Deciding first-order properties for sparse graphs

Zdeněk Dvořák and Daniel Král'
Deparment of Applied Mathematics and
Institute for Theoretical Computer Science (ITI)
Faculty of Mathematics and Physics, Charles University
Malostranské náměstí 25, 118 00 Prague, Czech Republic.
Email: {rakdver,kral}@kam.mff.cuni.cz

Robin Thomas
School of Mathematics
Georgia Institute of Technology
686 Cherry St, Atlanta, GA 30332-0160, USA.
Email: thomas@math.gatech.edu

***Abstract*—We present a linear-time algorithm for deciding first-order logic (FOL) properties in classes of graphs with bounded expansion. Many natural classes of graphs have bounded expansion: graphs of bounded tree-width, all proper minor-closed classes of graphs, graphs of bounded degree, graphs with no subgraph isomorphic to a subdivision of a fixed graph, and graphs that can be drawn in a fixed surface in such a way that each edge crosses at most a constant number of other edges. We also develop an almost linear-time algorithm for deciding FOL properties in classes of graphs with locally bounded expansion; those include classes of graphs with locally bounded tree-width or locally excluding a minor.**

More generally, we design a dynamic data structure for graphs belonging to a fixed class of graphs of bounded expansion. After a linear-time initialization the data structure allows us to test an FOL property in constant time, and the data structure can be updated in constant time after addition/deletion of an edge, provided the list of possible edges to be added is known in advance and their addition results in a graph in the class. In addition, we design a dynamic data structure for testing existential properties or the existence of short paths between prescribed vertices in such classes of graphs. All our results also hold for relational structures and are based on the seminal result of Nešetřil and Ossona de Mendez on the existence of low tree-depth colorings.

***Keywords*-algorithmic metatheorems; graphs with bounded expansion; graphs with bounded degree; minor-closed classes of graphs; graphs with locally bounded tree-width**

I. Introduction

A celebrated theorem of Courcelle [1] states that for every integer $k \geq 1$ and every property Π definable in monadic second-order logic (MSOL) there is a linear-time algorithm to decide whether a graph of tree-width at most k satisfies Π. While the theorem itself is probably not useful in practice because of the large constants involved, it does provide an easily verifiable condition that a certain problem is efficiently solvable. Courcelle's result led to the developoment of a whole new area of algorithmic results, known as algorithmic meta-theorems, see the survey [18]. For specific problems there is often a more efficient implementation, for instance following the axiomatic approach of [27].

While the class of graphs of tree-width at most k is fairly large, it does not include some important graph classes, such as planar graphs or graphs of bounded degree. Courcelle's theorem cannot be extended to these classes unless P=NP, because testing 3-colorability is NP-hard for planar graphs of maximum degree at most four [13] and yet 3-colorability is expressible in monadic second order logic.

Thus in an attempt at enlarging the class of input graphs, we have to restrict the set of properties we want to test. One of the first results in this direction was a linear-time algorithm of Eppstein [8], [9] for testing the existence of a fixed subgraph in planar graphs. He then extended his algorithm to minor-closed classes of graphs with locally bounded tree-width [10]. Testing a fixed subgraph can be defined in first order logic (FOL) by a Σ_1-sentence and several generalization of Eppstein's work in this direction appeared. The most general results include:

- a linear time algorithm of Frick and Grohe [11] for deciding FOL properties of planar graphs,
- an almost linear-time algorithm of Frick and Grohe [11] for deciding FOL properties for classes of graphs with locally bounded tree-width,
- a fixed parameter algorithm of Dawar, Grohe and Kreutzer [2] for deciding FOL properties for classes of graphs locally excluding a minor, and
- a linear-time algorithm of Nešetřil and Ossona de Mendez [22] for deciding Σ_1-properties for classes of graphs with bounded expansion.

We generalize all these results in two different ways: we consider classes of graphs with bounded expansion, which generalize the results mentioned above, and we also consider dynamic setting where the input can change during computation.

A. Classes of sparse graphs

We now present the concept of classes of graphs of bounded expansion, introduced by Nešetřil and Ossona de Mendez in [20], [21], [22], [23]. Examples of such classes of graphs include proper minor-closed classes of graphs, classes of graphs with bounded maximum degree, classes of graphs excluding a subdivision of a fixed graph, classes of graphs that can be embedded on a fixed surface with bounded number of crossings per each edge and many others [25]. Many structural and algorithmic properties generalize from proper minor-closed classes of graphs to classes of graphs with bounded expansion [5], [26]. Let us clarify here that there does not seem to be a close relation between this concept and the well-known notion of expander graphs.

All graphs considered in this paper are simple and finite. A class of graphs is *hereditary* if it is closed under taking subgraphs. An *r-shallow minor* of a graph G is a graph that can be obtained from G by removing some of the vertices and edges of G and then contracting vertex-disjoint subgraphs of radius at most r to single vertices (removing arising loops and parallel edges). The *grad (greatest reduced average density) of rank r* of a graph G is equal to the largest average density of an r-shallow minor of G. The grad of rank r of G is denoted by $\nabla_r(G)$. In particular, $2\nabla_0(G)$ is the maximum average degree of a subgraph of G. If $\mathcal{G}$ is a class of graphs, then the class of r-shallow minors of graphs contained in $\mathcal{G}$ is denoted by $\mathcal{G}\nabla r$.

A function $f : \mathbb{N} \to \mathbb{R}^+$ *bounds expansion* of a graph G if $\nabla_r(G) \leq f(r)$ for every integer $r \geq 0$. A class $\mathcal{G}$ of graphs has *bounded expansion* if there exists a function $f : \mathbb{N} \to \mathbb{R}^+$ that bounds the expansion of all graphs in $\mathcal{G}$.

If g_0 is a real-valued function on a class $\mathcal{G}$ of graphs, then

$$\limsup_{G\in\mathcal{G}} g_0(G)$$

is the supremum of all reals α such that $\mathcal{G}$ contains infinitely many graphs G with $g_0(G) \geq \alpha$. A class $\mathcal{G}$ of graphs is *nowhere-dense* if

$$\lim_{r\to\infty} \limsup_{G\in\mathcal{G}\nabla r} \frac{\log ||G||}{\log |G|} \leq 1.$$

Every class of graphs with bounded expansion is nowhere-dense [24], but the converse is not true.

On the other hand, every class of graphs with locally bounded tree-width or locally excluding a minor is nowhere-dense. Recall that a class $\mathcal{G}$ of graphs has *locally bounded tree-width* if there exists a function $f : \mathbb{N} \to \mathbb{N}$ such that the subgraph induced by the r-neighborhood of every vertex v of a graph $G \in \mathcal{G}$ has tree-width at most $f(r)$ for every $r \geq 0$. A class $\mathcal{G}$ *locally excludes a minor* if there exist graphs $H_1, H_2, \ldots$ such that no subgraph of any graph $G \in \mathcal{G}$ induced by the r-neighborhood of any vertex v of G contains H_r as a minor. Finally, a class $\mathcal{G}$ of graphs has *locally bounded expansion* if there exists function $f : \mathbb{N}\times\mathbb{N} \to \mathbb{R}^+$ such that the $\nabla_r(G) \leq f(d, r)$ for every graph G that is a subgraph of the d-neighborhood of a vertex in a graph from $\mathcal{G}$. Similarly, one can define classes of *locally nowhere-dense* graphs, but it turns out that every such class is a class of nowhere-dense graphs, too.

