Order-Invariant First-Order Logic over Hollow Trees

Julien Grange
ENS Paris & PSL & INRIA & CNRS, France
jgrange@di.ens.fr

Luc Segoufin
INRIA & ENS Paris & PSL, France
luc.segoufin@inria.fr

Abstract

We show that the expressive power of order-invariant first-order logic collapses to first-order logic over hollow trees. A hollow tree is an unranked ordered tree where every non leaf node has at most four adjacent nodes: two siblings (left and right) and its first and last children. In particular there is no predicate for the linear order among siblings nor for the descendant relation. Moreover only the first and last nodes of a siblinghood are linked to their parent node, and the parent-child relation cannot be completely reconstructed in first-order.

2012 ACM Subject Classification Theory of computation → Finite Model Theory

Keywords and phrases order-invariance, first-order logic

Digital Object Identifier 10.4230/LIPIcs.CSL.2020.23

Related Version `https://hal.inria.fr/hal-02310749/document`

Funding ANR QUID

1 Introduction

First-order logic (FO) is a classical formalism for expressing properties over finite structures. It is the building block of many other formalisms that are highly expressive such as MSO or logics using fixpoints such as LFP. An important and desirable feature of FO, and of all its extensions mentioned above, is that it expresses only intrinsic properties of the structure, i.e. properties invariant under isomorphisms. A limitation of FO is that it cannot express some simple properties. In particular, as it cannot distinguish between nodes that are related via some automorphism, it cannot always go through all the nodes of a structure in order to perform simple tasks such as counting them.

In many scenarios, in particular in computer science, the structures under investigation are stored on a disk: this yields an implicit order among the elements of the structure. It is then reasonable to use this order within the logical formalism. In the case of FO this means adding a new binary predicate that is interpreted as a linear order. However, we want to do this in such a way that closure under isomorphisms is retained: the expressible properties should only depend on the structure and not on the way it is stored on the disk, the latter being arbitrary and subject to change. When this property is verified we say that the formula is **order-invariant** and we denote by $<$-inv FO the set of first-order formulas that are order-invariant. We stress that being order-invariant is not a decidable property [4] hence $<$-inv FO is not a recursive set of formulas.

Obtaining a "real" logic (in the sense of Gurevich, in particular with a recursive syntax) that has exactly the same expressive power as $<$-inv FO is a challenging question. Solving the same question for $<$-inv LFP would solve the longstanding quest of finding a logic for PTime as it follows from Immermann-Vardi Theorem that $<$-inv LFP captures PTime.

In order to find a logic for $<$-inv FO, it is useful to understand a bit better its expressive power; such is the goal of this paper.

28th EACSL Annual Conference on Computer Science Logic (CSL 2020).
Editors: Maribel Fernández and Anca Muscholl; Article No. 23; pp. 23:1–23:16
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

An example, attributed to Gurevich, shows that $<$-inv FO is in general strictly more expressive than FO [1]. Another key result shows that $<$-inv FO retains the local property of FO [7]. It seems that it requires dense structures for $<$-inv FO to express strictly more than FO. For instance when the structures are trees it has been shown that $<$-inv FO has exactly the same expressive power than FO [4]. In [4] a "tree" is either a binary tree, where every node has at most three neighbors: its parent, its left child and its right child or, an unranked unordered tree where every node is related to its parent and all of its children, but no order is assumed among siblings.

The question of whether $<$-inv FO $=$ FO over any class of structures of bounded treewidth was left open in [4], where it is only shown that, over structures of bounded treewidth, $<$-inv FO can only express properties definable in MSO.

In order to show that $<$-inv FO collapses to FO over a class of structures of bounded treewidth, it is tempting to reduce the case of bounded treewidth to the case of trees, using tree decompositions. When trying this strategy one immediately faces two difficulties. The first one is, given two FO similar structures (in this introduction we informally say that two structures are "FO similar" if they satisfy the same FO sentences of quantifier rank k for some k sufficiently large and depending on the context), to exhibit a tree decomposition for each of them such that the resulting tree decompositions are FO similar. Once this is done, we can apply the known result over trees showing that the tree decompositions actually agree on all order-invariant properties of a given quantifier rank: they are $<$-inv FO similar. The second difficulty is then to lift the order-invariance similarity from the tree decompositions to the original structures.

The second difficulty could be solved easily if we could interpret the original structure within its tree decomposition. Unfortunately this cannot be done in first-order (this requires reachability as an element of the structure could appear in bags arbitrarily far away within the tree decomposition). This problem can be eliminated by assuming "domino treewidth", i.e. that an element appears in a bounded number of bags, which is equivalent to assuming bounded degree of the structure on top of bounded treewidth [5].

Even when assuming bounded degree, the first difficulty remains and we still do not know the precise expressive power of $<$-inv FO over structures of bounded degree and pathwidth 2! This paper is an attempt toward solving the pathwidth 2 case.

We show that $<$-inv FO collapses to FO over the class of hollow trees. Hollow trees are first-order structures with two binary relations that are interpreted so that the resulting structure is a tree with the following features: each node has at most four neighbors: its first child, its last child and possibly a left and a right sibling. One of the binary relation denotes the sibling relation while the other one denotes the partial parent-child relation. This model strictly extends the case of binary trees as a node may have arbitrarily many children. However it is less powerful than the unranked ordered model as a node is not directly related to its parent, unless it is the first or last of its children. Note that because of its locality, FO cannot reconstruct the complete parent-child relation of every node within a hollow tree (this can be done in MSO or using the transitive closure of the sibling relation).

It is not immediate to see how hollow trees are related to structures of pathwidth 2 and of bounded degree. It turns out that if in the model of hollow trees we only had one binary relation and could not distinguishing between the (partial) parent-child relation and the sibling one, then we would have a model that is FO equivalent to structures of bounded degree and pathwidth 2 in the sense that there exist FO-interpretations from one to the other (as depicted in the conclusion). In particular the collapse of $<$-inv FO to FO in one of them would imply the collapse in the other as we explain in Section 2.4. We leave the extension of our result to this class of structures as an open problem.

Our proof follows a strategy similar to the case of binary trees: we first exhibit a set of operations over hollow trees (actually over structures FO similar to hollow trees) that preserve order-invariance similarity. We then show that if two hollow trees are FO similar then one of them can be transformed using our set of operations into the other, lifting FO similarity to $<$-inv FO similarity. The first part is standard and makes use of the locality of $<$-inv FO [7]. The second part is more combinatorial and forms the main technical contribution of this paper.

Related work. Besides the papers already mentioned above, there exist several other publications related to our work. We will make use in our proof of the fact that $<$-inv FO $\subseteq$ MSO over classes of graphs of bounded treewidth, which has been initially claimed in [4]. Another proof of this result, extended to a broader class called "decomposable structures", can be found in [6].

If testing order invariance is undecidable for FO it is decidable for its two variable fragment [13].

Several authors considered order-invariance for more expressive logics (first-order with modulo predicates [11], MSO [6]) or with more expressive numerical predicates [9, 8, 2, 12]. Our proof technique follows lines similar to [4, 11] but is mildly related to the others.

Due to space limitations many of the proofs are omitted or just sketched in this long abstract. They can be found at `https://hal.inria.fr/hal-02310749/document`

2 Preliminaries

2.1 General notations

We consider relational structures and use classical terminology for them. We use Σ to denote a relational schema and Σ-structure to denote a structure over Σ. Our structures are always finite and are denoted through calligraphic upper-case letters and their domain through the corresponding standard upper-case letter. For instance, A would denote the domain of the structure $\mathcal{A}$. For a relation symbol $R \in \Sigma$ and a Σ-structure $\mathcal{A}$, we denote by $R^{\mathcal{A}}$ the interpretation of R in $\mathcal{A}$.

