Abstract
The HOM-problem asks whether the image of a regular tree language under a given tree homomorphism is again regular. It was recently shown to be decidable by Godoy, Giménez, Ramos, and Álvarez. In this paper, the $N$-weighted version of this problem is considered and its decidability is proved. More precisely, it is decidable in polynomial time whether the image of a regular $N$-weighted tree language under a nondeleting, nonerasing tree homomorphism is regular.

1 Introduction
The prominent model of finite-state string automata has seen a variety of extensions in the past few decades. Notably, their qualitative evaluation was generalized to a quantitative one to yield the weighted automata of [26]. These automata are able to neatly represent process factors such as costs, consumption of resources or time, and probabilities related to the input, and have been extensively studied [25]. Semirings [17, 18] present themselves as a well suited algebraic structure for evaluating the weights because of their generality as well as their reasonable computational efficiency that is derived from distributivity.

Parallel to this development, finite-state string automata have been generalized to process other forms of inputs such as infinite words [23], graphs [3] and trees [5]. Finite-state tree automata and the regular tree languages they generate have been widely researched since their introduction in [7, 27, 28]. These models prove to be useful in a variety of application areas including natural language processing [19], image generation [8], and compiler construction [29]. Many applications require both features: trees as more expressive input structure and quantitative evaluation. This led to the development of several weighted tree automata (WTA) models. An extensive overview can be found in [12, Chapter 9].

Finite-state tree automata have serious limitations; notably, they cannot guarantee that two specific subtrees are always equal in the accepted trees provided that those subtrees can be arbitrarily large. Similarly finite-state string automata cannot ensure that the number of $a$'s and $b$'s in the accepted words is equal. These restrictions are well-known [13], and the mentioned drawback was addressed in [21], where an extension was proposed that can explicitly require certain subtrees to be equal or different. This extension is very convenient in the study of tree transformations [12] that can duplicate subtrees, and it is also the primary tool used in the seminal paper [15] to prove the decidability of the HOM-problem.
The HOM-problem, a previously long-standing open question in the study of tree languages, asks whether the image of a regular tree language under a given tree homomorphism is also regular. The image need not be regular since tree homomorphisms can also duplicate subtrees. Indeed, if this duplication ability is removed from the tree homomorphism (e.g., linear tree homomorphisms), then the image is always regular [13]. The HOM-problem was recently solved in [15, 16], where the image is represented by a tree automaton with constraints, for which it is then determined whether it generates a regular tree language. Later the HOM-problem was shown to be EXPTIME-complete [6].

In the weighted case, decidability of the HOM-problem remains wide open. Previous research on the preservation of regularity in the weighted setting [4, 9, 10, 11] focuses on cases that explicitly exclude the duplication power of the homomorphism. Recently, the weighted HOM-problem over zero-sum free semirings was addressed, but only solved for significantly restricted inputs [22]. In the present work, we prove that the HOM-problem for regular weighted tree languages over the semiring \( \mathbb{N} \) of nonnegative integers can be decided in polynomial time. The proof outline is as follows: Consider such a regular \( \mathbb{N} \)-weighted tree language and a nondeleting, nonerasing tree homomorphism. First, we represent this image efficiently using an extension (WTGh) of weighted tree automata that permits constraints [20]. Next, we ask whether this WTGh generates a regular weighted tree language. This semantic problem is reduced to an easier, essentially syntactic property: the large duplication property. In turn, this allows us to prove decidability of the weighted HOM-problem in polynomial time by directly proving it for the large duplication property. If the WTGh for the homomorphic image does not have this property, then we give an effective construction of an equivalent \( \mathbb{N} \)-weighted tree automaton without constraints (albeit in exponential time), thus proving its regularity. Otherwise, we use a pumping lemma presented in [20] and isolate a strictly non-regular part from the WTGh. The most challenging part of our proof and our main technical contribution is showing that the remaining trees in the homomorphic image cannot compensate for the non-regular behavior of this isolated part. For this, we employ Ramsey's theorem [24] to identify a witness for the non-regularity of the whole weighted tree language.

Compared to the unweighted case where the HOM-problem is EXPTIME-complete [6], the \( \mathbb{N} \)-weighted HOM-problem can be decided in polynomial time. Both proofs reduce the (non)regularity of the homomorphic image in question to a decidable property of the tree grammar with constraints representing it; however, the unweighted regularity notion is very different from the corresponding notion for weighted tree languages over \( \mathbb{N} \). Unlike the Boolean semiring \( \mathbb{B} \), which corresponds to the unweighted case, the semiring \( \mathbb{N} \) can be embedded into a field, which allows us to apply methods of linear algebra. The large duplication property, to which we successfully reduce the \( \mathbb{N} \)-weighted HOM-problem, is certainly necessary, but insufficient in the unweighted case. This is due to the fact that the Boolean semiring is idempotent, which permits covering an irregular tree language with the help of a regular one (e.g., the union \( L \cup T_\Sigma = T_\Sigma \) of an irregular tree language \( L \) with the regular tree language \( T_\Sigma \) of all trees is again regular). We will prove that such covers cannot happen in the semiring \( \mathbb{N} \) of nonnegative integers. As a consequence, the large duplication property is necessary and sufficient for non-regularity of weighted tree languages over \( \mathbb{N} \). In summary, our overall strategy for approaching the HOM-problem is similar to [15], but the required notions and details of the proofs significantly differ.

Tree automata and grammars with constraints are applied in domains such as automated deduction [2] or security verification [1]. In this context, studying quantitative extensions of these models is naturally relevant. Tree structures are also central in XML, and homomorphic transformations on trees allow us to modify the codes while preserving the hierarchical structure. Moreover, the HOM-problem plays a role in the context of term rewriting [14]: For
a term rewrite system, the set of normal forms (i.e., terms to which no rule can be applied) can be expressed as the complement of a homomorphic image; a better understanding of these images can help generalize known results in this field.

2 Preliminaries

We denote the set of nonnegative integers by \( \mathbb{N} \). For \( i, j \in \mathbb{N} \) we let \([i, j] = \{ k \in \mathbb{N} | i \leq k \leq j \}\) and \([j] = [1, j]\). Let \( Z \) be an arbitrary set. The cardinality of \( Z \) is denoted by \(|Z|\), and the set of words over \( Z \) (i.e., the set of ordered finite sequences of elements of \( Z \)) is denoted by \( Z^* \).

Trees, Substitutions, and Contexts

A ranked alphabet \((\Sigma, \text{rk})\) consists of a finite set \(\Sigma\) and a mapping \(\text{rk}: \Sigma \rightarrow \mathbb{N}\) that assigns a rank to each symbol of \(\Sigma\). If there is no risk of confusion, then we denote the ranked alphabet \((\Sigma, \text{rk})\) by \(\Sigma\) alone. We write \(\sigma^{(k)}\) to indicate that \(\text{rk}(\sigma) = k\). Moreover, for every \(k \in \mathbb{N}\) we let \(\Sigma_k = \text{rk}^{-1}(k)\) and \(\text{rk}(\Sigma) = \max \{ k \in \mathbb{N} | \Sigma_k \neq \emptyset \}\) be the maximal rank of symbols of \(\Sigma\). Let \(X = \{ x_i | i \in \mathbb{N} \}\) be a countable set of (formal) variables. For every \(n \in \mathbb{N}\), we let \(X_n = \{ x_i | i \in [n] \}\). Given a ranked alphabet \(\Sigma\) and a set \(Z\), the set \(T_\Sigma(Z)\) of \(\Sigma\)-trees indexed by \(Z\) is the smallest set such that \(Z \subseteq T_\Sigma(Z)\) and \((\sigma(t_1), \ldots, t_k) \in T_\Sigma(Z)\) for every \(k \in \mathbb{N}\), \(\sigma \in \Sigma_k\), and \(t_1, \ldots, t_k \in T_\Sigma(Z)\). We abbreviate \(T_\Sigma(\emptyset)\) simply by \(T_\Sigma\), and any subset \(L \subseteq T_\Sigma\) is called a tree language.

Let \(\Sigma\) be a ranked alphabet, \(Z\) a set, and consider a tree \(t \in T_\Sigma(Z)\). The set \(\text{pos}(t)\) of positions of \(t\) is defined by \(\text{pos}(z) = \{ \epsilon \}\) for all \(z \in Z\) and by \(\text{pos}(\sigma(t_1, \ldots, t_k)) = \{ \epsilon \} \cup \{ i w | i \in [k], w \in \text{pos}(t_i) \}\) for all \(k \in \mathbb{N}\), \(\sigma \in \Sigma_k\), and \(t_1, \ldots, t_k \in T_\Sigma(Z)\). With their help, we define the size ‘\(\text{size}(t)\)’ and height ‘\(\text{ht}(t)\)’ of \(t\) as \(\text{size}(t) = |\text{pos}(t)|\) and \(\text{ht}(t) = \max_{w \in \text{pos}(t)} |w|\). Positions are partially ordered by the standard prefix order \(\preceq\) on \(\text{rk}(\Sigma)^*\), and they are totally ordered by the ascending lexicographic order \(\leq\) on \(\text{rk}(\Sigma)^*\), in which prefixes are larger; i.e., \(\epsilon\) is the largest element. More precisely, for \(v, w \in \text{pos}(t)\) if there exists \(u \in \text{rk}(\Sigma)^*\) with \(vu = w\), then we write \(v \preceq w\), call \(v\) a prefix of \(w\), and let \(v^{-1}w = w\) because \(u\) is uniquely determined if it exists. Provided that \(u = u_1 \cdots u_n\) with \(u_1, \ldots, u_n \in \text{rk}(\Sigma)\) we also define the path \([v, \ldots, w]\) from \(v\) to \(w\) as the sequence \((v, u_1 v, u_1 u_2 v, \ldots, w)\) of positions. Any two positions that are \(\leq\)-incomparable are called parallel.

