The HOM problem is decidable

Guillem Godoy Omer Gimenez Lander Ramos Carme Alvarez

June 26, 2009

Abstract

We close affirmatively a question which has been open for 35 years: decidability of the HOM problem. The HOM problem consists in deciding, given a tree homomorphism H and a regular tree languagle L represented by a tree automaton, whether $H(L)$ is regular.

For deciding the HOM problem, we develop new constructions and techniques which are interesting by themselves, and provide several significant intermediate results. For example, we prove that the universality problem is decidable for languages represented by tree automata with equality constraints, and that the equivalence and inclusion problems are decidable for images of regular languages through tree homomorphisms.

Our contributions are based on the following new results. We describe a simple transformation for converting a tree automaton with equality constraints into a tree automaton with disequality constraints recognizing the complementary language. We also define a new class of automaton with arbitrary disequality constraints and a particular kind of equality constraints. This new class essentially recognizes the intersection of a tree automaton with disequality constraints and the image of a regular language through a tree homomorphism. We prove decidability of emptiness and finiteness for this class by a pumping mechanism. The above constructions are combined adequately to provide an algorithm deciding the HOM problem.

1 Introduction

Finite-state tree automata (TA) [5] were introduced by Thatcher and Wright [17] in the context of circuit verification. Many famous researchers contributed to this school in the late 60's and the early 70's, establishing connections between automata and logic. In the 70's many new results were established concerning TA, which lose a bit their connections with the applications and were studied for their own. Applications of TA to program verification revived in the 80's, after the relative failure of automated deduction in this field. Automata, and in particular TA, also appeared as an approximation of programs on which fully automated tools can be used. New results were obtained connecting properties of programs or type systems or rewrite systems with automata. These applications are widely used nowadays and an extended matter of research.

TA are a well studied formalism for representing term languages, due to their good computational and expressiveness properties. They characterize the "regular term languages", a classical concept used, e.g., to describe the parse trees of a context-free grammar or the well-formed terms over a sorted signature, and to naturally capture type formalisms for tree-structured XML data [15, 2]. Similar as in the case of regular sets of words, regular term languages have numerous convenient properties such as closure under Boolean operations (intersection, union, negation), decidable properties such as inclusion and equivalence, and they are characterized by many different formalisms such as regular grammars, regular term expressions, congruence classes of finite index, deterministic bottom-up TA, nondeterministic top-down TA, or sentences of monadic second-order logic [5]. Deterministic TA, for instance, can be effectively minimized and give rise to efficient parsing.

When the used formalism for representing an infinite set of terms is not a TA, it is often expedient to decide whether the represented set is in fact regular, or when a given transformation preserves regularity. For example, when the set is described as the reachable configurations of a program/system from a starting configuration, regularity of such set allows checking interesting properties of reachable configurations, like validity.

1.1 The HOM problem

Tree homomorphisms are the natural extension of word homomorphisms to trees. They were defined in 1973 as a special case of tree transducers by Thatcher in [16]. From the beginning, it was noted that, the classical property stating that a word homomorphism preserves regularity, is no longer true when one deals with tree languages. For example, the tree language $\{g^n(a) | n \geq 0\}$ is regular since it is recognized by the tree automaton $\{a \to q, g(q) \to q\}$, where q is an accepting state. But its image through the tree homomorphism H defined by $H(a) = a$ and $H(g(x)) = f(H(x), H(x))$ is the set of complete trees over binary f and constant a, which is well-known to be non-regular. Since then, it has been a long-standing open question the decidability of the the HOM problem: given a regular tree language L and a tree homomorphism H, is $H(L)$ regular?

This problem is not only a fundamental theoretical question. Tree homomorphisms are a powerful representation system. Several representation mechanisms based on tree patterns are just particular cases of images of regular languages through three homomorphisms, and the set of reducible instances of any term rewrite system also can be represented in this way. Hence, it is not surprising that particular cases and variants of this problem have been studied along the past 35 years. For example, the connection between tree homomorphisms and tree transducers is widely studied in the handbook [10]. In particular, it is noted that the image of a regular language through a tree homomorphism can be represented as the rank of a bottom-up tree transducer. Regularity of the set of reducible terms by a given term rewrite system was proved decidable independently in [18, 14]. Regularity of the rank of a top-down tree transducer is shown undecidable in [9]. This kind of representation generalizes the one of ranks of bottom-up tree transducers. The HOM problem restricted to shallow homomorphisms was proved decidable in [3]. The HOM problem restricted to monadic signatures or to top-copying homomorphisms was proved decidable in [13].

In this work, we prove decidability of the entire HOM problem, without any further restriction. To this end, we develop new constructions and techniques, which are interesting by themselves, and provide several significant intermediate results. We detail them in the following subsections.

1.2 Tree Automata with constraints

In order to prove decidability of HOM, we make use of several ideas, techniques and results related to tree automata with disequality and equality constraints, $TA_{\neq,=}$ for short, developed along the last 15 years. For this reason, we dedicate this specialized subsection of the introduction to these remarkable results. As an example of tree automaton with equality constraints consider the following: $a \to q,\ f(q,q) \stackrel{1=2}{\to} q$. The constraint $1 = 2$ imposes that the first and second child must coincide for the application of the rule. Thus, the above automaton recognizes the set of complete trees over binary f and constant a.

Tree automata with equality constraints, $TA_=$ for short, have been studied in the early 80s by M. Dauchet and J. Mongy. If we consider its closure by boolean operations, we get the class of automata with equality and disequality constraints. Unfortunately, properties like emptiness, finiteness, and regularity of the represented language are undecidable, even for the restriced class of $TA_=$.

Emptiness of tree automata having only disequality constraints, $TA_{\neq}$ for short, was proved decidable in [7]. Exptime-completeness of this problem was proved in [6]. This result was developed in order to prove exptime-completeness of the ground reducibility problem. The decidability result in [7] was extended in [8] to the so-called class of reduction automata, which allow for arbitrary disequalities and a limited amount of equalities. Roughly speaking, a fixed number of equalities are permitted at each path of a run. This result was used to prove decidability of the first order theory of reduction. Emptiness and finiteness of tree automata with disequality and equality constraints between brothers (direct childs) was proved decidable in [4]. Regularity for this class was proved decidable in [3]. As a consequence, the HOM problem was proved decidable for the particular case of shallow homomorphisms. But also, these results were key for proving decidability of preservation of regularity for shallow term rewrite systems and innermost rewriting in [12]. Several other variants of tree automata with constraints have been developed, providing decidability results in logic and term rewriting. We do not mention them since there are lots, and the provided techniques have not been used in this paper.

1.3 Our contributions

In the path of solving the HOM problem, we develop several ideas and techniques which are interesting by themselves, and produce significant intermediate results.

Our first contribution consists in a construction for converting a $TA_=$ into a $TA_{\neq}$ recognizing the complement language of the first one. This construction is rather easy, but up to our knowledge, it has not been stated before, although similar arguments are used in [7] to construct a $TA_{\neq}$ recognizing the set of normal forms of a term rewrite system. The complement construction has significant consequences, like decidability of the universality problem for tree automata with equality constraints, and more generally, decidability of the inclusion of a regular language into the represented language by a $TA_=$. Moreover, it gives a simple proof of undecidability of regularity test for $TA_{\neq}$, and hence, for reduction tree automata (this question was left open in [3]).

In a second step, we define a new class of automata with constraints called tree automata with disequality and HOM equality constraints, $TA_{\neq,hom}$ for short. Essentially, they recognize the intersection language between a $TA_{\neq}$ and the image of a regular language through a tree homomorphism. We also define the particular subclass of TA_{hom} which recognize images of regular languages through tree homomorphisms.

This new class of tree automata with constraints is interesting by itself. It is essentially a particular subclass of $TA_{\neq,=}$, subsumes the class of tree automata with disequality and equality constraints between brothers, and is independent from the class of reduction automata. In contraposition to reduction automata, $TA_{\neq,=}$ permit an unbounded number of equalities at each path of a run. As we will see, emptiness and finiteness are decidable for $TA_{\neq,hom}$. Moreover, we will show how to construct a $TA_{\neq,hom}$ recognizing the intersection of the represented languages by a $TA_{\neq}$ and a TA_{hom}.

The above constructions and results allow to derive new significant consequences. Since two TA_{hom} A and B represent the images of regular languages through tree homomorphisms, by complementing A and intersecting with B we obtain a $TA_{\neq,hom}$ whose emptiness is equivalent to the inclusion $\mathcal{L}(A) \supseteq \mathcal{L}(B)$. Therefore, we are able to prove decidability of inclusion and equivalence problems for images of regular languages through tree homomorphisms.

Our decision algorithm for the HOM problem has a very simple description. First, it generates a TA_{hom} A recognizing the language $H(L)$. Second, it linearizes A into a TA B by removing all equality constraints and replacing the involved positions in the constraints by all possible valid terms up to a certain height. Third, it checks $\mathcal{L}(A) = \mathcal{L}(B)$, concludes "regularity" in the affirmative case, and concludes "non-regularity" in the negative case.

Nevertheless, proving that this algorithm decides HOM is more complicated, and requires to argue using all the above constructions.

1.4 Organization of the paper

In Section 2 we introduce basic concepts on terms and tree automata. In Section 3 we present the construction transforming a $TA_=$ into a $TA_{\neq}$. In Section 4 we define $TA_{\neq,hom}$ and TA_{hom} and their runs, show that the image of a regular language through a tree homomorphism can be recognized by a TA_{hom}, define the intersection of a $\mathrm{TA}_{hom}$and a $\mathrm{TA}_{\neq}$, the intersection of two runs, and the respective projections to recover the original runs from a run of the intersection. In Section 5, we define the concept of pumping of a run of a $\mathrm{TA}_{\neq,hom}$. Also we prove that for a big enough run there exists a pumping providing a smaller run, thus concluding decidability of the emptiness problem for $\mathrm{TA}_{\neq,hom}$. Moreover, we prove that for a big enough run there exists a pumping providing a bigger run, thus concluding decidability of the finiteness problem for $\mathrm{TA}_{\neq,hom}$. In Section 6 we show all the significant intermediate consequences of our constructions. In Section 7 we use all the developed techniques to prove decidability of the HOM problem.

2 Preliminaries

2.1 Terms

The size of a set S is denoted by $|S|$, and the powerset of S is denoted by 2^S. We assume that the reader is familiarized with terms, positions, substitutions and replacements. For more detailed explanations see [1].

A *signature* consists of an alphabet Σ, i.e., a finite set of symbols, together with a mapping that assigns to each symbol in Σ a natural number, its *arity*. We write $\Sigma^{(m)}$ to denote the subset of symbols in Σ that are of arity m, and we write $f^{(m)}$ to denote that f is a symbol of arity m. The *set of all terms over* Σ is denoted $\mathcal{T}(\Sigma)$ and is inductively defined as the smallest set T such that for every $f \in \Sigma^{(m)}$, $m \geq 0$, and $t_1, \ldots, t_m \in T$, the term $f(t_1, \ldots, t_m)$ is in T. For a term of the form $a()$ we simply write a. For instance, if $\Sigma = \{f^{(2)}, a^{(0)}\}$ then $\mathcal{T}(\Sigma)$ is the set of all terms that represent binary trees with internal nodes labeled f and leaves labeled a. We fix the set $\mathcal{X} = \{x_1, x_2, \ldots\}$ of variables, i.e., any set V of variables is always assumed to be a subset of $\mathcal{X}$. The set of terms over Σ with variables in $\mathcal{X}$, denoted $T(\Sigma \cup \mathcal{X})$, is the set of terms over $\Sigma \cup \mathcal{X}$ where every symbol in $\mathcal{X}$ has arity zero. By $|t|$ we denote the size of t, defined recursively as $|f(t_1, \ldots, t_m)| = 1 + |t_1| + \ldots + |t_m|$ for each $f \in \Sigma^{(m)}$, $m \geq 0$ and $t_1, \ldots, t_m \in \mathcal{T}(\Sigma)$, and $|x| = 1$ for each x in $\mathcal{X}$. By $\mathtt{height}(t)$ we denote the height of t, defined recursively as $\mathtt{height}(f(t_1, \ldots, t_m)) = 1 + \mathtt{max}(\mathtt{height}(t_1), \ldots, \mathtt{height}(t_m))$ for each $f \in \Sigma^{(m)}$, $k \geq 1$ and $t_1, \ldots, t_m \in \mathcal{T}(\Sigma)$, $\mathtt{height}(a) = 0$ for each $a \in \Sigma^{(0)}$, and $\mathtt{height}(x) = 0$ for each $x \in \mathcal{X}$. Positions in terms are sequences of natural numbers. Given a term $f(t_1, \ldots, t_m) \in \mathcal{T}(\Sigma)$, its set of positions $\mathtt{Pos}(t)$ is defined recursively as $\{\lambda\} \cup_{1 \leq i \leq m} \{i.p \mid p \in \mathtt{Pos}(t_i)\}$. Here, λ denotes the empty sequence (position of the root node), and . denotes concatenation. The subterm of t at position p is denoted by $t|_p$, and is formally defined as $t|_\lambda = t$, and $f(t_1, \ldots, t_m)|_{i.p} = t_i|_p$. $\mathtt{root}(f(t_1, \ldots, t_m))$ is f for any symbol f. Thus, the symbol of t occurring at position p is denoted by $\mathtt{root}(t|_p)$, and we say that t at position p is labeled by $\mathtt{root}(t|_p)$. For instance, for $s = g(f(a, b), c)$, $s|_1$ equals $f(a, b)$ and $\mathtt{root}(t|_{1.2})$ is b. For a set Γ, we use $\mathtt{Pos}_\Gamma(t)$ to denote the set of positions of t that are labeled by symbols in Γ. When a position p is of the form $p_1.p_2$, we say that p_1 is a prefix of p and p_2 is a suffix of p. Moreover, $p - p_1$ is p_2. For terms s, t and $p \in \texttt{Pos}(s)$, we denote by $s[t]_p$ the result of replacing the subterm at position p in s by the term t. More formally, $s[t]_\lambda$ is t, and $f(s_1, \ldots, s_m)[t]_{i.p}$ is $f(s_1, \ldots, s_{i-1}, s_i[t]_p, s_{i+1}, \ldots, s_m)$. For instance, $f(f(a,a),a)[a]_1 = f(a,a)$. A *substitution* σ is a mapping from variables to terms. It can be homomorphically extended to a function from terms to terms: $\sigma(t)$ denotes the result of simultaneously replacing in t every $x \in \texttt{Dom}(\sigma)$ by $\sigma(x)$. For example, if σ is $\{x \mapsto f(b,y), y \mapsto a\}$, then $\sigma(g(x,y))$ is $g(f(b,y),a)$. A rewrite rule is a pair of terms $l \to r$. Application of a rewrite rule $l \to r$ to a term $s[\sigma(l)]_p$ at position p produces the term $s[\sigma(r)]_p$. If R is a set of rules, application of a rule of R to a term s resulting into a term t is denoted by $s \to_R t$, and the reflexive-transitive closure of this relation is denoted by $\to_R^*$.

Along paper, unless the opposite is stated, by $t|_{p_1} = t|_{p_2}$ we mean that p_1 and p_2 are positions in $\texttt{Pos}(t)$ and the subterms of t at positions p_1 and p_2 coincide. On the the other side, by $t|_{p_1} \neq t|_{p_2}$ we mean that either p_1 or p_2 is not in $\texttt{Pos}(t)$, or that the subterms $t|_{p_1}$ and $t|_{p_2}$ are different. Note that, with this semantics, $t|_{p_1} \neq t|_{p_2}$ is the negation of $t|_{p_1} = t|_{p_2}$.

2.2 Tree automata with constraints

Tree automata and regular languages are well-known concepts of theoretical computer science[10, 11, 5] We assume that the reader knows the Boolean closure properties of regular tree languages and the decidability results on regular tree languages. Here we only recall the notion of tree automata with constraints.

The subsequent presentation is not the most usual one for tree automata with constraints, but it is an equivalent one. We use this presentation in order to make it more similar to other definitions of automata appearing in the rest of the article, thus preparing the reader for further definitions.

Definition 2.1 *(automata with constraints) A* tree automaton with disequality and equality constraints*, $TA_{\neq,=}$ for short, is a tuple $A = \langle Q, \Sigma, F, \Delta \rangle$, where Q is a set of states, Σ is a signature, $F \subseteq Q$ is the subset of final states, and Δ is a set of rules of the form $f(q_1, \ldots, q_m) \xrightarrow{c} q$, where $q_1, \ldots, q_n, q$ are in Q, f is in $\Sigma^{(m)}$ and c is a conjunction/set of atoms of the form $p_1 \neq p_2$ and $p_1 = p_2$ for arbitrary positions p_1, p_2. When all constraints in Δ contain only disequalities (respectively, equalities) we say that A is a $TA_{\neq}$ (respectively, a $TA_{=}$). When all the constraints are empty, we say that A is a TA.*

In order to define the concept of run of a $TA_{\neq,=}$ we define the alphabet for describing runs on terms, which are just terms with labels indicating which rule has been applied at each node.

Definition 2.2 *(alphabet of a run) Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a $TA_{\neq,=}$. The* alphabet of a run *of A is Δ, where each symbol of the form $(f(q_1, \ldots, q_m) \xrightarrow{c} q)$ has the same arity as this f.*

The resulting state *of a term r in $\mathcal{T}(\Delta)$ is q if r is of the form $(f(q_1, \ldots, q_m) \xrightarrow{c} q)(t_1, \ldots, t_m)$.*

The projection $\pi_\Sigma : \mathcal{T}(\Delta) \to \mathcal{T}(\Sigma)$ *is recursively defined as* $(\pi_\Sigma(f(q_1, \ldots, q_m) \xrightarrow{c} q)(t_1, \ldots, t_m)) = f(\pi_\Sigma(t_1), \ldots, \pi_\Sigma(t_m))$.

Definition 2.3 *(run) Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,=}$. We define the concept of* run *of A as a term in $\mathcal{T}(\Delta)$ satisfying certain conditions recursively as follows. Let $f(q_1, \ldots, q_m) \xrightarrow{c} q$ be a rule of Δ. Let $r_1, \ldots, r_m$ be runs of A with resulting states $q_1, \ldots, q_m$, respectively. Let t be $\pi_\Sigma(f(r_1, \ldots, r_m))$. Suppose $t|_{p_1} \neq t|_{p_2}$ for each $(p_1 \neq p_2) \in c$, and $t|_{p_1} = t|_{p_2}$ for each $(p_1 = p_2) \in c$. Then, $(f(q_1, \ldots, q_m) \xrightarrow{c} q)(r_1, \ldots, r_m)$ is a run of A on the term t.*

By $\mathcal{L}(A, q)$ we denote the set of terms t for which there exists a run r of A with resulting state q such that $\pi_\Sigma(r) = t$. The language accepted by A, denoted $\mathcal{L}(A)$, is $\bigcup_{q\in F} \mathcal{L}(A, q)$. A language L is regular *if there exists a TA A such that $\mathcal{L}(A) = L$ holds.*

Definition 2.4 *(tree homomorphisms) Let Σ_1, Σ_2 be two signatures. A* tree homomorphism *is a function $H : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ which can be defined as follows.*

Let X_m represent the set of variables $\{x_1, \ldots, x_m\}$, for each natural number m. The definition of a tree homomorphism $H : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ requires to define $H(f(x_1, \ldots, x_m))$ for each function symbol $f \in \Sigma_1$ of arity m as a term t_f in $\mathcal{T}(\Sigma_2 \cup X_m)$. After that, $H(f(t_1, \ldots, t_m))$ is defined, for each term $f(t_1, \ldots, t_m) \in \mathcal{T}(\Sigma_1)$ as $\{x_1 \mapsto H(t_1), \ldots, x_m \mapsto H(t_m)\}(t_f)$.

