非负整数加权 HOM 问题

Andreas Maletti ！ 莱比锡大学计算机科学研究所，04109 莱比锡，德国

Andreea-Teodora Nász ！ 莱比锡大学计算机科学研究所，德国莱比锡 04109。

Erik Paul ！

通信研究所 计算机科学系，莱比锡大学，邮编 04109，莱比锡，德国。德国

摘要

HOM 问题询问：给定一个树同态，一个正则树语言在该同态下的像是否仍为正则。Godoy、Giménez、Ramos 和 Àlvarez 最近证明了该问题是可判定的。本文研究了这一问题的 $\mathbb{N}$ 加权版本，并证明其可判定。更确切地说，判断正则 $\mathbb{N}$ 加权树语言在非删除、非擦除树同态下的像是否为正则，是多项式时间可解的。

2012 年 ACM 学科分类 计算理论 → 定量自动机；计算理论 → 可计算性；计算理论 → 树语言；计算理论→ 文法与上下文无关语言

关键词与短语 加权树自动机、判定问题、子树相等约束、树同态、HOM 问题、加权树文法、加权 HOM 问题

数字对象标识符 10.4230/LIPIcs.CVIT.2016.23

资助 *Andreea-Teodora Nász*：研究经费由萨克森自由州授予 T. Nasz 的奖学金提供。

arXiv:2305.04117v1 [cs.FL] 6 May 2023

1 引言

非确定性有限状态字符串自动机这一重要模型在过去几十年中经历了多种扩展。值得注意的是，通过 [21]中的加权自动机，其定性评估被推广为定量评估。这些自动机得到了广泛研究 [20], ，这主要归功于它们能够简洁地表示与所处理输入相关的成本、资源或时间消耗以及概率等过程因素。由于半环具有一般性，且其合理的计算效率源于分配律，因此半环 [13, 14] 成为一种非常适合用于权重评估的代数结构。

与这一发展平行，有限状态自动机已推广到处理其他形式的输入，例如无限词 [18] 和树 [2]。自[4, 22, 23]提出以来，有限状态树自动机及其生成的正则树语言 一直得到广泛研究。这些模型已被证明适用于多种应用领域，包括自然语言处理 [15], 、图像生成 [5], 和编译器构造 [24]。在许多情况下，应用需要将定量评估与作为更具表达力的输入结构的树相结合，由此推动了多种加权树自动机（WTA）模型的发展。 [9, 第 9]章提供了全面的概述。

有限状态树自动机存在若干重大局限，包括无法确保被接受的树中任意大小的两棵子树相等。这些限制早已广为人知 [10], ，而 [17], 针对上述缺陷提出了一种扩展模型，可明确要求特定子树相等或不等。这些模型在研究树变换时非常便利 [9], ，而树变换可以实现

© Andreas Maletti、Andreea-Teodora Nász 和 Erik Paul；依据知识共享署名 4.0 国际许可协议授权 第 42 届非常重要主题会议（CVIT 2016）。编辑：John Q. Open 和 Joan R. Access；第 23 号文章；第 23:1–23:15 页 信息学领域莱布尼茨国际论文集 达格施图尔城堡——信息学莱布尼茨中心，Dagstuhl 出版社，德国

子树复制；它们也是开创性论文 [11]，中使用的主要工具，该论文确立了 HOM 问题的可判定性。

HOM 问题是树语言研究中一个长期悬而未决的问题，它询问：给定一个树同态，正则树语言在该同态下的像是否也正则。由于树同态可以复制子树，因此其像不一定是正则的。事实上，如果从树同态中去掉这种复制能力（例如，线性树同态），那么其像总是正则的 [10]。经典的（布尔）HOM 问题最近在 [11, 12]，中得到了解决：该方法将像表示为一个带约束的树自动机，再判断它是否生成正则树语言。后来又证明该问题是 EXPTIME 完全的 [3]。

在加权情形下，HOM 问题的可判定性仍悬而未决。此前关于加权情形下正则性保持的研究 [1, 6, 7, 8]，主要关注明确排除同态复制能力的情况。在本文中，我们证明，对于正则 $\mathbb{N}$加权树语言，HOM 问题可以在多项式时间内轻松判定。我们的证明思路受到 [11] 的启发：考虑一个正则 $\mathbb{N}$加权树语言，以及一个非删除、非擦除树同态。首先，我们利用加权树自动机 [16] 的一种扩展（WTGh），高效地表示该语言在同态下的像。现在的问题是判定该 WTGh 的正则性，而我们工作的核心思路是将其（非）正则性归约为一个更易处理的问题：大复制性质。进而，我们在引理11中直接证明，大复制性质可以在多项式时间内判定。如果该同态像对应的 WTGh 不具有此性质，那么我们给出一个等价 $\mathbb{N}$加权 WTG 的有效构造（尽管需要指数时间），从而证明其正则性。否则，我们使用 [16] 中提出的抽引引理，并从 WTGh 中分离出一个严格非正则的部分。证明中最具挑战性的部分，也是我们的主要技术贡献，是证明同态像的剩余部分无法抵消这种非正则行为。为此，我们运用 Ramsey 定理 [19]，为整个加权树语言的非正则性找到一个见证。

2 预备知识

我们用 $\mathbb{N}$ 表示非负整数集。对于 $i,\ j \in \mathbb{N}$ 我们令 $[i,j] = \{k \in \mathbb{N} \mid i \le k \le j\}$和 $[j] = [1,j]$。设 Z 为任意集合。Z 的基数记为 $|Z|$，而由 Z 中的元素构成的词的集合（即由 Z 中元素组成的有序有限序列集合）记为 Z^*。

树、替换与上下文

秩字母表 (Σ, rk) 由有限集合 Σ 和映射 $\mathrm{rk}\colon \Sigma \to \mathbb{N}$ 构成，该映射为 Σ 中的每个符号赋予一个秩。若不会引起混淆，我们就用 Σ 单独表示秩字母表 (Σ, rk)。我们用 $\sigma^{(k)}$ 表示 $\mathrm{rk}(\sigma) = k$。此外，对于每个 $k \in \mathbb{N}$，令 $\Sigma_k = \mathrm{rk}^{-1}(k)$，且 $\mathrm{rk}(\Sigma) = \max\{k \in \mathbb{N} \mid \Sigma_k \neq \emptyset\}$ 为 Σ 中符号的最大秩。设 $X = \{x_i \mid i \in \mathbb{N}\}$ 为一个可数的（形式）变量集。对于每个 $n \in \mathbb{N}$，令 $X_n = \{x_i \mid i \in [n]\}$。给定秩字母表 Σ 和集合 Z，以 Z 为指标的 Σ-树集合 $T_\Sigma(Z)$ 是满足以下条件的最小集合：$Z \subseteq T_\Sigma(Z)$，并且对于每个 $k \in \mathbb{N}$、$\sigma \in \Sigma_k$ 以及 $t_1, \ldots, t_k \in T_\Sigma(Z)$，都有 $\sigma(t_1, \ldots, t_k) \in T_\Sigma(Z)$。我们将 $T_\Sigma(\emptyset)$ 简记为 T_Σ，其任意子集 $L \subseteq T_\Sigma$ 称为树语言。

