# 1 Preliminaries

The set of positive integers (including 0) is denoted by $\mathbb{N}$ and for a positive integer $n$, the set $\{1,\ldots,n\}$ of integers is denoted as $[n]$. For $m,n \in \mathbb{N}$, we write $[\![m,n]\!]$ for the interval $\{m,\ldots,n\}$. For a set $V$ and $x \in V$, the singleton $\{x\}$ shall be often written simply as $x$. The power set of a finite set $V$ is denoted by $2^V$ and we write $|V|$ to denote the size of $V$. A partition of a set $V$ is a collection $\{V_1,\ldots,V_n\}$ of non-empty and pairwise non-intersecting subsets of $V$, called *blocks*, such that $\bigcup_{1\leq i\leq n} V_i = V$. For an equivalence relation $\equiv$ on $V \times V$, we denote by $V/\equiv$ the set of equivalence classes of $\equiv$ and write $[Y]_\equiv$ to denote the equivalence class of $Y \in V$. Recall that the set of equivalence classes forms a partition.

A *set system* $\mathbf{S}$ is a pair $(S,\mathcal{S})$ where $S$ is a finite set and $\mathcal{S}$ is a collection of subsets of $S$. We refer to $S$ as the *ground set*, and members of $\mathcal{S}$ as *hyperedges*. We use boldface capital letters to denote set systems, *e.g.*, $\mathbf{S}$, $\mathbf{M}$; capital letters for ground sets, *e.g.*, $S$, $M$; and calligraphic letters for set of hyperedges, *e.g.*, $\mathcal{S}$, $\mathcal{M}$. We follow [6] for our graph terminology. For a graph $\mathsf{G}$, we denote by $V(\mathsf{G})$ its vertex set, and by $E(\mathsf{G})$ its edge set; an edge between $x$ and $y$ in an undirected graph is denoted by $xy$ (equivalently $yx$). It is common to call vertices of a tree *nodes*.

We are going to prove the following.

▶ **Theorem 1.** *Let $k$ and $q$ be positive integers. There is an elementary function $f$ such that every first-order formula $\varphi$ of quantifier-rank $q$ can be checked in time $f(k,q)\cdot poly(n)$ in graphs of linear clique-width at most $k$.*

We organise this section as follows. The notion of *linear clique-width* is introduced in Section 1.1, while *first-order logic* and *Feferman-Vaught Theorem* are introduced in Section 1.2.

## 1.1 Linear clique-width

We will follow [1] for the definition of *linear clique-width* as we will use their semi-group structure. If $k$ is a positive integer, a graph $\mathbf{G}$ is said *$k$-labeled* if every vertex of $\mathbf{G}$ receives a label from $[k]$, and each vertex of $\mathbf{G}$ labeled $i$ is called an *$i$-labeled vertex*. The labeling function of a $k$-labeled graph is denoted by $\alpha_{\mathbf{G}}$. The following operations are defined on $k$-labeled graphs:

**Relabeling operation** For every function $f:[k]\to[k]$, let $\rho_f$ be the operation that takes as input a $k$-labeled graph $\mathbf{G}$ and outputs the $k$-labeled graph $\mathbf{G}$ with labeling function $f\circ\alpha_{\mathbf{G}}$.

**Join operation** For every symmetric subset $S$ of $[k]\times[k]$, let $\otimes_S$ be the binary operation that takes as inputs two $k$-labeled graphs $\mathbf{G}$ and $\mathbf{H}$ and outputs the $k$-labeled graph $\mathbf{K}$ where $\alpha_{\mathbf{K}} = \alpha_{\mathbf{G}} \cup \alpha_{\mathbf{H}}$, and $\mathbf{K}$ is obtained from the disjoint union of $\mathbf{G}$ and $\mathbf{H}$ and adding all edges in the set $\{xy \mid x\in G, y\in H, (\alpha_{\mathbf{G}}(x),\alpha_{\mathbf{H}}(y))\in S\}$. We denote $\mathbf{K}$ as $\mathbf{G}\otimes_S\mathbf{H}$.

**Constant** For every $i\in[k]$, let $\mathsf{i}$ be the $k$-labeled graph with a single vertex labeled $i$ and no edge.

**Adding a vertex** For every $i\in[k]$ and $X\subseteq[k]$, let $\mathsf{a}_{i,X}$ be the operation that takes as input a $k$-labeled graph $\mathbf{G}$ and outputs $\mathbf{G}\otimes_S\mathsf{i}$ with labeling function $\alpha_{\mathbf{G}}\cup\alpha_{\mathsf{i}}$ where $S=\{i\}\times X\cup X\times\{i\}$.

Let $\mathbf{LCW}_k$ be the alphabet $\{\mathsf{a}_{i,X} \mid i\in[k], X\subseteq[k]\}\cup\{\rho_f \mid f:[k]\to[k]\}$. A *width-$k$ linear clique-width expression* is a word over the alphabet $\mathbf{LCW}_k$. Every width-$k$ linear clique-width expression $w$ can be evaluated inductively into a $k$-labeled graph, denoted by $\mathbf{val}(w)$, as follows:

- $\mathbf{val}(\rho_f)$ is the empty-graph,
- $\mathbf{val}(\mathsf{a}_{i,X})$ is the $k$-labeled graph $\mathsf{i}$,
- $\mathbf{val}(u\rho_f)$ is the $k$-labeled graph $\rho_f(\mathbf{val}(u))$,
- $\mathbf{val}(u\,\mathsf{a}_{i,X})$ is the $k$-labeled graph $\mathsf{a}_{i,X}(\mathbf{val}(u))$.