Given a relational signature Σ, first-order logic, FO(Σ), and monadic second-order logic, MSO(Σ), are defined in the standard way (see, e.g., [10]). The main formalism of interest here is order-invariant first-order logic, denoted $<$-inv FO(Σ). A sentence φ in FO($\Sigma \cup \{<\}$) belongs to $<$-inv FO(Σ) if for every Σ-structure $\mathcal{A}$, whether $(\mathcal{A}, <^{\mathcal{A}}) \models \varphi$ is independent of the choice of the linear order $<^{\mathcal{A}}$ on A. In that case, we write $\mathcal{A} \models \varphi$. For any $\mathcal{L} \in \{\text{FO}(\Sigma), \text{MSO}(\Sigma), <\text{-inv FO}(\Sigma)\}$ and two Σ-structures $\mathcal{A}$ and $\mathcal{B}$, we write $\mathcal{A} \equiv_k^{\mathcal{L}} \mathcal{B}$ to mean that $\mathcal{A}$ and $\mathcal{B}$ satisfy the same sentences of $\mathcal{L}$ of quantifier rank at most k. As usual we omit Σ when it is clear from the context.

We use the standard notion of FO-interpretations in order to define a new structure from an existing one. Given a FO-interpretation $\mathcal{I}$, we call **arity** of $\mathcal{I}$ the number of free variables in the formula of $\mathcal{I}$ which defines the domain of the new structure, and **depth** of $\mathcal{I}$ the maximum among the quantifier ranks of the formulas defining the domain and the new relations. It is a well known result that for every $\mathcal{A}, \mathcal{B}$, and $\mathcal{I}$ of arity a and depth d, and for every $k \in \mathbb{N}$, if $\mathcal{A} \equiv_{ak+d}^{\mathcal{L}} \mathcal{B}$ then $\mathcal{I}(\mathcal{A}) \equiv_k^{\mathcal{L}} \mathcal{I}(\mathcal{B})$.

Let $\mathcal{A}$ be a structure over a vocabulary containing the binary relation symbol R. We say that $U \subseteq A$ is R**-stable** if $\forall x \in U, \forall y \in A, (R(x,y) \vee R(y,x)) \rightarrow y \in U$.

For a set σ of symbols, we define the vocabulary $P_\sigma := \{P_s : s \in \sigma\}$, where every P_s is a unary relation symbol.

As usual the Gaifman graph of a relational structure $\mathcal{A}$ is the (unoriented) graph whose vertices are the elements of the domain of the structure and the edges relate two vertices that appear in the same tuple of a relation of $\mathcal{A}$. We denote by $\text{dist}_{\mathcal{A}}(x, y)$ the distance between x and y in the Gaifman graph of $\mathcal{A}$. Given two sets S and T of elements of A and $m \in \mathbb{N}$, we say that S and T are **m-distant** in $\mathcal{A}$, if $\text{dist}_{\mathcal{A}}(x, y) \geq m$ for all $x \in S$ and all $y \in T$. The **k-neighborhood** $\mathcal{N}^k_{\mathcal{A}}(x)$ of some $x \in A$ is the substructure of $\mathcal{A}$ induced by $\{y \in A : \text{dist}_{\mathcal{A}}(x, y) \leq k\}$ together with an additional constant interpreted as x. The **k-type** $\text{tp}^k_{\mathcal{A}}(x)$ of x in $\mathcal{A}$ is the isomorphism class of its k-neighborhood. We extend those definitions to tuples of elements in the usual way, fixing the tuples pointwise.

For $k \in \mathbb{N}$, we define the **k-enrichment** $\mathcal{E}_k(\mathcal{A})$ of a Σ-structure $\mathcal{A}$ as $\mathcal{A}$ itself where each element has been recolored with its k-type. $\mathcal{E}_k(\mathcal{A})$ is a structure over the vocabulary Σ augmented with a unary predicate for every k-type over Σ: there are a finite number of them as long as we consider classes of structures of bounded degree.

2.2 Hollow trees

An unranked ordered tree is a tree with a successor relation among the children of any node. We see unranked ordered trees as structures over the signature composed of two binary relation symbols S and S', where S is interpreted as the parent-child relation, and S' as the horizontal successor. A set of nodes that share the same parent is called a siblinghood.

We define a mapping H from the set of unranked ordered trees to structures over two binary predicates S and E. Given an unranked ordered tree $\mathcal{T}$, $H(\mathcal{T})$ is defined as follows:

- its domain is T
- $H(\mathcal{T}) \models S(x, y)$ iff $\mathcal{T} \models S(x, y)$ and y is either the first or the last of its siblings
- E is interpreted as the symmetrical closure of S'

The image of H is the set of **hollow trees**, denoted $\mathbb{H}$. If $\mathcal{P} = H(\mathcal{T})$ then $\mathcal{T}$ is the underlying tree structure of $\mathcal{P}$.

In other words, within a hollow tree, only the two children at the endpoints of a siblinghood know their parent. Notice that we do not distinguish between the first and last child, nor do we between the left and right sibling. This makes the model more general, as explained in Section 2.4. An example of hollow tree is given in the left part of Figure 1.

Figure 1 An example of hollow tree (left) and of hollow quasitree (right). The dotted arrows represent S and the plain (symmetrical) lines represent E.

Given a finite alphabet σ, we define $\mathbb{H}_\sigma$, the set of **hollow trees over σ**, as the set of colored extensions of hollow trees using the vocabulary P_σ, where the interpretations of the predicates of P_σ partition the domain.

2.3 Main result

If $\mathcal{C}$ is a class of structures, we say that $<$-inv FO $=$ FO over $\mathcal{C}$ if for each property definable in $<$-inv FO, there exists a first-order formula expressing this property over all structures of $\mathcal{C}$. Notice that for every σ, $\mathbb{H}_\sigma$ is a class of structures of treewidth 2. Therefore $<$-inv FO $\subseteq$ MSO over $\mathbb{H}_\sigma$ [4]. The main result we prove in this paper is:

▶ **Theorem 1.** *For all* σ*,* $<$-inv FO $=$ FO *over* $\mathbb{H}_\sigma$

We outline the proof here, and give more details in the rest of this paper.

Proof sketch. Our goal is to find some function f such that, $\forall\alpha \in \mathbb{N}, \forall\mathcal{P}, \mathcal{Q} \in \mathbb{H}_\sigma$, if $\mathcal{P} \equiv^{\text{FO}}_{f(\alpha)} \mathcal{Q}$ then $\mathcal{P} \equiv^{<\text{-inv FO}}_{\alpha} \mathcal{Q}$. This means that the equivalence relation $\equiv^{\text{FO}}_{f(\alpha)}$ refines $\equiv^{<\text{-inv FO}}_{\alpha}$. Both equivalence relations being of finite index and the former being definable in FO for every fixed α, the result follows.

To show this we fix some $\alpha \in \mathbb{N}$ and consider two hollow trees $\mathcal{P}$ and $\mathcal{Q}$, such that $\mathcal{P} \equiv^{\text{FO}}_{f(\alpha)} \mathcal{Q}$ for a large enough $f(\alpha)$. The general idea is to modify $\mathcal{Q}$ through some operations that are invisible to all formulas of $<$-inv FO of quantifier rank less than α, until we reach $\mathcal{P}$. This will ensure that $\mathcal{P} \equiv^{<\text{-inv FO}}_{\alpha} \mathcal{Q}$.

We will use two kinds of operations as described in Section 3: "swap operations", which preserve $<$-inv FO, and one which preserves MSO (and *a fortiori* $<$-inv FO as $<$-inv FO $\subseteq$ MSO over $\mathbb{H}_\sigma$ by [4]).

The MSO-preserving operation will be used in Section 3.3, in order to pump $\mathcal{Q}$ to make sure that every neighborhood type is present at least as many times in $\mathcal{Q}$ as in $\mathcal{P}$.

Once this is done, we explain in Section 4 how to transform $\mathcal{Q}$ with swap operations in order to include $\mathcal{P}$ into it. Since $\mathcal{Q}$ may be larger than $\mathcal{P}$, there could be some extra material in $\mathcal{Q}$ that we call "loops". The last step is to remove those loops and this is the goal of Section 6.

