Evaluating Restricted First-Order Counting Properties on Nowhere Dense Classes and Beyond

Jan Dreier
TU Wien, Austria

Daniel Mock
RWTH Aachen University, Germany

Peter Rossmanith
RWTH Aachen University, Germany

Abstract

It is known that first-order logic with some counting extensions can be efficiently evaluated on graph classes with bounded expansion, where depth-r minors have constant density. More precisely, the formulas are $\exists x_1 \dots x_k \# y \, \varphi(x_1, \dots, x_k, y) > N$, where φ is an FO-formula. If φ is quantifier-free, we can extend this result to *nowhere dense* graph classes with an almost linear FPT run time. Lifting this result further to slightly more general graph classes, namely almost nowhere dense classes, where the size of depth-r clique minors is subpolynomial, is impossible unless $\mathrm{FPT} = \mathrm{W}[1]$. On the other hand, in almost nowhere dense classes we can approximate such counting formulas with a small additive error. Note those counting formulas are contained in $\mathrm{FOC}(\{>\})$ but not $\mathrm{FOC}_1(\mathbf{P})$.

In particular, it follows that partial covering problems, such as partial dominating set, have fixed parameter algorithms on nowhere dense graph classes with almost linear running time.

2012 ACM Subject Classification Theory of computation → Parameterized complexity and exact algorithms; Theory of computation → Logic; Theory of computation → Computational complexity and cryptography; Mathematics of computing → Graph theory

Keywords and phrases nowhere dense, sparsity, counting logic, dominating set, FPT

Digital Object Identifier 10.4230/LIPIcs.ESA.2023.43

Related Version *Full Version*: `https://arxiv.org/abs/2307.01832`

Funding Supported by the German Science Foundation (DFG), grant no. DFG-927/15-2.

1 Introduction

First-order logic can be used to express algorithmic problems. FO-model checking on certain classes of structures is therefore a meta-algorithm, which solves many problems at the same time. For example, the three classical problems that started the research on parameterized complexity are all FO-expressible: Vertex Cover, Independent Set, and Dominating Set [6, 7]. Dominating Set with the natural parameter – the size of the minimal dominating set – is W[2]-complete on general graphs, but fixed parameter tractable (fpt) on many special graph classes. The study of *sparsity*, initiated by Nešetřil and Ossona de Mendez, has led to the concept of *bounded expansion* and *nowhere dense* graph classes [21]. They generalize many well-known notions of sparsity, such as bounded degree, planarity, bounded genus, bounded treewidth, (topological) minor-closed, etc. and have led to quite general algorithmic results [23, 11, 4, 10]. Most notably, Grohe, Kreutzer, and Siebertz showed that FO-model checking is fpt on nowhere dense graph classes [15]. This shows, e.g., that dominating set is fpt on nowhere dense graphs, a result that was already known: Dawar and Kreutzer were able to find a specific algorithm several years earlier [5] that solves generalizations of the dominating set problem. All of them are FO-expressible, which shows how strong meta-algorithms are.

31st Annual European Symposium on Algorithms (ESA 2023).
Editors: Inge Li Gørtz, Martin Farach-Colton, Simon J. Puglisi, and Grzegorz Herman; Article No. 43; pp. 43:1–43:17
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Partial dominating set, also called t-dominating set, is another generalization of dominating set: The input is a graph G and two numbers k and t. The question is, whether G contains k vertices that dominate at least t vertices. The parameter is k, as in the classical dominating set problem. (If you choose t as the parameter – which also makes sense – the problem becomes fixed-parameter tractable even on general graphs [18].) The length of an FO-formula expressing the existence of a partial dominating depends on t, which is not bounded by any function of k and therefore all the results on first-order model checking do not help when we parameterize by k only. Golovach and Villanger showed that partial dominating set remains hard on degenerate graphs [13], while Amini, Fomin, and Saurabh have shown that partial dominating set is fixed-parameter tractable in minor-closed graph classes, which generalized earlier positive results [1]. Very recently, this was improved to graph classes with bounded-expansion, while simultaneously using only linear fpt time instead of polynomial fpt time, i.e, the running time is now only $f(k)n$ [8].

This result was achieved by another meta-theorem for the counting logic FOC($\{>\}$) on classes of bounded expansion. FOC($\{>\}$) is a fragment of the logic FOC($\mathbf{P}$), introduced by Kuske and Schweikardt in order to generalize first-order logic to counting problems [20]. FOC($\mathbf{P}$) is a very expressive counting logic and allows counting quantifiers $\#\bar{y}\varphi(\bar{x},\bar{y})$, which count for how many $\bar{y}$ the FOC($\mathbf{P}$)-formula $\varphi(\bar{x},\bar{y})$ is true. Moreover, arithmetic operations are allowed as well as all predicates in $\mathbf{P}$, which might contain comparisons, equivalence modula a number, etc. Kuske and Schweikardt showed that the FOC($\mathbf{P}$)-model checking problem is fixed parameter tracktable on graphs of bounded degree and hard on trees of bounded depth. The fragment FOC($\{>\}$) is more restrictive and allows only counting quantifiers of single variables and no arithmetic operations. The only predicate is comparison against an arbitrary number, but not between counting terms. While FOC($\{>\}$)-model checking is still hard on trees of bounded depth, there is an "approximation scheme" for FOC($\{>\}$) on classes of bounded expansion [8]: An algorithm gives either the right answer or says "mayby," but only if the formula is both almost satisfied and not satisfied. For a fragment of FOC($\{>\}$), which captures in particular the partial dominating set problem, we can compute even an exact answer to the model checking problem in linear fpt time [8]. That fragment consists of formulas of the form

$$\exists x_1 \dots \exists x_k \# y \, \varphi(y, x_1, \dots, x_k) > N, \tag{1}$$

where φ is a first-order formula and N an arbitrary number. The semantics of the *counting quantifier* $\#y\,\varphi(y, v_1, \dots, v_k)$ is the number of vertices u in G such that G satisfies $\varphi(u, v_1, \dots, v_k)$. As an example, the existence of partial dominating set can be expressed as

$$\exists x_1 \dots \exists x_k \# y \bigvee_{i=1}^{k} E(y, x_i) \vee y = x_i > t, \tag{2}$$

where k is the number of the dominating, and t the number of dominated vertices. The length of the formula only depends on k. This implies that partial dominating set can be solved in linear fpt time on classes of bounded expansion.

There is another fragment of FOC($\mathbf{P}$), which should not be confused with FOC($\{>\}$). In $\mathrm{FOC}_1(\mathbf{P})$, introduced by Grohe and Schweikardt [16], the counting terms may contain at most one free variable. They show that $\mathrm{FOC}_1(\mathbf{P})$ is fixed-parameter tractable on nowhere dense graph classes [16]. Note that formula 2 is in FOC($\{>\}$) but not in $\mathrm{FOC}_1(\mathbf{P})$ as the counting term relies on k free variables. Hence, FOC($\{>\}$) and $\mathrm{FOC}_1(\mathbf{P})$ are orthogonal in there expressiveness.

Table 1 Results of this paper (in boldface) and some related known results. *Hard* means at least W[1]-hard. *PDS like* indicates problems similar to the partial dominating set problems: All problems that can be expressed by a FOC({>}) formula of the form (1). The mentioned approximation results are quite different. Numbers are approximated either with a relative or an absolute error.