The linear clique-width of a graph $\mathbf{G}$, denoted by $\mathsf{lcw}(\mathbf{G})$, is the least $k$ such that there is a word $w$ in $\mathbf{LCW}_k$ with $\mathbf{G}$ is isomorphic to $\mathbf{val}(w)$ after forgetting the labels of $\mathbf{val}(w)$.

## 1.2 First-order logic

We refer to [4] for a complete presentation of FO logic, and we shortly introduce it now. Define a *vocabulary* to be a finite set of relation names, each one being associated with an *arity* in $\mathbb{N}$. A *relational structure* $\mathbb{A}$ over the vocabulary $\Sigma$ (*$\Sigma$-structure* for short) consists in a set $A$, called the *universe*, and for each relation name $\mathsf{R} \in \Sigma$, a relation $\mathsf{R}^{\mathbb{A}} \subseteq A^k$ with $k$ the arity of $\mathsf{R}$.

Let $\mathcal{V}$ be a countable set of variables, each being either a variable ranging over individual elements of the universes, called an *FO variable*, and use lower-case letters to denote them. The *atomic formulas* are $x = y$ and $\mathsf{R}(x_1, \dots, x_k)$ where $\mathsf{R}$ is a $k$-ary relation name of $\Sigma$, $x_1, \dots, x_k$ are FO variables. An *FO formula* over $\Sigma$ is either an atomic formula, or it is of the inductive form $\neg\varphi$, $\varphi \vee \psi$, $\exists x\varphi$, where $\varphi$ and $\psi$ are FO formulas. We also use the classical syntactic sugars $x \neq y$, $\forall x\varphi$, $\varphi \wedge \psi$, $\varphi \to \psi$, and $\varphi \leftrightarrow \psi$ for the formulas $\neg(x = y)$, $\neg\exists x\neg\varphi$, $\neg(\neg\varphi \wedge \neg\psi)$, $\neg\varphi \vee \psi$, and $(\varphi \to \psi) \wedge (\psi \to \varphi)$, respectively.

A variable is *free* in a formula if it is not bound by a quantifier ($\exists$ or $\forall$). We write $\varphi(x_1, \dots, x_p)$ to say that $x_1, \dots, x_p$ are among the free variables of $\varphi$. An *FO sentence* is an FO formula without free variables. The size of a formula $\varphi$ is simply defined as the number of symbols in it and is denoted by $|\varphi|$. The *quantifier-rank* of an FO formula $\varphi$, denoted by $\mathsf{qr}(\varphi)$, is defined inductively as follows:

$$\mathsf{qr}(\varphi) = \begin{cases} 0 & \text{if } \varphi \text{ is atomic,} \\ \mathsf{qr}(\psi) & \text{if } \varphi = \neg\psi, \\ \max\{\mathsf{qr}(\psi_1), \mathsf{qr}(\psi_2)\} & \text{if } \varphi = \psi_1 \vee \psi_2, \\ 1 + \mathsf{qr}(\psi) & \text{if } \varphi = \exists x\psi. \end{cases}$$

An FO formula is *quantifier-free* if its quantifier-rank is 0. We denote by $\mathrm{FO}(\Sigma)$ the set of first-order formulas over $\Sigma$, and by $\mathrm{FO}^q(\Sigma)$ the set of first-order sentences of quantifier-rank at most $q$. We simply write FO or $\mathrm{FO}^q$ when $\Sigma$ is clear from the context. We denote by $\mathrm{FO}^q[t]$ the set of FO formulas with quantifier-rank at most $q$ and having at most $t$ free variables.

Let $\mathbb{A}$ be a $\Sigma$-structure and $\varphi$ be an FO formula. A *$\mathcal{V}$-valuation on $\mathbb{A}$* is a mapping $\nu$ that assigns to each FO variable of $\mathcal{V}$ an element of $A$. We say that $(\mathbb{A}, \nu)$ *models* $\varphi$, denoted by $(\mathbb{A}, \nu) \models \varphi$, when one of the following cases holds: $\varphi$ is $\mathsf{R}(x_1, \dots, x_k)$ for some relation name $\mathsf{R}$ of arity $k$ and $(\nu(x_1), \dots, \nu(x_k)) \in \mathsf{R}^{\mathbb{A}}$; $\varphi$ is $x = y$ and $\nu(x) = \nu(y)$; $\varphi$ is $\varphi_1 \vee \varphi_2$ and $(\mathbb{A}, \nu)$ models both $\varphi_1$ or $\varphi_2$; $\varphi$ is $\exists x\psi$ and there exists a $\mathcal{V}$-valuation $\nu'$ on $\mathbb{A}$ such that $(\mathbb{A}, \nu') \models \psi$ and $\nu$ and $\nu'$ agree on all variable names other than $x$. We say that $\mathbb{A}$ *models a formula* $\varphi$, denoted by $\mathbb{A} \models \varphi$, if $(\mathbb{A}, \nu) \models \varphi$ for some $\mathcal{V}$-valuation $\nu$ on $\mathbb{A}$. If $\varphi(x)$ is a formula with $x$ a free variable, then for a structure $\mathbb{A}$ and $u \in A$, we write $\varphi[u/x]$ to mean that any $\mathcal{V}$-valuation on $\mathbb{A}$ we will consider for $\varphi$ will map $x$ to $u$.

For a vocabulary $\Sigma$, let us denote by $\mathbf{S}^\Sigma$ the set of relational structures over the vocabulary $\Sigma$. A *class of relational structures over $\Sigma$* is a subset $\mathcal{C}$ of $\mathbf{S}^\Sigma$ which is closed under isomorphism.

If $\mathcal{L}$ is a set of FO formulas, we define the *$\mathcal{L}$-theory* of a $\Sigma$-structure $\mathbb{A}$, denoted by $\mathsf{Th}_{\mathcal{L}}(\mathbb{A})$, as the set of formulas in $\mathcal{L}$ that $\mathbb{A}$ models. It is worth mentionning that $\mathsf{Th}_{\mathcal{L}}(\mathbb{A}) = \mathsf{Th}_{\mathcal{L}}(\mathbb{B})$ whenever $\mathbb{A}$ is isomorphic to $\mathbb{B}$.