When performing the swap operations, there will be a constant need for reorganizing the S-edges (in particular to make sure that the loops are S-stable). Section 5 and Section 6.3 compile the results that allow us to do so. ◀

2.4 Bi-FO-interpretations and corollaries

Before we give more details about the proof of our main result, we recall in this section a classical tool for reducing the collapse of $<$-inv FO to FO from one class of structures to another. We then state a few corollaries of Theorem 1.

Let $\mathcal{C}_1, \mathcal{C}_2$ be two classes of structures over the respective vocabularies τ_1 and τ_2.

We say that $\mathcal{C}_1$ is **bi-FO-interpretable** through $\mathcal{C}_2$ if there exist two FO-interpretations $\mathcal{I}_{12}$ and $\mathcal{I}_{21}$, respectively from τ_1 to τ_2, and from τ_2 to τ_1, such that for every $\mathcal{A} \in \mathcal{C}_1$, $\mathcal{I}_{12}(\mathcal{A}) \in \mathcal{C}_2$ and $\mathcal{I}_{21}(\mathcal{I}_{12}(\mathcal{A})) \simeq \mathcal{A}$, where $\simeq$ denotes the existence of an isomorphism between two structures. The following result is rather straightforward:

▶ **Lemma 2.** *If* $\mathcal{C}_1$ *is bi-*FO*-interpretable through* $\mathcal{C}_2$ *and* $<$-inv FO $=$ FO *over* $\mathcal{C}_2$*, then* $<$-inv FO $=$ FO *over* $\mathcal{C}_1$

Recall that in the definition of hollow trees the relation E is symmetric. This turns out to be more general than choosing E as an arbitrary directed binary relation as shown in the following result where a **directed hollow tree** is defined as for hollow trees but with a directed binary relation E. Note that we do not assume that E is a successor relation among siblings, the direction of E could be arbitrary, but the result below works in particular when E is a successor relation. Via a simple bi-FO-interpretation which uses extra colors to encode the direction of the edges, we get the following result:

▶ **Corollary 3.** *For every σ, $<$-inv FO $=$ FO on the class of σ directed hollow trees*

Define a path over σ as a word over the alphabet σ, where the successor edges are symmetrical (the argument used in the proof of Corollary 3 guarantees that paths are a more general model than words). The class of paths over σ is obviously bi-FO-interpretable through $\mathbb{H}_\sigma$: just add a S-parent to the endpoints of the path, and then forget about it. Thus we get:

▶ **Corollary 4.** *For every alphabet σ, $<$-inv FO $=$ FO on the class of paths over σ.*

Similarly, a straightforward bi-FO-interpretation together with Theorem 1 give us back the result from [4] that $<$-inv FO $=$ FO on ranked trees.

3 Swaps and pumping

In this section we provide a few operations, denoted swaps, that preserve $\equiv_k^{<\text{-inv FO}}$. Although the k-type of every element will be left unchanged, applying these operations may break the somewhat rigid structure of hollow trees. In order to work with the intermediate structures, we loosen the definition of hollow trees and define hollow quasitrees as follows:

▶ **Definition 5.** *For $k > 0$ and σ a set of colors, we define the set of **hollow k-quasitrees on** σ, quasi-$\mathbb{H}_\sigma^k$, as the set of all finite structures over $\{E, S\} \cup P_\sigma$ such that the k-type of any of their elements is the k-type of some element in some hollow tree in $\mathbb{H}_\sigma$, and which are such that their relation E is acyclic.*

In other words a hollow quasitree differs from a hollow tree by its relation S which may not induce a tree structure: a node may have its S-children in two distinct siblinghoods and a hollow quasitree may have cycles using the relation S (but not using only the relation E). Note that by definition $\mathbb{H}_\sigma \subseteq$ quasi-$\mathbb{H}_\sigma^k$ for every k. An example of what a hollow quasitree could look like is given in the right part of Figure 1. Note that locally, it looks like a hollow tree.

Let $\mathcal{T} \in$ quasi-$\mathbb{H}_\sigma^k$. We define the **support** of $\mathcal{T}$ as its restriction to the vocabulary $P_\sigma \cup \{E\}$. The **n-enriched support** of $\mathcal{T}$, denoted $\mathcal{S}\text{upp}_n(\mathcal{T})$, is the support of its n-enrichment (and not the other way around). Hence, it keeps in memory the local behavior within $\mathcal{T}$. The set $\text{End}(\mathcal{T})$ of **endpoints** of $\mathcal{T}$ is the set of elements of the support having degree one. A connected component of the support of $\mathcal{T}$ is called a **thread**[1]. Note that by E-acyclicity of $\mathcal{T}$, each of its threads is a path, hence contains exactly two endpoints. We say that a hollow k-quasitree has the **matching endpoints property** if the two endpoints of each thread have the same S-parent. Note that a hollow tree has the matching endpoints property. Notice also that in a hollow k-quasitree, any thread of length less than $2k+1$ has matching endpoints. For $x, y \in T$ belonging to the same thread, $[x, y]$ denotes the set of elements that lie between them (formally, those who disconnect x from y in $\mathcal{S}\text{upp}_0(\mathcal{T})$), including x and y. We naturally define $[x, y[$ as $[x, y] \setminus \{y\}$.

The following lemma, implicit in the proof of locality of $<$-inv FO by Grohe and Schwentick [7], will allow us to prove that our operations preserve order-invariance equivalence:

▶ **Lemma 6.** *Let Σ be a relational vocabulary and let $p, \alpha \in \mathbb{N}$. There exists $o_p^\Sigma(\alpha) \in \mathbb{N}$ such that for every structure $\mathcal{A}$ over Σ, and for every p-tuples of elements $\bar{a}, \bar{b} \in A^p$ that have the same $o_p^\Sigma(\alpha)$-type in $\mathcal{A}$, there are two orders $<_{\bar{a}\bar{b}}$ and $<_{\bar{b}\bar{a}}$ on A such that*

[1] A thread is nothing other than a siblinghood when the quasitree is a tree.

- $(\mathcal{A}, <_{\bar{a}\bar{b}}) \equiv_\alpha^{\text{FO}} (\mathcal{A}, <_{\bar{b}\bar{a}})$
- $\bar{a}\bar{b}$ *is an initial segment of* $<_{\bar{a}\bar{b}}$
- $\bar{b}\bar{a}$ *is an initial segment of* $<_{\bar{b}\bar{a}}$

Our operations are divided into three families depending on whether we modify the relation S, the relation E, or whether we do a global pumping,

In the following, $\mathcal{R}$ is a hollow $(m+1)$-quasitree on σ.

3.1 crossing-S-swaps

Let $a, a', a'', b, b', b'' \in R$ be such that $S(a, a'), S(a, a''), S(b, b'), S(b, b'')$ and such that $\text{tp}^m_{\mathcal{R}}(a, a', a'') = \text{tp}^m_{\mathcal{R}}(b, b', b'')$. Let $\mathcal{R}^- := \mathcal{R} \setminus \{S(a, a'), S(a, a''), S(b, b'), S(b, b'')\}$ and assume that the sets $\{a', a''\}, \{b', b''\}$ and $\{a, b\}$ are pairwise $(2m+3)$-distant in $\mathcal{R}^-$.

Then $\mathcal{R}' := \mathcal{R}^- \cup \{S(a, b'), S(a, b''), S(b, a'), S(b, a'')\}$ is called the **m-guarded crossing-S-swap between a and b in $\mathcal{R}$** (see Figure 2).

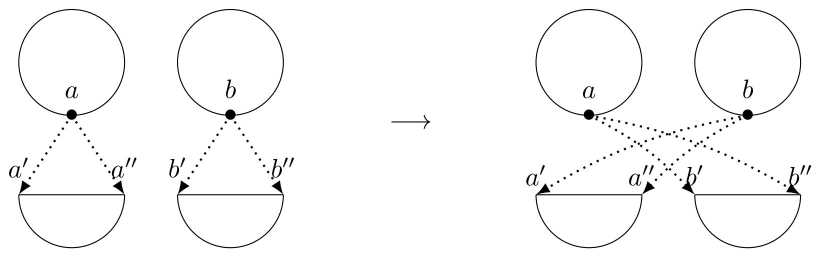

Figure 2 The crossing-S-swap between a and b.