设 Σ 为一个秩字母表，Z 为一个集合，且 $t \in T_\Sigma(Z)$。位置集合 $\mathrm{pos}(t)$ 定义如下：对所有 $z \in Z$，$\mathrm{pos}(z) = \{\varepsilon\}$；对所有 $k \in \mathbb{N}$、$\sigma \in \Sigma_k$ 和 t_1、...、$t_k \in T_\Sigma(Z)$，$\mathrm{pos}(\sigma(t_1, \ldots, t_k)) = \{\varepsilon\} \cup \{iw \mid i \in [k], w \in \mathrm{pos}(t_i)\}$。借助它们，我们将 t 的大小 ‘size(t)’ 和高度 ‘ht(t)’ 定义为 $\mathrm{size}(t) = |\mathrm{pos}(t)|$，且 $\mathrm{ht}(t) = \max_{w \in \mathrm{pos}(t)} |w|$。位置之间存在偏序

它们在 $[\mathrm{rk}(\Sigma)]^*$ 上按标准前缀序 $\leq$ 排列，并在 $[\mathrm{rk}(\Sigma)]^*$ 上按升序字典序 $\preceq$ 全序排列，其中前缀较大；也就是说，ε 是最大元素。更精确地说，对 v、$w \in \mathrm{pos}(t)$，若存在 $u \in [\mathrm{rk}(\Sigma)]^*$ 使得 $vu = w$，则记作 $v \leq w$，称 v 为 w 的前缀，并令 $v^{-1}w = u$，因为如果 u 存在，它是唯一确定的。若 $u = u_1 \cdots u_n$，其中 u_1、…、$u_n \in [\mathrm{rk}(\Sigma)]$，我们还将从 v 到 w 的路径 $[v, \ldots, w]$ 定义为位置序列 $(v, vu_1, vu_1u_2, \ldots, w)$。任意两个按 $\leq$ 不可比较的位置称为平行。

给定 $t, t' \in T_\Sigma(Z)$ 和 $w \in \mathrm{pos}(t)$，t 在 w 处的标签 $t(w)$、t 在 w 处的子树 $t|_w$，以及将 t' 代换到 t 在 w 处的代换 $t[t']_w$，定义为 $z(\varepsilon) = z|_\varepsilon = z$ 和 $z[t']_\varepsilon = t'$（对所有 $z \in Z$），以及 $t(\varepsilon) = \sigma$，$t(iw') = t_i(w')$，$t|_\varepsilon = t$，$t|_{iw'} = t_i|_{w'}$，$t[t']_\varepsilon = t'$，以及 $t[t']_{iw'} = \sigma(t_1, \ldots, t_{i-1}, t_i[t']_{w'}, t_{i+1}, \ldots, t_k)$（对所有具有 $k \in \mathbb{N}$，$\sigma \in \Sigma_k$，$t_1, \ldots, t_k \in T_\Sigma(Z)$ 的树 $t = \sigma(t_1, \ldots, t_k)$、所有 $i \in [k]$ 和所有 $w' \in \mathrm{pos}(t_i)$）。对于所有由符号构成的集合 $S \subseteq \Sigma \cup Z$，令 $\mathrm{pos}_S(t) = \{w \in \mathrm{pos}(t) \mid t(w) \in S\}$，并且对每个 $s \in \Sigma \cup Z$，用 $\mathrm{pos}_s(t)$ 表示 $\mathrm{pos}_{\{s\}}(t)$。t 中出现的变量集合为 $\mathrm{var}(t) = \{x \in X \mid \mathrm{pos}_x(t) \neq \emptyset\}$。最后，考虑 $n \in \mathbb{N}$ 和一个映射 $\theta' \colon X_n \to T_\Sigma(Z)$。那么通过代换，$\theta'$ 导出一个映射 $\theta \colon T_\Sigma(Z) \to T_\Sigma(Z)$，其定义为：对每个 $x \in X_n$，$\theta(x) = \theta'(x)$；对每个 $z \in Z \setminus X_n$，$\theta(z) = z$；以及对所有 $k \in \mathbb{N}$，$\sigma \in \Sigma_k$ 和 $t_1, \ldots, t_k \in T_\Sigma(Z)$，$\theta(\sigma(t_1, \ldots, t_k)) = \sigma(\theta(t_1) \ldots, \theta(t_k))$。对于 $t \in T_\Sigma(Z)$，我们将 $\theta(t)$ 记作 $t\theta$，或者更常见地记作 $t[x_1 \leftarrow \theta'(x_1) \ldots, x_n \leftarrow \theta'(x_n)]$。

设 $\square \notin \Sigma$。上下文 是一个具有 $\mathrm{pos}_\square(C) \neq \emptyset$ 的树 $C \in T_\Sigma(\square)$。更具体地说，如果 $n = |\mathrm{pos}_\square(C)|$，我们称 C 为 n-上下文。对于一个 n-上下文 C 和 $t_1, \ldots, t_n \in T_\Sigma$，我们按如下方式定义代换 $C[t_1, \ldots, t_n]$。设 $\mathrm{pos}_\square(C) = \{w_1, \ldots, w_n\}$ 是 C 中 $\square$ 的出现位置，并按字典序排列 $w_1 \prec \cdots \prec w_n$。接着令 $C[t_1, \ldots, t_n] = C[t_1]_{w_1} \cdots [t_n]_{w_n}$。

