

First-Order Query Evaluation with Cardinality Conditions

Martin Grohe¹ and Nicole Schweikardt^{*2}

¹RWTH Aachen University

²Humboldt-Universität zu Berlin

July 20, 2017

Abstract

We study an extension of first-order logic that allows to express cardinality conditions in a similar way as SQL’s COUNT operator. The corresponding logic $\text{FOC}(\mathbb{P})$ was introduced by Kuske and Schweikardt [16], who showed that query evaluation for this logic is fixed-parameter tractable on classes of databases of bounded degree.

In the present paper, we first show that the fixed-parameter tractability of $\text{FOC}(\mathbb{P})$ cannot even be generalised to very simple classes of databases of unbounded degree such as unranked trees or strings with a linear order relation.

Then we identify a fragment $\text{FOC}_1(\mathbb{P})$ of $\text{FOC}(\mathbb{P})$ which is still sufficiently strong to express standard applications of SQL’s COUNT operator. Our main result shows that query evaluation for $\text{FOC}_1(\mathbb{P})$ is fixed-parameter tractable on nowhere dense classes of databases.

1 Introduction

Query evaluation is one of the most fundamental tasks of a database system. A large amount of the literature in database theory and the related field of finite or algorithmic model theory is devoted to designing efficient query evaluation algorithms and to pinpointing the exact computational complexity of the task. The query languages that have received the most attention are the conjunctive queries and the more expressive relational calculus. The latter is usually viewed as the “logical core” of SQL, and is equivalent to first-order logic FO. Here, one identifies a database schema and a relational database of that schema with a relational signature σ and a finite σ -structure \mathcal{A} .

Apart from computing the entire query result, the query evaluation tasks usually studied are *model-checking* (check if the answer $q(\mathcal{A})$ of a Boolean query q on a database \mathcal{A} is “yes”) and *counting* (compute the number $|q(\mathcal{A})|$ of tuples that belong to the result $q(\mathcal{A})$ of a non-Boolean query q on a database \mathcal{A}); the counting problem is also relevant as the basis of computing probabilities. Such a task is regarded to be tractable for a query language L on a class \mathcal{C} of databases if it can be solved in time $f(k) \cdot n^c$ for an arbitrary function f and a constant c , where k is the size of the input query $q \in L$ and n the size of the input database $\mathcal{A} \in \mathcal{C}$. The task then is called *fixed-parameter tractable* (fpt, or “in FPT”), and *fixed-parameter linear* (fpl, or “in FPL”) in case that $c = 1$.

It is known that on unrestricted databases model-checking is $W[1]$ -hard for conjunctive queries [21], and the counting problem is $\#W[1]$ -hard already for acyclic conjunctive queries [5]. This means that under reasonable complexity theoretic assumptions, both problems are unlikely to be in FPT.

^{*}Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – SCHW 837/5-1.

A long line of research has focused on identifying restricted classes of databases on which query evaluation is fixed-parameter tractable for conjunctive queries, FO, or extensions of FO. For example, model-checking and counting for FO (even, for monadic second-order logic) is in FPL on classes of bounded tree-width [3, 2]. Model-checking and counting for FO are in FPL on classes of bounded degree [22, 10], in FPL on planar graphs and in FPT on classes of bounded local tree-width [11, 10], and in FPL on classes of bounded expansion [6, 14].

Grohe, Kreutzer, and Siebertz [13] recently provided an FPT model-checking algorithm for FO on classes of databases that are effectively nowhere dense. This gives a fairly complete characterisation of the tractability frontier for FO model-checking, as it is known that under reasonable complexity theoretic assumptions, any subgraph-closed class that admits an FPT-algorithm for FO model-checking has to be nowhere dense [15, 6]. The notion of nowhere dense classes was introduced by Nešetřil and Ossona de Mendez [19] as a formalisation of classes of “sparse” graphs. The precise definition of this notion will be relevant in this paper only in Section 8; for now it should suffice to note that the notion is fairly general, subsumes all classes of databases mentioned above, and there exist nowhere dense classes that do not belong to any of those classes.

The counting problem on nowhere dense classes is known to be in FPT for purely existential FO [20], but no extension to full FO is known [23]. Here, we obtain this extension as an immediate consequence of our technical main result. We study an extension of FO that allows to express cardinality conditions in a similar way as SQL’s COUNT operator. The corresponding logic $\text{FOC}(\mathbb{P})$ was introduced by Kuske and Schweikardt [16], who showed that model-checking and counting for this logic is fixed-parameter linear on classes of databases of bounded degree. The starting point for the work presented in this paper was the question whether this result can be extended to other “well-behaved” classes of databases, such as the classes mentioned above.

Our first result is that the fixed-parameter tractability of $\text{FOC}(\mathbb{P})$ cannot even be generalised to very simple classes of databases of unbounded degree such as unranked trees or strings with a linear order relation. Then, we identify a fragment $\text{FOC}_1(\mathbb{P})$ of $\text{FOC}(\mathbb{P})$ which still extends FO and is sufficiently strong to express standard applications of SQL’s COUNT operator. Our main result shows that model-checking and counting for $\text{FOC}_1(\mathbb{P})$ is in FPT on nowhere dense classes of databases. More precisely, for any effectively nowhere dense class \mathcal{C} of databases we present an algorithm that solves the model-checking problem and the counting problem in time $f(k, \epsilon) \cdot n^{1+\epsilon}$ for a computable function f and any $\epsilon > 0$, where k is the size of the input query $q \in \text{FOC}_1(\mathbb{P})$ and n is the size of the input database $\mathcal{A} \in \mathcal{C}$. Algorithms with such performance bounds are often called *fixed-parameter almost linear*. This generalises the result of [13] from FO to $\text{FOC}_1(\mathbb{P})$ and solves not only the model-checking but also the counting problem.

Our proof proceeds as follows. First, we reduce the query evaluation problem for $\text{FOC}_1(\mathbb{P})$ to the counting problem for rather restricted FO-formulas (Section 6). Combining this with the results on FO-counting mentioned above, we immediately obtain an FPT-algorithm for $\text{FOC}_1(\mathbb{P})$ on planar graphs and classes of bounded local tree-width [10], of bounded expansion [14], and of locally bounded expansion [23]. For nowhere dense classes, though, it is not so easy to generalise the FO model-checking algorithm of [13] to solve the counting problem. For this, we generalise the notion of “rank-preserving locality” of [13] from sentences to formulas with free variables and to counting terms (Section 7), which then enables us to lift the model-checking algorithm of [13] to an algorithm for the counting problem (Section 8).

The rest of the paper is structured as follows. Section 2 provides basic notations, Section 3 recalls the definition of $\text{FOC}(\mathbb{P})$ of [16], Section 4 provides the hardness results for $\text{FOC}(\mathbb{P})$ on unranked trees and strings with a linear order, Section 5 introduces $\text{FOC}_1(\mathbb{P})$ and gives a precise formulation of our main result, and Section 9 points out directions for future work.

2 Basic notation

We write \mathbb{Z} , \mathbb{N} , and $\mathbb{N}_{\geq 1}$ for the sets of integers, non-negative integers, and positive integers, resp. For all $m, n \in \mathbb{N}$, we write $[m, n]$ for the set $\{k \in \mathbb{N} : m \leq k \leq n\}$, and we let $[m] = [1, m]$. For a k -tuple $\bar{x} = (x_1, \dots, x_k)$ we write $|\bar{x}|$ to denote its *arity* k . By $()$ we denote the empty tuple, i.e., the tuple of arity 0.

A *signature* σ is a finite set of relation symbols. Associated with every relation symbol $R \in \sigma$ is a non-negative integer $\text{ar}(R)$ called the *arity* of R . The *size* $\|\sigma\|$ of a signature σ is the sum of the arities of its relation symbols. A σ -*structure* \mathcal{A} consists of a finite non-empty set A called the *universe* of \mathcal{A} , and a relation $R^{\mathcal{A}} \subseteq A^{\text{ar}(R)}$ for each relation symbol $R \in \sigma$. Note that according to these definitions, all signatures and all structures considered in this paper are *finite*, signatures are *relational* (i.e., they do not contain constants or function symbols), and signatures may contain relation symbols of arity 0. Note that there are only two 0-ary relations over a set A , namely \emptyset and $\{()\}$.

We write $\mathcal{A} \cong \mathcal{B}$ to indicate that two σ -structures \mathcal{A} and \mathcal{B} are isomorphic. A σ -structure \mathcal{B} is the *disjoint union* of two σ -structures \mathcal{A}_1 and \mathcal{A}_2 if $B = A_1 \cup A_2$, $A_1 \cap A_2 = \emptyset$, and $R^{\mathcal{B}} = R^{\mathcal{A}_1} \cup R^{\mathcal{A}_2}$ for all $R \in \sigma$.

Let σ' be a signature with $\sigma' \supseteq \sigma$. A σ' -*expansion* of a σ -structure \mathcal{A} is a σ' -structure \mathcal{B} such that $B = A$ and $R^{\mathcal{B}} = R^{\mathcal{A}}$ for every $R \in \sigma$. Conversely, if \mathcal{B} is a σ' -expansion of \mathcal{A} , then \mathcal{A} is called the σ -*reduct* of \mathcal{B} .

A *substructure* of a σ -structure \mathcal{A} is a σ -structure \mathcal{B} with universe $B \subseteq A$ and $R^{\mathcal{B}} \subseteq R^{\mathcal{A}}$ for all $R \in \sigma$. For a σ -structure \mathcal{A} and a non-empty set $B \subseteq A$, we write $\mathcal{A}[B]$ to denote the *induced substructure* of \mathcal{A} on B , i.e., the σ -structure with universe B , where $R^{\mathcal{A}[B]} = R^{\mathcal{A}} \cap B^{\text{ar}(R)}$ for all $R \in \sigma$.

Throughout this paper, when speaking of *graphs* we mean undirected graphs. The *Gaifman graph* $G_{\mathcal{A}}$ of a σ -structure \mathcal{A} is the graph with vertex set A and an edge between two distinct vertices $a, b \in A$ iff there exists $R \in \sigma$ and a tuple $(a_1, \dots, a_{\text{ar}(R)}) \in R^{\mathcal{A}}$ such that $a, b \in \{a_1, \dots, a_{\text{ar}(R)}\}$. The structure \mathcal{A} is called *connected* if its Gaifman graph $G_{\mathcal{A}}$ is connected; the *connected components* of \mathcal{A} are the connected components of $G_{\mathcal{A}}$.

The *distance* $\text{dist}^{\mathcal{A}}(a, b)$ between two elements $a, b \in A$ is the minimal number of edges of a path from a to b in $G_{\mathcal{A}}$; if no such path exists, we let $\text{dist}^{\mathcal{A}}(a, b) := \infty$. For a tuple $\bar{a} = (a_1, \dots, a_k) \in A^k$ and an element $b \in A$ we let $\text{dist}^{\mathcal{A}}(\bar{a}, b) := \min_{i \in [k]} \text{dist}^{\mathcal{A}}(a_i, b)$. For every $r \geq 0$, the *r -ball of \bar{a} in \mathcal{A}* is the set $N_r^{\mathcal{A}}(\bar{a}) = \{b \in A : \text{dist}^{\mathcal{A}}(\bar{a}, b) \leq r\}$. The *r -neighbourhood of \bar{a} in \mathcal{A}* is defined as $\mathcal{N}_r^{\mathcal{A}}(\bar{a}) := \mathcal{A}[N_r^{\mathcal{A}}(\bar{a})]$.

Let **vars** be a fixed countably infinite set of *variables*. A σ -*interpretation* $\mathcal{I} = (\mathcal{A}, \beta)$ consists of a σ -structure \mathcal{A} and an *assignment* β in \mathcal{A} , i.e., $\beta: \text{vars} \rightarrow A$. For $k \in \mathbb{N}$, for $a_1, \dots, a_k \in A$, and for pairwise distinct $y_1, \dots, y_k \in \text{vars}$, we write $\beta_{\frac{a_1, \dots, a_k}{y_1, \dots, y_k}}$ for the assignment β' in \mathcal{A} with $\beta'(y_j) = a_j$ for all $j \in [k]$, and $\beta'(z) = \beta(z)$ for all $z \in \text{vars} \setminus \{y_1, \dots, y_k\}$. For $\mathcal{I} = (\mathcal{A}, \beta)$ we let $\mathcal{I}_{\frac{a_1, \dots, a_k}{y_1, \dots, y_k}} = (\mathcal{A}, \beta_{\frac{a_1, \dots, a_k}{y_1, \dots, y_k}})$.

The *order* of a σ -structure \mathcal{A} is $|A|$, and the *size* of \mathcal{A} is $\|\mathcal{A}\| := |A| + \sum_{R \in \sigma} |R^{\mathcal{A}}|$. For a graph G we write $V(G)$ and $E(G)$ to denote its vertex set and edge set, respectively. Sometimes, we will shortly write ij (or ji) to denote an edge $\{i, j\}$ between the vertices i and j . The *size* of G is $\|G\| := |V(G)| + |E(G)|$. Note that up to a constant factor depending on the signature, a structure has the same size as its Gaifman graph.

3 Syntax and semantics of $\text{FOC}(\mathbb{P})$

In [16], Kuske and Schweikardt introduced the following logic $\text{FOC}(\mathbb{P})$ and provided an according notion of Hanf normal form, which was utilised to obtain efficient algorithms for evaluating $\text{FOC}(\mathbb{P})$ -queries on classes of structures of bounded degree. The syntax and semantics of $\text{FOC}(\mathbb{P})$ is defined as follows (the text is taken almost verbatim from [16]).

A *numerical predicate collection* is a triple $(\mathbb{P}, \text{ar}, \llbracket \cdot \rrbracket)$ where \mathbb{P} is a countable set of *predicate names*, $\text{ar} : \mathbb{P} \rightarrow \mathbb{N}_{\geq 1}$ assigns the *arity* to every predicate name, and $\llbracket \mathbb{P} \rrbracket \subseteq \mathbb{Z}^{\text{ar}(\mathbb{P})}$ is the *semantics* of the predicate name $P \in \mathbb{P}$. Basic examples of numerical predicates are $P_{\geq 1}$, $P_{=}$, P_{\leq} , Prime with $\llbracket P_{\geq 1} \rrbracket := \mathbb{N}_{\geq 1}$, $\llbracket P_{=} \rrbracket := \{(m, m) : m \in \mathbb{Z}\}$, $\llbracket P_{\leq} \rrbracket := \{(m, n) \in \mathbb{Z}^2 : m \leq n\}$, $\llbracket \text{Prime} \rrbracket := \{n \in \mathbb{N} : n \text{ is a prime number}\}$. For the remainder of this paper let us fix an arbitrary numerical predicate collection $(\mathbb{P}, \text{ar}, \llbracket \cdot \rrbracket)$ that contains the predicate $P_{\geq 1}$.

Definition 3.1 (FOC(\mathbb{P})[σ]). Let σ be a signature. The set of *formulas* and *counting terms* for $\text{FOC}(\mathbb{P})[\sigma]$ is built according to the following rules:

- (1) $x_1 = x_2$ and $R(x_1, \dots, x_{\text{ar}(R)})$ are *formulas*, where $R \in \sigma$ and $x_1, x_2, \dots, x_{\text{ar}(R)}$ are variables¹
- (2) if φ and ψ are formulas, then so are $\neg\varphi$ and $(\varphi \vee \psi)$
- (3) if φ is a formula and $y \in \text{vars}$, then $\exists y \varphi$ is a *formula*
- (4) if $P \in \mathbb{P}$, $m = \text{ar}(P)$, and t_1, \dots, t_m are counting terms, then $P(t_1, \dots, t_m)$ is a *formula*
- (5) if φ is a formula, $k \in \mathbb{N}$, and $\bar{y} = (y_1, \dots, y_k)$ is a tuple of k pairwise distinct variables, then $\# \bar{y} . \varphi$ is a *counting term*
- (6) every integer $i \in \mathbb{Z}$ is a *counting term*
- (7) if t_1 and t_2 are counting terms, then so are $(t_1 + t_2)$ and $(t_1 \cdot t_2)$

Note that first-order logic $\text{FO}[\sigma]$ is the fragment of $\text{FOC}(\mathbb{P})[\sigma]$ built by using only the rules (1)–(3). Let $\mathcal{I} = (\mathcal{A}, \beta)$ be a σ -interpretation. For every formula or counting term ξ of $\text{FOC}(\mathbb{P})[\sigma]$, the semantics $\llbracket \xi \rrbracket^{\mathcal{I}}$ is defined as follows.

- (1) $\llbracket x_1 = x_2 \rrbracket^{\mathcal{I}} = 1$ if $a_1 = a_2$ and $\llbracket x_1 = x_2 \rrbracket^{\mathcal{I}} = 0$ otherwise; $\llbracket R(x_1, \dots, x_{\text{ar}(R)}) \rrbracket^{\mathcal{I}} = 1$ if $(a_1, \dots, a_{\text{ar}(R)}) \in R^{\mathcal{A}}$, and $\llbracket R(x_1, \dots, x_{\text{ar}(R)}) \rrbracket^{\mathcal{I}} = 0$ otherwise; where $a_j := \beta(x_j)$ for $j \in \{1, \dots, \max\{2, \text{ar}(R)\}\}$
- (2) $\llbracket \neg\varphi \rrbracket^{\mathcal{I}} = 1 - \llbracket \varphi \rrbracket^{\mathcal{I}}$ and $\llbracket (\varphi \vee \psi) \rrbracket^{\mathcal{I}} = \max\{\llbracket \varphi \rrbracket^{\mathcal{I}}, \llbracket \psi \rrbracket^{\mathcal{I}}\}$
- (3) $\llbracket \exists y \varphi \rrbracket^{\mathcal{I}} = \max\{\llbracket \varphi \rrbracket^{\mathcal{I}}_a : a \in A\}$
- (4) $\llbracket P(t_1, \dots, t_m) \rrbracket^{\mathcal{I}} = 1$ if $(\llbracket t_1 \rrbracket^{\mathcal{I}}, \dots, \llbracket t_m \rrbracket^{\mathcal{I}}) \in \llbracket P \rrbracket$, and $\llbracket P(t_1, \dots, t_m) \rrbracket^{\mathcal{I}} = 0$ otherwise
- (5) $\llbracket \# \bar{y} . \varphi \rrbracket^{\mathcal{I}} = |\{(a_1, \dots, a_k) \in A^k : \llbracket \varphi \rrbracket^{\mathcal{I}}_{a_1, \dots, a_k} = 1\}|$, where $\bar{y} = (y_1, \dots, y_k)$
- (6) $\llbracket i \rrbracket^{\mathcal{I}} = i$
- (7) $\llbracket (t_1 + t_2) \rrbracket^{\mathcal{I}} = \llbracket t_1 \rrbracket^{\mathcal{I}} + \llbracket t_2 \rrbracket^{\mathcal{I}}$, $\llbracket (t_1 \cdot t_2) \rrbracket^{\mathcal{I}} = \llbracket t_1 \rrbracket^{\mathcal{I}} \cdot \llbracket t_2 \rrbracket^{\mathcal{I}}$

By $\text{FOC}(\mathbb{P})$ we denote the union of all $\text{FOC}(\mathbb{P})[\sigma]$ for arbitrary signatures σ . An *expression* is a formula or a counting term. As usual, for a formula φ and a σ -interpretation \mathcal{I} we will often write $\mathcal{I} \models \varphi$ to indicate that $\llbracket \varphi \rrbracket^{\mathcal{I}} = 1$. Accordingly, $\mathcal{I} \not\models \varphi$ indicates that $\llbracket \varphi \rrbracket^{\mathcal{I}} = 0$. If s and t are counting terms, then we write $s - t$ for the counting term $(s + ((-1) \cdot t))$.

Example 3.2. The following $\text{FOC}(\mathbb{P})$ -formula expresses that the sum of the numbers of nodes and edges of a directed graph is a prime:

$$\text{Prime} \left(\left(\#(x).x=x + \#(x, y).E(x, y) \right) \right).$$

The counting term $t := \#(z).E(y, z)$ denotes the out-degree of y .

¹in particular, if $\text{ar}(R) = 0$, then $R()$ is a formula

The $\text{FOC}(\mathbb{P})$ -formula $P_{\geq 1}(t)$ expresses that the out-degree of y is ≥ 1 . For better readability of such formulas we will often write “ $t \geq 1$ ” instead of “ $P_{\geq 1}(t)$ ”.

The $\text{FOC}(\mathbb{P})$ -formula

$$\exists x \text{ Prime } (\#(y).P_{=}(\#(z).E(x, z), \#(z).E(y, z)))$$

expresses that there is some number d (represented by a node x of out-degree d) such that the number of nodes of out-degree d is a prime.

The construct $\exists z$ binds the variable $z \in \text{vars}$, and the construct $\#\bar{y}$ in a counting term binds the variables from the tuple \bar{y} ; all other occurrences of variables are free. Formally, the set $\text{free}(\xi)$ of *free variables* of an $\text{FOC}(\mathbb{P})$ -expression ξ is defined inductively as follows:

- (1) $\text{free}(x_1=x_2) = \{x_1, x_2\}$ and $\text{free}(R(x_1, \dots, x_{\text{ar}(R)})) = \{x_1, \dots, x_{\text{ar}(R)}\}$
- (2) $\text{free}(\neg\varphi) = \text{free}(\varphi)$ and $\text{free}((\varphi \vee \psi)) = \text{free}(\varphi) \cup \text{free}(\psi)$
- (3) $\text{free}(\exists y \varphi) = \text{free}(\varphi) \setminus \{y\}$
- (4) $\text{free}(P(t_1, \dots, t_m)) = \text{free}(t_1) \cup \dots \cup \text{free}(t_m)$
- (5) $\text{free}(\#(y_1, \dots, y_k).\varphi) = \text{free}(\varphi) \setminus \{y_1, \dots, y_k\}$
- (6) $\text{free}(i) = \emptyset$ for $i \in \mathbb{Z}$
- (7) $\text{free}((t_1 + t_2)) = \text{free}((t_1 \cdot t_2)) = \text{free}(t_1) \cup \text{free}(t_2)$

We will often write $\xi(\bar{z})$, for $\bar{z} = (z_1, \dots, z_n)$ with $n \geq 0$, to indicate that at most the variables from $\{z_1, \dots, z_n\}$ are free in the expression ξ . A *sentence* is a formula without free variables, a *ground term* is a counting term without free variables.

Consider an $\text{FOC}(\mathbb{P})[\sigma]$ -counting term $t(\bar{x})$, for $\bar{x} = (x_1, \dots, x_m)$. For a σ -structure \mathcal{A} and a tuple $\bar{a} = (a_1, \dots, a_m) \in A^m$, we write $t^{(\mathcal{A}, \bar{a})}$ or $t^{\mathcal{A}}[\bar{a}]$ for the integer $\llbracket t \rrbracket^{(\mathcal{A}, \bar{a})}$, where β is an assignment in \mathcal{A} with $\beta(x_j) = a_j$ for all $j \in [m]$. For an $\text{FOC}(\mathbb{P})[\sigma]$ -formula $\varphi(\bar{x})$ we write $(\mathcal{A}, \bar{a}) \models \varphi$ or $\mathcal{A} \models \varphi[\bar{a}]$ to indicate that $\llbracket \varphi \rrbracket^{(\mathcal{A}, \bar{a})} = 1$. In case that $m = 0$ (i.e., φ is a sentence and t is a ground term), we simply write $t^{\mathcal{A}}$ instead of $t^{\mathcal{A}}[\bar{a}]$, and we write $\mathcal{A} \models \varphi$ instead of $\mathcal{A} \models \varphi[\bar{a}]$.