Graph class	FO-MC	$\mathrm{FOC}_1(\mathbf{P})$	FOC({>})	PDS like
bounded expansion	fpt [10]	fpt [16]	hard [8] $(1+\varepsilon)$-approx fpt [8]	fpt [8]
nowhere dense	fpt [15]	fpt [16]	hard, approx open	**fpt**[c]
almost nowhere dense	**hard**[a]	open	**hard**[a]	**hard**[a] **approx**$\pm\delta$ **fpt**[b]
general graphs	hard	hard	hard	hard

[a] Corollary 22, [b] Corollary 2, [c] Theorem 1

There has been some research about *low degree graphs*. A graph class has low degree if every (sufficiently large) graph has degree at most n^ε for every $\varepsilon > 0$. Examples are classes with bounded degree or classes with degree bounded by a polylogarithmic function. These graph classes are incomparable to nowhere dense classes. Especially, classes of low degree are not closed under subgraphs. On those classes, Grohe has shown that first-order model-checking can be solved in almost linear time [14]. Recently, Durand, Schweikardt, and Segoufin have generalized the result to query counting with constant delay and almost linear preprocessing time [9]. Vigny explores dynamic query evaluation on graph classes with low degree [24].

Almost nowhere dense is a property which subsumes both low degree and nowhere dense classes. Whereas a nowhere dense class $\mathcal{C}$ can be characterized that for every r graphs do not contain up to r times subdivided cliques of arbitrary sizes, for an almost nowhere dense class arbitrary sizes are allowed, but their growth must be bounded by subpolynomial function in the size of the graph.

Due to space limitations in this extended abstract many proofs, definitions, results, and comments can be found only in the appendix, which contains a full version of this paper. Of course, all main results are presented in this short version as well.

1.1 Our Results

In this work, we consider a fragment of FOC({>}), which we will call *PDS-like formulas*, namely formulas of the form

$$\exists x_1 \dots \exists x_k \# y\, \varphi(y, x_1, \dots, x_k) > N$$

for a quantifier-free FO-formula φ and an (arbitrarily big) number $N \in \mathbf{Z}$. This logic is strong enough to express the partial dominating set problem as formula (2) is contained in the fragment described above. Remember that this fragment and $\mathrm{FOC}_1(\mathbf{P})$ are orthogonal. Table 1 contains an overview of most of the results in this paper.

In formulas that start with existential quantifier it is natural to ask for a witness, if we can indeed fulfill the formula. For example, in the partial dominating set problem we are usually interested in actually finding the dominating set rather than verify than one exists. Often, this is not an issue as problems are self-reducible. Using self-reducibility to find a witness incurs a runtime penalty. The next theorem shows that solving the model checking problem, and finding a witness, for formulas in the form of 1 is possible.

▶ **Theorem 1.** *Let C be a nowhere dense graph class. For every $\varepsilon > 0$, every graph $G \in \mathcal{C}$ and every quantifier-free first-order formula $\varphi(y\bar{x})$ we can compute a vertex tuple $\bar{u}^*$ that maximizes $[\![\#y\,\varphi(y\bar{u}^*)]\!]^G$ in time $O(n^{1+\varepsilon})$.*

As an immediate corollary, we get that the model-checking problem for PDS-like formulas and thus, also the partial dominating set problem are solvable in almost linear fpt time on nowhere dense graph classes, where the parameter is the length of the formulas or the solution size k respectively. Moreover, our meta-algorithm does not only work for partial dominating set, but for variants such as partial total or partial connected dominating set as well.

Note that Theorem 1 does not follow from the fact that model-checking for $\mathrm{FOC}_1(\mathbf{P})$ or that query-counting for FO-logic is fixed-parameter tractable [16] as we do not count the number of solutions to a query, but the number of witnesses to some solution. Also, PDS-like formulas form a fragment orthogonal to $\mathrm{FOC}_1(\mathbf{P})$. Moreover, we were not able to prove Theorem 1 by using the result from [16] as a subroutine: formulas inside a counting quantifier are allowed to have at most one free variable and this weakens self-reducibility or similar techniques drastically.

The above theorem cannot be extended to the more general case of almost nowhere dense graph classes. It turns out that even for non-counting formulas this is not possible, as the (classical) dominating set problem becomes W[1]-hard on some almost nowhere dense graph classes. This lower bound implies as a special case that plain FO-model checking is intractable on some almost nowhere dense graph classes. As far as we are aware this does not follow directly from previously known results.

However, we can go beyond nowhere dense classes if we do not insist on an exact solution: The model-checking problem for PDS-like formulas can be approximated with an *additive* subpolynomial error in almost linear fpt time on almost nowhere dense classes of graphs. To be more precise, we get the following, slightly more general result.

▶ **Corollary 2.** *Let $\mathcal{C}$ be an almost nowhere dense class of graphs. For every $\varepsilon > 0$, every graph $G \in \mathcal{C}$ and every quantifier-free first-order formula $\varphi(y\bar{x})$, we can compute in time $O(n^{1+\varepsilon})$ a vertex tuple $\bar{u} \in V(G)^{|\bar{x}|}$ with*

$$|\max_{\bar{u}} [\![\#y\,\varphi(y\bar{u})]\!]^G - [\![\#y\,\varphi(y\bar{u}^*)]\!]^G| \leq n^{\varepsilon}.$$

Talking about characterizations of almost nowhere dense graph classes, we provide a plethora of different characterizations, similar to the ones for bounded expansion and nowhere denseness. We show that a class is almost nowhere dense classes if and only if measures like r-shallow (topological) minor, forbidden r-subdivisions and (weak) r-coloring numbers are bounded by $f(r,\varepsilon)n^{\varepsilon}$.

We also examine almost nowhere dense classes from an algorithmic point of view: Whereas it is “natural” to consider monotonicity as closure property for nowhere dense graph classes, it is similarly natural to consider closure under edge deletion for almost nowhere dense graph classes. Consider a graph class $\mathcal{C}$ which is closed under deleting edges. Then we show that the problem of finding an r times subdivided k-clique is fpt for every fixed r on $\mathcal{C}$ if and only if $\mathcal{C}$ is almost nowhere dense. In particular, for every graph class that is not almost nowhere dense, but closed under deletion of edges, there exists a number r such that finding r-subdivided k-cliques cannot be solved in fpt time under some complexity theoretic assumption, and, therefore, the FO model checking problem for formulas of the form $\exists\bar{x}\varphi(\bar{x})$ where $\varphi(\bar{x})$ is quantifier free and has predicates for adjacency and distance-r adjacency, cannot be solved either. The situation for distance-r independent set is different: Like finding an r-times subdivided clique it is fpt on almost nowhere dense graph classes, but there exists a graph class which is not almost nowhere dense and is closed under edge deletion where the problem is fpt.

1.2 Techniques

For Theorem 1, we use a novel dynamic programming technique on game trees of Splitter games. Splitter games were introduced by Grohe, Kreutzer, and Siebertz [15] to solve the first-order model-checking problem on nowhere dense classes. Together with their new concept of sparse neighborhood covers they achieved small recursion trees of constant depth.

Splitter games can be understood as a localized variation of the cops and robbers game for bounded treedepth (not to be confused with locally bounded treedepth). In contrast to [15] we apply a dynamic programming approach, similar to the ones used on bounded tree-depth decompositions. In contrast to bounded treedepth, a graph decomposes into neighborhoods of small radius instead of connected components when removing vertices according to Splitter's winning strategy. A challenge is that the resulting neighborhoods – in contrast to connected components – are not disjoint and lead to double counting for counting problems (an issue that does not occur in FO-model checking). To avoid double counting we introduce so-called cover systems specifically for the subgraph "induced" by the solution. The existence of such cover systems shows that there is a disjoint selection of small neighborhoods that cover all the vertices relevant to our counting problem. By solving a certain variation of the independent set problem, we can find such a selection and can safely combine the results of local parts of the graph as in dynamic programs for bounded tree-depth.