树同态与加权树文法

给定秩字母表 Σ 和 Γ，设 $h' \colon \Sigma \to T_\Gamma(X)$ 是一个映射，且对所有 $k \in \mathbb{N}$ 和 $\sigma \in \Sigma_k$，都有 $h'(\sigma) \in T_\Gamma(X_k)$。我们将 h' 扩展为 $h \colon T_\Sigma \to T_\Gamma$，定义为对所有 $\alpha \in \Sigma_0$，有 $h(\alpha) = h'(\alpha) \in T_\Gamma(X_0) = T_\Gamma$；并且对所有 $k \in \mathbb{N}$、$\sigma \in \Sigma_k$ 以及 t_1、…、$t_k \in T_\Sigma$，有 $h(\sigma(t_1, \ldots, t_k)) = h'(\sigma)[x_1 \leftarrow h(t_1) \ldots, x_k \leftarrow h(t_k)]$。映射 h 称为由 h' 诱导的树同态，我们将 h' 与其诱导的树同态 h 视为相同。为了分析判定过程的复杂度，我们将 h 的大小定义为 $\mathrm{size}(h) = \sum_{\sigma \in \Sigma} |\mathrm{pos}(h(\sigma))|$。若对所有 $k \in \mathbb{N}$ 和 $\sigma \in \Sigma_k$，$h'(\sigma) \notin X$（分别地，$\mathrm{var}(h'(\sigma)) = X_k$），则称 h 为非擦除的（分别为非删除的）。本文中，我们只考虑非擦除且非删除的树同态 $h \colon T_\Sigma \to T_\Gamma$；因此它们是输入有限的，即对每个 $u \in T_\Gamma$，其原像 $h^{-1}(u)$ 都是有限的，因为对于每个 $t \in h^{-1}(u)$，都有 $|t| \leq |u|$。任意映射 $A \colon T_\Sigma \to \mathbb{N}$ 称为 $\mathbb{N}$-加权树语言；对于每个 $u \in T_\Gamma$，我们定义加权树语言 $h_A \colon T_\Gamma \to \mathbb{N}$ 为 $h_A(u) = \sum_{t \in h^{-1}(u)} A(t)$，并称其为 A 在 h 下的像。此定义要求树同态是输入有限的；否则，定义中的求和并非有限，因此 $h_A(u)$ 的值未必有定义。

带等式约束的加权树文法（WTGc）[16] 是一个元组 $(Q, \Sigma, F, P, \mathrm{wt})$，其中 Q 是有限状态集，Σ 是输入符号的秩字母表，$F \colon Q \to \mathbb{N}$ 为每个状态指定一个终权重，P 是有限产生式集，产生式的形式为 (ℓ, q, E)，其中 $\ell \in T_\Sigma(Q) \setminus Q$，$q \in Q$，且 $E \subseteq \mathbb{N}^* \times \mathbb{N}^*$ 是有限子集；$\mathrm{wt} \colon P \to \mathbb{N}$ 为每个产生式指定一个权重。产生式 $p = (\ell, q, E) \in P$ 通常写作 $p = \ell \xrightarrow{E} q$ 或 $p = \ell \xrightarrow{E}_{\mathrm{wt}(p)} q$，其中树 ℓ 分别称为其左部，q 为其目标状态，E 为其等式约束。等式约束 $(v, v') \in E$ 也写作 $v = v'$。如果 $F(q) \neq 0$，则称状态 $q \in Q$ 为终态。

接下来，我们回顾文献 [16]中 WTGc 的推导语义。设 $(v, v') \in \mathbb{N}^* \times \mathbb{N}^*$ 是一个等式约束，且 $t \in T_\Sigma$。当且仅当 $v, v' \in \mathrm{pos}(t)$ 且 $t|_v = t|_{v'}$时，树 t 满足 (v, v')；对于等式约束的有限集 $C \subseteq \mathbb{N}^* \times \mathbb{N}^*$，若 t满足所有 $(v, v') \in C$，则记作 $t \models C$。设 $G = (Q, \Sigma, F, P, \mathrm{wt})$ 是一个 WTGc。句型（针对 G）是形如 $\xi \in T_\Sigma(Q)$ 的树。给定输入树 $t \in T_\Sigma$、句型 $\xi, \zeta \in T_\Sigma(Q)$、产生式 $p = \ell \xrightarrow{E} q \in P$以及位置 $w \in \mathrm{pos}(\xi)$，当且仅当 $\xi|_w = \ell$，$\zeta = \xi[q]_w$ 且 $t|_w \models E$ 时，我们记作 $\xi \Rightarrow_{G,t}^{p,w} \zeta$；也就是说，等式约束 E 在 $t|_w$ 上得到满足。若序列 $d = (p_1, w_1) \cdots (p_n, w_n) \in (P \times \mathbb{N}^*)^*$ 满足以下条件，则称其为针对 t 的推导（G）：存在 $\xi_0, \ldots, \xi_n \in T_\Sigma(Q)$，使得 $\xi_0 = t$，并且对所有 $i \in [n]$，均有 $\xi_{i-1} \Rightarrow_{G,t}^{p_i,w_i} \xi_i$。如果还满足 $w_1 \prec w_2 \prec \cdots \prec w_n$，则称 d为最左推导。注意，如果句型 $\xi_0, \ldots, \xi_n$ 存在，则它们是唯一确定的；并且对于针对 t 的任意推导 d，都存在 d 的一个唯一排列，使其成为针对 t 的最左推导。如果 $\xi_n \in Q$，则称 d 为完全推导；此时也称其为到达 ξ_n 的推导。针对 t 到 $q \in Q$ 的所有完全最左推导的集合记为 $D_G^q(t)$。到达某个终态的完全推导称为接受推导。如果对每个 $p \in P$，都存在一棵树 $t \in T_\Sigma$、一个终态 q 以及一个推导 $d = (p_1, w_1) \cdots (p_m, w_m) \in D_G^q(t)$，使得 $F(q) \cdot \mathrm{wt}_G(d) \neq 0$ 且 $p \in \{p_1, \ldots, p_m\}$；也就是说，如果每个产生式都用于某个接受推导中，则称 G 是修剪的。

令 $d = (p_1, w_1) \cdots (p_n, w_n) \in D_G^q(t)$，其中 $t \in T_\Sigma$ 和 $i \in [n]$满足某些条件。进一步地，令 $\{j_1, \ldots, j_\ell\}$ 为集合 $\{j \in [n] \mid w_i \leq w_j\}$，其索引为 $j_1 < \cdots < j_\ell$，对应的位置以 w_i 为前缀。我们将 $(p_{j_1}, w_i^{-1} w_{j_1}) \ldots, (p_{j_\ell}, w_i^{-1} w_{j_\ell})$ 称为包含在 d 中的 $t|_{w_i}$ 的推导。反过来，对于 $w \in \mathbb{N}^*$，我们将推导 $(p_1, ww_1) \cdots (p_n, ww_n)$ 简记为 wd。

推导 $d = (p_1, w_1) \cdots (p_n, w_n)$ 的权重 定义为 $\mathrm{wt}_G(d) = \prod_{i=1}^n \mathrm{wt}(p_i)$。由 G 生成的加权树语言，记作 $[\![G]\!]$: $T_\Sigma \to \mathbb{N}$，对所有 $t \in T_\Sigma$ 定义如下

$$[\![G]\!](t) = \sum_{q \in Q,\, d \in D_G^q(t)} F(q) \cdot \mathrm{wt}_G(d) \ .$$

对于 $t \in T_\Sigma$ 和 $q \in Q$，我们通常会使用由 $\mathrm{wt}_G^q(t)$ 定义的值 $\mathrm{wt}_G^q(t) = \sum_{d \in D_G^q(t)} \mathrm{wt}_G(d)$。利用分配律，$[\![G]\!](t)$ 随即可简化为 $[\![G]\!](t) = \sum_{q \in Q} F(q) \cdot \mathrm{wt}_G^q(t)$。如果两个 WTGc 生成相同的加权树语言，我们就称它们等价 。