Two formulas or two counting terms ξ and ξ' are *equivalent* (for short, $\xi \equiv \xi'$), if $\llbracket \xi \rrbracket^{\mathcal{I}} = \llbracket \xi' \rrbracket^{\mathcal{I}}$ for every σ -interpretation \mathcal{I} . The size $\|\xi\|$ of an expression is its length when viewed as a word over the alphabet $\sigma \cup \text{vars} \cup \mathbb{P} \cup \{, \} \cup \{=, \exists, \neg, \vee, (,)\} \cup \{\#, .\}$.

4 The hardness of evaluating $\text{FOC}(\mathbb{P})$ -queries

In [16] it was shown that on classes of structures of bounded degree, $\text{FOC}(\mathbb{P})$ -query evaluation is fixed-parameter linear (when using oracles for evaluating the numerical predicates in \mathbb{P}). In this section, we shall prove that there is no hope of extending this result to even very simple classes of structures of unbounded degree such as trees and words: on these classes, the $\text{FOC}(\mathbb{P})$ evaluation problem is as hard as the evaluation problem for first-order logic on arbitrary graphs. The latter is known to be PSPACE-complete [24] and, in the world of parameterised complexity theory, complete for the class $\text{AW}[*]$ [4] (also see [9]). The hardness results hold for all \mathbb{P} that contain the “equality predicate” $P_{=}$ or the “positivity predicate” $P_{\geq 1}$. The $\text{AW}[*]$ -hardness is the more relevant result for us here.² It shows that the evaluation problem is unlikely to have an algorithm running in time $f(k)n^c$ for an arbitrary function f and constant c , where k is the size of the input formula and n the size of the input structure.

²PSPACE-completeness already holds over a fixed structure with two elements.

To state our result formally, we focus on the model-checking problem, that is, the query evaluation problem for sentences. The *model-checking problem* for a logic L on a class \mathcal{C} of structures is the problem of deciding whether a given structure $\mathcal{A} \in \mathcal{C}$ satisfies a given L -sentence φ . A *polynomial fpt-reduction* between two such problems is a polynomial time many-one reduction that, given an instance \mathcal{A}, φ of the first model-checking problem, computes an instance \mathcal{A}', φ' of the second model-checking problem such that $\|\mathcal{A}'\|$ is polynomially bounded in $\|\mathcal{A}\|$ and $\|\varphi'\|$ is polynomially bounded in $\|\varphi\|$.

Theorem 4.1. *There is a polynomial fpt-reduction from the model-checking problem for FO on the class of all graphs to the model-checking problem for $\text{FOC}(\{\text{P}_=\})$ on the class of all trees.*

Proof. Let G be a graph, and let φ be an FO-sentence in the signature of graphs (consisting of a single binary relation symbol E). W.l.o.g. we assume that $V(G) = [n]$ for some $n \geq 1$. We shall define a tree T_G and an $\text{FOC}(\{\text{P}_=\})$ -sentence $\hat{\varphi}$ such that G satisfies φ if and only if T_G satisfies $\hat{\varphi}$. We construct the tree T_G as follows. The vertex set $V(T_G)$ consists of

- a “root” vertex r
- a vertex $a(i)$ for every $i \in [n]$
- vertices $b_j(i)$ and $c_j(i)$ for every $i \in [n]$ and $j \in [i+1]$
- a vertex $d(i, j)$ for every $i \in [n]$ and every neighbour j of i in G
- vertices $e_k(i, j)$ for every $i \in [n]$, every neighbour j of i in G , and every $k \in [j+1]$.

The edge set of T_G consists of

- edges $ra(i)$ for all $i \in [n]$
- edges $a(i)b_j(i)$ and $b_j(i)c_j(i)$ for all $i \in [n]$ and $j \in [i+1]$
- edges $a(i)d(i, j)$ and $d(i, j)e_k(i, j)$ for all $i \in [n]$, all neighbours j of i in G , and all $k \in [j+1]$.

Then, T_G is a tree (of height 3) that can be computed from G in quadratic time.

To define $\hat{\varphi}$, we need auxiliary formulas $\varphi_a(x), \varphi_b(x), \dots, \varphi_e(x)$ defining the sets of a, b, \dots, e -vertices, respectively. We start from the observations that the c -vertices $c_j(i)$ are the precisely those vertices of degree 1 whose unique neighbour has degree 2. The b vertices are the neighbours of the c -vertices, and the a -vertices are the neighbours of the b -vertices that are not c -vertices. The root vertex is the only vertex adjacent to all a vertices. The e -vertices are the vertices of degree 1 that are not c -vertices, and the d -vertices are the neighbours of the e -vertices.

Note that the vertices of G are in one-to-one correspondence to the a -vertices of T_G : vertex i corresponds to the unique a -vertex with exactly $(i+1)$ b -neighbours. To express that there is an edge between a -vertices x, x' , we say that x has a d -neighbour y such that the number of e -neighbours of y equals the number of b -neighbours of x' . This is precisely what the following $\text{FOC}(\{\text{P}_=\})$ -formulas says:

$$\psi_E(x, x') := \exists y \left(E(x, y) \wedge \text{P}_= \left(\#z. (E(y, z) \wedge \psi_e(z)), \#z. (E(x', z) \wedge \psi_b(z)) \right) \right).$$

Now we let $\hat{\varphi}$ be the formula obtained from φ by replacing each atom $E(x, x')$ by $\psi_E(x, x')$ and by relativizing all quantifiers to a -vertices, that is, replacing subformulas $\exists x \psi$ by $\exists x (\psi_a(x) \wedge \psi)$. Clearly, $\hat{\varphi}$ can be computed from φ in polynomial time. Moreover, it should be clear from the construction that G satisfies φ if and only if T_G satisfies $\hat{\varphi}$. \square

Corollary 4.2. *The parameterised model-checking problem for $\text{FOC}(\{\text{P}_=\})$ on the class of all trees is $\text{AW}[*]$ -complete.*

We encode strings over the alphabet Σ as structures \mathcal{A} of signature $\sigma := \{\leq\} \cup \{P_a : a \in \Sigma\}$, where the binary relation $\leq^{\mathcal{A}}$ is a linear order of A , and the unary relation $P_a^{\mathcal{A}}$ consists of the positions of all a s in the string, for each symbol $a \in \Sigma$.

Theorem 4.3. *There is a polynomial fpt-reduction from the model-checking problem for FO on the class of all graphs to the model-checking for $\text{FOC}(\{P_{=}\})$ on the class of all strings of alphabet $\Sigma := \{a, b, c\}$.*

Proof. We use a similar idea as in the proof of Theorem 4.1. Given a graph G with vertex set $[n]$ and an FO-sentence φ , we construct a string S_G and an $\text{FOC}(\{P_{=}\})$ -sentence $\hat{\varphi}$ such that G satisfies φ if and only if S_G satisfies $\hat{\varphi}$.

This time, we use substrings (instead of subtrees) to represent the vertices of G . For a vertex i with neighbours $\{j_1, \dots, j_m\}$ in G , we let s_i be the string

$$ac^i bc^{j_1} bc^{j_2} \dots bc^{j_m}.$$

Then we let S_G be the concatenation of the s_i for all $i \in [n]$. It is easy to complete the proof along the lines of the proof of Theorem 4.1. \square

Corollary 4.4. *The parameterised model-checking problem for $\text{FOC}(\{P_{=}\})$ on the class of all strings (of alphabet $\Sigma = \{a, b, c\}$) is $\text{AW}[*]$ -complete.*

Remark 4.5. Note that we can express “ $P_{=}(t_1, t_2)$ ” via “ $\neg P_{\geq 1}(t_1 - t_2) \wedge \neg P_{\geq 1}(t_2 - t_1)$ ”. Therefore, the results of Theorem 4.1, Corollary 4.2, Theorem 4.3, and Corollary 4.4 also hold for the logics $\text{FOC}(\{P_{\geq 1}\})$.

5 The fragment $\text{FOC}_1(\mathbb{P})$ of $\text{FOC}(\mathbb{P})$

In this section, we define a fragment of $\text{FOC}(\mathbb{P})$ called $\text{FOC}_1(\mathbb{P})$. This logic is an extension of FO that allows to formulate cardinality conditions concerning terms that have at most one free variable (hence the subscript 1 in “ FOC_1 ”). The logic $\text{FOC}_1(\mathbb{P})$ is designed in such a way that it, although being relatively expressive, still allows for efficient query evaluation algorithms on well-behaved classes of structures. This paper’s main result shows that $\text{FOC}_1(\mathbb{P})$ -query evaluation is fixed-parameter tractable on nowhere dense classes of structures.

Definition 5.1 ($\text{FOC}_1(\mathbb{P})[\sigma]$). Let σ be a signature.

The set of *formulas* and *counting terms* of $\text{FOC}_1(\mathbb{P})[\sigma]$ is built according to the rules (1)–(3) and (5)–(7) and the following restricted version of rule (4) of Definition 3.1:

(4') if $P \in \mathbb{P}$, $m = \text{ar}(P)$, and t_1, \dots, t_m are counting terms *such that* $|\text{free}(t_1) \cup \dots \cup \text{free}(t_m)| \leq 1$, then $P(t_1, \dots, t_m)$ is a *formula*

The first two formulas of Example 3.2 are in $\text{FOC}_1(\mathbb{P})$; the last formula of Example 3.2 and the formula $\psi_E(x, x')$ from the proof of Theorem 4.1 are not. Based on the logic $\text{FOC}_1(\mathbb{P})$, we define the following query language.

Definition 5.2 ($\text{FOC}_1(\mathbb{P})$ -queries). Let σ be a signature. An $\text{FOC}_1(\mathbb{P})[\sigma]$ -query is of the form

$$\{ (x_1, \dots, x_k, t_1, \dots, t_\ell) : \varphi \} \quad (*)$$

where $k, \ell \geq 0$, x_1, \dots, x_k are pairwise distinct variables, t_1, \dots, t_ℓ are $\text{FOC}_1(\mathbb{P})[\sigma]$ -counting terms with $\text{free}(t_i) \subseteq \{x_1, \dots, x_k\}$ for each $i \in [\ell]$, and φ is an $\text{FOC}_1(\mathbb{P})[\sigma]$ -formula with $\text{free}(\varphi) = \{x_1, \dots, x_k\}$.

When evaluated in a σ -structure \mathcal{A} , a query q of the form $(*)$ returns the result $q(\mathcal{A}) := \llbracket q \rrbracket^{\mathcal{A}} :=$

$$\left\{ (a_1, \dots, a_k, n_1, \dots, n_\ell) : \mathcal{A} \models \varphi[a_1, \dots, a_k] \text{ and } n_j = t_j^{\mathcal{A}}[a_1, \dots, a_k] \text{ for each } j \in [\ell] \right\}.$$

Let us demonstrate that the usual examples for uses of the COUNT operation in SQL can be expressed in this query language.

Example 5.3. In this example we consider $\text{FOC}_1(\mathbb{P})$ -queries where \mathbb{P} is empty, and deal with the database schema consisting of relations `Customer`(`Id`, `FirstName`, `LastName`, `City`, `Country`, `Phone`) and `Order`(`Id`, `OrderDate`, `OrderNumber`, `CustomerId`, `TotalAmount`).³

To list the number of customers in each country, one can use the SQL-statement

```
SELECT Country, COUNT(Id)
FROM Customer
GROUP BY Country
```

or the $\text{FOC}_1(\mathbb{P})$ -query $\{(x_{co}, t(x_{co})) : \varphi(x_{co})\}$ with $\varphi(x_{co}) := x_{co}=x_{co}$ and $t(x_{co}) := \#(x_{id}).\psi$ with $\psi :=$

$$\exists x_{fi} \exists x_{la} \exists x_{ci} \exists x_{ph} \text{Customer}(x_{id}, x_{fi}, x_{la}, x_{ci}, x_{co}, x_{ph}).$$

To return the total number of customers and the total number of orders stored in the database, we can use the SQL-statement⁴

```
SELECT(
  SELECT COUNT(*)
  FROM Customer
) AS No_Of_Customers,
(
  SELECT COUNT(*)
  FROM Order
) AS No_Of_Orders
```

or, equivalently, the $\text{FOC}_1(\mathbb{P})$ -query $\{(t_c, t_o) : \varphi\}$ for

$$\begin{aligned} t_c &:= \#(\bar{x}).\text{Customer}(\bar{x}) \\ t_o &:= \#(\bar{y}).\text{Order}(\bar{y}) \end{aligned}$$

with $\bar{x} = (x_{id}, x_{fi}, x_{la}, x_{ci}, x_{co}, x_{ph})$ and $\bar{y} = (y_{oid}, y_{od}, y_{on}, y_{cid}, y_{ta})$ and where φ is a sentence that is satisfied by *every* database, e.g., $\varphi := \neg \exists z \neg z=z$.

To list the total number of orders for each customer in Berlin, we can use the SQL-statement

```
SELECT C.FirstName, C.LastName, COUNT(O.Id)
FROM Customer C, Order O
WHERE C.City = Berlin AND O.CustomerId = C.Id
GROUP BY C.FirstName, C.LastName
```

or, equivalently, the $\text{FOC}_1(\mathbb{P})$ -query

$$\{(x_{fi}, x_{la}, t(x_{fi}, x_{la})) : \varphi(x_{fi}, x_{la})\}$$

with $t(x_{fi}, x_{la}) :=$

$$\#(y_{oid}). \exists y_{od} \exists y_{on} \exists y_{ta} \exists x_{id} \exists x_{ci} \exists x_{co} \exists x_{ph} \left(\text{Order}(\bar{y}) \wedge \text{Customer}(\bar{x}) \right)$$

³taken from <http://www.dofactory.com/sql/group-by>

⁴This statement shall work for MySQL, PostgreSQL, and Microsoft SQL server; to make it work for Oracle, the statement has to be appended by the line `FROM dual`.

for $\bar{y} = (y_{oid}, y_{od}, y_{on}, x_{id}, y_{ta})$ and $\bar{x} = (x_{id}, x_{fi}, x_{la}, x_{ci}, x_{co}, x_{ph})$ and $\varphi(x_{fi}, x_{la}) :=$

$$\exists x_{id} \exists x_{ci} \exists x_{co} \exists x_{ph} (\mathbf{Customer}(\bar{x}) \wedge R_{Berlin}(x_{ci})).$$

Here, we use an atomic statement $R_{Berlin}(x_{ci})$ to express that “ $x_{ci} = \text{Berlin}$ ”. Of course, to avoid such constructions, we could extend the definition of $\text{FOC}_1(\mathbb{P})$ in the usual way by allowing constants taken from a fixed domain **dom** of potential database entries (cf. [1]).

Our query language is also capable of expressing more complicated queries:

Example 5.4. Consider a numerical predicate collection that contains the equality predicate $P_ =$ with $\llbracket P_ = \rrbracket = \{(m, m) : m \in \mathbb{Z}\}$. For better readability of $\text{FOC}_1(\mathbb{P})$ formulas we will write $t = t'$ instead of $P_=(t, t')$.

Consider the signature $\sigma := \{E, R, B, G\}$ where E is a binary relation symbol and R, B, G are unary relation symbols. We view a σ -structure \mathcal{A} as a directed graph where each node $a \in A$ may be coloured with 0, 1, 2, or 3 of the colours R (red), B (blue), and G (green).

The ground term $t_R := \#(x).R(x)$ specifies the total number of red nodes. The term

$$t_\Delta(x) := \#(y, z). (E(x, y) \wedge E(y, z) \wedge E(z, x))$$

specifies the number of directed triangles in which x participates. The formula $\varphi_{\Delta, R}(x) := t_\Delta(x) = t_R$ is satisfied by all nodes x such that the number of triangles in which x participates is the same as the total number of red nodes. The ground term $t_{\Delta, R} := \#(x). \varphi_{\Delta, R}(x)$ specifies the total number of such nodes. The term

$$t_B(x) := \#(y). (E(x, y) \wedge B(y))$$

specifies the number of blue neighbours of node x .

For the formula $\varphi_{B, \Delta, R}(x) := t_B(x) = t_\Delta(x) + t_{\Delta, R}$ the $\text{FOC}_1(\mathbb{P})[\sigma]$ -query

$$\{ (x, y, t_B(x) \cdot t_\Delta(y)) : (\varphi_{B, \Delta, R}(x) \wedge G(y)) \}$$

outputs all tuples in $A^2 \times \mathbb{Z}$ of the form (x, y, n) such that n is the product of the number of blue neighbours of x and the number of triangles in which y participates, y is green, and x is a node whose number of blue neighbours is equal to the sum of the number of triangles in which x participates and the total number of nodes that participate in exactly as many triangles as there are red nodes.

When speaking of an *algorithm with \mathbb{P} -oracle* we mean an algorithm that has available an oracle to decide, at unit cost, whether $(i_1, \dots, i_m) \in \llbracket P \rrbracket$ when given a $P \in \mathbb{P}$ and a tuple of integers (i_1, \dots, i_m) of arity $m = \text{ar}(\mathbb{P})$.

The paper’s main result reads as follows (see Section 8 for a precise definition of nowhere dense classes).

Theorem 5.5 (Main Theorem). *Let \mathcal{C} be an effectively nowhere dense class of structures. There is an algorithm with \mathbb{P} -oracle which receives as input an $\epsilon > 0$, an $\text{FOC}_1(\mathbb{P})$ -query q of the form $(*)$ for some signature σ , a σ -structure \mathcal{A} from \mathcal{C} , and a tuple $\bar{a} \in A^k$, and decides whether $\mathcal{A} \models \varphi[\bar{a}]$, and if so, computes the numbers $n_j := t_j^{\mathcal{A}}[\bar{a}]$ for all $j \in [\ell]$. The algorithm’s running time is $f(\|q\|, \epsilon) \cdot \|\mathcal{A}\|^{1+\epsilon}$, for a computable function f .*

Since the counting problem for an $\text{FOC}_1(\mathbb{P})$ -formula $\varphi(\bar{x})$ for $\bar{x} = (x_1, \dots, x_k)$ coincides with the task of evaluating the ground term $\#\bar{x}. \varphi(\bar{x})$ of $\text{FOC}_1(\mathbb{P})$, we immediately obtain:

Corollary 5.6. *On effectively nowhere dense classes \mathcal{C} , the counting problem for $\text{FOC}_1(\mathbb{P})$ is fixed-parameter almost linear. That is, there is an algorithm with \mathbb{P} -oracle which receives as input an $\epsilon > 0$, an $\text{FOC}_1(\mathbb{P})$ -formula $\varphi(\bar{x})$ of some signature σ , and σ -structure \mathcal{A} from \mathcal{C} , and computes the number $|\varphi(\mathcal{A})|$ of tuples $\bar{a} \in A^{|\bar{x}|}$ with $\mathcal{A} \models \varphi[\bar{a}]$ in time $f(\|\varphi\|, \epsilon) \cdot \|\mathcal{A}\|^{1+\epsilon}$, for a computable function f .*

The first step towards proving Theorem 5.5 is to use a standard construction for getting rid of the free variables. Given a query q of the form $(*)$, we extend the signature σ by fresh unary relation symbols X_1, \dots, X_k and let $\tilde{\sigma} := \sigma \cup \{X_1, \dots, X_k\}$. Given a σ -structure \mathcal{A} and a tuple $\bar{a} \in A^k$, we consider the $\tilde{\sigma}$ -expansion $\tilde{\mathcal{A}}$ of \mathcal{A} where $X_i^{\tilde{\mathcal{A}}} := \{a_i\}$ for all $i \in [k]$.

It is straightforward to translate $\varphi(\bar{x})$ into a $\tilde{\sigma}$ -sentence $\tilde{\varphi}$ such that $\tilde{\mathcal{A}} \models \tilde{\varphi}$ iff $\mathcal{A} \models \varphi[\bar{a}]$; and similarly, for each $j \in [\ell]$ we can translate the term $t_j(\bar{x})$ into a ground term \tilde{t}_j of signature $\tilde{\sigma}$ such that $\tilde{t}_j^{\tilde{\mathcal{A}}} = t_j^{\mathcal{A}}[\bar{a}]$: W.l.o.g. assume that all occurrences of the variables x_1, \dots, x_k in φ and t_1, \dots, t_ℓ are free. We can choose $\tilde{\varphi} := \exists x_1 \dots \exists x_k (\bigwedge_{i=1}^k X_i(x_i) \wedge \varphi(\bar{x}))$. For each $j \in [\ell]$, the term t_j is built using $+$ and \cdot from integers and from terms of the form $\# \bar{y}.\theta(\bar{x}, \bar{y})$. By replacing each $\theta(\bar{x}, \bar{y})$ by $\tilde{\theta}(\bar{y}) := \exists x_1 \dots \exists x_k (\bigwedge_{i=1}^k X_i(x_i) \wedge \theta(\bar{x}, \bar{y}))$, we obtain a ground term \tilde{t}_j with the desired property.

To prove Theorem 5.5 it therefore suffices to prove the following.

Lemma 5.7. *Let \mathcal{C} be an effectively nowhere dense class of structures. There is an algorithm with \mathbb{P} -oracle which receives as input an $\epsilon > 0$, a σ -structure \mathcal{A} from \mathcal{C} (for some signature σ) and either an $\text{FOC}_1(\mathbb{P})[\sigma]$ -sentence φ or an $\text{FOC}_1(\mathbb{P})[\sigma]$ -ground term t . The algorithm decides whether $\mathcal{A} \models \varphi$ and computes $t^{\mathcal{A}}$, resp. Letting ξ be the input expression φ or t , the algorithm's running time is $f(\|\xi\|, \epsilon) \cdot \|\mathcal{A}\|^{1+\epsilon}$, for a computable function f .*

The remainder of the paper is dedicated to the proof of Lemma 5.7. In fact, we prove a slightly stronger result: We cannot only evaluate sentences and ground terms, but also formulas with one free variable and unary terms simultaneously at all elements of the input structure, within the same time bounds.

6 A decomposition of $\text{FOC}_1(\mathbb{P})$

The first step towards proving Lemma 5.7 is to provide a decomposition of $\text{FOC}_1(\mathbb{P})$ -expressions into simpler expressions that can be evaluated in a structure \mathcal{A} by exploring for each element a in \mathcal{A} 's universe only a local neighbourhood around a . This section's main result is the Decomposition Theorem 6.10.

Let us fix a signature σ .

6.1 Connected local terms

The following lemma summarises easy facts concerning neighbourhoods; the proof is straightforward.

Lemma 6.1. *Let \mathcal{A} be a σ -structure, $r \geq 0$, $k \geq 1$, and $\bar{a} = (a_1, \dots, a_k) \in A^k$. $\mathcal{N}_r^{\mathcal{A}}(a_1, a_2)$ is connected $\iff \text{dist}^{\mathcal{A}}(a_1, a_2) \leq 2r+1$. If $\mathcal{N}_r^{\mathcal{A}}(\bar{a})$ is connected, then $N_r^{\mathcal{A}}(\bar{a}) \subseteq N_{r+(k-1)(2r+1)}^{\mathcal{A}}(a_i)$, for each $i \in [k]$.*

The notion of local formulas is defined as usual [17]: Let $r \in \mathbb{N}$. An $\text{FOC}(\mathbb{P})[\sigma]$ -formula $\varphi(\bar{x})$ with free variables $\bar{x} = (x_1, \dots, x_k)$ is r -local around \bar{x} if for every σ -structure \mathcal{A} and all $\bar{a} \in A^k$ we have $\mathcal{A} \models \varphi[\bar{a}] \iff \mathcal{N}_r^{\mathcal{A}}(\bar{a}) \models \varphi[\bar{a}]$. A formula is *local* if it is r -local for some $r \in \mathbb{N}$.