Given \(t, t' \in T_\Sigma(Z)\) and \(w \in \text{pos}(t)\), the label \(t(w)\) of \(t\) at \(w\), the subtree \(t|w\) of \(t\) at \(w\), and the substitution \(t[t']_w\) of \(t'\) into \(t\) at \(w\) are defined by \(z(\epsilon) = z = z[t']_\epsilon = t'\) for all \(z \in Z\) and by \(t(\epsilon) = \sigma, t(iw') = t_i(w')\), \(t|w = t_i|w\), \(t[t']_w = t'|w\), and \(t[t']_w = \sigma(t_1, \ldots, t_{i-1}, t_i(t'[w], t_{i+1}, \ldots, t_k))\) for all trees \(t = \sigma(t_1, \ldots, t_k)\) with \(k \in \mathbb{N}\), \(\sigma \in \Sigma_k\), \(t_1, \ldots, t_k \in T_\Sigma(Z)\), and \(i \in [k]\), and all \(w' \in \text{pos}(t_i)\). For all sets \(S \subseteq \Sigma \cup Z\) of symbols, we let \(\text{pos}_S(t) = \{ \epsilon \} \cup \{ i w | i \in [k], w \in \text{pos}(t_i) \}\) for all \(S \subseteq \Sigma \cup Z\). The set of variables occurring in \(t\) is \(\text{var}(t) = \{ x \in X | \text{pos}(x) \neq \emptyset \}\). Finally, consider \(\sigma \in \Sigma\) and a mapping \(\theta: X_\Sigma \rightarrow T_\Sigma(Z)_\Sigma\). Then by substitution, \(\sigma\) induces a mapping \(\theta: T_\Sigma(Z) \rightarrow T_\Sigma(Z)_\Sigma\) defined by \(\theta(x) = \theta(x)\) for every \(x \in X_\Sigma\), \(\theta(z) = z\) for every \(z \in Z \setminus X_\Sigma\), and \(\theta(\sigma(t_1, \ldots, t_k)) = \sigma(\theta(t_1), \ldots, \theta(t_k))\) for all \(k \in \mathbb{N}\), \(\sigma \in \Sigma_k\), and \(t_1, \ldots, t_k \in T_\Sigma(Z)\). For \(t \in T_\Sigma(Z)\), we denote \(\theta(t)\) by \(\theta\) or, more commonly, by \(t[x_1 \leftarrow \theta(x_1), \ldots, x_n \leftarrow \theta(x_n)]\).

Let \(\square \notin \Sigma\). A context is a tree \(C \in T_\Sigma(\square)\) with \(\text{pos}_C(\Sigma) \neq \emptyset\). More specifically, we call \(C\) an \(n\)-context if \(n = |\text{pos}_C(\Sigma)|\). For an \(n\)-context \(C\) and \(t_1, \ldots, t_n \in T_\Sigma\), we define the substitution \(C[t_1, \ldots, t_n]\) as follows. Let \(\text{pos}_C(\Sigma) = \{ w_1, \ldots, w_n \}\) be the occurrences of \(\square\) in \(C\) in lexicographic order \(w_1 \prec \cdots \prec w_n\). Then we let \(C[t_1, \ldots, t_n] = C[t_1]_{w_1} \cdots [t_n]_{w_n}\).
Tree Homomorphisms and Weighted Tree Grammars

Given ranked alphabets $\Sigma$ and $\Gamma$, let $h': \Sigma \to T_\Gamma(X)$ be a mapping with $h'(\sigma) \in T_\Gamma(X_k)$ for all $k \in \mathbb{N}$ and $\sigma \in \Sigma_k$. We extend $h'$ to $h: T_\Sigma \to T_\Gamma$ by $h(\alpha) = h'(\alpha) \in T_\Gamma(X_0) = T_\Gamma$ for all $\alpha \in \Sigma_\alpha$ and $h(\sigma(t_1, \ldots, t_k)) = h'(\sigma)[x_1 \leftarrow h(t_1), \ldots, x_k \leftarrow h(t_k)]$ for all $k \in \mathbb{N}$, $\sigma \in \Sigma_k$, and $t_1, \ldots, t_k \in T_\Sigma$. The mapping $h$ is called the tree homomorphism induced by $h'$, and we identify $h'$ and its induced tree homomorphism $h$. For the complexity analysis of our decision procedure, we define the size of $h$ as $\text{size}(h) = \sum_{\sigma \in \Sigma} \text{size}(h(\sigma))$. We call $h$ nonerasing (respectively, nondeleting) if $h'(\sigma) \notin X$ (respectively, var($h'(\sigma)$) $= X_k$) for all $k \in \mathbb{N}$ and $\sigma \in \Sigma_k$. In this contribution, we will only consider nonerasing and nondeleting tree homomorphisms $h: T_\Sigma \to T_\Gamma$, which are therefore input finitary; i.e., the preimage $h^{-1}(u)$ is finite for every $u \in T_\Gamma$ since $|t| \leq |u|$ for every $t \in h^{-1}(u)$. Any mapping $A: T_\Sigma \to \mathbb{N}$ is called non-weighted tree language, and we define the weighted tree language $h_A: T_\Gamma \to \mathbb{N}$ for every $u \in T_\Gamma$ by $h_A(u) = \sum_{q \in h^{-1}(u)} A(t)$ and call it the image of $A$ under $h$. This definition relies on the tree homomorphism to be input-finitary; otherwise the defining sum is not finite, so the value $h_A(u)$ is not necessarily well-defined.

A weighted tree grammar with equality constraints (WTGc) [20] is a tuple $(Q, \Sigma, F, P, w_t)$ in which $Q$ is a finite set of states, $\Sigma$ is a ranked alphabet of input symbols, $F: Q \to \mathbb{N}$ assigns a final weight to every state, $P$ is a finite set of productions of the form $(\ell, q, E)$ with $\ell \in T_\Sigma(Q) \setminus Q$, $q \in Q$, and finite subset $E \subseteq N^* \times N^*$, and $w_t: P \to \mathbb{N}$ assigns a weight to every production. A production $p = (\ell, q, E) \in P$ is usually written $p = \ell \overset{E}{\rightarrow} q$ or $p = \ell \overset{E}{\rightarrow}(q,w_t(p)) q$, and the tree $\ell$ is called its left-hand side, $q$ is its target state, and $E$ are its equality constraints, respectively. Equality constraints $(v,v') \in E$ are also written as $v = v'$. A state $q \in Q$ is final if $F(q) \neq 0$.

Next, we recall the derivation semantics of WTGc from [20]. Let $(v,v') \in N^* \times N^*$ be an equality constraint and $t \in T_\Sigma$. The tree $t$ satisfies $(v,v')$ if and only if $v,v' \in \text{pos}(t)$ and $t|_w = t|_{v'}$, and for a finite set $C \subseteq N^* \times N^*$ of equality constraints, we write $t \models C$ if $t$ satisfies all $(v,v') \in C$. Let $G = (Q, \Sigma, F, P, w_t)$ be a WTGc. A sentential form (for $G$) is a tree $\xi \in T_\Sigma(Q)$. Given an input tree $t \in T_\Sigma$, sentential forms $\xi, \zeta \in T_\Sigma(Q), A$ production $p = \ell \overset{E}{\rightarrow} q \in P$, and a position $w \in \text{pos}(\xi)$, we write $\xi \overset{p,w}{\Rightarrow}_{G,t} \zeta$ if $\xi|_w = \ell$, $\zeta = \xi[\ell|_w]$, and $t|_w = E$; i.e., the equality constraints $E$ are fulfilled on $t|_w$. A sequence $d = (p_1, w_1) \cdots (p_m, w_m) \in (P \times N^*)^*$ is a derivation (of $G$) for $t$ if there exist $\xi_0, \ldots, \xi_n \in T_\Sigma(Q)$ such that $\xi_0 = t$ and $\xi_{i-1} \overset{p_i,w_i}{\Rightarrow}_{G,t} \xi_i$ for all $i \in [n]$. We call $d$ left-most if additionally $w_1 < w_2 < \cdots < w_m$. Note that the sentential forms $\xi_0, \ldots, \xi_n$ are uniquely determined if they exist, and for any derivation $d$ for $t$ there exists a unique permutation of $d$ that is a left-most derivation for $t$. We call $d$ complete if $\xi_n \in Q$, and in this case we also call it a derivation to $\xi_n$. The set of all complete left-most derivations for $t$ to $q \in Q$ is denoted by $D^\Sigma_G(t)$. A complete derivation to some final state is called accepting. If for every $p \in P$, there exists a tree $t \in T_\Sigma$, a final state $q$, and a derivation $(p_1, w_1) \cdots (p_m, w_m) \in D^\Sigma_G(t)$ such that $F(q) \cdot \prod_{i=1}^{m} w_t(p_i) \neq 0$ and $p \in \{p_1, \ldots, p_m\}$; i.e., if every production is used in an accepting derivation with nonzero weights, then $G$ is trim.

Let $d = (p_1, w_1) \cdots (p_n, w_n) \in D^\Sigma_G(t)$ for some $t \in T_\Sigma$ and $i \in [n]$. Moreover, let $\{j_1, \ldots, j_k\}$ be the set $\{j \in [n] \mid w_j \leq w_i\}$ with the indices $j_1 < \cdots < j_k$ of those positions of which $w_i$ is a prefix. We refer to $(p_{j_1}, w_i^{-1}w_{j_1}), \ldots, (p_{j_k}, w_i^{-1}w_{j_k})$ as the derivation for $t|_{w_i}$ incorporated in $d$. Conversely, for $w \in N^*$ we abbreviate the derivation $(p_1, w_1w_1) \cdots (p_n, w_nw_n)$ by $w_d$.

The weight of a derivation $d = (p_1, w_1) \cdots (p_n, w_n)$ is defined as $w_tG(d) = \prod_{i=1}^{m} w_t(p_i)$. The weighted tree language generated by $G$, written $[G]: T_\Sigma \to \mathbb{N}$, is defined for all $t \in T_\Sigma$ by $[G](t) = \sum_{q \in Q, d \in D^\Sigma_G(t)} F(q) \cdot w_tG(d)$. For $t \in T_\Sigma$ and $q \in Q$, we will often use the
value \( \text{wt}_G^q(t) \) defined as \( \text{wt}_G^q(t) = \sum_{d \in D_G^q(t)} \text{wt}_G(d) \). Using distributivity, \( \text{wt}_G^q(t) \) then simplifies to \( \text{wt}_G^q(t) = \sum_{q \in Q} F(q) \cdot \text{wt}_G^q(t) \). We call two WTGc equivalent if they generate the same weighted tree language.