If L is a language with relational and functional symbols, an *L-structure* A is a structure where these symbols are interpreted. The *Gaifman graph* of A is the undirected graph G_A with $V(G_A) = V(A)$ where two vertices a and b are adjacent if a and b are contained in the same tuple of a relation in A or $b = f(a)$ for a function f of A. We say that A is *guarded* by a graph G if $G_A \subseteq G$. The size of A, which is defined to be the sum of sizes of relations of A, is $O(|V(A)|)$ if A is guarded by a graph belonging to a fixed class of graphs with bounded expansion. This follows from the result that if G belongs to a class of graphs with bounded expansion, then its maximum average degree is bounded and thus G contains a linear number of complete subgraphs [28].

B. Our results

We study complexity of deciding properties that can be expressed in terms of a first-order logic formula for classes of sparse graphs and sparse relational structures. Our two algorithmic results that unify the results mentioned at the beginning are the following. We say that an algorithm is almost linear if its running time is bounded by $O(n^{1+\varepsilon})$ for every $\varepsilon > 0$, where n is the size of the input instance[1].

Theorem 1. *Let $\mathcal{G}$ be a class of graphs with bounded expansion, L a language and φ an L-sentence. There exists a linear time algorithm that decides whether an L-structure guarded by a graph $G \in \mathcal{G}$ satisfies φ.*

Theorem 2. *Let $\mathcal{G}$ be a class of graphs with locally bounded expansion, L a language and φ an L-sentence. There exists an almost linear time algorithm that decides whether an L-structure guarded by a graph $G \in \mathcal{G}$ satisfies φ.*

Our approach differs from the methods used in [2], [11] and is based on structural results on the existence of low tree-depth colorings for graphs with bounded expansion. After we announced our results in the survey paper [5], Dawar and Kreutzer [3] announced a proof of Theorem 1 and a proof of Theorem 2 for more general classes of nowhere-dense graphs. However, the proofs in the original version of [3] are incorrect. Kreutzer [19] informed us that they can prove Theorems 1 and 2 as stated in this paper using our notion of templates defined further.

We also consider dynamic setting and design the following data structures (the first can be viewed as a dynamic version of Theorem 1):

- for every class $\mathcal{G}$ of graphs with bounded expansion, a language L and an L-sentence φ, a data structure that is initialized with a graph $G \in \mathcal{G}$ and an L-structure A guarded by G in time $O(n)$ where n is the number of vertices of G and that supports:
  - adding a tuple to a relation of A in time $O(1)$ provided A stays guarded by G,
  - removing a tuple from a relation of A in time $O(1)$,
  - answering whether $A \models \varphi$ in time $O(1)$,
- for every class $\mathcal{G}$ of graphs with bounded expansion, an integer d_0 and a language L with no function symbols, a data structure that for an n-vertex graph $G \in \mathcal{G}$ is

[1] Some authors use a weaker notion, namely that for every $\varepsilon > 0$, there exists an algorithm with running time $O(n^{1+\varepsilon})$. All results stated in this paper hold with either of these two definitions.

initialized in time $O(n)$, and it supports the following operations starting with the empty L-structure A with $V(A) = V(G)$:

  - adding a tuple to a relation of A in time $O(1)$ provided A stays guarded by G,
  - removing a tuple from relation of A in time $O(1)$,
  - answering whether $A \models \varphi$ for any Σ_1-L-sentence φ with at most d_0 variables in time $O(|\varphi|)$ and outputting one of the satisfying assignments, and

  if $\mathcal{G}$ is only a class of nowhere-dense graphs, then the data structure supports:

  - adding a tuple to a relation of A in time $O(n^\varepsilon)$ provided A stays guarded by G,
  - removing a tuple from a relation of A in time $O(n^\varepsilon)$, and
  - answering whether $A \models \varphi$ for any Σ_1-L-sentence φ with at most d_0 variables in time $O(|\varphi|+n^\varepsilon)$ and if so, outputting one of the satisfying assignments,

  where ε is any positive real number, and
- for every class $\mathcal{G}$ of graphs with bounded expansion and an integer d, a data structure supporting queries whether two vertices are at distance at most d provided the input graph stays in $\mathcal{G}$ with the running times:

  - initialization time $O(n)$,
  - amortized time $O(\log^d n)$ for adding an edge,
  - time $O(1)$ for removing an edge,

  where n is the order of the stored graph, and, if $\mathcal{G}$ is only a class of nowhere-dense graphs, then the data structure has the following running times:

  - initialization time $O(n^{1+\varepsilon})$,
  - amortized time $O(n^\varepsilon)$ for adding an edge,
  - time $O(n^\varepsilon)$ for removing an edge,

  where ε is any positive real number.

The second of these data structures is needed in our linear-time algorithm for 3-coloring triangle-free graphs on surfaces [7], also see [6]. The last one is inspired by a data structure of Kowalik and Kurowski [15], [17] for deciding whether two vertices of a planar graph are connected by a path of length at most k, where k is a fixed constant.

II. Classes of graphs with bounded expansion

We briefly survey structural results that we need. A *rooted forest* F is a forest where every tree is an out-branching. The *depth* of F is the number of vertices of its longest directed path. The *closure* of F is the undirected graph obtained by adding all edges between vertices joined by a directed path and forgetting the orientation. The *tree-depth* of an (undirected) graph G is the smallest integer d such that G is a subgraph of the closure of a rooted forest of depth d. A vertex coloring of a graph G is a *low tree-depth coloring of order* K if the union of any s color classes, $s \leq K$, induces a subgraph with tree-depth at most s. The existence of low tree-depth colorings with bounded numbers of colors for graphs with bounded expansion is guaranteed by a seminal result of Nešetřil and Ossona de Mendez [21].

Before we state this result formally, we have to introduce more definitions. Consider an orientation of G. Let G' be the graph obtained from G by adding all edges xy such that:

- there exists a vertex z such that G contains an edge oriented from x to z and an edge oriented from z to y (*transitivity*), or
- there exists a vertex z such that G contains an edge oriented from x to z and an edge oriented from y to z (*fraternality*).

We call G' the *augmentation* of the orientation of G. The following was shown by Nešetřil and Ossona de Mendez [21]:

Theorem 3. *There exist functions f_1 and f_2 with the following property. Let G be a graph with expansion bounded by g. Consider an orientation of G such that each vertex has in-degree at most D, and let G' be the augmentation of this orientation of G. Then the expansion of G' is bounded by a function $g'(r) = f_1(g(f_2(r)), D)$.*

Let G be a graph from a class $\mathcal{G}$ with bounded expansion. Consider the following series of graphs: G_0 is obtained from G by orienting edges in such a way that the maximum in-degree of G_0 is at most $2\nabla_0(G)$. Let G_1 be the augmentation of this orientation of G_0, and orient G_1 so that the maximum in-degree of G_1 is at most $2\nabla_0(G_1)$; the edges present in G_0 do not necessarily have to preserve their orientation. In general, G_k is the augmentation of an orientation of G_{k-1} with maximum in-degree $2\nabla_0(G_{k-1})$. A greedy algorithm can be used to find such an orientation of G_{k-1}. The graph G_k is referred to as a *k-th augmentation* of G. By Theorem 3, the class $\mathcal{G}_k$ consisting of the graphs G_k obtained in the described way from the graphs in $\mathcal{G}$ has bounded expansion. It follows that $2\nabla_0(G_k) \leq d$ for some constant d depending only on the class $\mathcal{G}$ and k, and thus G_k is d-degenerate. In particular, the chromatic number of a k-th augmentation of a graph $G \in \mathcal{G}$ is bounded by a constant that depends on $\mathcal{G}$ only.

