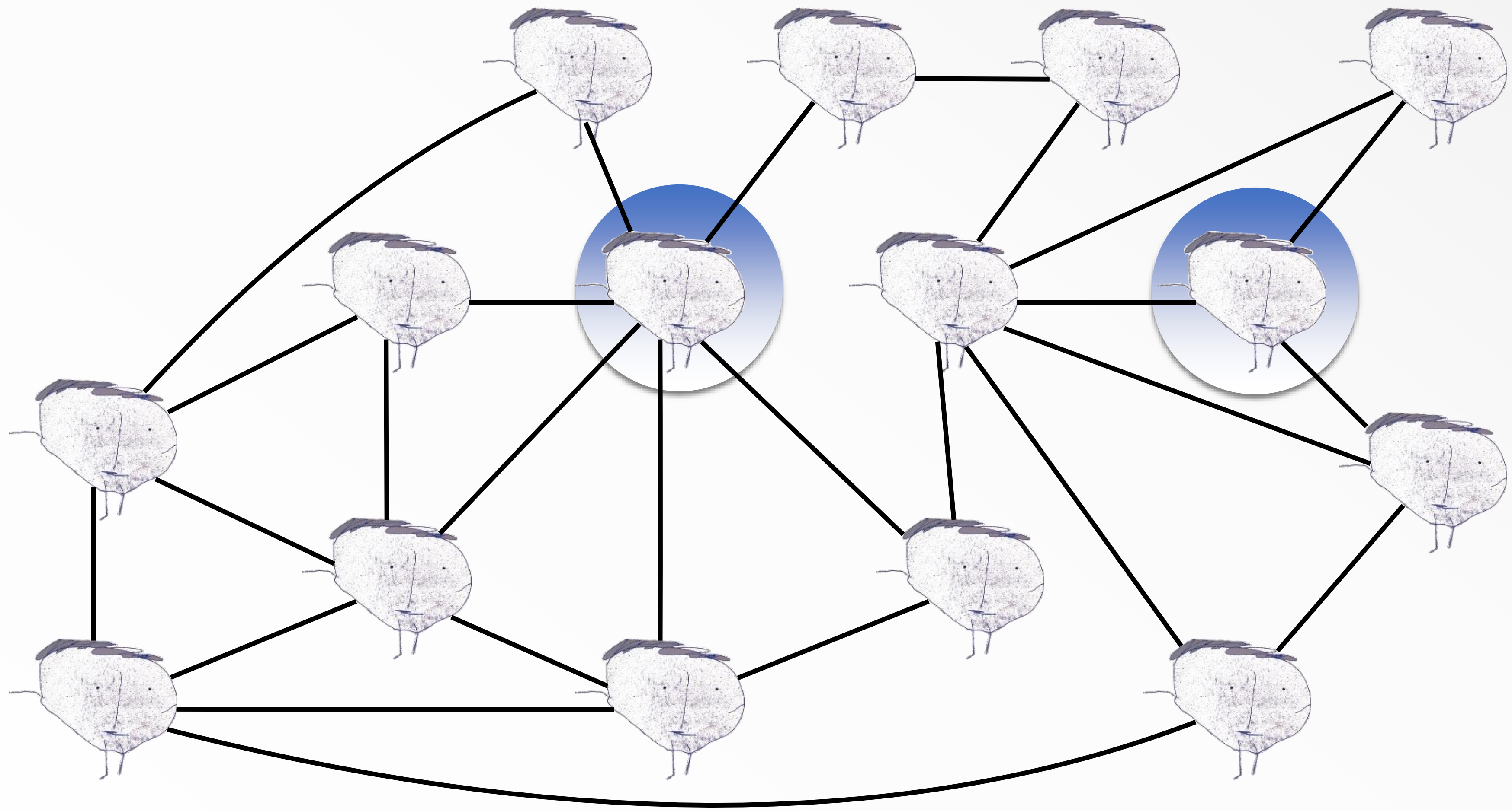


Introduction

Distance d Dominating Set

Input: Graph G, numbers k, d

Problem: Find a set $S \subseteq V(G)$ with $|S| \leq k$ such that for all $v \in V(G) \setminus S$ there is a $u \in S$ with $dist(u, v) \leq d$.

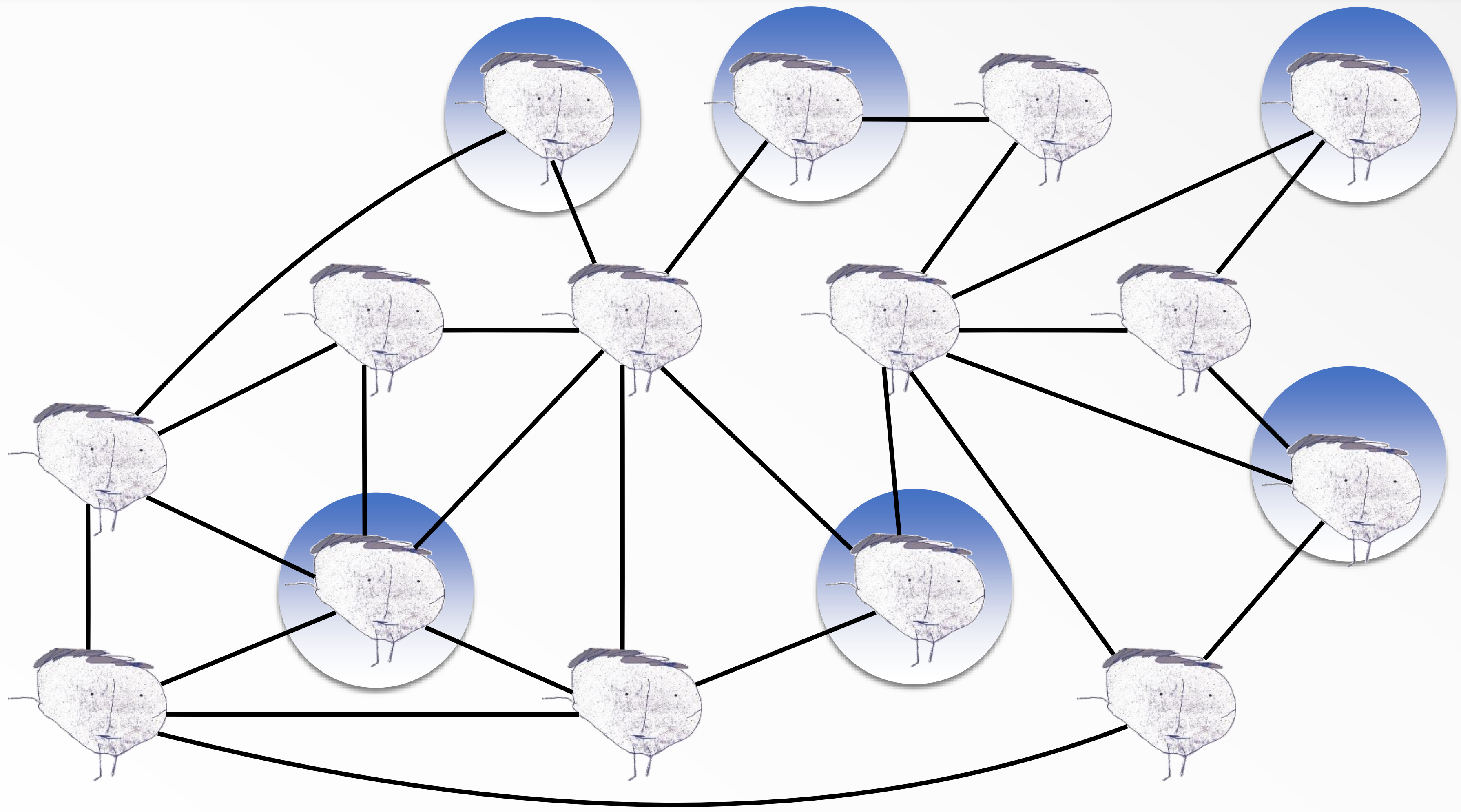


Introduction

Independent Set

Input: Graph G, number k

Problem: Find a set $S \subseteq V(G)$ with $|S| \geq k$ such that $\{u, v\} \notin E(G)$ for all $u, v \in S$.

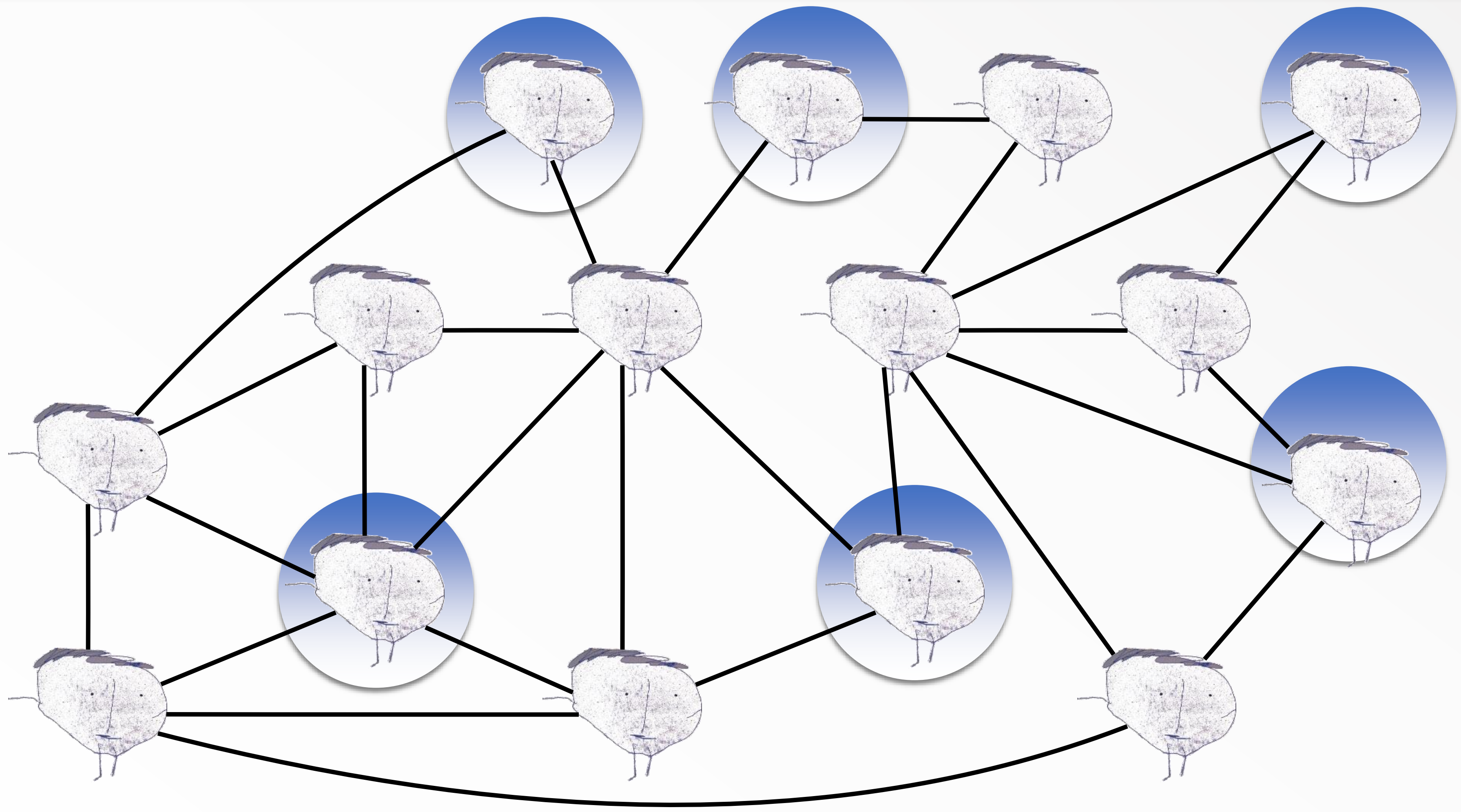


Introduction

Distance d Independent Set

Input: Graph G, numbers k, d

Problem: Find a set $S \subseteq V(G)$ with $|S| \geq k$ such that $\text{dist}(u, v) > d$ for all $u, v \in S$.



Complexity

These are only two examples of standard algorithmic problems on graphs.

Examples.

- Network Centres / Facility location problems / dominating sets
- Clique, Independent Set, Subgraph Containment
- Network design problems: Steiner trees or networks
- k-disjoint paths, Hamiltonian paths
- *k-Colourability*

Complexity.

The corresponding decision problems are all NP-complete in general.

Hence, it is expected that no efficient algorithms solving them exist.

Managing Complexity

Dealing with the complexity.

- Design heuristics
- Design exact algorithms optimising the (exponential) running time.
- Approximation algorithms
- Identify special classes of admissible inputs on which the problems become tractable.

Restricted cases sufficient for applications.