We call a WTGc \( (Q, \Sigma, F, P, wt) \) a weighted tree grammar (WTG) if \( E = \emptyset \) for every production \( \ell \xrightarrow{E} q \in P \); i.e., no production utilizes equality constraints. Instead of \( \ell \xrightarrow{\emptyset} q \) we also simply write \( \ell \rightarrow q \). Moreover, we call a WTGc a weighted tree automaton with equality constraints (WTAc) if \( \text{pos}_E(\ell) = \{\varepsilon\} \) for every production \( \ell \xrightarrow{E} q \in P \), and a weighted tree automaton (WTA) if it is both a WTG and a WTAc. The classes of WTGc and WTAc are equally expressive, and they are strictly more expressive than the class of WTA [20]. We call a weighted tree language regular if it is generated by a WTA and constraint-regular if it is generated by a WTGc.

Productions with weight 0 are obviously useless, so we may assume that \( \text{wt}(p) \neq 0 \) for every production \( p \). Finally, we define the size of a WTGc as follows.

\textbf{Definition 1.} Let \( G = (Q, \Sigma, F, P, wt) \) be a WTGc and \( p = \ell \xrightarrow{E} q \in P \) be a production. We define the height of \( p \) as \( \text{ht}(p) = \text{ht}(\ell) \) and its size as \( \text{size}(p) = \text{size}(\ell) \), the height of \( P \) as \( \text{ht}(P) = \max_{p \in P} \text{ht}(p) \) and its size as \( \text{size}(P) = \sum_{p \in P} \text{size}(p) \), and finally the height of \( G \) as \( \text{ht}(G) = |Q| \cdot \text{ht}(P) \) and its size as \( \text{size}(G) = |Q| + \text{size}(P) \).

It is known [20] that WTGc of a particular shape can represent homomorphic images of regular weighted tree languages. This subclass of WTGc will be central in our work.

\textbf{Definition 2.} A WTGc \( (Q, \Sigma, F, P, wt) \) is classic if every production \( p = \ell \xrightarrow{E} q \in P \) satisfies \( E \subseteq \text{pos}_Q(\ell)^2 \); i.e., all equality constraints point to the \( Q \)-labeled positions of its left-hand side. Without loss of generality, we can assume that every set \( E \) of equality constraints is reflexive, symmetric, and transitive, that is, an equivalence relation on a subset \( D \subseteq \text{pos}_Q(\ell) \), so not all occurrences of states need to be constrained.

A classic WTGc is eq-restricted if it has a so-called sink state \( \perp \in Q \setminus F \) such that (i) \( \sigma(\perp, \ldots, \perp) \rightarrow_1 \perp \) belongs to \( P \) for all \( \sigma \in \Sigma \), and no other productions target \( \perp \), and (ii) for every production \( \ell \xrightarrow{E} q \) with \( q \neq \perp \), if \( \text{pos}_Q(\ell) = \{p_1, \ldots, p_n\} \) and \( q_i = \ell(p_i) \) for \( i \in [n] \), the following conditions hold:

1. For each \( i \in [n] \), the set \( \{q_j \mid p_j \in [p_i]_{\equiv E} \} \setminus \{\perp\} \) is a singleton.
2. There exists exactly one \( p_j \in [p_i]_{\equiv E} \) such that \( q_j \neq \perp \).

In other words, an eq-restricted WTGc \( G \) has a designated nonfinal sink state \( \perp \in Q \) such that \( F(\perp) = 0 \) as well as \( p_\gamma = \gamma(\perp, \ldots, \perp) \rightarrow_1 \perp \in P \) and \( \text{wt}(p_\gamma) = 1 \) for every \( \gamma \in \Gamma \). In addition, every production \( p = \ell \xrightarrow{E} q \in P \) satisfies the following two properties. First, \( E \subseteq \text{pos}_Q(\ell)^2 \); i.e., all equality constraints point to the \( Q \)-labeled positions of its left-hand side. Second, \( \ell(v) = \perp \) and \( \ell(w) \neq \perp \) for every \( v \in [w']_E \setminus \{w\} \) and \( w' \) occurring in \( E \), where \( w = \min_{w' \in \text{pos}_Q^E} w' \); i.e., all but the lexicographically least position in each equivalence class of \( E \) are guarded by state \( \perp \). Essentially, an eq-restricted WTGc \( G \) performs its checks (and charges weights) exclusively on the lexicographically least occurrences of equality-constrained subtrees. All the other subtrees, which by means of the constraint are forced to coincide with another subtree, are simply ignored by the WTGc, which formally means that they are processed in the designated sink state \( \perp \). In the following, we will use \( \perp \) to denote such a sink state, and write \( Q \cup \{\perp\} \) to explicitly indicate its presence.

To simplify our terminology, we will refer to eq-restricted WTGc simply as WTGh.

\textbf{Theorem 3 (see [20, Theorem 5])}. Let \( G = (Q, \Sigma, F, P, wt) \) be a trim WTA and \( h : T_\Sigma \rightarrow T_\Gamma \) be a nondeleting and nonerasing tree homomorphism. Then there exists a trim WTGh \( G' \) with \( \text{wt}(G') = h[\text{wt}(G)] \). Moreover, \( \text{size}(G') \in \mathcal{O}(\text{size}(G) \cdot \text{size}(h)) \) and \( \text{ht}(G') \in \mathcal{O}(\text{ht}(h)) \).
Example 4. Let $G = (Q ∪ \{⊥\}, Γ, F, P, wt)$ with $Q = \{q, q_f\}$, $Γ = \{α(0), γ(1), δ(2)\}$, $F(q) = F(⊥) = 0$ and $F(q_f) = 1$, and the following set $P$ of productions.

$$\begin{align*}
\{α \rightarrow_1 q, γ(q) \rightarrow_2 q, δ(q, γ(⊥), q) \stackrel{1=21}{\rightarrow_1} q_f, α \rightarrow_1 ⊥, γ(⊥) \rightarrow_1 ⊥, δ(⊥, ⊥, ⊥) \rightarrow_1 ⊥\}
\end{align*}$$

The WTGc $G$ is a WTGh. It generates the homomorphic image $[G] = h_A$ for the tree homomorphism $h$ induced by the mapping $α \mapsto α$, $γ \mapsto γ(x_1)$, and $σ \mapsto δ(x_2, γ(x_2), x_1)$ applied to the regular weighted tree language $A$: $T_Σ \rightarrow N$ given by $A(t) = 2^{\text{pos}_t(0)}$ for every $t \in T_Σ$ with $Σ = \{α(0), γ(1), δ(2)\}$. The weighted tree language $[G]$ is itself not regular because its support is clearly not a regular tree language.

The restrictions in the definition of a WTGh allow us to trim it effectively.

Lemma 5. Let $G = (Q ∪ \{⊥\}, Γ, F, P, wt)$ be a WTGh. An equivalent, trim WTGh $G'$ can be constructed in polynomial time.

Proof. First, recall that we may assume $wt(p) \neq 0$ for every $p \in P$ because $wt_G(d) = 0$ for every derivation $d$ of $G$ that contains a production $p$ with $wt(p) = 0$. For the proof, we employ a simple reachability algorithm. For every $w \in N$ and $U \subseteq Q$, let

$$Q_0 = \emptyset \quad Q_{n+1} = Q_n \cup \bigcup_{(t \rightarrow q) \in P} \{q\} \quad \Pi_U = \bigcup_{(t \rightarrow q) \in P} \{\{q, q\}' \in U^2 \mid \text{pos}_q(ℓ) \neq \emptyset\}.$$ 

Since $Q$ is finite, there exists $N$ with $Q_N = Q_{N+1}$. Let $Q' = Q_N$. A straightforward proof shows that $q ∈ Q'$ if and only if for some $t \in T_Σ$ there exists $d ∈ D_G^n(t)$ with $wt_G(d) \neq 0$. To ensure the reachability of a final state, we let $δ$ be the smallest reflexive and transitive relation on $Q'$ that contains $Π_∅$. Then $P' = \{t \xrightarrow{E} q \in P \mid q \in Q', 3q'q ∈ Q' : F(qt) \neq 0, q \triangleq q'\}$, and the desired WTGh is simply $G' = (Q ∪ \{⊥\}, Γ, F, P', wt |_{P'})$.

3 Substitutions in the Presence of Equality Constraints

This short section recalls from [20] some definitions together with a pumping lemma for WTGh, which will be essential for deciding the integer-weighted HOM-problem. First, we need to refine the substitution of trees such that it complies with existing constraints.

Definition 6 (see [20] and cf. [15]). Let $G = (Q ∪ \{⊥\}, Γ, F, P, wt)$ be a WTGh. Moreover, let $q, q' ∈ Q$, $t, t' ∈ T_Σ$, and $d ∈ D_G^n(t)$ as well as $d' ∈ D_G^n(t')$ such that $q \neq ⊥ \neq q'$ and $d = d(p, ε)$ uses $p = c[q_1, ..., q_k]$ $E$ $q ∈ P$ as its final production. For every $i ∈ [k]$ let $w_i = \text{pos}_{q_i}(c)$ and $d_i$ be the unique leftmost derivation for $t_i = t|_{\text{pos}_{q_i}(c)}$ incorporated in $d$. Finally, for every $w ∈ T_Σ$ let $d^w_0$ be the unique leftmost derivation for $u \rightarrow ⊥$. For every $w ∈ \text{pos}(t)$ at which the production used in $d$ targets $q'$, we recursively define the derivation substitution $d[d']_w$ of $d'$ into $d$ at $w$ and the resulting tree $t[t']^d_w$ as follows. If $w = ε$, then $d[d']_w = d'$ and $t[t']^d_w = t'$. Otherwise $w = w_jw$ for some $j ∈ [k]$ and we have

$$d[d']_w = d'_1 \cdots d'_k(p, ε) \quad \text{and} \quad t[t']^d_w = c[t'_1, ..., t'_k],$$

where for each $i ∈ [k]$ we have

- if $i = j$ (i.e., $w_i$ is a prefix of $w$), then $d'_i = w_i |\overline{d'}_w$ and $t'_i = t_i |\overline{d'}_w$,
- if $q_i = ⊥$ and $w_i ∈ \overline{w_j}$ (i.e., it is a position that is equality restricted to $w_j$), then $d'_i = w_i |\overline{d'}_w$ and $t'_i = u$ with $u = t_j |\overline{d'}_w$,
- otherwise $d'_i = w_i |\overline{d'}_w$ and $t'_i = t_i$ (i.e., derivation and tree remain unchanged).