For an $r \in \mathbb{N}$ it is straightforward to construct an $\text{FO}[\sigma]$ -formula $\text{dist}_{\leq r}^{\sigma}(x, y)$ such that for every σ -structure \mathcal{A} and all $a, b \in A$ we have

$$\mathcal{A} \models \text{dist}_{\leq r}^{\sigma}[a, b] \iff \text{dist}^{\mathcal{A}}(a, b) \leq r.$$

To improve readability, we write $\text{dist}^{\sigma}(x, y) \leq r$ for $\text{dist}_{\leq r}^{\sigma}(x, y)$, and $\text{dist}^{\sigma}(x, y) > r$ for $\neg \text{dist}_{\leq r}^{\sigma}(x, y)$.

For every $k \in \mathbb{N}_{\geq 1}$ we let \mathcal{G}_k be the set of all undirected graphs G with vertex set $[k]$. For a graph $G \in \mathcal{G}_k$, a number $r \in \mathbb{N}$, a tuple $\bar{y} = (y_1, \dots, y_k)$ of k pairwise distinct variables, we

consider the formula

$$\delta_{G,r}^\sigma(\bar{y}) := \bigwedge_{\{i,j\} \in E(G)} \text{dist}^\sigma(y_i, y_j) \leq r \quad \wedge \quad \bigwedge_{\{i,j\} \notin E(G)} \text{dist}^\sigma(y_i, y_j) > r.$$

connected components of the r -neighbourhood $\mathcal{N}_r^\mathcal{A}(\bar{a})$ correspond to the connected components of G . Clearly, the formula $\delta_{G,2r+1}^\sigma(\bar{y})$ is r -local around its free variables \bar{y} .

The main ingredient of our decomposition of $\text{FOC}_1(\mathbb{P})$ -expressions are the connected local terms (*cl-terms*, for short), defined as follows.

Definition 6.2 (cl-Terms). Let $r \in \mathbb{N}$ and $k \in \mathbb{N}_{\geq 1}$. A *basic cl-term* (of radius r and width k) is a ground term g of the form

$$\#(y_1, \dots, y_k) \cdot (\psi(y_1, \dots, y_k) \wedge \delta_{G,2r+1}^\sigma(y_1, \dots, y_k))$$

or a unary term $u(y_1)$ of the form

$$\#(y_2, \dots, y_k) \cdot (\psi(y_1, \dots, y_k) \wedge \delta_{G,2r+1}^\sigma(y_1, \dots, y_k))$$

where $\bar{y} = (y_1, \dots, y_k)$ is a tuple of k pairwise distinct variables, $\psi(y_1, \dots, y_k)$ is an $\text{FO}[\sigma]$ -formula that is r -local around \bar{y} , and $G \in \mathcal{G}_k$ is *connected*.

A *cl-term* (of radius $\leq r$ and width $\leq k$) is built from basic cl-terms (of radius $\leq r$ and width $\leq k$) and integers by using rule (7) of Definition 3.1. I.e., a cl-term is a polynomial with integer coefficients, built from basic cl-terms t_1, \dots, t_ℓ (for $\ell \geq 0$).

Remark 6.3. Note that cl-terms are “easy” with respect to query evaluation in the following sense. Consider a basic cl-term $u(y_1)$ of the form

$$\#(y_2, \dots, y_k) \cdot (\psi(y_1, \dots, y_k) \wedge \delta_{G,2r+1}^\sigma(y_1, \dots, y_k)).$$

Recall from Definition 6.2 that G is a *connected* graph. Therefore, given a σ -structure \mathcal{A} and an element $a_1 \in A$, the number $u^\mathcal{A}[a_1]$ can be computed by only considering the R -neighbourhood of a_1 , for $R := r + (k-1)(2r+1)$ (cf. Lemma 6.1). After having computed the numbers $u^\mathcal{A}[a_1]$ for all $a_1 \in A$, the ground cl-term $g :=$

$$\#(y_1, \dots, y_k) \cdot (\psi(y_1, \dots, y_k) \wedge \delta_{G,2r+1}^\sigma(y_1, \dots, y_k))$$

can be evaluated easily, since $g^\mathcal{A} = \sum_{a_1 \in A} u^\mathcal{A}[a_1]$.

Our decomposition of $\text{FOC}_1(\mathbb{P})$ -expressions proceeds by induction on the construction of the input expression. The main technical tool for the construction is provided by the following lemma.

Lemma 6.4. Let $r \geq 0$, $k \geq 1$, and let $\bar{y} = (y_1, \dots, y_k)$ be a tuple of k pairwise distinct variables. Let $\psi(\bar{y})$ be an $\text{FO}[\sigma]$ -formula that is r -local around its free variables \bar{y} , and consider the terms g and $u(y_1)$ with

$$\begin{aligned} g &:= \#(y_1, \dots, y_k) \cdot \psi(y_1, \dots, y_k) \\ u(y_1) &:= \#(y_2, \dots, y_k) \cdot \psi(y_1, \dots, y_k). \end{aligned}$$

There exists a ground cl-term \hat{g} and a unary cl-term $\hat{u}(y_1)$, both of radius $\leq r$ and width $\leq k$, such that $\hat{g}^\mathcal{A} = g^\mathcal{A}$ and $\hat{u}^\mathcal{A}[a] = u^\mathcal{A}[a]$ for every σ -structure \mathcal{A} and every $a \in A$.

Furthermore, there is an algorithm which upon input of r and $\psi(\bar{y})$ constructs \hat{g} and $\hat{u}(y_1)$.

Proof. For a σ -structure \mathcal{A} and a formula $\vartheta(\bar{y})$ we consider the set

$$S_{\vartheta}^{\mathcal{A}} := \{ \bar{a} = (a_1, \dots, a_k) \in A^k : \mathcal{A} \models \vartheta[\bar{a}] \}.$$

Note that for every graph $G \in \mathcal{G}_k$ the formula

$$\psi_G(\bar{y}) := \psi(\bar{y}) \wedge \delta_{G, 2r+1}^{\sigma}(\bar{y})$$

is r -local around \bar{y} . Furthermore, for every σ -structure \mathcal{A} , the set $S_{\psi}^{\mathcal{A}}$ is the disjoint union of the sets $S_{\psi_G}^{\mathcal{A}}$ for all $G \in \mathcal{G}_k$. Therefore,

$$\begin{aligned} g &\equiv \sum_{G \in \mathcal{G}_k} \#(y_1, \dots, y_k) \cdot \psi_G(y_1, \dots, y_k) \quad \text{and} \\ u(y_1) &\equiv \sum_{G \in \mathcal{G}_k} \#(y_2, \dots, y_k) \cdot \psi_G(y_1, y_2, \dots, y_k). \end{aligned}$$

To complete the proof of Lemma 6.4, it therefore suffices to show that for every $G \in \mathcal{G}_k$ the terms

$$\begin{aligned} g_G^{\psi} &:= \#(y_1, \dots, y_k) \cdot \psi_G(y_1, \dots, y_k) \quad \text{and} \\ u_G^{\psi}(y_1) &:= \#(y_2, \dots, y_k) \cdot \psi_G(y_1, y_2, \dots, y_k) \end{aligned}$$

are equivalent to cl-terms of radius r . We prove this by an induction on the number of connected components of G . Precisely, we show that the following statement $(*)_c$ is true for every $c \in \mathbb{N}_{\geq 1}$.

$(*)_c$: For every $k \geq c$, for every tuple $\bar{y} = (y_1, \dots, y_k)$ of k pairwise distinct variables, for every $r \geq 0$, for every FO $[\sigma]$ -formula $\psi(\bar{y})$ that is r -local around \bar{y} , and for every graph $G \in \mathcal{G}_k$ that has at most c connected components, the terms g_G^{ψ} and $u_G^{\psi}(y_1)$ are equivalent to cl-terms of radius r .

The induction base for $c = 1$ is trivial: it involves only *connected* graphs G , for which by Definition 6.2 the terms g_G^{ψ} and $u_G^{\psi}(y_1)$ are basic cl-terms (of radius r).

For the induction step from c to $c+1$, consider some $k \geq c+1$ and a graph $G = (V, E) \in \mathcal{G}_k$ that has $c+1$ connected components. Let V' be the set of all nodes of V that are connected to the node 1, and let $V'' := V \setminus V'$.

Let $G' := G[V']$ and $G'' := G[V'']$ be the induced subgraphs of G on V' and V'' , respectively. Clearly, G is the disjoint union of G' and G'' , G' is connected, and G'' has c connected components.

To keep notation simple, we assume (without loss of generality) that $V' = \{1, \dots, \ell\}$ and $V'' = \{\ell+1, \dots, k\}$ for some ℓ with $1 \leq \ell < k$. For any tuple $\bar{z} = (z_1, \dots, z_k)$ we let $\bar{z}' := (z_1, \dots, z_{\ell})$ and $\bar{z}'' := (z_{\ell+1}, \dots, z_k)$.

Now consider a number $r \geq 0$ and the formula $\delta_{G, 2r+1}^{\sigma}(\bar{y})$ for $\bar{y} = (y_1, \dots, y_k)$. For every σ -structure \mathcal{A} and every tuple $\bar{a} = (a_1, \dots, a_k) \in A^k$ with $\mathcal{A} \models \delta_{G, 2r+1}^{\sigma}[\bar{a}]$, the r -neighbourhood $\mathcal{N}_r^{\mathcal{A}}(\bar{a})$ is the disjoint union of the r -neighbourhoods $\mathcal{N}_r^{\mathcal{A}}(\bar{a}')$ and $\mathcal{N}_r^{\mathcal{A}}(\bar{a}'')$.

Let $\psi(\bar{y})$ be an FO $[\sigma]$ -formula that is r -local around its free variables. By using the Feferman-Vaught Theorem (cf., [8, 18]), we can compute a decomposition of $\psi(\bar{y})$ into a formula $\hat{\psi}(\bar{y})$ (that depends on G) of the form

$$\bigvee_{i \in I} \left(\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \right),$$

where I is a finite non-empty set, each $\psi_i'(\bar{y}')$ is an FO $[\sigma]$ -formula that is r -local around \bar{y}' , each $\psi_i''(\bar{y}'')$ is an FO $[\sigma]$ -formula that is r -local around \bar{y}'' , and for every σ -structure \mathcal{A} and every $\bar{a} \in A^k$ with $\mathcal{A} \models \delta_{G, 2r+1}^{\sigma}[\bar{a}]$ the following is true:

(1) there exists at most one $i \in I$ such that $(\mathcal{A}, \bar{a}) \models (\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}''))$, and

(2) $\mathcal{A} \models \psi[\bar{a}] \iff \mathcal{A} \models \hat{\psi}[\bar{a}]$.

This implies that the set $S_{\psi_G}^{\mathcal{A}}$ is the disjoint union of the sets $S_{(\psi_i' \wedge \psi_i'' \wedge \delta_{G,r}^\sigma)}^{\mathcal{A}}$ for all $i \in I$. Consequently,

$$\begin{aligned} g_G^\psi &\equiv \sum_{i \in I} \#(y_1, \dots, y_k) \cdot (\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \wedge \delta_{G,2r+1}^\sigma(\bar{y})) \quad \text{and} \\ u_G^{\psi,i}(y_1) &\equiv \sum_{i \in I} \#(y_2, \dots, y_k) \cdot (\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \wedge \delta_{G,2r+1}^\sigma(\bar{y})). \end{aligned}$$

To complete the proof, it suffices to show that each of the terms

$$\begin{aligned} g_G^{\psi,i} &:= \#(y_1, \dots, y_k) \cdot (\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \wedge \delta_{G,2r+1}^\sigma(\bar{y})) \quad \text{and} \\ u_G^{\psi,i}(y_1) &:= \#(y_2, \dots, y_k) \cdot (\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \wedge \delta_{G,2r+1}^\sigma(\bar{y})) \end{aligned}$$

is equivalent to a cl-term of radius r .

By the definition of the formula $\delta_{G,2r+1}^\sigma(\bar{y})$ we obtain that the formula $\psi_i'(\bar{y}') \wedge \psi_i''(\bar{y}'') \wedge \delta_{G,2r+1}^\sigma(\bar{y})$ is equivalent to the formula

$$\underbrace{(\psi_i'(\bar{y}') \wedge \delta_{G',2r+1}^\sigma(\bar{y}'))}_{=: \vartheta'(\bar{y}')} \wedge \underbrace{(\psi_i''(\bar{y}'') \wedge \delta_{G'',2r+1}^\sigma(\bar{y}''))}_{=: \vartheta''(\bar{y}'')} \wedge \bigwedge_{\substack{j' \in V' \\ j'' \in V''}} \text{dist}^\sigma(y_{j'}, y_{j'') > 2r+1. \quad (1)$$

Therefore, for every σ -structure \mathcal{A} we have

$$\begin{aligned} S_{\psi_i' \wedge \psi_i'' \wedge \delta_{G,r}^\sigma}^{\mathcal{A}} &= (S_{\vartheta'}^{\mathcal{A}} \times S_{\vartheta''}^{\mathcal{A}}) \setminus T^{\mathcal{A}}, \quad \text{for} \\ T^{\mathcal{A}} &:= \{ \bar{a} \in A^k : \mathcal{A} \models \vartheta'[\bar{a}'], \mathcal{A} \models \vartheta''[\bar{a}''], (\mathcal{A}, \bar{a}) \not\models \bigwedge_{\substack{j' \in V' \\ j'' \in V''}} \text{dist}^\sigma(y_{j'}, y_{j'') > 2r+1 \}. \end{aligned}$$

Let \mathcal{H} be the set of all graphs $H \in \mathcal{G}_k$ with $H \neq G$, but $H[V'] = G'$ and $H[V''] = G''$. Clearly, every $H \in \mathcal{H}$ has at most c connected components. Furthermore, it is straightforward to see that for every σ -structure \mathcal{A} , the set $T^{\mathcal{A}}$ is the disjoint union of the sets

$$T_H^{\mathcal{A}} := \{ \bar{a} \in A^k : \mathcal{A} \models \vartheta'[\bar{a}'], \mathcal{A} \models \vartheta''[\bar{a}''], \mathcal{A} \models \delta_{H,2r+1}^\sigma(\bar{a}) \}$$

for all $H \in \mathcal{H}$. Since $T^{\mathcal{A}} \subseteq (S_{\vartheta'}^{\mathcal{A}} \times S_{\vartheta''}^{\mathcal{A}})$, we obtain that

$$(g_G^{\psi,i})^{\mathcal{A}} = |S_{\psi_i' \wedge \psi_i'' \wedge \delta_{G,r}^\sigma}^{\mathcal{A}}| = |S_{\vartheta'}^{\mathcal{A}}| \cdot |S_{\vartheta''}^{\mathcal{A}}| - \sum_{H \in \mathcal{H}} |T_H^{\mathcal{A}}|;$$

and this holds for every σ -structure \mathcal{A} . Therefore,

$$g_G^{\psi,i} \equiv \underbrace{(\# \bar{y}' \cdot \vartheta'(\bar{y}'))}_{=: t'} \cdot \underbrace{(\# \bar{y}'' \cdot \vartheta''(\bar{y}''))}_{=: t''} - \sum_{H \in \mathcal{H}} \underbrace{\# \bar{y} \cdot (\vartheta'(\bar{y}') \wedge \vartheta''(\bar{y}'') \wedge \delta_{H,2r+1}^\sigma(\bar{y}))}_{=: t_H}.$$

By the induction hypothesis $(*)_c$, each of the terms t' , t'' , and t_H is equivalent to a cl-term of radius r . Hence, also $g_G^{\psi,i}$ is equivalent to a cl-term of radius r .

To complete the proof, we need to show that also $u_G^{\psi,i}(y_1)$ is equivalent to a cl-term (of radius r). This can be done in a We proceed by a similar reasoning as above. Note that for every σ -structure \mathcal{A} and every $a_1 \in A$,

$$(u_G^{\psi,i})^{\mathcal{A}}[a_1] = |S_{\psi_i' \wedge \psi_i'' \wedge \delta_{G,2r+1}^\sigma}^{\mathcal{A}, a_1}|$$

where $S_{\psi'_i \wedge \psi''_i \wedge \delta_{G,2r+1}^\sigma}^{\mathcal{A},a_1}$ is defined as the set of all tuples $(a_2, \dots, a_k) \in A^{k-1}$ such that

$$(\mathcal{A}, (a_1, a_2, \dots, a_k)) \models \psi'_i(y_1, y_2, \dots, y_\ell) \wedge \psi''_i(y_{\ell+1}, \dots, y_k) \wedge \delta_{G,2r+1}^\sigma(y_1, y_2, \dots, y_k).$$

By (1) we know that $S_{\psi'_i \wedge \psi''_i \wedge \delta_{G,2r+1}^\sigma}^{\mathcal{A},a_1}$ is the set of all tuples $(a_2, \dots, a_k) \in A^{k-1}$ such that

$$(\mathcal{A}, (a_1, a_2, \dots, a_k)) \models \vartheta'(y_1, y_2, \dots, y_\ell) \wedge \vartheta''(y_{\ell+1}, \dots, y_k) \wedge \bigwedge_{\substack{j' \leq \ell \\ j'' \geq \ell+1}} \text{dist}^\sigma(y_{j'}, y_{j'') > 2r+1.$$

Analogously as above we have

$$S_{\psi'_i \wedge \psi''_i \wedge \delta_{G,2r+1}^\sigma}^{\mathcal{A},a_1} = (S_{\vartheta'}^{\mathcal{A},a_1} \times S_{\vartheta''}^{\mathcal{A}}) \setminus T^{\mathcal{A},a_1}, \quad \text{where}$$

$$S_{\vartheta'}^{\mathcal{A},a_1} := \{ (a_2, \dots, a_\ell) \in A^{\ell-1} : \mathcal{A} \models \vartheta'[a_1, a_2, \dots, a_\ell] \}$$

and where $T^{\mathcal{A},a_1}$ is the set of all tuples $(a_2, \dots, a_k) \in A^{k-1}$ such that

$$(\mathcal{A}, (a_1, a_2, \dots, a_k)) \models \vartheta'(y_1, y_2, \dots, y_\ell) \wedge \vartheta''(y_{\ell+1}, \dots, y_k) \wedge \neg \bigwedge_{\substack{j' \leq \ell \\ j'' \geq \ell+1}} \text{dist}^\sigma(y_{j'}, y_{j'') > 2r+1.$$

The set $T^{\mathcal{A},a_1}$ is the disjoint union of the sets $T_H^{\mathcal{A},a_1}$ for all $H \in \mathcal{H}$, where $T_H^{\mathcal{A},a_1}$ is defined as the set of all tuples $(a_2, \dots, a_k) \in A^{k-1}$ for which

$$(\mathcal{A}, (a_1, a_2, \dots, a_k)) \models \vartheta'(y_1, y_2, \dots, y_\ell) \wedge \vartheta''(y_{\ell+1}, \dots, y_k) \wedge \delta_{H,2r+1}^\sigma(y_1, y_2, \dots, y_k).$$

Since $T^{\mathcal{A},a_1} \subseteq (S_{\vartheta'}^{\mathcal{A},a_1} \times S_{\vartheta''}^{\mathcal{A}})$, we obtain that

$$(u_G^{\psi,i})^{\mathcal{A}}[a_1] = |S_{\psi'_i \wedge \psi''_i \wedge \delta_{G,2r+1}^\sigma}^{\mathcal{A},a_1}| = |S_{\vartheta'}^{\mathcal{A},a_1}| \cdot |S_{\vartheta''}^{\mathcal{A}}| - \sum_{H \in \mathcal{H}} |T_H^{\mathcal{A},a_1}|;$$

and this holds for every σ -structure \mathcal{A} and every $a_1 \in A$. Therefore,

$$u_G^{\psi,i}(y_1) \equiv t'(y_1) \cdot t'' - \sum_{H \in \mathcal{H}} t_H(y_1)$$

where

$$\begin{aligned} t_H(y_1) &:= \#(y_2, \dots, y_k) \cdot (\vartheta'(y_1, y_2, \dots, y_\ell) \wedge \vartheta''(y_{\ell+1}, \dots, y_k) \wedge \delta_{H,2r+1}^\sigma(y_1, y_2, \dots, y_k)), \\ t'' &:= \#(y_{\ell+1}, \dots, y_k) \cdot \vartheta''(y_{\ell+1}, \dots, y_k), \\ t'(y_1) &:= \begin{cases} \#(y_2, \dots, y_\ell) \cdot \vartheta'(y_1, y_2, \dots, y_\ell) & \text{if } \ell \geq 2, \\ \#(y_2) \cdot (\vartheta'(y_1) \wedge y_2 = y_1) & \text{if } \ell = 1. \end{cases} \end{aligned}$$

By the induction hypothesis $(*)_c$, each of the terms $t'(y_1)$, t'' , and $t_H(y_1)$ is equivalent to a cl-term of radius r . Hence, also $u_G^{\psi,i}(y_1)$ is equivalent to a cl-term of radius r . This completes the proof of Lemma 6.4. \square

As an easy consequence of Lemma 6.4 we obtain

Lemma 6.5. *Let $s \geq 0$ and let χ_1, \dots, χ_s be arbitrary sentences of signature σ .⁵ Let $r \geq 0$, $k \geq 1$, and let $\bar{y} = (y_1, \dots, y_k)$ be a tuple of k pairwise distinct variables. Let $\varphi(\bar{y})$ be a Boolean*

⁵We do not restrict attention to FO[σ]-sentences here — the χ_j 's may be sentences of any logic, e.g., FOC(\mathbb{P})[σ].

combination of the sentences χ_1, \dots, χ_s and of FO[σ]-formulas that are r -local around their free variables \bar{y} . Consider the ground term

$$g := \#(y_1, \dots, y_k) \cdot \varphi(y_1, \dots, y_k)$$

and the unary term

$$u(y_1) := \#(y_2, \dots, y_k) \cdot \varphi(y_1, y_2, \dots, y_k).$$

For every $J \subseteq [s]$ there is a ground cl-term \hat{g}_J and a unary cl-term $\hat{u}_J(y_1)$ (both of radius $\leq r$ and width $\leq k$) such that for every σ -structure \mathcal{A} there is exactly one set $J \subseteq [s]$ such that

$$\mathcal{A} \models \chi_J := \bigwedge_{j \in J} \chi_j \wedge \bigwedge_{j \in [s] \setminus J} \neg \chi_j,$$

and for this set J we have $\hat{g}_J^{\mathcal{A}} = g^{\mathcal{A}}$ and $\hat{u}_J^{\mathcal{A}}[a] = u^{\mathcal{A}}[a]$ for every $a \in A$.

Furthermore, there is an algorithm which upon input of r , $\varphi(\bar{y})$, and J constructs \hat{g}_J and $\hat{u}_J(y_1)$.

Proof. We can assume w.l.o.g. that $\varphi(\bar{y})$ is of the form

$$\bigvee_{J \subseteq [s]} (\chi_J \wedge \psi_J(\bar{y}))$$

where, for each $J \subseteq [s]$, $\psi_J(\bar{y})$ is an FO[σ]-formula that is r -local around its free variables \bar{y} .