Alternatively, tree homomorphisms can be defined in the following way as a function $H : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ satisfying the following condition. For any arbitrary set of variables $\mathcal{X}$, there exists an extension $\overline{H} : \mathcal{T}(\Sigma_1 \cup \mathcal{X}) \to \mathcal{T}(\Sigma_2 \cup \mathcal{X})$ of H such that, $H(x) = x$ for each x in $\mathcal{X}$, and for each term $t \in \mathcal{T}(\Sigma_1 \cup \mathcal{X})$ and each substitution $\sigma : \mathcal{X} \to \mathcal{T}(\Sigma_2 \cup \mathcal{X})$, $\overline{H}(\sigma(t))$ is $(\overline{H}(\sigma))(\overline{H}(t))$, where $(\overline{H}(\sigma))(x)$ is interpreted in the natural way as $\overline{H}(\sigma(x))$.

Definition 2.5 *The HOM problem is defined as follows:*
Input: *A TA A and a tree homomorphism H.*
Question: *Is $H(A)$ regular?*

3 The complement of a $\text{TA}_=$

For a given $\text{TA}_=$ $A = \langle Q, \Sigma, F, \Delta\rangle$ we want to construct a $\text{TA}_{\neq}$ B recognizing the complement of $\mathcal{L}(A)$. This construction is rather easy. We just need to consider 2^Q as the set of states of B. The intuitive meaning of each state $S \subseteq Q$ is that there exists a run with B of a term t with resulting state S if and only if, for each q in S, there is no run with A of t with resulting state q. In other words, using B we are computing sets of states wich cannot be reached using A. The rules $f(S_1, \ldots, S_m) \xrightarrow{D} S$ of B are constructed to ensure that, for those states q in S, no rule of A with right-hand side q can be applied.

Definition 3.1 *(Complement of a $TA_=$) Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_=$. Then, the* complement *$TA_{\neq}$ B of A is defined as the tuple $\langle 2^Q, \Sigma, \{F' |\ F \subseteq F' \subseteq Q\}, \Delta'\rangle$ where Δ' is the set of rules $f(S_1, \ldots, S_m) \xrightarrow{D} S$ satisfying the following:*

- *$S_1, \ldots, S_m, S \subseteq Q$.*

- *D is a conjunction of disequalities $p \neq q$ such that $p = q$ occurs in the constraint of some rule in Δ.*

- *For each q in S and each rule of the form $f(q_1, \ldots, q_m) \xrightarrow{c} q$ in Δ either there exists some i in $\{1, \ldots, m\}$ satisfying $q_i \in S_i$, or there exist positions p_1, p_2 such that $p_1 = p_2$ occurs in c and $p_1 \neq p_2$ occurs in D.*

Example 3.2 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_=$ with $Q = \{q\}$, $\Sigma = \{f^{(2)}, a^{(0)}\}$, $F = \{q\}$, and where Δ contains the two rules $a \to q$ and $f(q, q) \xrightarrow{1=2} q$. This TA$_=$ recognizes the language of the completes trees. We can construct the complement TA$_{\neq}$ as follows. The set of states is $2^Q = \{\emptyset, \{q\}\}$, the signature is the same Σ, the set of accepting states is $F' = \{\{q\}\}$ and some of the rules in Δ' are:*

- *$a \to \emptyset$*
- *$f(\emptyset, \emptyset) \to \emptyset$*
- *$f(\emptyset, \emptyset) \xrightarrow{1\neq2} \{q\}$*
- *$f(\{q\}, \emptyset) \to \{q\}$*
- *$f(\emptyset, \{q\}) \to \{q\}$*
- *$f(\{q\}, \{q\}) \to \{q\}$*

This automaton clearly recognizes the language of the uncomplete trees (because at each position we can non-deterministically check that either at least one child is uncomplete, or both childs are not equal). According to the definition of the complement of a TA$_=$, there exist more rules in Δ' (like $f(\{q\}, \{q\}) \xrightarrow{1\neq2} \{q\}$), but they are unnecessary or useless.

The following lemma establishes one of the directions of the statement mentioned above: whenever a state S is the result of a run of B on a term t, no state q in S can be the result of a run of A on t.

Lemma 3.3 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_=$, and $B = \langle 2^Q, \Sigma, \{F' |\ F \subseteq F' \subseteq Q\}, \Delta' \rangle$ be its complement TA$_{\neq}$. Let q be a state in Q and let S be a state of B containing q. Let t be a term in $\mathcal{L}(B, S)$.*

Then, t is not in $\mathcal{L}(A, q)$.

Proof. We prove it by contradiction. Let t be a minimum term in size contradicting the statement, i.e. there exists $S \subseteq Q$, a run r' of B satisfying $\pi_B(r') = t$ with resulting state S, a state q in S and a run r of A satisfying $\pi_\Sigma(r) = t$ with resulting state q, and no other term smaller than t acomplishes this statement.

We write t more explicitly as $f(t_1, \ldots, t_m)$, and the above runs r' and r as $(f(S_1, \ldots, S_m) \xrightarrow{D} S)(r'_1, \ldots, r'_m)$ and $(f(q_1, \ldots, q_m) \xrightarrow{c} q)(r_1, \ldots, r_m)$, respectively. Note that $S_1, \ldots, S_m$ are the resulting states of the runs $r'_1, \ldots, r'_m$ of

B on $t_1, \ldots, t_m$, respectively, and $q_1, \ldots, q_m$ are the resulting states of the runs $r_1, \ldots, r_m$ of A on $t_1, \ldots, t_m$, respectively.

By the definition of Δ', since q belongs to S, for the rule $f(q_1, \ldots, q_m) \xrightarrow{c} q$ it holds that either (i) there exists some i in $\{1, \ldots, m\}$ satisfying $q_i \in S_i$, or (ii) there exist positions p_1, p_2 such that $p_1 = p_2$ occurs in c and $p_1 \neq p_2$ occurs in D.

In case (i), t_i, S_i, r'_i, q_i, and r_i satisfy the assumed conditions for t, S, r', q and r, but also $|t_i| < |t|$ holds. This is in contradiction with the minimality of t.

In case (ii), by the definition of run applied on r, it holds $t|_{p_1} = t|_{p_2}$. But, similarly, by the definition of run applied on r', it holds $t|_{p_1} \neq t|_{p_2}$, which is a contradiction again. □

The following lemma establishes the other direction of the above statement, but for maximal S's, that is, given a term t, there exists a run r with B of t whose resulting state S is just the set of states q which cannot be the result of a run with A of t. Moreover, this is also the case for each subrun (subterm) of r.

Lemma 3.4 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_=$, and $B = \langle 2^Q, \Sigma, \{F' | \; F \subseteq F' \subseteq Q\}, \Delta' \rangle$ be its complement TA$_{\neq}$. Let t be a term.*

Then, there exists a run r of B satisfying $\pi_B(r) = t$ and such that, for each $p \in \mathtt{Pos}(r)$ it holds that $r|_p$ is a run with resulting state $\{q \in Q | \; t|_p \notin \mathcal{L}(A, q)\}$.

Proof. We prove it by induction on $|t|$. We write t more explicitly as $f(t_1, \ldots, t_m)$. By induction hypothesis, for each t_i there exists a run r_i of B satisfying $\pi_B(r_i) = t_i$ and such that, for each $p \in \mathtt{Pos}(r_i)$ it holds that $r_i|_p$ is a run with resulting state $\{q \in Q | \; t_i|_p \notin \mathcal{L}(A, q)\}$. In particular, the resulting state S_i of r_i is $\{q \in Q | \; t_i \notin \mathcal{L}(A, q)\}$.

Let D be the constraint defined as the conjunction of disequalities $p_1 \neq p_2$ such that $p_1 = p_2$ occurs in the constraint of some rule of A and $t|_{p_1} \neq t|_{p_2}$ holds (recall that, by $t|_{p_1} \neq t|_{p_2}$, we understand that either p_1 or p_2 is not in $\mathtt{Pos}(t)$, or that the subterms $t|_{p_1}$ and $t|_{p_2}$ are different). In order to conclude, it suffices to prove that $f(S_1, \ldots, S_m) \xrightarrow{D} S$ for $S = \{q \in Q | \; t|_p \notin \mathcal{L}(A, q)\}$ is a rule of B. To this end, we must show, for each q in S and each rule of the form $f(q_1, \ldots, q_m) \xrightarrow{c} q$ in Δ, that either some i in $\{1, \ldots, m\}$ satisfies $q_i \in S_i$, or there exist positions p_1, p_2 such that $p_1 = p_2$ occurs in c and $p_1 \neq p_2$ occurs in D. Thus, consider any of such q and $f(q_1, \ldots, q_m) \xrightarrow{c} q$ and suppose that all i in $\{1, \ldots, m\}$ satisfy $q_i \notin S_i$. Then, by the definition of each of such S_i, there exists a run r'_i of A satisfying $\pi_\Sigma(r'_i) = t_i$ and with resulting state q_i. Since q is in S, by the definition of S it holds that there is no run r' with resulting state q such that $\pi_\Sigma(r') = t$. Hence, $(f(q_1, \ldots, q_m) \xrightarrow{c} q)(r'_1, \ldots, r'_m)$ is not a run. Thus, some $p_1 = p_2$ occurring in the constraint c must be unsatisfied on t, i.e. $t|_{p_1} \neq t|_{p_2}$ for some $p_1 = p_2$ in c. By the election of D, $p_1 \neq p_2$ occurs in D. Thus, the existence of such p_1, p_2 concludes the proof. □

Theorem 3.5 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_=$, and $B = \langle 2^Q, \Sigma, \{F' | \; F \subseteq F' \subseteq Q\}, \Delta' \rangle$ be its complement TA$_{\neq}$. Then, $\overline{\mathcal{L}(A)} = \mathcal{L}(B)$.*

Proof.

- We first prove $\overline{\mathcal{L}(A)} \supseteq \mathcal{L}(B)$. Let t be a term in $\mathcal{L}(B)$. Then, there exists a run r' of B with resulting state S such that $\pi_B(r') = t$ and $F \subseteq S \subseteq Q$ hold. Hence, for each q in F, by Lemma 3.3 it follows that t is not in $\mathcal{L}(A, q)$. Thus, t is not in $\mathcal{L}(A)$.

- Now, we prove $\overline{\mathcal{L}(A)} \subseteq \mathcal{L}(B)$. Let t be a term which is not accepted by A. Then, for each state q of F, $t \notin \mathcal{L}(A, q)$ holds. By Lemma 3.4, t is in $\mathcal{L}(B, S)$ where S is $\{q \in Q |\ t|_p \notin \mathcal{L}(A, q)\}$, and since $F \subseteq S \subseteq Q$ holds, it follows that t is in $\mathcal{L}(B)$.

□

Proceeding analogously we could transform a TA$_{\neq}$ into a TA$_{=}$ recognizing the complement of the first. We do not develop this transformation since it is very similar and we do not use it for the proof of the HOM problem, but state the analogous consequence as follows.

Theorem 3.6 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_{\neq}$. Then, it can be computed a TA$_{=}$ $B = \langle 2^Q, \Sigma, \{F' |\ F \subseteq F' \subseteq Q\}, \Delta' \rangle$, called the complement TA$_{=}$ of A, such that. $\overline{\mathcal{L}(A)} = \mathcal{L}(B)$.*

4 Tree automata with disequality and HOM equality constraints

Our aim is to define a new class of automata, with a certain kind of equality constraints, recognizing images of tree homomorphisms applied to regular languages. These will be the tree automata with HOM constraints, denoted TA$_{hom}$, and they, essentially recognize the rank of a bottom up tree transducer. But we will define a more general class including also arbitrary disequality constraints, denoted TA$_{\neq,hom}$. The reason is that we will need to argue about the intersection language of the languages represented by a TA$_{hom}$and a TA$_{\neq}$.

The definition of TA$_{\neq,hom}$ has differences from the definition of TA$_{\neq,=}$. The left-hand side of rules are not necessarily flat. Thus, they directly use information of states computed at relative positions deeper than 1. The disequality constraints are arbitrary, but the equality constraints are restricted. They always refer to positions with identical computed states. The rules are also labelled. The labels are not relevant at all for further definitions of run, pumping, etc. We will use them later, when intersecting two automata, for keeping the necessary information to recover the runs of the two original automata from a run of their intersection automaton.

Definition 4.1 *(new automata) A* tree automaton with disequality and HOM equality constraints*, TA$_{\neq,hom}$ for short, is a tuple $A = \langle Q, \Sigma, F, \Delta \rangle$, where Q is a set of states, Σ is a signature, $F \subseteq Q$ is the subset of final states, and Δ is a set of labelled rules of the form $I : s \xrightarrow{c} q$, where I is the label, s is* *a term over $\mathcal{T}(\Sigma \cup Q) - Q$, interpreting the states of Q as 0-ary symbols, and c is a conjunction/set of atoms of the form $p_1 \neq p_2$ for arbitrary positions p_1, p_2, and atoms of the form $\hat{p}_1 = \hat{p}_2$, where $\hat{p}_1$ and $\hat{p}_2$ are positions satisfying $s|_{\hat{p}_1} = s|_{\hat{p}_2} \in Q$. Moreover, for all positions $\hat{p}_1, \hat{p}_2, \hat{p}_3$, if $(\hat{p}_1 = \hat{p}_2)$ and $(\hat{p}_2 = \hat{p}_3)$ occur in c, then $(\hat{p}_1 = \hat{p}_3)$ also occurs in c. By $h(A)$ we denote the maximum among the heights of left-hand sides of rules in Δ and the lengths of positions occurring in the constraints of Δ, and write h when A is clear from the context. When no constraint in Δ contains a disequality, we say that A is a TA$_{hom}$.*

Example 4.2 *As an example of TA$_{hom}$ consider $A_{\mathtt{ex}} = \langle Q, \Sigma, F, \Delta \rangle$, where $Q = F = \{q\}$, $\Sigma = \{f^{(3)}, h^{(1)}, a^{(0)}\}$, and Δ is the set of rules $\{\rho_1 = I_1 : a \rightarrow q, \rho_2 = I_2 : h(a) \rightarrow q, \rho_3 = I_3 : h(f(q,q,q)) \xrightarrow{1.1=1.2} q\}$. Note that the equality constraint refers to positions with the same state, which is mandatory for this kind of automata.*

As in the case of tree automata with constraints, in order to define the concept of run of a TA$_{\neq,hom}$ we define the alphabet for describing runs on terms, which are just terms with additional labels indicating which rule has been applied at each node. The difference with respect to the case of plain tree automata with constraints is that now we may also use function symbols in Σ for defining runs, but not only rules. Roughly speaking, this is because the rules are not applied at all the positions of a term. The projection π_Σ is overloaded to this alphabet.

Definition 4.3 *(alphabet of a run) Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_{\neq,hom}$. The* alphabet of a run *of A is $\Sigma \cup \Delta$, where each rule $(I : s \xrightarrow{c} q)$ has the same arity as the symbol $\mathtt{root}(s)$.*

The resulting state *of a term r in $\mathcal{T}(\Sigma \cup \Delta)$ is q if r is of the form $(I : s \xrightarrow{c} q)(t_1, \ldots, t_m)$.*

The projection *$\pi_\Sigma : \mathcal{T}(\Sigma \cup \Delta) \rightarrow \mathcal{T}(\Sigma)$ is recursively defined as $\pi_\Sigma(f(t_1, \ldots, t_m)) = f(\pi_\Sigma(t_1), \ldots, \pi_\Sigma(t_m))$ and as $\pi_\Sigma((I : s \xrightarrow{c} q)(t_1, \ldots, t_m)) = (\mathtt{root}(s))(\pi_\Sigma(t_1), \ldots, \pi_\Sigma(t_m))$ for each rule $(I : s \xrightarrow{c} q)$ in Δ.*

For a term t in $\mathcal{T}(\Sigma \cup \Delta)$ and a position p in $\mathtt{Pos}(t)$, we say that p is a Δ position *(of t) if $\mathtt{root}(t|_p)$ is in Δ. And in the case where p is not λ and the only proper prefix of p being a Δ position is λ, we say that p is a* first Δ position *(of t). We will usually denote with a hat ($\hat{p}$) the first Δ positions.*

Runs of TA$_{\neq,hom}$ are defined similarly to runs of TA$_{\neq,=}$. One of the differences is that, for a given equality $p_1 = p_2$, while a TA$_{\neq,=}$ checks for equality of the projected terms at the relative positions p_1 and p_2, a TA$_{\neq,hom}$ checks for equality of the subruns at the relative positions p_1 and p_2, before projecting. This difference is not relevant, since by interpreting equalities in the usual way we would have the same expressiveness, as can be seen in the proof of Lemma 4.8. We have chosen it for presentation purposes, thus making the proofs easier.

Definition 4.4 *(run) Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_{\neq,hom}$. We define the concept of a* run *of A as a term in $\mathcal{T}(\Sigma \cup \Delta)$ satisfying certain conditions recursively* *as follows. Let* $(I : s \xrightarrow{c} q)$ *be a rule of* Δ*, where* s *is of the form* $f(s_1, \ldots, s_m)$*. Let* $P = \{\hat{p}_1, \ldots, \hat{p}_n\}$ *be the set of positions of* s *such that* $(\hat{p} \in P \Leftrightarrow s|_{\hat{p}} \in Q)$ *holds. Let* $q_1, \ldots, q_n$ *be* $s|_{\hat{p}_1}, \ldots, s|_{\hat{p}_n}$*, respectively. Let* $r_1, \ldots, r_n$ *be runs of* A *with resulting states* $q_1, \ldots, q_n$*, respectively. Let* s' *be* $\pi_\Sigma(s[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n})$*. Suppose* $r_i = r_j$ *for each* $(\hat{p}_i = \hat{p}_j) \in c$*, and* $s'|_{p_1} \neq s'|_{p_2}$ *for each* $(p_1 \neq p_2) \in c$*. Then,* $(I : s \xrightarrow{c} q)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$ *is a* run *of* A*. Note that* P *is the set of first* Δ *positions of* r*.*

By $\mathcal{L}(A, q)$ *we denote the set of terms* t *for which there exists a run* r *of* A *with resulting state* q *such that* $\pi_\Sigma(r) = t$*. The language accepted by* A*, denoted* $\mathcal{L}(A)$*, is* $\bigcup_{q \in F} \mathcal{L}(A, q)$*.*

Example 4.5 *(Following example 4.2) The term* $r_{\texttt{ex}} = \rho_3(f(\rho_1, \rho_1, \rho_2(a)))$ *is a run of* $A_{\texttt{ex}}$ *with projection* $\pi_\Sigma(r_{\texttt{ex}}) = h(f(a, a, h(a)))$*.*

The following lemma establishes that TA_{hom} can be used to represent images of regular languages through tree homomorphisms.

Proposition 4.6 *Let* $B = \langle Q, \Sigma_1, F, \Delta \rangle$ *be a TA. Let* $H : \mathcal{T}(\Sigma_1) \to \mathcal{T}(\Sigma_2)$ *be a tree homomorphism. Then, it can be computed a* TA_{hom} A *satisfying* $H(\mathcal{L}(B)) = \mathcal{L}(A)$*.*

Proof. We define A as $\langle Q, \Sigma_2, F, \Delta' \rangle$ where Δ' is defined as follows. Let Δ'' be the set of rules of the form $\sigma(t_f) \xrightarrow{c} q$, for substitutions σ and terms t_f satisfying the following conditions:

- There exists a rule $f(q_1, \ldots, q_m) \to q$ in Δ for a function symbol f of arity m such that $H(f(x_1, \ldots, x_m)) = t_f$.
- Moreover, c is the conjunction of equalities $(\hat{p}_1 = \hat{p}_2)$ such that $t_f|_{\hat{p}_1}$ and $t_f|_{\hat{p}_2}$ are the same variable.
- The substitution σ is $\{x_1 \mapsto q_1, \ldots, x_m \mapsto q_m\}$.