It is straightforward to verify that $d[d']_w$ is a complete leftmost derivation for $t[t']^d_w$ to $q$. 
Example 7. Consider the WTGh $G$ of Example 4 and the following tree $t$ it generates into which we want to substitute the tree $t' = \gamma(\alpha)$ at position $w = 11$.

\[
\begin{array}{c}
\gamma \gamma \\
\alpha \gamma \\
\alpha
\end{array}
\quad
\begin{array}{c}
\gamma \\
\gamma \\
\alpha
\end{array}
\]

We consider the following complete left-most derivation for $t$ to $q_f$.

$$d = (\alpha \rightarrow q, 11) (\gamma(q) \rightarrow q, 1) (\alpha \rightarrow \bot, 211) (\gamma(\bot) \rightarrow \bot, 21)$$

$$\left( \begin{array}{c}
\alpha \rightarrow q, 31 \\
\gamma(q) \rightarrow q, 3
\end{array} \right) \left( \begin{array}{c}
\delta(q, \gamma(\bot), q) \overset{1=21}{\rightarrow} q_f, \varepsilon
\end{array} \right)$$

Moreover, let $d' = (\alpha \rightarrow q, 1) (\gamma(q) \rightarrow q, \varepsilon)$ and $d''_\bot = (\alpha \rightarrow \bot, 1) (\gamma(\bot) \rightarrow \bot, \varepsilon)$.

With the notation of Definition 6, in the first step we have $v_1 = 1$, $v_2 = 21$, $v_3 = 3$, $d_1 = d_3 = d''$, $d_2 = d''_\bot$, and $w = v_1^{-1} w = 1$. Respecting the only constraint $1 = 21$, we set $d'_1 = d_1$, $d'_2 = d_2$, $d'_3 = d_3$ and $\tilde{w} = v^{-1}_1 w = 1$. Eventually, $d'_4 = (\alpha \rightarrow q, 11)(\gamma(q) \rightarrow q, 1)(\gamma(q) \rightarrow q, \varepsilon)$ and $d''_\bot = (\alpha \rightarrow \bot, 11)(\gamma(\bot) \rightarrow \bot, 1)(\gamma(\bot) \rightarrow \bot, \varepsilon)$.

Hence, we obtain the following derivation $d[d''_\bot]_{11}$ for our new tree $t[t']_{11}$.

$$d[d''_\bot]_{11} = (\alpha \rightarrow q, 11) (\gamma(q) \rightarrow q, 11) (\gamma(q) \rightarrow q, 1) (\alpha \rightarrow \bot, 2111) (\gamma(\bot) \rightarrow \bot, 21)$$

$$\left( \begin{array}{c}
\alpha \rightarrow q, 31 \\
\gamma(q) \rightarrow q, 3
\end{array} \right) \left( \begin{array}{c}
\delta(q, \gamma(\bot), q) \overset{1=21}{\rightarrow} q_f, \varepsilon
\end{array} \right)$$

Although $t[31] = \alpha$ also coincides with the subtree $t[11] = \alpha$ we replaced, these two subtrees are not equality-constrained, so the simultaneous substitution does not affect $t[31]$.

The substitution of Definition 6 allows us to prove a pumping lemma for the class of WTGh: If $d$ is an accepting derivation of a WTGh $G = (Q \cup \{ \bot \}, \Sigma, F, P, \in)$ for a tree $t$ with $\text{ht}(t) > \text{ht}(G)$, then there exist at least $|Q \setminus \{ \bot \}| + 1$ positions $w_i > \cdots > w_{|Q|+1}$ in $t$ at which $d$ applies productions with non-sink target states. By the pigeonhole principle, there thus exist two positions $w_i > w_j$ in $t$ at which $d$ applies productions with the same non-sink target state. Employing the substitution we just defined, we can substitute $t[w_j]$ into $w_i$ and obtain a derivation of $G$ for $t[t[w_j]]_{w_i}$. This process can be repeated to obtain an infinite sequence of trees strictly increasing in size. Formally, the following lemma was proved in [20].

Lemma 8 ([20, Lemma 4]). Let $G = (Q \cup \{ \bot \}, \Sigma, F, P, \in)$ be a WTGh. Consider some tree $t \in T_G$ and non-sink state $q \in Q \setminus \{ \bot \}$ such that $\text{ht}(t) > \text{ht}(G)$ and $D_G^i(t) \neq \emptyset$. Then there are infinitely many pairwise distinct trees $t_0, t_1, \ldots$ such that $D_G^i(t_i) \neq \emptyset$ for all $i \in \mathbb{N}$.

Example 9. Recall the WTGh $G$ of Example 4. We have $\text{ht}(P) = 2$ and $\text{ht}(G) = 4$, but for simplicity, we choose the smaller tree $t = \delta(\gamma(\alpha), \gamma(\gamma(\alpha)), \gamma(\alpha))$, which we also considered in Example 7, since it also allows pumping. The derivation $d$ presented in Example 7 for $t$ applies the productions $(\alpha \rightarrow q)$ at $11$ and $\gamma(q) \rightarrow q$ at $1$, so we substitute $t[1] = \gamma(\alpha)$ at $11$ to obtain $t[\gamma(\alpha)]_{11}$. In fact, this is exactly the substitution we illustrated in Example 7.
4 The Decision Procedure

Let us now turn to the N-weighted version of the HOM-problem. In the following, we show that the regularity of the homomorphic image of a regular N-weighted tree language is decidable in polynomial time. More precisely, we prove the following theorem.

**Theorem 10.** The weighted HOM-problem over N is polynomial; i.e. for fixed ranked alphabets \( \Gamma \) and \( \Sigma \), given a trim WTA \( A \) over \( \Gamma \), and a nondeleting, nonerasing tree homomorphism \( h : T_\Gamma \rightarrow T_\Sigma \), it is decidable in polynomial time whether \( h_{[A]} \) is regular.

In the beginning, the proof of Theorem 10 resembles the unweighted case [15]: Given a regular weighted tree language \( A \) (represented by a trim WTA) and a tree homomorphism \( h \), we first construct a trim WTGh \( G \) for its image \([G] = h_A\) applying Theorem 3. We then show that \([G]\) is regular if and only if the equality constraints used in \( G \) only act on subtrees of height at most \( \text{ht}(G) \). In other words, if there exists a production \( \ell \overset{E}{\rightarrow} q \) in \( G \) such that for some equality constraint \( (u, v) \in E \) with non-sink state \( q = \ell(u) \) there exists a tree \( t \in T_\Sigma \) with \( \text{ht}(t) > \text{ht}(G) \) and \( D^G_G(t) \neq \emptyset \), then \([G]\) is not regular, and if no such production exists, then \([G]\) is regular. There are thus three parts to our proof. First, we show that the existence of such a production is decidable in polynomial time. Then we show that \([G]\) is regular if no such production exists. Finally, we show that \([G]\) is not regular if such a production exists. The latter part employs Ramsey’s theorem [24] and is the most significant technical contribution in our paper. For convenience, we attach a name to the property described here.

**Definition 11.** Let \( G = (Q \cup \{\bot\}, \Sigma, F, P, \text{wt}) \) be a trim WTGh. We say that \( G \) has the large duplication property if there exist a production \( \ell \overset{E}{\rightarrow} q \in \Sigma \), an equality constraint \( (u, v) \in E \) with \( \ell(u) \neq \bot = \ell(v) \), and a tree \( t \in T_\Sigma \) such that \( \text{ht}(t) > \text{ht}(G) \) and \( D^G_G(t) \neq \emptyset \).

We start with the decidability of the large duplication property.

**Lemma 12.** Consider a fixed ranked alphabet \( \Sigma \). The following is decidable in polynomial time: Given a trim WTGh \( G \), does it satisfy the large duplication property?

**Proof.** Let \( G = (Q \cup \{\bot\}, \Sigma, F, P, \text{wt}) \) and construct the directed graph \( G = (Q, E) \) with edges \( E = \bigcup_{\ell \rightarrow_E q} \{q', q \mid q' \in Q, \text{pos}_q(\ell) \neq \emptyset\}. \) Clearly, the large duplication property is equivalent to the condition that there exists a production \( \ell \overset{E}{\rightarrow} q \in \Sigma \), an equality constraint \( (u, v) \in E \) with \( \ell(u) \neq \bot = \ell(v) \), and a cycle from \( q' \) to \( q' \) in \( G \) and a path from \( q' \) to \( \ell(u) \) in \( G \). This equivalent condition can be checked in polynomial time. The equivalence of the two statements is easy to establish. If the large duplication property holds, then the pumping lemma [20, Lemma 4] exhibits the required cycle and path. Conversely, if the cycle and path exist, then the pumping lemma [20, Lemma 4] can be used to derive arbitrarily tall trees for which a derivation exists. ▶

Next, we show that if a WTGh \( G \) does not satisfy the large duplication property, then its generated weighted tree language \([G]\) is regular. To this end, we construct the linearization of \( G \). The linearization of a WTGh \( G \) is a WTA that simulates all derivations of \( G \) which only ensure the equality of subtrees of height at most \( \text{ht}(G) \). For this, we replace every production \( \ell \overset{E}{\rightarrow} q \) in \( G \) by the collection of all productions \( \ell' \rightarrow q \) which can be obtained by instantiating \( E \), i.e., substituting each position constrained by \( E \) with a compatible tree of height at most \( \text{ht}(G) \) that satisfies \( E \). Note that positions in \( \ell \) that are unconstrained by \( E \) are unaffected by these substitutions. Formally, we define the linearization following [15, Definition 7.1].
Definition 13. Let \( G = (Q \cup \{\bot\}, \Sigma, F, P, \text{wt}) \) be a WTG. The linearization \( \text{lin}(G) \) of \( G \) is the WTG \( \text{lin}(G) = (Q \cup \{\bot\}, \Sigma, F, P_{\text{lin}}, \text{wt}_{\text{lin}}) \), where \( P_{\text{lin}} \) and \( \text{wt}_{\text{lin}} \) are defined as follows. For \( \ell' \in T_{\Sigma}(Q) \setminus Q \) and \( q \in Q \), we let \( \ell' \rightarrow q \) \( \in P_{\text{lin}} \) if and only if there exist a production \( \ell \rightarrow q \in P \), positions \( w_1, \ldots, w_k \in \text{pos}_{Q \cup \{\bot\}}(\ell) \), and trees \( t_1, \ldots, t_k \in T_{\Sigma} \) with

- \( \{w_1, \ldots, w_k\} = \bigcup_{w \in \text{pos}_{Q \cup \{\bot\}}(\ell)} [w]_{E} \); i.e., \( E \) constrains exactly the positions \( w_1, \ldots, w_k \),
- \( t_1 = t_j \) if \( (w_i, w_j) \in E \) for all \( i, j \in [k] \),
- \( \ell' = \ell[t_1]_{w_1} \cdots [t_k]_{w_k} \), and
- \( D_{\ell}(w_i)(t_i) \neq \emptyset \) and \( \text{ht}(t_i) \leq \text{ht}(G) \) for all \( i \in [k] \).