To achieve our second result Corollary 2, we adapt the techniques of the proof for solving the corresponding exact counting problem on classes of bounded expansion [8]: We replace $\#y\,\varphi(y\bar{x})$ by a sum of gradually simpler counting terms until they are simple enough to be easily evaluated. During this process we use transitive fraternal augmentations and a functional representation to encode necessary information into the graph, which is needed during the above simplification of counting terms. Along the way some difficult to handle literals appear in only a few number of terms. Ignoring them leads to the imprecision of our approximation. As the number of functional symbols in (almost) nowhere dense graph classes is not bounded by a constant as it is the case in classes of bounded expansion, the techniques from [8] have to adapted and extended. The main problem why their proof cannot be used directly is that the replacement of formulas leads to formulas of constant size in the case of bounded expansion, but to a non-constant size in our case. Here we use some new tricks and observe, that even though the transformed formulas can be of subpolynomial length, they can basically be replaced by many short formulas.

2 Preliminaries

2.1 Weak coloring numbers

A central concept in this paper are generalized coloring numbers, especially the weak coloring numbers introduced by Kierstead and Yang [17]. An *ordering* π of a graph G is a linear ordering of its vertex set and the set of all such orderings is denoted by $\Pi(G)$.

▶ **Definition 3** (Kierstead and Yang [17]). *A vertex $u \in V$ is* weakly r-reachable *from a vertex $v \in V$ with respect to $\pi \in \Pi(G)$ if $u \leq_\pi v$ and there exists a path P from u to v of length at most r such that $u \leq_\pi w$ for each $w \in V(P)$. The set of weakly r-reachable vertices from v with respect to π is denoted by* $\mathrm{WReach}_r[G, \pi, v]$. *Note that v is always included in this set. We write* $\mathrm{wdist}_{G,\pi}(u, v) \leq d$ *if $u \in \mathrm{WReach}_r[G, \pi, v]$ or $v \in \mathrm{WReach}_r[G, \pi, u]$.*

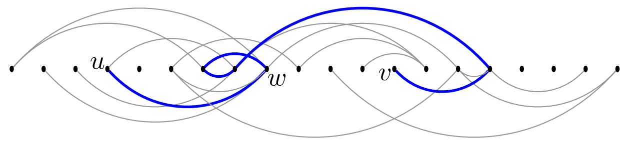

Figure 1 u is weakly 5-reachable from v by the highlighted path, but w is not weakly reachable from v.

The weak r-coloring number *of a graph G (and an ordering π) is defined as*

$$\mathrm{wcol}_r(G,\pi) := \max_{v\in V(G)} |\mathrm{WReach}_r[G,\pi,v]|$$
$$\mathrm{wcol}_r(G) := \min_{\pi\in\Pi(G(V))} \mathrm{wcol}_r(G,\pi).$$

The weak 1-coloring number of a graph is one more than its degeneracy, which is the smallest number d such that every subgraph $H \subseteq G$ has a vertex of degree at most d in H. The weak coloring number can be seen as a localized version of tree-depth, as

$$\mathrm{wcol}_1(G) \leq \mathrm{wcol}_2(G) \leq \cdots \leq \mathrm{wcol}_\infty(G) = \mathrm{td}(G) \text{ [21]}.$$

Figure 1 contains an example of weak r-reachability. Weak coloring numbers can be used to characterize nowhere dense graph classes:

▶ **Proposition 4** ([25, 22]). *A graph class $\mathcal{C}$ is nowhere dense if and only if there exists a function f such that for every $r \in \mathbf{N}$, every $\varepsilon > 0$, every graph $G \in \mathcal{C}$ satisfies* $\mathrm{wcol}_r(H) \leq f(r,\varepsilon)|H|^\varepsilon$ *for every $H \subseteq G$.*

2.2 Splitter game

We will use a game-based characterization of nowhere denseness introduced by Grohe, Kreutzer and Siebertz [15]. Given a graph G, a radius r and a number of rounds ℓ, *the (ℓ, r)-Splitter game* on G is an alternating game between two players called *Splitter* and *Connector*. The game starts on $G_0 = G$. In the ith round, the Connector chooses a vertex v_i from G_i. Then the Splitter chooses a vertex s_i from the radius-r neighborhood of v_i in G_i. The game continues on $G_{i+1} = G_i[v_i] - s_i$. Splitter wins if after ℓ rounds the graph is empty. Grohe, Kreutzer and Siebertz showed that nowhere dense graph classes can be characterized by Splitter games:

▶ **Proposition 5** ([15]). *Let $\mathcal{C}$ be a nowhere dense class of graphs. Then, for every $r > 0$, there is $\ell > 0$, such that for every $G \in \mathcal{C}$, Splitter has a strategy to win the (ℓ, r)-splitter game on G.*

Note that a winning move of Splitter in a current play can be computed in almost linear time [15, Remark 4.7].

2.3 Sparse neighborhood covers

Even though the splitter game ends after a bounded number of rounds ℓ for nowhere dense classes, the game tree, i.e. the tree spanned by all possible plays of Splitter and Connector, can still be large, e.g. in the dimensions of n^ℓ. To make the game trees small and useful for algorthmic use, Grohe, Kreutzer and Siebertz introduced sparse neighborhood covers [15]. These covers group “similar” neighborhoods into a small number cluster of bounded radius. These clusters can be used instead of the neighborhoods, reducing the size of the game tree to $O(n^{1+\varepsilon})$.

▶ **Definition 6** ([15]). *For a radius $r \in \mathbf{N}$, an* r-neighborhood cover $\mathcal{X}$ *of a graph G is a set of connected subgraphs of G called* clusters*, such that for every vertex $v \in V(G)$ there is some $X \in \mathcal{X}$ with $N_r[v] \subseteq V(X)$. The degree of v in $\mathcal{X}$ is the number of clusters that contain v and the radius of $\mathcal{X}$ is the maximal radius of a cover in $\mathcal{X}$. A class $\mathcal{C}$ admits* sparse neighborhood covers *if there exists $c \in \mathbf{N}$ and for all $r \in \mathbf{N}$ and all $\varepsilon > 0$ a number $d = d(r, \varepsilon)$ such that every graph $G \in \mathcal{C}$ admits an r-neighborhood cover of radius at most c and degree at most $d|G|^{\varepsilon}$.*

▶ **Proposition 7** ([15]). *Every nowhere dense class $\mathcal{C}$ of graphs admits a sparse neighborhood cover. For a graph $G \in \mathcal{C}$ and $r \in \mathbf{N}$ such an r-neighborhood cover can be computed in time $f(r, \varepsilon)n^{1+\varepsilon}$ for every $\varepsilon > 0$.*

Indeed, the existence of such covers is another characterization of nowhere dense classes.

▶ **Definition 8.** *For a graph G with a vertex order π, $r \in \mathbf{N}$ and a vertex $v \in V(G)$, we define $X_r[G, \pi, v]$ as $\{u \in V(G) \mid v \in \mathrm{WReach}_r[G, \pi, u]\}$. We let $\mathcal{X}_r = \{X_{2r}[G, \pi, v] \mid v \in V(G)\}$.*

From the proof of Proposition 7 it follows, that the set family $\mathcal{X}_r$ is such a sparse neighborhood cover.

2.4 Low treedepth colorings

A crucial algorithmic tool in the study of bounded expansion and nowhere dense graph classes are *low treedepth colorings*, also known as r-centered colorings.

▶ **Definition 9.** *An r-treedepth coloring of a graph G is a coloring of vertices of G such that any $r' \le r$ color classes induce a subgraph with treedepth at most r'.*

The following statement by Zhu [25] is modified such that it is constructive and holds also for a given vertex ordering π. It follows from the original proof.

▶ **Proposition 10** ([25, Proof of Thm. 2.6]). *If π is a vertex ordering of a graph G with $\mathrm{wcol}_{2^{r-2}}(G, \pi) \le m$, an r-treedepth coloring can be computed with at most m colors in time $O(mn)$.*

Graph classes of bounded expansion can be characterized by low treedepth colorings, i.e., each graph has an r-treedepth coloring with at most $f(r)$ many colors.