## 2 Proof of Theorem 1

As we have seen above, one can associate with each graph $\mathbf{G}$ of linear clique-width at most $k$ a word in $\mathbf{LCW}_k^+$ that can be evaluated into $\mathbf{G}$. Using the semi-group homomorphism $h$, defined in [1], that maps every word in $\mathbf{LCW}_k^+$ into an element of a semi-group of size at most $2^{2^{O(k)}}$, we obtain from Simon's Factorisation Forest Theorem that every graph $\mathbf{G}$ of linear clique-width at most $k$ admits a tree-like decomposition of height, called *$h$-rank of* $\mathbf{G}$, at most $3 \cdot 2^{2^{O(k)}}$. We will prove, by induction, that any FO-formula of quantifier-rank $q$ can be solved in time $f(q) \cdot poly(n)$, where $f$ is a tower of exponentials of height depending only on the $h$-rank of $\mathbf{G}$. We introduce Simon's Factorisation Forest Theorem in Section 2.1, the semi-group homomorphism in Section 2.2, an upper-bound on the number of $\mathrm{FO}^q$-theories based on the structure of the Simon's Forest Factorisation in Section 2.3, and the algorithm in Section 2.4 which uses Colcombet's deterministic algorithm for computing Simon's Forest Factorisation.

### 2.1 Simon's forest factorisation theorem

Remind that a *semi-group* is a set $S$ equipped with an associative binary operation. Notice also that $A^*$ is the set of finite words over the alphabet $A$, while $A^+$ is the set of non-empty finite words over $A$, and each equipped with concatenation $\cdot$ is a semi-group. An *idempotent element* in a semi-group $(S, \circ)$ is an element $e$ such that $e \circ e = e$. For two semi-groups $(S_1, \circ_1)$ and $(S_2, \circ_2)$, a *semi-group homomorphism* is a function $h : S_1 \to S_2$ such that $h(x \circ_1 y) = h(x) \circ_2 h(y)$.

Let $(S, \circ)$ be a semi-group and $A$ an alphabet. For a semi-group homomorphism $h : A^+ \to S$, an *$h$-factorisation* of a word $w \in A^*$ is a sequence $(w_1, \ldots, w_n)$ such that

1. $w = w_1 \cdot w_2 \cdot \cdots \cdot w_n$,
2. $|w_i| < |w|$ for all $i \in [n]$, and
3. $h(w_1) = h(w_2) = \cdots = h(w_n)$ is idempotent if $n \geq 3$.

The *$h$-rank* of a word $w \in A^*$ is defined inductively as follows : single letters have $h$-rank 1, and for every $w \in A^*$ of length at least 2, its $h$-rank is

$$1 + \min_{(w_1,\ldots,w_n) \text{ is an } h\text{-factorisation of } w} \left( \max_{1 \leq i \leq n} \{h\text{-rank of } w_i\} \right).$$

Imre Simon proved in [14] that the $h$-rank of any word is upper-bounded by a function on the size of the target semi-group, which we refer below with the improvement given in [9].

▶ **Theorem 2** (Simon's Forest Factorisation Theorem [9]). *Let $S$ be a finite semi-group and let $h : A^* \to S$ be a semi-group homomorphism. Then, every word $w \in A^+$ has h-rank at most* $3 \cdot |S|$.

### 2.2 A semi-group for words in $\mathrm{LCW}_k$

Our proof will be an induction based on the $h$-rank of words in $\mathbf{LCW}_k^+$, for some semi-group homorphism $h$. Let's define this semi-group homomorphism borrowed from [1].

A *$k$-derivation* is a triple $\sigma = (\mathbf{G}, \lambda, \gamma)$ where

- $\mathbf{G}$ is a $k$-labeled graph, called *underlying graph* of $\sigma$,
- $\lambda : \mathbf{G} \to 2^{[k]}$ assigns to each vertex $x$ of $G$ its *profile*,
- $\gamma : [k] \to [k]$ is a *relabeling* function.

An *atomic $k$-derivation* is a $k$-derivation whose underlying graph has at most one vertex.

The composition of two $k$-derivations $\sigma_1 = (\mathbf{G}_1, \lambda_1, \gamma_1)$ and $\sigma_2 = (\mathbf{G}_2, \lambda_2, \gamma_2)$, denoted by $\sigma_1 \otimes \sigma_2$, is the $k$-derivation $\sigma$ obtained as follows :

- the underlying graph of $\sigma$ is the graph obtained from the disjoint union of $\rho_{\gamma_2}(\mathbf{G}_1)$ and $\mathbf{G}_2$ where we add an edge between a vertex $x$ of $\mathbf{G}_1$ and a vertex $y$ of $\mathbf{G}_2$ whenever $\alpha_{\mathbf{G}_1}(x) \in \lambda_2(y)$. Notice that the labeling function of the underlying graph of $\sigma$ is $\gamma_2 \circ \alpha_{\mathbf{G}_1} \cup \alpha_{\mathbf{G}_2}$.
- The profile of $\sigma$ is $\lambda_1 \cup \gamma_1^{-1} \circ \lambda_2$.
- The relabeling function of $\sigma$ is $\gamma_2 \circ \gamma_1$.

As claimed in [1] it is not hard to see that $\otimes$ is associative, and so the set of $k$-derivations equipped with the composition operation $\otimes$ is a semi-group. Let $S_k$ be the semi-group generated by the set of atomic $k$-derivations. It is not hard to see that $S_k$ is finitely generated. The following proved in [1] is a reformulation of width-$k$ linear clique-width expressions.