For every such production \( \ell' 
\rightarrow q \) we define \( \text{wt}_{\text{lin}}(\ell' \rightarrow q) \) as the sum over all weights

\[
\text{wt}(\ell \rightarrow q) \cdot \prod_{i \in [k]} \text{wt}(\ell')_{w_i}(t_i)
\]

for all \( \ell \rightarrow q \in P \), \( w_1, \ldots, w_k \in \text{pos}_{Q \cup \{\bot\}}(\ell) \), and \( t_1, \ldots, t_k \in T_{\Sigma} \) as above.

If a trim WTG \( G \) does not satisfy the large duplication property, then every equality constraint in every derivation of \( G \) only ensures the equality of subtrees of height at most \( \text{ht}(G) \). Thus, \( \text{lin}(G) \) and \( G \) generate the same weighted tree language \( [G] = [\text{lin}(G)] \), which is then regular because \( \text{lin}(G) \) is a WTG. Thus we summarize:

Proposition 14. Let \( G \) be a trim WTG and suppose that \( G \) does not satisfy the large duplication property. Then \( [G] \) is a regular weighted tree language.

Finally, we show that if a WTG \( G = (Q \cup \{\bot\}, \Sigma, F, P, \text{wt}) \) satisfies the large duplication property, then \( [G] \) is not regular. For this, we first show that if \( G \) satisfies the large duplication property, then we can decompose it into two WTG \( G_1 \) and \( G_2 \) such that \( [G] = [G_1] + [G_2] \) and at least one of \([G_1]\) and \([G_2]\) is not regular. To conclude the desired statement, we then show that the sum \([G] = [G_1] + [G_2]\) is also not regular. For the decomposition, consider the following idea. Assume that there exists a production \( p = (\ell \rightarrow q) \in P \) as in the large duplication property such that \( F(q) \neq \emptyset \). Then we create two copies \( G_1 \) and \( G_2 \) of \( G \) as follows. In \( G_1 \) we set all final weights to 0, add a new state \( f \) with final weight \( F(q) \), and add the new production \( (\ell \rightarrow f) \) with the same weight as \( p \). On the other hand, in \( G_2 \) we set the final weight of \( q \) to 0, add a new state \( f \) with final weight \( F(q) \), and for every production \( p' = (\ell' \rightarrow q) \in P \) except \( p \), we add the new production \( \ell' \rightarrow f \) to \( G_2 \) with the same weight as \( p' \). Then \( [G] = [G_1] + [G_2] \) because every derivation of \( G \) whose last production is \( p \) is now a derivation of \( G_1 \) to \( f \), and every other derivation is either directly a derivation of \( G_2 \) or, in case of other derivations to \( q \), is a derivation of \( G_2 \) to \( f \).

By our assumption on the production \( p = (\ell \rightarrow q) \), there exist a tall tree \( t \in T_{\Sigma} \) with \( \text{ht}(t) > \text{ht}(G) \) and a constraint \( (u, v) \in E \) with \( \ell(u) \neq \bot = \ell(v) \) and \( D_{\ell}(u)(t) \neq \emptyset \). Thus, every tree \( t' \) generated by \( G_1 \) satisfies \( t' \models t \), and by Lemma 8, there exist infinitely many pairwise distinct trees with a derivation to \( \ell(u) \). The support (i.e., set of nonzero weighted trees) of \([G_1]\) is therefore not a regular tree language. This implies that \([G_1]\) is not regular, as the support of every regular weighted tree language over \( \mathbb{N} \) is a regular tree language [12].

In general, we cannot expect that a production \( \ell \rightarrow q \) as in the large duplication property exists that already targets a final state. We therefore “grow” productions from the top, beginning with a production whose target state is final, by substituting \( Q \)-labeled positions with left-hand sides of other productions until we have “synthesized” a production which satisfies the large duplication property. We then construct \( G_1 \) by adding this newly formed
production as a production to a new state $f$. We construct $G_2$ simply to ensure that it simulates all derivations of $G$ that are not already accounted for by $G_1$. Formally, we show the following lemma.

**Lemma 15.** Let $G = (Q \cup \{\bot\}, \Sigma, F, P, w_t)$ be a trim $WTGh$ that satisfies the large duplication property. Then there exist two trim $WTGh$ $G_1 = (Q_1 \cup \{\bot\}, \Sigma, F_1, P_1, w_{t1})$ and $G_2 = (Q_2 \cup \{\bot\}, \Sigma, F_2, P_2, w_{t2})$ such that $[G] = [G_1] + [G_2]$ and for some $f \in Q_1$ we have

- $F_1(f) \neq 0$ and $F_1(q) = 0$ for all $q \in Q_1 \setminus \{f\}$, and
- there exists exactly one production $p_1 = (\ell_1 \xrightarrow{E} f) \in P_1$ with target state $f$, and for this production there exists $(u, v) \in E_\ell$ with $\ell_1(u) \neq \ell_1(v) = \bot$ and an infinite sequence of pairwise distinct trees $t_0, t_1, t_2, \ldots \in T_\Sigma$ such that $D_G^{\ell_1(u)}(t_i) \neq \emptyset$ for all $i \in \mathbb{N}$.

**Proof.** Let $p = (\ell \xrightarrow{E} q) \in P$ be a production as in the large duplication property. Since $G$ is trim, there exist a tree $t' \in T_\Sigma$, a final state $q_1 \in Q$ with $F(q_1) \neq 0$, a derivation $d = (p_1, w_1) \cdots (p_m, w_m) \in D_G^t(t')$, and $i \in [m]$ such that $p_i = p$. In other words, there is a derivation utilizing production $p$. We let $p_j = \ell_j \xrightarrow{E} q_j$ for every $j \in [m]$, and let $w_i, \ldots, w_m$ be the sequence of prefixes of $w_i$ among the positions $\{w_1, \ldots, w_m\}$ in strictly descending order with respect to the prefix order. In particular, we have $w_i = w_i$ and $w_1 = \varepsilon$.

For a position $w$ and a set $E'$ of constraints, we define $wE' = \{(w, w) \mid (u, v) \in E'\}$. We want to join the left-hand sides of the productions $p_1, \ldots, p_k$ to a new production $\ell_1[w_1, w_2, \ldots, \ell_1[w_k] \xrightarrow{E'} q_1$ with $E = \bigcup_{i \in [k]} w_i E_i$. However, we need to ensure that $w_i, \ldots, w_k$ do not occur in $E_i$. Therefore, we assume that $p, t', q_i, d$, and $i$ above are chosen such that $w_i$ is of minimal length among all possible choices. Then we see as follows that $w_1, \ldots, w_k$ do not occur in $E_i$.

Let $(u, v) \in E$ with $\ell(u) \neq \ell(v) = \bot$ and $t \in T_\Sigma$ with $h(t) > h(G)$ and $D_G^{\ell(u)}(t) \neq \emptyset$. Suppose there exists $j \in [k]$ such that $w_i$ occurs in $E_i$. Then there exists $(u', v') \in E_{ij+1}$ with $w_{ij} = w_{ij+1}, u'$. Then the tree $t'[w'_{ij} | w_{ij}]$ shows us that $p_{ij+1}$ is also a production as in the large duplication property, but $|w_{ij+1}| < |w_{ij}|$, so $w_i$ is not of minimal length.

We define $G_1 = (Q_1 \cup \{\bot\}, \Sigma, F_1, P_1, w_{t1})$ as follows. Let $f \notin Q \cup \{\bot\}$ be a new state.

We set $Q_1 = Q \cup \{f\}$, $F_1(f) = F(q_1)$, and $F_1(q') = 0$ for all $q' \in Q_1$. For the production $p_1 = (\ell_1[w_1, w_2, \ldots, \ell_1[w_k] \xrightarrow{E'} f)$ with $E = \bigcup_{i \in [k]} w_i E_i$, we let $P_1 = P \cup \{p_1\}$, $w_{t1}(p_1) = \prod_{i \in [k]} w_i(p_i)$, and $w_{t1}(p') = w(p')$ for all $p' \in P$. Then $G_1$ simulates all derivations of $G$ with productions $p_1, \ldots, p_k$ at the positions $w_1, \ldots, w_k$, respectively.

For the existence of the infinite sequence of trees, let $(u, v) \in E$ with $\ell(u) \neq \ell(v) = \bot$ and $t \in T_\Sigma$ with $h(t) > h(G)$ and $D_G^{\ell(u)}(t) \neq \emptyset$. By Lemma 8, there exists an infinite sequence $t_0, t_1, t_2, \ldots \in T_\Sigma$ of pairwise distinct trees with $D_G^{\ell(u)}(t_i) \neq \emptyset$ for all $i \in \mathbb{N}$. Since $D_G^{\ell(u)}(t_i) \subseteq D_G^{\ell(u)}(t_i)$ for all $i \in \mathbb{N}$, this is the desired sequence. We conclude the definition of $G_1$ by noting that $(w_i u, w_i v) \in E_\ell$ and that the left-hand side $\ell_i$ of $p_i$ satisfies $\ell_i(w_i u, w_i v) = \ell(u)$.

Next, we construct $G_2$ such that it simulates all remaining derivations of $G$ in the following sense. If $d$ is a derivation of $G$ to a state different from $q$, then it is a derivation of $G_2$ to that same state. If $d$ is a derivation of $G$ to $q$ but its last production is not $p_k$, then it is simulated by a derivation of $G_2$ to a new state $f$. If $d$ is a derivation of $G$ and its last production is $p_k$ but the production at $w_{i_{k-1}}$ is not $p_{i_{k-1}}$, then it again is simulated by a derivation of $G_2$ to $f$, and so on. To have a more compact definition for $G_2$, we use the symbol $\square$ to denote a tree of height 0 and a term $[\ell_{i_k}]_{w_{i_k}} \cdots [\ell_{i_{i_{k-1}}} \cdots [\ell]_{w_1}]$ for $j = k$ to be read as $[\ell]_{w_1}$. We let $f \notin Q \cup \{\bot\}$ be a new state and define $G_2 = (Q_2 \cup \{\bot\}, \Sigma, F_2, P_2, w_{t2})$ by $Q_2 = Q \cup \{f\}$, $F_2(q) = 0$, $F_2(f) = F(q)$, and $F_2(q') = F(q')$ for all $q' \in Q \setminus \{q\}$. 