Now, we define Δ' as the set obtained by closing Δ'' by the fixpoint computation $\Delta'' := \Delta'' \cup \{s \xrightarrow{c} q | \exists (s \xrightarrow{c} q'), (q' \to q) \in \Delta'' : (s \notin Q)\}$, and afterwards by removing from Δ'' all rules of the form $q' \to q$. It is straightforward to see that $H(\mathcal{L}(B))$ equals $\mathcal{L}(A)$ by induction of the size of the involved terms. $\square$

Example 4.7 *Let* $A_r = \langle Q, \Sigma, F, \Delta \rangle$ *be a* TA *satisfying* $Q = F = \{q\}$*,* $\Sigma = \{a^{(0)}, b^{(0)}, g^{(2)}\}$ *and* $\Delta = \{a \to q, b \to q, g(q, q) \to q\}$*. That is, a TA recognizing the set of all trees with a binary symbol* g *and leaves* a *and* b*. Let* $\Sigma_{\texttt{ex}}$ *be the signature of the* TA_{hom} $A_{\texttt{ex}}$ *of example 4.2. Let* $H : \mathcal{T}(\Sigma) \to \mathcal{T}(\Sigma_{\texttt{ex}})$ *be the tree homomorphism defined as follows:*

- $H(a) = a$
- $H(b) = h(a)$
- $H(g(x_1, x_2)) = h(f(x_2, x_2, x_1))$

The construction of Proposition 4.6 produces $A_{\texttt{ex}}$, and it is easy to prove that $H(\mathcal{L}(A_r)) = \mathcal{L}(A_{\texttt{ex}})$ holds.

The following lemma establishes that a TA$_{hom}$ is essentially a particular case of a TA$_=$, that is, for each TA$_{hom}$ we can construct a TA$_=$ recognizing the same language. It will be useful to obtain a TA$_{\neq}$ recognizing the complement of a TA$_{hom}$. The difficult point to prove this inclusion is the fact that equalities of a TA$_{hom}$ ask for identical runs, while equalities of a TA$_=$ ask just for identical projected runs. But this is easy to solve by transforming a run of the constructed TA$_=$ in order to make that projected runs also identical as runs.

Lemma 4.8 *Given a TA$_{hom}$ $A_{hom} = \langle Q, \Sigma, F, \Delta\rangle$, a TA$_=$ $A_=$ can be computed satisfying $\mathcal{L}(A_{hom}) = \mathcal{L}(A_=)$. Moreover, the set of states Q' of $A_=$ includes Q, and for each q in Q, $\mathcal{L}(A_{hom}, q) = \mathcal{L}(A_=, q)$ holds.*

Proof. For making the proof easier to read, the Q' we will define does not include Q, but it includes a set $\{\mathbf{q}_q | q \in Q\}$, and by renaming each $\mathbf{q}_q$ to q the result holds.

We define $A_=$ as $\langle Q', \Sigma, F', \Delta'\rangle$, where Q' is $\{\mathbf{q}_t | t \text{ is a proper subterm of a left-hand side of a rule in } \Delta\} \cup \{\mathbf{q}_q | q \in Q\}$, F' is $\{\mathbf{q}_q | q \in F\}$ and Δ' is $\{f(\mathbf{q}_{t_1}, \ldots, \mathbf{q}_{t_m}) \rightarrow \mathbf{q}_{f(t_1,\ldots,t_m)} | \mathbf{q}_{f(t_1,\ldots,t_m)} \in Q'\} \cup \{f(\mathbf{q}_{t_1}, \ldots, \mathbf{q}_{t_m}) \xrightarrow{c} \mathbf{q}_q | (f(t_1, \ldots, t_m) \xrightarrow{c} q) \in \Delta\}$. It remains to prove $\mathcal{L}(A_{hom}) = \mathcal{L}(A_=)$, and, more in general, $\mathcal{L}(A_{hom}, q) = \mathcal{L}(A_=, \mathbf{q}_q)$ for each q in Q.

- We prove $\mathcal{L}(A_{hom}, q) \subseteq \mathcal{L}(A_=, \mathbf{q}_q)$ by induction on the size of the terms. Let t be a term in $\mathcal{L}(A_{hom}, q)$. Let r be a run of A_{hom} with resulting state q such that $\pi_\Sigma(r) = t$. Let r be of the form $(f(s_1, \ldots, s_m) \xrightarrow{c} q)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$. Let $t_1, \ldots, t_n$ be the terms $\pi_\Sigma(r_1), \ldots, \pi_\Sigma(r_n)$, respectively. Let $q_1, \ldots, q_n$ be the resulting states of $r_1, \ldots, r_n$. Note that each t_i is in its corresponding $\mathcal{L}(A_{hom}, q_i)$. Thus, by induction hypothesis, each t_i is also in its corresponding $\mathcal{L}(A_=, \mathbf{q}_{q_i})$. Therefore, there exist runs $r'_1, \ldots, r'_n$ of $A_=$ with resulting states $\mathbf{q}_{q_1}, \ldots, \mathbf{q}_{q_n}$, respectively, and such that $\pi_\Sigma(r'_1) = t_1, \ldots, \pi_\Sigma(r'_n) = t_n$ hold. For each i in $\{1, \ldots, m\}$ we define s'_i as the term satisfying $\texttt{Pos}(s'_i) = \texttt{Pos}(s_i)$, and for each p in $\texttt{Pos}(s_i)$, either $s_i|_p$ is in Q and $\texttt{root}(s'_i|_p)$ is $\mathbf{q}_{s_i|_p}$, or $\texttt{root}(s_i|_p)$ is a symbol g in Σ of a certain arity k and $\texttt{root}(s'_i|_p)$ is $g(\mathbf{q}_{s_i|_{p.1}}, \ldots, \mathbf{q}_{s_i|_{p.k}}) \rightarrow \mathbf{q}_{s_i|_p}$. It is straightforward to check that $r' = (f(\mathbf{q}_{s_1}, \ldots, \mathbf{q}_{s_m}) \xrightarrow{c} \mathbf{q}_q)(s'_1, \ldots, s'_m)[r'_1]_{\hat{p}_1} \ldots [r'_n]_{\hat{p}_n}$ is a run of $A_=$ satisfying $\pi_\Sigma(r') = t$.

- We prove $\mathcal{L}(A_{hom}, q) \supseteq \mathcal{L}(A_=, \mathbf{q}_q)$ by induction on the size of the involved terms. Let t be a term in $\mathcal{L}(A_=, \mathbf{q}_q)$. Let r be a run of $A_=$ with resulting state $\mathbf{q}_q$ such that $\pi_\Sigma(r) = t$. Let $\texttt{root}(r)$ be of the form $f(\mathbf{q}_{s_1}, \ldots, \mathbf{q}_{s_m}) \xrightarrow{c} \mathbf{q}_q$. By the definition of $A_=$, the existence of this rule in Δ' implies the existence of the rule $f(s_1, \ldots, s_m) \xrightarrow{c} q$

in Δ. Let $\hat{p}_1, \ldots, \hat{p}_n$ be the positions $\hat{p}_i$ satisfying $f(s_1, \ldots, s_m)|_{\hat{p}_i} \in Q$. Again by the definition of $A_=$, r is necessarily of the form $(f(\mathbf{q}_{s_1}, \ldots, \mathbf{q}_{s_m}) \xrightarrow{c} \mathbf{q}_q)(s'_1, \ldots, s'_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$, where each s'_i is the term satisfying $\texttt{Pos}(s'_i) = \texttt{Pos}(s_i)$, and for each p in $\texttt{Pos}(s_i)$, either $s_i|_p$ is in Q and $\texttt{root}(s'_i|_p)$ is $\mathbf{q}_{s_i|_p}$, or $\texttt{root}(s_i|_p)$ is a symbol g in Σ of a certain arity k and $\texttt{root}(s'_i|_p)$ is $g(\mathbf{q}_{s_i|_{p.1}}, \ldots, \mathbf{q}_{s_i|_{p.k}}) \to \mathbf{q}_{s_i|_p}$. Moreover, $r_1, \ldots, r_n$ are runs of $A_=$ with resulting states $\mathbf{q}_{s|_{\hat{p}_1}}, \ldots, \mathbf{q}_{s|_{\hat{p}_n}}$, respectively, where s is $f(s_1, \ldots, s_m)$, and such that $\pi_\Sigma(r_1) = t|_{\hat{p}_1}, \ldots, \pi_\Sigma(r_n) = t|_{\hat{p}_n}$ hold. By induction hypothesis, there exist runs $r'_1, \ldots, r'_n$ of A_{hom} with resulting states $s|_{\hat{p}_1}, \ldots, s_{\hat{p}_n}$, respectively, such that $\pi_\Sigma(r'_1) = t|_{\hat{p}_1}, \ldots, \pi_\Sigma(r'_n) = t|_{\hat{p}_n}$ hold. Note that, it could be the case that, for some $1 \leq i < j \leq n$, a constraint $(\hat{p}_i = \hat{p}_j)$ occurs in c, thus $t|_{\hat{p}_i} = t|_{\hat{p}_j}$ holds, but the runs r'_i and r'_j are different. For this reason, $(f(s_1, \ldots, s_m) \xrightarrow{c} q)(s_1, \ldots, s_m)[r'_1]_{\hat{p}_1} \ldots [r'_n]_{\hat{p}_n}$ is not necessarily a run. Let $r''_1, \ldots, r''_n$ be defined inductively as follows for each j in $\{1, \ldots, n\}$. If, for some $i < j$, a constraint $(\hat{p}_i = \hat{p}_j)$ occurs in c, then r''_j is defined as r''_i. Otherwise r''_j is defined as r'_j. It is straightforward to check that $r'' = (f(s_1, \ldots, s_m) \xrightarrow{c} q)(s_1, \ldots, s_m)[r''_1]_{\hat{p}_1} \ldots [r''_n]_{\hat{p}_n}$ is a run of A_{hom} with resulting state q and such that $\pi_\Sigma(r'')$ is t. □

□

From Lemmas 4.8, 3.3 and 3.4, the following corollary follows.

Corollary 4.9 *Given a TA$_{hom}$ $A_{hom} = \langle Q, \Sigma, F, \Delta\rangle$, a TA$_{\neq}$ $A_{\neq}$ can be computed satisfying the following conditions.*

- *The set of states of $A_{\neq}$ is $2^{Q'}$, where Q' is a set including Q.*
- *For each state q in Q, and each state S in $2^{Q'}$ containing q and each term t in $\mathcal{L}(A_{\neq}, S)$, t is not in $\mathcal{L}(A_{hom}, q)$.*
- *For each term t in $\mathcal{T}(\Sigma)$, there exists a run r of $A_{\neq}$ satisfying $\pi_\Sigma(r) = t$ and such that, for each $p \in \texttt{Pos}(r)$, $r|_p$ is a run of $A_{\neq}$ with a resulting state S satisfying $S \cap Q = \{q \in Q |\ t|_p \notin \mathcal{L}(A_{hom}, q)\}$.*

Definition 4.10 *Given a TA$_{hom}$ A_{hom}, by $\overline{A_{hom}}$ we define the TA$_{\neq}$ provided by Corollary 4.9.*

Now, we provide the necessary definitions and lemmas to intersect a TA$_{\neq}$ and a TA$_{hom}$ producing a TA$_{\neq,hom}$, and to intersect the corresponding runs. Here is when the labels play an important role, keeping the necessary information of the original runs in order to be able to recover the original runs by projection from a run of the produced TA$_{\neq,hom}$. We do not give the proofs of these lemmas, since they are rather straightforward. We do not mind about the labels of the initial TA$_{hom}$, and obviate them.

Definition 4.11 *(Intersection of a TA_{hom} and a $TA_{\neq}$) Let $A = \langle Q_A, \Sigma, F_A, \Delta_A\rangle$ be a TA_{hom}. Let $B = \langle Q_B, \Sigma, F_B, \Delta_B\rangle$ be a $TA_{\neq}$. We define the $TA_{\neq,hom}$ $A \cap B$ as $\langle Q_A \times Q_B, \Sigma, F_A \times F_B, \Delta\rangle$, where Δ is the set of rules $(I : s \xrightarrow{c} \langle q, q'\rangle)$ satisfying the following conditions.*

- *s is a term in $\mathcal{T}(\Sigma \cup (Q_A \times Q_B))$.*
- *there exists a rule $s' \xrightarrow{c'} q$ in Δ_A satisfying $\mathtt{Pos}_\Sigma(s') = \mathtt{Pos}_\Sigma(s)$ and for each $p \in \mathtt{Pos}_\Sigma(s')$, $\mathtt{root}(s'|_p) = \mathtt{root}(s|_p)$ holds.*
- *For each $\hat{p} \in \mathtt{Pos}(s)$ such that $s|_p$ is of the form $\langle q', q''\rangle$, $s'|_p$ is q'.*
- *I is a mapping $I : \mathtt{Pos}_\Sigma(s) \to \Delta_B$ satisfying the following conditions for each p in $\mathtt{Pos}_\Sigma(s)$: If $I(p)$ is of the form $f(q_1, \ldots, q_m) \xrightarrow{c''} q$, then $\mathtt{root}(s|_p)$ is f, and for each i in $\{1, \ldots, m\}$, either $s|_{p.i}$ is a state of the form $\langle \_, q_i\rangle$, or $I(p.i)$ is defined as a rule with right-hand side q_i.*
- *c is the set $c' \cup \{p.p_1 \neq p.p_2 | p \in \mathtt{Pos}_\Sigma(s) \wedge\ \exists t, c'', q' : (I(p) = (t \xrightarrow{c''} q') \wedge (p_1 \neq p_2) \in c'')\}$.*

Example 4.12 *Let $A_d = \langle Q'\Sigma, F', \Delta'\rangle$ be a $TA_{\neq}$ with $Q' = F' = \{q'\}$, $\Sigma = \Sigma_{\mathtt{ex}}$ (the signature of the TA_{hom} $A_{\mathtt{ex}}$ of example4.2), and where Δ' contains the rules:*

- *$\bar{\rho}_1 = a \to q'$*
- *$\bar{\rho}_2 = f(q', q', q') \xrightarrow{2 \neq 3} q'$*
- *$\bar{\rho}_3 = h(q') \xrightarrow{1.1 \neq 1.3} q'$*
- *$\bar{\rho}_4 = h(q') \xrightarrow{1.1 \neq 1.2} q'$*

We can compute the $TA_{\neq,hom}$ $\hat{A} = A_{\mathtt{ex}} \cap A_d$ obtaining $\langle Q \times Q', \Sigma, F \times F', \Delta\rangle$, where Δ contains the following rules:

- *$\hat{\rho}_1 = a \to \langle q, q'\rangle$, with label $I(\lambda) = \bar{\rho}_1$.*
- *$\hat{\rho}_2 = h(a) \xrightarrow{1.1 \neq 1.3} \langle q, q'\rangle$, with label $I(\lambda) = \bar{\rho}_3$, $I(1) = \bar{\rho}_1$. (Note that the constraint is always satisfied.)*
- *$\hat{\rho}_3 = h(a) \xrightarrow{1.1 \neq 1.2} \langle q, q'\rangle$, with label $I(\lambda) = \bar{\rho}_4$, $I(1) = \bar{\rho}_1$. (Note that the constraint is always satisfied.)*
- *$\hat{\rho}_4 = h(f(\langle q, q'\rangle, \langle q, q'\rangle, \langle q, q'\rangle)) \xrightarrow{\substack{1.1 = 1.2 \\ 1.2 \neq 1.3 \\ 1.1 \neq 1.3}} \langle q, q'\rangle$, with label $I(\lambda) = \bar{\rho}_3$, $I(1) = \bar{\rho}_2$.*
- *$\hat{\rho}_5 = h(f(\langle q, q'\rangle, \langle q, q'\rangle, \langle q, q'\rangle)) \xrightarrow{\substack{1.1 = 1.2 \\ 1.2 \neq 1.3 \\ 1.1 \neq 1.2}} \langle q, q'\rangle$, with label $I(\lambda) = \bar{\rho}_4$, $I(1) = \bar{\rho}_2$, which is useless because the constraint is unsatisfiable.*

Proposition 4.13 *Let A be a TA_{hom}. Let B be a $TA_{\neq}$. Then, $\mathcal{L}(A \cap B) = \mathcal{L}(A) \cap \mathcal{L}(B)$.*

Definition 4.14 *Let $A = \langle Q_A, \Sigma, F_A, \Delta_A \rangle$ be a TA_{hom}. Let $B = \langle Q_B, \Sigma, F_B, \Delta_B \rangle$ be a $TA_{\neq}$. Let $r = (I : s \xrightarrow{c} \langle q, q' \rangle)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$ be a run of $A \cap B$.*

The projected run *$\pi_{hom}(r)$ is defined rescursively as $(I : s \xrightarrow{c} q)(s_1, \ldots, s_m)[\pi_{hom}(r_1)]_{\hat{p}_1} \ldots [\pi_{hom}(r_n)]_{\hat{p}_n}$.*

The projected run *$r' = \pi_{\neq}(r)$ is defined recursively as follows. Let $r'_1, \ldots, r'_n$ be $\pi_{\neq}(r_1), \ldots, \pi_{\neq}(r_n)$, respectively. Let s' be any term satisfying $\texttt{Pos}(s') = \texttt{Pos}(s)$, and for each $p \in \texttt{Pos}_\Sigma(s)$ it holds that $\texttt{root}(s'|_p)$ is $I(p)$. Then, $\pi_{\neq}(r)$ is defined as $s'[r'_1]_{\hat{p}_1} \ldots [r'_n]_{\hat{p}_n}$.*

Proposition 4.15 *Let $A = \langle Q_A, \Sigma, F_A, \Delta_A \rangle$ be a TA_{hom}. Let $B = \langle Q_B, \Sigma, F_B, \Delta_B \rangle$ be a $TA_{\neq}$. Let r be a run of $A \cap B$. Then, $\pi_{hom}(r)$ is a run of A, and $\pi_{\neq}(r)$ is a run of B.*

Lemma 4.16 *Let $A = \langle Q_A, \Sigma, F_A, \Delta_A \rangle$ be a TA_{hom}. Let $B = \langle Q_B, \Sigma, F_B, \Delta_B \rangle$ be a $TA_{\neq}$. Let r_1, r_2 be runs of $A \cap B$. Let p be a position in $\texttt{Pos}(r_1) \cap \texttt{Pos}(r_2)$ such that $\texttt{root}(r_1|_p) = \texttt{root}(r_2|_p)$ holds.*

Then, $(\pi_{hom}(r_1))|_p = (\pi_{hom}(r_2))|_p$ holds. Moreover, if for all prefix p' of p, $\texttt{root}(r_1|_{p'}) = \texttt{root}(r_2|_{p'})$ holds, then $(\pi_{\neq}(r_1))|_p = (\pi_{\neq}(r_2))|_p$ holds.