If G is an orientation of a graph, then any proper coloring of its $(3d^2+1)$-th augmentation G' of G is a low tree-depth coloring of order d [21], also see [5] for further details. Moreover, the subgraph H of G' induced by the vertices of any $s \leq d$ color classes contains a rooted forest F with depth at most s such that H is a subgraph of the closure of F. We refer to this property of the augmentation as *depth-certifying* and call F a *depth-certifying forest*. Hence, we obtain the following for classes of graphs with bounded expansion [21] and classes of nowhere-dense graphs [24].

Theorem 4. *Let $\mathcal{G}$ be a class of graphs with bounded expansion and d an integer. There exists k such that any proper coloring of a k-th augmentation of $G \in \mathcal{G}$ is a low tree-depth coloring of order d and the k-th augmentation is depth-certifying. In particular, G has a low tree-depth coloring* *of order d with at most K colors where K depends on $\mathcal{G}$ and d only. Such a coloring of $G \in \mathcal{G}$ and corresponding depth-certifying forests can be found in linear time.*

Theorem 5. *Let $\mathcal{G}$ be a class of nowhere-dense graphs and d an integer. There exists k such that any proper coloring of a k-th augmentation of a graph $G \in \mathcal{G}$ is a low tree-depth coloring of order d and the k-th augmentation is depth-certifying. In particular, G has a low tree-depth coloring of order d with at most $O(n^\varepsilon)$ colors for every $\varepsilon > 0$. Moreover, such a coloring of $G \in \mathcal{G}$ and corresponding depth-certifying forests can be found in almost linear time.*

III. Deciding FOL properties

We now present a proof of Theorem 1; here, we consider languages L with unary function symbols only.

We start with a lemma for removing quantifiers from an FOL formula (Lemma 9). If X is a set of L-terms (variables and compositions of unary functions applied to variables), an *X-template* T is a rooted forest with vertex set $V(T)$ with a mapping $\alpha_T : X \to V(T)$ such that $\alpha_T^{-1}(w) \neq \emptyset$ for every leaf w of T. If φ is a quantifier-free L-formula, then a *φ-template* is an X-template where X is the set of all terms appearing in φ. The *depth* of T is the maximum depth of a tree of T. Two X-templates T and T' are isomorphic if there exists a bijection $f : V(T) \to V(T')$ such that

- f is an isomorphism of T and T' as rooted forests, in particular, w is a root iff $f(w)$ is a root, and
- $f(\alpha_T(t)) = \alpha_{T'}(t)$ for every L-term $t \in X$.

The number of X-templates with a given depth is finite:

Proposition 6. *For every finite set of terms X and every integer d, there exist only finitely many non-isomorphic X-templates of depth at most d.*

Let X be a set of terms with variables $\{x_1, \dots, x_n\}$. An *embedding* of an X-template T in a rooted forest F is a mapping $\nu : V(T) \to V(F)$ such that $\nu(r)$ is a root of F for every root r of a tree of T and ν is an isomorphism of T and the subforest of F induced by $\nu(V(T))$. Let S be an L-structure guarded by the closure of F, and $v_1, \dots, v_n \in V(S)$. We say that the embedding ν is *$(v_1, \dots, v_n)$-admissible* for S if for every term $t \in X$, we have $\nu(\alpha_T(t)) = t(v_1, \dots, v_n)$ where $t(v_1, \dots, v_n)$ denotes the element of $V(S)$ obtained by substituting v_i for x_i in the term t and evaluating the functions forming the term t (in particular, if $x_i \in X$, then $\nu(x_i) = v_i$). Note that ν is uniquely determined by S and the values v_1, …, v_n.

If F is a rooted forest, then the function $p : V(F) \to V(F)$ is the *F-predecessor function* if $p(v)$ is the parent of v unless v is a root of F; if v is a root of F, $p(v)$ is v.

It is not hard to show that there exists a quantifier-formula testing whether an embedding is admissible.

Lemma 7 (Testing admissibility)**.** *Let $d \geq 0$ be an integer, L a language including a function symbol p and X a finite set of terms with variables x_1, …, x_n. If T is an X-template of depth at most d, then there exists a quantifier-free formula $\xi_T(x_1, \dots, x_n)$ such that for every rooted forest F and every L-structure S guarded by the closure of F such that p^S is the F-predecessor function in S, and for every n-tuple $v_1, \dots, v_n \in V(S)$, the L-structure S satisfies $\xi_T(v_1, \dots, v_n)$ if and only if there exists a $(v_1, \dots, v_n)$-admissible embedding of T in the forest F for S.*

The core of our argument is formed by the next lemma. Throughout the paper, we call a formula *simple* if it does not contain a composition of two or more functions.

Lemma 8. *Let d be an integer, L a language, $\varphi(x_0, \dots, x_n)$ a simple quantifier-free L-formula that is a conjunction of atomic formulas and their negations, and T a φ-template. There exist an integer K and an $\overline{L}$-formula $\overline{\varphi}_T$ such that*

- *$\overline{L}$ is the language with $\overline{L}^r = L^r \cup \{U_1, \dots, U_k\}$ and $\overline{L}^f = L^f \cup \{p\}$ where $U_1, \dots, U_k$ are new nullary or unary relations, $k \leq K$,*
- *$\overline{\varphi}_T$ is quantifier-free and the variables $x_1, \dots, x_n$ are the only variables that appear in $\overline{\varphi}_T$, but $\overline{\varphi}_T$ need not be simple, and*
- *for every rooted forest F with depth at most d and every L-structure S guarded by the closure of F, there exists an $\overline{L}$-structure $\overline{S}$ with $V(S) = V(\overline{S})$ such that for every $v_1, \dots, v_n \in V(S)$,*
  $S \models \varphi(v_0, v_1, \dots, v_n)$ for some $v_0 \in V(S)$ such that there exists a $(v_0, \dots, v_n)$-admissible embedding of T in F for S if and only if $\overline{S} \models \overline{\varphi}_T(v_1, \dots, v_n)$
  where $p^{\overline{S}}$ is the F-predecessor function and the relations $U_1^{\overline{S}}, \dots, U_k^{\overline{S}}$ can be computed (by listing the singletons they contain) in linear time given F and S. The interpretation of other symbols of L is preserved.

Proof: Fix a φ-template T and let q be the T-predecessor function. Let X be the set of all terms appearing in φ. Let ξ_T be the formula from Lemma 7. Finally, let $K = \max(\Delta, c) + 1$ where Δ is the maximum degree of T and c is the number of components of T.