51:10 Weighted HOM-Problem for Nonnegative Integers
For the set \( P_2 \) of productions, we let
\[
P_2 = P \cup \bigcup_{j \in [k]} \left\{ \square[\ell_{i_k}]w_{i_k} \cdots [\ell_{i_{j+1}}]w_{i_{j+1}}[\ell']w_{i_j} \xrightarrow{E_f} f \bigg| p' = (\ell' \xrightarrow{E'} q_i) \in P \setminus \{p_i\}, \right. \\
E_f = w_{i_j}E' \cup \bigcup_{j' = j+1}^k w_{i_{j'}}E_{i_{j'}} \bigg\}.
\]
For a production \( p_f = \square[\ell_{i_k}]w_{i_k} \cdots [\ell_{i_{j+1}}]w_{i_{j+1}}[\ell']w_{i_j} \xrightarrow{E_f} f \) constructed from \( p' \) as above we let \( \text{wt}(p_f) = \text{wt}(p') \cdot \prod_{j' = j+1}^k \text{wt}(p_{i_{j'}}) \) and for every \( p' \in P \) we let \( \text{wt}_2(p') = \text{wt}(p') \). Then we have \( \|G\|(t) = \|G_1\|(t) + \|G_2\|(t) \) for every \( t \in T_\Sigma \). Note that trimming \( G_1 \) and \( G_2 \) will not remove any of the newly added productions under the assumption that \( G \) is trim.

\[\textbf{Example 16.}\] We present an example for the decomposition in Lemma 15. Consider the trim WTGh \( G = (Q \cup \{\bot\}, \Sigma, P, F, \text{wt}) \) with \( Q = \{q_0, q, q_f\} \), \( \Sigma = \{\alpha^{(0)}, \gamma^{(1)}, \sigma^{(2)}, \gamma_1^{(1)} \} \), final weights \( F(q_f) = 1 \) and \( F(q_0) = F(\bot) = F(\bot) = 0 \), and the set \( P = P_L \cup P' \) defined by \( P' = \{ \alpha \rightarrow_{\bot} q_0, \gamma(q_0) \rightarrow_{\bot} q_0, \sigma(q_0, \bot) \xrightarrow{1_{\bot}} q, \gamma_1(\bot) \xrightarrow{\gamma} q, \gamma_2(q) \rightarrow_{\bot} q, \gamma(\gamma(q_0)) \rightarrow_{\bot} q_f \} \) and the usual productions targeting \( \bot \) in \( P_L \). Trees of the form \( \gamma(\gamma(\gamma(\alpha))) \cdots \) of arbitrary height are subject to the constraint \( 1 = 2 \), so \( G \) satisfies the large duplication property.

We consider \( t' \) as in Figure 1 and use its (unique) derivation in \( G \). Following the approach sketched above, we choose a new state \( f \) and define \( G_1 = (Q \cup \{f\} \cup \{\bot\}, \Sigma, F_1, P_1, \text{wt}_1) \), where \( F_1(f) = 1 \) and \( F_1(q) = 0 \) for every \( q \in Q \cup \{\bot\} \), and \( P_1 = P \cup \{p_i\} \) with the new production \( p_f \) depicted in Figure 1, which joins all the productions of \( G \) used to derive \( t' \), from the one evoking the large duplication property to the one targeting a final state. It remains to construct a WTGh \( G_2 \) such that \( \|G\| = \|G_1\| + \|G_2\| \). All productions of \( G \) still occur in \( G_2 \), but \( q_f \) is not final anymore. Instead, we add a state \( f \) with \( F(f) = F(q_f) = 1 \) and make sure that this state adopts all other accepting derivations that formerly led to \( q_f \).

For this, we handle first the derivations that coincide with the derivation for \( t' \) at the juncture positions \( \varepsilon \) and \( 1 \), but not at 2. This leads to the following new productions \( p_1' \) and \( p_2' \):
\[
p_1' = \sigma \xrightarrow{\gamma_1} q_0 \xrightarrow{\gamma_2} \sigma \xrightarrow{\gamma_1} \sigma \xrightarrow{\gamma_2} f \\
p_2' = \sigma \xrightarrow{\gamma_1} q_0 \xrightarrow{\gamma_2} \sigma \xrightarrow{\gamma_1} \sigma \xrightarrow{\gamma_2} f.
\]
Next we cover the derivations that differ from the derivation for \( t' \) at the position 1 but coincide with it at the root. This leads to the new productions:

\[
p_1^2 = \begin{array}{c}
\sigma \\
\gamma_2 \\
\end{array} \xrightleftharpoons[11=12]{4} f \\
q_0 \quad q_0 \\
\downarrow \quad \downarrow
\]

\[
p_2^2 = \begin{array}{c}
\sigma \\
\gamma_2 \\
\end{array} \xrightarrow{4} f \\
q_0 \quad q_0
\]

Apart from the production incorporated at the root of \( p_t \), no other production of \( G \) targets \( q_t \) directly, so no more productions are added to \( P_2 \).

Finally, we define the WTGH \( G_2 = (Q \cup \{f\} \cup \{\bot\}, \Sigma, F_2, P_2, \text{wt}_2) \) with \( F_2(f) = F(q_t) = 1 \), \( F_2(q_t) = F_2(q_0) = F_2(\bot) = 0 \), and \( P_2 = P \cup \{p_1^2, p_2^2\} \cup \{p_2^1, p_2^3\} \).

It remains to show that the existence of a decomposition \([G] = [G_1] + [G_2]\) as in Lemma 15 implies the non-regularity of \([G]\). For this, we employ the following idea. Consider a ranked alphabet \( \Sigma \) containing a letter \( \sigma \) of rank 2, a WTA \( G' = (Q, \Sigma, F, P, \text{wt}) \) over \( \Sigma \) (which exemplifies \( G_2 \)), and a sequence \( t_0, t_1, t_2, \ldots \in T_\Sigma \) of pairwise distinct trees. At this point, we assume that \( P \) contains all possible productions, but we may have \( \text{wt}(p) = 0 \) for \( p \in P \). Using the initial algebra semantics [12], we can find a matrix representation for the weights assigned by \( G' \) to trees of the form \( \sigma(t_i, t_j) \) as follows. We enumerate the states \( Q = \{q_1, \ldots, q_n\} \) and for every \( i \in \mathbb{N} \) define a (column) vector \( \nu_i \in \mathbb{N}^n \) by \( (\nu_i)_k = \text{wt}_{G'}(t_i) \) for \( k \in [n] \). Furthermore, we define a matrix \( N \in \mathbb{N}^{n \times n} \) by \( N_{kh} = \sum_{q \in Q} F(q) \cdot \text{wt}(\sigma(q_k, q_h) \rightarrow q) \) for \( k, h \in [n] \). Then \([G'](\sigma(t_i, t_j)) = \nu_i^T N \nu_j \) for all \( i, j \in \mathbb{N} \), where \( \nu_i^T \) is the transpose of \( \nu_i \).

We employ this matrix representation to show that the sum of \([G']\) and the (non-regular) characteristic function \( 1_L \) of the tree language \( L = \{\sigma(t_i, t_i) \mid i \in \mathbb{N}\} \) is not regular. We proceed by contradiction and assume that \([G'] + 1_L\) is regular. Thus we can find an analogous matrix representation using a matrix \( N' \) and vectors \( \nu'_i \) for \([G'] + 1_L\). Since the trees \( t_0, t_1, t_2, \ldots \) are pairwise distinct, we can write

\[
([G'] + 1_L)(\sigma(t_i, t_j)) = (\nu'_i)^T N' \nu'_j = [G'](\sigma(t_i, t_j)) + \delta_{ij} = \nu_i^T N \nu_j + \delta_{ij}
\]

for all \( i, j \in \mathbb{N} \), where \( \delta_{ij} \) denotes the Kronecker delta. The vectors \( \nu'_i \) and \( \nu_i \) contain nonnegative integers, so we may consider the concatenated vectors \( (\nu'_i, \nu_i) \) as vectors of \( \mathbb{Q}^m \) where \( m \in \mathbb{N} \) is the sum of number of states of \( G' \) and of the WTA we assumed recognizes \([G'] + 1_L\). Since \( \mathbb{Q}^m \) is a finite-dimensional \( \mathbb{Q}\)-vector space, the \( \mathbb{Q}\)-vector space spanned by the family \( ((\nu'_i, \nu_i))_{i \in \mathbb{N}} \) is also finite dimensional. We may thus select a finite generating set from \( ((\nu'_i, \nu_i))_{i \in \mathbb{N}} \). For simplicity, we assume that \( (\nu'_1, \nu_1), \ldots, (\nu'_K, \nu_K) \) form such a generating set. Thus there exist \( a_1, \ldots, a_K \in \mathbb{Q} \) with \( (\nu'_K, \nu_K) = \sum_{i \in [K]} a_i (\nu'_i, \nu_i) \). Applying the usual distributivity laws for matrix multiplication, we reach a contradiction as follows.

\[
([G'] + 1_L)(\sigma(t_{K+1}, t_{K+1})) = (\nu'_K)^T N' \nu'_{K+1} = \sum_{i \in [K]} a_i (\nu'_i)^T N' \nu'_{K+1} = \sum_{i \in [K]} a_i \nu_i^T N \nu_{K+1} + \nu_K^T N \nu_{K+1} = [G'](\sigma(t_{K+1}, t_{K+1}))
\]

For the general case, we do not want to assume that \([G_2]\) is regular, so we cannot assume to have a matrix representation as we had for \([G']\) above. In order to make our idea work, we identify a set of trees for which the behavior of \([G_1] + [G_2]\) resembles that of \([G'] + 1_L\); more precisely, we construct a context \( C \) and a sequence \( t_0, t_1, t_2, \ldots \) of pairwise distinct trees...
such that \((G_1 \cup G_2)(C(t_i, t_j)) = \nu_i^{(1)} N_{k}^{(2)} + \delta_{ij} \mu_k\) for all \(i, j \in \mathbb{N}\) and additionally, \(\mu_k > 0\) for all \(k \in \mathbb{N}\). This representation then allows us to perform linear algebra computations in order to prove that \([G_1] + [G_2]\) is non-regular. Unfortunately, working with a 2-context \(C\) may be insufficient if \(G_1\) uses constraints of the form \(\{v = v', v' = v''\}\), where more than two positions are constrained to be pairwise equivalent. Therefore, we have to consider more general n-contexts \(C\) and then identify a sequence of trees such that the equation above is satisfied on \(C(t_i, t_j, t_k, \ldots, t_l)\).