Definition 4.17 *(intersection of runs of a TA_{hom} and a $TA_{\neq}$) Let t be a term in $\mathcal{T}(\Sigma)$, and let $r_A = (s \xrightarrow{c} q)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$ and r_B be runs of A and B such that $\pi_\Sigma(r_A) = \pi_\Sigma(r_B) = t$. The* intersected run *$r = r_A \cap r_B$ is defined recursively to satisfy the following conditions:*

- $\texttt{Pos}(r) = \texttt{Pos}(r_A) = \texttt{Pos}(r_B)$.
- *For each i in $\{1, \ldots, n\}$, $r|_{\hat{p}_i} = (r_i \cap (r_B|_{\hat{p}_i}))$ holds.*
- *For each p in $\texttt{Pos}_\Sigma(s) - \{\lambda\}$, $\texttt{root}(r|_p) = \texttt{root}(r_A|_p)$.*
- *$\texttt{root}(r)$ is $(I : s \xrightarrow{c} \langle q, q' \rangle)$, where q' is the resulting state of r_B, and for each p in $\texttt{Pos}_\Sigma(s)$, $I(p) = r_B|_p$ holds.*

Example 4.18 *(Continues example 4.12) The following run $\hat{r}$ is a run of $\hat{A}$: $\hat{r} = \hat{\rho}_4(f(\hat{\rho}_1, \hat{\rho}_1, \hat{\rho}_2(a)))$, with projection $\pi_\Sigma(\hat{r}) = h(f(a, a, h(a)))$, and it is the intersection of the runs $\pi_{hom}(\hat{r}) = \rho_3(f(\rho_1, \rho_1, \rho_2(a)))$ (but without labels) and $\pi_{\neq}(\hat{r}) = \bar{\rho}_3(\bar{\rho}_2(\bar{\rho}_1, \bar{\rho}_1, \bar{\rho}_3(\bar{\rho}_1)))$.*

Proposition 4.19 *Let A be a TA_{hom}. Let B be a $TA_{\neq}$. Let r_A, r_B be runs of A and B, respectively. Then, $\pi_{\neq}(r_A \cap r_B) = r_B$ and $\pi_{hom}(r_A \cap r_B) = r_A$ hold.*

5 Pumpings

Pumping is a traditional concept in automata theory, and in particular, they are very useful to reason about tree automata. The basic idea is to convert a given run r into another run by replacing a subterm at a certain position p in r by a run r', thus obtaining a run $r[r']_p$. For plain tree automata, the necessary and sufficient condition to ensure that $r[r']_p$ is a run is that the resulting states of $r|_p$ and r' coincide, since application of a rule at a certain position depends only on the resulting states of the subruns of the direct childs. When the tree automata has equality and disequality constraints, the constraints may be falsified when replacing a subrun by a new run. For $\mathrm{TA}_{\neq,hom}$, we will define a notion of pumping ensuring that the equality constraints are satisfied, while nothing is guaranteed for the disequality constraints.

Definition 5.1 *(pumping of a run) Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r, r' be runs of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A and* $\mathtt{root}(r|_{\bar{p}}) = \mathtt{root}(r')$ *holds. The* pumping *of r' into r at position $\bar{p}$, denoted $r[[r']]_{\bar{p}}$, is a term in $\mathcal{T}(\Sigma \cup \Delta)$ defined recursively as follows. Let r be of the form $\langle f, I : s \xrightarrow{c} q\rangle(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$.*

Suppose first that $\bar{p}$ is λ. Then, $r[[r']]_{\bar{p}}$ is r'.

Otherwise, suppose that $\bar{p}$ is of the form $\hat{p}_i.\bar{p}'$ for some i in $\{1, \ldots, n\}$, and let r'_i be $r_i[[r']]_{\bar{p}'}$. Then, $r[[r']]_{\bar{p}}$ is $r[r'_1]_{\hat{p}_1} \ldots [r'_n]_{\hat{p}_n}$, where each r'_j is defined as r'_i in the case where j is i or $(\hat{p}_i = \hat{p}_j)$ occurs in c, and r'_j is defined as r_j otherwise.

The case where $\bar{p}$ is not λ and no $\hat{p}_i$ is a prefix of $\bar{p}$ is not possible, by the condition that $r|_{\bar{p}}$ is a run.

Note that $r[[r']]_{\bar{p}}$ is just a new term in $\mathcal{T}(\Sigma \cup \Delta)$. Nevertheless, by abuse of notation, when we write $r[[r']]_{\bar{p}}$ we sometimes consider it as the action of constructing a pumping by assumming that r, r' and $\bar{p}$ are still explicit.

While the definition above preserves satisfaction of equalities in the constraints, nothing is guaranteed for disequalities. The following basic lemma is independent from the definition of pumping, but it will be very useful to reason about pumpings. It intuitively says that, when a concrete disequality is falsified as a consequence of a multiple replacement of occurrences of one term by another new term, the new term is uniquely determined.

Lemma 5.2 *Let s and t be terms in $\mathcal{T}(\Sigma)$ such that $s \neq t$. Let $P = \{p_1, \ldots, p_n\}$ be a set of positions in* $\mathtt{Pos}(s)$*. Let $P' = \{p'_1, \ldots, p'_k\}$ be a set of positions in* $\mathtt{Pos}(t)$*. Suppose that $s|_{p_1} = \ldots = s|_{p_n} = t|_{p'_1} = \ldots = t|_{p'_k}$ holds.*

Then, there exists at most one term u satisfying $s[u]_{p_1} \ldots [u]_{p_n} = t[u]_{p'_1} \ldots [u]_{p'_k}$

Proof. We prove it by induction on $|s| + |t|$, and distinguishing cases depending on whether some p_i or p'_i is λ or not.

If some p_i or p'_i is λ, say p_1, then n is 1 and since $s \neq t$ and $s|_{p_1}$ coincides with all $t|_{p'_j}$, it follows that no p'_j is λ. Therefore, either k is 0 and hence

$t[u]_{p'_1}\ldots[u]_{p'_k}$ is t for any term u, or k is not 0 and hence u is a proper subterm of $t[u]_{p'_1}\ldots[u]_{p'_k}$ for any term u. In the first case, only $u=t$ makes the equality $s[u]_{p_1}\ldots[u]_{p_n}=u=t=t[u]_{p'_1}\ldots[u]_{p'_k}$ true, and we are done. In the second case, no u satisfies the equality $s[u]_{p_1}\ldots[u]_{p_n}=t[u]_{p'_1}\ldots[u]_{p'_k}$, and we are done.

Otherwise, suppose that none of the p_i's and p'_i's is λ. Since s and t are different, either $\texttt{root}(s)\neq\texttt{root}(t)$, or s and t are of the form $f(s_1,\ldots,s_m), f(t_1,\ldots,t_m)$, respectively, and there exists some i in $\{1,\ldots,m\}$ such that s_i and t_i are different. In the first case it is obvious that no u satisfies $s[u]_{p_1}\ldots[u]_{p_n}=t[u]_{p'_1}\ldots[u]_{p'_k}$, and we are done. Thus, assume that the second case holds for a certain i, and let P_i and P'_i be $\{p|i.p\in P\}$ and $\{p|i.p\in P'\}$, respectively. The terms s_i, t_i, and sets of positions P_i and P'_i satisfy the conditions of the lemma, and $|s_i|+|t_i|<|s|+|t|$. Hence, induction hypothesis apply for them, and it follows also for s and t that there exists at most one term u such that $s[u]_{p_1}\ldots[u]_{p_n}=t[u]_{p'_1}\ldots[u]_{p'_k}$ holds. □

The following lemma connects with the previous one because it argues that a pumping is just a multiple replacement of occurrences of one term by another new term.

Lemma 5.3 *Let $A=\langle Q,\Sigma,F,\Delta\rangle$ be a $TA_{\neq,hom}$. Let r be a run of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A.*

Then, there exist parallel positions $\bar{p}_1,\ldots,\bar{p}_k$ including $\bar{p}$ such that, for all run r' satisfying $\texttt{root}(r')=\texttt{root}(r|_{\bar{p}})$, $r|_{\bar{p}_1}=\ldots=r|_{\bar{p}_k}$ and $r[[r']]_{\bar{p}}=r[r']_{\bar{p}_1}\ldots[r']_{\bar{p}_k}$ hold.

Moreover, if a prefix p of $\bar{p}$ satisfies that $r|_p$ is a run and for some i in $\{1,\ldots,k\}$, p is a prefix of $\bar{p}_i$ and $|\bar{p}_i-p|\leq h(A)$, then, $|\{p'|p\leq p'<\bar{p}\wedge r|_{p'}$ is a run$\}|$ is bounded by $h(A)$.

Proof. It is easy to verify that the set of positions $\texttt{reppos}(r,\bar{p})$, defined recursively as follows for a given run r and a position $\bar{p}$ under the above assumptions, satisfies the statement. $\texttt{reppos}(r,\lambda)$ is defined as $\{\lambda\}$, and $\texttt{reppos}((I:s\xrightarrow{c}q)(s_1,\ldots,s_m)[r_1]_{\hat{p}_1}\ldots[r_n]_{\hat{p}_n},\hat{p}_i.\tilde{p})$ is defined as $\{\hat{p}_j.\tilde{p}'|(j=i\vee(\hat{p}_i=\hat{p}_j)\in c)\wedge\tilde{p}'\in\texttt{reppos}(r_i,\tilde{p})\}$ □

When a pumping does not produce a run, there must exists at least one falsified disequality, and one of them must be a prefix of the position $\bar{p}$ of the pumping.

Lemma 5.4 *Let $A=\langle Q,\Sigma,F,\Delta\rangle$ be a $TA_{\neq,hom}$. Let r,r' be runs of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A and $\texttt{root}(r|_{\bar{p}})=\texttt{root}(r')$ holds. Suppose that $r[[r']]_{\bar{p}}$ is not a run.*

Then, there exists a position p such that $\texttt{root}((r[[r']]_{\bar{p}})|_p)$ is of the form $(I:s\xrightarrow{c}q)$, and there exists $(p_1\neq p_2)$ in c satisfying $\pi_\Sigma(r[[r']]_{\bar{p}})|_{p.p_1}=\pi_\Sigma(r[[r']]_{\bar{p}})|_{p.p_2}$. Moreover, p can be chosen to be a prefix of $\bar{p}$.

Proof. We prove it by induction on $\texttt{height}(r)$. If $\bar{p}$ is λ, then, $r[[r']]_{\bar{p}}$ is r', which is a run, thus contradicting the statement. Hence, without loss of generality, r is of the form $(I:s\xrightarrow{c}q)(s_1,\ldots,s_m)[r_1]_{\hat{p}_1}\ldots[r_n]_{\hat{p}_n}$ and $\bar{p}$ is of the form $p_1.\tilde{p}$.

If $r_1[[r']]_{\tilde{p}}$ is not a run, then, by induction hypothesis, there exists a prefix p' of $\tilde{p}$ such that $\mathtt{root}((r_1[[r']]_{\tilde{p}})|_{p'})$ is of the form $(I' : s' \xrightarrow{c'} q')$, and there exists $(p_1 \neq p_2)$ in c' satisfying $\pi_\Sigma(r_1[[r']]_{\tilde{p}})|_{p'.p_1} = \pi_\Sigma(r_1[[r']]_{\tilde{p}})|_{p'.p_2}$. By defining p as $p_1.p'$ the lemma follows.

Otherwise, assume that $r_1[[r']]_{\tilde{p}}$ is a run. Since $r[[r']]_{\bar{p}}$ is not a run, this can only be due to the existence of a disequality $(p_1 \neq p_2)$ in c satisfying $\pi_\Sigma(r[[r']]_{\bar{p}})|_{p_1} = \pi_\Sigma(r[[r']]_{\bar{p}})|_{p_2}$. By defining p as λ the lemma follows. ☐

Definition 5.5 *(replaced positions and falsified disequalities) Let* $A = \langle Q, \Sigma, F, \Delta \rangle$ *be a* $TA_{\neq,hom}$*. Let r be a run of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A.*

According to Lemma 5.3, there exist parallel positions $\bar{p}_1 = \bar{p}, \ldots, \bar{p}_k$ such that $r[[r']]_{\bar{p}} = r[r']_{\bar{p}_1} \ldots [r']_{\bar{p}_k}$ for any run r' satisfying $\mathtt{root}(r|_{\bar{p}}) = \mathtt{root}(r')$. We call the replaced positions *of any of such pumpings $r[[r']]_{\bar{p}}$ to these positions. We will usually denote them with a bar ($\bar{p}_i$), or with a tilde ($\tilde{p}_i$).*

Let r' be a run such that $\mathtt{root}(r|_{\bar{p}}) = \mathtt{root}(r')$ but $r[[r']]_{\bar{p}}$ is not a run. According to Lemma 5.4, there exists a position p such that $\mathtt{root}((r[[r']]_{\bar{p}})|_p)$ is of the form $(I : s \xrightarrow{c} q)$, and there exists $(p_1 \neq p_2)$ in c satisfying $\pi_\Sigma(r[[r']]_{\bar{p}})|_{p.p_1} = \pi_\Sigma(r[[r']]_{\bar{p}})|_{p.p_2}$. In such a case, we say that the pumping $r[[r']]_{\bar{p}}$ falsifies *$(p_1 \neq p_2)$ at p. Moreover, if some $\bar{p}_i$ is a proper prefix of $p.p_1$ ($p.p_2$) we say that the pumping $r[[r']]_{\bar{p}}$* close-falsifies *$(p_1 \neq p_2)$ at $\langle p, \bar{p}_i, p.p_1 \rangle$ (at $\langle p, \bar{p}_i, p.p_2 \rangle$). Note that, it may happen that $r[[r']]_{\bar{p}}$ close-falsifies $(p_1 \neq p_2)$ at $\langle p, \bar{p}_i, p.p_1 \rangle$ and at $\langle p, \bar{p}_i, p.p_2 \rangle$. In the case where $r[[r']]_{\bar{p}}$ falsifies $(p_1 \neq p_2)$ at p but no $\bar{p}_i$ is a proper prefix of p_1 or of p_2, we say that $r[[r']]_{\bar{p}}$* far-falsifies *$(p_1 \neq p_2)$ at p.*

Example 5.6 *(Continues Example 4.18) Let r_p be a run of $\hat{A}$, defined as $r_p = \hat{\rho}_2(\hat{\rho}_1)$. Then, the pumping $\hat{r}[[r_p]]_{1.1}$ is the run $\hat{r} = \hat{\rho}_4(f(\hat{\rho}_2(a), \hat{\rho}_2(a), \hat{\rho}_2(a)))$, which far-falsifies the disequalities $1.1 \neq 1.3$ and $1.2 \neq 1.3$ at position λ.*

In order to give an example of a close-falsified disequality we define another example. Let $A_{\mathtt{ex2}} = \langle Q, \Sigma, F, \Delta \rangle$ be a $TA_{\neq,hom}$ where $Q = F = \{q\}$ holds, $\Sigma = \{f^{(2)}, h^{(1)}, a^{(0)}\}$ holds, and Δ is the following conjunction of rules:

- $\rho_1 = a \rightarrow q$
- $\rho_2 = h(q) \xrightarrow{1.1 \neq 1.2} q$
- $\rho_3 = f(q, q) \xrightarrow{1=2} q$

Let $r_{\mathtt{ex2}}$ be the following run of $A_{\mathtt{ex2}}$: $\rho_3(\rho_2(\rho_2(\rho_1)), \rho_2(\rho_2(\rho_1)))$, with projection $\pi_\Sigma(r_{\mathtt{ex2}}) = f(h(h(a)), h(h(a)))$, and let r_p be $\rho_3(\rho_1, \rho_1)$. Then, the pumping $r_{\mathtt{ex2}}[[r_p]]_{1.1}$ is: $\rho_3(\rho_2(\rho_3(\rho_1, \rho_1)), \rho_2(\rho_3(\rho_1, \rho_1)))$ which is not a run, because it close-falsifies $(1.1 \neq 1.2)$ at $\langle 1, 1.1, 1.1.1 \rangle$ (or at $\langle 1, 1.1, 1.1.2 \rangle$), and it close-falsifies $(1.1 \neq 1.2)$ at $\langle 2, 2.1, 2.1.1 \rangle$ (or at $\langle 2, 2.1, 2.1.2 \rangle$).

The following technical definitions and lemmas are used to argue for the existence of certain pumpings when a term is big enough. These proofs are developed in the following two subsections.

Lemma 5.7 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r, r' be runs of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A and $\texttt{root}(r|_{\bar{p}}) = \texttt{root}(r')$ holds.*

If $r[[r']]_{\bar{p}}$ close-falsifies some disequality, then $r[[r']]_{\bar{p}}$ close-falsifies a disequality $(p_1 \neq p_2)$ at a tuple $\langle p, \bar{p}_i, p.p_1\rangle$ where p is a prefix of $\bar{p}$.

Proof. We prove it by induction on $\texttt{height}(r)$. If $\bar{p}$ is λ, then, $r[[r']]_{\bar{p}}$ is r', which is a run, thus contradicting the statement. Hence, without loss of generality, assume that r is of the form $(I : s \xrightarrow{c} q)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$ and $\bar{p}$ is of the form $\hat{p}_1.\tilde{p}$.

If $r_1[[r']]_{\tilde{p}}$ close-falsifies some disequality, then, by induction hypothesis, $r_1[[r']]_{\tilde{p}}$ close-falsifies a disequality $(p_1 \neq p_2)$ at a tuple $\langle p', \tilde{p}_i, p'.p_1\rangle$ where p' is a prefix of $\tilde{p}$. By defining p as $\hat{p}_1.p'$ and $\bar{p}_i$ as $\hat{p}_1.\tilde{p}_i$ the lemma follows.

Otherwise, assume that $r_1[[r']]_{\tilde{p}}$ does not close-falsify any disequality. Since $r[[r']]_{\bar{p}}$ close-falsifies some disequality, it must be a disequality $(p_1 \neq p_2)$ close-falsified at a tuple of the form $\langle \lambda, \bar{p}_i, \lambda.p_1\rangle$. By defining p as λ the lemma follows. □

Definition 5.8 *(h-similar terms) Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. We denote as $h(A)$ to the maximum between the maximum length of positions occurring in the constraints of the rules of Δ, and the maximum height of a left-hand side of a rule in Δ. We write h when A is clear from the context.*

Let s and t be terms in $\mathcal{T}(\Sigma)$. By $s =_h t$ we denote that the following conditions hold:

- *For each position p with $|p| \leq h$, $(p \in \texttt{Pos}(s) \Leftrightarrow p \in \texttt{Pos}(t))$ holds.*
- *For each $p \in \texttt{Pos}(s)$ with $|p| \leq h$, $(\texttt{root}(s|_p) = \texttt{root}(t|_p))$ holds.*
- *For each positions $p_1, p_2 \in \texttt{Pos}(s)$ with $|p_1|, |p_2| \leq h$, $((s|_{p_1} = s|_{p_2}) \Leftrightarrow (t|_{p_1} = t|_{p_2}))$ holds.*

Note that $=_h$ is an equivalence relation with a finite number of classes depending on Δ. By $\mathcal{H}(A)$, or $\mathcal{H}$ when A is clear from the context, we denote such a number of classes.