▶ **Note 7.** *A particular case where the distance condition is met is when* $dist_{\mathcal{R}}(a, b) \geq 2m+5$.

▶ **Lemma 8.** *For all $\alpha \in \mathbb{N}$ there exists $s(\alpha) \in \mathbb{N}$ such that for all $m \geq s(\alpha)$, and every hollow $(m+1)$-quasitree $\mathcal{R}$,*

if $\mathcal{R}'$ is the m-guarded crossing-S-swap between a and b in $\mathcal{R}$,

then $\mathcal{R}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{R}$, and $\forall x \in R$, $tp^{m+1}_{\mathcal{R}'}(x) = tp^{m+1}_{\mathcal{R}}(x)$. Moreover $\mathcal{R}' \in$ quasi-$\mathbb{H}^{m+1}_\sigma$ and $\mathcal{S}upp_{m+1}(\mathcal{R}') = \mathcal{S}upp_{m+1}(\mathcal{R})$.

Proof sketch. In order to prove that $\mathcal{R}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{R}$ we need to exhibit a linear order over $\mathcal{R}$ and one over $\mathcal{R}'$ such that we can play an α-round Ehrenfeucht-Fraïssé game between the resulting ordered structures. The linear orders are constructed using Lemma 6 applied to (a', a'') and (b', b'') and the structure $\mathcal{R}^-$. A simple FO-interpretation is then used to transfer the corresponding orders onto $\mathcal{R}$ and $\mathcal{R}'$. Proving that the type of an element is unchanged is straightforward. ◀

3.2 E-swaps

We define four different kinds of E-swaps.

Let $a, b, a', b' \in R$ be such that $E(a, b), E(a', b')$, a, b and a', b' appear in two different threads of $\mathcal{R}$ and such that $\{a, b, a', b'\}$ and $\text{End}(\mathcal{R})$ are $(2m+3)$-distant in $\mathcal{S}\text{upp}_0(\mathcal{R})$. Furthermore, assume that $\text{tp}^m_{\mathcal{R}}(a, b) = \text{tp}^m_{\mathcal{R}}(a', b')$. Let $\mathcal{R}' := \mathcal{R} \setminus \{E(a, b), E(a', b')\} \cup \{E(a, b'), E(a', b)\}$.

Then $\mathcal{R}'$ is called the **m-guarded crossing-E-swap between ab and $a'b'$ in $\mathcal{R}$** (c.f. Figure 3).

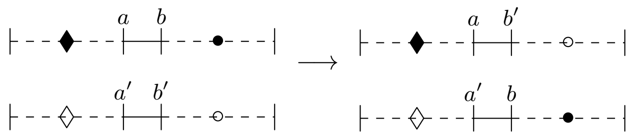

Figure 3 Illustration of the m-guarded crossing-E-swap between ab and $a'b'$ in $\mathcal{R}$.

Let $a, b, b', a' \in R$ appear in that order in a single thread of $\mathcal{R}$, such that $E(a, b), E(a', b')$, and such that $\{a, b, a', b'\}$ and $\mathrm{End}(\mathcal{R})$ are $(2m + 3)$-distant in $\mathcal{S}\mathrm{upp}_0(\mathcal{R})$. Furthermore, assume that $\mathrm{tp}^m_{\mathcal{R}}(a, b) = \mathrm{tp}^m_{\mathcal{R}}(a', b')$. Let $\mathcal{R}' := \mathcal{R} \setminus \{E(a, b), E(a', b')\} \cup \{E(a, b'), E(a', b)\}$. Then $\mathcal{R}'$ is called the **m-guarded mirror-E-swap at $[b, b']$ in** $\mathcal{R}$ (c.f. Figure 4).

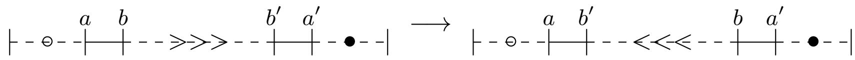

Figure 4 Illustration of the m-guarded mirror-E-swap at $[b, b']$ in $\mathcal{R}$.

Consider now $a, b, c, d, a', b', c', d' \in R$ appearing in that order in a single thread of $\mathcal{R}$ such that $E(a, b), E(c, d), E(a', b'), E(c', d')$ and such that $\{a, b, c, d, a', b', c', d'\}$ and $\mathrm{End}(\mathcal{R})$ are $(2m + 3)$-distant in $\mathcal{S}\mathrm{upp}_0(\mathcal{R})$. Furthermore, assume that $\mathrm{tp}^m_{\mathcal{R}}(a, b) = \mathrm{tp}^m_{\mathcal{R}}(a', b')$ and $\mathrm{tp}^m_{\mathcal{R}}(c, d) = \mathrm{tp}^m_{\mathcal{R}}(c', d')$.
Let $\mathcal{R}' := \mathcal{R} \setminus \{E(a, b), E(a', b'), E(c, d), E(c', d')\} \cup \{E(a, b'), E(a', b), E(c, d'), E(c', d)\}$. $\mathcal{R}'$ is called the **m-guarded segment-E-swap between** $[b, c]$ **and** $[b', c']$ **in** $\mathcal{R}$ (c.f. Figure 5).

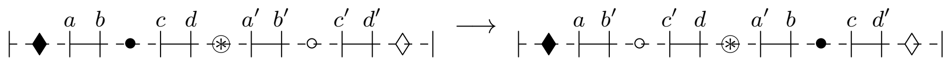

Figure 5 Illustration of the m-guarded segment-E-swap between $[b, c]$ and $[b', c']$ in $\mathcal{R}$.

Finally, let a, b, a', b', a'', b'' be elements of R appearing in that order in a single thread of $\mathcal{R}$, such that $E(a, b), E(a', b')$ and $E(a'', b'')$ and $\{a, b, a', b', a'', b''\}$ and $\mathrm{End}(\mathcal{R})$ are $(2m+3)$-distant in $\mathcal{S}\mathrm{upp}_0(\mathcal{R})$. Furthermore, suppose that $\mathrm{tp}^m_{\mathcal{R}}(a, b) = \mathrm{tp}^m_{\mathcal{R}}(a', b') = \mathrm{tp}^m_{\mathcal{R}}(a'', b'')$.

Let $\mathcal{R}' := \mathcal{R} \setminus \{E(a, b), E(a', b'), E(a'', b'')\} \cup \{E(a, b'), E(a', b''), E(a'', b)\}$. $\mathcal{R}'$ is called the **m-guarded contiguous-segment-E-swap between** $[b, a']$ **and** $[b', a'']$ **in** $\mathcal{R}$ (c.f. Figure 6).

As long as m is large enough, all the m-guarded E-swaps preserve $\equiv^{<\text{-inv FO}}_{\alpha}$ and the $(m + 1)$-type of every element:

▶ **Lemma 9.** *For all $\alpha \in \mathbb{N}$ there exists $s(\alpha) \in \mathbb{N}$ such that for every $m \geq s(\alpha)$ and every hollow $(m + 1)$-quasitree $\mathcal{R}$, if $\mathcal{R}'$ is either*

- *the m-guarded crossing-E-swap between ab and $a'b'$ in $\mathcal{R}$*
- *the m-guarded mirror-E-swap at $[b, b']$ in $\mathcal{R}$*
- *the m-guarded contiguous-segment-E-swap between $[b, a']$ and $[b', a'']$ in $\mathcal{R}$*
- *the m-guarded segment-E-swap between $[b, c]$ and $[b', c']$ in $\mathcal{R}$*

then $\mathcal{R}' \equiv^{<\text{-inv FO}}_{\alpha} \mathcal{R}$, $\forall x \in R$, $tp^{m+1}_{\mathcal{R}'}(x) = tp^{m+1}_{\mathcal{R}}(x)$ and $\mathcal{R}' \in$ quasi-$\mathbb{H}^{m+1}_{\sigma}$.