3 Exact Evaluation on Nowhere Dense Classes

In this section we consider the model-checking problem for formulas $\exists x_1 \dots x_k \# y\, \varphi(y\bar{x}) > N$ on nowhere dense graph classes for quantifier-free first-order formulas φ. We show that this problem can be solved in almost linear fpt time by solving its optimization variant $\max_{\bar{u} \in V(G)^{\bar{x}}} \# y\, [\![\varphi(y\bar{u})]\!]$.

3.1 Radius-r Decomposition Tree

In the following, we will introduce a new kind of decomposition, which heavily relies on the ideas from [15]. We call it the radius-r decomposition tree. For illustration, consider a tree-depth decomposition of a graph G. It has the property that after the removal of the root v in the decomposition, for each connected component C of $G - v$ there exists a child of v in the decomposition that contains C. In the radius-r decomposition tree, not every connected component is represented by a child but every radius-r neighborhood of $G - v$

instead. Another difference is that these neighborhoods are not necessarily disjoint. We will use this radius-r decomposition tree as the structure on which a dynamic program will solve $\max_{\bar{u}} [\![\#y\,\varphi(y\bar{u})]\!]^G$.

▶ **Definition 11.** *Let G be a graph. Let $r, \ell \in \mathbf{N}$ be such that splitter has a winning strategy for the ℓ-round radius-$2r$ splitter game on G. Let π be an ordering of G.*

A radius-r decomposition tree *$T_r(G, \pi, \ell)$ is a pair (T, β) where T is a tree of depth ℓ and $\beta\colon V(T) \to V(G)$. We construct it recursively. If G is empty, $T_r(G, \pi, \ell)$ is the empty tree.*

Let $s \in V(G)$ be the first move of the winning strategy of splitter for the $(\ell, 2r)$-splitter game on G. The root is a node t with $\beta(t) = s$. For every $v \in V(G)$ we append the decomposition tree $T_r(G[X_v], \pi, \ell - 1)$ where $X_v = X_{2r}[G - s, \pi, v]$.

Note that the case $\ell = 0$ while the graph is not empty, cannot happen due to the Splitter having a winning strategy.

▶ **Corollary 12.** *Let G be a graph, π a vertex ordering of G, $r, \ell \in \mathbf{N}$ and $T = T_r(G, \pi, \ell)$ a radius-r decomposition tree. Let $t \in V(T)$ be a node and T_t be the subtree of T starting at t. Then for every $u \in W := \beta(V(T_t)) \setminus \{\beta(t)\}$ there exists a child t' of t such that $N_r^{G[W]}[u] \subseteq \beta(T_{t'})$.*

▶ **Lemma 13.** *Let G be a graph, π a vertex ordering of G and $r, \ell \in \mathbf{N}$. Then, the radius-r decomposition tree $T = T_r(G, \pi, \ell)$ (Definition 11) has size $|T| \le \mathrm{wcol}_{2r}(G, \pi)^\ell n$ and depth ℓ. The construction time is linear in $|T|$.*

Proof. By construction, the depth of the tree is determined by the depth of the splitter game, which is ℓ.

Consider the root path P_t of some node $t \in V(T)$. Then $\beta(P_t) \subseteq \mathrm{WReach}_{2r}[G, \pi, \beta(t)]$. As the length of P_t is at most ℓ, $\beta(t)$ appears at most $\mathrm{WReach}_{2r}[G, \pi, \beta(t)]^\ell \le \mathrm{wcol}_{2r}(G, \pi)^\ell$ times (as a β-label of nodes) in T. Thus, $|T| \le \mathrm{wcol}_{2r}(G, \pi)^\ell n$. ◀

▶ **Corollary 14.** *Let $\mathcal{C}$ be a nowhere dense graph class. For every $r \in \mathbf{N}$ the r-decomposition tree has constant depth, almost linear size and can be computed in almost linear time.*

3.2 Cover Systems

Given a subgraph H in G with a vertex ordering π of G. A *cover system* of H in G is a family $\mathcal{Z}$ of clusters $Z_i = X_r[G, \pi, v] \in Z$ for some $r \in \mathbf{N}$ such that every connected component C of H is contained in some Z_i. A cover system is *non-overlapping* if all distinct clusters have an empty intersection.

▶ **Lemma 15.** *For every graph G with a vertex ordering π, every $D \subseteq V(G)$ of size k, there exists a cover system of $G[N[D]]$ in G of size at most k where each cluster has the same radius $r \le 2^k$.*

Proof. We start with the clusters $X_2[G, \pi, \min_\pi N[d]]$ for every $d \in D$. Call this collection $\mathcal{Z}$. Note that $\mathcal{Z}$ is already a valid cover system of $G[N[D]]$ in G. If two distinct clusters $X_r[G, \pi, z]$ and $X_r[G, \pi, z']$ from $\mathcal{Z}$ intersect, we replace both with a new cluster $X_{2r}[G, \pi, \min_\pi\{z, z'\}]$ in $\mathcal{Z}$. Every vertex or edge covered by the two old clusters stays covered in the new one. Also, if two clusters $X_r[G, \pi, z]$ and $X_{r'}[G, \pi, z']$ are of a different radius, say, $r' < r$, we replace $X_{r'}[G, \pi, z']$ with $X_r[G, \pi, z']$ to match the radii of all the clusters.

We repeat this until no intersecting clusters remain. As the number of clusters decreases with every step, the radius is at most 2^k at the end. ◀

For Theorem 1, one needs to find clusters from $\mathcal{X}_r$ which are disjoint and maximize the sum of weights of clusters. This is captured by the following definition. We can solve this problem in almost linear time on nowhere dense graph classes, by noticing that the intersection graphs of the sparse neighborhood covers $\mathcal{X}_r$ are almost nowhere dense. Then, one can use treedepth colorings and LinEMSOL.

▶ **Definition 16** (Disjoint Cluster Maximization)**.** *Given a graph, a set system $\mathcal{X}_r$ as defined in Definition 8, labelled by a function $\Lambda : \mathcal{X}_r \to 2^\Lambda$ of size k. Each combination of a cluster $X \in \mathcal{X}_r$ and label $\lambda \in \Lambda(X)$ is weighted by a function w.*

Problem: Find pairwise disjoint clusters $X_1, \dots, X_k \in \mathcal{X}_r$ such that for each label $\lambda_i \in \Lambda$ the cluster X_i is labeled λ_i and $X_1, \dots, X_k$ maximize $\sum_{i=1}^k w(X_i, \lambda_i)$ for such cluster sets.

Parameter: r, k

Let Ω be the set of weighted positive conjunctive clauses $(\mu, \omega(y\bar{x}))$, $\bar{z} \subseteq \bar{x}$ and $\bar{u} \in V(G)^{|\bar{x}|}$. With $\Omega|_{\bar{z}}$ we denote a subset of Ω with weighted clauses $(\mu, \omega(y\bar{x}))$ where every variable occurring in ω is from $\bar{z}$. We define $\Omega|_{\bar{z}}[Z, \bar{u}]$ as $\sum_{v \in Z} \sum_{(\mu,\omega) \in \Omega|_{\bar{z}}} \mu[\![\omega(v\bar{u})]\!]^G$. Note that $\Omega|_{\bar{z}}[Z, \bar{u}]$ depends only on the assignment of $\bar{z}$ and does not need the full assignment $\bar{u}$ of $\bar{x}$.

To illustrate the following lemma, consider a positive conjunctive clause $\omega(y\bar{x}\bar{z})$, sets $P, W \subseteq V(G)$ and $\bar{u} \in P^{\bar{x}}, \bar{w} \in W^{\bar{z}}$. To count the fulfilling vertices $v \in W$ of ω, i.e. $\Omega[W, \bar{u}]$, we want to reduce this task to counting on cover systems of $N[\bar{w}]$. However, as not all fulfilling vertices in W are adjacent to $\bar{w}$, we need to be more careful.