Lemma 5.9 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r, r_1, r_2 be runs of A, and let $\bar{p}$ be a position such that $r|_{\bar{p}}$ is a run of A and $\texttt{root}(r|_{\bar{p}}) = \texttt{root}(r_1) = \texttt{root}(r_2)$. Suppose that $r[[r_1]]_{\bar{p}}$ and $r[[r_2]]_{\bar{p}}$ are not runs. Let $\bar{p}_1, \ldots, \bar{p}_k$ be the replaced positions of both $r[[r_1]]_{\bar{p}}$ and $r[[r_2]]_{\bar{p}}$.*

- *If both $r[[r_1]]_{\bar{p}}$ and $r[[r_2]]_{\bar{p}}$ far-falsify the same $(p_1 \neq p_2)$ at the same p, then, $\pi_\Sigma(r_1) = \pi_\Sigma(r_2)$.*
- *If $\pi_\Sigma(r|_{\bar{p}}) =_h \pi_\Sigma(r_1) =_h \pi_\Sigma(r_2)$ and both $r[[r_1]]_{\bar{p}}$ and $r[[r_2]]_{\bar{p}}$ close-falsify the same $(p_1 \neq p_2)$ at the same $\langle p, \bar{p}_i, p.p_1\rangle$, then $\pi_\Sigma(r)|_{\bar{p}.((p.p_1)-\bar{p}_i)} \neq \pi_\Sigma(r_1)|_{(p.p_1)-\bar{p}_i} = \pi_\Sigma(r_2)|_{(p.p_1)-\bar{p}_i}$.*

Proof. The first item is a direct consequence of previous definitions and Lemma 5.2.

For the second item, we first prove that no position in $\{\bar{p}_1, \ldots, \bar{p}_k\}$ can be a proper prefix of $p.p_2$ by contradiction. Thus, assume that, for some position in $\{\bar{p}_1, \ldots, \bar{p}_k\}$, say $\bar{p}_j$, $\bar{p}_j$ is a proper prefix of $p.p_2$. Note that $r|_{\bar{p}} = r|_{\bar{p}_i} = r|_{\bar{p}_j}$. Thus, $\pi_\Sigma(r|_{\bar{p}})|_{(p.p_1)-\bar{p}_i} = \pi_\Sigma(r)|_{\bar{p}.((p.p_1)-\bar{p}_i)} = \pi_\Sigma(r)|_{\bar{p}_i.((p.p_1)-\bar{p}_i)} = \pi_\Sigma(r)|_{p.p_1} \neq \pi_\Sigma(r)|_{p.p_2} = \pi_\Sigma(r)|_{\bar{p}_j.((p.p_2)-\bar{p}_j)} = \pi_\Sigma(r)|_{\bar{p}.((p.p_2)-\bar{p}_j)} = \pi_\Sigma(r|_{\bar{p}})|_{(p.p_2)-\bar{p}_j}$ holds, and since $\pi_\Sigma(r|_{\bar{p}}) =_h \pi_\Sigma(r_1)$, then $\pi_\Sigma(r_1)|_{(p.p_1)-\bar{p}_i} \neq \pi_\Sigma(r_1)|_{(p.p_2)-\bar{p}_j}$ holds. But this is in contradiction with the fact that $r[[r_1]]_{\bar{p}}$ close-falsifies $(p_1 \neq p_2)$ at $\langle p, \bar{p}_i, p.p_1\rangle$.

Second, we show that $p.p_2$ is not a prefix of any position in $\{\bar{p}_1, \ldots, \bar{p}_k\}$, again by contradiction. Thus, assume that, for some position in $\{\bar{p}_1, \ldots, \bar{p}_k\}$, say $\bar{p}_j$, $p.p_2$ is a prefix of $\bar{p}_j$. Then, r_1 is a subterm of $r[[r_1]]_{\bar{p}}|_{p.p_2}$. On the other side, since $\bar{p}_i$ is a proper prefix of $p.p_1$, it holds that $r[[r_1]]_{\bar{p}}|_{p.p_1}$ is a proper subterm of r_1. Thus, $r[[r_1]]_{\bar{p}}|_{p.p_1} = r[[r_1]]_{\bar{p}}|_{p.p_2}$ is not possible, contradicting the fact that $r[[r_1]]_{\bar{p}}$ close-falsifies $(p_1 \neq p_2)$ at $\langle p, \bar{p}_i, p.p_1\rangle$.

From the two above previous facts, we conclude that $p.p_2$ is parallel with all replaced positions $\bar{p}_1, \ldots, \bar{p}_n$. Thus, $r|_{p.p_2} = r[[r_1]]_{\bar{p}}|_{p.p_2} = r[[r_2]]_{\bar{p}}|_{p.p_2}$ holds. Moreover, recall that $\pi_\Sigma(r)|_{\bar{p}.((p.p_1)-\bar{p}_i)} = \pi_\Sigma(r)|_{p.p_1} \neq \pi_\Sigma(r)|_{p.p_2} = \pi_\Sigma(r|_{p.p_2})$ holds, and note that $\pi_\Sigma(r_1)|_{(p.p_1)-\bar{p}_i} = \pi_\Sigma(r[[r_1]]_{\bar{p}})|_{p.p_1} = \pi_\Sigma(r[[r_1]]_{\bar{p}})|_{p.p_2} = \pi_\Sigma(r[[r_1]]_{\bar{p}}|_{p.p_2})$, and $\pi_\Sigma(r_2)|_{(p.p_1)-\bar{p}_i} = \pi_\Sigma(r[[r_2]]_{\bar{p}})|_{p.p_1} = \pi_\Sigma(r[[r_2]]_{\bar{p}})|_{p.p_2} = \pi_\Sigma(r[[r_2]]_{\bar{p}}|_{p.p_2})$ hold. Therefore, $\pi_\Sigma(r)|_{\bar{p}.((p.p_1)-\bar{p}_i)} \neq \pi_\Sigma(r_1)|_{(p.p_1)-\bar{p}_i} = \pi_\Sigma(r_2)|_{(p.p_1)-\bar{p}_i}$ also hold, and we are done. □

5.1 decreasing pumpings

When a term is big enough, it can be argued the existence of a pumping decreasing the size and producing a correct run. This allows to prove decidability of emptiness. The proof of this fact follows the ideas presented in [8].

In the following bounds, let N_Δ be the number of different disequalities in the rules of Δ. We write N when Δ is clear from the context.

Lemma 5.10 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let s be a term of minimal size among all the terms accepted by A. Let M be $\mathcal{H} \cdot |\Delta| \cdot (2Nh)^{(2Nh+1)}$. Then,* $\texttt{height}(s) \leq h\big[((N+1)^M - 1)(N+1)/N\big]^M$.

Proof. Let r be a run of A with an accepting resulting state and such that $\pi_\Sigma(r) = s$. Let $\check{p}$ be a position of r. In order to conclude, it suffices to bound $|\check{p}|$ by $h\big[((N+1)^M - 1)(N+1)/N\big]^M$.

We will apply a conceptual process dealing with a data structure of the form $\langle S, \{\langle E_1, P_1\rangle, \ldots, \langle E_k, P_k\rangle\}\rangle$, where all $S, E_1, P_1, \ldots, E_k, P_k$ are sets of positions satisfying the following invariants:

- $S \cup E_1 \cup \ldots \cup E_k$ is the set of all positions p such that $p \leq \check{p}$ holds and $r|_p$ is a run.
- The sets $S, E_1, \ldots, E_k$ are pairwise disjoint.

- For each i in $\{1, \ldots, k\}$ and each two positions $\bar{p}, p \in E_i$, $s|_{\bar{p}} =_h s|_p$ and $\texttt{root}(r|_{\bar{p}}) = \texttt{root}(r|_p)$ hold.
- $P_1, \ldots, P_k$ contain positions that are suffixes of positions occurring in the constraint of some rule in Δ.
- For each i in $1, \ldots, k$, and each $\bar{p}, p \in E_i$, and each $\breve{p}$ in P_i, $s|_{\bar{p}.\breve{p}} = s|_{p.\breve{p}}$ holds.

Starting the process: The first tuple of our process will be $\langle \emptyset, \{\langle E_1, \emptyset\rangle, \ldots, \langle E_k, \emptyset\rangle\}\rangle$, where $\{E_1, \ldots, E_k\}$ is the partition satisfying that $E_1 \cup \ldots \cup E_k$ is the set of all positions p such that $p \leq \breve{p}$ holds, $r|_p$ is a run, and two positions $\bar{p}, p$ are in the same E_i if and only if $(s|_{\bar{p}} =_h s|_p \wedge \texttt{root}(r|_{\bar{p}}) = \texttt{root}(r|_p))$ holds. It is clear that the first tuple satisfies the invariant.

A step of the process: Now, at each step of the process with a current tuple $\langle S, \{\langle E_1, P_1\rangle, \ldots, \langle E_k, P_k\rangle\}\rangle$, it is chosen the minimum position $\bar{p}$ in $E_1 \cup \ldots \cup E_k$. Without loss of generality, suppose that $\bar{p}$ is in E_1. Note that $\bar{p}$ is a proper prefix of all positions p in $E_1 - \{\bar{p}\}$, and that $r|_{\bar{p}}$ and all $r|_p$ for each of such p are runs. Moreover, by the minimality of s, no $r[[r|_p]]_{\bar{p}}$ for each of such p is a run. By Lemma 5.4, each of such $r[[r|_p]]_{\bar{p}}$ falsifies some disequality $(p_1 \neq p_2)$ at some p'. Moreover, by Lemma 5.7, in the case where $r[[r|_p]]_{\bar{p}}$ close-falsifies a disequality $(p_1 \neq p_2)$ at some $\langle p', \bar{p}', p'.p_1\rangle$, this p' can be chosen to be a prefix of $\bar{p}$. Now, the process considers the set S' of all positions p in $E_1 - \{\bar{p}\}$ such that $r[[r|_p]]_{\bar{p}}$ far-falsifies some disequality. Note that, by the minimality of $\bar{p}$, on the one side $|\bar{p}| \leq |S| + 1$, and on the other side, by the first item of Lemma 5.9, $|S'| \leq N|\bar{p}|$. Thus, $|S'|$ is bounded by $N(|S| + 1)$. Now note that, for each p in $E_1 - (\{\bar{p}\} \cup S')$, $r[[r|_p]]_{\bar{p}}$ close-falsifies some disequality $(p_1 \neq p_2)$ at some $\langle p', \bar{p}', p'.p_1\rangle$. Moreover, by Lemma 5.7, for each of such p we can assume that the corresponding p' is a prefix of $\bar{p}$. Note also that, by the last part of the statement in Lemma 5.3, such a p' can be chosen only among $h(A)$ possibilities. The process constructs a partition $\{E'_1, \ldots, E'_n\}$ of $E_1 - (\{\bar{p}\} \cup S')$ and satisfying the following condition: for each i in $\{1, \ldots, n\}$, there exists a position $\breve{p}_i$, a disequality $(p_1 \neq p_2)$, and a tuple $\langle p', \bar{p}', p'.p_1\rangle$, such that, for each position p in E_i, the pumping $r[[r|_p]]_{\bar{p}}$ close-falsifies $(p_1 \neq p_2)$ at $\langle p', \bar{p}', p'.p_1\rangle$, p' is a prefix of $\bar{p}$ and $p'.p_1 - \bar{p}'$ equals $\breve{p}_i$ (note that there could be several different elections for the partition $\{E'_1, \ldots, E'_n\}$ if some $r[[r|_p]]_{\bar{p}}$ close-falsifies several disequalities). For each of such i's, disequalities $(p_1 \neq p_2)$, and tuples $\langle p', \bar{p}', p'.p_1\rangle$, by the second item of Lemma 5.9, $\pi_\Sigma(r)|_{\bar{p}.((p'.p_1)-\bar{p}')}$ is different from $\pi_\Sigma(r|_p)|_{(p'.p_1)-\bar{p}'}$ for each p in E'_i, and all of such $\pi_\Sigma(r|_p)|_{(p'.p_1)-\bar{p}'}$ are identical. It follows $\pi_\Sigma(r)|_{\bar{p}.\breve{p}_i} \neq \pi_\Sigma(r)|_{p.\breve{p}_i}$ for each p in E'_i, and all $\pi_\Sigma(r)|_{p.\breve{p}_i}$ are identical for all p in E'_i. Thus, by the invariants of the process, $\breve{p}_i$ is not in P_1 (note that this consequence is valid for $\breve{p}_1, \ldots, \breve{p}_n$). The tuple constructed for the next step is $\langle S \cup \{\bar{p}\} \cup S', \{\langle E'_1, P_1 \cup \{\breve{p}_1\}\rangle, \ldots, \langle E'_n, P_1 \cup \{\breve{p}_n\}\rangle, \langle E_2, P_2\rangle, \ldots, \langle E_k, P_k\rangle\}\rangle$. From the above observations it follows that the new tuple satisfies the invariants, too.

Recall that, at each step, $\langle E_1, P_1\rangle$ is removed and new $\langle E'_1, P_1 \cup \{\breve{p}_1\}\rangle, \ldots, \langle E'_n, P_1 \cup \{\breve{p}_n\}$ are added, where each of such $P_1 \cup \{\breve{p}_i\}$ satisfy that $\breve{p}_i$ is not in P_1. Also recall that, by the invariants, the sets P_j contain suffixes of positions occurring at disequalities in the constraints of rules in Δ. There are at most $2Nh$ different suffixes of this kind. Moreover, by the last part of the statement in Lemma 5.3, the triples $\langle p', \bar{p}', p'.p_1 \rangle$ are constructed by choosing p' among h possibilities, and p_1 among $2N$ possibilities, from which $\bar{p}'$ is uniquely determined. Hence, there are at most $2Nh$ possible triples. Thus, when $\langle E_1, P_1 \rangle$ is replaced by $\langle E_1', P_1 \cup \{\check{p}_1\}\rangle, \ldots, \langle E_n', P_1 \cup \{\check{p}_n\}\rangle$, such an n is bounded by $2Nh$. Therefore, the number of execution steps of the process is bounded by $M = \mathcal{H} \cdot |\Delta| \cdot (2Nh)^{(2Nh+1)}$. It follows that the process terminates, and S contains all prefixes p' of p such that $r|_{p'}$ is a run when it halts. Let S_i represent the set S at the i'th execution step. By the above remarks $|S_{i+1}| \leq |S_i| + N(|S_i| + 1) + 1$. By solving this serie for $|S_0| = 0$ we conclude $|S_i| \leq ((N+1)^i - 1)(N+1)/N$, and hence, the number of prefixes p' of $|p|$ such that $r|_{p'}$ is a run is bounded by $|S_M| \leq \left[((N+1)^M - 1)(N+1)/N\right]^M$, and this concludes the proof since, as a consequence, h times this number bounds $|p|$. □

Corollary 5.11 *The emptiness problem is decidable for $TA_{\neq,hom}$.*

5.2 Increasing pumpings

When a term is big enough, it can be argued the existence of infinite pumpings increasing the size and producing correct runs. This allows to prove decidability of finiteness, but also, it will be a key point for the decidability of the HOM problem.

Definition 5.12 *Let A be a $TA_{\neq,hom}$. We define $\check{h}(A)$ as $\mathcal{H} \cdot |\Delta| \cdot (h(A)^2 + h(A)) \cdot ((2Nh(A)+1)^{(2Nh(A)+1)})$.*

The key lemma of this subsection is Lemma 5.19. For its proof, we need to develop several intermediate technical results related to falsified disequalities.

Lemma 5.13 *Let $A = \langle Q, \Sigma, F, \Delta \rangle$ be a $TA_{\neq,hom}$. Let r be a run of A satisfying $\texttt{height}(r) > \check{h}(A)$. Let $\check{p}$ be a position in $\texttt{Pos}(r)$ satisfying $|\check{p}| = \texttt{height}(r)$. Then, there are two positions $p, \bar{p}$ of $\texttt{Pos}(r)$ satisfying the following conditions:*

- *$p < \bar{p} \leq \check{p}$ and $|p| + h(A)^2 < |\bar{p}|$.*
- *$r|_p$ and $r|_{\bar{p}}$ are runs such that $\texttt{root}(r|_p) = \texttt{root}(r|_{\bar{p}})$.*
- *$r[[r|_p]]_{\bar{p}}$ does not close-falsify any disequality.*

Proof. We proceed by contradiction by assuming that such two positions do not exist. Thus, for any two positions $p, \bar{p}$ satisfying $p < \bar{p} \leq \check{p}$, $|p| + h^2 < |\bar{p}|$, $r|_p$ and $r|_{\bar{p}}$ are runs, and $\texttt{root}(r|_p) = \texttt{root}(r|_{\bar{p}})$, it holds that the pumping $r[[r|_p]]_{\bar{p}}$ close-falsifies some disequality.

Since $|\check{p}| = \texttt{height}(r)$ holds, in particular, $|\check{p}| > \check{h}(A)$ holds. From the prefixes of $\check{p}$ we can choose a set E with $|E| \geq \check{h}(A)/(\mathcal{H} \cdot |\Delta| \cdot (h^2 + h)) = (2Nh+1)^{(2Nh+1)}$ and satisfying the following conditions.

- For each position p in E, $r|_p$ is a run.
- For each two positions $p, \bar{p}$ in E, $\pi_\Sigma(r|_p) =_h \pi_\Sigma(r|_{\bar{p}})$ and $\texttt{root}(r|_p) = \texttt{root}(r|_{\bar{p}})$ hold.
- For each two positions $p, \bar{p}$ in E, satisfying $p < \bar{p}$, $|p| + h^2 < |\bar{p}|$ holds.

By our assumptions, for any two positions $p < \bar{p}$ in E, the pumping $r[[r|_p]]_{\bar{p}}$ close-falsifies some disequality.

We will proceed by modifying E and a set of positions P as follows by preserving the above conditions, and also by preserving the following invariant: P contains suffixes of positions occurring at disequalities in the constraints of rules in Δ, and for each position $\breve{p}$ in P and each two positions $p < \bar{p}$ in E, $\pi_\Sigma(r|_{p.\breve{p}}) = \pi_\Sigma(r|_{\bar{p}.\breve{p}})$ holds.

Initially, P is $\emptyset$ and E is defined as above. At each step, we consider the maximum position $\bar{p}$ of E in size. Note that, for each position $p < \bar{p}$ in E the pumping $r[[r|_p]]_{\bar{p}}$ close-falsifies some disequality $(p_1 \neq p_2)$ at some $\langle p', \bar{p}', p'.p_1\rangle$. Moreover, by Lemma 5.7, for each of such p we can assume that the corresponding p' is a prefix of $\bar{p}$. Note also that, by the last part of the statement in Lemma 5.3, such a p' can be chosen only among h possibilities.

Let $\{E_1, \ldots, E_n\}$ be a partition of $E - \{\bar{p}\}$ satisfying the following condition: for each i in $\{1, \ldots, n\}$, there exists a position $\breve{p}_i$, a disequality $(p_1 \neq p_2)$, and a tuple $\langle p', \bar{p}', p'.p_1\rangle$, such that, for each position p in E_i, the pumping $r[[r|_p]]_{\bar{p}}$ close-falsifies $(p_1 \neq p_2)$ at $\langle p', \bar{p}', p'.p_1\rangle$, p' is a prefix of $\bar{p}$ and $p'.p_1 - \bar{p}'$ equals $\breve{p}_i$ (note that there could be several different elections for the partition $\{E_1, \ldots, E_n\}$ if some $r[[r|_p]]_{\bar{p}}$ close-falsifies several disequalities). For each of such i's, disequalities $(p_1 \neq p_2)$, and tuples $\langle p', \bar{p}', p'.p_1\rangle$, by the second item of Lemma 5.9, $\pi_\Sigma(r)|_{\bar{p}.((p'.p_1) - \bar{p}')}$ is different from $\pi_\Sigma(r|_p)|_{(p'.p_1) - \bar{p}'}$ for each p in E_i, and all of such $\pi_\Sigma(r|_p)|_{(p'.p_1) - \bar{p}'}$ are identical. It follows $\pi_\Sigma(r)|_{\bar{p}.\breve{p}_i} \neq \pi_\Sigma(r)|_{p.\breve{p}_i}$ for each p in E_i, and all $\pi_\Sigma(r)|_{p.\breve{p}_i}$ are identical for all p in E_i. Thus, by the invariant, $\breve{p}_i$ is not in P (note that this consequence is valid for $\breve{p}_1, \ldots, \breve{p}_n$). For the next step, we choose E as the E_i with maximum cardinality $|E_i|$, and choose P as $P \cup \{\breve{p}_i\}$. From the above observations it follows that the new E, P satisfy the invariants, too.