Let $t = f(x_i)$ be a term appearing in φ. If $\alpha_T(t)$ is neither an ancestor nor a descendant of $\alpha_T(x_i)$, then for any choice of $v_1, \dots, v_n \in V(S)$, there is no $(v_0, \dots, v_n)$-admissible embedding of T for S because v_i and $f^S(v_i)$ are adjacent in the Gaifmann graph of S; in particular, one is a descendent of the other in F. Hence, we can set $\overline{\varphi}_T$ to $\perp$. So, we can assume the following:

The images under α_T of all function images of each x_i are ancestors or descendants of $\alpha_T(x_i)$. (1)

If $\alpha_T(x_0)$ is an ancestor of $\alpha_T(t)$, say $q^k(\alpha_T(t)) = \alpha_T(x_0)$ for $k \geq 0$, where t is a term such that x_0 does not appear in t, then $\overline{\varphi}_T$ will be the formula obtained from $\varphi \wedge \xi_T$ by replacing each x_0 with the term $p^k(t)$. Clearly, $\overline{S} \models \overline{\varphi}_T(v_1, \dots, v_n)$ if and only if there is a choice of

v_0 in $V(F)$ such that $S \models \varphi(v_0,\dots,v_n)$ and there is a $(v_0,\dots,v_n)$-admissible embedding of T in F for S. So, we can assume the following:

$$\text{Every } t \text{ such that } \alpha_T(t) \text{ is contained in the subtree of } T \text{ rooted at } \alpha_T(x_0) \text{ is a function image of } x_0. \quad (2)$$

We now define an auxiliary formula φ' to be the formula obtained from φ by replacing all subformulas of the form:

- $t = t'$ where t and t' are terms and $\alpha_T(t) \neq \alpha_T(t')$,
- $t \neq t'$ where t and t' are terms and $\alpha_T(t) = \alpha_T(t')$, or
- $R(t_1,\dots,t_m)$ such that $\alpha_T(t_1),\dots,\alpha_T(t_m)$ are not vertices of a clique in the closure of T,

with $\perp$ since such a subformula is not satisfied for any choice of v_0 for which there exists a $(v_0,\dots,v_n)$-admissible embedding of T in F for S. It follows that for every v_0 such that there is a $(v_0,\dots,v_n)$-admissible embedding of T in F for S, $S \models \varphi(v_0,\dots,v_n)$ if and only if $S \models \varphi'(v_0,\dots,v_n)$.

Suppose first that the tree of T containing $\alpha_T(x_0)$ also contains an α_T-image of another variable. Let v be the nearest ancestor of $\alpha_T(x_0)$ in T such that there exists a term $t_v \in X$ that does not contain x_0 and v is an ancestor of $\alpha_T(t_v)$. Note that $v \neq \alpha_T(x_0)$ by (2). Let d_v be the depth of v in T, d_{x_0} the depth of $\alpha_T(x_0)$ and m the number of children of v in T. Let $t_1,\dots,t_{m-1}$ be terms such that $\alpha_T(t_i)$, $1 \le i \le m-1$, are vertices of different subtrees rooted at a child of v not containing $\alpha_T(x_0)$. Observe that the variable x_0 does not appear in t_1, …, t_{m-1} by (1).

Let X_0 be the subset of X consisting of terms in which x_0 appears, and let T_0 be the X_0-template obtained from T by taking the minimal rooted subtree containing $\alpha_T(X_0)$ and the root of the tree containing $\alpha_T(x_0)$, and restricting the function α_T to X_0. Further, let X_0' be the subset of X consisting of X_0 and the terms t such that $\alpha_T(t)$ lies on the path between the root and $\alpha_T(x_0)$, and let X_0'' be the subset of X_0 consisting of the terms mapped to a descendant of v.

We define a unary relation $U_1(w)$ to be the set of elements w of F at depth d_v+1 such that the subtree of w in F contains an element v_0 at depth d_{x_0} (in F) such that

- there is a (v_0)-admissible embedding of the template T_0 in F for S, and
- all clauses appearing in the conjunction φ' with terms from X_0' only and with at least one term from X_0'' are true with $x_0 = v_0$ and the terms $t \in X_0' \setminus X_0$, say $\alpha_T(t) = q^k(\alpha_T(x_0))$, replaced with $p^{\overline{S},k}(v_0)$.

The relation $U_1(w)$ can be computed as follows: for every element $v_0 \in V(S)$ at depth d_{x_0} of F, evaluate all terms in X_0 and test whether the tree T_0 and the rooted subtree of F containing the values of the terms are isomorphic as rooted trees (this can be done in time linear in the size of T_0 which is constant). If they are isomorphic, evaluate the clauses in the conjunction φ' with terms from X_0' only and with at least one term from X_0'' with the terms in $X_0' \setminus X_0$ replaced with $p^{\overline{S},k}(v_0)$. If all of them are true, add the predecessor w of v_0 at depth d_v+1 in F to U_1. Since the time spent by checking every vertex v_0 at depth d_{x_0} of F is constant, the time needed to compute U_1 is linear.

We define the unary relation $U_i(w)$, $2 \le i \le m+1$, to contain all w at depth d_v such that $U_1(w')$ for at least $i-1$ children w' of w. The relations $U_i(w)$, $2 \le i \le m+1$, can be computed in linear time from U_1.

Let φ'' be the formula obtained from φ' by removing all clauses with terms from X_0' only that contain at least one term from X_0''. Observe that if t is a term in φ'' such that x_0 appears in t, i.e., $t \in X_0 \setminus X_0''$, then $\alpha_T(t)$ lies on the path between v and the root. Let φ''' be the formula obtained from φ'' by replacing every term t, in which x_0 appears, with $p^k(t_v)$, where k is the integer such that $\alpha_T(t) = q^k(\alpha_T(t_v))$. Let T' be the $(X \setminus X_0)$-template obtained from T by taking the minimal rooted subtree containing $\alpha_T(X \setminus X_0)$ and restricting the function α_T to $X \setminus X_0$. The formula $\overline{\varphi}_T$ will then be the conjunction of the following formulas:

- the formula $\varphi'''(x_1,\dots,x_n)$,
- the formula $\xi_{T'}$ from Lemma 7 for the $(X \setminus X_0)$-template T', and
- the formulas
$$\left(\bigwedge_{i\in Y} U_1(p^{k_i-1}(t_i))\right) \Rightarrow U_{|Y|+2}(p^k(t_v))$$
for all subsets Y of the set $\{1,\dots,m-1\}$ where k is the integer such that $q^k(\alpha_T(t_v)) = v$ and k_i, $i = 1,\dots,m-1$, are the integers such that $q^{k_i}(\alpha_T(t_i)) = v$.

If $\overline{S} \models \overline{\varphi}_T(v_1,\dots,v_n)$, then there is a $(v_1,\dots,v_n)$-admissible embedding of T' in F and the vertex $v = p^k(t_v)$ has a son w such that $U_1(w)$ and the subtree of F rooted in w does not contain the value of any term appearing in $X \setminus X_0$ (this is guaranteed by the last type of formulas in the definition of $\overline{\varphi}_T$). In particular, the subtree rooted in w contains a vertex v_0 such that the $(v_1,\dots,v_n)$-admissible embedding of T' can be extended to a $(v_0,\dots,v_n)$-admissible embedding of T in F for S and all clauses in the conjunction of φ' containing terms from X_0' are satisfied with $x_0 = v_0$. Since $\overline{S} \models \varphi'(v_1,\dots,v_n)$, it follows that $S \models \varphi(v_0,\dots,v_n)$. The argument that the existence of v_0 such that $S \models \varphi(v_0,\dots,v_n)$ and the existence of a $(v_0,\dots,v_n)$-admissible embedding of T in F for S implies that $\overline{S} \models \overline{\varphi}_T(v_1,\dots,v_n)$ follows the same lines.

The case that the tree of T that contains the vertex $\alpha_T(x_0)$ does not contain α_T-image of another variable is handled similarly. In this case, the predicate U_1 is defined for the roots of the trees of F, and the predicates U_2, …, U_{m+1} are nullary predicates such that U_i is true if U_1 is satisfied for at least $i-1$ roots of F. ∎

Lemma 8 yields the following.