如果 WTGc $(Q, \Sigma, F, P, \mathrm{wt})$ 满足对每个产生式 $\ell \xrightarrow{E} q \in P$ 均有 $E = \emptyset$，即没有产生式使用等式约束，我们就称其为加权树文法 (WTG)。我们也常将 $\ell \xrightarrow{\emptyset} q$ 简写为 $\ell \to q$。此外，如果 WTGc 对每个产生式 $\ell \xrightarrow{E} q \in P$ 均满足 $\mathrm{pos}_\Sigma(\ell) = \{\varepsilon\}$，则称其为带等式约束的加权树自动机 (WTAc)；如果它同时是 WTG 和 WTAc，则称其为加权树自动机 (WTA)。WTGc 和 WTAc 的表达能力相同，且严格强于 WTA [16] 类。若一个加权树语言由 WTA 生成，则称其为正则；若由 WTGc 生成，则称其为约束正则 。权重为 0 的产生式显然没有用处，因此我们可以假设对每个产生式 p 都有 $\mathrm{wt}(p) \neq 0$ 。最后，我们定义如下的 WTGc 大小。

▶ 定义*1.* 设 $G = (Q, \Sigma, F, P, \mathrm{wt})$ 为一个 *WTGc*，$p = \ell \xrightarrow{E} q \in P$为一个产生式。我们定义 p 的高度为 $\mathit{ht}(p) = \mathit{ht}(\ell)$，其大小为 $\mathit{size}(p) = |\mathrm{pos}(\ell)|$；$P$的高度为 $\mathit{ht}(P) = \max_{p \in P} \mathit{ht}(p)$，其大小为 $\mathit{size}(P) = \sum_{p \in P} \mathit{size}(p)$；最后，$G$ 的高度为 $\mathit{ht}(G) = |Q| \cdot \mathit{ht}(P)$，其大小为 $\mathit{size}(G) = |Q| + \mathit{size}(P)$。

众所周知，[16] WTGc 可用于表示正则加权树语言的同态像。设 A: $T_\Sigma \to \mathbb{N}$ 是一个正则加权树语言（由一个 WTA 有效给出），且 h: $T_\Sigma \to T_\Gamma$ 是一个树同态。根据 [16, 定理 5]，我们可以构造一个具有特定形式的 WTGc $G = (Q, \Gamma, F, P, \mathrm{wt})$，使得 $[\![G]\!] = h_A$。更确切地说，

构造得到的 WTGc G 具有一个指定的非终结汇状态 $\bot \in Q$，满足 $F(\bot) = 0$，且 $p_\gamma = \gamma(\bot, \dots, \bot) \to \bot \in P$，并且对每个 $\gamma \in \Gamma$，$\mathrm{wt}(p_\gamma) = 1$。此外，每个产生式 $p = \ell \xrightarrow{E} q \in P$ 都满足以下两个性质。首先，$E \subseteq \mathrm{pos}_Q(\ell)^2$；也就是说，所有等式约束都指向其左部中带有 Q 标签的位置。不失一般性，我们可以假设等式约束集合 E 具有自反性、对称性和传递性；也就是说，它是子集 $D \subseteq \mathrm{pos}_Q(\ell)$ 上的一个等价关系，因此并非所有状态的出现位置都需要受到约束。其次，对每个 $v \in [w']_E \setminus \{w\}$ 和 $w' \in D$，$\ell(v) = \bot$，且 $\ell(w) \neq \bot$，其中 $w = \min_{\preceq}[w']_E$；也就是说，在 E 的每个等价类中，除了字典序最小的位置外，其余位置都由状态 $\bot$ 进行标记。本质上，WTGc G 只在受等式约束的子树中字典序最小的出现位置执行检查（并收取权重）。所有其他子树因约束而被迫与另一棵子树相同，WTGc 会直接忽略它们；形式上，这意味着它们由指定的汇状态 $\bot$ 处理。下文中，我们用 $\bot$ 表示这样的汇状态，并用 $Q \cup \{\bot\}$ 明确表示它的存在。

刚才讨论的特殊形式的 [16] WTGc 被称为 eq-restricted，但由于它们是本文主要关注的对象，我们在此将它们简称为加权树文法。以下命题的构造性证明可见附录。

▶ 定理 2（参见 [16, 定理 5]）。设 $G = (Q, \Sigma, F, P, \mathrm{wt})$ 是一个修剪加权树自动机，且 $h: T_\Sigma \to T_\Gamma$ 是一个非删除且非擦除的树同态。那么存在一个修剪加权树文法 G'，满足 $[\![G']\!] = h_{[\![G]\!]}$。此外，$\text{大小}(G') \in O(\text{大小}(G) \cdot \text{大小}(h))$，且 $\mathit{ht}(G') \in O(\text{大小}(h))$。

▶ **例 3.** 令 $G = (Q \cup \{\bot\}, \Gamma, F, P, \mathrm{wt})$ 由 $Q = \{q, q_f\}$、$\Gamma = \{\alpha^{(0)}, \gamma^{(1)}, \delta^{(3)}\}$、$F(q) = F(\bot) = 0$ 以及 $F(q_f) = 1$ 构成，并设其产生式集合 P 如下。

$$\left\{\alpha \to_1 q,\ \gamma(q) \to_2 q,\ \delta\big(q, \gamma(\bot), q\big) \xrightarrow{1=21}_1 q_f,\quad \alpha \to_1 \bot,\ \gamma(\bot) \to_1 \bot,\ \delta(\bot, \bot, \bot) \to_1 \bot\right\}$$

加权树文法 WTGc G 是一个加权树文法 WTGh。它生成树同态 h（由映射 $\alpha \mapsto \alpha$，$\gamma \mapsto \gamma(x_1)$ 和 $\sigma \mapsto \delta\big(x_2, \gamma(x_2)\ x_1\big)$ 诱导）作用于正则加权树语言 A 所得的同态像 $[\![G]\!] = h_A$：对每个满足 $\Sigma = \{\alpha^{(0)}, \gamma^{(1)}, \sigma^{(2)}\}$ 的 $t \in T_\Sigma$，由 $A(t) = 2^{|\mathrm{pos}_\gamma(t)|}$ 给出。加权树语言 $[\![G]\!]$ 本身并非正则，因为其支撑显然不是正则树语言。

WTGh 定义中的限制条件使我们能够使用简单的可达性算法有效地对其进行修剪。有关更多细节，请参阅附录。

▶ **引理 4。** 令 $G = (Q \cup \{\bot\}, \Sigma, F, P, \mathrm{wt})$ 为一个加权树文法。可以在多项式时间内构造出一个等价的、修剪的加权树文法 G'。