Note that, at each step, P increases its cardinality by 1. Also recall that the set P contains suffixes of positions occurring at disequalities in the constraints of rules in Δ. There are at most $2Nh$ different suffixes of this kind. It follows that the number of execution steps is bounded by $2Nh$. Moreover, by the last part of the statement in Lemma 5.3, the triples $\langle p', \bar{p}', p'.p_1\rangle$ are constructed by choosing p' among h possibilities, and p_1 among $2N$ possibilities, from which $\bar{p}'$ is uniquely determined. Hence, there are at most $2Nh$ possible triples. Thus, the partition $E_1, \ldots, E_n$ has at most $2Nh$ parts. Hence, the cardinal of E is substracted by 1 and then divided by at most $2Nh$ at each execution step, i.e. $|E_{j+1}| \geq \lceil (|E_j| - 1)/(2Nh) \rceil$. Note that, according to the assumptions, E must be empty at the last execution step. Thus, the starting E satisfies $|E| < ((2Nh+1)^{(2Nh+1)})$, and this is in contradiction with $|E| \geq (2Nh+1)^{(2Nh+1)}$. $\square$

Lemma 5.14 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r, r' be runs of A, and let $\bar{p}$ be a position such that $\mathtt{root}(r|_{\bar{p}}) = \mathtt{root}(r')$. Let p_1, p_2 be two positions such that $r|_{p_1}$ is a run, $r|_{p_1} \neq r|_{p_2}$ holds, and both p_1, p_2 are prefixes of replaced positions in $r[[r']]_{\bar{p}}$. Then, either $r[[r']]_{\bar{p}}|_{p_1}$ is a proper subterm of $r[[r']]_{\bar{p}}|_{p_2}$, or $r[[r']]_{\bar{p}}|_{p_2}$ is a proper subterm of $r[[r']]_{\bar{p}}|_{p_1}$.*

Proof. We prove it by induction on $|\bar{p}|$. We write r more explicitly as $(I : s \xrightarrow{c} q)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$. If p_1 or p_2 is λ, the result follows trivially. Thus, assume that none of them is λ. If p_2 is a position in $\mathtt{Pos}_\Sigma(s)$, then, since both p_1 and p_2 are prefixes of replaced positions and $r|_{p_1}$ is a run, it follows that p_1 is of the form $\hat{p}_i.p_1'$, and p_2 is a proper prefix of a certain $\hat{p}_j$ satisfying that $(\hat{p}_i = \hat{p}_j)$ occurs in c. The result follows trivially by the definition of pumping. Thus, assume that p_2 is not in $\mathtt{Pos}_\Sigma(s)$. In this case, p_1 is of the form $\hat{p}_i.p_1'$, and p_2 is of the form $\hat{p}_j.p_2'$ where $(\hat{p}_i = \hat{p}_j)$ occurs in c. Let $\hat{p}_k$ be the prefix of $\bar{p}$ among the positions $\hat{p}_1, \ldots, \hat{p}_n$. Note that either $\hat{p}_i$ is $\hat{p}_k$ or $(\hat{p}_i = \hat{p}_k)$ occurs in c, and either $\hat{p}_j$ is $\hat{p}_k$ or $(\hat{p}_j = \hat{p}_k)$ occurs in c. Now, observe that the runs $r|_{\hat{p}_k}, r'$, the position $\bar{p} - \hat{p}_k$, and the two positions $\hat{p}_k.p_1'$ and $\hat{p}_k.p_2'$ satisfy the assumptions of the lemma and $|\bar{p} - \hat{p}_k| < |\bar{p}|$ holds. Therefore, by induction hypothesis, either $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_1'}$ is a proper subterm of $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_2'}$, or $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_2'}$ is a proper subterm of $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_1'}$. Finally, since either $\hat{p}_i$ is $\hat{p}_k$ or $(\hat{p}_i = \hat{p}_k)$ occurs in c, and either $\hat{p}_j$ is $\hat{p}_k$ or $(\hat{p}_j = \hat{p}_k)$ occurs in c, then $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_1'} = r[[r']]_{\bar{p}}|_{p_1}$ and $(r|_{\hat{p}_k})[[r']]_{\bar{p}-\hat{p}_k}|_{p_2'} = r[[r']]_{\bar{p}}|_{p_2}$ hold, and hence, the result follows. □

Lemma 5.15 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r, r' be runs of A, and let $\bar{p}$ be a position such that $\mathtt{root}(r|_{\bar{p}}) = \mathtt{root}(r')$. Let p be a prefix of $\bar{p}$ such that $\mathtt{height}(r|_p) = |(\bar{p} - p)| + \mathtt{height}(r|_{\bar{p}})$ holds. Let p_1 be a position such that $p.p_1$ is a prefix of a replaced position in $r[[r']]_{\bar{p}}$. Let k be $|\{p' | p \leq p' < p.p_1 \wedge r|_{p'}$ is a run$\}|$.*

Then, $\mathtt{height}(r[[r']]_{\bar{p}}|_{p.p_1}) \geq |\bar{p} - p| + \mathtt{height}(r') - k \cdot h(A)$.

Proof. We prove it by induction on $|p_1|$. If $r|_{p.p_1}$ is not a run, then, since p_1 is a prefix of a replaced position, p_1 can be enlarged to satisfy that $r|_{p.p_1}$ is a run, by preserving the rest of assumptions and the value for k. Thus, assume that $r|_{p.p_1}$ is a run.

Assume that p_1 is λ. Then k is 0, $\mathtt{height}(r[[r']]_{\bar{p}}|_{p.p_1}) = \mathtt{height}(r[[r']]_{\bar{p}}|_p) \geq |(\bar{p} - p)| + \mathtt{height}(r')$ holds, and hence, $\mathtt{height}(r[[r']]_{\bar{p}}|_{p.p_1}) \geq |\bar{p} - p| + \mathtt{height}(r') - k \cdot h$ follows.

Otherwise, assume that p_1 is not λ. We write r more explicitly as $(I : s \xrightarrow{c} q)(s_1, \ldots, s_m)[r_1]_{\hat{p}_1} \ldots [r_n]_{\hat{p}_n}$. Let $\hat{p}_j$ be the prefix of $\bar{p}$ among the positions $\hat{p}_1, \ldots, \hat{p}_n$. Since $r|_{p.p_1}$ is a run, $p.p_1$ is of the form $p.\hat{p}_i.p_1'$ for some $\hat{p}_i$ such that either $\hat{p}_i$ is $\hat{p}_j$, or $(\hat{p}_i = \hat{p}_j)$ occurs in c. Now, note that the runs r, r', the position $\bar{p}$, and the positions $p.\hat{p}_j$ and p_1' satisfy the assumptions of the lemma for $k - 1$, and $|p_1'| < |p_1|$ holds. Thus, by induction hypothesis, $\mathtt{height}(r[[r']]_{\bar{p}}|_{p.\hat{p}_j.p_1'}) \geq |\bar{p} - (p.\hat{p}_j)| + \mathtt{height}(r') - (k - 1) \cdot h$ holds.

Since either $\hat{p}_i$ is $\hat{p}_j$ or $(\hat{p}_i = \hat{p}_j)$ occurs in c, it follows that $r[[r']]_{\bar{p}}|_{p.p_1}$ coincides with $r[[r']]_{\bar{p}}|_{p.p_j.p'_1}$. Since $|\bar{p} - p| \leq |\bar{p} - (p.\hat{p}_j)| + h$ holds, it follows $\texttt{height}(r[[r']]_{\bar{p}}|_{p.p_1}) \geq |\bar{p} - p| + \texttt{height}(r') - k \cdot h$ and we are done. □

Corollary 5.16 *Suppose the hypothesis of the previous lemma and* $\texttt{height}(r') > \texttt{height}(r|_{\bar{p}}) + k \cdot h(A)$*,*

Then, $\texttt{height}(r[[r']]_{\bar{p}}|_{p.p_1}) > \texttt{height}(r|_p)$*.*

Lemma 5.17 *Let* $A = \langle Q, \Sigma, F, \Delta \rangle$ *be a* $TA_{\neq,hom}$*. Let* r *be a run of* A *satisfying* $\texttt{height}(r) > \check{h}(A)$*. Let* $\check{p}$ *be a position in* $\texttt{Pos}(r)$ *satisfying* $|\check{p}| = \texttt{height}(r)$*. Then, there exist two positions* $p, \bar{p}$ *of* $\texttt{Pos}(r)$ *satisfying the following conditions:*

- $p < \bar{p} \leq \check{p}$ *and* $|p| + h(A)^2 < |\bar{p}|$*.*
- $r|_p$ *and* $r|_{\bar{p}}$ *are runs such that* $\texttt{root}(r|_p) = \texttt{root}(r|_{\bar{p}})$*.*
- $r[[r|_p]]_{\bar{p}}$ *is a run.*

Proof. We consider $p, \bar{p}$ to be the two positions given by Lemma 5.13. In order to conclude, it suffices to prove that $r[[r|_p]]_{\bar{p}}$ does not far-falsify any disequality. We proceed by contradiction by assumming that it far-falsifies a disequality $(p_1 \neq p_2)$ at a position p'. Thus, $\pi_\Sigma(r|_{p'.p_1}) \neq \pi_\Sigma(r|_{p'.p_2})$ and $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1}) = \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2})$ hold. By Lemma 5.4, p' can be assumed to be a prefix of $\bar{p}$. Note that, by the definition of far-falsification, no replaced position in $r[[r|_p]]_{\bar{p}}$ is a proper prefix of $p'.p_1$ nor $p'.p_2$. Since $\pi_\Sigma(r|_{p'.p_1}) \neq \pi_\Sigma(r|_{p'.p_2})$ holds, there exists a position p'' satisfying the following conditions:

- $|p''| \leq h$.
- no replaced position in $r[[r|_p]]_{\bar{p}}$ is a proper prefix of $p'.p_1.p''$ nor $p'.p_2.p''$.
- $\pi_\Sigma(r|_{p'.p_1.p''}) \neq \pi_\Sigma(r|_{p'.p_2.p''})$.
- either (i) the roots of $\pi_\Sigma(r|_{p'.p_1.p''})$ and $\pi_\Sigma(r|_{p'.p_2.p''})$ differ, or (ii) some of $r|_{p'.p_1.p''}$ or $r|_{p'.p_2.p''}$ is a run.

We choose p'' to be minimal in size satisfying the above conditions. Note that, since $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1}) = \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2})$ holds, $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) = \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$ also holds.

In case (i), it follows $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) \neq \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$, a contradiction. Thus, assume that case (ii) holds. At this point, we distinguish the following cases:

- Assume that both $p'.p_1.p''$ and $p'.p_2.p''$ are prefixes of replaced positions in $r[[r|_p]]_{\bar{p}}$. Since either $r|_{p'.p_1.p''}$ or $r|_{p'.p_2.p''}$ is a run, by Lemma 5.14, either $r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}$ is a proper subterm of $r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''}$ or $r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''}$ is a proper subterm of $r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}$. In any case, $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) \neq \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$ follows, a contradiction.

- Assume that none of $p'.p_1.p''$ and $p'.p_2.p''$ is a prefix of a replaced position in $r[[r|_p]]_{\bar{p}}$. In this case, $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''})$ coincides with $\pi_\Sigma(r|_{p'.p_1.p''})$, and $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$ coincides with $\pi_\Sigma(r|_{p'.p_2.p''})$. This is in contradiction with $\pi_\Sigma(r|_{p'.p_1.p''}) \neq \pi_\Sigma(r|_{p'.p_2.p''})$ and $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) = \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$.

- Finally, assume that only one of $p'.p_1.p''$ and $p'.p_2.p''$ is a prefix of a replaced position in $r[[r|_p]]_{\bar{p}}$. We assume that it is $p'.p_1.p''$ (the other case reaches a contradiction analogously). Thus, $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''})$ coincides with $\pi_\Sigma(r|_{p'.p_2.p''})$. Note that $\mathtt{height}(r|_{p'.p_2.p''}) < \mathtt{height}(r|_{p'})$. By the minimality election for p'', it holds that $k = |\{p'''|\ p' \leq p''' < p'.p_1.p'' \wedge r|_{p'''} \text{ is a run}\}|$ is smaller than or equal to h. Recall that p' is a prefix of $\bar{p}$, $p'.p_1.p''$ is a prefix of a replaced position in $r[[r|_p]]_{\bar{p}}$, and since $\bar{p}$ is a prefix of $\check{p}$, $\mathtt{height}(r|_{p'}) = |(\bar{p} - p')| + \mathtt{height}(r|_{\bar{p}})$ holds. Recall also that $|\bar{p} - p| > h^2$ holds, and hence, $\mathtt{height}(r|_p) > \mathtt{height}(r|_{\bar{p}}) + h^2$ holds. Thus, by Corollary 5.16, $\mathtt{height}(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) \geq \mathtt{height}(r|_{p'})$ holds. It follows $\mathtt{height}(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) > \mathtt{height}(r|_{p'.p_2.p''}) = \mathtt{height}(\pi_\Sigma(r|_{p'.p_2.p''})) = \mathtt{height}(\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''}))$. Therefore, $\pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_1.p''}) \neq \pi_\Sigma(r[[r|_p]]_{\bar{p}}|_{p'.p_2.p''}))$ holds, a contradiction.

□

Corollary 5.18 *Let $A = \langle Q, \Sigma, F, \Delta\rangle$ be a $TA_{\neq,hom}$. Let r be a run of A satisfying* $\mathtt{height}(r) > \check{h}(A)$. *Let $\check{p}$ be a position in* $\mathtt{Pos}(r)$ *satisfying* $|\check{p}| = \mathtt{height}(r)$. *Let $\bar{p}$ be the minimum prefix of $\check{p}$ in size satisfying that $r|_{\bar{p}}$ is a run and $|\bar{p}| > h(A)^2$.*

Then, there exists a run r' satisfying that $r[[r']]_{\bar{p}}$ is a run and $\mathtt{height}(r') > \mathtt{height}(r|_{\bar{p}})$.

Proof. Let $p, \tilde{p}$ be the positions given by Lemma 5.17. Thus, they satisfy the following conditions:

- $p < \tilde{p} \leq \check{p}$ and $|p| + h^2 < |\tilde{p}|$.
- $r|_p$ and $r|_{\tilde{p}}$ are runs such that $\mathtt{root}(r|_{\tilde{p}}) = \mathtt{root}(r|_p)$.
- $r[[r|_p]]_{\tilde{p}}$ is a run.

In order to conclude, it suffices to observe that the pumping $r[[r|_p]]_{\tilde{p}}$ at $\tilde{p}$ can be seen as a pumping at $\bar{p}$, since $r[[r|_p]]_{\tilde{p}}$ equals $r[[((r|_{\bar{p}})[[r|_p]]_{\tilde{p}-\bar{p}})]]_{\bar{p}}$. □

Lemma 5.19 *Let A be a $TA_{\neq,hom}$. Let r be a run of A such that* $\mathtt{height}(r) > \check{h}(A)$. *Then, there exists a position $\bar{p}$ in r and infinitely many different runs $r_1, r_2, \ldots$ of A such that:*

- *$|\bar{p}| > h(A)^2$ and $r|_{\bar{p}}$ is a run.*
- *all* $\mathtt{root}(r|_{\bar{p}}), \mathtt{root}(r_1), \mathtt{root}(r_2), \ldots$ *coincide.*
- *all pumpings $r[[r_1]]_{\bar{p}}, r[[r_2]]_{\bar{p}}, \ldots$ are runs.*

Proof. Let $\check{p}$ be any position in $\mathtt{Pos}(r)$ satisfying $|\check{p}| = \mathtt{height}(r)$. We choose $\bar{p}$ as the minimum prefix of $\check{p}$ in size satisfying that $r|_{\bar{p}}$ is a run and $|\bar{p}| > h^2$. By Corollary 5.18, there exists a run r_1 satisfying that $r_1' := r[[r_1]]_{\bar{p}}$ is a run and $\mathtt{height}(r_1) > \mathtt{height}(r|_{\bar{p}})$. Note that $\bar{p}$ is also the minimum prefix of a position in $\mathtt{Pos}(r_1')$ with length $\mathtt{height}(r_1')$. Thus, Corollary 5.18 can be applied again, concluding the existence of a run r_2 satisfying that $r_2' := r_1'[[r_2]]_{\bar{p}} = r[[r_2]]_{\bar{p}}$ is a run and $\mathtt{height}(r_2) > \mathtt{height}(r_2'|_{\bar{p}}) = \mathtt{height}(r_1)$. We conclude by noting that this inference can be iterated again and again. □

Corollary 5.20 *The finiteness problem is decidable for TA$_{\neq,hom}$.*

Proof. By Lemma 5.19, in order to decide this problem for a given a TA$_{\neq,hom}$$A = \langle Q, \Sigma, F, \Delta\rangle$, it suffices to check whether there is a run r of A with an accepting resulting state and satisfying $\mathtt{height}(r) > \check{h}(A)$. This question can be easily reduced to the emptiness problem, which is decidable, according to Corollary 5.11. To this end, it suffices to straightforwardly construct a new TA$_{\neq,hom}$ A' accepting the same language as $\mathcal{L}(A)$ minus the terms with height smaller than or equal to $\check{h}(A)$, and then decide emptiness of $\mathcal{L}(A')$. □

6 Consequences

Theorem 6.1 *The inclusion problem $\mathcal{L}(A) \supseteq \mathcal{L}(B)$ is decidable for a TA$_=$ A and a TA B given as input.*

Proof. By Theorem 3.5, the complement TA$_{\neq}$ $\bar{A}$ of A recognizes $\overline{\mathcal{L}(A)}$. It is well-known how to compute a new TA$_{\neq}$ recognizing the intersection of the languages represented by a TA$_{\neq}$ and a TA (in fact, our Definition 4.11 subsumes this construction). Thus, a TA$_{\neq}$ $\bar{A} \cap B$ recognizing $\overline{\mathcal{L}(A)} \cap \mathcal{L}(B)$ can be computed. It is well-known (see [7, 6]) that emptiness of a TA$_{\neq}$ is decidable (in fact, our Corollary 5.11 subsumes this result). Thus, we conclude by noting that deciding emptiness of $\overline{\mathcal{L}(A)} \cap \mathcal{L}(B)$ is equivalent to deciding $\mathcal{L}(A) \supseteq \mathcal{L}(B)$. □

Corollary 6.2 *The universality problem is decidable for TA$_=$.*

Proof. Deciding $\mathcal{L}(A) = \mathcal{T}(\Sigma)$ is equivalent to deciding $\mathcal{L}(A) \supseteq \mathcal{T}(\Sigma)$, and since $\mathcal{T}(\Sigma)$ is regular, from Theorem 6.1 it follows that universality is decidable. □

Corollary 6.3 *The finiteness problem of $\mathcal{L}(B) - \mathcal{L}(A)$ is decidable for a TA$_=$ A and a TA B given as input.*

Proof. As in the proof of Theorem 6.1, we can construct a TA$_{\neq}$ recognizing $\overline{\mathcal{L}(A)} \cap \mathcal{L}(B)$. In order to conclude, we mention that it is well-known that finiteness of a TA$_{\neq}$ is decidable (in fact, our Corollary 5.20 subsumes this result).