▶ **Lemma 17.** *Let G be a graph, Ω a set of weighted positive conjunctive clauses $(\mu, \omega(y\bar{x}\bar{z}))$, $P, W \subseteq V(G)$ disjoint, $\bar{u} \in P^{\bar{x}}, \bar{w} \in W^{\bar{z}}$ such that $N[\bar{w}] \subseteq P \cup W$. For every cover system $\mathcal{Z}$ of $G[N[\bar{w}]]$ in $G[W]$ it holds that*

$$\Omega[W, \bar{u}\bar{w}] = \Omega|_{y\bar{x}}[W, \bar{u}\bar{w}] + \sum_{Z \in \mathcal{Z}} (\Omega|_{y\bar{x}\bar{z}_Z}[Z, \bar{u}\bar{w}] - \Omega|_{y\bar{x}}[Z, \bar{u}])$$

where $\bar{z}_Z$ are the variables z_i from $\bar{z}$ which are assigned to a vertex in Z.

Let us consider how a solution $\bar{u}$ for $\#y\, \varphi(y\bar{x})$ interacts with a radius-r decomposition of the input graph G where r is chosen appropriately big, e.g. 2^k resulting from Lemma 15. First, we transform φ into a set of positive clauses Ω, making the application of Lemma 17 possible.

Consider some node t in T_r. When applying Lemma 17 with P as the vertices of the root path of t and W as T_t, we see that the resulting cover system $\mathcal{Z}$ corresponds to a selection of children of t in T_r, as both use the sets X_r from Definition 8. Now imagine that we know $\Omega_{y\bar{x}\bar{z}_Z}[Z, \bar{u}]$ for every $Z \in \mathcal{Z}$. Note that this number only depends on the assignment of $\bar{x}\bar{z}_z$ and not the vertices assigned outside P and Z. With Lemma 17 we can combine these numbers into $\Omega[W, \bar{u}]$ without needing to know the actual assignments of $\bar{z}_Z$ in the cover system anymore! Note that $\Omega_{y\bar{x}}[Z]$ is easily computable while only knowing $\bar{u}$ and not $\bar{w}$.

Thus, we can compute $[\![\#y\, \varphi(y\bar{u})]\!]$ bottom-up using the radius-r decomposition while only considering the vertices assigned in $\bar{u}$ which are contained in the root path of the considered vertex.

3.3 Dynamic Program

To determine $\max_{\bar{u}} \#y\, \varphi(y\bar{u})$ for a quantifier-free formula $\varphi(y\bar{x})$ we recursively compute the following information in the decomposition tree of G (bottom-up, if you will). Consider some node t of T and a partial assignment α of $\bar{x}$ to the root path $\beta(P_t)$. The interesting information is: How many vertices underneath t, i.e. in $V(G_t)$, fulfill φ under the "best" choice on completing the assignment α to vertices in $V(G_t)$. Then the answer to the problem can be read off the information for the root node.

Assume we already know this kind of information for every child t' of t. To compute this information for t, we branch how the variables x_i that are not assigned under α are distributed among the children of t. Then the table entries of these children are combined in a suitable way. We do this for every distribution among children and take the maximum of the resulting values. If a vertex corresponding to t fulfills with the assignment the formula φ, it gets counted towards the number of "fulfilling" vertices.

However, we have to take more into consideration. First, branching on the distribution of the unassigned variables x_is under α among the children of t is not fast enough, as there are around n^k possibilities for that. Instead, we branch on how the unassigned variables are partitioned. For every such partition, we formalize the optimal choice of children t_i such that they contain exactly the unassigned variables from the i-th part, as an optimization problem.

Secondly, the graphs $G_{t'}$ spanned by each child t' of t are in general not disjoint. Combining the counts of two overlapping graphs yields to double counting. We circumvent this in the above optimization problem.

Thirdly, we need to keep track of how the vertices in the root path P_t are adjacent to the variables x_i that are assigned underneath t. We cannot branch on the complete assignment as the number of those is too high.

Before we turn to the dynamic program on the decomposition tree, we consider something simpler:

Let G be a graph and $\varphi(y\bar{x})$ be quantifier-free FO formula. Consider the pair (P, W) which is a set of vertices $P = \{v_1, \dots, v_k\} \subseteq V(G)$ and a set $W \subseteq V(G)$ that is disjoint with P. We are interested in how many vertices v in $G[P \cup W]$ satisfy $\varphi(v\bar{u})$ for an optimal choice of $\bar{u} \in (P \cup W)^{|\bar{u}|}$. For this, we keep track of $M_\alpha^{(P,W)}[S]$, which is the number of fulfilling vertices $v \in W$ wrt. φ, $\hat{\alpha}$ and S, maximizing over S-completions $\hat{\alpha}$ on W.

We can "forget" a vertex v, i.e., derive the information of $(P, W \cup \{v\})$ from the information $(P \cup \{v\}, W)$ as follows: Assume the maximum number of fulfilling vertices in W is x for a given partial assignment α on $P \cup \{v\}$ and adjacency profile S on $P \cup \{v\}$. Then the number of fulfilling vertices in $W \cup \{v\}$ is $x + 1$ if v satisfies φ with the assignment α and adjacency profile S, or x otherwise. However, neither α nor S are valid assignments or adjacency profiles for P. Hence, we need to adjust these so that we can formulate this information for $(P, W \cup \{v\})$. For this, we need to remove v from α and add the neighborhood of v in P to S as S_i, for every i with $\alpha(x_i) = v$. Then, $M_\alpha^{(P\cup\{v\},W)}[S] = M_{\alpha|_P}^{(P,W\cup\{v\})}[S'](+1)$ where $\alpha|_P$ is the assignment α without v and S' is the adjacency profile as described above.

One can also combine the information of two structures (P, W_1) and (P, W_2) to get the information of $(P, W_1 \uplus W_2)$ if W_1 and W_2 are disjoint. This is also known as "merge." Consider some assignment α on P and some adjacency profile S on P. Then the number of fulfilling vertices in $U \uplus W$ wrt φ, α and S is the $\max\{M_\alpha^{P,W_1}[S_1] + M_\alpha^{P,W_2}[S_2] \mid S_1 \uplus S_2 = S\}$.

Indeed however, the algorithm does not take a quantifier-free formula φ but a set of weighted positive conjunctive clauses. Instead of just counting the fulfilled vertices, it computes the added up weight of them wrt. to the weights of the clauses.

► **Theorem 1.** *Let C be a nowhere dense graph class. For every $\varepsilon > 0$, every graph $G \in \mathcal{C}$ and every quantifier-free first-order formula $\varphi(y\bar{x})$ we can compute a vertex tuple $\bar{u}^*$ that maximizes $[\![\#y\, \varphi(y\bar{u}^*)]\!]^G$ in time $O(n^{1+\varepsilon})$.*

4 Characterizing Almost Nowhere Dense Graph Classes

In this section, we provide various characterizations of almost nowhere dense classes, i.a. via bounded depth minors and generalized coloring numbers.