3 存在等式约束时的替换

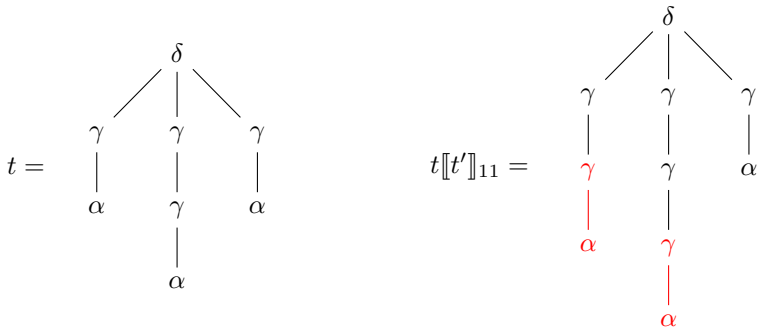
本节简要回顾 [16] 中的一些定义，以及一个关于加权树文法的抽引引理；该引理对于判定整数加权同态问题至关重要。首先，我们需要细化树的代换，使其符合现有的约束。

▶ 定义 5（见 [16]，另参见 [11]）。设 $G = (Q \cup \{\bot\}, \Sigma, F, P, \mathrm{wt})$ 是一个加权树文法，并设 $d = (p_1, w_1) \cdots (p_m, w_m)$ 是一棵树 $t \in T_\Sigma$ 到状态 $q \in Q \cup \{\bot\}$ 的完全最左推导。此外，设 $j \in [m]$，使得当且仅当 $q_j = \bot$ 时有 $q = \bot$；即，产生式 p_j 的目标状态 q_j 为 $\bot$，当且仅当 $q = \bot$。注意，只要 $q = \bot$，便自动有 $q_j = \bot$。最后，设 d' 是某棵树 $t' \in T_\Sigma$ 到 q_j 的一个推导，并设 $d'_\bot$ 是 t' 的推导，其中每个出现的状态均为 $\bot$。我们递归地定义将 d' 在 w_j 处代换入 d 的代换 $d[\![d']\!]_{w_j}$ 如下。

- 如果 $w_j = \varepsilon$ （即 $j = m$），则定义 $d[\![d']\!]_{w_j} = d'$。否则，设 $p_m = \ell \xrightarrow{E} q$ 是最后使
- 用的产生式，$pos_Q(\ell) = \{v_1, \ldots, v_n\}$，并设 $d_1, \ldots, d_n$ 分别是并入 d 中的 $t|_{v_1}, \ldots, t|_{v_n}$ 的推导。显然存在 $s \in [n]$ 使得 $v_s \le w_j$。设 $\hat{w} = v_s^{-1} w_j$ 为 d_s 中出现的一个位置。相应地，我们定义 $d'_s = d_s[\![d']\!]_{\hat{w}}$；对于每个 $i \in [n] \setminus \{s\}$，当 $(v_i, v_s) \in E$ 时定义 $d'_i = d_i[\![d'_\perp]\!]_{\hat{w}}$，否则定义 $d'_i = d_i$。然后，通过重新排列推导 $(v_1 d'_1) \cdots (v_n d'_n)(p_m, w_m)$，使其成为最左推导，即可得到 $d[\![d']\!]_{w_j}$。

由 $d[\![d']\!]_{w_j}$ 推导出的树记作 $t[\![t']\!]^d_{w_j}$，如果上下文中已明确 t 的原始推导，也可简单记作 $t[\![t']\!]_{w_j}$。

▶ **例 6.** 考虑例 3 中的加权树文法 G 及其生成的下列树 t，我们想在位置 $w = 11$ 处将树 $t' = \gamma(\alpha)$ 代入其中。



我们考虑从 t 到 q_f 的以下完全最左推导。

$$d = \big(\alpha \to q, 11\big)\big(\gamma(q) \to q, 1\big) \quad \big(\alpha \to \perp, 211\big)\big(\gamma(\perp) \to \perp, 21\big)$$
$$\big(\alpha \to q, 31\big)\big(\gamma(q) \to q, 3\big)\big(\delta(q, \gamma(\perp), q) \xrightarrow{1=21} q_f, \varepsilon\big)$$

此外，设 $d' = (\alpha \to q、1)(\gamma(q) \to q、\varepsilon)$ 和 $d'_\perp = (\alpha \to \perp、1)(\gamma(\perp) \to \perp、\varepsilon)$。根据定义 5 的记号，在第一步中，我们有 $v_1 = 1、v_2 = 21、v_3 = 3、d_1 = d_3 = d'、d_2 = d'_\perp$ 和 $\hat{w} = v_1^{-1} w = 1$。遵循唯一的约束 $1 = 21$，我们令 $d'_1 = d_1[\![d']\!]_{\hat{w}} = d'[\![d']\!]_1、d'_2 = d_2[\![d'_\perp]\!]_{\hat{w}} = d'_\perp[\![d'_\perp]\!]_1$ 和 $d'_3 = d_3 = d'$。最终，$d'_1 = (\alpha \to q, 11)(\gamma(q) \to q, 1)(\gamma(q) \to q, \varepsilon)$ 和 $d'_2 = (\alpha \to \perp, 11)(\gamma(\perp) \to \perp, 1)(\gamma(\perp) \to \perp, \varepsilon)$。因此，我们得到了新树 $t[\![t']\!]_{11}$ 的以下推导 $d[\![d']\!]_{11}$。

$$d[\![d']\!]_{11} = \big(\alpha \to q, 111\big)\big(\gamma(q) \to q, 11\big)\big(\gamma(q) \to q, 1\big)\big(\alpha \to \perp, 2111\big)\big(\gamma(\perp) \to \perp, 211\big)$$
$$\big(\gamma(\perp) \to \perp, 21\big) \quad \big(\alpha \to q, 31\big)\big(\gamma(q) \to q, 3\big)\big(\delta(q, \gamma(\perp), q) \xrightarrow{1=21} q_f, \varepsilon\big)$$

尽管 $t|_{31} = \alpha$ 也与我们替换的子树 $t|_{11} = \alpha$ 重合，但这两棵子树不受等式约束，因此同时替换不会影响 $t|_{31}$。

定义 5 的代换使我们能够证明加权树文法的一个抽引引理：如果 d 是加权树文法 $G = (Q \cup \{\perp\}, \Sigma, F, P, \mathrm{wt})$ 的一个接受推导，且对应于一棵满足 $\mathrm{ht}(t) > \mathrm{ht}(G)$ 的树 t，那么至少存在 $|Q \setminus \{\perp\}| + 1$ 个位置 $w_1 > \cdots > w_{|Q|+1}$，在这些位置上，t 应用的产生式具有非汇点目标状态。由鸽巢原理可知，t 中存在两个位置 $w_i > w_j$，在这两个位置上，d 应用的产生式具有相同的非汇点目标状态。利用刚才定义的代换，我们可以将 $t|_{w_j}$ 代换到 w_i 中，从而得到 G 对 $t[\![t|_{w_j}]\!]_{w_i}$ 的一个推导。重复这一过程，可以得到一个大小严格递增的无限树序列。形式化地说，以下引理已在 [16] 中得到证明。