For every $J \subseteq [s]$ let \hat{g}_J and $\hat{u}_J(y_1)$ be the cl-terms obtained by Lemma 6.4 for the terms $g_J := \#\bar{y} \cdot \psi_J(\bar{y})$ and $u_J(y_1) := \#(y_2, \dots, y_k) \cdot \psi_J(\bar{y})$.

Now consider an arbitrary $J \subseteq [s]$ and a σ -structure \mathcal{A} with $\mathcal{A} \models \chi_J$. Clearly,

$$g^{\mathcal{A}} = (\#\bar{y} \cdot \psi_J(\bar{y}))^{\mathcal{A}} = \hat{g}_J^{\mathcal{A}},$$

and

$$u^{\mathcal{A}}[a] = (\#(y_2, \dots, y_k) \cdot \psi_J(\bar{y}))^{\mathcal{A}}[a] = \hat{u}_J^{\mathcal{A}}[a], \quad \text{for every } a \in A.$$

Hence, the proof of Lemma 6.5 is complete. \square

6.2 A connected local normalform for FO

Definition 6.6. A formula in *Gaifman normal form* is a Boolean combination of FO[σ]-formulas $\psi(\bar{x})$ that are local around their free variables \bar{x} , and of *basic local sentences*, i.e., FO[σ]-sentences χ of the form

$$\exists y_1 \dots \exists y_k \left(\bigwedge_{1 \leq i < j \leq k} \text{dist}^\sigma(y_i, y_j) > 2r \wedge \bigwedge_{1 \leq i \leq k} \psi(y_i) \right),$$

where $k \geq 1$, $r \geq 0$, and $\psi(y)$ is an FO[σ]-formula that is r -local around its unique free variable y . The number r is called the *radius* of χ .

Theorem 6.7 (Gaifman [12]). Every FO[σ]-formula $\varphi(\bar{x})$ is equivalent to a formula in Gaifman normal form.

Furthermore, there is an algorithm which transforms an input formula $\varphi(\bar{x})$ into an equivalent formula $\varphi'(\bar{x})$ in Gaifman normal form. The algorithm also outputs the radius of each basic local sentence of φ' , and a number r such that every local formula $\psi(\bar{x})$ in φ' is r -local around \bar{x} .

By combining Lemma 6.4 with Gaifman's locality theorem, we obtain the following normal form for FO, which may be of independent interest.

Theorem 6.8 (cl-Normalform). *Every $\text{FO}[\sigma]$ -formula $\varphi(\bar{x})$ is equivalent to a Boolean combination of $\text{FO}[\sigma]$ -formulas $\psi(\bar{x})$ that are local around their free variables \bar{x} , and of statements of the form “ $g \geq 1$ ”, for a ground cl-term g .*

Furthermore, there is an algorithm which transforms an input $\text{FO}[\sigma]$ -formula $\varphi(\bar{x})$ into an equivalent such formula $\varphi'(\bar{x})$. The algorithm also outputs the radius of each ground cl-term in φ' , and a number r such that every local formula $\psi(\bar{x})$ in φ' is r -local around \bar{x} .

Proof. By Theorem 6.7 it suffices to translate a basic local sentence into a statement of the form “ $g \geq 1$ ” for a ground cl-term g .

For a basic local sentence $\chi := \exists y_1 \cdots \exists y_k \vartheta(y_1, \dots, y_k)$ with $\vartheta(y_1, \dots, y_k) :=$

$$\bigwedge_{1 \leq i < j \leq k} \text{dist}^\sigma(y_i, y_j) > 2r \quad \wedge \quad \bigwedge_{1 \leq i \leq k} \psi(y_i)$$

let g_χ be the ground term

$$g_\chi := \#(y_1, \dots, y_k) \cdot \vartheta(y_1, \dots, y_k).$$

Note that $\vartheta(y_1, \dots, y_k)$ is r -local around its free variables. Hence, by Lemma 6.4 we obtain a ground cl-term \hat{g}_χ such that $\hat{g}_\chi^\mathcal{A} = g_\chi^\mathcal{A}$ for every σ -structure \mathcal{A} . Furthermore, $\mathcal{A} \models \chi \iff g_\chi^\mathcal{A} \geq 1 \iff \hat{g}_\chi^\mathcal{A} \geq 1$. This completes the proof of Theorem 6.8. \square

We use the notion *cl-normalform* to denote the formulas $\varphi'(\bar{x})$ provided by Theorem 6.8. Note that these cl-normalforms do not necessarily belong to FO , but can be viewed as formulas in $\text{FOC}_1(\{\mathbb{P}_{\geq 1}\})$ (recall that $\llbracket \mathbb{P}_{\geq 1} \rrbracket = \mathbb{N}_{\geq 1}$), since statements of the form “ $g \geq 1$ ” can be expressed via $\mathbb{P}_{\geq 1}(g)$.

6.3 A decomposition of $\text{FOC}_1(\mathbb{P})$ -expressions

Our decomposition of $\text{FOC}_1(\mathbb{P})$ utilises Theorem 6.8 and is based on an induction on the maximal nesting depth of constructs of the form $\# \bar{y}$. We call this nesting depth the *#-depth* $d_\#(\xi)$ of a given formula or term ξ . Formally, $d_\#(\varphi)$ is defined as follows:

- (1) $d_\#(\varphi) := 0$, if φ is a formula of the form $x_1 = x_2$ or $R(x_1, \dots, x_{\text{ar}(R)})$
- (2) $d_\#(\neg \varphi) := d_\#(\varphi)$ and $d_\#((\varphi \vee \psi)) := \max\{d_\#(\varphi), d_\#(\psi)\}$
- (3) $d_\#(\exists y \varphi) := d_\#(\varphi)$
- (4) $d_\#(\mathbb{P}(t_1, \dots, t_m)) := \max\{d_\#(t_1), \dots, d_\#(t_m)\}$,
- (5) $d_\#(\# \bar{y} \cdot \varphi) := d_\#(\varphi) + 1$,
- (6) $d_\#(i) := 0$, for all terms $i \in \mathbb{Z}$
- (7) $d_\#((t_1 + t_2)) := d_\#((t_1 \cdot t_2)) := \max\{d_\#(t_1), d_\#(t_2)\}$, for all terms t_1 and t_2 .

The base case of our decomposition of $\text{FOC}_1(\mathbb{P})$ is provided by the following lemma; the lemma’s proof utilises Theorem 6.8.

Lemma 6.9. *Let φ be an $\text{FOC}_1(\mathbb{P})[\sigma]$ -formula of the form $\mathbb{P}(t_1, \dots, t_m)$ with $\mathbb{P} \in \mathbb{P}$, $m = \text{ar}(\mathbb{P})$, and where t_1, \dots, t_m are counting terms of #-depth at most 1. Then, φ is equivalent to a Boolean combination of*

- (i) *formulas of the form $\mathbb{P}(t'_1, \dots, t'_m)$, for cl-terms t'_1, \dots, t'_m where $\text{free}(t'_i) = \text{free}(t_i)$ for all $i \in [m]$,*
- (ii) *statements of the form “ $g \geq 1$ ” for ground cl-terms g , and*

(iii) statements of the form $P'(i_1, \dots, i_{m'})$ for $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$.

Furthermore, there is an algorithm which transforms an input formula φ into such a Boolean combination φ' , and which also outputs the radius of each cl-term in φ' .

Proof. Let φ be of the form $P(t_1, \dots, t_m)$ with $P \in \mathbb{P}$, $m = \text{ar}(P)$, and where t_1, \dots, t_m are counting terms of $\#$ -depth at most 1. From Definition 5.1 we know that either $\text{free}(\varphi) = \emptyset$ or $\text{free}(\varphi) = \{x\}$ for a variable x . Furthermore, we know that for every $i \in [m]$ the counting term t_i is built by using addition and multiplication based on integers and on counting terms θ' of the form $\#\bar{z}.\theta$, for a tuple of variables $\bar{z} = (z_1, \dots, z_k)$, such that $\text{free}(\theta) \setminus \{z_1, \dots, z_k\} \subseteq \{x\}$. Let Θ' be the set of all these counting terms θ' and let Θ be the set of all the according formulas θ .

By assumption we have $d_{\#}(\varphi) \leq 1$. Therefore, every $\theta \in \Theta$ has $\#$ -depth 0. We can thus view each such θ as an $\text{FO}[\sigma]$ -formula, possibly enriched by atomic sentences of the form $P'(i_1, \dots, i_{m'})$ with $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$.

By Theorem 6.8, for each θ in Θ we obtain an equivalent formula $\varphi^{(\theta)}$ in cl-normalform, possibly enriched by atomic sentences of the form $P'(i_1, \dots, i_{m'})$ with $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$. Let Φ be the set of all these $\varphi^{(\theta)}$.

For each θ in Θ , the formula $\varphi^{(\theta)}$ is a Boolean combination of (a) $\text{FO}[\sigma]$ -formulas that are local around the free variables of θ , and (b) statements of the form “ $g \geq 1$ ” for a ground cl-term g , and (c) statements of the form $P'(i_1, \dots, i_{m'})$ with $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$.

Let χ_1, \dots, χ_s be a list of all statements of the forms (b) or (c), such that each formula in Φ is a Boolean combination of statements in $\{\chi_1, \dots, \chi_s\}$ and of $\text{FO}[\sigma]$ -formulas that are local around their free variables. For every $J \subseteq [s]$ let $\chi_J := \bigwedge_{j \in J} \chi_j \wedge \bigwedge_{j \in [s] \setminus J} \neg \chi_j$.

Let $r \in \mathbb{N}$ be such that each of the local $\text{FO}[\sigma]$ -formulas that occur in a formula in Φ is r -local around its free variables. For each θ' in Θ' of the form $\#\bar{z}.\theta$, we apply Lemma 6.5 to the term

$$t^{(\theta')} := \#\bar{z}.\varphi^{(\theta)}$$

and obtain for every $J \subseteq [s]$ a cl-term $\hat{t}_J^{(\theta')}$ with the same free variables as θ' , for which the following is true:

- If $\text{free}(\theta') = \emptyset$, then $(\theta')^{\mathcal{A}} = (\hat{t}_J^{(\theta')})^{\mathcal{A}}$ for every σ -structure \mathcal{A} with $\mathcal{A} \models \chi_J$.
- If $\text{free}(\theta') = \{x\}$, then $(\theta')^{\mathcal{A}}[a] = (\hat{t}_J^{(\theta')})^{\mathcal{A}}[a]$ for every σ -structure \mathcal{A} with $\mathcal{A} \models \chi_J$ and every $a \in A$.

Thus, for each $J \subseteq [s]$ we have

$$(\chi_J \wedge P(t_1, \dots, t_m)) \equiv (\chi_J \wedge P(t_{1,J}, \dots, t_{m,J}))$$

where, for every $i \in [m]$, we let $t_{i,J}$ be the cl-term obtained from t_i by replacing each occurrence of a term $\theta' \in \Theta'$ by the term $\hat{t}_J^{(\theta')}$. In summary, we obtain the following:

$$\begin{aligned} \varphi &= P(t_1, \dots, t_m) \\ &\equiv \bigvee_{J \subseteq [s]} (\chi_J \wedge P(t_1, \dots, t_m)) \\ &\equiv \bigvee_{J \subseteq [s]} (\chi_J \wedge P(t_{1,J}, \dots, t_{m,J})) =: \varphi'. \end{aligned}$$

The formula χ_J is a Boolean combination of statements of the form “ $g \geq 1$ ” for ground cl-terms g and statements of the form $P'(i_1, \dots, i_{m'})$ for $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$. Furthermore, each of the terms $t_{i,J}$ is a cl-term with $\text{free}(t_{i,J}) = \text{free}(t_i)$. Thus, the proof of Lemma 6.9 is complete. \square

Theorem 6.10 (Decomposition of $\text{FOC}_1(\mathbb{P})$). *Let z be a fixed variable in vars . For every $d \in \mathbb{N}$ and every $\text{FOC}_1(\mathbb{P})[\sigma]$ -expression ξ which is either a formula $\varphi(\bar{x})$ or a ground term t of $\#$ -depth $d_\#(\xi) = d$, there exists a sequence $(L_1 \dots, L_{d+1}, \xi')$ with the following properties.*

(I) $L_i = (\tau_i, \iota_i)$, for every $i \in \{1, \dots, d+1\}$, where

- τ_i is a finite set of relation symbols of arity ≤ 1 that do not belong to $\sigma_{i-1} := \sigma \cup \bigcup_{j < i} \tau_j$, and
- ι_i is a mapping that associates with every symbol $R \in \tau_i$ a formula $\iota_i(R)$
 - (i) of the form $P(t_1, \dots, t_m)$, where $P \in \mathbb{P}$, $m = \text{ar}(P)$, and t_1, \dots, t_m are cl-terms of signature σ_{i-1} , such that $\text{free}(t_j) \subseteq \{z\}$ for each $j \in [m]$, or
 - (ii) of the form “ $g \geq 1$ ” for ground cl-terms g of signature σ_{i-1} .

If R has arity 0, then $\iota_i(R)$ has no free variable. If R has arity 1, then z is the unique free variable of $\iota_i(R)$ (thus, $\iota_i(R)$ is of the form (i)).

(II) If ξ is a ground term t , then $\xi' := t'$ is a ground cl-term of signature σ_{d+1} .

If ξ is a formula $\varphi(\bar{x})$, then $\xi' := \varphi'(\bar{x})$ is a Boolean combination of (A) $\text{FO}[\sigma_{d+1}]$ -formulas $\psi(\bar{x})$ that are local around their free variables \bar{x} , where $\sigma_{d+1} := \sigma \cup \bigcup_{1 \leq i \leq d+1} \tau_i$, and (B) statements of the form $R()$ where R is a 0-ary relation symbol in σ_{d+1} . In case that $\text{free}(\varphi) = \emptyset$, φ' only contains statements of the latter form.

(III) For every σ -interpretation $\mathcal{I} = (\mathcal{A}, \beta)$ we have

$$\llbracket \xi \rrbracket^{\mathcal{I}} = \llbracket \xi' \rrbracket^{\mathcal{I}_{d+1}}$$

(i.e., $t^{\mathcal{A}} = (t')^{\mathcal{A}_{d+1}}$ in case that ξ is a ground term t , and $\mathcal{I} \models \varphi$ iff $\mathcal{I}_{d+1} \models \varphi'$ in case that ξ is a formula φ), where $\mathcal{I}_{d+1} = (\mathcal{A}_{d+1}, \beta)$, and \mathcal{A}_{d+1} is the σ_{d+1} -expansion of \mathcal{A} defined as follows: $\mathcal{A}_0 := \mathcal{A}$, and for every $i \in [d+1]$, \mathcal{A}_i is the σ_i -expansion of \mathcal{A}_{i-1} , where for every unary $R \in \tau_i$ we have

$$R^{\mathcal{A}_i} := \{ a \in A : (\mathcal{A}_{i-1}, a) \models \iota_i(R) \}$$

and for every 0-ary $R \in \tau_i$ we have

$$R^{\mathcal{A}_i} := \begin{cases} \{ () \} & \text{if } \mathcal{A}_{i-1} \models \iota_i(R), \\ \emptyset & \text{if } \mathcal{A}_{i-1} \not\models \iota_i(R). \end{cases}$$

Moreover, there is an algorithm which constructs such a sequence $D = (L_1, \dots, L_{d+1}, \xi')$ for an input expression ξ . The algorithm also outputs the radius of each cl-term in D , and a number r such that every formula of type (A) in φ' is r -local around its free variables.

Proof. We first prove the theorem's statement for the case that the input expression ξ is a formula $\varphi(\bar{x})$.

We proceed by induction on i to construct for all $i \in \{0, 1, \dots, d\}$ a tuple $L_i = (\tau_i, \iota_i)$ and an $\text{FOC}_1[\sigma_i]$ -formula $\varphi_i(\bar{x})$ of $\#$ -depth $(d-i)$, such that for every σ -interpretation $\mathcal{I} = (\mathcal{A}, \beta)$ and the interpretation $\mathcal{I}_i := (\mathcal{A}_i, \beta)$ we have $\mathcal{I} \models \varphi \iff \mathcal{I}_i \models \varphi_i$.

For $i = 0$ we are done by letting $\tau_0 := \emptyset$, $\sigma_0 := \sigma$, $\varphi_0 := \varphi$, and letting ι_0 be the mapping with empty domain. Now assume that for some $i < d$, we have already constructed $L_i = (\tau_i, \iota_i)$ and φ_i . To construct $L_{i+1} = (\tau_{i+1}, \iota_{i+1})$ and φ_{i+1} , we proceed as follows.

Let Π be the set of all $\text{FOC}_1(\mathbb{P})[\sigma_i]$ -formulas of $\#$ -depth ≤ 1 of the form $P(t_1, \dots, t_m)$, for $P \in \mathbb{P}$ and $m = \text{ar}(P)$, that occur in φ_i .

Now consider an arbitrary formula π in Π of the form $P(t_1, \dots, t_m)$. From Definition 5.1 we know that there is a variable y such that $\text{free}(t_j) \subseteq \{y\}$ for every $j \in [m]$. By Lemma 6.9, π is equivalent to a Boolean combination π' of

- (a) formulas of the form $P(t'_1, \dots, t'_m)$, for cl-terms t'_1, \dots, t'_m of signature σ_i , where $\text{free}(t'_j) = \text{free}(t_j) \subseteq \{y\}$ for each $j \in [m]$,
- (b) statements of the form “ $g \geq 1$ ” for ground cl-terms g of signature σ_i , and
- (c) statements of the form $P'(i_1, \dots, i_{m'})$ for $P' \in \mathbb{P}$, $m' = \text{ar}(P')$, and integers $i_1, \dots, i_{m'}$.

For each statement χ of the form (b) or (c), we include into τ_{i+1} a 0-ary relation symbol R_χ , we replace each occurrence of χ in π' with the new atomic formula $R_\chi()$, and we let $\iota_{i+1}(R_\chi) := \chi$. For each statement χ in π of the form (a) we proceed as follows. If $\text{free}(\chi) = \emptyset$, then we include into τ_{i+1} a 0-ary relation symbol R_χ , we replace each occurrence of χ in π' with the new atomic formula $R_\chi()$, and we let $\iota_{i+1}(R_\chi) := \chi$. If $\text{free}(\chi) = \{y\}$, then we include into τ_{i+1} a unary relation symbol R_χ , we replace each occurrence of χ in π' by the new atomic formula $R_\chi(y)$, and we let $\iota_{i+1}(R_\chi)$ be the formula obtained from χ by consistently replacing every free occurrence of the variable y by the variable z . We write π'' for the resulting formula π' .

Clearly, π'' is a quantifier-free $\text{FO}[\sigma_{i+1}]$ -formula, for $\sigma_{i+1} := \sigma_i \cup \tau_i$; in particular, it has $\#$ -depth 0. It is straightforward to see that for every σ -interpretation $\mathcal{I} = (\mathcal{A}, \beta)$ we have

$$\mathcal{I}_i \models \pi \iff \mathcal{I}_{i+1} \models \pi'',$$

for $\mathcal{I}_i := (\mathcal{A}_i, \beta)$ and $\mathcal{I}_{i+1} := (\mathcal{A}_{i+1}, \beta)$.

The induction step is completed by letting φ_{i+1} be the formula obtained from φ_i by replacing every occurrence of a formula $\pi \in \Pi$ by the formula π'' . It can easily be verified that φ_{i+1} is an $\text{FOC}_1(\mathbb{P})[\sigma_{i+1}]$ -formula of $\#$ -depth $d_\#(\varphi_i) - 1 = ((d-i)-1) = (d-(i+1))$, and that $\mathcal{I}_i \models \varphi_i \iff \mathcal{I}_{i+1} \models \varphi_{i+1}$.

By the above induction we have constructed L_1, \dots, L_d and an $\text{FOC}_1(\mathbb{P})[\sigma_d]$ -formula φ_d of $\#$ -depth 0. Since $d_\#(\varphi_d) = 0$, we can view φ_d as an $\text{FO}[\sigma_d]$ -formula, possibly enriched by atomic sentences of the form $P(i_1, \dots, i_m)$ with $P \in \mathbb{P}$, $m = \text{ar}(P)$, and integers i_1, \dots, i_m . By Theorem 6.8 we obtain an equivalent formula $\tilde{\varphi}$ of signature σ_d in cl-normalform, possibly enriched by atomic sentences of the form $P(i_1, \dots, i_m)$ with $P \in \mathbb{P}$, $m = \text{ar}(P)$, and integers i_1, \dots, i_m . I.e., $\tilde{\varphi}$ is a Boolean combination of

- (A) $\text{FO}[\sigma_d]$ -formulas that are local around their free variables \bar{x} ,
- (B) statements of the form “ $g \geq 1$ ”, for a ground cl-term g of signature σ_d , and
- (C) statements of the form $P(i_1, \dots, i_m)$ with $P \in \mathbb{P}$, $m = \text{ar}(P)$, and integers i_1, \dots, i_m .

For each statement χ of the form (B) or (C) we include into τ_{d+1} a new relation symbol R_χ of arity 0, we replace each occurrence of χ in $\tilde{\varphi}$ by the new atomic formula $R_\chi()$, and we let $\iota_{d+1}(R_\chi) := \chi$. Letting $\xi' := \varphi'$ be the resulting formula $\tilde{\varphi}$ completes the proof of Theorem 6.10 for the case that the input expression ξ is a formula.

Let us now turn to the case where the input expression ξ is a ground term t . Then, t is built using $+$ and \cdot from integers and from ground terms g of the form $\# \bar{y}. \theta(\bar{y})$. Let S be the set of all these ground terms g , and let Θ be the set of all according formulas $\theta(\bar{y})$. We have already proven the theorem's statement for the case where the input expression is a formula. For each $\theta \in \Theta$, we therefore obtain a sequence $D_\theta = (L_1^\theta, \dots, L_{d_\theta+1}^\theta, \theta')$ with $d_\theta = d_\#(\theta)$ and $L_i^\theta = (\tau_i^\theta, \iota_i^\theta)$.

Each term $g \in S$ is of the form $\# \bar{y}. \theta(\bar{y})$ for some $\theta \in \Theta$. Clearly, $g^{\mathcal{A}} = (g')^{\mathcal{A}_{d_\theta+1}}$ for $g' := \# \bar{y}. \theta'(\bar{y})$. Note that θ' is local around its free variables \bar{y} . Therefore, from Lemma 6.4(a) we obtain a ground cl-term \hat{g}' that is equivalent to g' . We let $\xi' := t'$ be the ground cl-term obtained from t by replacing every term $g \in S$ by \hat{g}' . We are done by letting $L_i := (\tau_i, \iota_i)$ for each $i \in \{1, \dots, d+1\}$, where τ_i (and ι_i) is the union of τ_i^θ (and ι_i^θ , respectively) for all $\theta \in \Theta$. This finally completes the proof of Theorem 6.10. \square

We call the sequence $(L_1, \dots, L_{d_{\#}(\xi)+1}, \xi')$ obtained from Theorem 6.10 for an $\text{FOC}_1(\mathbb{P})$ -formula or ground term ξ a *cl-decomposition* of ξ .