Lemma 9 (Quantifier elimination lemma). *Let $d \ge 0$ be an integer, L a language and φ a simple L-formula of the* *form $\exists x_0\ \varphi'$ such that φ' is a quantifier-free L-formula with free variables $x_0,\dots,x_n$. There exists an integer K and an $\overline{L}$-formula $\overline{\varphi}$ such that the following holds:*

- *$\overline{L}$ is the language with $\overline{L}^r = L^r \cup \{U_1,\dots,U_k\}$ and $\overline{L}^f = L^f \cup \{p\}$ where $U_1,\dots,U_k$ are new nullary or unary relations and $k \le K$,*
- *$\overline{\varphi}$ is quantifier-free and the variables $x_1,\dots,x_n$ are the only variables that appear in $\overline{\varphi}$, and*
- *for every rooted forest F with depth at most d and every L-structure S guarded by the closure of F, there exists an $\overline{L}$-structure $\overline{S}$ with $V(S) = V(\overline{S})$ such that for every $v_1,\dots,v_n \in V(S)$,*
  $S \models \varphi(v_1,\dots,v_n)$ *if and only if* $\overline{S} \models \overline{\varphi}(v_1,\dots,v_n)$
  where $p^{\overline{S}}$ is the F-predecessor function and the relations $U_1^{\overline{S}},\dots,U_K^{\overline{S}}$ can be computed (by listing the singletons they contain) in linear time given F and S. The interpretation of other symbols of L is preserved.

Proof: Let d, L and φ' be fixed. Without loss of generality, the formula φ' is in the disjunctive normal form and all the variables $x_0,\dots,x_n$ appear in φ'. If $n=0$, set $K=1$, enhance L with a nullary relation U_1 and set $\overline{\varphi} = U_1$. The value of U_1 can be determined in linear time by testing all possible choices for $x_0 \in V(S)$.

If $n \ge 1$, the proof proceeds by induction on the length of φ'. If φ' is a disjunction of two or more conjunctions, i.e., $\varphi' = \varphi_1 \vee \varphi_2$, we apply induction to $\exists x_0 \varphi_1$ and $\exists x_0 \varphi_2$.

In the rest, we assume that φ' is a conjunction. By Lemma 8, for every φ'-template T of depth at most d, there exists a quantifier-free $\overline{L}$-formula $\overline{\varphi}_T$ such that for every $v_1,\dots,v_n \in V(S)$, $\overline{S} \models \overline{\varphi}_T(v_1,\dots,v_n)$ if and only if there exists v_0 such that there is a $(v_0,v_1,\dots,v_n)$-admissible embedding of T in F for S and $S \models \varphi'(v_0,v_1,\dots,v_n)$. The desired formula $\overline{\varphi}$ is then obtained as the disjunction of the finitely many (see Proposition 6) $\overline{L}$-formulas $\overline{\varphi}_T$ where the disjunction runs over all choices of φ'-templates T. ■

We are now ready to prove Theorem 1.

Proof of Theorem 1: The proof proceeds by induction on the number of quantifiers contained in φ. If φ is quantifier-free, then there is nothing to prove. Hence, assume that φ contains at least one quantifier. Let A be the input L-structure and G a graph guarding A. We can assume that φ is simple: add an edge oriented from u to v to the graph G for every $u,v \in V(A)$ such that $u = f^A(v)$; since the number of function symbols f is bounded, the maximum in-degree is increased by a constant only. For each composition of functions, we define a new functional symbol. The resulting structure is guarded by a graph $H = G^{(k)}$ where k is the maximum number of compositions of functions in φ.

Since $\forall x\ \psi$ is equivalent to $\neg\exists x\ \neg\psi$, we can assume that φ contains a subformula of the form $\exists x_0\psi$, where ψ is a formula with variables $x_0,x_1,\dots,x_N$ and with no quantifiers. Let N_0 be the number $N+1$ increased by the number of distinct function images of $x_0,x_1,\dots,x_N$ appearing in ψ. Let $K = 3N_0^2+1$ and consider a coloring of a K-th augmentation $H^{(K)}$ of H; this coloring is a depth-certifying low-tree-depth coloring of H of order N_0 (see the discussion before Theorem 4). Let K_0 be the number of colors used by this coloring and C_i, $i=1,\dots,K_0$, a unary relation containing vertices with the i-th color.

For each function symbol f and $i=1,\dots,K_0$, let C_i^f be the predicate such that $C_i^f(v)$ is true for $v \in V(A)$ if and only if the color of $f(v)$ is i. The colors of the variables and their function images appearing in ψ can be described by an N_0-tuple α of numbers between 1 and K_0. For such an N_0-tuple α, let φ_α be the conjunction of the terms of form $C_{\alpha_i}(x_i)$ and $C^f_{\alpha_j}(x_i)$ that verifies that the colors of $x_0,\dots,x_N$ and their function images are consistent with α. Clearly, $\exists x_0\psi$ is equivalent to the disjunction of the formulas $\exists x_0(\psi \wedge \varphi_\alpha)$ ranging through all choices of α.

For each function symbol $f \in L^f$, we introduce a new function symbol f_α defined by $f_\alpha(v) = f(v)$ if both v and $f(v)$ have a color in α and by $f_\alpha(v) = v$ otherwise. For each relation symbol $R \in L^r$ of arity greater than one, let R_α be defined by restricting R to the vertices with color α. Let A_α be the corresponding relational structure. Let φ''_α be the formula obtained from $\exists x_0(\psi \wedge \varphi_\alpha)$ by replacing each function symbol f by f_α and each relation symbol R of arity greater than one by R_α. For a fixed N_0-tuple α, the formula $\exists x_0(\psi \wedge \varphi_\alpha)$ is true for A if and only if φ''_α is true for A_α. Note that A_α is guarded by the graph H_α obtained from H by removing the edges incident with the colors not in α. Since $N_0 \le K$, $H^{(K)}$ contains an out-branching F_α of depth at most N_0 whose closure contains H_α.

Apply Lemma 9 to φ''_α and F_α, obtaining a formula ψ_α and a structure A'_α. We claim that for any choice of $x_1,\dots,x_N$, $A'_\alpha \models \psi_\alpha$ is satisfied in A'_α if and only if $\exists x_0(\psi \wedge \varphi_\alpha)$ is satisfied in A. If the colors of $x_1,\dots,x_N$ and their function images do not agree with α, then both formulas are not satisfied. Otherwise, the values of $x_1,\dots,x_N$ correspond to the vertices of F_α and the formula ψ_α is in A'_α satisfied if and only if $\exists x_0(\psi \wedge \varphi_\alpha)$ is satisfied in A.

Note that the application of Lemma 9 for each N_0-tuple α extends the language L by a single unary function, which is determined by F_α, and several nullary and unary relations, in addition to the new symbols defined in A_α. Since the number of choices of α is bounded by a function of N_0 and K_0, the number of new functional and relational symbols depends only on the formula ψ and the class $\mathcal{G}$.

Replace now the subformula $\exists x_0\psi$ in φ by the disjunction of ψ_α with all choices of α. The resulting formula is guarded by a graph $H^{(K)} = G^{(k+K)}$, and because it contains one less quantifier than φ, we can apply induction.

Since each application of Lemmas 9 in the induction can be performed in linear time and the number of their applications is bounded by a function depending on φ and $\mathcal{G}$ only, the statement of the theorem follows. ■

The proof of Theorem 2 is based on locality of first order formulas as formalized in Gaifman's theorem. Following the approach from [11], the graph G is covered with neighborhoods in which localized sentences are evaluated and then combined to determine the value of φ. Due to space limitations, details have to be omitted.

IV. Dynamic data structures

We start with data structures for testing Σ_1-properties.