▶ **引理 7** ([16, 引理 4])。 设 $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$ 是一个加权树文法。考虑某棵树 $t \in T_\Sigma$ 和非汇状态 $q \in Q \setminus \{\bot\}$，使得 $\mathrm{ht}(t) > \mathrm{ht}(G)$ 且 $D^q_G(t) \neq \emptyset$。那么存在无穷多个两两不同的树 $t_0, t_1, \ldots$，使得对所有 $i \in \mathbb{N}$， $D^q_G(t_i) \neq \emptyset$。

▶ **例 8**。 回顾例 3 中的加权树文法 G。我们有 $\mathrm{ht}(P) = 2$ 和 $\mathrm{ht}(G) = 4$，但为求简洁，我们选择较小的树 $t = \delta(\gamma(\alpha)\ \gamma(\gamma(\alpha))\ \gamma(\alpha))$，我们在例 6 中也考虑过这棵树，因为它同样可以进行抽引。例 6 中给出的推导 d 对于 t在 11 处应用产生式 $(\alpha \to q)$，并在 1 处应用 $\gamma(q) \to q$，因此我们在 11 处代换为 $t|_1 = \gamma(\alpha)$，得到 $t[\![\gamma(\alpha)]\!]_{11}$。事实上，这正是我们在例 6 中展示的代换。

4 判定过程

现在我们来讨论 $\mathbb{N}$-加权版本的 HOM 问题。接下来，我们将证明，正则 $\mathbb{N}$-加权树语言的同态像是否正则，可以在多项式时间内判定。更准确地说，我们证明以下定理。

▶**定理 9. </style}在 上的加权 HOM 问题可在多项式时间内解决；也就是说，对于固定的秩字母表 和，给定一个在 上的修剪加权树自动机，以及一个非删除、非擦除树同态：** $T_\Gamma \to T_\Sigma$, 可以在多项式时间内判定 $h_{[\![H]\!]}$ 是否为正则语言。

证明中，我们遵循非加权情形 [11] 的总体思路。给定一个正则加权树语言 A（由修剪加权树自动机表示）和一个树同态 h，我们首先应用定理 2，构造一个修剪加权树文法 G，用于表示其像 $[\![G]\!] = h_A$。接着，我们证明，$[\![G]\!]$ 是正则的，当且仅当对于 G 的所有推导，推导中出现的等式约束仅适用于高度至多为 $\mathrm{ht}(G)$ 的子树。换言之，如果存在一个产生式 $\ell \xrightarrow{E} q$，属于 G，使得对于某个等式约束 $(u,\ v) \in E$，其中非汇点状态为 $q = \ell(u)$，存在一棵树 $t \in T_\Sigma$，满足 $\mathrm{ht}(t) > \mathrm{ht}(G)$ 且 $D^q_G(t) \neq \emptyset$，那么 $[\![G]\!]$ 就不是正则的；若不存在这样的产生式，则 $[\![G]\!]$ 就是正则的。因此，我们的证明分为三个部分。首先，我们证明是否存在这样的产生式可以在多项式时间内判定。然后，我们证明若不存在这样的产生式，则 $[\![G]\!]$ 是正则的。最后，我们证明若存在这样的产生式，则 $[\![G]\!]$ 不是正则的。为方便起见，我们为这里描述的性质命名。

▶ **定义10**。 令 $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$ 为一个修剪*WTGh*。若存在一个产生式 $\ell \xrightarrow{E} q \in P$、一个等式约束 $(u, v) \in E$，其中 $\ell(u) \neq \bot = \ell(v)$，以及一棵树 $t \in T_\Sigma$，使得 $\mathrm{ht}(t) > \mathrm{ht}(G)$ 且 $D^{\ell(u)}_G(t) \neq \emptyset$，则称 G 具有大复制性质。

我们从大复制性质的可判定性开始。

▶**引理11**。 考虑一个固定的秩字母表 Σ。以下问题可在多项式时间内判定：给定一个修剪 *WTGh* G，它是否满足大复制性质？

证明。令 $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$，并构造有向图 $G = (Q, E)$，其边为 $E = \bigcup_{\ell \xrightarrow{E} q \in P} \{(q', q) \mid q' \in Q, \mathrm{pos}_{q'}(\ell) \neq \emptyset\}$。显然，大复制性质等价于以下条件：存在一个产生式 $\ell \xrightarrow{E} q \in P$、一个等式约束 $(u, v) \in E$ with $\ell(u) \neq \bot = \ell(v)$，以及一个状态 $q' \in Q \setminus \{\bot\}$，使得在 G 中存在一条从 q' 到 q' 的环，并且在 G 中存在一条从 q' 到 q 的路径。这个等价条件可以在多项式时间内检验。两个陈述的等价性很容易证明。如果大复制性质成立，那么抽引引理 [16, Lemma 4] 给出了所需的环和路径。反过来，如果环和路径存在，那么可以使用抽引引理 [16,Lemma 4] 推导出任意高的树，且存在相应的推导。◀

接下来，我们证明，如果加权树文法 G 不满足大复制性质，就会生成正则 $[\![G]\!]$。为此，我们构造 G 的 线性化。加权树文法 G 的线性化是一种加权树文法，它模拟 G 的所有推导，这些推导只确保高度至多为 $\mathrm{ht}(G)$ 的子树等价。具体做法是，将 G 中的每个产生式 $\ell \xrightarrow{E} q$ 替换为所有可通过以下方式得到的产生式 $\ell' \to q$：将受 E 约束的每个位置替换为高度至多为 $\mathrm{ht}(G)$ 且满足 E 的等式约束的相容树。注意，ℓ 中不受 E 约束的位置不受这些代换的影响。形式上，我们按照定义 7.1] 定义线性化 [11, 。

▶ 定义12。设 $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$ 是一个加权树文法。G 的线性化 $\mathrm{lin}(G)$ 是加权树文法 $\mathrm{lin}(G) = (Q \cup \{\bot\}\ \Sigma, F, P_{\mathrm{lin}}, \mathrm{wt}_{\mathrm{lin}})$，其中 P_{lin} 和 $\mathrm{wt}_{\mathrm{lin}}$ 定义如下。对于 $\ell' \in T_\Sigma(Q) \setminus Q$ 和 $q \in Q$，当且仅当存在一个产生式 $(\ell \xrightarrow{E} q) \in P$、位置 $w_1, \ldots, w_k \in \mathrm{pos}_{Q \cup \{\bot\}}(\ell)$，以及树 $t_1, \ldots, t_k \in T_\Sigma$，使得 $\{w_1, \ldots, w_k\} = \bigcup_{w \in \mathrm{pos}}$