Assume, we have available an efficient algorithm \mathbb{A} for computing the value $u^{\mathcal{B}}[b_1]$ of a unary basic cl-term $u(y_1)$ in a structure \mathcal{B} for all values $b_1 \in B$. This algorithm can also be used to compute the value of a ground basic cl-term $g := \#(y_1, \dots, y_k) \cdot \psi(y_1, \dots, y_k)$ in \mathcal{B} , since $g^{\mathcal{B}} = \sum_{b_1 \in B} u^{\mathcal{B}}[b_1]$ for the unary basic cl-term $u(y_1) := \#(y_2, \dots, y_k) \cdot \psi(y_1, y_2, \dots, y_k)$.

We argue that by Theorem 6.10, the algorithm \mathbb{A} can also be used to evaluate an $\text{FOC}_1(\mathbb{P})$ -expression ξ that is either a ground term t or a sentence φ in a σ -structure \mathcal{A} . To evaluate ξ in a \mathcal{A} we can proceed as follows.

- (1) Use Theorem 6.10 to compute a cl-decomposition $D = (L_1, \dots, L_{d+1}, \xi')$ of ξ , for $d := d_{\#}(\xi)$.
- (2) Let $\mathcal{A}_0 := \mathcal{A}$.
- (3) For each $i \in [d+1]$, compute the σ_i -expansion of \mathcal{A}_{i-1} . To achieve this, consider for each $R \in \tau_i$ the formula $\iota_i(R)$. This formula is a very simple statement concerning one or several cl-terms (each of which is a polynomial built from integers and basic cl-terms). Let t_1, \dots, t_s be the list of all basic cl-terms that appear in $\iota_i(R)$. For each $j \in [s]$ use algorithm \mathbb{A} to compute the values $t_j^{\mathcal{A}}[a]$ for all $a \in A$ (resp., the value $t_j^{\mathcal{A}}$, if t_j is ground). Then, combine the values and use a \mathbb{P} -oracle to check for each $a \in A$ whether $\iota_i(R)$ is satisfied by (\mathcal{A}_{i-1}, a) , and store the new relation $R^{\mathcal{A}_i}$ accordingly.
- (4) If ξ is a sentence φ , then φ' is a Boolean combination of statements of the form $R()$, for 0-ary relation symbols $R \in \sigma_{d+1}$. Thus, checking whether $\mathcal{A}_{d+1} \models \varphi'$ boils down to evaluating a propositional formula, and hence is easy.

If ξ is a ground term t , then t' is a ground cl-term. I.e., t' is a polynomial built from integers and ground basic cl-terms t'_1, \dots, t'_s for some $s \geq 1$. For each $j \in [s]$ we use algorithm \mathbb{A} to compute the value of t'_j in \mathcal{A}_{d+1} . Afterwards, we combine these values to compute the value of t' in \mathcal{A}_{d+1} .

From [10, 14, 23] we obtain fixed-parameter almost linear algorithms for counting the number of solutions of FO-queries on planar graphs, classes of bounded local tree-width, classes of bounded expansion, and—most generally—classes of locally bounded expansion. By the above approach, this immediately provides us with an FPT algorithm for $\text{FOC}_1(\mathbb{P})$ on these classes. For nowhere dense classes, though, it is not so easy to generalise the FO model-checking algorithm of [13] to compute the values of unary cl-terms. The remainder of the paper is dedicated to this task.

7 Neighbourhood covers and local evaluation

The techniques of the previous section enable us to reduce the evaluation of $\text{FOC}_1(\mathbb{P})$ -sentences and ground terms to the evaluation of unary basic cl-terms. To obtain an efficient algorithm for evaluating the latter on structures \mathcal{A} from a nowhere dense class of structures, we need to provide a variant of basic cl-terms (along with techniques to decompose such terms) that are based on so-called neighbourhood covers.

An *r-neighbourhood cover* of a structure \mathcal{A} is a mapping $\mathcal{X} : A \rightarrow 2^A$ such that for every $a \in A$ the set $\mathcal{X}(a)$ is connected in the Gaifman graph $G_{\mathcal{A}}$ of \mathcal{A} and it holds that $N_r^{\mathcal{A}}(a) \subseteq \mathcal{X}(a)$. The sets $\mathcal{X}(a)$ (for $a \in A$), and depending on the context also the induced substructures $\mathcal{A}[\mathcal{X}(a)]$, are called the *clusters* of the cover. Usually, we want the clusters to have small radius, where the *radius* of a connected set $X \subseteq A$ is the least s such that there is a $c \in X$ such that $X \subseteq N_s^{\mathcal{A}[X]}(c)$. Moreover, we want a neighbourhood cover to be *sparse*, which means that no $b \in A$ appears in too many of the sets $\mathcal{X}(a)$. We will see later (Theorem 8.1) that in structures from a nowhere

dense class of structures we can efficiently construct sparse r -neighbourhood covers of radius at most $2r$. In this section, we do not have to worry about the radius or sparsity of neighbourhood covers.

We need some additional terminology for neighbourhood covers. We write $X \in \mathcal{X}$ to express that X is a cluster of \mathcal{X} , i.e., $X = \mathcal{X}(a)$ for some $a \in A$. We say that a cluster $X \in \mathcal{X}$ s -covers a tuple $\bar{a} \in A^k$ if $N_s^A(\bar{a}) \subseteq X$. Note that $\mathcal{X}(a)$ r -covers a , but there may be other clusters $X \in \mathcal{X}$ that r -cover a as well.

FO^+ is the extension of first-order logic by adding new atomic formulas $\text{dist}(x, y) \leq d$, with the obvious meaning. Note that FO^+ is only a syntactic extension and not more expressive than FO , because the “distance atoms” $\text{dist}(x, y) \leq d$ can be replaced by first-order formulas. However, a first-order formula expressing $\text{dist}(x, y) \leq d$ has quantifier rank $\log d$. We now introduce a complicated 2-parameter “rank measure” for FO^+ -formulas where the contribution of distance atoms to the rank is fine-tuned to exactly the right value. Let $q, \ell \in \mathbb{N}$. We say that an FO^+ -formula φ has q -rank at most ℓ if it has quantifier-rank at most ℓ and each distance atom $\text{dist}(x, y) \leq d$ in the scope of $i \leq \ell$ quantifiers satisfies $d \leq (4q)^{q+\ell-i}$. We let $f_q(\ell) := (4q)^{q+\ell}$.

7.1 Rank-preserving locality

The main result of this subsection, Theorem 7.1, allows us to reduce the “global” evaluation of a first-order formula in a structure \mathcal{A} to the “local” evaluation of formulas in the clusters of a neighbourhood cover of \mathcal{A} .

To formulate the theorem, we need some more notation. We need to *normalise* FO^+ -formulas in such a way that for every (finite) signature σ and all $q, \ell, k \in \mathbb{N}$ the set $\Phi^+(\sigma, k, q, \ell)$ of normalised $\text{FO}^+[\sigma]$ -formulas of q -rank at most ℓ with free variables among x_1, \dots, x_k is finite and that every $\text{FO}^+[\sigma]$ -formula of q -rank at most ℓ with free variables among x_1, \dots, x_k is equivalent to a formula in $\Phi^+(\sigma, k, q, \ell)$. We can do this in such a way that the set $\Phi^+(\sigma, k, q, \ell)$ is computable from σ, k, q, ℓ and that there is a normalisation algorithm that, given an $\text{FO}^+[\sigma]$ -formula of q -rank at most ℓ with free variables among x_1, \dots, x_k , computes an equivalent formula in $\Phi^+(\sigma, k, q, \ell)$.

For a signature σ and numbers $q, \ell \in \mathbb{N}$, we let $\sigma \star (q, \ell)$ be the signature obtained from σ by adding a fresh unary relation symbol P_φ for each $\varphi(x_1) \in \Phi^+(\sigma, 1, q, \ell)$. For σ -structure \mathcal{A} and an r -neighbourhood cover \mathcal{X} of \mathcal{A} , we let $\mathcal{A} \star_{\mathcal{X}}(q, \ell)$ be the $\sigma \star (q, \ell)$ -expansion of \mathcal{A} in which P_φ is interpreted by the set of all $a \in A$ such that $\mathcal{A}[\mathcal{X}(a)] \models \varphi[a]$. We let $\sigma \star^0(q, \ell) := \sigma$ and $\mathcal{A} \star_{\mathcal{X}}^0(q, \ell) := \mathcal{A}$. For $i \geq 0$, we let $\sigma \star^{i+1}(q, \ell) := (\sigma \star^i(q, \ell)) \star (q, \ell)$ and $\mathcal{A} \star_{\mathcal{X}}^{i+1}(q, \ell) := (\mathcal{A} \star_{\mathcal{X}}^i(q, \ell)) \star_{\mathcal{X}}(q, \ell)$. Note that for $i < j$ we have $\sigma \star^i(q, \ell) \subseteq \sigma \star^j(q, \ell)$, and the $\sigma \star^i(q, \ell)$ -structure $\mathcal{A} \star_{\mathcal{X}}^i(q, \ell)$ is the $\sigma \star^i(q, \ell)$ -reduct of the $\sigma \star^j(q, \ell)$ -structure $\mathcal{A} \star_{\mathcal{X}}^j(q, \ell)$.

An (r, k) -independence sentence is of the form

$$\exists x_1 \dots \exists x_{k'} \left(\bigwedge_{1 \leq i < j \leq k'} \text{dist}(x_i, x_j) > r' \wedge \bigwedge_{1 \leq i \leq k'} \psi(x_i) \right),$$

where $k' \leq k$ and $r' \leq r$ and $\psi(x)$ is quantifier-free.

Let \mathcal{A} be a σ -structure, let $k \geq 1$, let $\bar{a} = (a_1, \dots, a_k) \in A^k$ and let $r \geq 0$. By $G_{\bar{a}, r}^{\mathcal{A}}$ we denote the graph with vertex set $[k]$ where there is an edge between nodes i and j iff $i \neq j$ and $\text{dist}^{\mathcal{A}}(a_i, a_j) \leq r$. We will often omit the superscript \mathcal{A} and simply write $G_{\bar{a}, r}$.

We say that \bar{a} is r -connected if the graph $G_{\bar{a}, r}$ is connected. An r -component of \bar{a} in \mathcal{A} is the vertex set of a connected component of the graph $G_{\bar{a}, r}$.

For an arbitrary set $J \subseteq [k]$, by \bar{a}_J we denote the projection of \bar{a} to the positions in J .

Theorem 7.1 (Rank-Preserving Normal Form). *Let $q, k \in \mathbb{N}$ such that $k \leq q$, let $\ell := q - k$, $r := f_q(\ell)$, and let $\sigma^* := \sigma \star^\ell(q, \ell)$. Let $\varphi(\bar{x})$, where $\bar{x} = (x_1, \dots, x_k)$, be an $\text{FO}^+[\sigma]$ -formula of q -rank at most ℓ .*

Then for each graph $G \in \mathcal{G}_k$ there are an $m_G \in \mathbb{N}$ and for each $i \in [m_G]$

- a Boolean combination ξ_G^i of (r, q) -independence sentences of signature σ^* and
- for each connected component I of G an $\text{FO}^+[\sigma^*]$ -formula $\psi_{G,I}^i(\bar{x}_I)$ of q -rank at most ℓ

such that the following holds.

- (1) For all σ -structures \mathcal{A} , all kr -neighbourhood covers \mathcal{X} of \mathcal{A} , and all $\bar{a} \in A^k$ we have $\mathcal{A} \models \varphi[\bar{a}]$ if and only if for $G := G_{\bar{a},r}$ and $\mathcal{A}^* := \mathcal{A} \star_{\mathcal{X}}^\ell(q, \ell)$ there is an $i \in [m_G]$ such that $\mathcal{A}^* \models \xi_G^i$ and for every connected component I of G there is an $X \in \mathcal{X}$ that r -covers \bar{a}_I and $\mathcal{A}^*[X] \models \psi_{G,I}^i[\bar{a}_I]$.
- (2) For all σ -structures \mathcal{A} , all kr -neighbourhood covers \mathcal{X} of \mathcal{A} , and all $\bar{a} \in A^k$ there is at most one $i \in [m_G]$ for $G := G_{\bar{a},r}$ such that the conditions of (1) hold.
- (3) For all σ -structures \mathcal{A} , all kr -neighbourhood covers \mathcal{X} of \mathcal{A} , all $\bar{a} \in A^k$, all connected components I of $G := G_{\bar{a},r}$, all $X, X' \in \mathcal{X}$ that both r -cover \bar{a}_I , and all $i \in [m_G]$, $\mathcal{A}^*[X] \models \psi_{G,I}^i[\bar{a}_I] \iff \mathcal{A}^*[X'] \models \psi_{G,I}^i[\bar{a}_I]$.
- (4) The ξ_G^i and $\psi_{G,I}^i$ can be computed from q, k, G, φ .

The proof is based on an Ehrenfeucht-Fraïssé game argument, which in its basic structure is similar to the proof of Gaifman's theorem by Ehrenfeucht-Fraïssé games (see [7]). Compared to the “Rank-Preserving Locality Theorem” of [13], the main challenge here is to deal with free variables, which turned out to be nontrivial (as can already be seen from the complicated statement of the theorem).

Let \mathcal{A}, \mathcal{B} be σ -structures. A *partial r -isomorphism* from \mathcal{A} to \mathcal{B} is a mapping $h : X \rightarrow Y$ with domain $X \subseteq A$ and range $Y \subseteq B$ that is an isomorphism between the induced substructure $\mathcal{A}[X]$ and the induced substructure $\mathcal{B}[Y]$ and in addition, preserves distances up to r , that is, for all $a, a' \in X$ either $\text{dist}^{\mathcal{A}}(a, a') = \text{dist}^{\mathcal{B}}(h(a), h(a'))$ or $\text{dist}^{\mathcal{A}}(a, a') > r$ and $\text{dist}^{\mathcal{B}}(h(a), h(a')) > r$.

Let $\bar{a} = (a_1, \dots, a_k) \in A^k$, $\bar{b} = (b_1, \dots, b_k) \in B^k$ and $q, \ell \in \mathbb{N}$ such that $\ell \leq q$. The ℓ -round EF_q^+ -game on $(\mathcal{A}, \bar{a}, \mathcal{B}, \bar{b})$ is played by two players, called *Spoiler* and *Duplicator*. The game is played for ℓ rounds. In round i , Spoiler picks an element $a_{k+i} \in A$ or an element $b_{k+i} \in B$. If Spoiler picks $a_{k+i} \in A$, then Duplicator answers by choosing a $b_{k+i} \in B$ and if Spoiler picks $b_{k+i} \in B$, then Duplicator answers by choosing an $a_{k+i} \in A$. Duplicator wins the game if for $0 \leq i \leq \ell$, the mapping $a_j \mapsto b_j$ for $1 \leq j \leq k+i$ is a partial $f_q(\ell-i)$ -isomorphism.

Theorem 7.2 ([13]). *For all $k, q, \ell \in \mathbb{N}$ with $0 \leq \ell \leq q$, all σ -structures \mathcal{A} and \mathcal{B} , and all tuples $\bar{a} \in A^k$ and $\bar{b} \in B^k$, the following are equivalent.*

- (1) Duplicator has a winning strategy for the ℓ -round EF_q^+ game on $(\mathcal{A}, \bar{a}, \mathcal{B}, \bar{b})$.
- (2) (\mathcal{A}, \bar{a}) and (\mathcal{B}, \bar{b}) satisfy the same $\text{FO}^+[\sigma]$ -formulas $\varphi(x_1, \dots, x_k)$ of q -rank at most ℓ .

In order to prove Theorem 7.1, we need the following lemma.

Lemma 7.3. *Let $q, k \in \mathbb{N}$ such that $k \leq q$, and let $\ell := q - k$ and $r := f_q(\ell)$. Let \mathcal{A} and \mathcal{B} be σ -structures, let \mathcal{X} and \mathcal{Y} be r -neighbourhood covers of \mathcal{A} and \mathcal{B} , respectively, and let $\mathcal{A}^* := \mathcal{A} \star_{\mathcal{X}}^\ell(q, \ell)$ and $\mathcal{B}^* := \mathcal{B} \star_{\mathcal{Y}}^\ell(q, \ell)$. Let $\bar{a} = (a_1, \dots, a_k) \in A^k$ and $\bar{b} = (b_1, \dots, b_k) \in B^k$ such that the following conditions are satisfied.*

- (i) For all $i, j \in [k]$ it holds that $\text{dist}^{\mathcal{A}}(a_i, a_j) \leq r \iff \text{dist}^{\mathcal{B}}(b_i, b_j) \leq r$.
- (ii) For every r -component J of \bar{a} there are clusters $X \in \mathcal{X}$ and $Y \in \mathcal{Y}$ such that X r -covers \bar{a}_J and Y r -covers \bar{b}_J and $(\mathcal{A}^*[X], \bar{a}_J)$ and $(\mathcal{B}^*[Y], \bar{b}_J)$ satisfy the same $\text{FO}^+[\sigma^*]$ -formulas $\varphi(\bar{x}_J)$ of q -rank at most ℓ .
- (iii) The structures \mathcal{A}^* and \mathcal{B}^* satisfy the same (r, q) -independence sentences.

Then (\mathcal{A}, \bar{a}) and (\mathcal{B}, \bar{b}) satisfy the same $\text{FO}^+[\sigma]$ sentences of q -rank at most ℓ .

Proof. For $k \leq p \leq q$, let $r_p := f_q(q-p)$. Then $r_k = r$. Let $\sigma^p := \sigma \star^{q-p}(q, \ell)$ and $\mathcal{A}^p := \mathcal{A} \star_{\mathcal{X}}^{q-p}(q, \ell)$, $\mathcal{B}^p := \mathcal{B} \star_{\mathcal{Y}}^{q-p}(q, \ell)$. Then $\mathcal{A}^k = \mathcal{A}^*$ and $\mathcal{A}^q = \mathcal{A}$, and similarly, $\mathcal{B}^k = \mathcal{B}^*$ and $\mathcal{B}^q = \mathcal{B}$.

We shall prove that Duplicator has a winning strategy for the ℓ -round EF_q^+ game on $(\mathcal{A}, \bar{a}, \mathcal{B}, \bar{b})$. Positions of the game will be pairs of tuples $\bar{a}^p = (a_1, \dots, a_p)$, $\bar{b}^p = (b_1, \dots, b_p)$ of length $k \leq p \leq q$ whose first k entries coincide with the entries of \bar{a}, \bar{b} .

We shall prove by induction on p that Duplicator can maintain the following invariants for the position \bar{a}^p, \bar{b}^p reached after $(p-k)$ rounds of the game.

- (I) For all $i, j \in [p]$, either $\text{dist}^{\mathcal{A}}(a_i, a_j) = \text{dist}^{\mathcal{B}}(b_i, b_j) \leq r_p$ or $\text{dist}^{\mathcal{A}}(a_i, a_j) > r_p$ and $\text{dist}^{\mathcal{B}}(b_i, b_j) > r_p$.
- (II) For every r_p -component J of \bar{a}^p in \mathcal{A} there are induced substructures \mathcal{A}_J^p of \mathcal{A}^p and \mathcal{B}_J^p of \mathcal{B}^p such that
 - a. $N_{r_p}^{\mathcal{A}}(\bar{a}_J^p) \subseteq A_J$ and $N_{r_p}^{\mathcal{B}}(\bar{b}_J^p) \subseteq B_J$;
 - b. $(\mathcal{A}_J^p, \bar{a}_J^p)$ and $(\mathcal{B}_J^p, \bar{b}_J^p)$ satisfy the same $\text{FO}^+[\sigma^p]$ formulas $\varphi(\bar{x}_J)$ of q -rank at most $(q-p)$.

Note that for $p = q$ this, in particular, implies that after $(q-k) = \ell$ rounds Duplicator's winning condition is satisfied.

For the base step $p = k$ we let $\mathcal{A}_J^k := \mathcal{A}^*[X]$ and $\mathcal{B}_J^k := \mathcal{B}^*[Y]$. Then (II) follows from (ii) and (I) follows from (i) and (ii).

Now suppose that (I) and (II) hold for \bar{a}^p, \bar{b}^p , and in round $(p+1-k)$ of the game, Spoiler picks an element $a_{p+1} \in A$.

Case 1: $\text{dist}^{\mathcal{A}}(a_i, a_{p+1}) \leq r_{p+1}$ for some $i \in [p]$.

Let $\bar{x} = (x_1, \dots, x_p)$ and $\bar{x}' = (x_1, \dots, x_p, x_{p+1})$. Let I be the r_p -component of \bar{a}^p that contains i , and let J be the r_{p+1} -component of $\bar{a}^{p+1} = (a_1, \dots, a_{p+1})$ that contains i and $p+1$. Then $J \subseteq I \cup \{p+1\}$, because $r_p \geq 2r_{p+1}$. Let $\varphi(\bar{x}'_J)$ be the conjunction of all $\psi(x_1, \dots, x_{|J|}) \in \Phi^+(\sigma^{p+1}, |J|, q, q-p-1)$ such that $\mathcal{A}_I^p \models \psi[\bar{a}_I^{p+1}]$. Let

$$\chi(\bar{x}_I) := \exists x_{p+1} (\text{dist}(x_i, x_{p+1}) \leq r_{p+1} \wedge \varphi(\bar{x}'_J)).$$

Then χ is of q -rank at most $(q-p)$, and $\mathcal{A}_I^p \models \chi[\bar{a}_I]$. Hence by (II-b), $\mathcal{B}_I^p \models \chi[\bar{b}_I]$. Let b_{p+1} be a witness for the existential quantifier in χ . Duplicator chooses b_{p+1} as her answer. We let \mathcal{A}_J^{p+1} be the σ^{p+1} -reduct of \mathcal{A}_I^p , and for all other r_{p+1} -components J' of \bar{a}^{p+1} we let $\mathcal{A}_{J'}^{p+1}$ be the σ^{p+1} -reduct of $\mathcal{A}_{I'}^p$, where I' is the r_p -component of \bar{a}^p that contains J' . We define the structure $\mathcal{B}_{J'}^{p+1}$ similarly. Then (I) and (II) are satisfied for $p+1$.

Case 2: $r_{p+1} < \text{dist}^{\mathcal{A}}(a_j, a_{p+1})$ for all $j \in [p]$ and $\text{dist}^{\mathcal{A}}(a_i, a_{p+1}) \leq r_p - r_{p+1}$ for some $i \in [p]$.

Let I be the r_p -component of \bar{a}^p that contains i . Then $a_{p+1} \in A_I$ by assumption (II-a).

By assumption (II-b), Duplicator wins the $(q-p)$ -round EF_q^+ game on $(\mathcal{A}_I, \bar{a}^p, \mathcal{B}_I, \bar{b}^p)$. Let b_{p+1} be her answer if Spoiler plays a_{p+1} . Then $\text{dist}^{\mathcal{B}}(b_j, b_{p+1}) > r_{p+1}$ for all $j \in [p]$. This holds for $j \in I$ by (II-b) and for $j \notin I$ because $\text{dist}^{\mathcal{B}}(b_i, b_j) > r_p$. Thus (I) holds for $p+1$.

To see that (II) holds, note that $\{p+1\}$ is an r_{p+1} -component of \bar{a}^{p+1} . We let $\mathcal{A}_{\{p+1\}}^{p+1} := \mathcal{A}^{p+1}[\mathcal{X}(a_{p+1})]$ and $\mathcal{B}_{\{p+1\}}^{p+1} := \mathcal{B}^{p+1}[\mathcal{Y}(b_{p+1})]$. Then (II-a) holds, because \mathcal{X} and \mathcal{Y} are r -neighbourhood covers. (II-b) holds because a_{p+1} and b_{p+1} satisfy the same formulas $P_\varphi(x)$ for symbols $P_\varphi \in \sigma^p = \sigma \star^{q-p}(q, \ell)$.