Corollary 6.4 *The regularity test is undecidable for TA$_{\neq}$, and thus, for reduction tree automata.*

Proof. Since regularity is undecidable for $TA_=$, our transformation of a $TA_=$ into a $TA_{\neq}$ recognizing the complement is a reduction from this problem into regularity of a $TA_{\neq}$. Since $TA_{\neq}$ are a particular case of reduction automata, the second part of the statement follows, too.

Theorem 6.5 *The inclusion problem is decidable for images of tree homomorphisms, that is, $\mathcal{L}(H_A(A)) \supseteq \mathcal{L}(H_B(B))$ is decidable for a TA A, a TA B, and tree homomorphisms H_A and H_B given as input.*

Proof. By Proposition 4.6, two $TA_=$ A' and B' recognizing $\mathcal{L}(H_A(A))$ and $\mathcal{L}(H_B(B))$, respectively, can be computed. By Lemma 4.8 and Theorem 3.5, the complement $TA_{\neq}$ $\bar{A}'$ of A' recognizes $\overline{\mathcal{L}(A')}$. By Proposition 4.13, $\bar{A}' \cap B'$ recognizes $\overline{\mathcal{L}(A')} \cap \mathcal{L}(B')$. By Corollary 5.11, emptiness of $\overline{\mathcal{L}(A')} \cap \mathcal{L}(B')$ is decidable. Thus, we conclude by noting that deciding emptiness of $\overline{\mathcal{L}(A')} \cap \mathcal{L}(B')$ is equivalent to deciding $\mathcal{L}(H_A(A)) \supseteq \mathcal{L}(H_B(B))$. □

Corollary 6.6 *The equivalence problem is decidable for images of tree homomorphisms, that is, $\mathcal{L}(H_A(A)) = \mathcal{L}(H_B(B))$ is decidable for a $TA_=$ A, a TA B, and tree homomorphisms H_A and H_B given as input.*

Corollary 6.7 *The inclusion and equivalence problems are decidable for ranks of bottom up tree transducers.*

Corollary 6.8 *The finiteness problem of $\mathcal{L}(H_A(A)) - \mathcal{L}(H_B(B))$ is decidable for a TA A, a TA B, and tree homomorphisms H_A and H_B given as input.*

Proof. As in the proof of Theorem 6.5, we can construct a $TA_{\neq}$ recognizing $\overline{\mathcal{L}(H_A(A))} \cap \mathcal{L}(H_B(B))$. In order to conclude, we note that, by Corollary 5.20, finiteness of $\overline{\mathcal{L}(H_A(A))} \cap \mathcal{L}(H_B(B))$ is decidable. □

7 Decision of the HOM problem

In a first subsection we present a simple algorithm deciding the HOM problem. In two next subsections we argue the correctness of the algorithm.

7.1 The algorithm deciding the HOM problem

Definition 7.1 *(linearization of a TA_{hom}) Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA_{hom}. Let h be a natural number. The* linearization *of A_{hom} by h is the TA_{hom} $\langle Q, \Sigma, F, \Delta' \rangle$, denoted $\texttt{linearize}(A_{hom}, h)$, where Δ' is the set of all rules of the form $s[s_1]_{p_1} \ldots [s_n]_{p_n} \to q$ such that:*

- *A rule of the form $s \xrightarrow{c} q$ occurs in Δ.*
- *$p_1, \ldots, p_n$ are the positions ocurring in c.*
- *For each i in $\{1, \ldots, n\}$, s_i is a term in $\mathcal{L}(A_{hom}, s|_{p_i})$ such that $\texttt{height}(s_i) \leq h$.*

- *For each i, j in $\{1, \dots, n\}$ such that p_i is different from p_j and $(p_i = p_j)$ occurs in c, $s_i = s_j$ holds.*

It is straightforward that a linearization of any TA_{hom} is computable and recognizes a regular language, since no equality constraints appear. It is also clear that $\mathcal{L}(A_{hom})$ includes the language of any of its linearizations. Moreover, in the case where $\mathcal{L}(A_{hom})$ is included in some of its linearizations, we can conclude that $\mathcal{L}(A_{hom})$ is regular.

Example 7.2 *Let $A_{\texttt{ex3}} = \langle Q, \Sigma, F, \Delta\rangle$ be a TA_{hom} such that $Q = F = \{q\}$ holds, $\Sigma = \{f^{(2)}, a^{(0)}\}$ holds, and the rules of Δ are:*

- $\rho_1 = a \to q$
- $\rho_2 = f(q, q) \xrightarrow{1=2} q$
- $\rho_3 = f(f(q, q), q) \to q$
- $\rho_4 = f(q, f(q, q)) \to q$

Then, the linearization $\texttt{linearize}(A_{\texttt{ex3}}, 0)$ is $\langle Q, \Sigma, F, \Delta'\rangle$, where Δ' is the set of rules $\{\rho'_1, \rho_2, \rho_3, \rho_4\}$ such that ρ'_1 is the rule $f(a, a) \to q$. It is easy to see that the language of the linearization contains the original language of $A_{\texttt{ex3}}$, and it follows that $\mathcal{L}(A_{\texttt{ex3}})$ is regular.

But if we consider a $TA_{\neq,hom}$ $A_{\texttt{ex4}} = \langle Q, \Sigma, F, \Delta''\rangle$, where Δ'' is $\Delta - \{\rho_3\}$, it can be proved that no linearization of $A_{\texttt{ex4}}$ contains the original language.

The key point for deciding the HOM problem using linearization is stated by the following lemma.

Lemma 7.3 *Let A_{hom} be a TA_{hom}. Let $\check{h}$ be $\check{h}(A_{hom} \cap \overline{A_{hom}})$. Suppose that $\mathcal{L}(A_{hom})$ is not included in $\mathcal{L}(\texttt{linearize}(A_{hom}, \check{h}))$. Then, $\mathcal{L}(A_{hom})$ is not regular.*

The proof of this lemma is done along the next subsections. It provides a simple decision algorithm for the HOM problem, described as follows.

- Input: A tree automaton A and a tree homomorphism H.
- Construct a TA_{hom} A_{hom} recognizing $H(A)$.
- Construct the linearization B of A_{hom} by $\check{h}(A_{hom} \cap \overline{A_{hom}})$.
- If $\mathcal{L}(\texttt{linearize}(A_{hom}, \check{h}(A_{hom} \cap \overline{A_{hom}})))$ includes $\mathcal{L}(A_{hom})$ then halt with output "REGULAR".
- Otherwise, halt with output "NON-REGULAR".

Theorem 7.4 *The HOM problem is decidable.*

To conclude, it remains to prove Lemma 7.3.

7.2 A non-terminating process detecting non-regularity

In order to prove Lemma 7.3, we describe a non-terminating process. We emphasize that this process is not executed in order to decide regularity (the algorithm has already been presented in previous section). It will just help us to argue about the certainness of Lemma 7.3.

The process deals with sets of terms with equality constraints. They represent an infinite set of ground terms: the ones obtained by applying substitutions holding the constraints.

Definition 7.5 *(constrained terms) Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_{hom}$. A* constrained term *with respect to A_{hom} is a pair $t|c$, where t is a term in $\mathcal{T}(\Sigma \cup Q)$, and c is a conjunction/set of equalities of positions $(p_1 = p_2)$ satisfying that $t|_{p_1} = t|_{p_2} \in Q$ holds and p_1 and p_2 are different. Moreover, if $(p_1 = p_2), (p_2 = p_3)$ occur in c, then $(p_1 = p_3)$ also occurs in c, for arbitrary positions p_1, p_2, p_3. We identify a term t with a constrained term $t|\emptyset$, that is, with an empty conjunction of equalities. We define also the replacement of a subterm $t|_p$ in a term t at position p by a constrained term $s|c$, denoted $t[s|c]_p$, as $t[s]| \bigwedge_{(p_1=p_2)\in c}(p.p_1 = p.p_2)$.*

An instance *of $t|c$ is a ground term of the form $t[s_1]_{p_1} \dots [s_n]_{p_n}$, where $\{p_1, \dots, p_n\}$ are the positions p_i satisfying $t|_{p_i} \in Q$, $\{s_1, \dots, s_n\}$ are ground terms satisfying $s_i \in \mathcal{L}(A_{hom}, t|_{p_i})$ such that, for each $(p_i = p_j)$ occurring in c, $s_i = s_j$ holds. The set of instances of a constrained term $t|c$ with respect to A_{hom} is denoted $\texttt{instances}(t|c, A_{hom})$, or $\texttt{instances}(t|c)$ when A_{hom} is clear from the context. The* set of instances *of a set S of constrained terms with respect to a TA$_{hom}$ A_{hom}, denoted $\texttt{instances}(S, A_{hom})$ or $\texttt{instances}(S)$ when A_{hom} is clear from the context, is $\bigcup_{(t|c)\in S}(\texttt{instances}(t|c))$.*

Definition 7.6 *(linearization of constrained terms) Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA$_{hom}$. Let h be a natural number. Let $t|c$ be a constrained term with respect to A_{hom}. The* linearization *of $t|c$ by h, denoted $\texttt{linearize}(t|c, h)$, is the set of terms $t[s_1]_{p_1} \dots [s_n]_{p_n}$, where $\{p_1, \dots, p_n\}$ are the positions occurring in c, $\{s_1, \dots, s_n\}$ are ground terms satisfying $s_i \in \mathcal{L}(A_{hom}, t|_{p_i})$, each $\texttt{height}(s_i)$ is smaller than or equal to h, and for each $(p_i = p_j)$ occurring in c, $s_i = s_j$ holds.*

The process has a TA$_{hom}$ $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ as input, constructs a set S of constrained terms with respect to A_{hom}, and proceeds by modifying S until it (eventually) detects a condition implying non-regularity. The following invariants are satisfied at the beginning of each step of the process:

- I1. Each $t|c$ in S satisfies that c is $\emptyset$.
- I2. $\texttt{instances}(S)$ is equal to $\mathcal{L}(A_{hom})$.
- I3. For each two terms t_1, t_2 in S and each two positions p_1, p_2 satisfying that $t_1|_{p_1}$ and $t_2|_{p_2}$ are in Q, $||p_2| - |p_1|| \leq h(A_{hom})$.

The description of the process is as follows

1. Input: A TA$_{hom}$ $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$.
2. Assign $S := F$.
3. If all terms in S are in $\mathcal{T}(\Sigma)$, i.e. no state in Q occurs in S, then halt with output "REGULAR".
4. Otherwise, let p be a position minimal in size among $\{p' \mid \exists t \in S : t|_{p'} \in Q\}$. Let t be a term of S satisfying $t|_p \in Q$. Let q be $t|_p$. Assign $S := (S - \{t\}) \cup \{t[s|c]_p \mid (s \xrightarrow{c} q) \in \Delta\}$. (Note that, at this point, S satisfies all the invariants except I1).
5. While S does not satisfy I1 do:
   a. Let $t'|c'$ be a constrained term in S where c' is not empty.
   b. If there exists a position p' in c' and a term t'' in $\texttt{instances}(t'|c') - \texttt{instances}(S - \{t'|c'\})$ such that $\texttt{height}(t''|_{p'}) > \check{h}(A_{hom} \cap \overline{A_{hom}})$, then halt with output "NON-REGULAR".
   c. Otherwise, assign $S := (S - \{t'|c'\}) \cup \texttt{linearize}(t'|c', \check{h}(A_{hom} \cap \overline{A_{hom}}))$.
6. Go to 3.

It is clear that the above process satisfies all the invariants when it passes throught item 3. We insists again that this process is not executed. Thus, it does not matter if the used instructions are computable (nevertheless they are).

Lemma 7.7 *If the process does not halt with output "NON-REGULAR", then* $\mathcal{L}(A_{hom}) \subseteq \mathcal{L}(\texttt{linearize}(A_{hom}, \check{h}(A_{hom} \cap \overline{A_{hom}})))$.

Proof. Let Δ' the set of rules of $\texttt{linearize}(A_{hom}, \check{h}(A_{hom} \cap \overline{A_{hom}}))$. Note that, when step 3 is executed, any term u inside S satisfies $u \rightarrow^*_{\Delta'} q'$ for some $q' \in F$. Thus, in order to conclude, it suffices to prove that any term u in $\mathcal{L}(A_{hom})$ is included in S at some point of the execution, under the assumptions of the lemma.

Assume that the process does not halt with output "NON-REGULAR". Let u be a term in $\mathcal{L}(A_{hom})$. Note that, either the process halts with output "REGULAR" at step 3, or it does not halt. In the first case, by Invariant I2, u belongs to S when the process halts. In the second case, at some point of the execution in step 3, the minimal position p in size among $\{p' \mid \exists t \in S : t|_{p'} \in Q\}$ satisfies $|p| > \texttt{height}(u)$. Thus, by Invariant I2, at this point of the execution, u is also in S, and we are done. □

Hence, in order to prove Lemma 7.3, it rests to see that, when the process halts with output "NON-REGULAR", $\mathcal{L}(A_{hom})$ is not regular. This reduces to check that, when the condition in step 5b of the process is satisfied, it follows that $\mathcal{L}(A_{hom})$ is not regular. This is done in the following subsection.

7.3 Correctness of the process

We will use the following lemma, which characterizes when a term is not an instance of a constrained term.

Lemma 7.8 *Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA_{hom}. Let $s|d$ be a constrained term. Let t be a term in $\mathcal{T}(\Sigma) - \texttt{instances}(\{s|d\})$. Then, one of the following conditions hold:*

- *There is a position p in $\texttt{Pos}_\Sigma(s) \cap \texttt{Pos}(t)$ such that $\texttt{root}(s|_p)$ is different from $\texttt{root}(t|_p)$.*
- *There is a position p in $\texttt{Pos}_Q(s)$ such that $t|_p$ is not in $\mathcal{L}(A, s|_p)$.*
- *There is an equality $(p_1 = p_2)$ in d such that $t|_{p_1}$ is different from $t|_{p_2}$.*

The following two lemmas allow to conclude that, when condition in step 5b of the process is satisfied, it follows that $\mathcal{L}(A_{hom})$ is not regular.

Lemma 7.9 *Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA_{hom}. Let S be a set of constrained terms satisfying I2 and I3. Let $t|c$ be a constrained term in S. Let $\check{p}_1, \ldots, \check{p}_n$ be the positions $\check{p}_i$ satisfying $t|_{\check{p}_i} \in Q$. Suppose that $\check{p}_1$ occurs in c, and that, without loss of generality, $\check{p}_2, \ldots, \check{p}_k$ are all positions $\check{p}_j$ such that $(\check{p}_1 = \check{p}_j)$ occurs in c. Suppose that $t' = t[t_1]_{\check{p}_1} \ldots [t_1]_{\check{p}_k}[t_{k+1}]_{\check{p}_{k+1}} \ldots [t_n]_{\check{p}_n}$ is a term in $\texttt{instances}(t|c) - \texttt{instances}(S - \{t|c\})$ such that $\texttt{height}(t_1) > \check{h}(A_{hom} \cap \overline{A_{hom}})$ holds.*

Then, there exist infinitely many terms $t_{1,1}, t_{1,2}, \ldots$ such that all $t[t_{1,j}]_{\check{p}_1} \ldots [t_{1,j}]_{\check{p}_k}[t_{k+1}]_{\check{p}_{k+1}} \ldots [t_n]_{\check{p}_n}$ are also terms in $\texttt{instances}(t|c) - \texttt{instances}(S - \{t|c\})$.

Proof. By the definition of instance, there exists a run r_1 of A_{hom} such that $\pi_\Sigma(r_1) = t_1$ and the resulting state of r_1 is $t|_{p_1}$. Let $\bar{r}$ be a run of $\overline{A_{hom}}$ satisfying $\pi_\Sigma(\bar{r}) = t'$ and all conditions given by Corollary 4.9. In particular, for each $p \in \texttt{Pos}(\bar{r})$ it holds that $\bar{r}|_p$ is a run with a resulting state including $\{q \in Q |\ t'|_p \notin \mathcal{L}(A_{hom}, q)\}$. Let $\hat{r}_1$ be the run $r_1 \cap \bar{r}|_{p_1}$ of $A_{hom} \cap \overline{A_{hom}}$. By Lemma 5.19, there exist a position $\bar{p}$ in $\hat{r}_1$ and infinitely many different runs $\hat{r}_{1,1}, \hat{r}_{1,2}, \ldots$ of $A_{hom} \cap \overline{A_{hom}}$ such that:

- $|\bar{p}| > h(A_{hom} \cap \overline{A_{hom}})^2$ and $\hat{r}_1|_{\bar{p}}$ is a run.
- All $\texttt{root}(\hat{r}_1|_{\bar{p}}), \texttt{root}(\hat{r}_{1,1}), \texttt{root}(\hat{r}_{1,2}), \ldots$ coincide.
- All pumpings $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}, \hat{r}_1[[\hat{r}_{1,2}]]_{\bar{p}}, \ldots$ are runs.

We define $\hat{t}_{1,1} := \pi_\Sigma(\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}})$, $\hat{t}_{1,2} := \pi_\Sigma(\hat{r}_1[[\hat{r}_{1,2}]]_{\bar{p}})$, ...

Now, consider p as any position in $\texttt{Pos}(\hat{r}_1)$ satisfying $|p| \leq h(A_{hom} \cap \overline{A_{hom}})$. Since $|\bar{p}| > h(A_{hom} \cap \overline{A_{hom}})^2$, by the last part of the statement in Lemma 5.3, no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$ is a proper prefix of p. Thus, for each of such positions p and each $j \geq 1$, $\texttt{root}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}|_p)$ coincides with $\texttt{root}(\hat{r}_1|_p)$.

By Lemma 4.19, $\pi_{\neq}(\hat{r}_1) = \bar{r}|_{\breve{p}_1}$ and $\pi_{hom}(\hat{r}_1) = r_1$ hold, and by Lemma 4.16, for each of such positions p and each $j \geq 1$, the resulting states of $\bar{r}|_{\breve{p}_1}|_p$ and $(\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}))|_p$ coincide. Moreover, again by Lemma 4.16, the resulting states of r_1 and $\pi_{hom}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})$ coincide, i.e. they are $t|_{\breve{p}_1}$.

One of the particular implications of the above comments is that each $t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ is an instance of $t|c$. Thus, at first look, we could try to define the desired terms $t_{1,1}, t_{1,2}, \ldots$ of the statement of the lemma as $\hat{t}_{1,1}, \hat{t}_{1,2}, \ldots$, respectively. The problem is that some $t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ could also be in $\mathtt{instances}(S - \{t|c\})$. In order to conclude, it suffices to see that only a finite number of them satisfy this condition. To this end, we show that, for each $s|d$ in $S - \{t|c\}$, at most a finite number of terms $t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ are instances of $s|d$.

Consider any constrained term $s|d$ in $S - \{t|c\}$. Since t' is not an instance of $s|d$, according to Lemma 7.8, we can distinguish the following cases:

- Assume that there is a position p in $\mathtt{Pos}_\Sigma(s) \cap \mathtt{Pos}(t')$ such that $\mathtt{root}(s|_p)$ is different from $\mathtt{root}(t'|_p)$. We distinguish the following cases:

  - Suppose first that p is of the form $\breve{p}_i.p'.p''$ for some i in $\{1, \ldots, k\}$ and some positions p', p'' such that $|p'| = h(A_{hom} \cap \overline{A_{hom}})$. By Invariant I3, it holds that $s|_{\breve{p}_i.p'}$ is a term in $\mathcal{T}(\Sigma)$, i.e. without any symbol in Q. Note that $t'|_{\breve{p}_i.p'}$ is a term different from $s|_{\breve{p}_i.p'}$ because they differ at the symbol located at their relative position p''. Recall that no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$ is a proper prefix of p'. Thus, by Lemma 5.2, at most one term $\hat{t}_{1,j}$ makes $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{\breve{p}_i.p'}$ equal to $s|_{\breve{p}_i.p'}$, and we are done.
  - Second, suppose p is not of the form $\breve{p}_i.p'.p''$ for some i in $\{1, \ldots, k\}$ and some positions p', p'' such that $|p'| = h(A_{hom} \cap \overline{A_{hom}})$. In this case, $\mathtt{root}(t'|_p)$ coincides with all $\mathtt{root}((t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_p)$, and thus, it differs from $\mathtt{root}(s|_p)$. Hence, none of these terms is an instance of $s|d$, and we are done.