▶ **Definition 18** (Almost nowhere dense). *A graph class $\mathcal{C}$ is* almost nowhere dense *if for every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that no graph $G \in \mathcal{C}$ with $|G| \geq n_0$ contains $K_{\lceil |G|^\varepsilon \rceil}$ as a depth-r minor.*

▶ **Theorem 19.** *Let $\mathcal{C}$ be a graph class. The following statements are equivalent.*

1. *$\mathcal{C}$ is almost nowhere dense.*
2. *For every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that no graph $G \in \mathcal{C}$ with $|G| \geq n_0$ contains $K_{\lceil |G|^\varepsilon \rceil}$ as a depth-r minor.*
3. *For every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that no graph $G \in \mathcal{C}$ with $|G| \geq n_0$ contains $K_{\lceil |G|^\varepsilon \rceil}$ as a depth-r topological minor.*
4. *For every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that no graph $G \in \mathcal{C}$ with $|G| \geq n_0$ contains an r'-subdivision of $K_{\lceil |G|^\varepsilon \rceil}$ as a subgraph for any $r' \leq r$.*
5. *For every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that $\mathrm{wcol}_r(G) \leq |G|^\varepsilon$ for every graph $G \in \mathcal{C}$ with $|G| \geq n_0$.*
6. *For every $r \in \mathbf{N}$, $\varepsilon > 0$ there exists n_0 such that $\mathrm{col}_r(G) \leq |G|^\varepsilon$ for every graph $G \in \mathcal{C}$ with $|G| \geq n_0$.*

The characterizations from Theorem 19 are very similar to those for nowhere dense classes. The only difference in the characterizations 1. to 4. would be the size of the forbidden cliques: for nowhere dense classes, the size would be $f(r)$ instead of $\lceil |G|^\varepsilon \rceil$. Similarly, if we would substitute "for every $G \in \mathcal{C}$" with "for every subgraph $G \subseteq H \in \mathcal{C}$" in characterizations 5 and 6 would characterize nowhere dense classes. Note that every almost nowhere dense class which is monotone, i.e. closed under taking subgraphs, is also nowhere dense.

Conversely, if a class $\mathcal{C}$ is almost nowhere dense, then its subgraph-closure $\mathcal{C}_\subseteq$ is not almost nowhere dense in general. Consider for this the class of graphs which for every $n \in \mathbf{N}$ contains independent set of size n with a clique of size $\log n$, i.e. the graph $I_n \cup K_{\log n}$. This class is almost nowhere dense but its subgraph-closure contains cliques K_n of every size n as member, and so, all graphs.

5 Approximation on Almost Nowhere Dense

In this section we consider the same problem as before, i.e., finding vertices for $\bar{x}$ that satisfy $\#y\,\varphi(\bar{x}y) > N$ but on almost nowhere dense classes of graphs. Here, we give an approximation algorithm with an additive error. For this, we use completely different techniques compared to Section 3. We first show how to reduce the corresponding model-checking problem to approximate sums over unary functions. Then we present the approximate optimization algorithm in Theorem 20.

The main result of this section is the following approximate optimization algorithm with additive error.

▶ **Theorem 20.** *There exists a computable function f such that for every graph G and every quantifier-free first-order formula $\varphi(y\bar{x})$ we can compute a vertex tuple $\bar{u}^*$ with*

$$|\max_{\bar{u}} [\![\#y\,\varphi(y\bar{u})]\!]^G - [\![\#y\,\varphi(y\bar{u}^*)]\!]^G| \leq 4^{|\varphi|}\mathrm{wcol}_2(G)^{O(|\varphi|)}$$

in time $\mathrm{wcol}_{f(|\varphi|)}(G)^{f(|\varphi|)}n$.

For the *approximate model-checking* problem with an additive error δ, similar to [8], we want an algorithm such that

1. the algorithm returns "yes" only if G satisfies the formula,
2. returns "no" only if G does not satisfy the formula,
3. returns $\perp$ only if the optimum is within δ to N.

The option $\perp$ can be seen as "I do not know" as the computed result and the desired result are so close that the difference falls into the additive error δ.

Given the approximate optimization algorithm from Theorem 20, we can easily build an approximate model-checking algorithm as described above for the formula $\exists\bar{x}\#y\,\varphi(y\bar{x}) > N$ by computing a vertex tuple $\bar{u}^*$ from the theorem. If $N - [\![\#y\,\varphi(y\bar{u}^*)]\!]^G \leq \delta$, answer $\perp$. Otherwise, answer "yes" or "no" according whether $[\![\#y\,\varphi(y\bar{u}^*)]\!]^G > N$ or not. Note that δ cannot be chosen freely as it depends on the graph (respectively, its weak coloring numbers).

The runtime of the algorithm from Theorem 20 is fpt if the weak r-coloring numbers are bounded by n^ε for $r \leq f(|\varphi|)$. This is the case for almost nowhere dense classes. This is in contrast to the results of [8] where the running time of their algorithms is bounded by $f(\mathrm{wcol}_{f(|\varphi|)})||G||$ which is fpt on classes of bounded expansion but is not fpt on nowhere dense and almost nowhere dense classes.

This gives us the following corollary.

▶ **Corollary 2.** *Let $\mathcal{C}$ be an almost nowhere dense class of graphs. For every $\varepsilon > 0$, every graph $G \in \mathcal{C}$ and every quantifier-free first-order formula $\varphi(y\bar{x})$, we can compute in time $O(n^{1+\varepsilon})$ a vertex tuple $\bar{u} \in V(G)^{|\bar{x}|}$ with*

$$|\max_{\bar{u}} [\![\#y\,\varphi(y\bar{u})]\!]^G - [\![\#y\,\varphi(y\bar{u}^*)]\!]^G| \leq n^\varepsilon.$$

6 Hardness Results

In this section, we try to see how far the above result can or cannot be extended to either a bigger class of problems or to more general graph classes. Exemplary, we examine the distance-r versions of the dominating set, independent set and clique problem. Note that in contrast to the section before, we do not consider the partial problem versions. We see that each of these problems behave differently in this context. The distance-r dominating set problem is already hard for distance 1 on some almost nowhere dense graph classes, whereas distance-r independent set and distance-r clique are both fpt on almost nowhere dense graph classes.

As for graph classes, we consider classes that are closed under removing edges because monotone graph classes are very well understood and the notions of nowhere dense and almost nowhere dense coincide on those classes. Interestingly, for graph classes closed under removing edges the distance-r clique problem is fpt for all distances r if and only if the class is almost nowhere dense (under some complexity theoretic assumptions). However, there exist graph classes which are closed under removing edges but not almost nowhere dense that allow for fpt algorithms for the distance-r independent set problem. The difference of behavior between distance-r clique and distance-r independent set is also intriguing as the FO-formulation of these problems has exactly one quantifier alternation for both.

6.1 Exact Evaluation Beyond Nowhere Dense Classes

The following lemma proves that dominating set is W[1]-hard on almost nowhere dense classes.

The class of bipartite graphs with sides L and R where L has polylogarithmic size is almost nowhere dense: A witness for this is a vertex ordering that starts with L and starts . Only the vertices from L are weakly r-reachable from any vertex. Hence, $\mathrm{wcol}_r(G) \le |L| + 1$ for each r.

▶ **Theorem 21.** *In bipartite graphs whose left side has $2k(k-1)\lceil \log(n) \rceil$ vertices and whose right side has n vertices it is W[1]-hard to decide whether there are $\binom{k}{2}$ right-side vertices dominating all left-side vertices.*

Proof. We reduce from colorful clique. Assume we have a k-partite graph G of size n consisting of parts $V_0, \ldots, V_{k-1}$ (each of a different color) and want to find a colorful clique of size k. Without loss of generality, we can assume n to be large enough that $\binom{2\lceil \log(n) \rceil}{\lceil \log(n) \rceil - 1} \ge n$. This means, we can find for each $v \in V(G)$ a unique binary encoding $\mathrm{enc}(v)$ of length $2\lceil \log(n) \rceil$ such that the first bit is set to one and in total exactly half the bits are set to one. Let $\overline{\mathrm{enc}}(v)$ be the binary complement of $\mathrm{enc}(v)$. We construct a bipartite graph H, whose left side is partitioned into cells C_{ij} for $0 \le i \ne j < k$, each of size $2\lceil \log(n) \rceil$, and whose right side will be specified soon. The vertices of each cell are ordered. When we say for a given vertex v from the right side and cell C that v *is connected to* C *according to a specified encoding*, we mean that for $1 \le l \le 2\lceil \log(n) \rceil$, v is connected to the lth vertex of C if and only if the lth bit in the encoding is set to one. For $0 \le i < k$ we define

$$\mathrm{succ}_i(j) = \begin{cases} j + 1 \bmod k & i \ne j + 1 \bmod k \\ j + 2 \bmod k & \text{otherwise.} \end{cases}$$

For all $0 \le i < j < k$ and all $u \in V_i$ and $v \in V_j$ such that $uv \in E(G)$, add a vertex $x_{u,v}$ to the right side and

- connect $x_{u,v}$ to $C_{i,j}$ according to $\mathrm{enc}(u)$,
- connect $x_{u,v}$ to $C_{i,\mathrm{succ}_i(j)}$ according to $\overline{\mathrm{enc}}(u)$,
- connect $x_{u,v}$ to $C_{j,i}$ according to $\mathrm{enc}(v)$,
- connect $x_{u,v}$ to $C_{j,\mathrm{succ}_j(i)}$ according to $\overline{\mathrm{enc}}(v)$. ◀

We can reduce the aforementioned dominating set variation to the classical dominating set problem by connecting the right side to a fresh vertex.