Proof sketch. The proof is a tedious case analysis. Basically it amounts to the following idea: if the elements involved in the swap are far away from each other then we can use Lemma 6 in the structure $\mathcal{R}$ minus the E-edges of interest, and get orders on $\mathcal{R}$ and $\mathcal{R}'$ which make these structures similar as in the proof of Lemma 8.

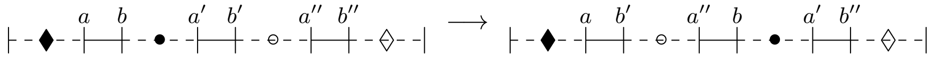

Figure 6 Illustration of the m-guarded contiguous-segment-E-swap between $[b, a']$ and $[b', a'']$ in $\mathcal{R}$.

On the other hand, if the elements are close to each other, then the fact that they share the same type induces some periodicity on their neighborhoods. These neighborhoods can therefore be decomposed into several consecutive similar pieces. We can then apply Lemma 6 to these smaller components to conclude. ◀

3.3 Pumping

The next operation makes use of the fact that $<$-inv FO $\subseteq$ MSO over hollow trees. Hence our hollow trees can be "pumped" in order to duplicate some of their parts.

Given a structure $\mathcal{A}$ and a k-type τ, we denote by $|\mathcal{A}|_\tau$ the number of elements of A whose k-type is τ. We will essentially use 0-types as our structures will be enriched by recoloring each element by its k-type. In view of this we denote by $[\![\mathcal{A}]\!]$ the function $\tau \mapsto |\mathcal{A}|_\tau$ whose domain is the set of 0-types over the considered vocabulary.

Let $d, D \in \mathbb{N}$, and f, g be functions from a same domain to $\mathbb{N}$. We say that $f \leq_d^D g$ if for every x in the domain:

- if $f(x) \leq d$, then $f(x) = g(x)$
- if $f(x) \neq g(x)$, then $g(x) \geq f(x) + D$

By $f < g$, we mean that $\forall x, f(x) < g(x)$ or $f(x) = g(x) = 0$.

In the following proposition $<$-inv FO can be replaced by MSO.

▶ **Proposition 10.** *$\forall \alpha, n, d \in \mathbb{N}, \exists M \in \mathbb{N}, \forall D \in \mathbb{N}, \forall \mathcal{P}, \mathcal{Q} \in \mathbb{H}_\sigma$, if $\mathcal{P} \equiv_M^{\text{FO}} \mathcal{Q}$, then there exists $\mathcal{Q}' \in \mathbb{H}_\sigma$ such that $\mathcal{Q}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{Q}$ and $[\![\mathcal{E}_{n+1}(\mathcal{P})]\!] \leq_d^D [\![\mathcal{E}_{n+1}(\mathcal{Q}')]\!]$.*

Proof sketch. This is a pumping argument: by setting M large enough, we make sure in FO that if a $(n+1)$-type has more occurrences in $\mathcal{P}$ than in $\mathcal{Q}$, then it has enough occurrences in $\mathcal{Q}$ so that we can find a context in $\mathcal{Q}$ containing at least one occurrence, and no occurrence of a rare type, such that we can duplicate this context inside $\mathcal{Q}$ without changing its MSO-type. ◀

4 Inclusion and pseudo-inclusion

Recall that our ultimate goal is to show that if two hollow trees agree on the same FO sentences of quantifier rank $f(\alpha)$ then they agree on all $<$-inv FO sentences of quantifier rank α. For this, we will show that if $\mathcal{P}$ and $\mathcal{Q}$ are hollow trees that agree on all FO sentences of quantifier rank $f(\alpha)$ then we can use operations such as the swap operations described in Section 3 to transform $\mathcal{Q}$ into $\mathcal{P}$. As these operations preserve $<$-inv FO we get the desired result.

In this section we perform the first step towards transforming $\mathcal{Q}$ into $\mathcal{P}$. We show that using the swap operations we can transform $\mathcal{Q}$ into $\mathcal{Q}'$ so that $\mathcal{Q}'$ "includes" $\mathcal{P}$. The resulting structure $\mathcal{Q}'$ will be a hollow quasitree. In the next sections we will continue the transformation and remove from $\mathcal{Q}'$ all the extra material it contains, deriving $\mathcal{P}$.

In order to define what we mean by "inclusion" we need the notion of a n-abstract context of a hollow quasitree. Intuitively this is a S-stable n-enriched substructure. More formally, given a hollow quasitree $\mathcal{T} \in$ quasi-$\mathbb{H}_\sigma^n$ and a set U of its domain that is S-stable, then $\mathcal{C} := \mathcal{T}_{|U}$, together with the function $\mathrm{tp}^n(.)$ that maps $x \in U$ to its n-type in $\mathcal{T}$, is called a **n-abstract context** denoted $\mathcal{C} = \mathrm{Ctxt}_n(\mathcal{T}|_U)$. The set of n-abstract contexts is denoted Ctxt_σ^n. Note that $\mathrm{tp}^n(x)$ denotes $\mathrm{tp}^n_{\mathcal{T}}(x)$ and not $\mathrm{tp}^n_{\mathcal{C}}(x)$. We need to remember, at least locally, how $\mathcal{C}$ was glued to the rest of $\mathcal{T}$ in order to preserve n-types when moving $\mathcal{C}$ to some other place.

We are now ready to define the notion of "inclusion". We actually define both "inclusions" and "pseudo-inclusions". We will need to pseudo-include a hollow quasitree into another (Proposition 12), and then to include an abstract context into a hollow quasitree (Proposition 13). Since a hollow k-quasitree $\mathcal{T} \in$ quasi-$\mathbb{H}_\sigma^k$ can be seen as a k-abstract context ($\mathcal{T} = \mathrm{Ctxt}_k(\mathcal{T}|_T)$), we only need to define (pseudo-)inclusions from an abstract context into a hollow quasitree.

▶ **Definition 11.** *Let $k \in \mathbb{N}$, $\mathcal{U} \in Ctxt_\sigma^k$ and $\mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^k$. We say that $h : U \to Q$ is a k-**pseudo-inclusion** if h is injective and for all $x, y, z \in U$ the following is verified:*

1. *$tp^k_{\mathcal{Q}}(h(x)) = tp^k(x)$,*
2. *if x and y are in the same thread of $\mathcal{U}$ then $h(x)$ and $h(y)$ are also on the same thread of $\mathcal{Q}$ and if moreover $z \in [x, y]$ then $h(z) \in [h(x), h(y)]$,*
3. *if $\mathcal{U} \models E(x, y)$ and t is the E-neighbor of $h(x)$ in $[h(x), h(y)]$ then t is the image of y by an isomorphism (induced by the fact that they share the same k-type) between the n-neighborhood of x and that of $h(x)$.*

*If $\mathcal{U} \models E(x, y)$ and $\mathcal{Q} \not\models E(h(x), h(y))$ then $\{x, y\}$ is said to be a **jumping pair** for h, and $tp^{k-1}_{Q}(h(x), t)$, where t is the E-neighbor of $h(x)$ in $[h(x), h(y)]$, is called its type.*[2]

*A k-pseudo-inclusion is said to be **reduced** if there is at most one jumping pair of a given type.*

*A k-pseudo-inclusion is called a k-**inclusion** if it has no jumping pairs, that is if it preserves E.*

The last condition of pseudo-inclusion is a complication induced by the fact that E is not oriented and that we thus cannot distinguish between the two siblings of a node. It ensures that h preserves the neighborhoods in the right order. We can now state the main result of this section. Note that the precondition that $\mathcal{Q}$ has more realizations for each type than $\mathcal{U}$ or $\mathcal{P}$ will not be a problem in view of Proposition 10. The second proposition is stronger than the first one as it derives inclusion instead of pseudo-inclusion, but it requires the stronger hypothesis that every occurring type has strictly more realizations in $\mathcal{Q}$ than in $\mathcal{U}$.