Isolating this desired sequence of trees is the most technically involved proof in our paper. We illustrate the effect of this selection in Example 19 below. Along the way, we will use the following version of Ramsey’s theorem [24]. For a set \(X\), we denote by \(\binom{X}{2}\) the set of all subsets of \(X\) of size 2.

\textbf{Theorem 17.} Let \(k \geq 1\) be an integer and \(f : \binom{\binom{[2]}{2}}{2} \to [k]\) a mapping. There exists an infinite subset \(E \subseteq \mathbb{N}\) such that \(f\left|_{\binom{\binom{[2]}{2}}{2}} \equiv i\) for some \(i \in [k]\).

\textbf{Lemma 18.} Let \(G = (Q \cup \{\bot\}, \Sigma, F, P, wt)\) be a trim WTHG. If \(G\) satisfies the large duplication property, then there exists an integer \(r \geq 2\), an r-context \(C \in T_{\Sigma}(\square)\), trees \((t_i)_{i \in \mathbb{N}} \subseteq T_{\Sigma}\), an integer \(m \in \mathbb{N}\), row vectors \((v_i^{(1)})_{n \in \mathbb{N}} \subseteq \mathbb{N}^m\), column vectors \((v_i^{(2)})_{n \in \mathbb{N}} \subseteq \mathbb{N}^m\), a matrix \(N \in \mathbb{N}^{m \times m}\), and weights \((\mu_i)_{i \in \mathbb{N}} \subseteq \mathbb{N}\) \{0\} with \([G](C(t_k, t_h, \ldots, t_h)) = v_i^{(1)} N v_i^{(2)} + \delta_{kh} \mu_k\) for all \(k, h \in \mathbb{N}\).

\textbf{Proof.} By Lemma 15 there exist two trim WTHG \(G_1 = (Q_1 \cup \{\bot\}, \Sigma, F_1, P_1, wt_1)\) and \(G_2 = (Q_2 \cup \{\bot\}, \Sigma, F_2, P_2, wt_2)\) with \([G_1](t) = [G_1](t) + [G_2](t)\) for all \(t \in T_{\Sigma}\). Additionally, there exists \(f \in Q_1\) with \(F_{1}(f) \neq 0\) and \(F_{1}(f) = 0\) for all \(q \in Q_1 \setminus \{f\}\) and there exists exactly one production \(p_1 = (q_i \xrightarrow{E_i} f) \in P_1\) whose target state is \(f\). Finally, for this production \(p_1\) there exists \((u^{(1)}, v^{(1)}) \in E_{\Sigma}\) with \(\ell_i(u^{(1)}) \neq \ell_i(v^{(1)})\) is \(\perp\) and an infinite sequence \(t_0, t_1, t_2, \ldots \in T_{\Sigma}\) of pairwise distinct trees with \(D_{G_1}^{(u^{(1))}}(t_i) \neq 0\) for all \(i \in \mathbb{N}\).

Let \(t \in T_{\Sigma}\) be such that \(D_{G_1}^{(1)}(t) \neq 0\), and let \(u_1^{(1)}, \ldots, u_r^{(1)}\) be an enumeration of all positions that are equality-constrained to \(u^{(1)}\) via \(E_{\Sigma}\), where we assume that \(u_1^{(1)} = u^{(1)}\). We define a context \(C = \ell_{[\text{pos}(C)]} u^{(1)}\). Then \([G_1](C(t_i, t_j, t_k, \ldots, t_l)) > 0\) if \(i = j\).

Let us establish some additional notations. Let \(k, h \in \mathbb{N}\) and assume there is \(q \in Q_2\) with \(F_2(q) \neq 0\) and \(d = (p_i, w_i) \cdots (p_m, w_m) \in D_{G_2}^{(d)}(C(t_k, t_h, \ldots, t_h))\). Let \(p_i = q_i \xrightarrow{E_i} q_i\) for every \(i \in [m]\), and for a set \(X \subseteq \text{pos}(C(t_k, t_h, \ldots, t_h))\), we let \(i_1 < \cdots < i_n\) be such that \(w_{i_1}, \ldots, w_{i_n}\) is an enumeration of \((w_{i_1}, \ldots, w_{i_n}) \cap X\); i.e., all positions in \(X\) to which \(d\) applies productions. We set \(d(x) = (p_{i_1}, w_{i_1}) \cdots (p_{i_n}, w_{i_n})\), \(wt_2(d(x)) = \prod_{j \in [n]} wt_2(p_{i_j})\), and \(D_{kk} = \{d' : \text{pos}(C) \mid \exists q' \in Q_2 : D_{G_2}(q') \neq 0, d' \in D_{G_2}^{(d)}(C(t_k, t_h, t_h, \ldots, t_h))\}\).

We now employ Ramsey’s theorem in the following way. For \(k, h \in \mathbb{N}\) with \(k < h\), we consider the mapping \(\{k, h\} \mapsto D_{kh}\). This mapping has a finite range as every \(D_{kh}\) is a set of finite words over the alphabet \(P_2 \times \text{pos}(C)\) of length at most \(\text{size}(C)\). Thus, by Ramsey’s theorem, we obtain a subsequence \((t_i)_{i \in \mathbb{N}}\) with \(D_{kh} = D_{<}\) for all \(k, h \in \mathbb{N}\) and some set \(D_{<}\). For simplicity, we assume \(D_{kh} = D_{<}\) for all \(k, h \in \mathbb{N}\) with \(k < h\). Similarly, we select a further subsequence and assume \(D_{kh} = D_{\geq}\) for all \(k, h \in \mathbb{N}\) with \(k > h\). Finally, the mapping \(k \mapsto D_{kh}\) also has a finite range, so by the pigeonhole principle, we may select a further subsequence and assume that \(D_{kh} = D_{=}\) for all \(k, h \in \mathbb{N}\) and some set \(D_{=}\). In the following, we show that \(D_{<} = D_{\geq} \subseteq D_{=}\).

For now, we assume \(D_{<} \neq \emptyset\), let \((p_1, w_1) \cdots (p_m, w_m) \in D_{<}\), and let \(p_i = q_i \xrightarrow{E_i} q_i\) for every \(i \in [m]\). Also, we define \(C_{kh} = C(t_k, t_h, t_h, \ldots, t_h)\), \(C_{k\bot} = C(t_k, t_h, t_h, \ldots, t_h)\), and \(C_{\bot h} = C(t_k, t_h, t_h, \ldots, t_h)\) for \(k, h \in \mathbb{N}\). We show that every constraint from every \(E_i\) is satisfied on all \(C_{kh}\) with \(k, h \geq 1\), not just for \(k < h\). More precisely, let \(i \in [m]\), \((u', v') \in E_i\).
and \((u, v) = (w; u', v; v')\). We show \(C_{kh}|u = C_{kh}|v\) for all \(k, h \geq 1\). Note that by assumption, \(C_{kh}|u = C_{kh}|v\) is true for all \(k, h \in \mathbb{N}\) with \(k < h\). We show our statement by a case distinction depending on the position of \(u\) and \(v\) in relation to the positions \(u(1)\), ..., \(u(t)\).

1. If both \(u\) and \(v\) are parallel to \(u(1)\), then \(C_{ij}|u\) and \(C_{ij}|v\) do not depend on \(i\). Thus, \(C_{ij}|u = C_{ij}|v\) for all \(j \geq 1\) implies the statement.

2. If \(u\) is in prefix-relation with \(u(1)\) and \(v\) is parallel to \(u(1)\), then \(C_{ij}|v\) does not depend on \(j\). If \(u \leq u(1)\), then by our assumption that \((t_i)_{i \in \mathbb{N}}\) are pairwise distinct, we obtain the contradiction \(C_{02}|v = C_{02}|u \neq C_{12}|u = C_{12}|v\), where \(C_{02}|v \neq C_{12}|v\) should hold. Thus, we have \(u(1) \leq u\) and in particular, \(C_{ij}|u\) does not depend on \(j\). Thus, for all \(i, j \geq 1\) we obtain \(C_{ij}|u = C_{i,i+1}|u = C_{i,i+1}|v = C_{0,i+1}|v = C_{0,i+1}|u = C_{ij}|u = C_{ij}|v\). If \(v\) is in prefix-relation with \(u(1)\) and \(u\) is parallel to \(u(1)\), then we come to the same conclusion by formally exchanging \(u\) and \(v\) in this argumentation.

3. If \(u\) and \(v\) are both in prefix-relation with \(u(1)\), then \(u\) and \(v\) being parallel to each other implies \(u(1) \leq u\) and \(u(1) \leq v\). In particular, both \(u\) and \(v\) are parallel to all \(u(1), \ldots, u(t)\). Thus, we obtain, as in the first case, that \(C_{ij}|u\) and \(C_{ij}|v\) do not depend on \(j\) and the statement follows from \(C_{i,i+1}|u = C_{i,i+1}|v\) for all \(i \in \mathbb{N}\).

Let \(k, h \geq 1\) and \(d_C \in D_C\), and let \(q \in Q_2, d_k, d_{k+1} \in D_G^2(C_{k,k+1})\), and \(d_{h-1,k} \in D_G^2(C_{h-1,k})\) such that \(d_C = d_{k,k+1}|\pos(C) = d_{h-1,k}|\pos(C)\). Then for \(d_k = d_k|\pos(C)\), \(d_h = d_h|\pos(C)\), we can reorder \(d = d_k d_h d_C\) to a complete left-most derivation of \(G_2\) for \(C_{kh}\) as all equality constraints from \(d_k\) are satisfied by the assumption on \(d_k, d_{k+1}\), all equality constraints from \(d_h\) are satisfied by the assumption on \(d_{h-1,k}\), and all equality constraints from \(d_C\) are satisfied by our case distinction. Considering the special cases \(k = 2, h = 1\), and \(k = h = 1\), and the definitions of \(D_{>}\) and \(D_{=}\), we obtain \(D_C \subseteq D_{21} = D_{=}\) and \(D_C \subseteq D_{11} = D_{>}\), and hence, \(D_C \subseteq D_{>}\) and \(D_C \subseteq D_{=}\).