▶ **Corollary 22.** *There exists an almost nowhere dense graph class $\mathcal{C}$ where the dominating set problem is $W[1]$-hard and cannot be solved in time $n^{o(k)}$ assuming ETH. This implies also the hardness of the fragments PDS-like, $\mathrm{FOC}_1(\mathbf{P})$, and $\mathrm{FOC}(\{>\})$ of $\mathrm{FOC}(\mathbf{P})$ on $\mathcal{C}$.*

Note that this result does not follow from the intractability result of FO-logic on subgraph-closed somewhere dense classes, i.e. not nowhere dense classes.

6.2 Beyond Distance One

We showed that the dominating set problem is $W[1]$-hard on some almost nowhere dense graph class. However, this is not true for the distance-r clique and independent set problem.

Distance-r clique and independent set on the other hand are fpt on almost nowhere dense graph classes. Here, we use low treedepth colorings to solve existential FO formulas. With the right formulation and inclusion-exclusion this works even for distance-r independent set which cannot be expressed as a purely existential FO formula.

▶ **Theorem 23.** *There exists a computable function f such that for every graph G the distance-r clique problem can be solved in time $\mathrm{wcol}_{f(k,r)}(G)^{f(k,r)} n$.*

Proof. We can solve this problem with the help of subgraph queries where each subgraph is an $\leq r$-subdivision of a k-clique. These subgraphs have less than $k^2(r+1)$ vertices and there are at most $(r+1)^{k^2}$ of them. Subgraph queries can be done by checking an existential FO-formula using Theorem 20. ◀

▶ **Theorem 24.** *There exists a computable function f such that for each graph G the distance-r independent set problem can be solved in time* $\mathrm{wcol}_{f(k,r)}(G)^{f(k,r)}n$.

Proof sketch. We count specially designed subgraphs to solve this problem. These subgraphs encode that there are vertices $v_1, \dots v_k$ which have some distance $d(v_i, v_j)$ from each other. As the distance constraint "$d(v_i, v_j) \geq r+1$" for the distance-r independent set problem cannot be expressed this way, we use inclusion-exclusion to compute the number of such graphs. To count them, we use low treedepth colorings whose number of colors are bounded by weak coloring numbers. ◀

▶ **Corollary 25.** *For every almost nowhere dense graph class $\mathcal{C}$, every $r \in \mathbf{N}$ and every real $\varepsilon > 0$ both the distance-r clique problem and the distance-r independent set problem can be solved in time $O(n^{1+\varepsilon})$ given a graph $G \in \mathcal{C}$.*

6.3 Beyond Almost Nowhere Dense

For graph classes that are closed under removing *vertices and edges*, i.e., monotone graph classes, we know a lot already. Most importantly, FO-model checking is fpt on such classes if and only if the class is nowhere dense (unless FPT $= W[1]$) [15]. We now want to consider graph classes that are only closed under removing *edges*. Here the concept of almost nowhere dense graph classes becomes interesting.

The following observation follows directly from characterization 6 in Theorem 19. If $\mathcal{P}$ is a parameterized problem that can be solved in time $\mathrm{col}_{f(k)}(G)^{f(k)}n$ and $\mathcal{C}$ is an almost nowhere dense graph class, then $\mathcal{P}$ can be solved on $\mathcal{C}$ in almost linear fpt time $f(k,\varepsilon)n^{1+\varepsilon}$ for every $\varepsilon > 0$. We complement this by showing that the distance-r clique problem is most likely not fpt on all graph classes that are not almost nowhere dense, but closed under removing edges. Hence, under certain common complexity theoretic assumptions, if a graph class $\mathcal{C}$ is closed under removal of edges then distance-r clique is fpt on $\mathcal{C}$ iff $\mathcal{C}$ is almost nowhere dense.

▶ **Theorem 26.** *Let $\mathcal{C}$ be a graph class that is not almost nowhere dense, but closed under removing edges. Then there exists a number r, such that one cannot solve the distance-r' clique problem parameterized by solution size in fpt time on $\mathcal{C}$ for all $r' \leq r$ unless* i.o.W[1] $\subseteq$ FPT.

Similar hardness results in parameterized complexity are usually built on the hardness assumption FPT $\neq W[1]$. The complexity class *i.o.*W[1] should be read as "infinitely often in W[1]" and needs to be explained.

▶ **Definition 27.** *For a language L and an integer n let $L_n = L \cap \{0,1\}^n$. A language L is in $i.o.C$ for a complexity class C if there is some $L' \in C$ such that $L'_n = L_n$ for infinitely many input lengths n.*

Considering the infinite often variant $i.o.C$ of a complexity class C is an established technique in complexity theory (i.e., [3, 2]). To prove our result, we show that a graph class $\mathcal{C}$ that is not almost nowhere dense, contains an infinite sequence of graphs having cliques of polynomial size as bounded depth topological minors. If $\mathcal{C}$ is also closed under removal of edges then having bounded depth topological clique minors of size n implies the existence of subdivisions of arbitrary graphs H of size n as induced subgraphs. Extra care needs to be taken to make sure that all paths connecting the principal vertices should be of equal length, since otherwise a reduction would need to try out an exponential number of possible length combinations to finally find the correct subdivision of H that is contained in $\mathcal{C}$. The following corollary is a direct consequence of Theorem 19.4.

▶ **Corollary 28.** *Let $\mathcal{C}$ be some graph class that is not almost nowhere dense. Then there are r, ε and an infinite sequence of strictly ascending numbers $n_0, n_1, \ldots$ such that for all $i \in \mathbf{N}$ there is a graph $G \in \mathcal{C}$ of order at most n_i that contains an r'-subdivision of $K_{\lceil n_i^{\varepsilon} \rceil}$ as a subgraph for some $r' \leq r$.*

The consequence i.o.W[1] $\subseteq$ FTP is weaker than W[1] $\subseteq$ FPT. We could use the latter in Theorem 26 if we required a stronger precondition, i.e., that $\mathcal{C}$ has "witnesses" for input lengths $n_0, n_1, n_2, \ldots$ such that the gap between n_i and n_{i+1} is only polynomial. This approach has been used, e.g., in proving lower bounds on the running time of MSO-model checking in graph classes where the treewidth grows polylogarithmically [19, 12].

Proof of Theorem 26. Let r and ε be the constants (depending on $\mathcal{C}$) from Corollary 28. Assume that the distance-$(r+1)$ clique problem on $\mathcal{C}$ is fpt when parameterized by solution size. We will present a Turing reduction showing that the (usual) clique problem on the class of all graphs is infinitely often in FPT.