▶ **Proposition 12.** *For every $\alpha, m \in \mathbb{N}$, there exists $N \in \mathbb{N}$ such that $\forall \mathcal{P}, \mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^{N+1}$, if $[\![\mathcal{E}_{N+1}(\mathcal{P})]\!] \leq [\![\mathcal{E}_{N+1}(\mathcal{Q})]\!]$, then there exists $\mathcal{Q}' \in$ quasi-$\mathbb{H}_\sigma^{m+1}$ such that $\mathcal{Q}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{Q}$, $[\![\mathcal{E}_{m+1}(\mathcal{Q}')]\!] = [\![\mathcal{E}_{m+1}(\mathcal{Q})]\!]$ and h that is a $(m+1)$-pseudo-inclusion from $\mathcal{P}$ into $\mathcal{Q}'$.*

▶ **Proposition 13.** *For every $\alpha, m \in \mathbb{N}$, there exists $N \in \mathbb{N}$ such that $\forall \mathcal{U} \in Ctxt_\sigma^{N+1}$, $\forall \mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^{N+1}$, if $[\![\mathcal{E}_{N+1}(\mathcal{U})]\!] < [\![\mathcal{E}_{N+1}(\mathcal{Q})]\!]$, then there exists $\mathcal{Q}' \in$ quasi-$\mathbb{H}_\sigma^{m+1}$ such that $\mathcal{Q}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{Q}$, $[\![\mathcal{E}_{m+1}(\mathcal{Q}')]\!] = [\![\mathcal{E}_{m+1}(\mathcal{Q})]\!]$ and $\mathcal{U}$ is $(m+1)$-included in $\mathcal{Q}'$.*

Proof sketch. Both propositions have a similar proof: we first prove Proposition 12, and explain afterwards how to move from pseudo-inclusions to inclusions.

[2] This is an ease of notation; to be more precise, we should make the type of a jumping pair symmetrical.

We define the pseudo-inclusion h step by step, extending the domain of h thread by thread and, inside each thread, from one of its endpoint to the other. At each step we modify $\mathcal{Q}$ using E-swaps, if necessary.

We give a special treatment to short threads and portions of the long threads that are close to the endpoints: in that case, no modification of $\mathcal{Q}$ is required as the cardinality precondition ensures the presence of the necessary sequences within $\mathcal{Q}$. We then move to the parts of the long threads that are far from the endpoints, adding them one node at a time to the domain of the pseudo-inclusion. Note that as all the elements involved in the E-swaps to come are distant from the endpoints, the E-swaps involved are guarded.

Let x' be the last node of the current thread t that has been given an image by h, and let x be the next node to which we want to extend the domain of h. By hypothesis, we know that there exists a node $y \notin \mathrm{Im}(h)$ far from any endpoint, that has the same $(m+1)$-type as x. We denote by y' the neighbor of y that has the same m-type as x', and by $\hat{x}$ the neighbor of $h(x')$ having the same m-type as x.

We proceed to a case analysis depending on the relative position of $y, y', h(x')$ and $\hat{x}$. If y', y are on the same thread as $h(x'), \hat{x}$ and in the same direction (in particular when $y = \hat{x}$), we simply set $h(x)$ to y and we are done. If not, one of the E-swaps will place y to the desired position.

For instance, if y', y are on the same thread as $h(x'), \hat{x}$ but in the reverse direction (c.f. Figure 7, where the double line represents $\mathrm{Im}(h)$), then we consider the m-guarded mirror-E-swap at $[\hat{x}, y]$ in $\mathcal{Q}$ and extend h by setting $h(x)$ to y.

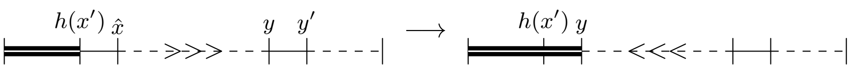

Figure 7 $h(x'), \hat{x}$ and y', y are in the same thread, but in reverse order: we use a mirror-E-swap.

Now, if y is on a thread that does not intersect $\mathrm{Im}(h)$ (c.f. Figure 8), we consider the m-guarded crossing-E-swap between $h(x')\hat{x}$ and $y'y$ in $\mathcal{Q}$, and extend h by setting $h(x)$ to y.

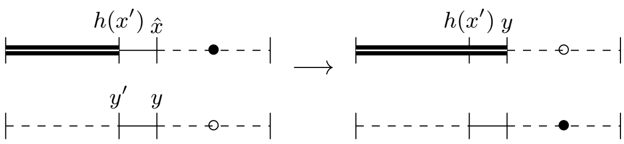

Figure 8 y is on a thread disjoint from $\mathrm{Im}(h)$: we use a crossing-E-swap.

If y', y are in the same direction as $h(x'), \hat{x}$, and are between $h(z)$ and $h(z')$ where z and z' are consecutive node of the current thread (c.f. Figure 9).

Then we consider the m-guarded segment-E-swap between $[u', y']$ and $[h(z'), h(x')]$ in $\mathcal{Q}$, and extend h by setting $h(x)$ to y.

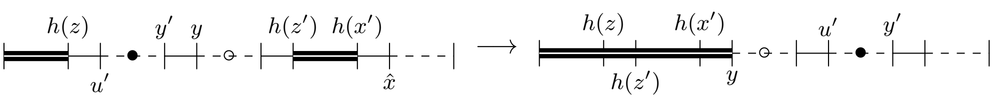

Figure 9 y', y are between the images of two already included neighbors: we use a segment-E-swap.

There are a few other cases that are treated similarly. This concludes the proof for pseudo-inclusion.

For Proposition 13, as we wish to construct an inclusion, we need to make sure that there is no "jump" in the mapping.

Note that among all the previously mentioned cases, only one didn't guarantee the absence of a jump, namely when y', y are on the same thread as $h(x'), \hat{x}$ and in the right direction, but when $y \neq \hat{x}$. We then use the stronger hypothesis on the number of types in $\mathcal{Q}$, which guarantees that there also exist z, z' verifying the same conditions as y, y' (cf. Figure 10). We consider the m-guarded contiguous-segment-E-swap between $[\hat{x}, y']$ and $[y, z']$ in $\mathcal{Q}$, and extend h by setting $h(x)$ to y. h is now an inclusion.

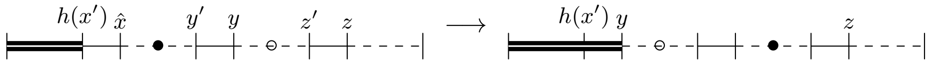

Figure 10 y', y, z', z and $h(x'), \hat{x}$ are on the same thread, in the same order: we use a contiguous-segment-E-swap to avoid a jump in the inclusion.

◀

5 Tools for reorganizing S-edges

In the previous section, we have seen how to "rewrite" $\mathcal{Q}$ using E-swap operations in order to pseudo-include $\mathcal{P}$ into the resulting quasitree. By definition, the pseudo-inclusion h of $\mathcal{P}$ into $\mathcal{Q}$ respects the enriched support but can be completely wild relatively to the S-edges. For instance, in $\mathcal{Q}$, the endpoints of a thread may not have the same S-parent. In this section we show how to use S-swaps in order to ensure that our pseudo-inclusion mapping takes into account (to various degrees) the S-edges. We say that two nodes of a quasitree are S-siblings if they share the same S-parent.

In Section 5.1, we show how to make sure that the pseudo-inclusion respects the S-siblings relation. In Section 5.2 we show how to ensure that the image of a pseudo-inclusion is S-stable. S-stability is required to define and operate on the loops, as will be established in Section 6.

5.1 S-siblings re-association

The following Lemma shows how to modify a pseudo-inclusion in order for it to preserve the S-siblings relation. Note that it doesn't necessarily mean that the image structure has the matching endpoint property because the initial structure itself may not have this property as it is derived from a quasitree.