We may have additional information about the structure of inputs.

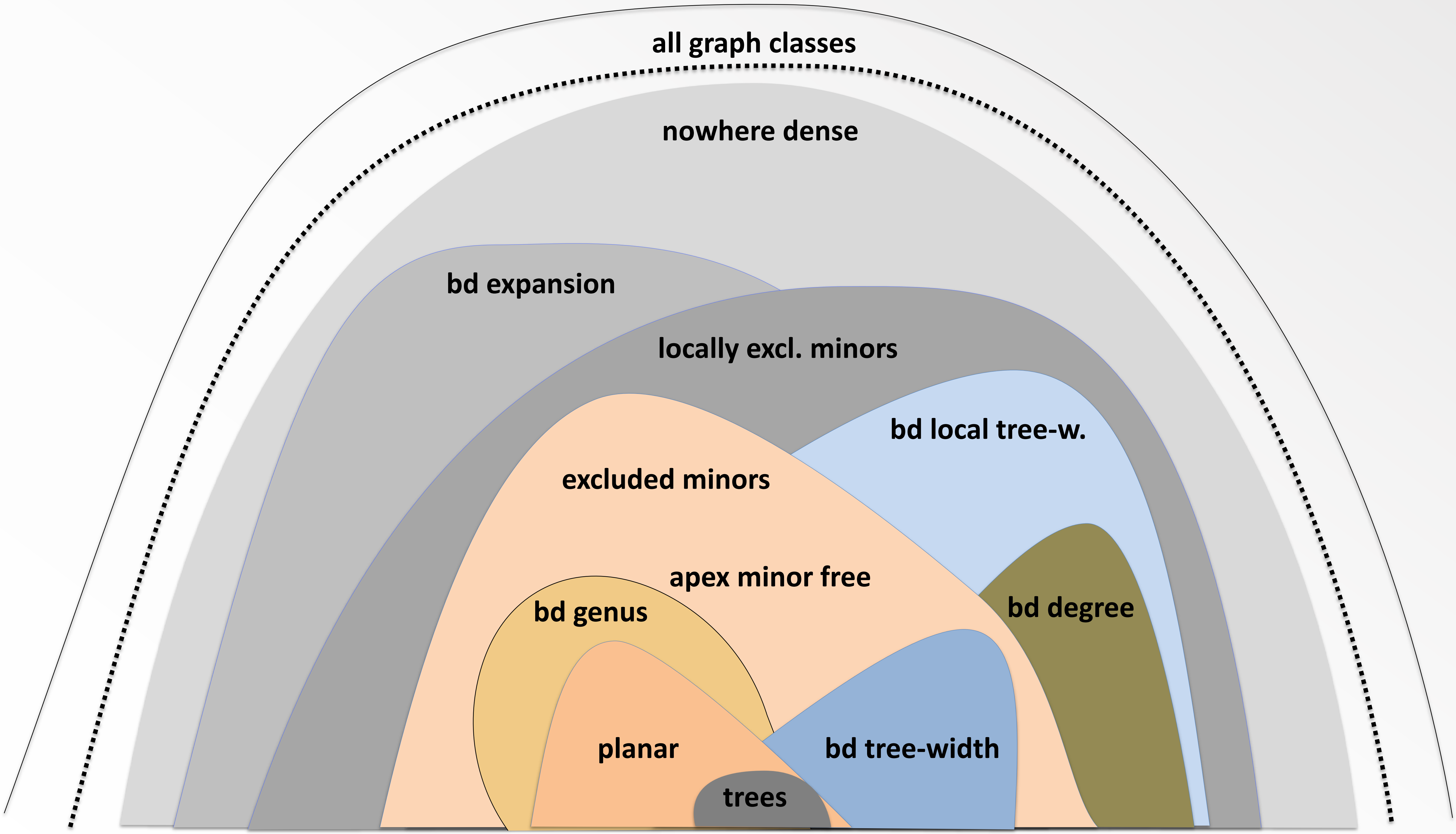
Examples.

road maps: almost planar

communication networks: sparse (moderate number of edges)

Aim. Develop efficient algorithms restricted to special classes of graphs.

Structural Graph Theory to the Rescue



Models of Efficiency

Models of efficiency.

Solvability in polynomial time.

But Dominating Set is NP-hard on very restricted classes of graphs.

In concrete applications.

The dominating set might be rather small (PC size 15).

So $2^{O(\sqrt{k})} \cdot n$ may be ok but $2^{\sqrt{n}}$ would not.

Parameterized Complexity.

Restrict the exponential behaviour to a specific/small part of the input.

For instance the size of the solution or some structural parameter.

Try to find algorithms running in time

$$2^{O(\sqrt{k})} \cdot n \quad \text{or} \quad 2^{O(k)} \cdot n \quad \text{or} \quad f(k) \cdot n^c$$

Parameterized Complexity

Parameterized Problem. ***Pair*** (P, k).

Independent Set

Input: Graph G, number k

Parameter: k *(or* $k+d$ *or a structural par.: tree width, excluded* K_t*)*

Problem: Find $S \subseteq V(G)$ with $|S| \geq k$ st $\{u, v\} \notin E(G)$ for all $u, v \in S$.

Definition.

A parameterized problem (P, k) is called **fixed-parameter tractable** if it can be solved in time $f(k) \cdot n^c$ for some computable function f and constant c.

Parameterized intractability. W[1]-hardness, W[2] …..

Problems such as Independent/Dominating Set … W[1]-hard on general graphs

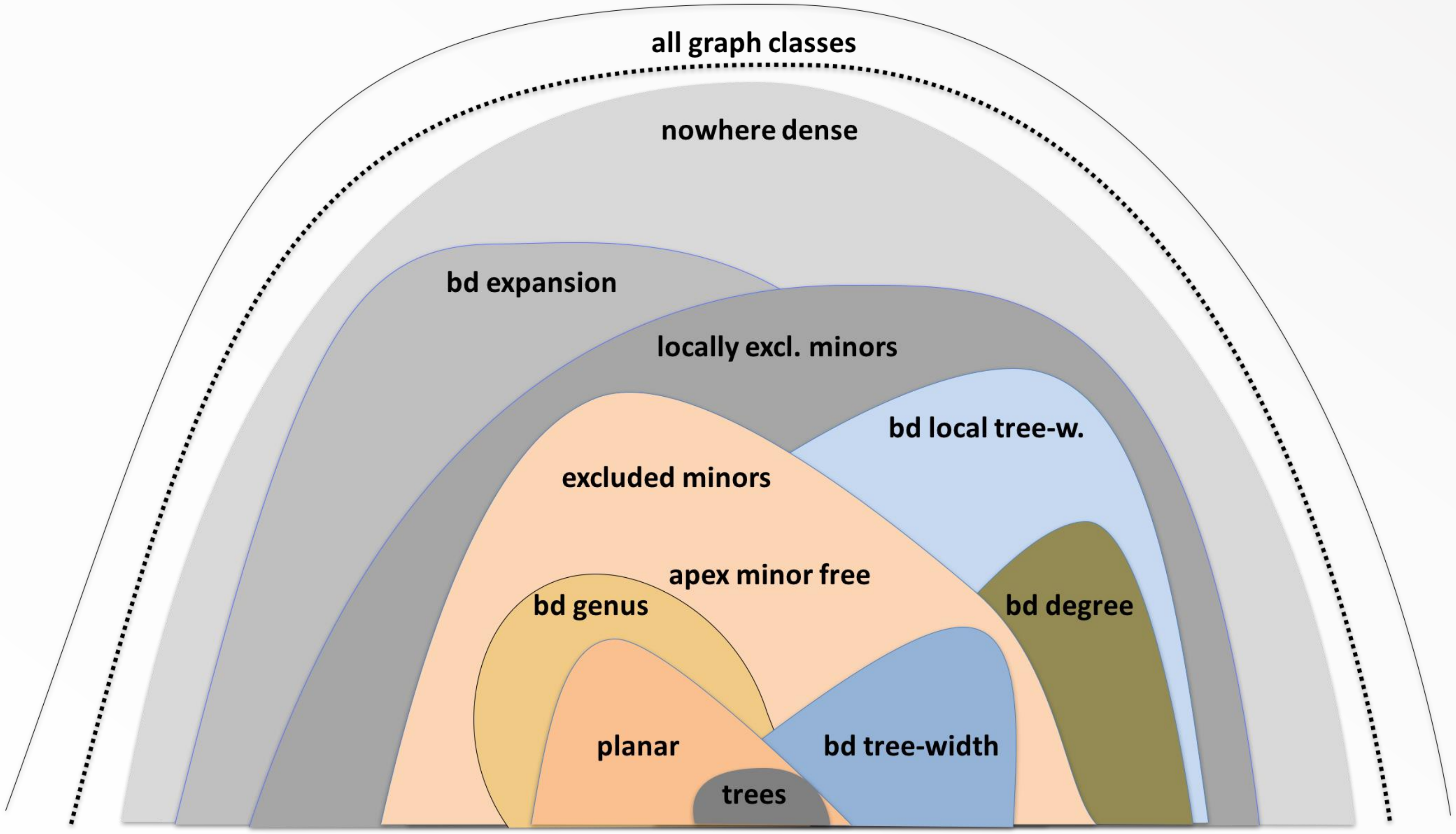
Algorithmic Graph Structure Theory

Design of Algorithms.

A lot of research has gone into developing and improving algorithms for specific problems on certain classes of graphs.

How far can we go up in this hierarchy of graph classes for problems such as dominating sets?

Is tractability of dominating set on degenerate graphs merely a special case or a witness of a general phenomenon with broad algorithmic applications.



Algorithmic Graph Structure Theory

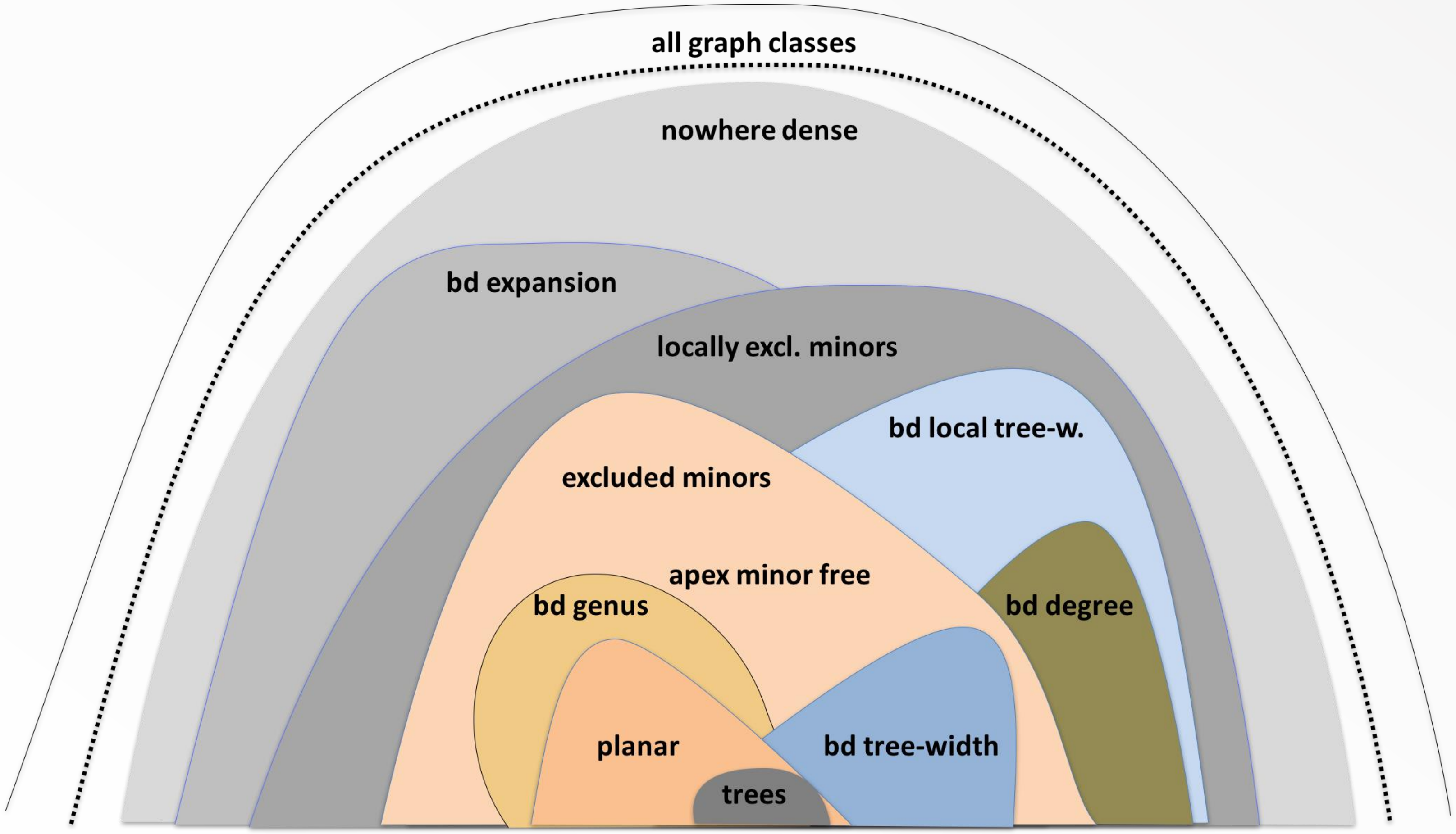
In this talk.