Theorem 10. *Let L be a language with no function symbols, d_0 a fixed integer and $\mathcal{G}$ a class of graphs with bounded expansion. There exists a data structure representing an L-structure S such that*

- *given a graph $G \in \mathcal{G}$, the data structure is initialized in linear time with S being initially empty,*
- *the data structure representing an L-structure S can be changed to represent an L-structure S' by adding or removing a tuple from one of the relations in constant time provided that both S and S' are guarded by G,*
- *the data structure determines in time bounded by $O(|\varphi|)$ whether a given Σ_1-L-sentence φ with at most d_0 variables is satisfied by S, and if so, it outputs one of the satisfying assignments.*

Theorem 11. *Let L be a language with no function symbols, k_0 a fixed integer, ε a positive real and $\mathcal{G}$ a class of nowhere-dense graphs. There exists a data structure representing an L-structure S such that*

- *given a graph $G \in \mathcal{G}$, the data structure is initialized in time $O(n^{1+\varepsilon})$ with S being initially empty,*
- *the data structure representing an L-structure S can be changed to represent an L-structure S' by adding or removing a tuple from one of the relations in time $O(n^{\varepsilon})$ provided that both S and S' are guarded by G,*
- *the data structure determines in time bounded by $O(|\varphi| + n^{\varepsilon})$ whether a given Σ_1-L-sentence φ with at most k_0 variables is satisfied by S, and if so, it outputs one of the satisfying assignments.*

Theorems 10 and 11 follow from Lemma 12 which we prove further. To do so, we need additional definitions.

Let L be a language with no function symbols. For an integer k, a *k-labelled* L-structure is an L-structure S with a partial injective mapping $\alpha : [1, k] \to V(S)$, i.e., α need not be defined for all integers between 1 and k.

The *trunk* of a k-labelled L-structure S is the L-structure obtained from S by removing all relations with elements only from $\alpha([1, k])$. A k-labelled L-structure S is *hollow* if it is equal to its trunk. Two k-labelled L-structures S_1 and S_2 are *k-isomorphic* if their trunks are isomorphic through an isomorphism commuting with mappings α_1 and α_2.

Suppose now that an L-structure S is guarded by the closure of a rooted tree T. For a vertex v of T at depth d, let $P_T(v)$ denote the path from the root of T to v and $T\langle v\rangle$ the elements the subtree of v (including v itself). Then, $S\langle v\rangle$ denotes the set of all d-labelled L-structures S' such that S' is an induced substructure of S with elements only in $P_T(v) \cup T\langle v\rangle$ and $\alpha(i) = w$ for every vertex $w \in P_T(v) \cap V(S')$ at depth $i-1$. If a vertex of $P_T(v)$ at depth $i-1$ is not contained in S', then $\alpha(i)$ is not defined.

Lemma 12. *Let L be a language with no function symbols, d_0 a fixed integer and F a rooted forest of depth at most d_0. There exists a data structure representing an L-structure S guarded by the closure of F such that*

- *the data structure is initialized in linear time,*
- *the data structure representing an L-structure S can be changed to represent an L-structure S' by adding or removing a tuple from one of the relations in constant time provided that both S and S' are guarded by the closure of F, and*
- *the data structure determines in time bounded by $O(|\varphi|)$ whether a given Σ_1-L-sentence φ with at most d_0 variables is satisfied by S, and if so, it outputs one of the satisfying assignments.*

Proof: For every vertex v of F at depth d, we will store

- the list of all relations from S that contain v and all their elements are in $P_F(v)$, and
- the list of non-d-isomorphic d-labelled hollow L-structures with at most d_0 elements that are d-isomorphic to a d-labelled L-structure contained in $S\langle v\rangle$.

Observe that there are only finitely many non-d-isomorphic d-labelled L-structures with at most d_0. If v is a non-leaf vertex of F, there will be a third list associated with v:

- the list of non-isomorphic $(d+1)$-labelled hollow L-structures S' with at most d_0 elements that appear in the second list of at least one child of v; for each such S', there will be stored the list of all such children v.

In addition, there will be a global list of all non-isomorphic induced L-substructures of S with at most d_0 elements.

Let us describe how these lists are initialized. The initialization of the first type of lists is trivial: put each relation to the list of its element that is farthest from the root.

Initialization of other lists is more difficult. We proceed from the leaves. If v is a leaf at depth d, then the second list of v contains only those hollow d-labelled L-structures S' that are formed by vertices on $P(v)$ such that if $v \in V(S')$, then S' contains precisely all unary relations of S containing v, and if $v \notin V(S')$, then S' contains no relations at all.

Suppose now that v is not a leaf. The third list associated with v can be initialized by merging the second type of lists of children of v. We describe how it can be decided whether a d-labelled hollow L-structure S' should be contained in the list of v of the second type. Assume that $S\langle v\rangle$ contains a d-labelled hollow L-structure S'' that is d-isomorphic to S'.

Then, $V(S'')$ can be decomposed into disjoint subsets $V_0, V_1, \ldots, V_m$ such that $V_0 = V(S'') \cap P(v)$, each of the sets V_i, $i = 1, \ldots, m$, is fully contained in a subtree of a child v_i of v, and different subsets $V_1, \ldots, V_m$ are contained in different subtrees. Observe that every relation of S'' must be contained in $V_0 \cup V_i$ for some $i = 1, \ldots, m$, and the only relations of S'' contained in V_0 are those that contain v.

Hence, the existence of S'' can be tested by considering all partitions of $V(S')$ into disjoint subsets $V_0, V_1, \ldots, V_m$ such that $\alpha([1,d]) \subseteq V_0$, $|V_0 \setminus \alpha([1,d])| \leq 1$ and every relation of S' is contained in $V_0 \cup V_i$ for some $i = 1, \ldots, m$, and then testing the existence of children $v_1, \ldots, v_m$ such that the second list of v_i contains a $(d+1)$-labelled hollow L-structure $(d+1)$-isomorphic to the $(d+1)$-labelled hollow L-structure of S' induced by $V_0 \cup V_i$; if $|V_0 \setminus \alpha([1,d])| = 1$, then $\alpha(d+1)$ is defined to be equal to the unique element of $V_0 \setminus \alpha([1,d])$ and we also test whether the relations of S' containing $\alpha(d+1)$ are precisely those relations of S restricted to $P(v)$ that contain v (those in the first list of v).

We now describe how to test the existence of children $v_1, \ldots, v_m$. Let W be the set of children of v such that: if v has at most m children with their second list containing a $(d+1)$-labelled hollow L-structure $(d+1)$-isomorphic to the substructure of S' induced by $V_0 \cup V_i$, then W contains all such children of v. If v has more than m such children, then W contains arbitrary m of these children. Clearly, $|W| \leq m^2 \leq d_0^2$. In order to test the existence of such children $v_1, \ldots, v_m$ of v, we form an auxiliary bipartite subgraph B: one part of B is formed by numbers $1, \ldots, m$ and the other part by children of v contained in W. A child $w \in W$ is joined to a number i if the second list of w contains a $(d+1)$-labelled hollow L-structure $(d+1)$-isomorphic to the substructure of S' induced by $V_0 \cup V_i$.

Now observe that the children $v_1, \ldots, v_m$ exist if and only if B has a matching of size m. Since the number of disjoint non-empty partitions of $V(S')$ to $V_0, \ldots, V_m$ is bounded, testing the existence of a d-labelled hollow L-structure S'' can be performed in constant time for v.

It remains to construct the global list of induced L-substructures S_0 with at most d_0 elements. For every L-structure S' with at most d_0 elements, we compute the list of trees of F that contain S'. Since S_0 is an induced substructure of S' if and only if there exist element-disjoint L-structures $S_1, \ldots, S_m$ contained in distinct trees of F such that $S_0 = S_1 \cup \cdots \cup S_m$, we can compute the global list using the auxiliary bipartite graph described earlier. Since all structures involved contain at most d_0 elements, this phase requires time linear in the number of trees of F.