▶ **Lemma 14.** *$\forall \alpha, m \in \mathbb{N}, \exists N \in \mathbb{N}, \forall \mathcal{W} \in Ctxt_\sigma^N, \forall \mathcal{Q} \in quasi\text{-}\mathbb{H}_\sigma^N$, if $h : W \to Q$ is a N-pseudo-inclusion, then there exists some $\mathcal{Q}' \in quasi\text{-}\mathbb{H}_\sigma^{m+1}$ and some $(m+1)$-pseudo-inclusion $h' : W \to Q'$ such that $\mathcal{Q}' \equiv_\alpha^{<\text{-inv FO}} \mathcal{Q}$, $\mathcal{S}upp_{m+1}(\mathcal{Q}') \simeq \mathcal{S}upp_{m+1}(\mathcal{Q})$ and, if x and y are S-siblings in $\mathcal{W}$, then so are $h'(x)$ and $h'(y)$ in $\mathcal{Q}'$.*

Proof sketch. We correct the S-edges two by two: let x, y be two S-siblings in $\mathcal{W}$ such that $h(x), h(y)$ are not S-siblings in $\mathcal{Q}$, and let $z \in Q$ be the S-sibling of $h(x)$.

z and $h(y)$ must have the same $(N-2)$-type: we can use a crossing-E-swap or a mirror-E-swap (depending on whether they are the endpoints of a same thread) to exchange their positions and make sure $h(x)$ and $h(y)$ are S-siblings.

However, for these swaps to be guarded, we must operate far enough from the endpoints. This can be done as long as we choose N large enough. ◀

A particular case of the previous lemma is when $\mathcal{W}$ is a hollow tree and h is surjective: then $\mathcal{Q}'$ has the matching endpoints property. This result will be useful in the proof of Proposition 18.

5.2 S-stabilization

The image of a pseudo-inclusion has no reason to be S-stable, thus neither has its complement. However, this is a crucial requirement to apply the results presented in the next section, Section 6, in order to remove the extra material not in the image of the pseudo-inclusion.

The next result provides a method to ensure that the image (and its complement) of a pseudo-inclusion is S-stable.

Recall that a pseudo-inclusion is said to be reduced if there is at most one jumping pair of a given type. At the end of this process, we get a reduced pseudo-inclusion, which will allow us to minimize the complement of its image in Section 6.1.

▶ **Proposition 15.** *For every $\alpha, m \in \mathbb{N}$, there exist $N, d, D \in \mathbb{N}$ such that, for every $\mathcal{P} \in \mathbb{H}_\sigma$, $\mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^{N+1}$ such that $[\![\mathcal{E}_{N+1}(\mathcal{P})]\!] \leq_d^D [\![\mathcal{E}_{N+1}(\mathcal{Q})]\!]$ and $\mathcal{P}$ is $(N+1)$-pseudo-included in $\mathcal{Q}$ through some h, there are some h' and $\mathcal{Q}' \in$ quasi-$\mathbb{H}_\sigma^{m+1}$ such that $\mathcal{Q}' \equiv_\alpha^{<\text{inv FO}} \mathcal{Q}$, $\mathcal{S}upp_{m+1}(\mathcal{Q}') \simeq \mathcal{S}upp_{m+1}(\mathcal{Q})$, h' is a reduced $(m+1)$-pseudo-inclusion of $\mathcal{P}$ in $\mathcal{Q}'$ and $Q' \setminus \mathrm{Im}(h')$ is S-stable in $\mathcal{Q}'$.*

Proof sketch. We consider all the pairs of elements x, y which break the S-stability of $\mathrm{Im}(h)$, i.e. such that $S(x,y)$, $x \in \mathrm{Im}(h)$ and $y \notin \mathrm{Im}(h)$. If there are many of them, then at least two of them are far from each other and we can apply a crossing-S-swap to correct the mapping h. We end up with a bounded number of problematic pairs that can be corrected separately. ◀

6 Removing unnecessary material

In this section we show how to remove the material in $\mathcal{Q}$ that is not present in the image of the pseudo-inclusion of $\mathcal{P}$. From the previous section we can assume that the pseudo-inclusion mapping preserves the S-siblings relation and that its image is S-stable. The remaining part of $\mathcal{Q}$ is then a union of "loops" in the sense that they connect nodes that have the same type. After defining properly the notion of loop, we will use in Section 6.1 a pumping argument in order to reduce the size of the loop to some constant while preserving $\equiv_\alpha^{<\text{inv FO}}$. In Section 6.2 we then show how to remove small loops without affecting the order-invariant equivalence class. Finally, in Section 6.3 we show that if a hollow tree and a hollow quasitree have the same enriched support, then they are $\equiv_\alpha^{<\text{inv FO}}$: this concludes the proof of Theorem 1.

We start with the definition of an abstract loop.

Let $n \in \mathbb{N}$. Let $\mathrm{Type}_\sigma^n[2]$ denote the set of $(n-1)$-types for pairs over the vocabulary $P_\sigma \cup \{E, S\}$, of degree ≤ 4. Let Σ_n be the vocabulary enriching $P_\sigma \cup \{E, S\}$ with two unary symbols J_τ^1 and J_τ^2 for every $\tau \in \mathrm{Type}_\sigma^n[2]$.

Let h be a reduced n-pseudo-inclusion from $\mathcal{P} \in \mathbb{H}_\sigma$ to $\mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^n$, such that $V := Q \setminus \mathrm{Im}(h)$ is S-stable.

Let $\mathcal{Q}_+$ be an extension of $\mathcal{Q}$ to Σ_n obtained in the following way. Since h is reduced, for every $\tau \in \mathrm{Type}_\sigma^n[2]$, there is at most one jumping pair of type τ. If there isn't, J_τ^1 and J_τ^2 are interpreted as the empty set. Else, let $\{x, x'\}$ be this pair, and u' (resp. u) be the E-neighbor of $h(x)$ (resp. $h(x')$) in $[h(x), h(x')]$. Interpret J_τ^1 as $\{h(x), u'\}$ and J_τ^2 as $\{h(x'), u\}$ (the assignments $x \mapsto 1$ and $x' \mapsto 2$ are arbitrary). This is illustrated on the left part of Figure 11, where the double line represents $\mathrm{Im}(h)$. We say that $\mathcal{Q}_+$ is a ***h*-jump-extension of** $\mathcal{Q}$.

We define $\mathcal{V}_+ = \mathrm{Ctxt}_n(\mathcal{Q}_+|_V)$ as the extension of $\mathrm{Ctxt}_n(\mathcal{Q}|_V)$ to Σ_n where every J^i_τ is defined consistently with $\mathcal{Q}_+$ (i.e. $\forall x \in V, \mathcal{V}_+ \models J^i_\tau(x)$ iff $\mathcal{Q}_+ \models J^i_\tau(x)$). This process is illustrated in Figure 11. $\mathcal{V}_+$ is called an **n-abstract loop**. Let $\mathbb{L}^n_\sigma$ be the set of n-abstract loops.

Figure 11 Example of a h-jump-extension $\mathcal{Q}_+$ of $\mathcal{Q}$ (on the left), and its associated abstract loop $\mathcal{V}_+$ of support $V := Q \setminus \mathrm{Im}(h)$ (on the right).

Every Σ_n-structure will have a '+' symbol in its name. When we omit it, we mean the reduction of the structure to $P_\sigma \cup \{E, S\}$ (for instance, from $\mathcal{V}_+ \in \mathbb{L}^n_\sigma$, we get $\mathcal{V} := \mathrm{Ctxt}_n(\mathcal{Q}|_V) \in \mathrm{Ctxt}^n_\sigma$).

6.1 Loop minimization

It will be crucial to bound the size of the loops left by a pseudo-inclusion. The following result does this using a simple pumping argument.

▶ **Proposition 16.** *For every $\alpha, n \in \mathbb{N}$, there exists $N \in \mathbb{N}$ such that for every $\mathcal{P} \in \mathbb{H}_\sigma$, $\mathcal{Q} \in$ quasi-$\mathbb{H}^n_\sigma$ and reduced n-pseudo-inclusion $h : P \to Q$, if $V := Q \setminus \mathrm{Im}(h)$ is S-stable then there exists some $\mathcal{Q}' \in$ quasi-$\mathbb{H}^n_\sigma$ and a reduced n-pseudo-inclusion $h' : P \to Q'$ such that $\mathcal{Q}' \equiv^{<\text{-inv FO}}_\alpha \mathcal{Q}$, $U := Q' \setminus \mathrm{Im}(h')$ is S-stable and $|U| \leq N$.*

6.2 Loop elimination

It now remains to get rid of the small loops. This is a consequence of the "aperiodicity" of $<$-inv FO: we cannot distinguish in $<$-inv FO between k and $k+1$ copies of the same object if k is sufficiently large. Starting from a small loop, we can use the inclusion results of Section 4 to recreate many copies of the loop within $\mathcal{Q}$, then, according to the following proposition, get rid of one copy using aperiodicity.