Explore the general frontier of tractability for *natural types* of problems.

Identify *natural classes of problems*!

Simple way of proving that a problem is tractable on a specific graph class.

How can we identify promising classes of graphs?



Enters Logic

Meta-Theorems

Meta-Theorems.

> *Every problem satisfying certain criteria L are tractable on every class of graphs satisfying a property P.*

Ideally, infer tractability of a problem directly from its mathematical description.

Theorem. Every graph property that can be described using only

- *there is a/for all sets of edges/vertices*
- *there is a/for all vertices/edges*
- *Boolean combinations*
- *there is an edge between u and v …*

can be solved in linear time on any class of graphs of bounded tree-width.

Example. *3-Colourability*

> A graph G is *3-Colourable* if there are 3 sets of vertices such that every vertex of G belongs to a set and for all edges both endpoints have different colours.

Courcelle's Theorem

Courcelle's Theorem. (Courcelle'90)

> *Every algorithmic property definable in monadic second-order logic can be decided in linear time on any class of graphs of bounded tree width.*

Monadic Second-Order Logic (MSO).

Hamiltonian path. $\exists P \subseteq E(G) . (P \textit{ is a path} \wedge \forall x. x \in V(P))$

3-Colourability.

$$\exists C_1 C_2 C_3 . \forall x \, (x \in C_1 \cup C_2 \cup C_3) \wedge \forall \{u, v\} \in E(G) . \neg \bigvee_{i=1}^{3} (u \in C_i \wedge v \in C_i)$$

Efficient implementation. *Sequoia project by Langer, Rossmanith … '10*

- Implementation of MSO evaluation on graphs using tree width
- Has been used for real world tasks: computing optimal placement of radio transmitters for Hannover urban train network
- For some problems even beats ILP-based approaches using CPLEX
- Run time very close to optimal (theoretical) lower bounds for Dominating Set

Algorithmic Meta-Theorems

Theorem. (Courcelle ’90)

Every graph property definable in Monadic Second-Order Logic can be decided in linear time on any class of graphs of bounded tree width.

General form of an algorithmic meta theorem.

Every problem definable in a given logic $\mathcal{L}$ is tractable on any class $\mathcal{C}$ of graphs satisfying a certain property.

Rephrased in parameterized complexity. Let $\mathcal{C}$ be a class of graphs.

Then the following problem is fixed parameter tractable

MC($\mathcal{L}$, $\mathcal{C}$)	
Input:	Graph $G \in \mathcal{C}$, formula $\varphi \in \mathcal{L}$
Parameter:	$\lvert\varphi\rvert$ (or $\lvert\varphi\rvert + tw(G)$ or $\lvert\varphi\rvert + excluded\ K_t$)
Problem:	Decide $G \models \varphi$?

Algorithmic Meta-Theorems

Rephrased in parameterized complexity. Let $\mathcal{C}$ be a class of graphs.

Then the following problem is fixed parameter tractable

MC($\mathcal{L}$, $\mathcal{C}$)
Input: Graph $G \in \mathcal{C}$, formula $\varphi \in \mathcal{L}$
Parameter: $|\varphi|$ (or $|\varphi| + tw(G)$ or $|\varphi| + excluded\ K_t$)
Problem: Decide $G \models \varphi$?

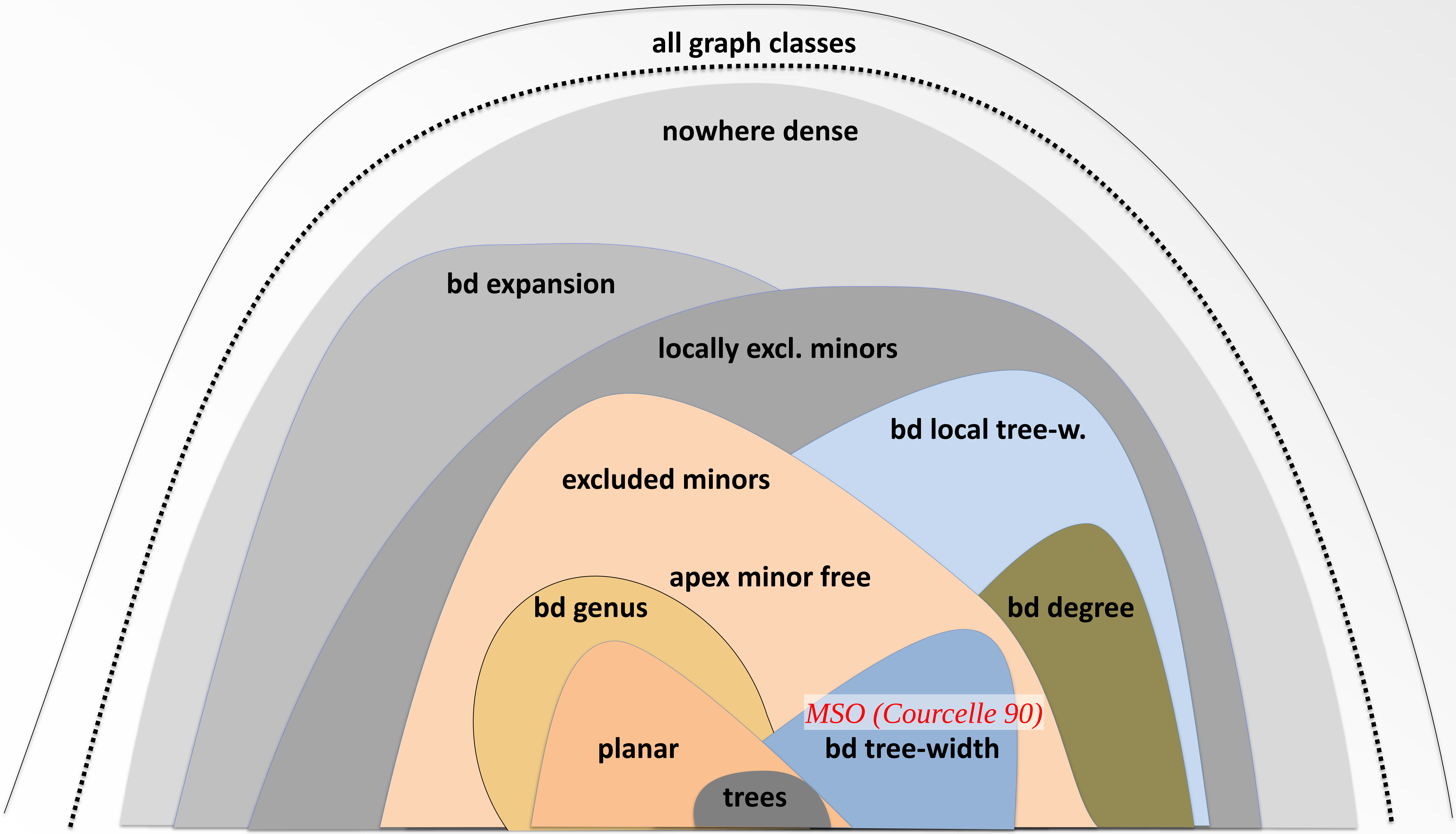
Research programme.

For important logics $\mathcal{L}$ such as first-order or monadic second-order logic:

identify structural parameter $\mathcal{P}$ such that

MC($\mathcal{C}$, $\mathcal{L}$) is FPT for a class $\mathcal{C}$ if, and only if, $\mathcal{C}$ has property $\mathcal{P}$.

Algorithmic Meta-Theorems



Characterising MSO Tractability

Theorem. (K., Tazari ’10)

Let $\mathcal{C}$ be a class of graphs closed under sub-graphs.

If the tree width of $\mathcal{C}$ is not poly-logarithmically, or $\log^{28} n$, bounded then $MC(MSO, \mathcal{C})$ is not fpt unless SAT can be solved in sub-exponential time.

Theorem. (Courcelle ’90)

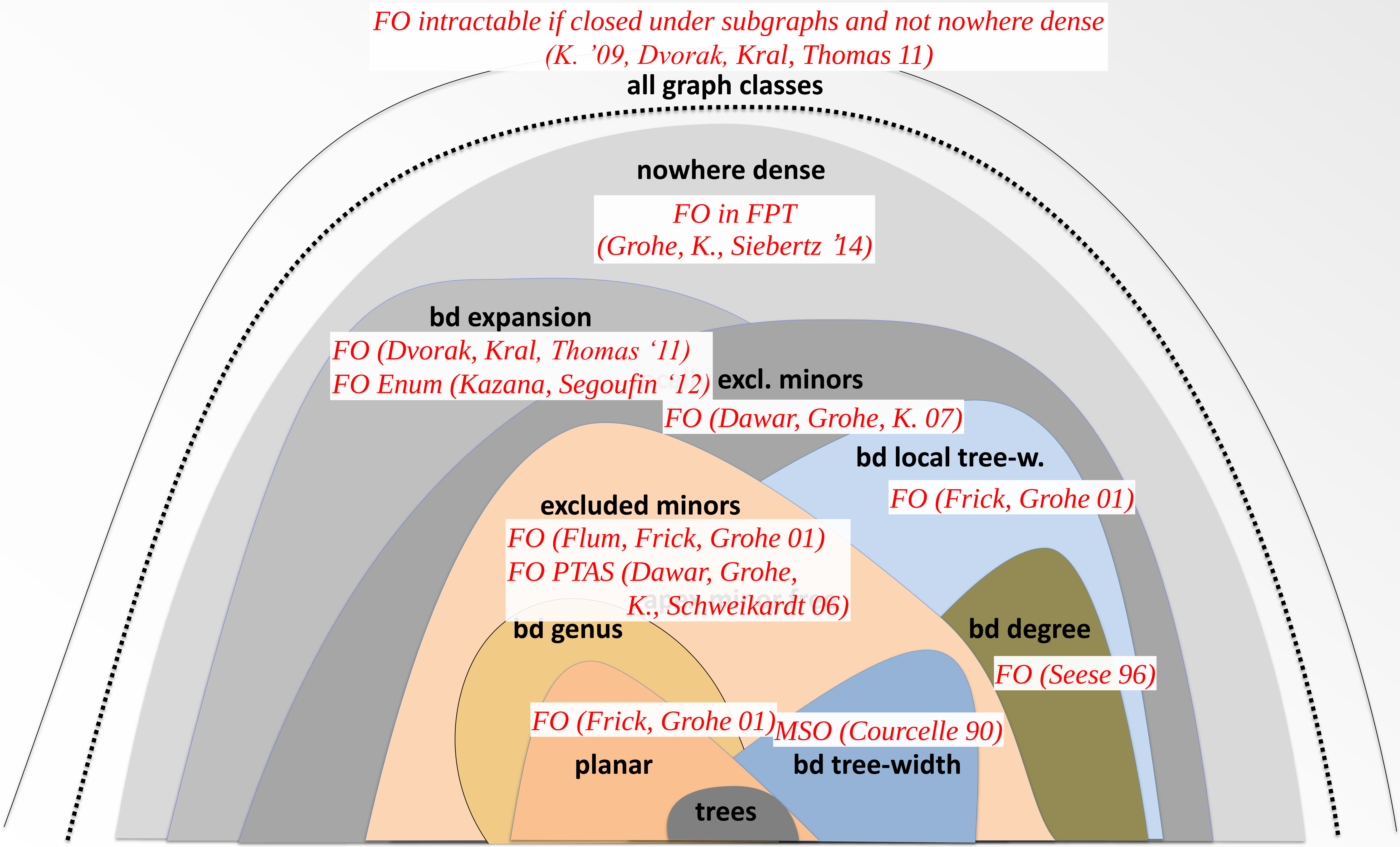
Monadic Second-Order Logic (MSO) is fixed-parameter tractable on any class of graphs of constant tree width.

Note. This applies to MSO_2 and MSO_1 and tree width. The lower bound for MSO_1 and clique-width is open.

Theorem. (Seese ‘96)

Every graph property definable in first-order logic can be decided in linear time on any class of graphs of bounded maximum degree.