- Assume that there is a position p in $\mathtt{Pos}_Q(s)$ such that $t'|_p$ is not in $\mathcal{L}(A, s|_p)$. Let q be $s|_p$ and let S_p be the resulting state of $\bar{r}|_p$. By the election of $\bar{r}$, it holds that q is in S_p.

  Let us fix a $j \geq 1$ and suppose that $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k}$ is a run of $\overline{A_{hom}}$. Note that, by Invariant I3, either no position in $\breve{p}_1, \ldots, \breve{p}_k$ is a prefix of p, or p is of the form $\breve{p}_i.p'$ for some i in $\{1, \ldots, k\}$ and some p' satisfying $|p'| \leq h(A_{hom}) \leq h(A_{hom} \cap \overline{A_{hom}})$. In the second case, recall that, for such a position p' , $\mathtt{root}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}|_{p'})$ coincides with $\mathtt{root}(\hat{r}_1|_{p'})$. Thus, in any case, the resulting state of $(\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k})|_p$ is S_p. By Lemma 4.9, it follows that $t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ is not an instance of $s|d$.

Hence, in order to conclude, it suffices to argue that the terms $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k}$ are runs of $\overline{A_{hom}}$ except for a finite number of j's.

Note that $\bar{r}$ and each term $\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})$ is a run of $\overline{A_{hom}}$. Thus, if, for a concrete j, $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k}$ is not a run of $\overline{A_{hom}}$, there must exist a position p'' satisfying the following assumptions:

- $p'' < \breve{p}_i$ for some i in $\{1, \ldots, k\}$.
- $\texttt{root}(\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k})|_{p''}$ is a rule $f(q_1, \ldots, q_m) \xrightarrow{e} q'$ where e contains a disequality $p_1 \neq p_2$ such that $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p''.p_1}$ is equal to $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p''.p_2}$.

Since $\bar{r}$ is a run of $\overline{A_{hom}}$, $t'|_{p''.p_1}$ is different from $t'|_{p''.p_2}$. Recall that, all replaced positions in $\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}$ have length greater than $h(A_{hom} \cap \overline{A_{hom}})$. Thus, $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p''.p_1}$ and $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p''.p_2}$ can be obtained from $t'|_{p''.p_1}$ and $t'|_{p''.p_2}$, respectively, by replacing $t'|_{\breve{p}_1.\bar{p}}$ by $\pi_\Sigma(\hat{r}_{1,j}))$ at some positions which are independent from j. By Lemma 5.2, only one term can satisfy this statement for such p'' and $p_1 \neq p_2$.

The elections for p'' and $p_1 \neq p_2$ are finitely bounded. Thus, at most for a finite number of j's, $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1} \ldots [\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_k}$ is not a run, and we are done.

- Finally, assume that there is an equality $(p_1 = p_2)$ in d such that $t'|_{p_1}$ is different from $t'|_{p_2}$. Note that both $s|_{p_1}$ and $s|_{p_2}$ are identical and in Q. Hence, by Invariant I3, p_1 and p_2 are not of the form $\breve{p}_i.p'.p''$ for some i in $\{1, \ldots, k\}$ and some positions p', p'' such that $|p'| = h(A_{hom}) \leq h(A_{hom} \cap \overline{A_{hom}})$. Note also that, in the case where some $\breve{p}_i$ for i in $\{1, \ldots, k\}$ is a prefix of p_1 (p_2), no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$ is a proper prefix of $p_1 - \breve{p}_i$ ($p_2 - \breve{p}_i$). Thus, $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p_1}$ and $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p_2}$ can be obtained from $t'|_{p_1}$ and $t'|_{p_2}$, respectively, by replacing $t'|_{\breve{p}_1.\bar{p}}$ by $\pi_\Sigma(\hat{r}_{1,j})$ at some positions which are independent from j. By Lemma 5.2, at most one term $\hat{t}_{1,j}$ makes $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p_1}$ equal to $(t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [\hat{t}_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n})|_{p_2}$. Hence, for all the remaining j's, $t[\hat{t}_{1,j}]_{\breve{p}_1} \ldots [t'_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ is not an instance of $s|d$, due to the same reason as t', and we are done.

□

Lemma 7.10 *Let $A_{hom} = \langle Q, \Sigma, F, \Delta \rangle$ be a TA_{hom}. Let S be a set of constrained terms satisfying I2 and I3. Let $t|c$ be a constrained term in S. Let $\breve{p}_1, \ldots, \breve{p}_n$ be the positions $\breve{p}_i$ satisfying $t|_{\breve{p}_i} \in Q$. Suppose that $\breve{p}_1$ occurs in c, and that, without loss of generality, $\breve{p}_2, \ldots, \breve{p}_k$ are all positions $\breve{p}_j$ such that $(\breve{p}_1 = \breve{p}_j)$ occurs in c. Suppose that there exist terms $t_{k+1}, \ldots, t_n$ and infinitely* *many terms* $t_{1,1}, t_{1,2}, \ldots$ *such that all* $t[t_{1,j}]_{\breve{p}_1} \ldots [t_{1,j}]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ *are in* $\texttt{instances}(t|c) - \texttt{instances}(S - \{t|c\})$.

Then, $\texttt{instances}(S)$ *is not regular.*

Proof. We proceed by contradiction by assuming that $\texttt{instances}(S)$ is regular. Note that, in particular, C is a TA_{hom}. Thus, let C be a TA recognizing $\texttt{instances}(S)$. Among all the terms $t_{1,j}$ we choose one, called t_1, such that $\texttt{height}(t_1) > \check{h}(C \cap \overline{A_{hom}})$ holds. Note that $t' = t[t_1]_{\breve{p}_1} \ldots [t_1]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ is in $\texttt{instances}(t|c) - \texttt{instances}(S - \{t|c\})$. In particular, t' is in $\texttt{instances}(S)$, and hence, t' belongs to $\mathcal{L}(C)$. Thus, there exists a run r of C with a resulting accepting state such that $\pi_\Sigma(r) = t'$. Let $\bar{r}$ be a run of $\overline{A_{hom}}$ satisfying $\pi_\Sigma(\bar{r}) = t'$ and all conditions given by Lemma 4.9. In particular, for each $p \in \texttt{Pos}(\bar{r})$ it holds that $\bar{r}|_p$ is a run with a resulting state including $\{q \in Q |\ t'|_p \notin \mathcal{L}(A_{hom}, q)\}$. Let $\hat{r}_1$ be the run $r|_{\breve{p}_1} \cap \bar{r}|_{\breve{p}_1}$ of $C \cap \overline{A_{hom}}$. By Lemma 5.19, there exist a position $\bar{p}$ in $\hat{r}_1$ and infinitely many different runs $\hat{r}_{1,1}, \hat{r}_{1,2}, \ldots$ of $C \cap \overline{A_{hom}}$ such that:

- $|\bar{p}| > h(C \cap \overline{A_{hom}})^2$ and $\hat{r}_1|_{\bar{p}}$ is a run.
- All $\texttt{root}(\hat{r}_1|_{\bar{p}}), \texttt{root}(\hat{r}_{1,1}), \texttt{root}(\hat{r}_{1,2}), \ldots$ coincide.
- All pumpings $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}, \hat{r}_1[[\hat{r}_{1,2}]]_{\bar{p}}, \ldots$ are runs.

We define $\hat{t}_{1,1} := \pi_\Sigma(\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}})$, $\hat{t}_{1,2} := \pi_\Sigma(\hat{r}_1[[\hat{r}_{1,2}]]_{\bar{p}})$, ….

Now, consider p as any position in $\texttt{Pos}(\hat{r}_1)$ satisfying $|p| \leq h(C \cap \overline{A_{hom}})$. Since $|\bar{p}| > h(C \cap \overline{A_{hom}})^2$, by the last part of the statement in Lemma 5.3, no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$ is a proper prefix of p. Thus, for each of such positions p and each $j \geq 1$, $\texttt{root}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}|_p)$ coincides with $\texttt{root}(\hat{r}_1|_p)$. By Lemma 4.19, $\pi_{\neq}(\hat{r}_1) = \bar{r}|_{\breve{p}_1}$ and $\pi_{hom}(\hat{r}_1) = r|_{\breve{p}_1}$ hold, and by Lemma 4.16, for each of such positions p and each $j \geq 1$, the resulting states of $\bar{r}|_{\breve{p}_1}|_p$ and $(\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}))|_p$ coincide. Moreover, again by Lemma 4.16, the resulting states of $r|_{\breve{p}_1}$ and $\pi_{hom}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})$ coincide.

One of the particular implications of the above comments is that each $t[\hat{t}_{1,j}]_{\breve{p}_1}$ is in $\mathcal{L}(C)$. It is also clear that $t'[\hat{t}_{1,j}]_{\breve{p}_1}$, for $\hat{t}_{1,j} \neq t_1$, is not an instance of $t|c$, since the terms at the positions $\breve{p}_1$ and $\breve{p}_2$ differ. Thus, in order to reach a contradiction, it suffices to prove that, for some j satisfying $\hat{t}_{1,j} \neq t_1$, $t'[\hat{t}_{1,j}]_{\breve{p}_1}$ is not an instance of $S - \{t|c\}$. To conclude we will prove that, only for a finite number of j's, the terms $t'[\hat{t}_{1,j}]_{\breve{p}_1}$ are instances of $S - \{t|c\}$. To this end, we show that, for each $s|d$ in $S - \{t|c\}$, at most a finite number of terms $t'[\hat{t}_{1,j}]_{\breve{p}_1} = t[\hat{t}_{1,j}]_{\breve{p}_1}[t_1]_{\breve{p}_2} \ldots [t_1]_{\breve{p}_k}[t_{k+1}]_{\breve{p}_{k+1}} \ldots [t_n]_{\breve{p}_n}$ are instances of $s|d$.

Consider any constrained term $s|d$ in $S - \{t|c\}$. Since t' is not an instance of $s|d$, according to Lemma 7.8, we can distinguish the following cases:

- Assume that there is a position p in $\texttt{Pos}_\Sigma(s) \cap \texttt{Pos}(t')$ such that $\texttt{root}(s|_p)$ is different from $\texttt{root}(t'|_p)$. We distinguish the following cases:
  - Suppose first that p is of the form $\breve{p}_1.p'.p''$ for some positions p', p'' such that $|p'| = h(C \cap \overline{A_{hom}}) \geq h(A_{hom})$. By Invariant I3, it holds

that $s|_{\breve{p}_1.p'}$ is a term in $\mathcal{T}(\Sigma)$, i.e. without any symbol in Q. Note that $t'|_{\breve{p}_1.p'}$ is a term different from $s|_{\breve{p}_1.p'}$ because they differ at the symbol located at their relative position p''. Recall that no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$ is a proper prefix of p'. Thus, by Lemma 5.2, at most one term $\hat{t}_{1,j}$ makes $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{\breve{p}_1.p'}$ equal to $s|_{\breve{p}_1.p'}$, and we are done.

  – Second, suppose p is not of the form $\breve{p}_1.p'.p''$ for some positions p', p'' such that $|p'| = h(C \cap \overline{A_{hom}})$. In this case, $\mathtt{root}(t'|_p)$ coincides with all $\mathtt{root}(t'[\hat{t}_{1,j}]_{\breve{p}_1}|_p)$, and thus, it differs from $\mathtt{root}(s|_p)$. Hence, none of these terms is an instance of $s|d$, and we are done.

- Assume that there is a position p in $\mathtt{Pos}_Q(s)$ such that $t'|_p$ is not in $\mathcal{L}(A, s|_p)$. Let q be $s|_p$ and let S_p be the resulting state of $\bar{r}|_p$. By the election of $\bar{r}$, it holds that q is in S_p.

  Let us fix a $j \geq 1$ and suppose that $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1}$ is a run of $\overline{A_{hom}}$. Note that, by Invariant I3, either $\breve{p}_1$ is not a prefix of p, or p is of the form $\breve{p}_1.p'$ for some p' satisfying $|p'| \leq h(A_{hom}) \leq h(C \cap \overline{A_{hom}})$. In the second case, recall that, for such position p' , $\mathtt{root}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}|_{p'})$ coincides $\mathtt{root}(\hat{r}_1|_{p'})$. Thus, in any case, the resulting state of $(\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1}|_p$ is S_p. By Lemma 4.9, it follows that $t'[\hat{t}_{1,j}]_{\breve{p}_1}$ is not an instance of $s|d$.

  Hence, in order to conclude, it suffices to argue that the terms $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1}$ are runs of $\overline{A_{hom}}$ except for a finite number j's.

  Note that $\bar{r}$ and each term $\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})$ is a run of $\overline{A_{hom}}$. Thus, if, for a concrete j, $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1}$ is not a run of $\overline{A_{hom}}$, there must exist a position p'' satisfying the following assumptions:

  – $p'' < \breve{p}_1$.
  – $\mathtt{root}(\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1})|_{p''}$ is a rule $f(q_1, \ldots, q_m) \xrightarrow{e} q'$ where e contains a disequality $p_1 \neq p_2$ such that $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p''.p_1}$ is equal to $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p''.p_2}$.

  Since $\bar{r}$ is a run of $\overline{A_{hom}}$, $t'|_{p''.p_1}$ is different from $t'|_{p''.p_2}$. Recall that, all replaced positions in $\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}}$ have length greater than $h(C \cap \overline{A_{hom}})$. Thus, $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p''.p_1}$ and $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p''.p_2}$ can be obtained from $t'|_{p''.p_1}$ and $t'|_{p''.p_2}$, respectively, by replacing $t'|_{\breve{p}_1.\bar{p}}$ by $\pi_\Sigma(\hat{r}_{1,j}))$ at some positions which are independent from j. By Lemma 5.2, only one term can satisfy this statement for such p'' and $p_1 \neq p_2$.

  The elections for p'' and $p_1 \neq p_2$ are finitely bounded. Thus, at most for a finite number of j's, $\bar{r}[\pi_{\neq}(\hat{r}_1[[\hat{r}_{1,j}]]_{\bar{p}})]_{\breve{p}_1}$ is not a run, and we are done.

- Finally, assume that there is an equality $p_1 = p_2$ in d such that $t'|_{p_1}$ is different from $t'|_{p_2}$. Note that both $s|_{p_1}$ and $s|_{p_2}$ are identical and in Q. Hence, by Invariant I3, p_1 and p_2 are not of the form $\breve{p}_1.p'.p''$ for some positions p', p'' such that $|p'| = h(A_{hom}) \leq h(C \cap \overline{A_{hom}})$. Note also that, in the case where $\breve{p}_1$ is a prefix of p_1 (p_2), no replaced position in $\hat{r}_1[[\hat{r}_{1,1}]]_{\bar{p}}$

is a proper prefix of $p_1 - \breve{p}_1$ ($p_2 - \breve{p}_1$). Thus, $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p_1}$ and $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p_2}$ can be obtained from $t'|_{p_1}$ and $t'|_{p_2}$, respectively, by replacing $t'|_{\breve{p}_1.\bar{p}}$ by $\pi_\Sigma(\hat{r}_{1,j})$ at some positions which are independent from j. By Lemma 5.2, at most one term $\hat{t}_{1,j}$ makes $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p_1}$ equal to $t'[\hat{t}_{1,j}]_{\breve{p}_1}|_{p_2}$. Hence, for all the rest j's, $t'[\hat{t}_{1,j}]_{\breve{p}_1}$ is not an instance of $s|d$, due to the same reason as t', and we are done.

□

8 conclusion

We have closed affirmatively the open question of the decidability of the HOM problem. It remains to study in detail several aspects related to the complexity of the problem. In particular, it will be interesting to study the new class $\mathrm{TA}_{\neq,hom}$ and their properties. Our constructions also provide new tools to deal with tree homomorphisms applied to regular languages. It would be interesting to study further consequences of them.

References

[1] Franz Baader and Tobias Nipkow. *Term Rewriting and All That.* Cambridge University Press, New York, 1998.

[2] V. Benzaken, G. Castagna, H. Hosoya, B.C. Pierce, and S. Vansummeren. *The Encyclopedia of Database Systems*, chapter "XML Typechecking". Springer. To appear.

[3] B. Bogaert, F. Seynhaeve, and S. Tison. The recognizability problem for tree automata with comparison between brothers. In W. Thomas, editor, *Foundations of Software Science and Computation Structures - FOSSACS'99*, volume 1578 of *LNCS*, pages 150–164, Berlin, Germany, 1999. Springer-Verlag.

[4] B. Bogaert and S. Tison. Equality and disequality constraints on direct subterms in tree automata. In *International Symposium on Theoretical Aspects of Computeter Science*, pages 161–171, 1992.

[5] H. Comon, M. Dauchet, R. Gilleron, C. Löding, F. Jacquemard, D. Lugiez, S. Tison, and M. Tommasi. Tree automata techniques and applications. Available at http://www.grappa.univ-lille3.fr/tata, 2007.

[6] H. Comon and F. Jacquemard. Ground reducibility is EXPTIME-complete. In *Logic in Computer Science (LICS)*, pages 26–34, 1997.

[7] Hubert Comon and Florent Jacquemard. Ground reducibility and automata with disequality constraints. In *STACS*, pages 151–162, 1994.

[8] M. Dauchet, A.-C. Caron, and J.-L. Coquidé. Automata for reduction properties solving. *Journal of Symbolic Computation*, 20(2):215–233, 1995.

[9] Z. Fülöp. Undecidable properties of deterministic top-down tree transducers. *Theoret. Comput. Sci.*, 134:311–328, 1994.

[10] F. Gécseg and M. Steinby. Tree automata, 1984.

[11] F. Gécseg and M. Steinby. Tree languages. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages, Volume 3*, chapter 1. Springer, 1997.

[12] G. Godoy and E. Huntingford. Innermost-reachability and innermost-joinability are decidable for shallow term rewrite systems. In F. Baader, editor, *Term Rewriting and Applications, 18th International Conference, RTA*, volume 4533 of *LNCS*, pages 184–199. Springer, June 2007.

[13] G. Godoy, S. Maneth, and S. Tison. Classes of tree homomorphisms with decidable preservation of regularity. In *FoSSaCS*, volume 4962 of *LNCS*, pages 127–141. Springer, 2008.

[14] G. Kucherov and M. Tajine. Decidability of regularity and related properties of ground normal form languages. *Inf. Comput.*, 118:91–100, 1995.

[15] M. Murata, D. Lee, M. Mani, and K. Kawaguchi. Taxonomy of XML schema languages using formal language theory. *ACM Trans. Internet Techn.*, 5(4):660–704, 2005.

[16] J. W. Thatcher. Tree automata: an informal survey, 1973.

[17] James W. Thatcher and Jesse B. Wright. Generalized finite automata theory with an application to a decision problem of second-order logic. *Mathematical Systems Theory*, 2(1):57–81, 1968.

[18] Sándor Vágvölgyi and Rémi Gilleron. For a rewrite system it is decidable whether the set of irreducible, ground terms is regognizable. *Bulletin of the EATCS*, 48:197–209, 1992.