- $_{\bot}(\ell)[w]_E$；即，E 恰好约束位置 w_1、…、w_k，
- $t_i = t_j$ 若 $(w_i, w_j) \in E$ 对所有 $i, j \in [k]$，$\ell' = \ell[t_1]_{w_1} \cdots$
- $[t_k]_{w_k}$ 都成立，并且 $D^{\ell(w_i)}{}_G(t_i) \neq \emptyset$ 且 $\mathrm{ht}(t_i) \leq \mathrm{ht}(G)$
- 对所有 $i \in [k]$ 都成立。

对于每个这样的产生式 $\ell' \to q$，我们将 $\mathrm{wt}_{\mathrm{lin}}(\ell' \to q)$ 定义为所有权重之和

$$\mathrm{wt}(\ell \xrightarrow{E} q) \cdot \prod_{i \in [k]} \mathrm{wt}_G^{\ell(w_i)}(t_i)$$

对所有 $(\ell \xrightarrow{E} q) \in P$, $w_1, \ldots, w_k \in \mathrm{pos}_{Q \cup \{\bot\}}(\ell)$，以及如上所述的 $t_1, \ldots, t_k \in T_\Sigma$。

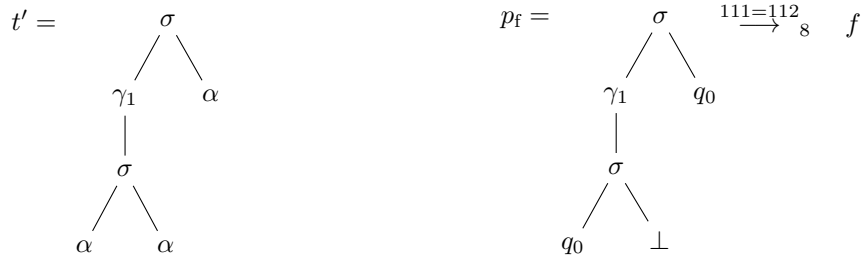
如果修剪WTGh G 不满足大复制性质，那么 G 的每个推导中的每个等式约束都只能确保高度至多为 $\mathrm{ht}(G)$ 的子树相等。因此，$\mathrm{lin}(G)$ 和 G 生成相同的加权树语言 $[\![G]\!] = [\![\mathrm{lin}(G)]\!]$；由于 $\mathrm{lin}(G)$ 是一个 WTG，该语言因而是正则的。由此我们总结如下：

▶**命题 13.** 设 G 是一个修剪*WTGh*，并且 G 不满足大复制性质。那么，$[\![G]\!]$是一个正则加权树语言。

最后，我们证明，如果一个 WTGh $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$ 满足大复制性质，那么 $[\![G]\!]$ 不是正则的。为此，我们首先证明，如果 G 满足大复制性质，那么可以将其分解为两个 WTGh G_1 和 G_2，使得 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$，且 $[\![G_1]\!]$ 和 $[\![G_2]\!]$ 中至少有一个不是正则的。为得出所需结论，我们接着证明它们的和 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$ 也不是正则的。关于这种分解，考虑以下思路。假设存在一个产生式 $p = (\ell \xrightarrow{E} q) \in P$，如大复制性质中所述，满足 $F(q) \neq 0$。然后，我们按如下方式创建 G 的两个副本 G_1 和 G_2。在 G_1 中，我们将所有终权重设为 0，添加一个终权重为 $F(q)$的新状态 f，并添加新产生式 $(\ell \xrightarrow{E} f)$，其权重与 p相同。另一方面，在 G_2中，我们将 q 的终权重设为 0，添加一个终权重为 $F(q)$的新状态 f，并且对于除 p之外的每个产生式 $p' = (\ell' \xrightarrow{E'} q) \in P$，都向 G_2 添加新产生式 $\ell' \xrightarrow{E'} f$，其权重与 p'相同。于是 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$，因为 G 的每个以 p 为最后一个产生式的推导，现在都是从 G_1 到 f的推导；而其他推导要么直接是 G_2 的推导，要么在其他推导到 q的情况下，是从 G_2 到 f的推导。

根据我们对产生式 $p = (\ell \xrightarrow{E} q)$ 的假设，存在一棵高树 $t \in T_\Sigma$，其高度 $\mathrm{ht}(t) > \mathrm{ht}(G)$，以及一个约束 $(u, v) \in E$，满足 $\ell(u) \neq \bot = \ell(v)$ 且 $D_G^{\ell(u)}(t) \neq \emptyset$。因此，由 G_1 生成的每棵树 t' 都满足 $t'|_u = t'|_v$；根据引理 7，存在无穷多棵两两不同的树，它们都能通过推导得到 $\ell(u)$。因此，$p =$ 的支撑（即非零权重树的集合

图 1 树 t' 和新的产生式 p_f



t）不是正则树语言。这意味着 $[\![G_1]\!]$ 不是正则的
as 每个关于 $\mathbb{N}$ 的正则加权树语言的支撑都是正则树语言 [9] .

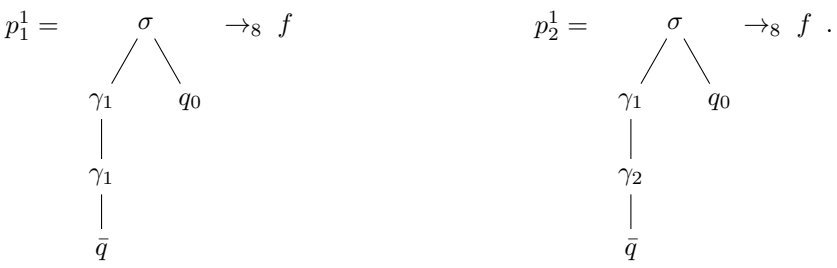
一般而言，我们不能指望存在一个产生式 $\ell \xrightarrow{E} q$，如大复制性质中所述，满足 $F(q) \neq 0$。关于一般情形的细节，请参阅附录。

▶ 引理 *14.* 设 $G = (Q \cup \{\bot\}\ \Sigma, F, P, wt)$ 是一个满足大复制性质的修剪*WTGh*。那么存在两个修剪 *WTGh* $G_1 = (Q_1 \cup \{\bot\}\ \Sigma, F_1, P_1, wt_1)$ 和$G_2 = (Q_2 \cup \{\bot\}\ \Sigma, F_2, P_2, wt_2)$，使得 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$，并且对某个 $f \in Q_1$，有$F_1(f) \neq 0$且 $F_1(q) = 0$，对所有 $q \in Q_1 \setminus \{f\}$ 均成立，并且