Algorithmic Meta-Theorems

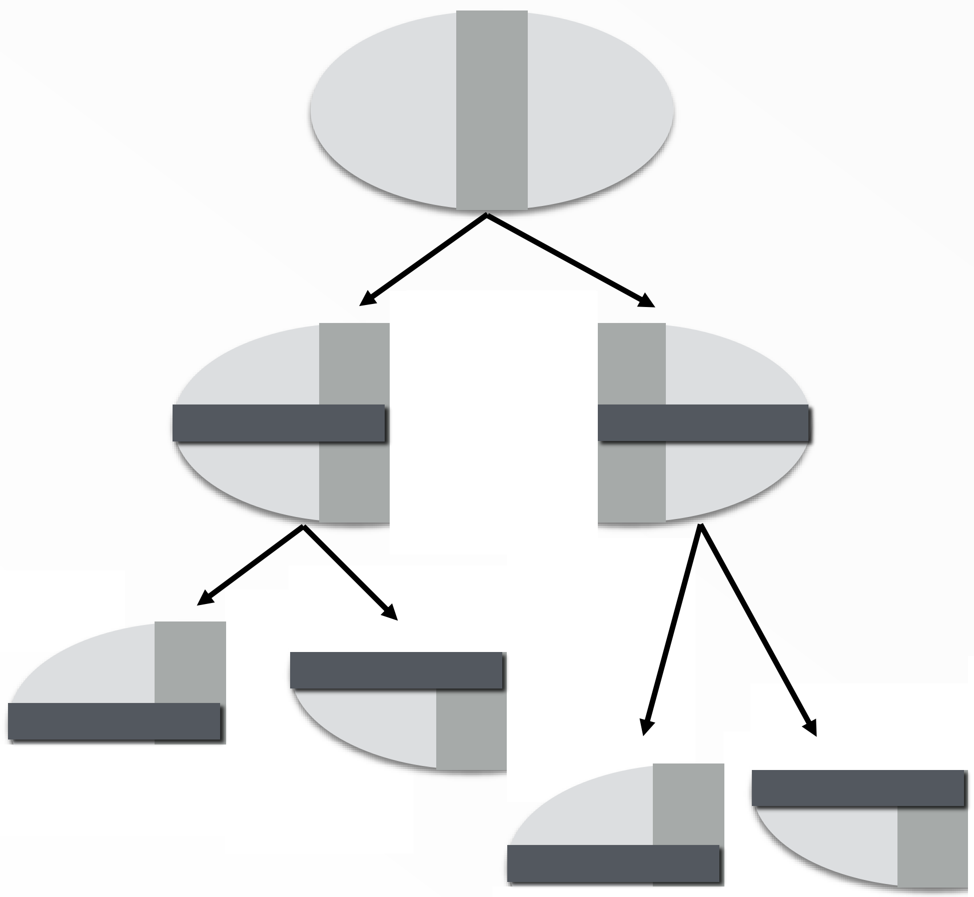


Logic vs. Structure vs. Algorithms

Structure

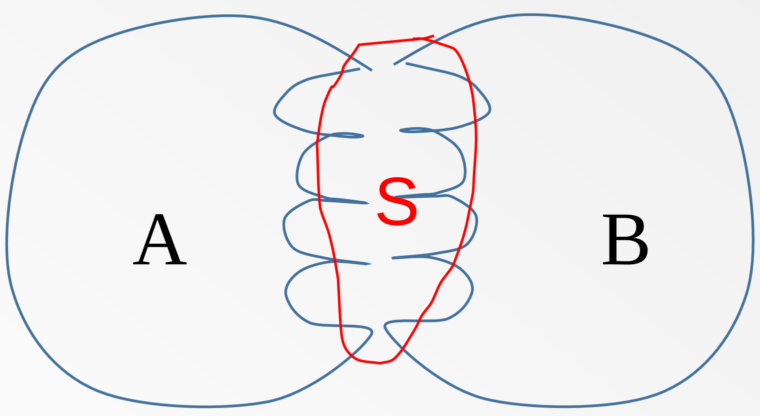
Tree width.

Recursively split along constant size separators.



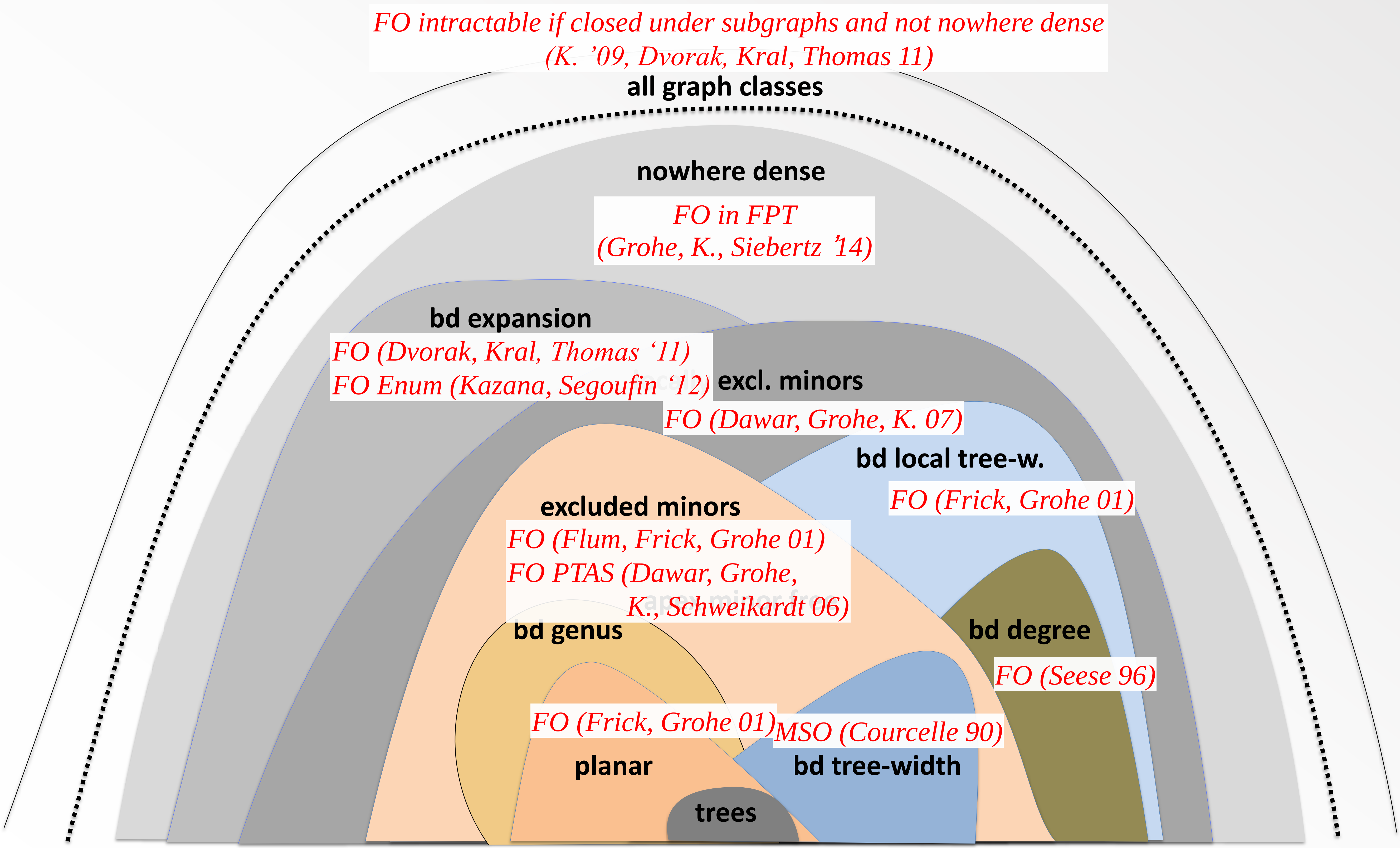
Logic

Feferman-Vaught Theorems.



Knowing the set of formulas true at S in A and the set of formulas true at S in B is enough to compute the set of formulas true at S in $A \cup B$.

Algorithmic Meta-Theorems



Model Checking on Nowhere Dense Classes

Theorem. ***(Grohe, K., Siebertz '14)***

Every problem definable in first-order logic can be decided in time $O(n^{1+\epsilon})$, for every $\varepsilon > 0$, on any class of graphs that is nowhere dense.

Examples of first-order definable problems.

- Dominating Sets, Independent Sets, … (for fixed solution size)
- Steiner trees (for fixed solution size) etc.
- Subgraph Homo- or Isomorphism $H \rightarrow_{hom} G$ (for fixed H)

Theorem. ***(K. 09, Dvorak, Kral, Thomas '11)***

If a class C closed under subgraphs is not nowhere dense, then FO-model-checking is not fixed-parameter tractable (unless AW[$*$] = FPT).

Corollary.

Let $\mathcal{C}$ be a class of graphs closed under taking subgraphs.

$MC(FO, \mathcal{C}) \in FPT$ if, and only if, $\mathcal{C}$ is nowhere dense.

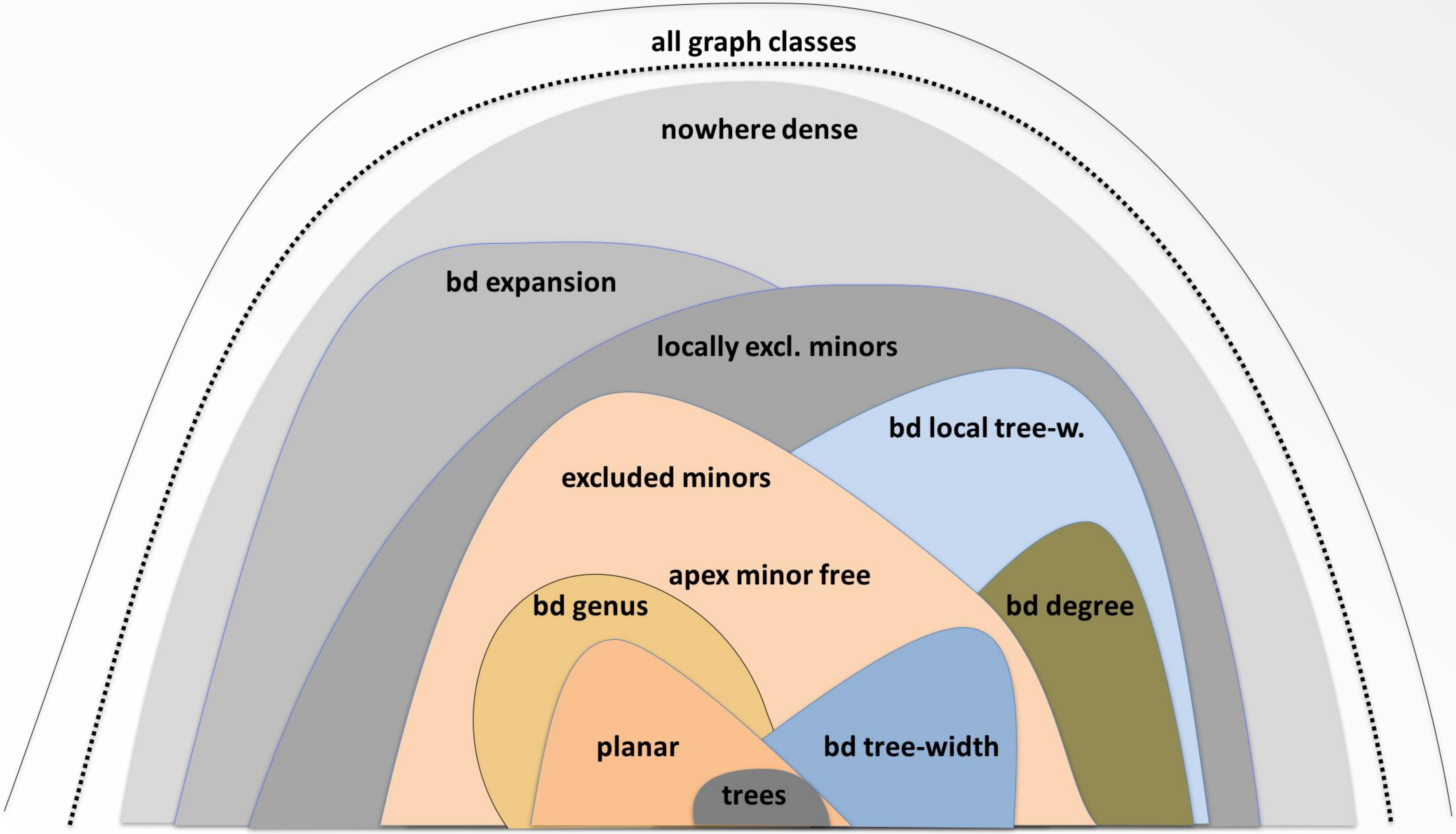
Nowhere Dense Classes of Graphs

Sparse Classes of Graphs

Observation.

The classes of graphs studied so far exhibit very different properties.