▶ **Proposition 17.** *$\forall \alpha \in \mathbb{N}, \exists l \in \mathbb{N}, \forall m \in \mathbb{N}, \exists n \in \mathbb{N}, \forall M \in \mathbb{N}, \exists K \in \mathbb{N}$ such that for every abstract loop $\mathcal{U}_+ \in \mathbb{L}^{n+1}_\sigma$ and every $\mathcal{Q} \in$ quasi-$\mathbb{H}^{n+1}_\sigma$ such that $|U| \leq M$, $(l+1) \cdot [\![\mathcal{E}_{n+1}(\mathcal{U})]\!] < [\![\mathcal{E}_{n+1}(\mathcal{Q})]\!]$ and such that for every $(n+1)$-type χ that occurs in $\mathcal{U}$, $|\mathcal{Q}|_\chi \geq K$, there exists $\mathcal{Q}' \in$ quasi-$\mathbb{H}^m_\sigma$ such that $\mathcal{Q}' \equiv^{<\text{-inv FO}}_\alpha \mathcal{Q}$ and $[\![\mathcal{E}_m(\mathcal{Q})]\!] = [\![\mathcal{E}_m(\mathcal{Q}')]\!] + [\![\mathcal{E}_m(\mathcal{U})]\!]$*

Proof sketch. The proof is based on the well known result that first-order formulas of quantifier-rank k cannot distinguish between a linear order of length 2^k and a linear order of length $2^k + 1$ (see, for instance, [10]). Hence if a loop is repeated at least $2^k + 1$ times, we can eliminate one instance without changing the $\equiv^{<\text{-inv FO}}_k$ class of the structure.

First, we include many copies of the loop in $\mathcal{Q}$. The inclusion may not preserve S-edges: the next step is to re-associate these S-edges with crossing-S-swaps in order for these copies to be isomorphic. This is made possible by the hypothesis on the number of occurrences of types appearing in $\mathcal{U}$: it gives us room to make sure the crossing-S-swaps are guarded.

Once this is done, we can remove one copy in a $<$-inv FO-indistinguishable way. ◀

6.3 S-parents re-association

We now turn to the last step of the proof of Theorem 1.

After the removal of the extra material in $\mathcal{Q}$, we have transformed our initial hollow tree $\mathcal{Q}$ into a hollow quasitree having the same number of occurrences of any type as the initial $\mathcal{P}$. They both have the same threads but may differ with their S-edges. The following proposition states that they are $\equiv_\alpha^{<\text{-inv FO}}$, thus ending the proof of Theorem 1.

The techniques used in the proof of the following proposition are strongly reminiscent of those used in [3]; it requires a notion of vertical-S-swaps adapted to hollow trees.

▶ **Proposition 18.** *$\forall \alpha \in \mathbb{N}$, there exists $n_1 \in \mathbb{N}$ such that $\forall \mathcal{P} \in \mathbb{H}_\sigma, \forall \mathcal{Q} \in$ quasi-$\mathbb{H}_\sigma^{n_1}$, if $\mathcal{S}upp_{n_1}(\mathcal{P}) \simeq \mathcal{S}upp_{n_1}(\mathcal{Q})$ then $\mathcal{P} \equiv_\alpha^{<\text{-inv FO}} \mathcal{Q}$.*

7 Conclusion

We have shown that $<$-inv FO $=$ FO over hollow trees. In order to lift this result to structures of pathwidth 2 and bounded degree, it suffices to show that $<$-inv FO $=$ FO over structures that have the same underlying graph than hollows trees, but without the possibility to distinguish a sibling from a child. In other words, there is only one binary relation that is the union of E and S. It turns out that there is a bi-FO-interpretation from structures of pathwidth 2 and bounded degree through this class of structures, as illustrated in Figure 12.

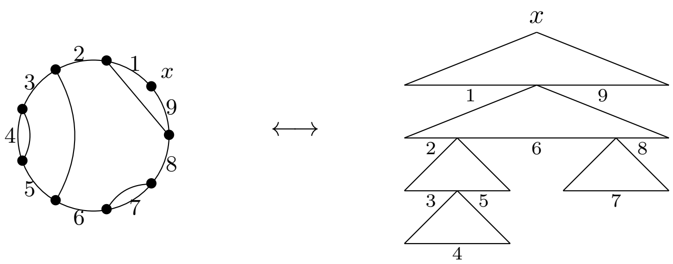

Figure 12 From a typical pathwidth 2 graph of degree 3 to a hollow tree where E and S are indistinguishable.

Unfortunately our proof does not extend to this class of structures as it was crucial in our proof to distinguish between E-swaps and S-swaps. We leave this generalization as an open problem.

We also have no idea yet on what to do when the degree is not assumed to be bounded, as we are then also facing the second difficulty mentioned in the introduction, namely reinterpreting the initial structure within its tree representation.

In this paper we bypassed the first problem mentioned in the introduction, finding similar tree decompositions given similar structures, by working directly on trees. This problem seems unavoidable when working with graphs. There are examples of similar structures of treewidth 2 that do not have any similar tree decompositions of width 2. It might even be the case that for all k there are two similar structures of treewidth 2 that do not have similar tree decomposition of width k. If that were true, completely new ideas would be needed to solve the treewidth 2 case.

References

1. Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases*. Addison-Wesley, 1995. URL: `http://webdam.inria.fr/Alice/`.
2. Matthew Anderson, Dieter van Melkebeek, Nicole Schweikardt, and Luc Segoufin. Locality from Circuit Lower Bounds. *SIAM J. Comput.*, 2012. doi:10.1137/110856873.
3. Michael Benedikt and Luc Segoufin. Regular tree languages definable in FO and in FO_{mod}. *ACM Trans. Comput. Log.*, 2009. doi:10.1145/1614431.1614435.
4. Michael Benedikt and Luc Segoufin. Towards a characterization of order-invariant queries over tame graphs. *J. Symb. Log.*, 2009. doi:10.2178/jsl/1231082307.
5. Hans L. Bodlaender and Joost Engelfriet. Domino Treewidth. *J. Algorithms*, 1997. doi:10.1006/jagm.1996.0854.
6. Michael Elberfeld, Marlin Frickenschmidt, and Martin Grohe. Order Invariance on Decomposable Structures. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS*, 2016. doi:10.1145/2933575.2934517.
7. Martin Grohe and Thomas Schwentick. Locality of order-invariant first-order formulas. *ACM Trans. Comput. Log.*, 2000. doi:10.1145/343369.343386.
8. Frederik Harwath and Nicole Schweikardt. Regular tree languages, cardinality predicates, and addition-invariant FO. In *29th International Symposium on Theoretical Aspects of Computer Science, STACS*, 2012. doi:10.4230/LIPIcs.STACS.2012.489.
9. Frederik Harwath and Nicole Schweikardt. On the locality of arb-invariant first-order logic with modulo counting quantifiers. In *Computer Science Logic, CSL*, 2013. doi:10.4230/LIPIcs.CSL.2013.363.
10. Leonid Libkin. *Elements of Finite Model Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer, 2004. doi:10.1007/978-3-662-07003-1.
11. Hannu Niemistö. On Locality and Uniform Reduction. In *20th IEEE Symposium on Logic in Computer Science, LICS*, 2005. doi:10.1109/LICS.2005.32.
12. Nicole Schweikardt and Luc Segoufin. Addition-Invariant FO and Regularity. In *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS*, 2010. doi:10.1109/LICS.2010.16.
13. Thomas Zeume and Frederik Harwath. Order-Invariance of Two-Variable Logic is Decidable. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS*, 2016. doi:10.1145/2933575.2933594.