▶ **Lemma 3** ([1, Lemma 4.2]). *If $\mathbf{G}$ has linear clique-width at most $k$, then it is the underlying graph of a $k$-derivation from $S_k$.*

Let $\sigma = (\mathbf{G}, \lambda, \gamma)$ be a $k$-derivation. For $c = (i, X) \in [k] \times 2^{[k]}$, we call the set of $i$-labeled vertices of $\mathbf{G}$ with profile $X$ a *$c$-class* and denote it by $\sigma[c]$. Let $\mathcal{C}_k$ denote the set $\{(i, X) \in [k] \times 2^{[k]}\}$, that we call for simplicity *classes*. We are now ready to define the finite semi-group $T_k$, called *abstraction semi-group*, which is a substructure of the abstraction semi-group defined in [1].

▶ **Definition 4.** *The* abstraction *of a $k$-derivation $\sigma$, denoted by $[\sigma]$, is the triple $(L, \gamma)$ where:*

- *$L$ is the set $\{c \in \mathcal{C}_k \mid \sigma[c] \neq \emptyset\}$,* i.e., *the set of non-empty $c$-classes.*
- *$\gamma$ is the relabeling function of $\sigma$.*

The following now summarises the semi-group structure of $T_k$, the set of abstractions of $k$-derivations and is corrolary of the fact that "having the same abstraction" is a congruence in $S_k$.

▶ **Lemma 5** ([1]). *There is an associative operation $[\tilde{\otimes}]$ such that $[\sigma_1 \otimes \sigma_2] = [\sigma_1]\,[\tilde{\otimes}][\sigma_2]$. Moreover, the set $T_k$ has size at most $2^{2^{O(k)}}$.*

The induction will be on $k$-derivations, and so we will for simplicity use first-order logic on $k$-derivations instead of graphs. We will consider each $k$-derivation $\sigma = (\mathbf{G}, \lambda, \gamma)$ as the relational structure over the vocabulary $\mathsf{edg}$ representing the edge relation of the underlying graph $\mathbf{G}$, $k$ constants $\mathsf{c}_1, \ldots, \mathsf{c}_k$ representing the set $[k]$ and disjoint from the vertex set of the underlying graph, the predicate $\mathsf{P}_c$, for a class $c \in \mathcal{C}_k$, where $P_c(x)$ holds if $x$ is a vertex and belongs to the $c$-class, and binary predicate $\rho$ representing the relabeling function $\gamma$ (we can add the axiom that $\rho$ is a function on every formula using $\rho$). We will need the following which is straighforward.

▶ **Lemma 6.** *Let $k$ and $q$ be positive integers. If the two $k$-derivations $\sigma_1$ and $\sigma_2$ are such that $\mathsf{Th}_{FO^q}(\sigma_1) = \mathsf{Th}_{FO^q}(\sigma_2)$, then $[\sigma_1] = [\sigma_2]$.*

**Proof.** If $c \in \mathcal{C}_k$ is a class such that $\sigma_1[c] \neq \emptyset$, but $\sigma_2[c] = \emptyset$, then the formula $\mathsf{P_c}(x)$ will be satisfied by $\sigma_1$, but not $\sigma_2$. One checks in a similar way that they have the same relabeling function. ◀

It is well-known from Feferman-Vaught Theorem [7] that the FO-theory of a generalised product of two structures can be computed from the FO-theories of the two operands, where examples of generalised products are quantifier-free transductions [11]. We refer to [4] for the definition of transductions, however it is not hard to prove that the composition operation of $k$-derivations is a quantifier-free transduction.

▶ **Observation 7.** *Let $k$ be a positive integer. There is a quantifier-free transduction $\tau$ on the vocabulary of $k$-derivations such that $\sigma_1 \otimes \sigma_2 = \tau(\sigma_1 \oplus \sigma_2)$, for every two $k$-derivations $\sigma_1$ and $\sigma_2$.*

We can therefore state the following version of Feferman-Vaught Theorem for the $\otimes$ operation. We refer to [7, 11] for more information.

▶ **Theorem 8** ([7, Theorem 5.4])**.** *Let $s$ and $q$ be positive integers. Then, for every sequence $\sigma_1, \ldots, \sigma_s$ of $k$-derivations, $\mathsf{Th}_{FO^q}(\sigma)$ depends only on $\mathsf{Th}_{FO^q}(\sigma_1), \ldots, \mathsf{Th}_{FO^q}(\sigma_s)$.*

For a positive integer $q$, we write $\sigma_1 \equiv_q \sigma_2$ if $\mathsf{Th}_{\mathrm{FO}^q}(\sigma_1) = \mathsf{Th}_{\mathrm{FO}^q}(\sigma_2)$. Notice that $\equiv_q$ is an equivalence relation, and by Lemma 6, if $\sigma_1 \equiv_q \sigma_2$, then $[\sigma_1] = [\sigma_2]$. We can derive the following as a corollary of Theorem 8.

▶ **Lemma 9.** *Let $k$ and $q$ be positive integers. If $\sigma_1$ and $\sigma_2$ are two $\equiv_q$-equivalent $k$-derivations, then, for every two $k$-derivations $\sigma_l$ and $\sigma_r$, it holds that $\sigma_l \otimes \sigma_1 \otimes \sigma_r \equiv_q \sigma_l \otimes \sigma_2 \otimes \sigma_r$.*

**Proof.** By Theorem 8, the $\equiv_q$-equivalence class of $\sigma_l \otimes \sigma_1 \otimes \sigma_r$ depends only on the $\equiv_q$-equivalence classes of $\sigma_l$, $\sigma_1$ and $\sigma_r$. Since $\sigma_1$ and $\sigma_2$ are $\equiv_q$-equivalent, the statement follows by Theorem 8. ◀

## 2.3 An upper-bound on the number of FO$^q$-theories

Let $h_k : S_k \to T_k$ be the semi-group homomorphism described in Section 2.2. Let $f_1(q) = 2^{k \cdot 2^k \cdot k^k}$, and for every $\ell > 1$, let $f_\ell(q) = 2^{2^{2 \cdot (f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}}}$. It is not hard to check that for every $\ell$, $f_\ell(q)$ is a tower of 2 whose height depends only on $\ell$. We are going to prove the following, which combines with Lemma 3 and Theorem 2 implies Theorem 1.