Case 3: $\text{dist}^{\mathcal{A}}(a_i, a_{p+1}) > r_p - r_{p+1}$ for all $i \in [p]$.

Let Ψ be the set of all $\psi(x_1) \in \Phi^+(\sigma^{p+1}, 1, q, q-p-1)$ such that $\mathcal{A}^{p+1}[\mathcal{X}(a_{p+1})] \models \psi[a_{p+1}]$,

or equivalently, $\mathcal{A}^p \models P_\psi[a_{p+1}]$. Let C be the set of all $c \in A$ such that $\mathcal{A}^p \models \bigwedge_{\psi \in \Psi} P_\psi[c]$, and let D be the set of all $d \in B$ such that $\mathcal{B}^p \models \bigwedge_{\psi \in \Psi} P_\psi[d]$.

If there is a $d \in D$ such that $\text{dist}^{\mathcal{B}}(b_i, d) > r_{p+1}$ for all $i \in [p]$, then we can let $b_{p+1} := d$ and argue as in case 2.

So suppose for contradiction that for each $d \in D$ there is a $j(d) \in [p]$ such that $\text{dist}^{\mathcal{B}}(b_{j(d)}, d) \leq r_{p+1}$.

Claim 1. There are $s, t, m \in \mathbb{N}$ such that $2r_{p+1} \leq s \leq t - 4r_{p+1}$ and $t < r_p$ and $m \leq p$ and there are elements $c_1, \dots, c_m \in C$ with $\text{dist}^{\mathcal{A}}(c_i, c_j) > t$ for all $1 \leq i < j \leq m$, but there are no elements $c_1, \dots, c_{m+1} \in C$ with $\text{dist}^{\mathcal{A}}(c_i, c_j) > s$ for all $1 \leq i < j \leq m+1$.

Proof. We let $s^{(1)} := 2r_{p+1}$ and $m^{(1)}$ maximum such that there are $c_1, \dots, c_{m^{(1)}} \in C$ with $\text{dist}^{\mathcal{A}}(c_i, c_j) > s^{(1)}$ for all $1 \leq i < j \leq m^{(1)}$.

Suppose for contradiction that $m^{(1)} > p$. As \mathcal{A}^* and \mathcal{B}^* satisfy the same (r, q) -independence sentences and $2r_{p+1} \leq r$ and $p+1 \leq q$, there are $d_1, \dots, d_{p+1} \in D$ such that $\text{dist}^{\mathcal{B}}(d_i, d_j) > 2r_{p+1}$. By the pigeonhole principle, there are distinct $i, i' \in [p+1]$ such that $j(d_i) = j(d_{i'}) =: j$ and thus $\text{dist}^{\mathcal{B}}(d_i, b_j) \leq r_{p+1}$ and $\text{dist}^{\mathcal{B}}(d_{i'}, b_j) \leq r_{p+1}$. This implies $\text{dist}^{\mathcal{B}}(d_i, d_{i'}) \leq 2r_{p+1}$, which is a contradiction. Hence $m^{(1)} \leq p$.

Now suppose that $s^{(h)}, m^{(h)}$ are defined for some $h \geq 1$ in such a way that there are $c_1, \dots, c_{m^{(h)}} \in C$ with $\text{dist}^{\mathcal{A}}(c_i, c_j) > s^{(h)}$, and $m^{(h)}$ is maximum with this property. Let $s^{(h+1)} := s^{(h)} + 4r_{p+1}$. Let $m^{(h+1)}$ be maximum such that there are $c'_1, \dots, c'_{m^{(h+1)}} \in C$ with $\text{dist}^{\mathcal{A}}(c'_i, c'_j) > s^{(h+1)}$. Then $m^{(h+1)} \leq m^{(h)}$. We continue this construction until $m^{(h+1)} = m^{(h)}$. We let $s := s^{(h)}$ and $t := s^{(h+1)}$.

Since $p \geq m^{(1)} > m^{(2)} > \dots > m^{(h)} = m^{(h+1)} \geq 1$, we have $h \leq p$ and thus

$$t = s^{(1)} + 4r_{p+1}h = 4r_{p+1}(h + 1/2) < 4r_{p+1}(p+1) \leq 4qr_{p+1} = r_p. \quad \lrcorner$$

In the following, we let m, s, t be as in the claim. As \mathcal{A}^* and \mathcal{B}^* satisfy the same (r, q) -independence sentences and $t \leq r_p \leq r$ and $m \leq p \leq q$, there are $d_1, \dots, d_m \in D$ such that $\text{dist}^{\mathcal{A}}(d_i, d_j) > t$. As $\text{dist}(d_i, d_j) > t \geq 2r_{p+1}$, we have $j(d_i) \neq j(d_{i'})$ for all distinct $i, i' \in [m]$. Without loss of generality we may assume that for every $i \in [m]$ we have $j(d_i) = i$. Then $\text{dist}^{\mathcal{B}}(d_i, b_i) \leq r_{p+1}$ and hence $\text{dist}^{\mathcal{B}}(b_i, b_{i'}) > t - 2r_{p+1}$ for distinct $i, i' \in [m]$. As $t - 2r_{p+1} < r_p$, it follows from (II-a) that

$$\text{dist}^{\mathcal{A}}(a_i, a_{i'}) > t - 2r_{p+1}$$

for distinct $i, i' \in [m]$.

Claim 2. For every $i \in m$ there is a $c_i \in C$ such that $\text{dist}^{\mathcal{A}}(a_i, c_i) \leq r_{p+1}$.

Proof. Let $i \in [m]$, let I be the r_p -component of \bar{b} that contains i . Let

$$\chi(\bar{x}_I) := \exists y \left(\text{dist}(x_i, y) \leq r_{p+1} \wedge \bigwedge_{\psi \in \Psi} P_\psi(y) \right).$$

Then $\mathcal{B}_I^p \models \chi(\bar{b}_I)$, because d_i can be taken as witnesses for the existential quantifier. Moreover, $\chi(\bar{x}_I)$ has q -rank at most $(q-p)$, and thus $\mathcal{A}_I^p \models \chi(\bar{a}_I)$. This implies that there is a $c_i \in C$ such that $\text{dist}^{\mathcal{A}}(a_i, c_i) \leq r_{p+1}$. \lrcorner

Let $c_1, \dots, c_m \in C$ as in Claim 2. Then for distinct $i, i' \in [m]$ we have

$$\text{dist}^{\mathcal{A}}(c_i, c_{i'}) \geq \text{dist}^{\mathcal{A}}(a_i, a_{i'}) - 2r_{p+1} > t - 4r_{p+1} \geq s.$$

Moreover,

$$\text{dist}^{\mathcal{A}}(c_i, a_{p+1}) \geq \text{dist}^{\mathcal{A}}(a_i, a_{p+1}) - r_{p+1} > r_p - 2r_{p+1} \geq s.$$

Thus $c_1, \dots, c_m, a_{p+1} \in C$ have mutual distance greater than s , which contradicts Claim 1. \square

To prove Theorem 7.1, it will be convenient to use the language of *types*. A (k, q, ℓ) -type of signature σ is a subset $\mathbf{t} \subseteq \Phi^+(\sigma, k, q, \ell)$, and we let $\mathfrak{T}(\sigma, k, q, \ell) := 2^{\Phi^+(\sigma, k, q, \ell)}$ be the set of all (k, q, ℓ) -type of signature σ . Note that for each σ -structure \mathcal{A} and k -tuple $\bar{a} \in A^k$, there is a unique $\mathbf{t}_{k,q,\ell}(\mathcal{A}, \bar{a}) \in \mathfrak{T}(\sigma, k, q, \ell)$, which we call the (k, q, ℓ) -type of \mathcal{A}, \bar{a} , such that $\mathcal{A} \models \varphi[\bar{a}]$ for all $\varphi(\bar{x}) \in \mathbf{t}_{k,q,\ell}(\mathcal{A}, \bar{a})$ and $\mathcal{A} \models \neg\varphi[\bar{a}]$ for all $\varphi(\bar{x}) \in \Phi^+(\sigma, k, q, \ell) \setminus \mathbf{t}_{k,q,\ell}(\mathcal{A}, \bar{a})$. It is sometimes convenient to identify a type $\mathbf{t} \in \mathfrak{T}(\sigma, k, q, \ell)$ with the formula $\bigwedge_{\varphi(\bar{x}) \in \mathbf{t}} \varphi(\bar{x}) \wedge \bigwedge_{\varphi(\bar{x}) \in \Phi^+(\sigma, k, q, \ell) \setminus \mathbf{t}} \neg\varphi(\bar{x})$ and use notations such as $\mathbf{t}(\bar{y})$ for the formula obtained from $\mathbf{t} = \mathbf{t}(\bar{x})$ by substituting the variables in \bar{y} for those in \bar{x} and write $\mathcal{A} \models \mathbf{t}[\bar{a}]$ instead of $\mathbf{t} = \mathbf{t}_{k,q,\ell}(\mathcal{A}, \bar{a})$.

Let $\Xi(\sigma, r, k)$ be the set of all (r, k) -independence sentences of signature σ . An (r, k) -independence type of signature σ is a subset $\mathbf{i} \subseteq \Xi(\sigma, r, k)$, and we let $\mathfrak{I}(\sigma, r, k) := 2^{\Xi(\sigma, r, k)}$ be the set of all (r, k) -independence types of signature σ . We identify $\mathbf{i} \in \mathfrak{I}(\sigma, r, k)$ with the sentence $\bigwedge_{\xi \in \mathbf{i}} \xi \wedge \bigwedge_{\xi \in \Xi(\sigma, r, k) \setminus \mathbf{i}} \neg\xi$. Then for each σ -structure \mathcal{A} there is a unique $\mathbf{i}_{r,k}(\mathcal{A}) \in \mathfrak{I}(\sigma, r, k)$, which we call the (r, k) -independence type of \mathcal{A} , such that $\mathcal{A} \models \mathbf{i}_{r,k}(\mathcal{A})$.

Proof of Theorem 7.1. We fix a graph $G \in \mathcal{G}_k$ and let I_1, \dots, I_n be the connected components of G . For every $j \in [n]$, let $k_j := |I_j|$ and $\mathfrak{T}_j := \mathfrak{T}(\sigma^*, k_j, q, \ell)$. Furthermore, let $\mathfrak{I} := \mathfrak{I}(\sigma^*, r, q)$. A *combined type* is a tuple

$$\mathbf{c} = (\mathbf{i}, \mathbf{t}_1, \dots, \mathbf{t}_n) \in \mathfrak{I} \times \mathfrak{T}_1 \times \dots \times \mathfrak{T}_n,$$

and we let \mathfrak{C} be the set of all combined types.

Let \mathcal{A} be a σ -structure, \mathcal{X} a kr -neighbourhood cover of \mathcal{A} , and $\bar{a} \in A^k$ with $G_{\bar{a},r} = G$. Let $\mathcal{A}^* := \mathcal{A} \star_{\mathcal{X}}^{\ell}(q, \ell)$. We say that $(\mathcal{A}, \mathcal{X}, \bar{a})$ *satisfies* a combined type $\mathbf{c} = (\mathbf{i}, \mathbf{t}_1, \dots, \mathbf{t}_n) \in \mathfrak{C}$ if $\mathcal{A}^* \models \mathbf{i}$ and there are $X_1, \dots, X_n \in \mathcal{X}$ such that X_j r -covers \bar{a}_{I_j} and $\mathcal{A}^*[X_j] \models \mathbf{t}_j[\bar{a}_{I_j}]$ for all $j \in [n]$.

Claim 1. $(\mathcal{A}, \mathcal{X}, \bar{a})$ satisfies some combined type $\mathbf{c} \in \mathfrak{C}$.

Proof. We let $\mathbf{i} := \mathbf{i}_{r,q}(\mathcal{A}^*)$. For each $j \in [n]$, say, with $\bar{a}_{I_j} = (a_{i_1}, \dots, a_{i_{k_j}})$, we let $X_j := \mathcal{X}(a_{i_1})$. Since \bar{a}_{i_j} is connected we have $\{a_{i_1}, \dots, a_{i_{k_j}}\} \subseteq N_{(k-1)r}^{\mathcal{A}}(a_{i_1})$ and thus

$$N_r^{\mathcal{A}}(a_{i_1}, \dots, a_{i_{k_j}}) \subseteq N_{kr}^{\mathcal{A}}(a_{i_1}) \subseteq X_j,$$

because \mathcal{X} is a kr -neighbourhood cover. We let $\mathbf{t}_j := \mathbf{t}_{k,q,\ell}(\mathcal{A}^*[X_j], \bar{a}_{I_j})$ and $\mathbf{c} = (\mathbf{i}, \mathbf{t}_1, \dots, \mathbf{t}_n)$. Then $(\mathcal{A}, \mathcal{X}, \bar{a})$ satisfies \mathbf{c} . \lrcorner

Observe that if there are $\mathbf{c} = (\mathbf{i}, \mathbf{t}_1, \dots, \mathbf{t}_n) \in \mathfrak{C}$ and $\mathbf{c}' = (\mathbf{i}', \mathbf{t}'_1, \dots, \mathbf{t}'_n) \in \mathfrak{C}$ such that $(\mathcal{A}, \mathcal{X}, \bar{a})$ satisfies both \mathbf{c} and \mathbf{c}' , then $\mathbf{i} = \mathbf{i}'$, and $(\mathcal{A}, \mathcal{X}, \bar{a})$ also satisfies all $\mathbf{c}'' = (\mathbf{i}, \mathbf{t}''_1, \dots, \mathbf{t}''_n)$ such that $\mathbf{t}''_i \in \{\mathbf{t}_i, \mathbf{t}'_i\}$ for all i . This implies that there are an $\mathbf{i}(\mathcal{A}, \mathcal{X}, \bar{a}) \in \mathfrak{I}$ and sets $\mathfrak{T}^1(\mathcal{A}, \mathcal{X}, \bar{a}) \subseteq \mathfrak{T}_1, \dots, \mathfrak{T}^n(\mathcal{A}, \mathcal{X}, \bar{a}) \subseteq \mathfrak{T}_n$ such that $(\mathcal{A}, \mathcal{X}, \bar{a})$ satisfies a combined type \mathbf{c} if and only if

$$\mathbf{c} \in \underbrace{\{\mathbf{i}(\mathcal{A}, \mathcal{X}, \bar{a})\} \times \mathfrak{T}^1(\mathcal{A}, \mathcal{X}, \bar{a}) \times \dots \times \mathfrak{T}^n(\mathcal{A}, \mathcal{X}, \bar{a})}_{=: \mathfrak{C}(\mathcal{A}, \mathcal{X}, \bar{a})}.$$

Claim 2. Let $\psi(\bar{x}_{I_j}) := \bigvee_{\mathbf{t} \in \mathfrak{T}^j(\mathcal{A}, \mathcal{X}, \bar{a})} \mathbf{t}$. Then for all $X, X' \in \mathcal{X}$ that r -cover \bar{a}_{I_j} we have

$$\mathcal{A}^*[X] \models \psi(\bar{x}_{I_j}) \iff \mathcal{A}^*[X'] \models \psi(\bar{x}_{I_j}).$$

Proof. Straightforward. \lrcorner

Now let \mathcal{B} be another σ -structure, \mathcal{Y} a kr -neighbourhood cover of \mathcal{B} , and $\bar{b} \in B^k$ with $G_{\bar{b},r} = G$. Then it follows from Lemma 7.3 that if both $(\mathcal{A}, \mathcal{X}, \bar{a})$ and $(\mathcal{B}, \mathcal{Y}, \bar{b})$ satisfy some combined type $\mathfrak{c} \in \mathfrak{C}$ then

$$\mathcal{A} \models \varphi[\bar{a}] \iff \mathcal{B} \models \varphi[\bar{b}]. \quad (2)$$

Let $\mathfrak{C}_1^\varphi, \dots, \mathfrak{C}_m^\varphi$ be a list of all sets $\mathfrak{C}(\mathcal{A}, \mathcal{X}, \bar{a})$ for some σ -structure \mathcal{A} , kr -neighbourhood cover \mathcal{X} of \mathcal{A} , and $\bar{a} \in A^k$ such that $G_{\bar{a},r} = G$ and $\mathcal{A} \models \varphi[\bar{a}]$.

Claim 3. For all σ -structures \mathcal{A} , all kr -neighbourhood covers \mathcal{X} of \mathcal{A} , and all $\bar{a} \in A^k$ with $G_{\bar{a},r} = G$, we have

$$\mathcal{A} \models \varphi[\bar{a}] \iff (\mathcal{A}, \mathcal{X}, \bar{a}) \text{ satisfies some } \mathfrak{c} \in \bigcup_{i=1}^m \mathfrak{C}_i^\varphi.$$

Proof. The implication “ \implies ” is immediate from Claim 1 and the definition of $\mathfrak{C}_1^\varphi, \dots, \mathfrak{C}_m^\varphi$, and the converse implication follows from (2). \square

We let $m_G := m$. Let $i \in [m]$ and suppose that $\mathfrak{C}_i^\varphi = \{\mathfrak{i}_i\} \times \mathfrak{T}_i^1 \times \dots \times \mathfrak{T}_i^n$. We let $\xi_G^i := \mathfrak{i}_i$, and $\psi_{G,I_j}^i(\bar{x}_{I_j}) := \bigvee_{\mathfrak{t} \in \mathfrak{T}_i^j} \mathfrak{t}$.

Now (1) and (2) follow from the construction and Claim 3, and (3) follows from Claim 2.

We omit the proof of (4), which is essentially the same as the corresponding part of the proof of the Rank Preserving Locality Theorem in [13]. \square

7.2 Rank-preserving term localisation

Let $\delta_{G,r}(\bar{y})$ be the $\text{FO}^+[\sigma]$ -formula obtained from the $\text{FO}[\sigma]$ -formula $\delta_{G,r}^\sigma(\bar{y})$ of Section 6 by replacing every subformula of the form $\text{dist}^\sigma(y_i, y_j) \leq r$ (resp., $> r$) by the “distance atom” $\text{dist}(y_i, y_j) \leq r$ (resp., its negation).

Next, we define a variant of the cl-terms of Section 6 that is based on neighbourhood covers. These “cover-cl-terms” are no counting terms of the logic $\text{FOC}_1(\mathbb{P})$; they are abstract objects that come with their own semantics.

Definition 7.4 (Cover-cl-Term). Let $r, m \geq 0, k \geq 1$. A *basic cover-cl-term* with parameters (r, k, m) and of signature σ is an object g of the form

$$\#(y_1, \dots, y_k) \cdot (\delta_{G,r}(y_1, \dots, y_k) \wedge \psi(y_1, \dots, y_k))$$

or an object $u(y_1)$ of the form

$$\#(y_2, \dots, y_k) \cdot (\delta_{G,r}(y_1, \dots, y_k) \wedge \psi(y_1, \dots, y_k))$$

where $\bar{y} = (y_1, \dots, y_k)$ is a tuple of k pairwise distinct variables, G is a *connected* graph in \mathcal{G}_k , and $\psi(y_1, \dots, y_k)$ is an $\text{FO}^+[\sigma]$ -formula such that the following is true for all σ -structures \mathcal{A} , all $\bar{a} \in A^k$ with $G_{\bar{a},r}^A = G$, all m -neighbourhood covers \mathcal{X} of \mathcal{A} , and all clusters X and X' of \mathcal{X} that r -cover \bar{a} :

$$\mathcal{A}[X] \models \psi[\bar{a}] \iff \mathcal{A}[X'] \models \psi[\bar{a}].$$

We say that g and $u(y_1)$ are of q -rank at most ℓ iff ψ is of q -rank at most ℓ .

Semantics: For a σ -structure \mathcal{A} and an m -neighbourhood cover \mathcal{X} of \mathcal{A} we let $g^{\mathcal{A}, \mathcal{X}}$ be the number of tuples $\bar{a} \in A^k$ such that $G_{\bar{a},r}^A = G$ (i.e., $\mathcal{A} \models \delta_{G,r}[\bar{a}]$) and $\mathcal{A}[X] \models \psi[\bar{a}]$ for some (and hence, all) clusters X of \mathcal{X} that r -cover \bar{a} . Similarly, for $a_1 \in A$ we let $u^{\mathcal{A}, \mathcal{X}}[a_1]$ be the number of tuples $(a_2, \dots, a_k) \in A^{k-1}$ such that for $\bar{a} := (a_1, a_2, \dots, a_k)$ we have $G_{\bar{a},r}^A = G$ and $\mathcal{A}[X] \models \psi[\bar{a}]$ for some (hence, all) clusters X of \mathcal{X} that r -cover \bar{a} .

A *cover-cl-term with parameters* (r, k, m) is built from integers and basic cover-cl-terms with parameters (r', k', m') with $r' \leq r, k' \leq k, m' \leq m$ by using rule (7) of Definition 3.1. The cover-cl-term is of q -rank at most ℓ if all its basic cover-cl-terms are of q -rank at most ℓ .

We generalise the notion to graphs $G \in \mathcal{G}_k$ that are not connected.

Definition 7.5 (Cover-Term). Let $r, m \geq 0$, $k \geq 1$. A *cover-term* with parameters (r, k, m) and of signature σ is of the form

$$\begin{aligned} g &:= \#(y_1, \dots, y_k) \cdot (\delta_{G,r}(\bar{y}) \wedge \bigwedge_{I \in C} \psi_I(\bar{y}_I)) \quad \text{or} \\ u(y_1) &:= \#(y_2, \dots, y_k) \cdot (\delta_{G,r}(\bar{y}) \wedge \bigwedge_{I \in C} \psi_I(\bar{y}_I)) \end{aligned}$$

where $k \geq 1$, $\bar{y} = (y_1, \dots, y_k)$ is a tuple of k pairwise distinct variables, $G \in \mathcal{G}_k$, C is the set consisting of all connected components I of G , and for every $I \in C$, $\psi_I(\bar{y}_I)$ is an $\text{FO}^+[\sigma]$ -formula such that for all σ -structures \mathcal{A} , all $\bar{a} = (a_1, \dots, a_k) \in A^k$ with $G_{\bar{a},r}^{\mathcal{A}} = G[I]$, all m -neighbourhood covers \mathcal{X} of \mathcal{A} , and all clusters X and X' of \mathcal{X} that r -cover \bar{a}_I we have

$$\mathcal{A}[X] \models \psi_I[\bar{a}_I] \iff \mathcal{A}[X'] \models \psi_I[\bar{a}_I]. \quad (**)$$

Semantics: For a σ -structure \mathcal{A} and an m -neighbourhood cover \mathcal{X} of \mathcal{A} we let $g^{\mathcal{A},\mathcal{X}}$ be the number of tuples $\bar{a} = (a_1, \dots, a_k) \in A^k$ such that $G_{\bar{a},r} = G$ and for all $I \in C$, $\mathcal{A}[X] \models \psi_I[\bar{a}_I]$ for some (and hence, all) clusters X of \mathcal{X} that r -cover \bar{a}_I . Furthermore, for every $a_1 \in A$ we let $u^{\mathcal{A},\mathcal{X}}[a_1]$ be the number of tuples $(a_2, \dots, a_k) \in A^{k-1}$ such that for $\bar{a} := (a_1, a_2, \dots, a_k)$ we have $G_{\bar{a},r} = G$ and for all $I \in C$, $\mathcal{A}[X] \models \psi_I[\bar{a}_I]$ for some (and hence, all) clusters X of \mathcal{X} that r -cover \bar{a}_I .