By Corollary 28 for infinitely many $n_0, n_1, \dots \in \mathbf{N}$ there exists a graph from $\mathcal{C}$ of size at most $n_i^{1/\varepsilon}$ that contains an r'-subdivision of a clique of size n_i as a subgraph for some $r' \leq r$. Let us pick one $n = n_i$. Suppose we want to decide whether a graph G with n vertices contains a clique of size k. Since $\mathcal{C}$ is closed under removal of edges, there exist $r' \leq r$, and $n \leq N \leq n^{1/\varepsilon}$ such that $\mathcal{C}$ contains a graph $H_{r',N}$ consisting of an r'-subdivision of G together with N isolated vertices. Now for all k, G contains a clique of size k iff $H_{r',N}$ contains a distance-$(r'+1)$ clique of size k. Assume for contradiction we had an algorithm that decides in time at most $f(r',k)n^c$ whether a graph in $\mathcal{C}$ of size n contains an distance-$(r'+1)$ clique for $r' \leq r$. (For graphs not in $\mathcal{C}$, the algorithm may give a wrong answer, but we can modify it to construct and test a witness of a distance-$(r'+1)$ clique on yes-instances. Hence, we can assume that the algorithm never returns "no" on yes-instances.)

The existence of such an algorithm yields us an FPT algorithm for the k-clique problem on general graphs: For all $r' \leq r$, and $n \leq N \leq n^{1/\varepsilon}$, we run this (hypothetical) fpt algorithm in parallel on $H_{r',N}$ for $f(r',k)N^c$ time steps. Then G contains a clique of size k iff for at least one value of r' and N we have $H_{r',N} \in \mathcal{C}$ and $H_{r,N}$ contains a distance-$(r'+1)$ k-clique.

As the k-clique problem is $W[1]$-hard, we get the desired result. ◀

Note that this result does not extend to the distance-r independent set problem. Consider the class of graphs where at least half of its vertices are isolated. Then the distance-r independent set problem is trivially FPT for this graph class. However, this graph class is closed under removing edges, but it is not almost nowhere dense.

References

1 Omid Amini, Fedor V. Fomin, and Saket Saurabh. Implicit branching and parameterized partial cover problems. *J. Comput. Syst. Sci.*, 77(6):1159–1171, 2011. doi:10.1016/j.jcss.2010.12.002.

2 Richard Beigel, Lance Fortnow, and Frank Stephan. Infinitely-often autoreducible sets. *SIAM Journal on Computing*, 36(3):595–608, 2006.

3 Leonard Berman. On the structure of complete sets: Almost everywhere complexity and infinitely often speedup. In *17th Annual Symposium on Foundations of Computer Science (sfcs 1976)*, pages 76–80, 1976. doi:10.1109/SFCS.1976.22.

4 Anuj Dawar, Martin Grohe, and Stephan Kreutzer. Locally excluding a minor. In *22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland, Proceedings*, pages 270–279. IEEE Computer Society, 2007. doi:10.1109/LICS.2007.31.

5 Anuj Dawar and Stephan Kreutzer. Domination problems in nowhere-dense classes. In Ravi Kannan and K. Narayan Kumar, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2009, December 15-17, 2009, IIT Kanpur, India*, volume 4 of *LIPIcs*, pages 157–168. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2009. doi:10.4230/LIPIcs.FSTTCS.2009.2315.

6 Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, 1999. doi:10.1007/978-1-4612-0515-9.

7 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

8 Jan Dreier and Peter Rossmanith. Approximate evaluation of first-order counting queries. In Dániel Marx, editor, *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms, SODA 2021, Virtual Conference, January 10 - 13, 2021*, pages 1720–1739. SIAM, 2021. doi:10.1137/1.9781611976465.104.

9 Arnaud Durand, Nicole Schweikardt, and Luc Segoufin. Enumerating answers to first-order queries over databases of low degree. *Log. Methods Comput. Sci.*, 18(2), 2022. doi:10.46298/lmcs-18(2:7)2022.

10 Zdenek Dvorák, Daniel Král', and Robin Thomas. Testing first-order properties for subclasses of sparse graphs. *J. ACM*, 60(5):36:1–36:24, 2013. doi:10.1145/2499483.

11 Markus Frick and Martin Grohe. Deciding first-order properties of locally tree-decomposable structures. *J. ACM*, 48(6):1184–1206, 2001. doi:10.1145/504794.504798.

12 Robert Ganian, Petr Hliněný, Alexander Langer, Jan Obdržálek, Peter Rossmanith, and Somnath Sikdar. Lower bounds on the complexity of MSO1 model-checking. *Journal of Computer and System Sciences*, 80(1):180–194, 2014.

13 Petr A. Golovach and Yngve Villanger. Parameterized complexity for domination problems on degenerate graphs. In Hajo Broersma, Thomas Erlebach, Tom Friedetzky, and Daniël Paulusma, editors, *Graph-Theoretic Concepts in Computer Science, 34th International Workshop, WG 2008, Durham, UK, June 30 - July 2, 2008. Revised Papers*, volume 5344 of *Lecture Notes in Computer Science*, pages 195–205, 2008. doi:10.1007/978-3-540-92248-3_18.

14 Martin Grohe. Generalized model-checking problems for first-order logic. In Afonso Ferreira and Horst Reichel, editors, *STACS 2001, 18th Annual Symposium on Theoretical Aspects of Computer Science, Dresden, Germany, February 15-17, 2001, Proceedings*, volume 2010 of *Lecture Notes in Computer Science*, pages 12–26. Springer, 2001. doi:10.1007/3-540-44693-1_2.

15 Martin Grohe, Stephan Kreutzer, and Sebastian Siebertz. Deciding first-order properties of nowhere dense graphs. *J. ACM*, 64(3):17:1–17:32, 2017. doi:10.1145/3051095.

16 Martin Grohe and Nicole Schweikardt. First-order query evaluation with cardinality conditions. In Jan Van den Bussche and Marcelo Arenas, editors, *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, Houston, TX, USA, June 10-15, 2018*, pages 253–266. ACM, 2018. doi:10.1145/3196959.3196970.

17 Henry A. Kierstead and Daqing Yang. Orderings on graphs and game coloring number. *Order*, 20(3):255–264, 2003. doi:10.1023/B:ORDE.0000026489.93166.cb.

18 Joachim Kneis, Daniel Mölle, and Peter Rossmanith. Partial vs. complete domination: t-dominating set. In Jan van Leeuwen, Giuseppe F. Italiano, Wiebe van der Hoek, Christoph Meinel, Harald Sack, and Frantisek Plasil, editors, *SOFSEM 2007: Theory and Practice of Computer Science, 33rd Conference on Current Trends in Theory and Practice of Computer Science, Harrachov, Czech Republic, January 20-26, 2007, Proceedings*, volume 4362 of *Lecture Notes in Computer Science*, pages 367–376. Springer, 2007. doi:10.1007/978-3-540-69507-3_31.

19 Stephan Kreutzer and Siamak Tazari. Lower bounds for the complexity of monadic second-order logic. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*, pages 189–198. IEEE Computer Society, 2010. doi:10.1109/LICS.2010.39.

20 Dietrich Kuske and Nicole Schweikardt. First-order logic with counting. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005133.

21 Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity - Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and combinatorics*. Springer, 2012. doi:10.1007/978-3-642-27875-4.

22 Jaroslav Nešetřil and Patrice Ossona de Mendez. On nowhere dense graphs. *European Journal of Combinatorics*, 32(4):600–617, 2011. doi:10.1016/j.ejc.2011.01.006.

23 Detlef Seese. Linear time computable problems and first-order descriptions. *Math. Struct. Comput. Sci.*, 6(6):505–526, 1996. doi:10.1017/s0960129500070079.

24 Alexandre Vigny. Dynamic query evaluation over structures with low degree. *CoRR*, abs/2010.02982, 2020. arXiv:2010.02982.

25 Xuding Zhu. Colouring graphs with bounded generalized colouring number. *Discret. Math.*, 309(18):5562–5568, 2009. doi:10.1016/j.disc.2008.03.024.