▶ **Proposition 10.** *Let $q$ be a fixed positive integer. The number of $\equiv_q$-equivalence classes on $k$-derivations of $h_k$-rank at most $\ell$ is upper-bounded by $f_\ell(q)$.*

The proof will be by induction on the $h_k$-rank of $k$-derivations of $S_k$, and we follow the structure of Simon's factorisation.

**Basic case.** By definition, every atomic $k$-derivation has $h_k$-rank 1. Because the underlying graph of each atomic $k$-derivation is a single vertex, an atomic $k$-derivation is obtained by choosing a label, the profile of the single vertex and a labeling function. So, the number of atomic $k$-derivations is upper-bounded by $k \cdot 2^k \cdot k^k$. Since each satisfied FO-formula corresponds to a family of non-isomorphic substructure, each FO$^q$-theory corresponds to a family of family of substructures. So, the number of equivalence classes of $\equiv_q$ on atomic $k$-derivations is upper-bounded by $f_1(q) = 2^{2^{k \cdot 2^k \cdot k^k}}$.

**Binary factorisation.** Assume that a $k$-derivation $\sigma$ of $h_k$-rank $\ell$ is equal to $\sigma_1 \otimes \sigma_2$ and the $h_k$-rank of both $\sigma_1$ and of $\sigma_2$ is at most $\ell - 1$. From Theorem 8, one can decide $\mathsf{Th}_{\mathrm{FO}^q}(\sigma)$ from $\mathsf{Th}_{\mathrm{FO}^q}(\sigma_1)$ and $\mathsf{Th}_{\mathrm{FO}^q}(\sigma_2)$, *i.e.*, each equivalence class of $\equiv_q$ on $k$-derivations of $h_k$-rank at most $\ell$ admitting a binary factorisation is a subset of pairs of equivalence classes on $k$-derivations of $h_k$-rank at most $\ell - 1$, whose number is upper-bounded by $2^{2^{f_{\ell-1}(q)\cdot f_{\ell-1}(q)}}$.

**Unranked factorisation.** Assume now that a $k$-derivation $\sigma = (\mathbf{G}, \lambda, \gamma)$ of $h_k$-rank $\ell$ is equal to $\sigma_1 \otimes \sigma_2 \otimes \cdots \otimes \sigma_n$ with $h_k(\sigma_1) = h_k(\sigma_2) = \cdots = h_k(\sigma_n) = h_k(\sigma) = (L, \gamma)$ is idempotent, and, for each $i \in [n]$, the $h_k$-rank of $\sigma_i$ is at most $\ell - 1$. Notice that because $h_k(\sigma)$ is idempotent, the function $\gamma$ is also idempotent (on the semi-group of functions $[[k] \to [k]]$), and then by the definition of the operation $\otimes$, the relabeling function of each $k$-derivation $\sigma_i$, for $i \in [n]$, is $\gamma$. So, let's denote $\sigma_s$ by $(\mathbf{G}_s, \lambda_s, \gamma)$, for each $s \in [n]$. As noticed in [1], we have $\gamma(i) = i$, for each $i \in \gamma([k])$. Similarly, a $c$-class is non-empty in $\sigma_s$ if and only if it is non-empty in $\sigma_t$ if and only if it is non-empty in $\sigma$, for all $s, t \in [n]$. Since all the $\sigma_i$'s have the same relabeling function, by Lemma 9, we can derive the following.

▶ **Lemma 11.** *The two $k$-derivations $\sigma$ and $\mathcal{R}_1 \otimes \mathcal{R}_2 \otimes \cdots \otimes \mathcal{R}_n$ are $\equiv_q$-equivalent where $\mathcal{R}_s$ is any representative of the $\equiv_q$-equivalence class of $\sigma_s$, for each $s \in [n]$.*

Let $\mathcal{A}^q_{k,\ell}$ be an alphabet where letters are in one-to-one correspondence with a fixed set of representatives of the $\equiv_q$-equivalence classes on $k$-derivations of $h_k$-rank at most $\ell - 1$. By inductive hypothesis, the size of $\mathcal{A}^q_{k,\ell}$ is upper-bounded by $f_{\ell-1}(q)$. Because each $\sigma_s$, for $s \in [n]$ has $h_k$-rank at most $\ell - 1$, we can replace the word $\sigma_1\sigma_2\cdots\sigma_n$ by the word $w(\sigma) = w_1 w_2 \cdots w_n$ on $\mathcal{A}^q_{k,\ell}$, where, for each $s \in [n]$, $w_s$ is the $\equiv_q$-representative of $\sigma_s$ in $\mathcal{A}^q_{k,\ell}$. By Lemma 11, the two $k$-derivations $\sigma$ and $w_1 \otimes w_2 \otimes \cdots \otimes w_n$ are $\equiv_q$-equivalent. Our goal is to prove that if $n > 2q$, then there is a set $\mathcal{F}$ of words $w_1 w_2 \cdots w_r$ of length at most $2q$ on the alphabet $\mathcal{A}^q_{k,\ell}$ and such that the $\equiv_q$-equivalence class of $\sigma$ can be derived from the $\equiv_q$-equivalence classes of words in the set $\mathcal{F}$. Let's add the new letter $\emptyset^\gamma_\lambda$ as the triple $(\emptyset, \lambda, \gamma)$, and for every $k$-derivation $\sigma' = (\mathbf{G}', \lambda', \gamma')$, the $k$-derivation $\sigma' \otimes \emptyset^\gamma_\lambda$ is the $k$-derivation $(\rho_\gamma(\mathbf{G}'), \lambda', \gamma \circ \gamma')$.