Lemma 7.6. Let σ be a relational signature, let $r \geq 0$, $k \geq 1$, $m \geq 0$, and consider cover-terms

$$\begin{aligned} g &:= \#(y_1, \dots, y_k) \cdot (\delta_{G,r}(\bar{y}) \wedge \bigwedge_{I \in C} \psi_I(\bar{y}_I)) \\ u(y_1) &:= \#(y_2, \dots, y_k) \cdot (\delta_{G,r}(\bar{y}) \wedge \bigwedge_{I \in C} \psi_I(\bar{y}_I)) \end{aligned}$$

with parameters (r, k, m) .

There exists a ground cover-cl-term \hat{g} and a unary cover-cl-term $\hat{u}(y_1)$, both with parameters (r, k, m) , such that $\hat{g}^{\mathcal{A},\mathcal{X}} = g^{\mathcal{A},\mathcal{X}}$ and $\hat{u}^{\mathcal{A},\mathcal{X}}[a_1] = u^{\mathcal{A},\mathcal{X}}[a_1]$, for every σ -structure \mathcal{A} , every m -neighbourhood cover \mathcal{X} of \mathcal{A} , and every $a_1 \in A$.

If ψ_I has q -rank at most ℓ for each $I \in C$, then also \hat{g} and $\hat{u}(y_1)$ have q -rank at most ℓ .

Furthermore, there is an algorithm which upon input of (r, k, m) , G , and $(\psi_I)_{I \in C}$ constructs \hat{g} and $\hat{u}(y_1)$.

Proof. We proceed by induction on the number $c := |C|$ of connected components of G . Precisely, we show that the following statement $(*)_c$ is true for every $c \in \mathbb{N}_{\geq 1}$.

$(*)_c$: Let $k \geq c$. Let $G \in \mathcal{G}_k$ consist of at most c connected components, and let C be the set consisting of all connected components I of G . Let $\bar{y} = (y_1, \dots, y_k)$ be a tuple of k pairwise distinct variables. Let $r, m \geq 0$. For every $I \in C$ let $\psi_I(\bar{y}_I)$ be an $\text{FO}^+[\sigma]$ -formula such that for all σ -structures \mathcal{A} , all $\bar{a} = (a_1, \dots, a_k) \in A^k$ with $G_{\bar{a},r}^{\mathcal{A}} = G[I]$, all m -neighbourhood covers \mathcal{X} of \mathcal{A} , and all clusters X and X' of \mathcal{X} that cover \bar{a}_I we have

$$\mathcal{A}[X] \models \psi_I[\bar{a}_I] \iff \mathcal{A}[X'] \models \psi_I[\bar{a}_I].$$

Then, for the cover-terms g and $u(y_1)$ (as stated in the lemma) there are cover-cl-terms \hat{g} and $\hat{u}(y_1)$ with parameters (r, k, m) such that $\hat{g}^{\mathcal{A},\mathcal{X}} = g^{\mathcal{A},\mathcal{X}}$ and $\hat{u}^{\mathcal{A},\mathcal{X}}[a_1] = u^{\mathcal{A},\mathcal{X}}[a_1]$ holds for every σ -structure \mathcal{A} , every m -neighbourhood cover \mathcal{X} of \mathcal{A} and all $a_1 \in A$.

The induction base for $c = 1$ is trivial, since g and $u(y_1)$ are basic cover-cl-terms.

For the induction step from c to $c+1$, consider some $k \geq c+1$ and a graph $G = (V, E) \in \mathcal{G}_k$ that has $c+1$ connected components. V' be the connected component of G that contains the node 1 and let $V'' := V \setminus V'$.

Let $G' := G[V']$ and $G'' := G[V'']$ be the induced subgraphs of G on V' and V'' , respectively. Clearly, G is the disjoint union of G' and G'' , G' is connected, G'' has c connected components, and $C'' := C \setminus \{V'\}$ is the set of all connected components of G'' .

To keep notation simple, we assume (without loss of generality) that $V' = \{1, \dots, k'\}$ and $V'' = \{k'+1, \dots, k\}$ for some k' with $1 \leq k' < k$. For any tuple $\bar{z} = (z_1, \dots, z_k)$ we let $\bar{z}' := (z_1, \dots, z_{k'})$ and $\bar{z}'' := (z_{k'+1}, \dots, z_k)$.

Now consider numbers $r, m \geq 0$ and formulas $\psi_I(\bar{y}_I)$, for each $I \in C$, as in $(*)_{c+1}$'s assumption.

For every σ -structure \mathcal{A} and every m -neighbourhood cover \mathcal{X} of \mathcal{A} we let

$$S^{\mathcal{A}, \mathcal{X}}$$

be the set of all tuples $\bar{a} = (a_1, \dots, a_k) \in A^k$ such that $G_{\bar{a}, r}^{\mathcal{A}} = G$ and where for each $I \in C$ we have $\mathcal{A}[X] \models \psi_I[\bar{a}_I]$ for some (and hence, every) cluster X of \mathcal{X} that covers \bar{a}_I . Clearly, $g^{\mathcal{A}, \mathcal{X}} = |S^{\mathcal{A}, \mathcal{X}}|$ for the ground cover-term

$$g := \#(y_1, \dots, y_k) \cdot (\delta_{G, r}(\bar{y}) \wedge \bigwedge_{I \in C} \psi_I(\bar{y}_I)).$$

Similarly, we let

$$S_1^{\mathcal{A}, \mathcal{X}}$$

be the set of all tuples $\bar{a}' = (a_1, \dots, a_{k'}) \in A^{k'}$ such that $G_{\bar{a}', r}^{\mathcal{A}} = G'$ and where $\mathcal{A}[X] \models \psi_{V'}[\bar{a}']$ for some (and hence, every) cluster X of \mathcal{X} that covers \bar{a}' . Clearly, $g_1^{\mathcal{A}, \mathcal{X}} = |S_1^{\mathcal{A}, \mathcal{X}}|$ for the *basic cover-cl-term*

$$\hat{g}_1 := \#(y_1, \dots, y_{k'}) \cdot (\delta_{G', r}(y_1, \dots, y_{k'}) \wedge \psi_{V'}(y_1, \dots, y_{k'})).$$

Furthermore, we let

$$S_2^{\mathcal{A}, \mathcal{X}}$$

be the set of all tuples $\bar{a}'' = (a_{k'+1}, \dots, a_k) \in A^{k-k'}$ such that $G_{\bar{a}'', r}^{\mathcal{A}} = G''$ and where for each $I \in C''$ we have $\mathcal{A}[X] \models \psi_I[\bar{a}''_I]$ for some (and hence, every) cluster X of \mathcal{X} that covers \bar{a}''_I . Clearly, $g_2^{\mathcal{A}, \mathcal{X}} = |S_2^{\mathcal{A}, \mathcal{X}}|$ for the ground cover-term

$$g_2 := \#(y_{k'+1}, \dots, y_k) \cdot (\delta_{G'', r}(y_{k'+1}, \dots, y_k) \wedge \bigwedge_{I \in C''} \psi_I(\bar{y}_I)).$$

By the induction hypothesis we know that $(*)_c$ holds. Hence, there is a *cover-cl-term* \hat{g}_2 such that $\hat{g}_2^{\mathcal{A}, \mathcal{X}} = g_2^{\mathcal{A}, \mathcal{X}} = |S_2^{\mathcal{A}, \mathcal{X}}|$ is true for all σ -structures \mathcal{A} and all m -neighbourhood covers \mathcal{X} of \mathcal{A} .

Note that for every σ -structure \mathcal{A} and every m -neighbourhood cover \mathcal{X} of \mathcal{A} we have

$$S^{\mathcal{A}, \mathcal{X}} = (S_1^{\mathcal{A}, \mathcal{X}} \times S_2^{\mathcal{A}, \mathcal{X}}) \setminus T^{\mathcal{A}, \mathcal{X}},$$

where

$$T^{\mathcal{A}, \mathcal{X}}$$

is the set of all tuples $\bar{a} = (a_1, \dots, a_k) \in A^k$ such that $\bar{a}' \in S_1^{\mathcal{A}, \mathcal{X}}$, $\bar{a}'' \in S_2^{\mathcal{A}, \mathcal{X}}$, and there are $j' \in \{1, \dots, k'\}$ and $j'' \in \{k'+1, \dots, k\}$ such that $\text{dist}^{\mathcal{A}}(a_{j'}, a_{j''}) \leq r$.

Let \mathcal{H} be the set of all graphs $H \in \mathcal{G}_k$ with $H \neq G$, but $H[V'] = G'$ and $H[V''] = G''$. Clearly, every $H \in \mathcal{H}$ has at most c connected components. Furthermore, it is straightforward

to see that for every σ -structure \mathcal{A} and every m -neighbourhood cover \mathcal{X} of \mathcal{A} , the set $T^{\mathcal{A},\mathcal{X}}$ is the disjoint union of the sets

$$T_H^{\mathcal{A},\mathcal{X}} := \{ \bar{a} \in A^k : \bar{a}' \in S_1^{\mathcal{A},\mathcal{X}} \text{ and } \bar{a}'' \in S_2^{\mathcal{A},\mathcal{X}} \text{ and } G_{\bar{a},r}^{\mathcal{A}} = H \}$$

for all $H \in \mathcal{H}$.

Clearly,

$$g^{\mathcal{A},\mathcal{X}} = |S^{\mathcal{A},\mathcal{X}}| = |S_1^{\mathcal{A},\mathcal{X}}| \cdot |S_2^{\mathcal{A},\mathcal{X}}| - \sum_{H \in \mathcal{H}} |T_H^{\mathcal{A},\mathcal{X}}|;$$

and this holds for every σ -structure \mathcal{A} and every m -neighbourhood cover \mathcal{X} of \mathcal{A} .

To finish the proof of the lemma's statement concerning g , it therefore suffices to construct for each $H \in \mathcal{H}$ a cover-cl-term \hat{g}_H such that $\hat{g}_H^{\mathcal{A},\mathcal{X}} = |T_H^{\mathcal{A},\mathcal{X}}|$ for every σ -structure \mathcal{A} and every m -neighbourhood cover \mathcal{X} of \mathcal{A} — afterwards, we are done by choosing

$$\hat{g} := \hat{g}_1 \cdot \hat{g}_2 - \sum_{H \in \mathcal{H}} \hat{g}_H.$$

Let us consider a fixed $H \in \mathcal{H}$. Note that every connected component of H is a union of one or more connected components of G . Let I_1, \dots, I_s be the connected components of H (for $s \leq c$). For each $j \in [s]$ let C_j be the subset of C such that $I_j = \bigcup_{I \in C_j} I$. W.l.o.g. let $V' \in C_1$.

For each $j \in [s]$ let

$$\psi_{I_j}^H(\bar{y}_{I_j}) := \bigwedge_{I \in C_j} \psi_I(\bar{y}_I).$$

It is not difficult to verify that for all $j \in [s]$, all σ -structures \mathcal{A} , all $\bar{a} = (a_1, \dots, a_k) \in A^k$ with $G_{\bar{a}_{I_j},r}^{\mathcal{A}} = H[I_j]$, all m -neighbourhood covers \mathcal{X} of \mathcal{A} , and all clusters X and X' of \mathcal{X} that r -cover \bar{a}_{I_j} we have

$$\mathcal{A}[X] \models \psi_{I_j}^H[\bar{a}_{I_j}] \iff \mathcal{A}[X'] \models \psi_{I_j}^H[\bar{a}_{I_j}].$$

Hence, we can build the cover-term

$$g_H := \#(y_1, \dots, y_k) \cdot (\delta_{H,r}(\bar{y}) \wedge \bigwedge_{j \in [s]} \psi_{I_j}^H(\bar{y}_{I_j}))$$

and obtain by the induction hypothesis $(*)_c$ a ground cover-cl-term \hat{g}_H such that $\hat{g}_H^{\mathcal{A},\mathcal{X}} = g_H^{\mathcal{A},\mathcal{X}}$ for all σ -structures \mathcal{A} and all m -neighbourhood covers \mathcal{X} of \mathcal{A} . To finish the proof of the lemma's statement concerning g it remains to show that $g_H^{\mathcal{A},\mathcal{X}} = |T_H^{\mathcal{A},\mathcal{X}}|$.

By definition, $g_H^{\mathcal{A},\mathcal{X}} = |U_H^{\mathcal{A},\mathcal{X}}|$, for the set

$$U_H^{\mathcal{A},\mathcal{X}}$$

of all tuples $\bar{a} = (a_1, \dots, a_k) \in A^k$ such that $G_{\bar{a},r}^{\mathcal{A}} = H$ and for all $j \in [s]$, $\mathcal{A}[X] \models \psi_{I_j}^H[\bar{a}_{I_j}]$ for some (and hence, all) clusters X of \mathcal{X} that cover \bar{a}_{I_j} . It is straightforward to verify that $U_H^{\mathcal{A},\mathcal{X}} = T_H^{\mathcal{A},\mathcal{X}}$. This completes the proof of the lemma's statement concerning g .

The proof of the lemma's statement concerning $u(y_1)$ follows by an analogous reasoning. \square

By combining this lemma with Theorem 7.1 we obtain the following lemma.

Lemma 7.7 (Localisation Lemma). *Let $q, k \in \mathbb{N}$ with $k \leq q$, let $\ell := q - k$, $r := f_q(\ell)$, and let $\sigma^* := \sigma \star^\ell(q, \ell)$. Let $\varphi(\bar{x})$, where $\bar{x} = (x_1, \dots, x_k)$, be an $\text{FO}^+[\sigma]$ -formula of q -rank at most ℓ . Consider the terms*

$$\begin{aligned} g &:= \#(x_1, \dots, x_k) \cdot \varphi(x_1, \dots, x_k) \\ u(x_1) &:= \#(x_2, \dots, x_k) \cdot \varphi(x_1, \dots, x_k). \end{aligned}$$

There exists an $s \geq 0$ and (r, q) -independence sentences χ_1, \dots, χ_s of signature σ^* such that for every $J \subseteq [s]$ there are a ground cover-cl-term \hat{g}_J and a unary cover-cl-term $\hat{u}_J(x_1)$, both with parameters (r, k, kr) , of q -rank at most ℓ , and of signature σ^* , such that for every σ -structure \mathcal{A} and every kr -neighbourhood cover \mathcal{X} of \mathcal{A} there is exactly one $J \subseteq [s]$ with

$$\mathcal{A}^* \models \chi_J := \bigwedge_{j \in J} \chi_j \wedge \bigwedge_{j \in [s] \setminus J} \neg \chi_j$$

for $\mathcal{A}^* := \mathcal{A} \star_{\mathcal{X}}^{\ell}(q, \ell)$, and for this J we have $\hat{g}_J^{\mathcal{A}^*, \mathcal{X}} = g^{\mathcal{A}}$ and $\hat{u}_J^{\mathcal{A}^*, \mathcal{X}}[a] = u^{\mathcal{A}}[a]$ for every $a \in A$.

Furthermore, there is an algorithm which computes χ_1, \dots, χ_s and $(\hat{g}_J, \hat{u}_J(x_1))_{J \subseteq [s]}$ upon input of $q, k, \varphi(\bar{x})$.

Proof. We apply Theorem 7.1 to the formula $\varphi(\bar{x})$ and let χ_1, \dots, χ_s be the list of all (r, q) -independence sentences of signature σ^* that occur in one of the ξ_G^i for some $G \in \mathcal{G}_k$ and $i \in [m_G]$. For every $J \subseteq [s]$ let $S(J)$ be the set of all (G, i) with $G \in \mathcal{G}_k$ and $i \in [m_G]$ for which the propositional formula obtained from ξ_G^i by replacing every occurrence of χ_j by *true* if $j \in J$ and by *false* if $j \notin J$, evaluates to *true*.

For every $G \in \mathcal{G}_k$ we write $C(G)$ for the set of all connected components of G . For every (G, i) with $G \in \mathcal{G}_k$ and $i \in [m_G]$, consider the objects

$$g_{(G, i)} := \#(x_1, \dots, x_k) \cdot (\delta_{G, r}(\bar{x}) \wedge \bigwedge_{I \in C(G)} \psi_{G, I}^i(\bar{x}_I))$$

and

$$u_{(G, i)}(x_1) := \#(x_2, \dots, x_k) \cdot (\delta_{G, r}(\bar{x}) \wedge \bigwedge_{I \in C(G)} \psi_{G, I}^i(\bar{x}_I)).$$

From the statement of Theorem 7.1 we know that these objects are cover-terms of signature σ^* ; and using Lemma 7.6, we can translate these into cover-cl-terms $\hat{g}_{(G, i)}$ and $\hat{u}_{(G, i)}(x_1)$ with parameters (r, k, kr) and of q -rank at most ℓ .

For every σ -structure \mathcal{A} and every kr -neighbourhood cover \mathcal{X} of \mathcal{A} , there is a unique set $J \subseteq [s]$ such that $\mathcal{A}^* \models \chi_J$, for $\mathcal{A}^* := \mathcal{A} \star_{\mathcal{X}}^{\ell}(q, \ell)$. From the statement of Theorem 7.1 we obtain for

$$\hat{g}_J := \sum_{(G, i) \in S(J)} \hat{g}_{(G, i)}$$

and

$$\hat{u}_J(x_1) := \sum_{(G, i) \in S(J)} \hat{u}_{(G, i)}(x_1)$$

that $\hat{g}_J^{\mathcal{A}^*, \mathcal{X}} = g^{\mathcal{A}}$ and $\hat{u}_J^{\mathcal{A}^*, \mathcal{X}}[a] = u^{\mathcal{A}}[a]$ for every $a \in A$. \square

7.3 The Removal Lemma

Recall that by \bar{z}_I we denote the projection of a tuple $\bar{z} = (z_1, \dots, z_k)$ to the coordinates in $I \subseteq [k]$. We extend the notation by letting $\bar{z}_{\setminus I} := \bar{z}_{[k] \setminus I}$.

Let σ be a signature and let $r \in \mathbb{N}$. For every relation symbol $R \in \sigma$ of arity $k \geq 1$ and for every set $I \subseteq [k]$ we introduce a fresh $(k - |I|)$ -ary relation symbol \tilde{R}_I , and we let $\tilde{\sigma}$ be the set of all these relation symbols. We let $\tilde{\sigma}_r$ be the extension of $\tilde{\sigma}$ by fresh unary relation symbols S_i for all $i \in [r]$. For every σ -structure \mathcal{A} of order $|A| \geq 2$ and every $d \in A$, we let $\mathcal{A} \wr d$ be the $\tilde{\sigma}$ -structure with universe $A \setminus \{d\}$ and relations

$$\tilde{R}_I^{\mathcal{A} \wr d} := \{ \bar{a}_{\setminus I} : \bar{a} \in R^{\mathcal{A}} \text{ and } I = \{i \in [k] : a_i = d\} \}$$

for every k -ary $R \in \sigma$ and every $I \subseteq [k]$. Furthermore, we let $\mathcal{A} \wr_r d$ be the $\tilde{\sigma}_r$ -expansion of $\mathcal{A} \wr d$ in which each S_i is interpreted by the set of all $b \in A \setminus \{d\}$ such that $\text{dist}^{\mathcal{A}}(d, b) \leq i$. Note that (for fixed σ and r), we can compute $\mathcal{A} \wr_r d$ from \mathcal{A} and d in linear time.

Lemma 7.8 (Removal Lemma for Formulas). *Let $q, \ell \in \mathbb{N}$ and $r := f_q(\ell)$. Then for every $\text{FO}^+[\sigma]$ -formula $\varphi(\bar{x})$ of q -rank at most ℓ , where $\bar{x} = (x_1, \dots, x_k)$, and for every set $I \subseteq [k]$ there is an $\text{FO}^+[\tilde{\sigma}_r]$ -formula $\tilde{\varphi}_I(\bar{x}_{\setminus I})$ of q -rank at most ℓ such that for all σ -structures \mathcal{A} of order $|A| \geq 2$, all $d \in A$, and all $\bar{a} = (a_1, \dots, a_k) \in A^k$ such that $I = \{i \in [k] : a_i = d\}$, we have*

$$\mathcal{A} \models \varphi[\bar{a}] \iff \mathcal{A} \upharpoonright_r d \models \tilde{\varphi}_I[\bar{a}_{\setminus I}].$$

Furthermore, there is an algorithm that computes $\tilde{\varphi}_I$ from φ and I .

Proof. For the proof it will be convenient to consider sets V of variables instead of sets I of indices. When given a formula φ , we construct for every finite set V of variables a formula $\tilde{\varphi}_V$ with $\text{free}(\tilde{\varphi}_V) = \text{free}(\varphi) \setminus V$, such that for all σ -structures \mathcal{A} of order $|A| \geq 2$, all $d \in A$, and all assignments β in \mathcal{A} with $V \cap \text{free}(\varphi) = \{x \in \text{free}(\varphi) : \beta(x) = d\}$, we have

$$(\mathcal{A}, \beta) \models \varphi \iff (\tilde{\mathcal{A}}, \tilde{\beta}) \models \tilde{\varphi}_V, \quad (***)$$

for $\tilde{\mathcal{A}} := \mathcal{A} \upharpoonright_r d$ and for every assignment $\tilde{\beta}$ in $\mathcal{A} \upharpoonright_r d$ with $\tilde{\beta}(z) = \beta(z)$ for all variables $z \notin V$. Note that the lemma then follows by choosing $\tilde{\varphi}_I := \tilde{\varphi}_V$ for $V := \{x_i : i \in I\}$.

We construct $\tilde{\varphi}_V$ by induction on the shape of the given formula φ . The most interesting cases are the base cases.

- If φ is of the form $R(x_1, \dots, x_k)$ with $k = \text{ar}(R)$, then let $I := \{i \in [k] : x_i \in V\}$ and choose $\tilde{\varphi}_V := \tilde{R}_I(\bar{x}_{\setminus I})$.
- If φ is of the form $x_1 = x_2$, then

$$\tilde{\varphi}_V := \begin{cases} x_1 = x_2 & \text{if } \{x_1, x_2\} \cap V = \emptyset \\ \text{true} & \text{if } \{x_1, x_2\} \subseteq V \\ \text{false} & \text{otherwise.} \end{cases}$$

- If φ is of the form $\text{dist}(x_1, x_2) \leq i$, then

$$\tilde{\varphi}_V := \begin{cases} \text{true} & \text{if } x_1 \in V, x_2 \in V \\ S_i(x_2) & \text{if } x_1 \in V, x_2 \notin V \\ S_i(x_1) & \text{if } x_1 \notin V, x_2 \in V \\ \text{dist}(x_1, x_2) \leq i \vee \bigvee_{\substack{0 \leq i_1, i_2 \leq i, \\ i_1 + i_2 = i}} (S_{i_1}(x_1) \wedge S_{i_2}(x_2)) & \text{if } x_1 \notin V, x_2 \notin V. \end{cases}$$

The inductive step is straightforward:

- If φ is of the form $\neg\psi$, then $\tilde{\varphi}_V := \neg\tilde{\psi}_V$.
- If φ is of the form $(\psi \vee \chi)$, then $\tilde{\varphi}_V := (\tilde{\psi}_V \vee \tilde{\chi}_V)$.
- If φ is of the form $\exists x \psi$, then $\tilde{\varphi}_V := (\tilde{\psi}_{V \cup \{x\}} \vee \exists x \tilde{\psi}_{V \setminus \{x\}})$.