We have shown that the data structure can be initialized in linear time. Let us now focus on updating the structure and answering queries. Consider a tuple $(v_1, \ldots, v_k)$ that is added to a relation R or removed from a relation R. Let r be the root of a tree in F that contains all the elements $v_1, \ldots, v_k$ and assume that $v_1, \ldots, v_k$ appear in this order on a path from r. By the definition, the only lists affected by the change are those associated with vertices on the path $P(v_k)$. Recomputing each of these lists requires constant time (we proceed in the same way as in the initialization phase except we do not have to swap through the children of the vertices on the path to determine which of them contain particular k-labelled hollow L-substructure S' in their lists). Since the number of vertices on the path $P(v_k)$ is at most d_0, updating the data structure requires constant time only.

It remains to describe how queries are answered. Since a Σ_1-sentence with $d \leq d_0$ variables is satisfied if and only if an L-structure contains an induced L-substructure with at d elements that satisfies the sentence, the queries can be answered in constant by inspecting the global list. The satisfying assignment can be provided in constant time if during the computation for each substructure a certificate why it was included in the list is stored (this requires constant time overhead only). ∎

We now give a data structure for testing FOL properties.

Theorem 13. *Let $\mathcal{G}$ be a class of graphs with bounded expansion, L a language and φ an L-sentence. There exists a data structure that is initialized with an n-vertex graph $G \in \mathcal{G}$ and an L-structure A guarded by G in time $O(n)$ and supports the following operations:*

- *adding a tuple to a relation of A in constant time provided A stays guarded by G,*
- *removing a tuple from a relation of A in constant time,*
- *determining in constant time whether $A \models \varphi$.*

In Theorem 13, we do not allow to change function values of functions from L to simplify our exposition; this does not present a loss of generality as one can model functions as binary relations. Theorem 13 follows from a dynamized version of Theorem 1.

Theorem 14. *Let $\mathcal{G}$ be a class of graphs with bounded expansion, L a language and φ an L-sentence. There exists a language L' and a quantifier-free L'-sentence φ' and a data structure representing an L'-structure A' that can be initialized with an n-vertex graph $G \in \mathcal{G}$ and an L-structure A guarded by G in time $O(n)$, $V(A) = V(A')$, such that*

- *$A \models \varphi$ if and only if $A' \models \varphi'$, in particular, testing whether $A \models \varphi$ can be performed in constant time,*
- *adding a tuple to a relation of A can be done in constant time provided A stays guarded by G, and*
- *removing a tuple from a relation of A can be done in constant time.*

Theorem 14 is based on a dynamization of Lemma 8.

Lemma 15. *Let d be a positive integer, L a language, $\varphi(x_0, \ldots, x_n)$ a simple quantifier-free L-formula that is a conjunction of atomic formulas and their negations, and T a φ-template. There exists an integer K and an $\overline{L}$-formula $\overline{\varphi}_T$ such that*

- *$\overline{L}$ is the language with $\overline{L}^r = L^r \cup \{U_1, \ldots, U_k\}$ and $\overline{L}^f = L^f \cup \{p\}$ where $U_1, \ldots, U_k$ are new nullary or unary relations, $k \leq K$,*

- *$\overline{\varphi}_T$ is quantifier-free and the variables $x_1, \ldots, x_n$ are the only variables that appear freely in $\overline{\varphi}_T$, but $\overline{\varphi}_T$ need not be simple, and*
- *for every rooted forest F with depth at most d and every L-structure S guarded by the closure of F, there exists an $\overline{L}$-structure $\overline{S}$ with $V(S) = V(\overline{S})$ such that for every $v_1, \ldots, v_n \in V(S)$*

  $S \models \varphi(v_0, v_1, \ldots, v_n)$ for some $v_0 \in V(S)$ such that there exists a $(v_0, \ldots, v_n)$-admissible embedding of T in F for S if and only if $\overline{S} \models \overline{\varphi}_T(v_1, \ldots, v_n)$

  where $p^{\overline{S}}$ is the F-predecessor function and the relations $U_1^{\overline{S}}, \ldots, U_k^{\overline{S}}$ can be computed (by listing the singletons they contain) in linear time given F and S. The interpretation of other symbols is preserved in $\overline{S}$,
- *adding to or removing a tuple from a relation of S results in adding and removing a constant number of singletons from unary relations among $U_1^{\overline{S}}, \ldots, U_k^{\overline{S}}$, and these changes can be computed in constant time, provided S stays guarded by the closure of F.*

Proof: We need to describe how the relations $U_1^{\overline{S}}, \ldots, U_k^{\overline{S}}$ can be updated in constant time after adding or removing a tuple to a relation of S. Let us consider in more detail the case analyzed in the proof of Lemma 8 and leave to the reader the case mentioned at the end of the proof of Lemma 8. Recall (see the proof of Lemma 8 for notation) that $U_1(w)$ is a unary relation containing elements w of F at depth $d_v + 1$ such that the subtree of w in F contains an element v_0 at depth d_{x_0} (in F) with the following properties:

- there is a (v_0)-admissible embedding of the template T_0 in F for S, and
- all clauses appearing in the conjunction φ' with terms from X_0' only and with at least one term from X_0'' are true with $x_0 = v_0$ and the terms $t \in X_0' \setminus X_0$, say $\alpha_T(t) = q^k(\alpha_T(x_0))$, replaced with $p^{\overline{S},k}(v_0)$.

Since none of the functions of S changes, the first condition cannot change when adding or removing a tuple to a relation of S. The second one can change only when a tuple containing a term from X_0'' with $x_0 = v_0$ is added or removed. Since the values of the terms in X_0'' with $x_0 = v_0$ appear only in a subtree of w, only a single element can be added to or removed from U_1. Based on the tuple we add or remove, we can identify which w can be added to or removed from U_1 and, using the data structure from the proof of Lemma 12, we can test in constant time the existence of v_0 satisfying the second condition (the values of all terms from X_0 with $x_0 = v_0$ are in the subtree of w and of those in $X_0' \setminus X_0$ are on the path from w to the root).

Once the relation U_1 is updated, the relations $U_2, \ldots, U_k$ can be updated in constant time as well: keep a counter determining the number of children in U_1. ■

We are now ready to prove Theorem 14.

Proof of Theorem 14: When the L-sentence φ is fixed in Theorem 1, the language L' and the L'-sentence φ' are also fixed. Hence, the only object that changes when the relations of A changes are relations in A'; since the functions in A' stay the same and thus the m-th augmentation of the graph G from Theorem 1 that guards A' stays the same.

We now have to inspect the proofs of Lemma 9 and Theorem 1 in more detail. Since the graph G and all its augmentations stay the same, the coloring used in Lemma 9 also does not change. In particular, at each step of the inductive proof of Theorem 1, every rooted forest F to which Lemma 9 is applied stays the same. In the proof of Lemma 9, we replace use of Lemma 8 with use of Lemma 15 and observe that every change in S results in a constant number of changes in $\overline{S}$ which can be identified in constant time. Hence, in the inductive proof of Theorem 1, a single change in A results in constantly many changes to the structure obtained in the first inductive step, which result in constantly many changes to the structure obtained in the second inductive step, etc. Since time to update the final L'-structure A' is constant for each of constantly many choices that propagates through the induction from a single change of A, the overall update time is constant. ■

At the end of this section, we present our generalization of the data structure designed by Kowalik and Kurowski [15], [17].