For a word $w = w_1 \cdots w_r$ on $\mathcal{A}^q_{k,\ell} \cup \{\emptyset^\gamma_\lambda\}$, an FO formula $\theta$ and a $\mathcal{V}$-valuation $\nu$ on $w_1 \otimes w_2 \otimes \cdots \otimes w_r$, we denote by $b^{\theta,\nu}_w$ the predicate $(w_1 \otimes w_2 \otimes \cdots \otimes w_r, \nu) \models \theta$.

▶ **Lemma 12.** *Let $q$ and $t$ be positive integers and let $\varphi$ be an FO-formula in FO$^q[t]$. If $n \geq 2(q+t)+1$, then, for every $\mathcal{V}$-valuation $\nu$ on $\sigma$ of the $t$ variables of $\varphi$, there is a set of $\mathcal{F}_\varphi$ of pairs $(w, \theta)$ and a boolean function $B_\varphi$ using the predicates $\{b^{\theta,\nu}_w \mid (w, \theta) \in \mathcal{F}_\varphi\}$, where each $w$ is a word on $\mathcal{A}^q_{k,\ell} \cup \{\emptyset^\gamma_\lambda\}$ and is of length at most $2(q+t)$ and $\theta$ a sub-formula of $\varphi$, such that*

$$(\sigma, \nu) \models \varphi \qquad \textit{if and only if} \qquad B_\varphi \textit{ is satisfiable.}$$

**Proof.** The proof is by induction on $|\varphi| + q$ and follows the structure of FO formulas.

1. If $\varphi$ is quantifier-free, *i.e.*, $q = 0$, then the satisfiability of $\varphi$ depends only on the $k$-derivations in the Forest Factorisation of $\sigma$ that contain $\nu(x)$, for $x$ a free variable in $\varphi$. Let $\sigma_{i_1}, \ldots, \sigma_{i_r}$ be the $k$-derivations that contain $\nu(x)$, for all free variables $x$ of $\varphi$ (with $r \leq t$). Then, let $w$ be obtained from $w(\sigma)$ by removing the letters whose indices are not in $\{i_1, \ldots, i_r\}$ and by adding $\emptyset^\gamma_\lambda$ between $w_{i_j}$ and $w_{i_{j+1}}$ when $i_j < i_{j+1} - 1$, *i.e.*, $i_j$ and $i_{j+1}$ are not consecutive, and possibly after $w_{i_r}$ if $i_r \neq n$. Let $\mathcal{F}_\varphi = \{(w, \varphi)\}$ and $B_\varphi = b^{\varphi,\nu}_w$. It is straigforward to check that $(\sigma, \nu)$ satisfies $\varphi$ if and only if $B_\varphi$ holds.
2. If $\varphi = \neg\psi$, then by inductive hypothesis there are a set $\mathcal{F}_\psi$ and a boolean function $B_\psi$ on $\{b^{\theta,\nu}_w \mid (w, \theta) \in \mathcal{F}_\psi\}$ such that $(\sigma, \nu)$ satisfies $\psi$ if and only if $B_\psi$ is satisfied, *i.e.*, $(\sigma, \nu)$ satisfies $\neg\psi$ if and only if $\neg B_\psi$ is satisfied. We therefore let $\mathcal{F}_\varphi = \mathcal{F}_\psi$ and $B_\varphi = \neg B_\psi$.

3. If $\varphi = \psi_1 \vee \psi_2$, then by inductive hypothesis, there are, respectively, sets $\mathcal{F}_1$ and $\mathcal{F}_2$, and boolean functions $B_1$ and $B_2$ on, respectively, $\{b_w^{\theta,\nu} \mid (w,\theta) \in \mathcal{F}_1\}$ and $\{b_w^{\theta,\nu} \mid (w,\theta) \in \mathcal{F}_2\}$, such that $(\sigma, \nu)$ satisfies $\psi_1$ (resp. $\psi_2$) if and only if $B_1$ (resp. $B_2$) is satisfied. Let $\mathcal{F}_\varphi = \mathcal{F}_1 \cup \mathcal{F}_2$ and $B_\varphi = B_1 \vee B_2$. We can thus conclude by inductive hypothesis that $(\sigma, \nu)$ satisfies $\varphi$ if and only if $B_\varphi$ is satisfied.
4. Assume finally that $\varphi = \exists x \psi$. For each vertex $u \in V(\mathbf{G})$, let $\nu_u$ be the $\mathcal{V}$-valuation on $\sigma$ where $\nu_u(x) = u$ and $\nu_u(y) = \nu(y)$ for every other free variable in $\psi$. By inductive hypothesis, there are a set $\mathcal{F}_u$ and a boolean function $B_u$ on $\{b_w^{\theta,\nu'} \mid (w,\theta) \in \mathcal{F}_u\}$ such that $(\sigma, \nu')$ satisfies $\psi$ if and only if $B_u$ is satisfied. Let $\mathcal{F}_\varphi = \bigcup_{u \in V(\mathbf{G})} \mathcal{F}_u$ and $B_\varphi = \vee_{u \in V(\mathbf{G})} B_u$. We therefore have by inductive hypothesis that $(\sigma, \nu)$ satisifies $\varphi$ if and only if $B_\varphi$ is satisfied.

◀

The following shows that we can replace $\sigma$ by a finite number of $k$-derivations so that the $\equiv_q$-equivalence of $\sigma$ can be derived from their $\equiv_q$-equivalence classes.