But they are all relatively *sparse*, i.e. a low number of edges.

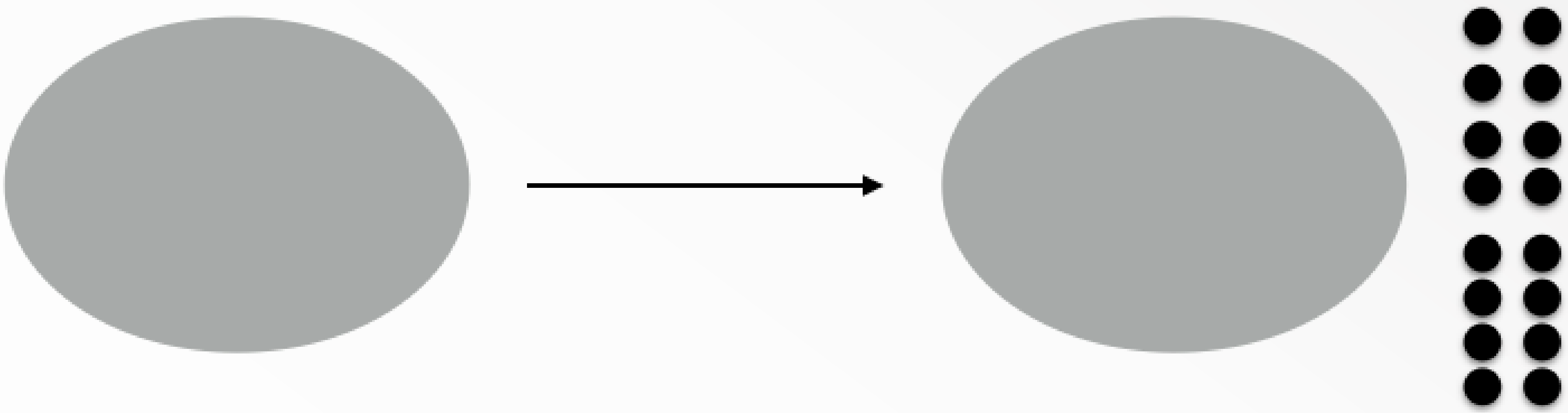


Sparse Classes of Graphs

Question. What are sparse graphs or sparse classes of graphs?

Attempt 1. *Bounded average degree*

Study classes of graphs G where $\frac{|E(G)|}{|V(G)|} \leq d$ for some constant d.



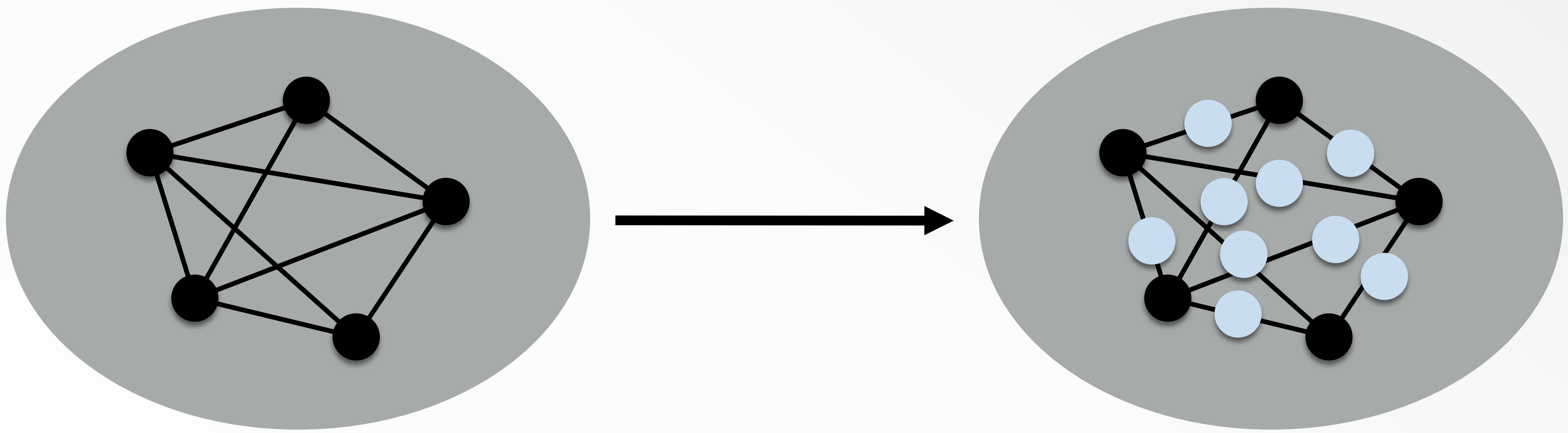
Property 1. A sparse class of graphs should be preserved by taking subgraphs.

Sparse Classes of Graphs

Property 1. A sparse class of graphs should be preserved by taking subgraphs.

Attempt 2. *Bounded degeneracy*

A graph is d-degenerate if every subgraph $H \subseteq G$ contains a vertex of degree $\leq d$.



Property 2. A sparse class of graphs should be preserved by "undoing" subdivisions of bounded length.

Nowhere dense classes of graphs.

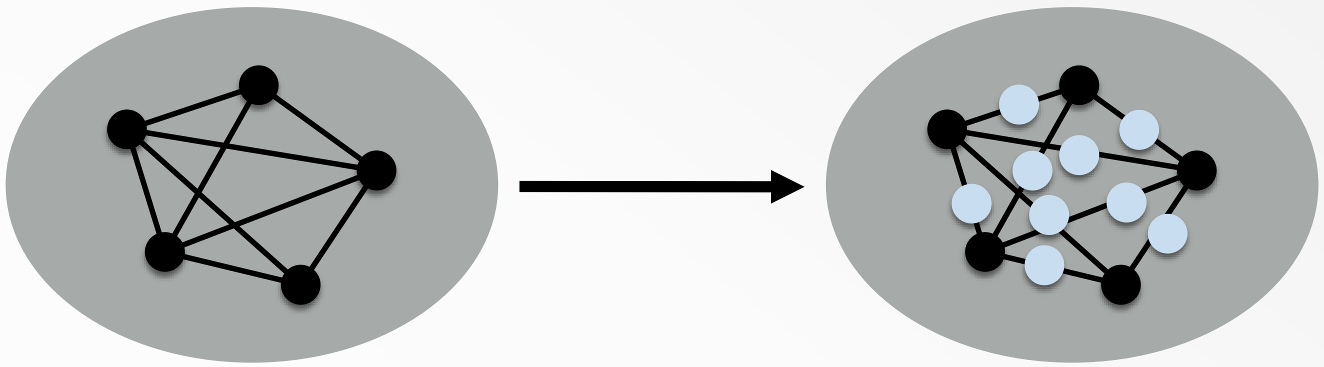
Exactly the classes with Property 1 and 2.

Topological Minors

Definition.

Let H be a graph.

1. A *subdivision* of H is a graph obtained from H by replacing edges by pairwise vertex disjoint paths.
2. H is a *topological minor* of G, $H \preccurlyeq^t G$, if a subdivision of H is isomorphic to a subgraph of G.



An r-*subdivision* of H is a graph obtained from H by replacing edges by pairwise vertex disjoint paths of length at most r.

H is an r *-shallow topological minor* of G, $H \preccurlyeq^t_r G$, if an r -subdivision of H is isomorphic to a subgraph of G.

Nowhere Dense Classes of Graphs

Definition. (Nešetril, Ossona de Mendez)

A class $\mathcal{C}$ of graphs is *nowhere dense* if for every $r \geq 1$ there is a number $f(r)$ such that $K_{f(r)} \not\preccurlyeq_r G$ for all $G \in \mathcal{C}$.

If the function $f : r \to f(r)$ is computable then we call $\mathcal{C}$ *effectively* nowhere dense.

Examples.

- Graph classes excluding a fixed minor
- Graph classes of bounded local tree width or locally excluding a minor.
- Classes of bounded expansion.

Non-Examples.

- 2-degenerate graphs.
- Bounded average degree classes.
- Classes of bounded rank or clique width.

Equivalent Definition

Nowhere dense classes of graphs have many equivalent characterisations, making it a very robust and seemingly natural concept.

Definition through density. (Nešetril, Ossona de Mendez)

A class $\mathcal{C}$ is nowhere dense if, and only if,

$$\lim_{r,n\to\infty} \max\left\{\frac{\log|E(H)|}{\log|V(H)|} : G \in \mathcal{C}, |G| = n \; and \; H \preccurlyeq_r G\right\} \leq 1\,.$$

Theorem.

For every class $\mathcal{C}$

$$\lim_{r,n\to\infty} \max\left\{\frac{\log|E(H)|}{\log|V(H)|} : G \in \mathcal{C}, |G| = n \; and \; H \preccurlyeq_r G\right\} \in \{0, 1, 2\}\,.$$

Characterisations of Nowhere Dense Graphs

Theorem.

A class $\mathcal{C}$ is nowhere dense if, and only if,

1. for every r there is a graph not contained as r-shallow topological minor
2. the edge density of every r-shallow minor is bounded by $n^{o(1)}$
3. Splitter wins the Splitter-Game
4. for every k, every graph $G \in C$ can be coloured by $n^{o(1)}$ colours so that every k colour classes induce a subgraph of tree-width $\leq k$.
5. $\mathcal{C}$ is uniformly quasi-wide
6. the weak colouring numbers on $\mathcal{C}$ are bounded for every r.
7. …

Back to Model Checking

Model Checking on Sparse Graphs

Theorem.

Every problem definable in first-order logic can be decided in time $O(n^{1+\varepsilon})$, for every $\varepsilon>0$, on any class of graphs that is nowhere dense.

Proof.

Proof combines logical aspects and algorithmic/graph structural parts.

Logical part is the most complicated part.

Structural and algorithmic tools of independent interest.

- *Efficient neighbourhood covers*
  Neighbourhood covers improving the known bounds for planar and excluded minor classes
- *Splitter game*
  A new game based characterisation of nowhere dense classes.
  Yields bounded search tree techniques.

Subgraph Isomorphism

We illustrate the main proof ideas by the following example.

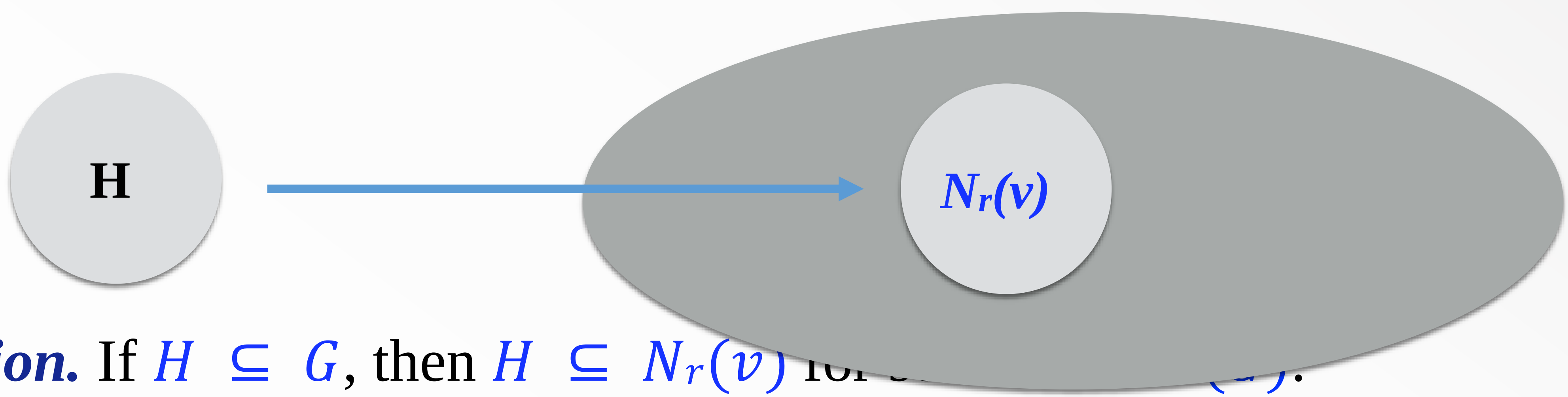
Example. Fix a connected graph H with $|V(H)| = r$.

H-Subgraph Problem.

Input: Graph G

Problem: Does G contain a subgraph isomorphic to H ?

Goal. show that it can be solved in time $O(n^{1+\varepsilon})$, for every $\varepsilon>0$, on nowhere dense graph classes.



Observation. If $H \subseteq G$, then $H \subseteq N_r(v)$ [illegible].

Hence, it suffices to look at r-neighbourhoods in G.