It is easy to see that $\tilde{\varphi}_V$ satisfies (***). This completes the proof of Lemma 7.8. \square

A *basic term* is a term $t(\bar{x})$ of the form $\# \bar{y}. \varphi(\bar{x}, \bar{y})$ for an FO^+ -formula $\varphi(\bar{x}, \bar{y})$. The q -rank of $t(\bar{x})$ is the q -rank of φ , and the *width* of $t(\bar{x})$ is $|\bar{x}| + |\bar{y}|$. Usually, we are only interested in ground basic terms, where $|\bar{x}| = 0$ and unary basic terms, where $|\bar{x}| = 1$.

Lemma 7.9 (Removal Lemma for Terms). *Let σ be a signature. Let $q, k \in \mathbb{N}$ with $k \leq q$, let $\ell := q - k$ and $r := f_q(\ell)$.*

- (a) For every ground basic term g of signature σ , width k , and q -rank at most ℓ there is a list $\hat{g}_1, \dots, \hat{g}_m$ of ground basic terms of signature $\tilde{\sigma}_r$, width at most k , and q -rank at most ℓ such that for all σ -structures \mathcal{A} of order $|A| \geq 2$ and all $d \in A$,

$$g^{\mathcal{A}} = \sum_{i=1}^m \hat{g}_i^{\mathcal{A}|_{r,d}}$$

Furthermore, there is an algorithm that, given g , computes $\hat{g}_1, \dots, \hat{g}_m$.

- (b) For every unary basic term $u(x)$ of signature σ , width k , and q -rank $\leq \ell$ there are a list $\hat{g}_1, \dots, \hat{g}_m$ of ground basic terms and a list $\hat{u}_1(x), \dots, \hat{u}_n(x)$ of unary basic terms, all of signature $\tilde{\sigma}_r$, width at most k , and q -rank at most ℓ , such that for all σ -structures \mathcal{A} of order $|A| \geq 2$ and all $a, d \in A$,

$$u^{\mathcal{A}}[a] = \begin{cases} \sum_{i=1}^m \hat{g}_i^{\mathcal{A}|_{r,d}} & \text{if } a = d, \\ \sum_{i=1}^n \hat{u}_i^{\mathcal{A}|_{r,d}}[a] & \text{if } a \neq d. \end{cases}$$

Furthermore, there is an algorithm that, given $u(x)$, computes $\hat{g}_1, \dots, \hat{g}_m, \hat{u}_1(x), \dots, \hat{u}_n(x)$.

Proof. We only prove assertion (b); the proof of (a) is similar. Let

$$u(x_1) := \#(x_1, \dots, x_k). \varphi(x_1, \dots, x_k),$$

where $\varphi(\bar{x})$ is an $\text{FO}^+[\sigma]$ -formula of q -rank at most ℓ . We apply Lemma 7.8 to $\varphi(\bar{x})$ and obtain formulas $\tilde{\varphi}_I(\bar{x}_{\setminus I})$ for all $I \subseteq [k]$. Let $\psi_1(\bar{x}_1), \dots, \psi_m(\bar{x}_m)$ be an enumeration of all formulas $\tilde{\varphi}_I(\bar{x}_{\setminus I})$ with $1 \in I$, and let $\vartheta_1(x_1, \bar{x}'_1), \dots, \vartheta_n(x_n, \bar{x}'_n)$ be an enumeration of all formulas $\tilde{\varphi}_I(\bar{x}_{\setminus I})$ with $1 \notin I$. We let $\hat{g}_i := \# \bar{x}_i. \psi_i(\bar{x}_i)$ and $\hat{u}_j(x_1) := \# \bar{x}'_j. \vartheta_j(x_1, \bar{x}'_j)$. \square

8 Nowhere dense structures

The concept of nowhere dense graph classes tries to capture the intuitive meaning of “sparse graphs” in a fairly general, yet still useful way. The original definition of nowhere dense classes (see [20]), which is relatively complicated, refers to the edge densities of “flat minors” of the graphs in the class. The definition has turned out to be very robust, and there are several seemingly unrelated characterisations of nowhere dense graph classes. Most useful for us is a characterisation in terms of a the so-called “splitter game” due to [13], which we use as our definition.

Let G be a graph and $\rho, r > 0$. The (ρ, r) -splitter game on G is played by two players called *Connector* and *Splitter* as follows. We let $G_0 := G$. In round $i+1$ of the game, Connector chooses an element $a_{i+1} \in V(G_i)$. Then Splitter chooses an element $b_{i+1} \in N_r^{G_i}(a_{i+1})$. If $N_r^{G_i}(a_{i+1}) \setminus \{b_{i+1}\} = \emptyset$, then Splitter wins the game. Otherwise, the game continues with

$$G_{i+1} := G_i[N_r^{G_i}(a_{i+1}) \setminus \{b_{i+1}\}].$$

If Splitter has not won after ρ rounds, Connector wins.

A *strategy* for Splitter is a function f that associates to every partial play $(a_1, b_1, \dots, a_i, b_i)$ with associated sequence G_0, \dots, G_i of graphs and move $a_{i+1} \in V(G_i)$ by Connector a $b_{i+1} \in N_r^{G_i}(a_{i+1})$. A strategy f is a *winning strategy* for Splitter in the (ρ, r) -splitter game on G if Splitter wins every play in which she follows the strategy f . If Splitter has a winning strategy, we say that she *wins* the (ρ, r) -splitter game on G .

For a class \mathcal{C} of graphs and a function $\lambda : \mathbb{N} \rightarrow \mathbb{N}$, we say that Splitter wins the λ -splitter game on \mathcal{C} if for every $r \in \mathbb{N}$ and every $G \in \mathcal{C}$ she wins the $(\lambda(r), r)$ -splitter game on G . A class \mathcal{C} of graphs is *nowhere dense* if there is a function $\lambda : \mathbb{N} \rightarrow \mathbb{N}$ such that Splitter wins the

λ -splitter game on \mathcal{C} . If λ is computable, the class \mathcal{C} is *effectively nowhere dense*. A class \mathcal{C} of structures is *(effectively) nowhere dense* if the class of Gaifman graphs of all structures in \mathcal{C} is (effectively) nowhere dense.

It follows from [13] that a class \mathcal{C} of graphs is nowhere dense (in the sense just defined) if and only if it is nowhere dense in the sense of [20].

It is easy to see that if Splitter wins the (ρ, r) -splitter game on a graph G , then she also wins it on all subgraphs of G . Thus if we close a nowhere dense class of graphs under taking subgraphs, the class remains nowhere dense.

Finally, we mention that for every nowhere dense class \mathcal{C} of graphs there is a function f such that for every $\epsilon > 0$ and every graph $G \in \mathcal{C}$, if $|V(G)| \geq f(\epsilon)$ then $\|G\| \leq |V(G)|^{1+\epsilon}$ (see [20]).

8.1 Sparse neighbourhood covers

Let us now turn to *sparse* neighbourhood covers of nowhere dense graphs. Let \mathcal{X} be an r -neighbourhood cover of a graph G (or of some structure \mathcal{A} with Gaifman graph G). The *radius* of \mathcal{X} is the least s such that all clusters of \mathcal{X} have radius at most s , that is, for every $X \in \mathcal{X}$ there is a $c \in X$ such that $X \subseteq N_s^{G[X]}(c)$. We call each such c an s -centre of X . In the following, an (r, s) -neighbourhood cover of G is an r -neighbourhood cover of radius at most s .

The *degree* of a vertex $a \in V(G)$ in a neighbourhood cover \mathcal{X} is the number of clusters $X \in \mathcal{X}$ such that $a \in X$. The *maximum degree* $\Delta(\mathcal{X})$ is the maximum of the degrees of all vertices $a \in V(G)$. Note that $\sum_{X \in \mathcal{X}} |X| \leq |V(G)| \cdot \Delta(\mathcal{X})$.

Theorem 8.1 ([13]). *Let \mathcal{C} be a nowhere dense class of graphs. Then there is a function f and an algorithm that, given an $\epsilon > 0$, an $r \in \mathbb{N}$, and a graph $G \in \mathcal{C}$ with $n := |V(G)| \geq f(r, \epsilon)$, computes an $(r, 2r)$ -neighbourhood cover of G of maximum degree at most n^ϵ in time $f(r, \epsilon) \cdot n^{1+\epsilon}$. Furthermore, if \mathcal{C} is effectively nowhere dense, then f is computable.*

We remark that the construction of [13] also yields, together with an $(r, 2r)$ -neighbourhood cover \mathcal{X} of G , a function $\text{cen} : \mathcal{X} \rightarrow V(G)$ that associates with each cluster $X \in \mathcal{X}$, a $2r$ -centre $\text{cen}(X)$ for X . Moreover, it is easy to see that for a given neighbourhood cover \mathcal{X} of G we can compute in linear time a data structure that associates with each $X \in \mathcal{X}$ the list of all $a \in V(G)$ with $\mathcal{X}(a) = X$.

8.2 The main algorithm

In this section, we complete the proof of Lemma 5.7. We fix a numerical predicate collection $(\mathbb{P}, \text{ar}, \llbracket \cdot \rrbracket)$ and a signature σ . Let \mathcal{C} be a nowhere dense class of structures, and let $\mathcal{G}_{\mathcal{C}}$ be the class of the Gaifman graphs of all structures in \mathcal{C} . Without loss of generality we may assume that $\mathcal{G}_{\mathcal{C}}$ is closed under taking subgraphs and that \mathcal{C} is the class of all structures whose Gaifman graph is in $\mathcal{G}_{\mathcal{C}}$. Let $\lambda : \mathbb{N} \rightarrow \mathbb{N}$ such that Splitter wins the λ -splitter game on $\mathcal{G}_{\mathcal{C}}$.

We need to design an algorithm with \mathbb{P} -oracle which receives as input an $\epsilon > 0$, a σ -structure \mathcal{A} from \mathcal{C} and an $\text{FOC}_1(\mathbb{P})[\sigma]$ -expression ξ which is either a sentence φ or a ground term t . The algorithm decides whether $\mathcal{A} \models \varphi$ and computes $t^{\mathcal{A}}$, respectively. The algorithm's running time is $f(p, \epsilon)n^{1+\epsilon}$, where $p := \|\xi\|$ is the size of the input expression and $n := |\mathcal{A}|$ is the order of the input structure.

Our algorithm is similar to the model-checking algorithm for FO-sentences on nowhere dense classes of graphs from [13]. The design and analysis of our algorithm relies on subroutines and results from [13]. However, we present a high level outline of the algorithm that should be accessible without knowledge of [13].

The Decomposition Theorem 6.10 reduces the evaluation of $\text{FOC}_1(\mathbb{P})[\sigma]$ -sentences and ground terms to the evaluation of first-order sentences and cl-terms over some signature $\tau \supseteq \sigma$ that extends σ by relation symbols of arity ≤ 1 . Note that every τ -expansion of a σ -structure \mathcal{A} has

the same Gaifman graph as \mathcal{A} and hence also belongs to \mathcal{C} . The evaluation of first-order sentences has been taken care of in [13]. The evaluation of cl-terms can further be reduced to basic cl-terms. In fact, it is not important that we have cl-terms; the important thing is that we have basic terms with at most one free variable.

So all that remains is the evaluation of basic terms, either ground terms g or unary terms $u(x_1)$. To simplify the notation, we just assume that these terms are in our original signature σ . Moreover, we focus on unary terms here; ground terms can be dealt with similarly.

Hence, the input of our algorithm is an $\epsilon > 0$, a σ -structure \mathcal{A} and a unary basic term $u(x_1)$ of width k and q -rank at most ℓ , where $q = k + \ell$. As usual, we let $r = f_q(\ell)$. Our algorithm is supposed to compute $u^{\mathcal{A}}[a]$ for all $a \in A$.

The algorithm proceeds in the following steps.

1. Let $\delta := \frac{\epsilon}{2\lambda(2kr)}$.
If $|A| < f(rk, \delta)$ for the function f of Theorem 8.1, evaluate t by brute force and stop.
Otherwise, compute a $(kr, 2kr)$ -neighbourhood cover \mathcal{X} of \mathcal{A} of maximum degree at most n^δ , where $n := |A|$. In addition, compute for each $X \in \mathcal{X}$ a $2kr$ -centre $\text{cen}(X)$ and the set of all elements $a \in A$ with $\mathcal{X}(a) = X$.
2. Let $\sigma^\star := \sigma \star^\ell(q, \ell)$. Compute $\mathcal{A}^\star := \mathcal{A} \star_{\mathcal{X}}^\ell(q, \ell)$, using the algorithm described in [13].
3. Applying the Localisation Lemma (Lemma 7.7), compute (r, q) -independence sentences χ_1, \dots, χ_s and cover-cl-terms $(\hat{g}_J, \hat{u}_J(x_1))_{J \subseteq [s]}$ with parameters (r, k, kr) and of q -rank at most ℓ such that the evaluation of $u(x_1)$ in \mathcal{A} reduces to the evaluation of these sentences and terms in $\mathcal{A}^\star, \mathcal{X}$.
4. Evaluate the independence sentences χ_1, \dots, χ_s in \mathcal{A}^\star using the algorithm of [13].
Obviously, there is exactly one set $J \subseteq [s]$ such that $\mathcal{A}^\star \models \chi_J$ for $\chi_J := \bigwedge_{j \in J} \chi_j \wedge \bigwedge_{j \in [s] \setminus J} \neg \chi_j$.
5. Compute $\hat{u}_J^{\mathcal{A}^\star, \mathcal{X}}[a] = u^{\mathcal{A}}[a]$ for every $a \in A$.

It remains to explain in detail how the last step is carried out. Consider a basic cover-cl-term $\hat{u}(x_1)$ that occurs in $\hat{u}_J(x_1)$ and is of the form

$$\hat{u}(x_1) := \#(x_2, \dots, x_{k'}) \cdot (\delta_{G, r'}(x_1, \dots, x_{k'}) \wedge \psi(x_1, \dots, x_{k'}))$$

for a connected graph $G \in \mathcal{G}_{k'}$, a $k' \leq k$, an $r' \leq r$, and an $\text{FO}^+[\sigma^\star]$ -formula $\psi(x_1, \dots, x_{k'})$ of q -rank at most ℓ .

Let $a \in A$ and $X := \mathcal{X}(a)$. As \mathcal{X} is a kr -neighbourhood cover and G is connected, X r -covers every tuple $\bar{a} = (a_1, \dots, a_{k'})$ such that $G_{\bar{a}, r'} = G$ and $a_1 = a$. Recall from Definition 7.4 that $\hat{u}^{\mathcal{A}^\star, \mathcal{X}}[a]$ is the number of tuples $\bar{a} \in A^{k'}$ such that $a_1 = a$ and $G_{\bar{a}, r'} = G$ and $\mathcal{A}^\star[X] \models \psi[\bar{a}]$.

To be able to compute this number efficiently, we introduce a fresh unary relation symbol Q and let \mathcal{B}_X be the $(\sigma^\star \cup \{Q\})$ -expansion of $\mathcal{A}^\star[X]$ where Q is interpreted by the set of all $a \in A$ such that $\mathcal{X}(a) = X$. Let

$$t(x_1) := \#(x_2, \dots, x_{k'}) \cdot (\delta_{G, r'}(x_1, \dots, x_{k'}) \wedge \psi(x_1, \dots, x_{k'}) \wedge Q(x_1)).$$

What our algorithm needs to do now is evaluate $t(x_1)$ in the structures \mathcal{B}_X , for all $X \in \mathcal{X}$. This is done in the following steps, which form the expanded version of step 5 of the algorithm.

5. For all $X \in \mathcal{X}$
 - a. Compute \mathcal{B}_X

- b. Let $c := \text{cen}(X)$, and let d be Splitter's answer if Connector plays c in the first round of the $(\lambda(2kr), 2kr)$ -splitter game on G_A . It is explained in [13] how d can be computed efficiently.
- c. Compute $\mathcal{B}' := \mathcal{B}_X \downarrow_r d$.
- d. Apply the Removal Lemma (Lemma 7.9) to the unary basic term $t(x_1)$ and recursively evaluate the resulting basic terms in \mathcal{B}' .
- e. For each $a \in Q^{\mathcal{B}_X}$, use the results of the recursive calls to compute $t^{\mathcal{B}_X}[a]$ according to the Removal Lemma.

The algorithm terminates with a recursion depth of at most $\lambda(2kr)$, because in the recursive call we only need to consider the $(\lambda(2kr)-1, 2kr)$ -splitter game.

Let us analyse the running time of the algorithm. We express the running time in terms of the order n of the input structure and the number ρ of rounds of the Splitter game. Initially, we have $\rho = \lambda(2kr)$. The dependence on the class \mathcal{C} , the signature σ , and the parameters k, ℓ goes into the constants; of course $\lambda(2kr)$ depends on \mathcal{C}, k, ℓ . If $n \leq n_0$ for some constant n_0 (depending on $\mathcal{C}, \sigma, k, \ell, \epsilon$) then the algorithm terminates in constant time in line 1. If $\rho = 1$, then Splitter wins the game in 1 round, which means that every connected component of G_A only consists of a single vertex. Thus either $|A| = 1$ and the algorithm terminates in line 1 in constant time or the algorithm makes n recursive calls and each of these recursive calls terminates in constant time. Thus we have the two basic equations $T(n, \rho) = O(1)$ if $n \leq n_0$, and $T(n, 1) = O(n)$ otherwise.

Suppose $n > n_0$ and $\rho > 1$. Lines 1–4 can be carried out in time $O(n^{1+\delta})$ (by Theorem 8.1 for line 1). To analyse the time spent on line 5, let $X \in \mathcal{X}$ be of size $n_X := |X|$. Lines 5.a–e can be carried out in time $O(\|\mathcal{B}_X\|) = O(\|\mathcal{A}[X]\|) = O(n_X^{1+\delta})$, because $\mathcal{A}[X]$ is from the nowhere dense class \mathcal{C} . The recursive calls in line 5.e require time $O(T(n_X, \rho-1))$. Thus the time spent on line 5 is $\sum_{X \in \mathcal{X}} O(T(n_X, \rho-1) + n_X^{1+\delta})$, and, recalling that $\delta = \epsilon/2\lambda(2kr)$ with $\rho = \lambda(2kr)$, we obtain a recurrence equation

$$T(n, \rho) = \sum_{X \in \mathcal{X}} O(T(n_X, \rho-1) + n_X^{1+\epsilon/2\rho}) + O(n^{1+\epsilon/2\rho}).$$

The same recurrence was obtained in [13], and it was shown there that it yields the desired running time $O(n^{1+\epsilon})$. This completes our description and analysis of the algorithm and hence the proof of Lemma 5.7.

9 Open questions

To conclude, let us point out some open questions.

(1) Can our approach be generalised to an extension of FO which, apart from COUNT, also supports further aggregate operations of SQL, such as SUM and AVG?

(2) Can our approach be generalised to support database updates? In [16] this was achieved for FOC(\mathbb{P}) on bounded degree classes. But for other classes, e.g., planar graphs or classes of bounded local tree-width (let alone nowhere dense classes), this is open even for FO.

(3) Can our approach be generalised to obtain an algorithm that enumerates the query result with constant-delay? In [23] such an algorithm was obtained for FO-queries on classes of locally bounded expansion. Can our machinery of Sections 6 and 7 help to generalise the result to nowhere dense classes?

References

- [1] S. Abiteboul, R. Hull, and V. Vianu. *Foundations of Databases*. Addison-Wesley, 1995.

- [2] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *J. Algorithms*, 12(2):308–340, 1991.
- [3] B. Courcelle. The monadic second-order logic of graphs. I. recognizable sets of finite graphs. *Inf. Comput.*, 85(1):12–75, 1990.
- [4] R. Downey, M. Fellows, and K. Regan. Descriptive complexity and the W-hierarchy. In P. Beame and S. Buss, editors, *Proof Complexity and Feasible Arithmetic*, volume 39 of *AMS-DIMACS Volume Series*, pages 119–134. AMS, 1998.
- [5] A. Durand and S. Mengel. Structural tractability of counting of solutions to conjunctive queries. In *Joint 2013 EDBT/ICDT Conferences, ICDT '13 Proceedings, Genoa, Italy, March 18-22, 2013*, pages 81–92, 2013.
- [6] Z. Dvořák, D. Král, and R. Thomas. Deciding first-order properties for sparse graphs. In *51th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2010, October 23-26, 2010, Las Vegas, Nevada, USA*, pages 133–142, 2010.
- [7] H.-D. Ebbinghaus and J. Flum. *Finite Model Theory*. Springer Verlag, 2nd edition, 1999.
- [8] S. Feferman and R. Vaught. The first order properties of algebraic systems. *Fund. Math.*, 47:57–103, 1959.
- [9] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer-Verlag, 2006.
- [10] M. Frick. Generalized model-checking over locally tree-decomposable classes. *Theory of Computing Systems*, 37(1):157–191, 2004.
- [11] M. Frick and M. Grohe. Deciding first-order properties of locally tree-decomposable structures. *J. ACM*, 48(6):1184–1206, 2001.
- [12] H. Gaifman. On local and non-local properties. In J. Stern, editor, *Proceedings of the Herbrand Symposium, Logic Colloquium '81*, pages 105–135. North Holland, 1982.
- [13] M. Grohe, S. Kreutzer, and S. Siebertz. Deciding first-order properties of nowhere dense graphs. *ArXiv (CoRR)*, arXiv:1311.3899v2 [cs.LO], 2014. Conference Version in Proceedings STOC'14.
- [14] W. Kazana and L. Segoufin. Enumeration of first-order queries on classes of structures with bounded expansion. In *Proceedings of the 32nd ACM Symposium on Principles of Database Systems*, pages 297–308, 2013.
- [15] S. Kreutzer. Algorithmic meta-theorems. In J. Esparza, C. Michaux, and C. Steinhorn, editors, *Finite and Algorithmic Model Theory*, chapter 5, page 177–270. Cambridge University Press, 2011.
- [16] D. Kuske and N. Schweikardt. First-order logic with counting: At least, *weak* Hanf normal forms always exist and can be computed! In *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS'17*, 2017. Full version available at CoRR abs/1703.01122.
- [17] L. Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004.
- [18] J. A. Makowsky. Algorithmic uses of the Feferman-Vaught theorem. *Ann. Pure Appl. Logic*, 126(1-3):159–213, 2004.

- [19] J. Nešetřil and P. Ossona de Mendez. First order properties on nowhere dense structures. *J. Symb. Log.*, 75(3):868–887, 2010.
- [20] J. Nešetřil and P. Ossona de Mendez. *Sparsity*. Springer-Verlag, 2012.
- [21] C. H. Papadimitriou and M. Yannakakis. On the complexity of database queries. *J. Comput. Syst. Sci.*, 58(3):407–427, 1999.
- [22] D. Seese. Linear time computable problems and first-order descriptions. *Mathematical Structures in Computer Science*, 6(6):505–526, 1996.
- [23] L. Segoufin and A. Vigny. Constant delay enumeration for FO queries over databases with local bounded expansion. In *20th International Conference on Database Theory, ICDT 2017, March 21-24, 2017, Venice, Italy*, pages 20:1–20:16, 2017.
- [24] M. Vardi. The complexity of relational query languages. In *Proceedings of the 14th ACM Symposium on Theory of Computing*, pages 137–146, 1982.