▶ **Lemma 13.** *Let $q$ be a positive integer. There are at most $(f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}$ pairs $(w, \theta)$ with $w$ a word of length at most $2q$ on $\mathcal{A}_{k,\ell}^q \cup \{\emptyset_\lambda^\gamma\}$ and $\theta$ a formula in $FO^{q'}[t]$ with $q' + t \leq q$.*

**Proof.** First, The number of words on the alphabet $\mathcal{A}_{k,\ell}^q \cup \{\emptyset_\lambda^\gamma\}$ and of length at most $2q$ is upper-bounded by $(f_{\ell-1}(q)+1)^{2q}$. Now, each $\theta$ in $\mathrm{FO}^{q'}[t]$ with $q' + t \leq q$ is also a formula in $\mathrm{FO}^q$ by replacing $\theta$ by $\exists x_1 \cdots x_t \theta$. For each such word $w = w_1 \cdots w_r$, let $\sigma(w)$ be $w_1 \otimes w_2 \otimes \cdots \otimes w_r$. By applying Theorem 8, we obtain that the number of distinct $\mathrm{FO}^q$-theories among such $\sigma(w)$ is upper-bounded by $2^{(2^{f_{\ell-1}(q)})^{2q}}$. Therefore, the number of such pairs is upper-bounded by $(f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}$. ◀

We can now give an upper-bound on the number of $\equiv_q$-equivalence classes.

**Proof of Proposition 10.** By Lemma 12, for every $k$-derivation $\sigma$ of $h_k$-rank $\ell$ and admitting a Forest Factorisation into $\sigma_1 \cdots \sigma_n$, either $n \leq 2q$ or, for every FO sentence $\varphi$ in $\mathrm{FO}^q$, there are a family $\mathcal{F}_\varphi$ of pairs $(w, \theta)$ with $\theta$ a sentence in $\mathrm{FO}^q$, $w$ a word of length at most $2q$ on $\mathcal{A}_{k,\ell}^q \cup \{\emptyset_\lambda^\gamma\}$ and a boolean function $B_\varphi$ on $\{b_w^\theta \mid (w,\theta) \in \mathcal{F}_\varphi\}$ such that $\sigma$ satisfies $\varphi$ if and only if $B_\varphi$ is satisfied. By Lemma 13, the number of such pairs is upper-bounded by $(f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}$. Let's denote by $\mathcal{T}$ the set of such pairs and let $\mathcal{B}$ be the set of boolean functions on subsets of $\mathcal{T}$. We can conclude that every $\mathrm{FO}^q$ sentence satisfied by such a $\sigma$ is associated with a a boolean function in $\mathcal{B}$. It is worth noticing that the satisfaction of any boolean function in $\mathcal{B}$ does depend only on the $\mathrm{FO}^q$-theories of $k$-derivations of $h_k$-rank at most $\ell - 1$. Therefore, we can consider that for every $k$-derivation $\sigma$, its $\mathrm{FO}^q$-theory is the set

$$\{B_\varphi \in \mathcal{B} \mid B_\varphi \text{ is satisfied and } B_\varphi \text{ is associated with } (\sigma, \varphi)\}.$$

Now, since the number of boolean functions on $p$ variables is upper-bounded by $2^p$, the set $\mathcal{B}$ is then upper-bounded by $2^{2 \cdot (f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}}$, *i.e.*, the number of possible $\mathrm{FO}^q$-theories is upper-bounded by $2^{2^{2 \cdot (f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}}}$. ◀

## 2.4 The algorithm

It is proved in [8] that if a graph has linear clique-width at most $k$, then one can compute in time $2^{O(k^3)} \cdot n^3$ a width-$p$ linear clique-width expression with $p \leq 2^k + 1$, *i.e.*, if a graph $\mathbf{G}$ has linear clique-width at most $k$, then for some $p \leq 2^k + 1$, one can construct in time $2^{O(k^3)} \cdot n^3$ a word $w$ using atomic $p$-derivations in $S_p$, evaluating into a $p$-derivation $\sigma$ having $\mathbf{G}$ as underlying graph. Moreover, any $\mathrm{FO}^q$-sentence $\varphi$ on $\mathbf{G}$ can be translated into an $\mathrm{FO}^q$-sentence $\varphi^\#$ on $\sigma$ with the property that $\mathbf{G}$ satisfies $\varphi$ if and only if $\sigma$ satisfies $\varphi^\#$. The following version of Simon's Factorisation Forest Theorem was proved in [2] and allows to compute the Simon's Forest Factorisation.

▶ **Theorem 14** ([2]). *Let $\Sigma$ be an alphabet. For every finite semi-group $S$ and every semi-group homomorphism $h : \Sigma^+ \to S$, one can construct a deterministic finite state automata that takes as input a word $w$ in $\Sigma^+$ and outputs in time $O(|w|)$ a Simon's Forest Factorisation of $w$ of $h$-rank at most $3 \cdot |S|$.*

The algorithm is then a classical bottom-up dynamic programming algorithm that computes the $\mathrm{FO}^q$-theory by following the Forest Factorisation given by Theorem 14. Recall that the height of the tree is upper-bounded by $3 \cdot 2^{2^{O(p)}}$ and $p \leq 2^k + 1$. The $\mathrm{FO}^q$-theory is computed as follows:

1. If the $h_p$-rank of $\sigma$ is 0, then $\sigma$ is an atomic $p$-derivation. So, we compute all the non-isomorphic atomic $p$-derivations that are substructures of $\sigma$.
2. Assume now that the $h_p$-rank of a subword $\sigma'$ of $\sigma$ is $\ell$. We compute the $\mathrm{FO}^q$-theory of $\sigma'$ as follows.
   a. If $\sigma'$ admits a binary factorisation into $\sigma_1$ and $\sigma_2$, then we have already computed the $\mathrm{FO}^q$-theories of $\sigma_1$ and $\sigma_2$. Therefore, the $\mathrm{FO}^q$-theory of $\sigma'$ can be computed by taking a subset $\mathcal{T}_1$ of the $\mathrm{FO}^q$-theory of $\sigma_1$, a subset $\mathcal{T}_2$ of the $\mathrm{FO}^q$-theory of $\sigma_2$ and a boolean function on $\mathcal{T}_1 \cup \mathcal{T}_2$. Whenever the boolean function is satisfied, we add it to the $\mathrm{FO}^q$-theory of $\sigma'$ (by keeping only non-equivalent ones). Since the $\mathrm{FO}^q$-theories of $\sigma_1$ and of $\sigma_2$ are upper-bounded by $f_{\ell-1}(q)$, the computation can be done in time $2^{O(f_{\ell-1}(q) \cdot f_{\ell-1}(q))}$.
   b. If $\sigma'$ admits an unranked factorisation into $\sigma_1\sigma_2\cdots\sigma_n$, then we compute the $\mathrm{FO}^q$ from left to right as follows. Let $t$ be the maximum such that we have computed the $\mathrm{FO}^q$-query of $\sigma_1 \otimes \cdots \otimes \sigma_t$ (recall that $t \geq 1$ as the $\mathrm{FO}^q$-theory of each $\sigma_s$, for $s \in [n]$, is already computed). We can now use the same procedure as in the binary case to compute the $\mathrm{FO}^q$-query of $\sigma_1 \otimes \cdots \otimes \sigma_{t+1}$. Since we keep only the non-equivalent ones, by Proposition 10, the time complexity is upper-bounded by a polynomial on $2^{2^{2\cdot(f_{\ell-1}(q)+1)^{2q} \cdot 2^{2^{(2^{f_{\ell-1}(q)})^{2q}}}}}$.

## References


1. Mikołaj Bojańczyk, Martin Grohe, and MichałPilipczuk. Definable decompositions for graphs of bounded linear cliquewidth. *Log. Methods Comput. Sci.*, 17(1):Paper No. 5, 40, 2021.
2. Thomas Colcombet. A combinatorial theorem for trees. In Lars Arge, Christian Cachin, Tomasz Jurdzinski, and Andrzej Tarlecki, editors, *Automata, Languages and Programming, 34th International Colloquium, ICALP 2007, Wroclaw, Poland, July 9-13, 2007, Proceedings*, volume 4596 of *Lecture Notes in Computer Science*, pages 901–912. Springer, 2007. doi:10.1007/978-3-540-73420-8_77.
3. Bruno Courcelle. The monadic second-order logic of graphs. I. Recognizable sets of finite graphs. *Inform. and Comput.*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.
4. Bruno Courcelle and Joost Engelfriet. *Graph structure and monadic second-order logic*, volume 138 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, 2012. A language-theoretic approach, With a foreword by Maurice Nivat. doi:10.1017/CBO9780511977619.
5. Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory Comput. Syst.*, 33(2):125–150, 2000. doi:10.1007/s002249910009.
6. Reinhard Diestel. *Graph theory*, volume 173 of *Graduate Texts in Mathematics*. Springer-Verlag, Berlin, third edition, 2005.
7. S. Feferman and R. L. Vaught. The first order properties of products of algebraic systems. *Fundam. Informaticae*, 47(1-3):57–103, 1959. URL: http://eudml.org/doc/213526.
8. Jisu Jeong, Eun Jung Kim, and Sang-il Oum. The "art of trellis decoding" is fixed-parameter tractable. *IEEE Trans. Inform. Theory*, 63(11):7178–7205, 2017. doi:10.1109/TIT.2017.2740283.
9. Manfred Kufleitner. The height of factorization forests. In *Mathematical foundations of computer science 2008*, volume 5162 of *Lecture Notes in Comput. Sci.*, pages 443–454. Springer, Berlin, 2008. doi:10.1007/978-3-540-85238-4_36.
10. Michael Lampis. First order logic on pathwidth revisited again. In Kousha Etessami, Uriel Feige, and Gabriele Puppis, editors, *50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10-14, 2023, Paderborn, Germany*, volume 261 of *LIPIcs*, pages 132:1–132:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023. doi:10.4230/LIPIcs.ICALP.2023.132.
11. Johann A. Makowsky. Algorithmic uses of the feferman-vaught theorem. *Ann. Pure Appl. Log.*, 126(1-3):159–213, 2004. doi:10.1016/j.apal.2003.11.002.
12. Michal Pilipczuk. Problems parameterized by treewidth tractable in single exponential time: A logical approach. In Filip Murlak and Piotr Sankowski, editors, *Mathematical Foundations of Computer Science 2011 - 36th International Symposium, MFCS 2011, Warsaw, Poland, August 22-26, 2011. Proceedings*, volume 6907 of *Lecture Notes in Computer Science*, pages 520–531. Springer, 2011. doi:10.1007/978-3-642-22993-0_47.
13. Detlef Seese. Linear time computable problems and logical descriptions. In Andrea Corradini and Ugo Montanari, editors, *Joint COMPUGRAPH/SEMAGRAPH Workshop on Graph Rewriting and Computation, SEGRAGRA 1995, Volterra, Italy, August 28 - September 1, 1995*, volume 2 of *Electronic Notes in Theoretical Computer Science*, pages 246–259. Elsevier, 1995. doi:10.1016/S1571-0661(05)80203-8.
14. Imre Simon. Factorization forests of finite height. *Theoret. Comput. Sci.*, 72(1):65–94, 1990. doi:10.1016/0304-3975(90)90047-L.