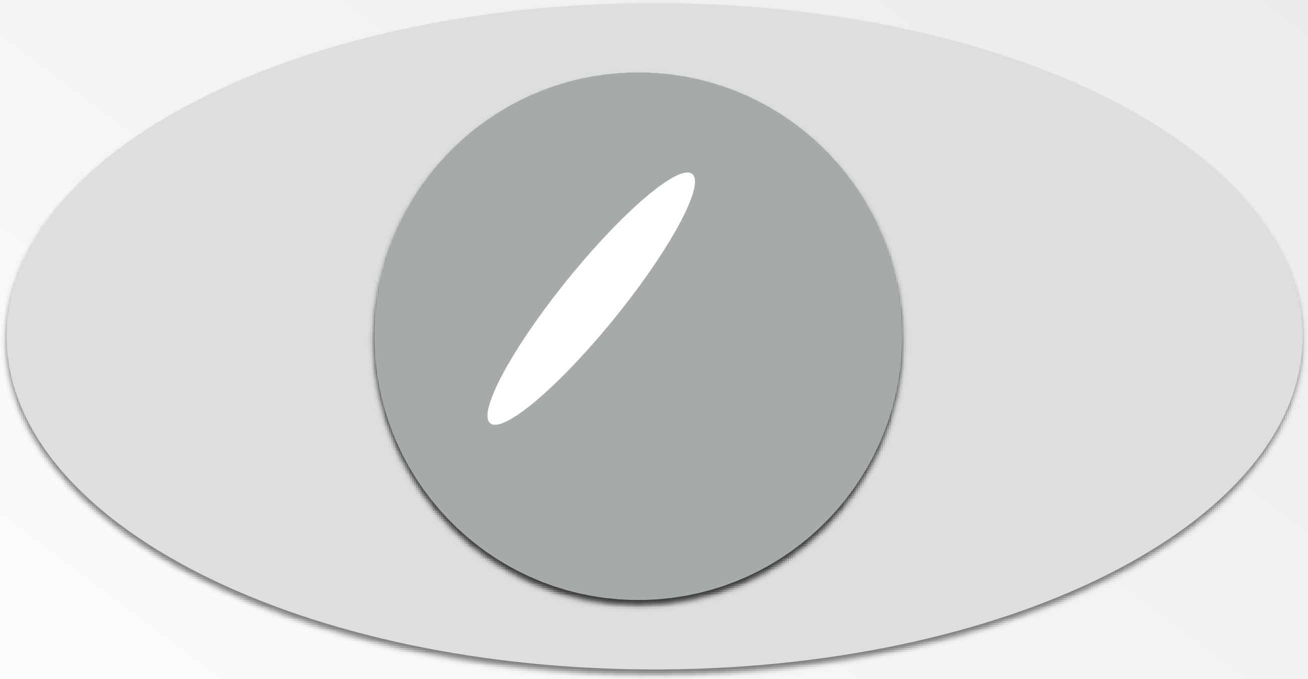
But r - neighbourhoods are not well behaved in nowhere dense graphs.

The Splitter Game

(l, d, r)-Splitter Game on G

Graph G, parameters $l, d, r>0$

Players ***Connector*** and ***Splitter***



Initialisation: $G_0 := G$

Round $i+1$*:*

1. **C** chooses $v_{i+1} \in V(G_i)$

2. **S** chooses $W_i \subseteq N_r^{G_i}(v_{i+1})$ of size at most d

We set $G_{i+1} \coloneqq G_i[N_r^{G_i}(v_{i+1}) \setminus W_{i+1}]$.

S wins if $G_{i+1} = \emptyset$. Otherwise the game continues.

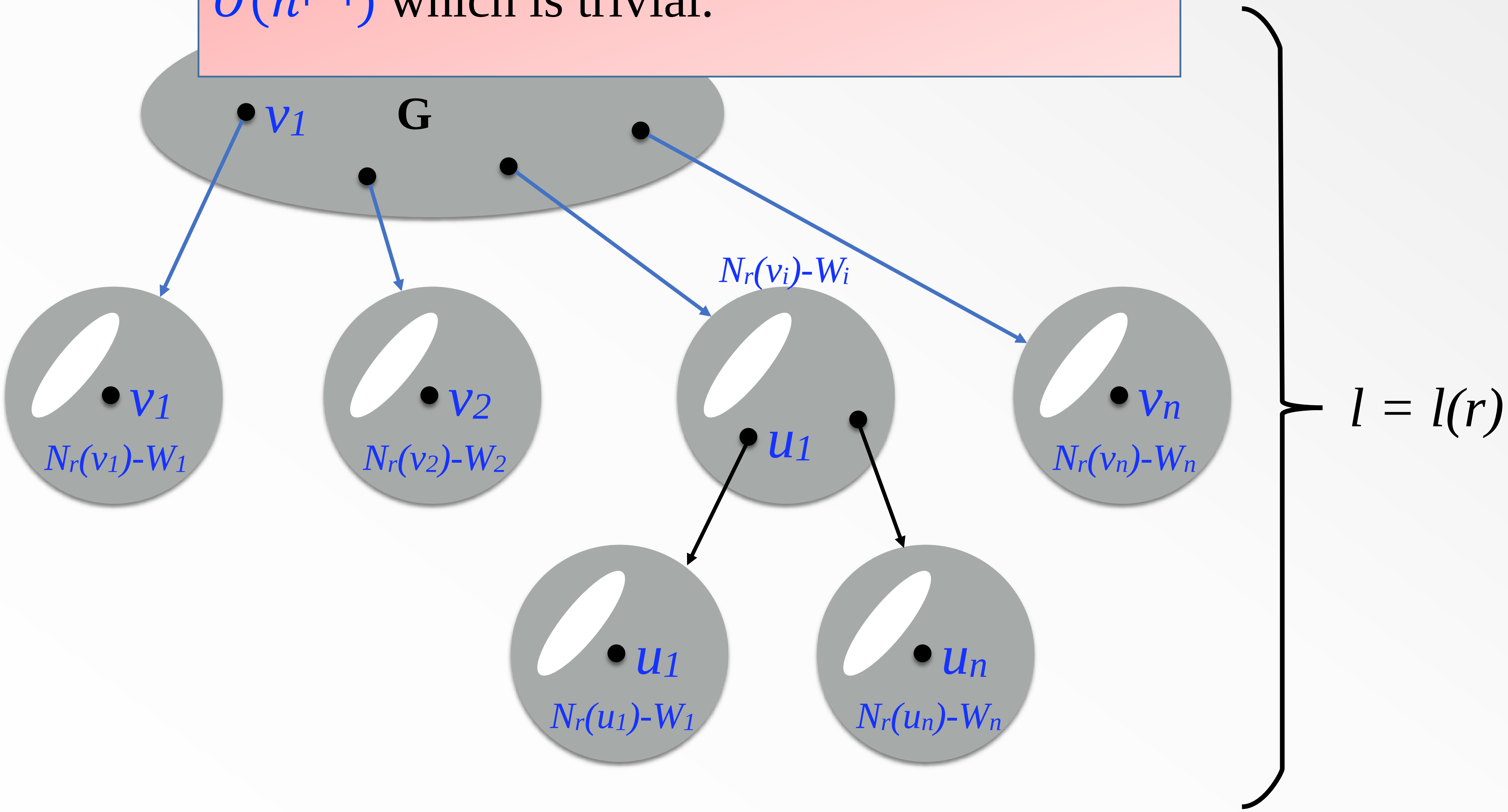
If **S** has not won after l rounds, then **C** wins.

Bounded Depth Search Trees

Theorem. (G... K... Siebertz '14)

A class C is now... d such that Splitter wins th...

The entire search tree has size $l(r) \cdot n^{O(l)}$.
This is bad as l depends on $r = |V(H)|$.
Hence, this way we can decide $H \subseteq G$ in time $O(n^{|H|})$ which is trivial.



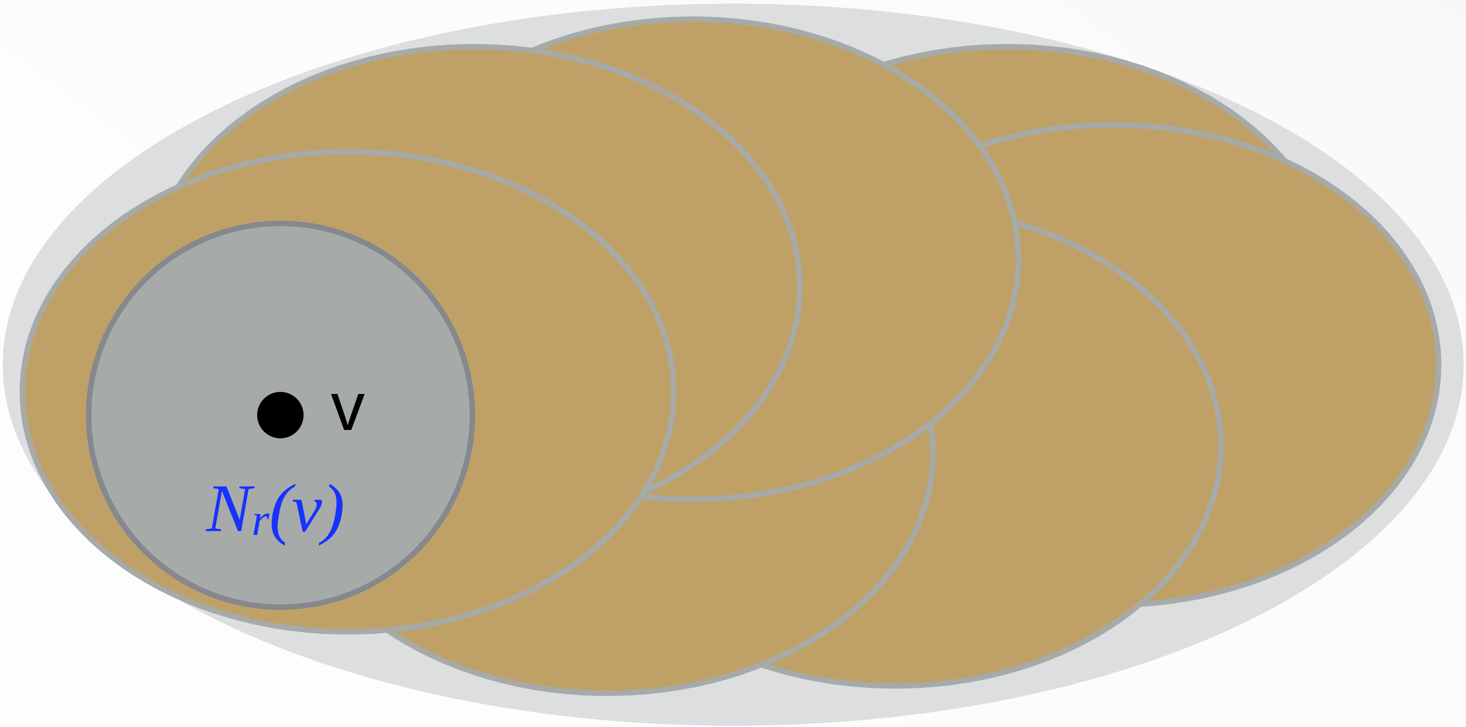
Neighbourhood Covers

Definition. Fix a radius $r > 0$.

1. An r-*neighbourhood cover* $\mathcal{N}$ of a graph G is a set of connected subgraphs of G called *clusters* such that for every $v \in V(G)$ there is some $N \in \mathcal{N}$ with $N_r(v) \subseteq N$.

2. The *radius* of $\mathcal{N}$ is the maximum radius of any of its clusters.

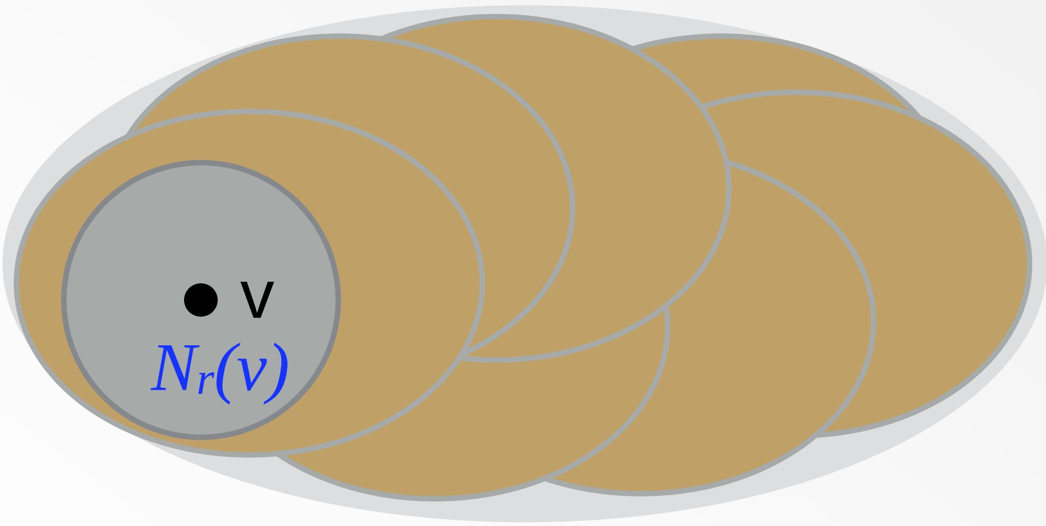
3. The *degree* of $\mathcal{N}$ is $\max |\{ N \in \mathcal{N} : v \in V(G) \}|$ over all $v \in V(G)$.