Theorem 16. *Let L be a language containing only binary relation symbols, ℓ an integer, and $\mathcal{G}$ a hereditary class of graphs with bounded expansion. There exists a data structure representing an L-structure S with $G_S \in \mathcal{G}$ such that*

- *the data structure can be initialized in time $O(|S|)$,*
- *it can be transformed to represent $S + R(u, v)$, $R \in L^r$, in amortized time $O(\log^\ell |S|)$, given that $G_{S+R(u,v)} \in \mathcal{G}$,*
- *it can be transformed to represent $S - R(u, v)$, $R \in L^r$, in constant time, and*
- *a query whether*

  $$(\exists x_1) \ldots (\exists x_{\ell-1}) R_1(u, x_1) \wedge \ldots \wedge R_\ell(x_{\ell-1}, v)$$

  for a given pair of elements u and v and a given sequence $R_1, \ldots, R_\ell$ can be answered in constant time. Moreover, in the positive case, a choice of $x_1, \ldots, x_{\ell-1}$ that satisfy $R_1(u, x_1) \wedge \ldots \wedge R_\ell(x_{\ell-1}, v)$ can also be found in constant time.

For classes of nowhere-dense graphs, we obtain:

Theorem 17. *Let L be a language containing only binary relation symbols, ℓ an integer, ε a positive real, and $\mathcal{G}$ a hereditary class of nowhere-dense graphs. There exists a data structure representing an L-structure S, where $n = |V(S)|$, with $G_S \in \mathcal{G}$ such that*

- *the initial data structure representing S can be built in time $O(n^{1+\varepsilon})$,*

- *it can be transformed to represent $S+R(u,v)$, $R \in L^r$, in amortized time $O(n^\varepsilon)$, assuming that $G_{S+R(u,v)} \in \mathcal{G}$,*
- *it can be transformed to represent $S-R(u,v)$, $R \in L^r$, in time $O(n^\varepsilon)$, and*
- *it answers queries whether*

$$(\exists x_1)\ldots(\exists x_{\ell-1}) R_1(u,x_1) \wedge \ldots \wedge R_\ell(x_{\ell-1},v)$$

for a given pair of vertices u and v and a given sequence $R_1,\ldots,R_\ell$ in time $O(n^\varepsilon)$. Moreover, in the positive case, a choice of $x_1,\ldots,x_{\ell-1}$ that satisfy $R_1(u,x_1) \wedge \ldots \wedge R_\ell(x_{\ell-1},v)$ can also be found in time $O(n^\varepsilon)$.

ACKNOWLEDGEMENT

The authors have been supported by Institute for Theoretical Computer Science (ITI), the project 1M0545 of the Ministry of Education of Czech Republic. The third author was partially supported by NSF under Grant No. DMS-0701077.

REFERENCES

[1] B. Courcelle, The monadic second-order logic of graph I. Recognizable sets of finite graphs, Inform. and Comput. 85 (1990), 12–75.

[2] A. Dawar, M. Grohe and S. Kreutzer, *Locally excluding a minor*, in: Proc. LICS'07, IEEE Computer Society Press, 270–279.

[3] A. Dawar and S. Kreutzer, *Parameterized Complexity of First-Order Logic*, Electronic Colloquium on Computational Complexity, TR09-131 (2009).

[4] Z. Dvořák, K. Kawarabayashi and R. Thomas, *Three-coloring triangle-free planar graphs in linear time*, in: Proc. SODA'09, ACM&SIAM, 2009, 1176–1182.

[5] Z. Dvořák and D. Král', *Algorithms for classes of graphs with bounded expansion*, in: Proc. WG'09, LNCS vol. 5911, Springer, 2009, 17–32.

[6] Z. Dvořák, D. Král' and R. Thomas, *Coloring triangle-free graphs on surfaces*, in: Proc. SODA'09, ACM&SIAM, 2009, 120–129.

[7] Z. Dvořák, D. Král' and R. Thomas, *Three-coloring triangle-free graphs on surfaces VI. A linear-time algorithm*, in preparation.

[8] D. Eppstein, *Subgraph isomorphism in planar graphs and related problems*, in: Proc. SODA'95, ACM&SIAM, 632–640.

[9] D. Eppstein, *Subgraph isomorphism in planar graphs and related problems*, J. Graph Algorithms Appl. 3 (1999), 1–27.

[10] D. Eppstein, *Diameter and treewidth in minor-closed graph families*, Algorithmica 27 (2000), 275–291.

[11] M. Frick and M. Grohe, *Deciding first-order properties of locally tree-decomposable structures*, J. ACM 48 (2001), 1184–1206.

[12] H. Gaifman, *On local and non-local properties*, in: Proc. Herbrands Symp. Logic Coloq., North-Holland, 1982.

[13] M. Garey, D. Johnson and L. Stockmeyer, *Some simplified NP-complete graph problems*, Theoret. Comput. Sci. 1 (1976) 237–267.

[14] H. Grötzsch, *Ein Dreifarbensatz für dreikreisfreie Netze auf der Kugel*, Wiss. Z. Martin-Luther-Universität, Halle, Wittenberg, Math.-Nat. Reihe 8 (1959), 109–120.

[15] L. Kowalik and M. Kurowski, *Short path queries in planar graphs in constant time*, in: Proc. STOC'03, 143–148.

[16] L. Kowalik, *Fast 3-coloring triangle-free planar graphs*, in: Proc. ESA'04, LNCS vol. 3221, Springer, 2004, 436–447.

[17] L. Kowalik and M. Kurowski, *Oracles for bounded length shortest paths in planar graphs*, ACM Trans. Algorithms 2 (2006), 335–363.

[18] S. Kreutzer, *Algorithmic meta-theorems*, to appear in a workshop volume for a workshop held in Durham 2006 as part of the Newton institute special programme on Logic and Algorithms. An extended abstract appeared in: Proc. IWPEC'08, LNCS vol. 5018, Springer, 2008, 10–12.

[19] S. Kreutzer, private communication, July 2010.

[20] J. Nešetřil and P. Ossona de Mendez, *Linear time low treewidth partitions and algorithmic consequences*, in: Proc. STOC'06, 391–400.

[21] J. Nešetřil and P. Ossona de Mendez, *Grad and classes with bounded expansion I. Decompositions.*, Eur. J. Comb. 29 (2008), 760–776.

[22] J. Nešetřil and P. Ossona de Mendez, *Grad and classes with bounded expansion II. Algorithmic aspects.*, Eur. J. Comb. 29 (2008), 777–791.

[23] J. Nešetřil and P. Ossona de Mendez, *Grad and classes with bounded expansion III. Restricted graph homomorphism dualities.*, Eur. J. Comb. 29 (2008), 1012–1024.

[24] J. Nešetřil and P. Ossona de Mendéz, On nowhere dense graphs, manuscript, 2008.

[25] J. Nešetřil, P. Ossona de Mendez and D. Wood, *Characterisations and Examples of Graph Classes with Bounded Expansion*, preprint (arXiv:0902.3265v2).

[26] J. Nešetřil and P. Ossona de Mendez, *Structural properties of sparse graphs*, in: M. Grötschel, G. O. H. Katona (eds.): Building Bridges Between Mathematics and Computer Science, Bolyai Society Mathematical Studies vol. 19, Springer, 2008.

[27] N. Roberson and P. D. Seymour, *Graph minors. XIII: the disjoint paths problem*, J. Combin. Theory Ser. B 63 (1995), 65–110.

[28] D. Wood, *On the maximum number of cliques in a graph*, Graphs Combin. 23 (2007), 337–352.