The converse inclusion \(D_{>} \subseteq D_{=}\) follows with an analogous reasoning. In conclusion, we obtain \(D_C = D_{>} \subseteq D_{=}\). By the reasoning above, the case \(D_C = \emptyset\) we excluded earlier is only possible if also \(D_{>} = \emptyset\), in which case we again have \(D_C = D_{>} \subseteq D_{=}\).

Let \(d_1, \ldots, d_n\) be an enumeration of \(D_{<}\), \(i \in [n]\), and \(k \in \mathbb{N}\). We define the sets

\[
D_{i,k}^{(1)} = \{d|\pos(C) = \pos(C)\} \mid d \in D_G^2(C_{k,k+1}), d_i = d|\pos(C), q \in Q_2\}
\]

\[
D_{i,k}^{(2)} = \{d|\pos(C) = \pos(C)\} \mid d \in D_G^2(C_{k,k+1}), d_i = d|\pos(C), q \in Q_2\}
\]

and the corresponding weights \(\nu_{i,k}^{(1)} = \sum_{d \in D_{i,k}^{(1)}} \wt_2(d)\) and \(\nu_{i,k}^{(2)} = \sum_{d \in D_{i,k}^{(2)}} \wt_2(d)\). Let \(q_i\) be the target state of the last production in \(d_i\) and define \(\nu_i = F(q_i) \cdot \wt_2(d_i)\). For all \(k, h \in \mathbb{N}\) we have \(\|G_2\| = \nu_{k}^{(1)} + \nu_{k}^{(2)} + \delta_{kh}\mu_k\) for nonnegative \((\mu_k)_{k \in \mathbb{N}}\) which stem from the fact that potentially \(D_{=} \cap D_{>} = \emptyset\). We arrange the weights \(\nu_{i,k}^{(1)}\) into a row vector \(\mathbf{\nu}_{i,k}^{(1)}\), and the weights \(\nu_{i,k}^{(2)}\) into a column vector \(\mathbf{\nu}_{i,k}^{(2)}\), and the weights \(\nu_i\) into a diagonal matrix \(N\) such that \(\|G_2\| = \nu_i^{(1)} N \nu_{i,k}^{(2)} + \delta_{kh}\mu_k\). Finally, recall that \(\|G_2\| > 0\) iff \(k = h\) for all \(k, h \in \mathbb{N}\). Thus we set \(\mu_k = \nu_i^{(1)} N \nu_{i,k}^{(2)} + \delta_{kh}\mu_k\) with \(\mu_k > 0\) for all \(k, h \in \mathbb{N}\). 

Before concluding the correctness of our decision procedure for the weighted HOM-problem, we want to exemplify how the Lemma 12 acts on a simple weighted tree language.

**Example 19.** Consider the WTGH \(G = \{q, qf, \downarrow\}, \{a(0), g(1), f(2)\}, F, P, wt\) with final weights \(F(qf) = 1, F(q) = F(\downarrow) = 0\) and the following productions:

\[
P = \{ a \rightarrow q, \quad g(q) \rightarrow 2q, \quad f(q, \downarrow) \stackrel{1=2}{\rightarrow} qf, \quad f(q, g(\downarrow)) \stackrel{1=21}{\rightarrow} qf \} \cup P
\]
where \( P_{\perp} = \{ a \rightarrow \perp, \, g(\perp) \rightarrow \perp, \, f(\perp, \perp) \rightarrow \perp \} \). The production \( f(q, \perp) \stackrel{1}{\rightarrow} q_f \) and the tree \( g^{ht}(G) \) satisfy the conditions in the large duplication property, so let \( G_1 \) denote the WTGh constructed according to Lemma 15 which simulates all derivations of \( G \) that use this production at \( \varepsilon \). Consider the sequence \( t_i = g^{ht}(G) \) for \( i \in \mathbb{N} \). The context \( C = f(\square, \square) \) satisfies \( \|G_1\|(C(t_i, t_j)) \neq 0 \) iff \( i = j \). In order to reproduce the linear-algebra argument from the special case of \( \|G'\| + 1 \), described above, we need a matrix representation for the remaining part \( \|G_2\| \), possibly with an additional factor \( \delta_{ij} \). In terms of the weights computed by \( G_2 \), we can achieve this by the condition that \( \|G_2\|(C(t_i, t_j)) \neq 0 \) either for all \( i, j \in \mathbb{N} \), or for none, or only if \( i = j \). However, because of the production \( f(q, g(\perp)) \stackrel{1}{\rightarrow} q_f \), for each \( i \) we have \( \|G_2\|(C(t_i, t_{i+1})) \neq 0 \) and \( \|G_2\|(C(t_i, t_j)) = 0 \) for all \( j \neq i + 1 \). To fix this issue, we may select the subsequence \( \{t_{2i}\}_{i \in \mathbb{N}} \). In that case, we have \( \|G_2\|(C(t_{2i}, t_{2j})) = 0 \) for all \( i, j \in \mathbb{N} \), and the matrix representation for \( \|G_2\| \) is trivial.

Let us now conclude the decidability of the \( \mathbb{N} \)-weighted HOM-problem.

**Theorem 20.** Let \( G = (Q \cup \{ \perp \}, \Sigma, F, P, wt) \) be a trim WTGh. If \( G \) satisfies the large duplication property, then \( G \) is not regular.

**Proof.** Let \( C \in T_{\Sigma}(\square), \, \langle t_i \rangle_{i \in \mathbb{N}} \subseteq T_{\Sigma}, \, m \in \mathbb{N}, \, (v_1^{(1)})_{n \in \mathbb{N}}, (v_2^{(2)})_{n \in \mathbb{N}} \subseteq \mathbb{N}^m, \, N \in \mathbb{N}^{m \times m}, \) and \( (\mu_i)_{i \in \mathbb{N}} \subseteq \mathbb{N} \setminus \{0\} \) be as in Lemma 18, i.e., \( [G](C(t_k, t_h, t_i, \ldots, t_h)) = \nu_k^{(1)} N_{t_k}^{(2)} + \delta_{kh} \mu_k \) for all \( k, h \in \mathbb{N} \). If \( [G] \) is regular, then we can assume a representation for all \( k, h \in \mathbb{N} \) as \( [G](C(t_k, t_h, t_i, \ldots, t_h)) = g(\kappa_k, \kappa_h, \kappa_i, \ldots, \kappa_h) \), where \( \kappa_h \) is a finite vector of weights over \( \mathbb{N} \) where each entry corresponds to the sum of all derivations for \( t_h \) to a specific state of a WTA, and \( g \) is a multilinear map encoding the weights of the derivations for \( C(\square, \square, \ldots, \square) \) depending on the specific input states at the \( \square \) nodes and the target state at the root \( \epsilon \). We choose \( K \) such that the concatenated vectors \( \langle \kappa_1, \nu_1^{(1)} \rangle, \ldots, \langle \kappa_K, \nu_K^{(1)} \rangle \) form a generating set of the \( \mathbb{Q} \)-vector space spanned by \( \langle (\kappa_i, \nu_i^{(1)}) \rangle_{i \in \mathbb{N}} \). Then there are coefficients \( a_1, \ldots, a_K \in \mathbb{Q} \) with \( \kappa_{K+1} = \sum_{i \in [K]} a_i \kappa_i \) and \( \nu_{K+1}^{(1)} = \sum_{i \in [K]} a_i \nu_i^{(1)} \). Thus, we reach our contradiction by

\[
\nu_{K+1}^{(1)} N_{K+1}^{(2)} + \mu_{K+1} = g(\kappa_{K+1}, \kappa_{K+1}, \ldots, \kappa_{K+1}) = \sum_{i \in [K]} a_i g(\kappa_i, \kappa_{K+1}, \ldots, \kappa_{K+1})
\]

\[
= \sum_{i \in [K]} a_i \nu_i^{(1)} N_{K+1}^{(2)} = \nu_{K+1}^{(1)} N_{K+1}^{(2)}.
\]

**5 Conclusion**

In this contribution, we proved that the \( \mathbb{N} \)-weighted HOM-problem is decidable. Formally, given a regular weighted tree language \( A \) over \( \mathbb{N} \) and a nondeleting, nonerasing tree homomorphism \( h \) as input, it is decidable in polynomial time whether the homomorphic image \( h_A \) is again regular. This was achieved by reducing the HOM-problem to the newly introduced large duplication property, which formalizes the non-regular behavior of the investigated weighted tree language \( h_A \), and then showing that this property is decidable.

Initially, \( h_A \) is represented by a generalized tree grammar (WTGh) as introduced in [20]. Such a device expresses the duplication of subtrees performed by \( h \) by means of explicit equality constraints. This WTGh is trimmed and tested directly for the large duplication property. If it does not satisfy this property, we construct an equivalent weighted tree grammar without constraints, which proves regularity of the generated weighted tree language. However, if the trim WTGh for \( h_A \) does satisfy the large duplication property, then no equivalent weighted
Weighted HOM-Problem for Nonnegative Integers

tree grammar exists. To prove this, we first identify a special sequence of productions, isolate it from the remainder of the WTGc, and then prove that it induces a non-regularity which cannot be compensated by the remaining derivations of the WTGh.

We require $h$ to be nondeleting and nonerasing simply to ensure that $h_A$ is well-defined in general. These properties have no impact on the correctness of the reduction or the computational complexity of the large duplication property, to which we reduce the $N$-weighted HOM-problem. Indeed, our decision procedure for this problem is polynomial, while the unweighted HOM-problem is EXPTIME-complete [6]. In the $N$-weighted setting we proved that the large duplication property is sufficient for non-regularity; this is the main technical difficulty and utilizes RAMSEY’S theorem to identify a sequence of trees that acts as a witness for the non-regularity of the homomorphic image. A matrix representation that resembles the initial algebra semantics is then utilized to prove non-regularity. In the unweighted case the large duplication property is clearly necessary, but not sufficient. This difference is caused by the different algebraic structures of the underlying semirings. Whereas the semiring $\mathbb{N}$ embeds into a field, the BOOLEAN semiring is idempotent, which can be used to cover non-regular behavior with regular behavior making it irrelevant. Essentially we proved that such covers are impossible in $\mathbb{N}$, which simplifies the execution of the decision procedure and allows us to prove polynomial-time decidability of the $N$-weighted HOM-problem.

References