Neighbourhood Covers

Definition. Fix a radius $r > 0$.

1. An *r-neighbourhood cover* $\mathcal{N}$ of a graph G is a set of connected subgraphs of G called *clusters* such that for every $v \in V(G)$ there is some $N \in \mathcal{N}$ with $N_r(v) \subseteq N$.
2. The *radius* of $\mathcal{N}$ is the maximum radius of any of its clusters.
3. The *degree* of $\mathcal{N}$ is $\max |\{ N \in \mathcal{N} : v \in V(G) \}|$ over all $v \in V(G)$.



Neighbourhood covers are studied in

- distributed algorithms
- graph spanners

The goal usually is to mimimise the radius and the degree of the cover.

Neighbourhood Covers

Theorem.

Let $\mathcal{C}$ be a nowhere dense class of graphs.
There are neighbourhood covers of radius $2r$ and degree $O(n^{\varepsilon})$.

Precisely: For every radius $r > 0$ and $\varepsilon > 0$ there is an n_0 such that

for every $G \in \mathcal{C}$ with $|V(G)| = n > n_0$ we can compute an r-neighbourhood cover $\mathcal{N}$ of G with radius $2r$ and degree n^{ε} in time $O(n^{1+\epsilon})$.

If $\mathcal{C}$ is closed under taking subgraphs then this is *if, and only if,* i.e. constant radius and $O(n^{\varepsilon})$ *degree neighbourhood covers imply nowhere dense.*

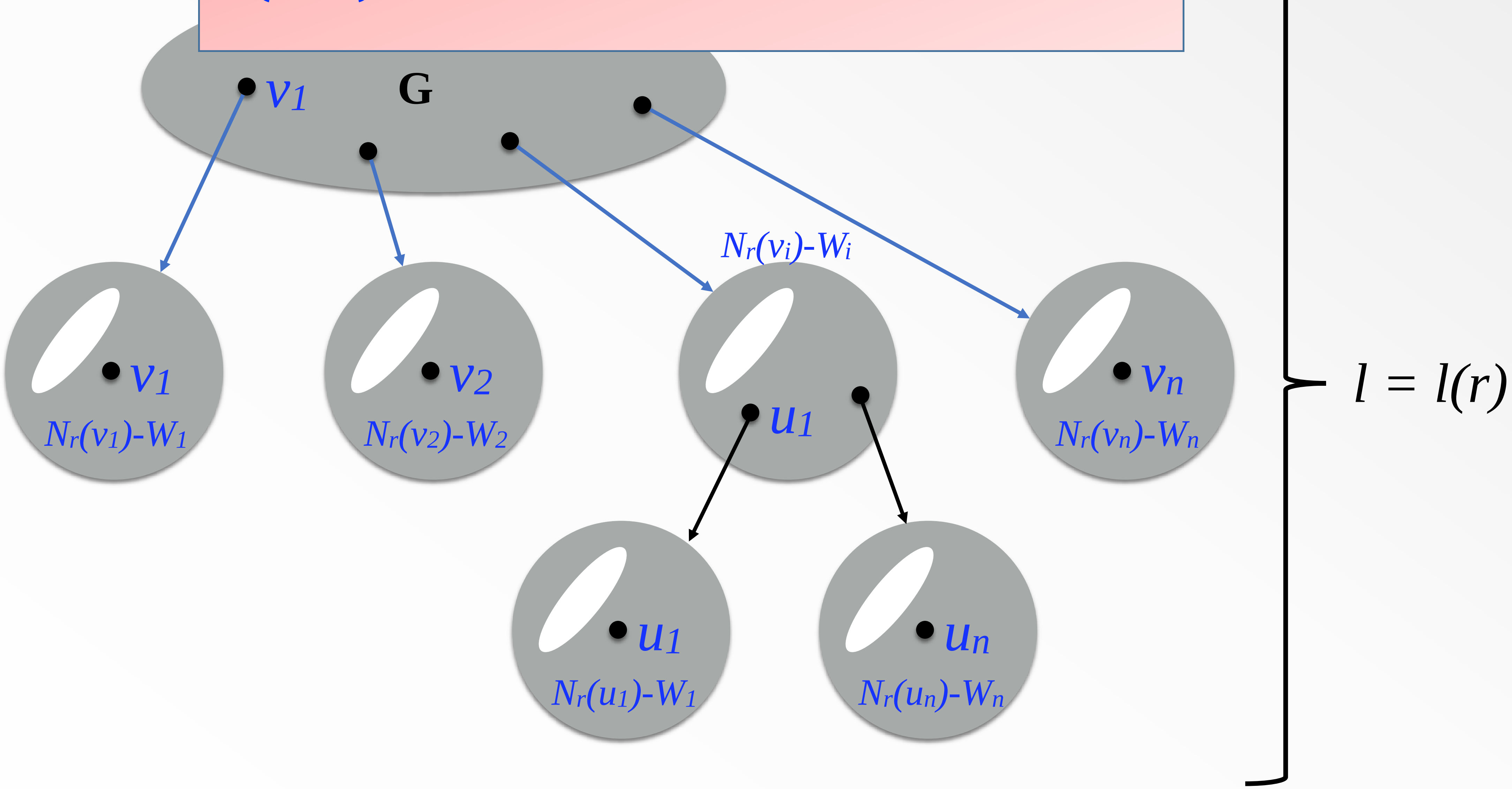
Furthermore, for classes of **bounded expansion** the degree is constant.

Bounded Depth Search Trees

Theorem. (G... K. Siebertz '14)

A class C is now ... d such that Splitter wins th...

The entire search tree has size $l(r) \cdot n^{O(l)}$.
This is bad as l depends on $r = |V(H)|$.
Hence, this way we can decide $H \subseteq G$ in time $O(n^{|H|})$ which is trivial.



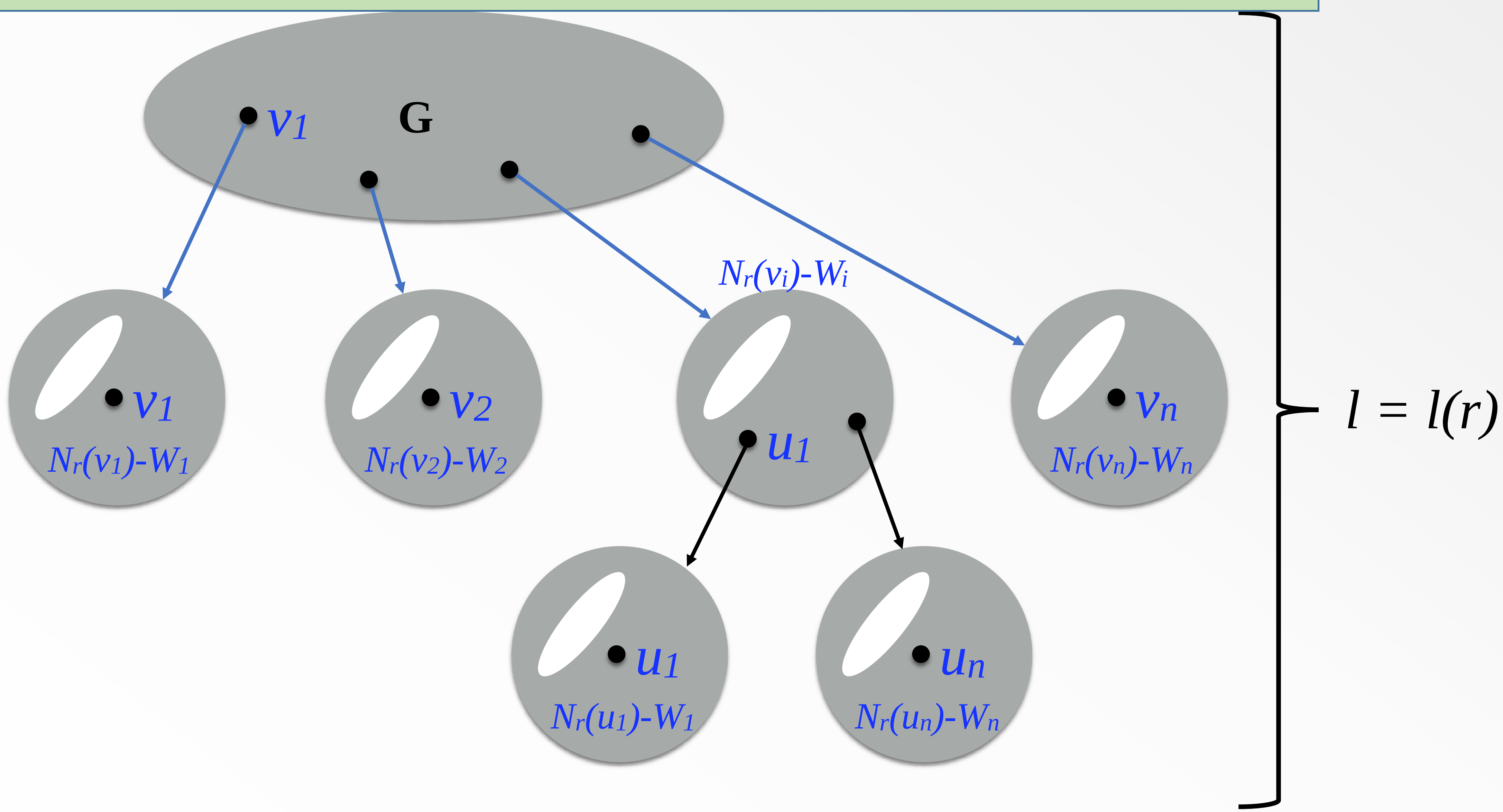
Bounded Depth Search Trees

Theore rtz '14)

A c that

Spl

If instead of $N_r(v)$ for all $v \in V(G)$ we use a neighbourhood cover at each level, the entire search tree has size $O(l(r) \cdot n^{1+\epsilon})$. In this way, we can decide $H \subseteq G$ in time $O(n^{1+\epsilon})$.



Back to Logic: Gaifman Normal Form

The connection to model checking is given by Gaifman's Normal Form.

Definition.

1. A formula $\psi^r(x)$ is r-local, if for all G and all $v \in V(G)$:

   $G \vDash \psi^r(v)$ only depends on $N_r(v)$.

2. A formula $\varphi \in FO$ with no free variables is in *Gaifman Normal Form*, if it is a Boolean combination of basic local sentences of the form

$$\exists x_1 \ldots \exists x_k \left(\bigwedge_{1 \le i < j \le k} dist(x_i, x_j) > 2r \wedge \bigwedge_{i=1}^{k} \psi^r(x_i) \right)$$

   where $\psi^r(x_i)$ is an r-local formula.

Theorem. (Gaifman '82)

Every formula is effectively equivalent to a formula in Gaifman Normal Form.

First-Order Model Checking

Problem.

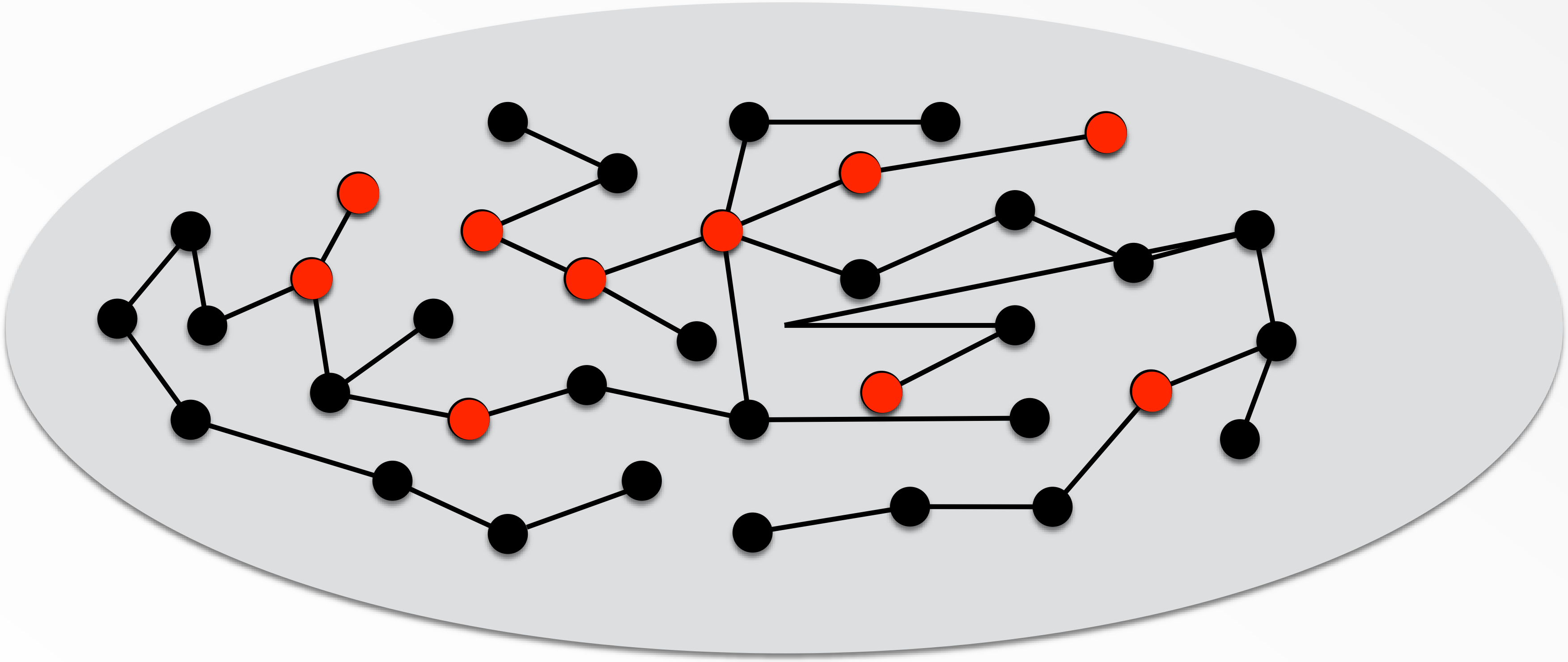
Input: $\varphi \coloneqq \exists x_1 \dots \exists x_k \left(\bigwedge_{1 \le i < j \le k} dist\left(x_i, x_j\right) > 2r \wedge \bigwedge_{i=1}^{k} \psi^r(x_i) \right)$

$G \in \mathcal{C}$

Problem: decide $G \vDash \varphi$?

Step 1. For every $v \in V(G)$ compute $N_r(v)$ and decide whether $N_r(v) \vDash \psi^r(v)$.

If *yes* then colour v red.



First-Order Model Checking

Problem.

Input: $\varphi \coloneqq \exists x_1 \ldots \exists x_k \left(\bigwedge_{1 \le i < j \le k} dist(x_i, x_j) > 2r \wedge \bigwedge_{i=1}^{k} \psi^r(x_i) \right)$

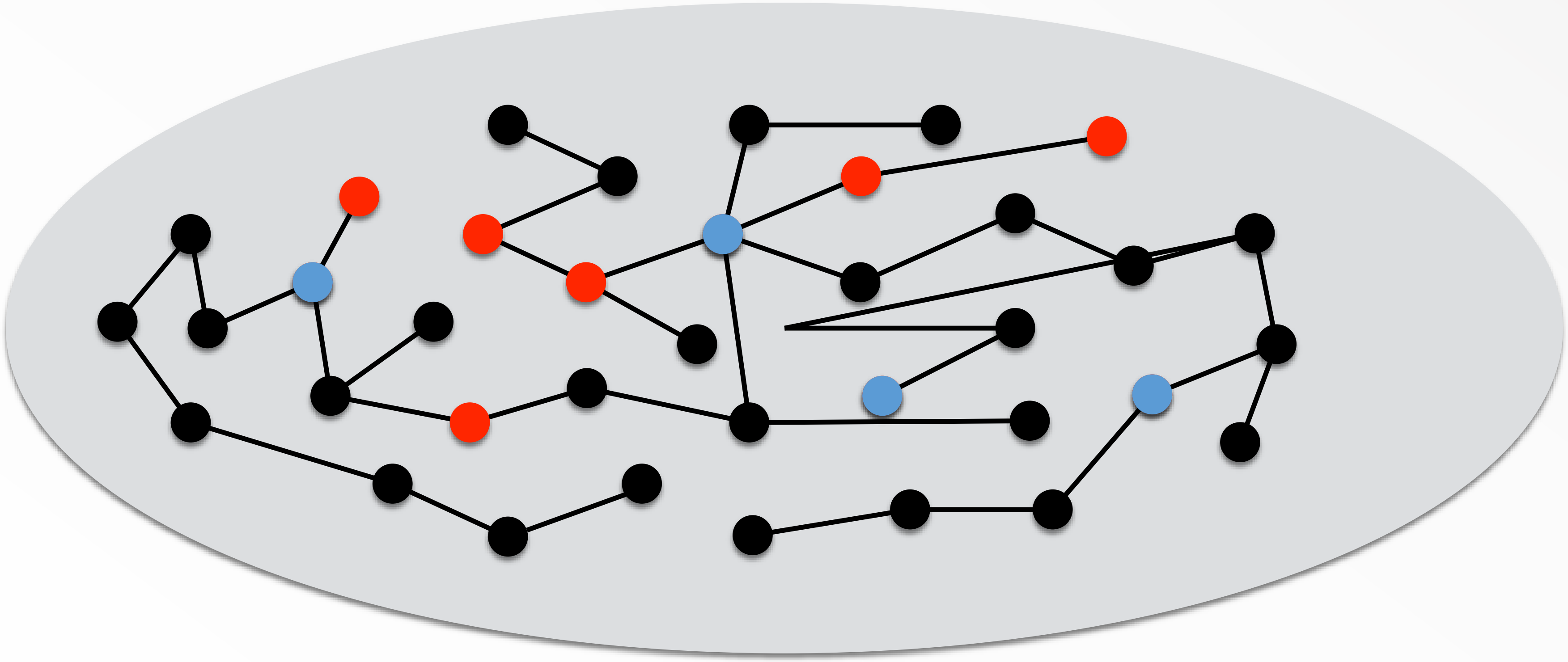
$G \in \mathcal{C}$

Problem: decide $G \vDash \varphi$?

Step 2. Find a *red* 2r-independent set of size k.

a. take a greedy approach.

b. If unsuccessful, all red nodes are in at most k different 2r neighbourhoods.



Back to Logic: Gaifman Normal Form

The connection to model checking is given by Gaifman's Normal Form.

Definition.

1. A formula $\psi^r(x)$ is r-local, if for all G and all $v \in V(G)$:

   $G \vDash \psi^r(v)$ only depends on $N_r(v)$.

2. A formula $\varphi \in FO$ with no free variables is in *Gaifman Normal Form*, if it is a Boolean combination of basic local sentences of the form

$$\exists x_1 \dots \exists x_k \left(\bigwedge_{1 \le i < j \le k} dist(x_i, x_j) > 2r \wedge \bigwedge_{i=1}^{k} \psi^r(x_i) \right)$$

   where $\psi^r(x_i)$ is an r-local formula.

Theorem. (Gaifman '82)

Every formula is effectively equivalent to a formula in Gaifman Normal Form.

Model Checking on Nowhere Dense Classes

Theorem. ***(Grohe, K., Siebertz '14)***

Every problem definable in first-order logic can be decided in time $O(n^{1+\epsilon})$, for every $\varepsilon > 0$, on any class of graphs that is nowhere dense.

Examples of first-order definable problems.

- Dominating Sets, Independent Sets, … (for fixed solution size)
- Steiner trees (for fixed solution size) etc.
- Subgraph Homo- or Isomorphism $H \to G$ (for fixed H)

Theorem. ***(K. 09, Dvorak, Kral, Thomas '11)***

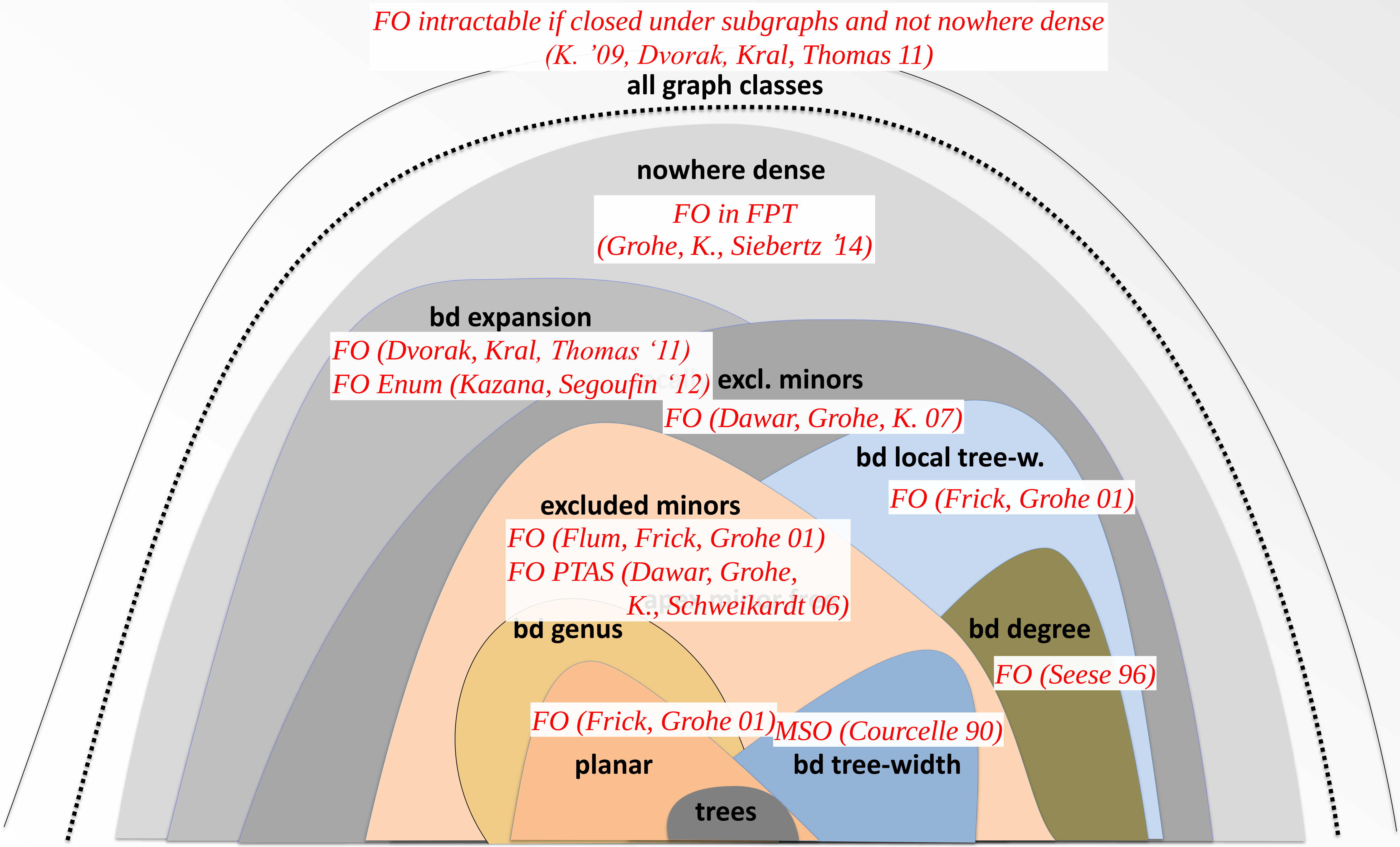
If a class C closed under subgraphs is not nowhere dense, then FO-model-checking is not fixed-parameter tractable (unless AW[$*$] = FPT).

Corollary.

Let $\mathcal{C}$ be a class of graphs closed under taking subgraphs.

$MC(FO, \mathcal{C}) \in FPT$ if, and only if, $\mathcal{C}$ is nowhere dense.

Algorithmic Meta-Theorems



Conclusion

Nowhere dense classes of graphs.

- A natural concept with many alternative characterisations
- Natural limit of algorithmic tractability (if closed under sungraphs).
- Occur naturally in practical applications.

Digraphs.

Bounded expansion can be generalised to digraphs.

We get very similar characterisations and applications.

Conclusion

Research programme.

For important logics $\mathcal{L}$ such as first-order or monadic second-order logic:

identify structural parameter $\mathcal{P}$ such that

$MC(\mathcal{C}, \mathcal{L})$ is FPT for a class $\mathcal{C}$ if, and only if, $\mathcal{C}$ has property $\mathcal{P}$.

Monadic Second-Order Logic with edge set quantification:

More or less completed.

Monadic Second-Order Logic without edge set quantification:

Tractability on graph classes of bounded rank or clique width.

No lower bound.

First-Order Logic.

Completed for classes of graphs closed under taking subgraphs.

Future Work

What if our classes are no longer sparse? And no longer closed under taking subgraphs?

Wait till tomorrow.