- 存在且仅存在一个产生式 $p_\mathrm{f} = (\ell_\mathrm{f} \xrightarrow{E_\mathrm{f}} f) \in P_1$，其目标状态为 f；对于该产生式，存在 $(u, v) \in E_\mathrm{f}$，使得 $\ell_\mathrm{f}(u) \neq \ell_\mathrm{f}(v) = \bot$，并且存在一个由两两不同的树组成的无限序列 $t_0, t_1, t_2, \ldots \in T_\Sigma$，使得对于所有 $i \in \mathbb{N}$，均有 $D^{\ell_\mathrm{f}(u)}{}_{G_1}(t_i) \neq \emptyset$。

▶ 例 15. 我们给出一个引理 14 中分解的示例。考虑修剪WTGh $G = (Q \cup \{\bot\}\ \Sigma, P, F, \mathrm{wt})$，其中 $Q = \{q_0, \bar{q}, q_\mathrm{f}\}$、 $\Sigma = \{\alpha^{(0)}, \gamma^{(1)}, \sigma^{(2)}, \gamma_1^{(1)}, \gamma_2^{(1)}\}$、终权重 $F(q_\mathrm{f}) = 1$和 $F(q_0) = F(\bar{q}) = F(\bot) = 0$，以及由 $P' = \{\, \alpha \to_1 q_0,\ \gamma(q_0) \to_1 q_0,\ \sigma(q_0, \bot) \xrightarrow{1=2}{}_2 \bar{q},\ \gamma_1(\bar{q}) \to_2 \bar{q},\ \gamma_2(\bar{q}) \to_2 \bar{q},\ \sigma(\bar{q}, q_0) \to_2 q_\mathrm{f} \}$定义的集合 $P = P_\bot \cup P'$，并采用以 $\bot$ 为目标状态、位于 $P_\bot$ 中的通常产生式。任意高度的形如 $\gamma(\cdots(\gamma(\alpha))\cdots)$ 的树都受到约束 $1 = 2$，因此 G 满足大复制性质。

我们考虑图 1 中的 t'，并使用它在 G中的（唯一）推导。按照上面概述的方法，我们选择一个新状态 f，并定义 $G_1 = (Q \cup \{f\} \cup \{\bot\}\ \Sigma, F_1, P_1, \mathrm{wt}_1)$，其中对于每个 $q \in Q \cup \{\bot\}$，都有 $F_1(f) = 1$ 和 $F_1(q) = 0$，并且 $P_1 = P \cup \{p_\mathrm{f}\}$，其采用图 1 所示的新产生式 p_f。该产生式将 G 中用于推导 t'的所有产生式连接起来，从触发大复制性质的产生式到指向终态的产生式。接下来需要构造一个加权树文法 G_2，使得 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$。 G 的所有产生式仍然出现在 G_2中，但 q_f不再是终态。取而代之的是，我们添加一个状态 f，使得 $F_2(f) = F(q_\mathrm{f}) = 1$，并确保该状态接纳所有其他原先导向 q_f 的接受推导。为此，我们首先处理在交接位置 ε 和 1 处与 t' 的推导一致、但在 2 处不一致的推导。这会产生以下新产生式 p_1^1 和 p_2^1:



接下来，我们处理在位置 1 处与 t' 的推导不同、但在根节点处与之相同的推导。这会产生以下新产生式

$$p_1^2 = \sigma(\sigma(q_0, q_0), \bot) \xrightarrow{11=12}_4 f(\sigma(\gamma_2(q_0), q_0)) \qquad p_2^2 = \sigma(\gamma_2(q_0), q_0) \to_4 f(\gamma_2(q_0)) .$$

除了并入 p_f 根部的产生式外，没有其他产生式会直接以 q_f 为目标，因此不会再向 P_2 添加产生式。

最后，我们定义加权树文法 $G_2 = (Q \cup \{f\} \cup \{\bot\}\ \Sigma, F_2, P_2, \mathsf{wt}_2)$，其中 $F_2(f) = F(q_\mathrm{f}) = 1$、$F_2(q_\mathrm{f}) = F_2(q_0) = F_2(\bar{q}) = F_2(\bot) = 0$，且 $P_2 = P \cup \{p_1^1, p_2^1\} \cup \{p_1^2, p_2^2\}$。

接下来要证明，存在如引理 14 中所述的分解 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$，意味着 $[\![G]\!]$ 的非正则性。为此，我们采用以下思路。考虑一个包含一个秩为 2 的字母 σ 的秩字母表 Σ，一个以 Σ 为底层字母表的加权树自动机 $G' = (Q, \Sigma, F, P, \mathsf{wt})$（它体现了 G_2），以及一个由两两不同的树组成的序列 $t_0, t_1, t_2, \ldots \in T_\Sigma$。此时，我们假设 P 包含所有可能的产生式，但对于 $p \in P$，可能有 $\mathsf{wt}(p) = 0$。利用初始代数语义 [9], ，我们可以为 G' 赋予形如 $\sigma(t_i, t_j)$ 的树的权重找到一个矩阵表示，如下所示。我们将状态 $Q = \{q_1, \ldots, q_n\}$ 编号，并对每个 $i \in \mathbb{N}$ 定义一个（列）向量 $\nu_i \in \mathbb{N}^n$，使得 $(\nu_i)_k = \mathsf{wt}_{G'}^{q_k}(t_i)$，其中 $k \in [n]$。此外，我们定义矩阵 $N \in \mathbb{N}^{n \times n}$，使得 $N_{kh} = \sum_{q \in Q} F(q) \cdot \mathsf{wt}(\sigma(q_k, q_h) \to q)$，其中 $k, h \in [n]$。于是，对于所有 $i, j \in \mathbb{N}$，都有 $[\![G']\!](\sigma(t_i, t_j)) = \nu_i^\mathrm{T} N \nu_j$，其中 ν_i^T 是 ν_i 的转置。

我们利用这一矩阵表示，证明 $[\![G']\!]$ 与树语言 $L = \{\sigma(t_i, t_i) \mid i \in \mathbb{N}\}$ 的（非正则）特征函数 1_L 之和不是正则的。我们采用反证法，假设 $[\![G']\!] + 1_L$ 是正则的。因此，我们可以利用矩阵 N' 和向量 ν_i'，为 $[\![G']\!] + 1_L$ 构造一个类似的矩阵表示。由于树 $t_0, t_1, t_2, \ldots$ 两两不同，我们可以写出

$$\big([\![G']\!] + 1_L\big)\big(\sigma(t_i, t_j)\big) = (\nu_i')^\mathrm{T} N' \nu_j' = [\![G']\!]\big(\sigma(t_i, t_j)\big) + \delta_{ij} = \nu_i^\mathrm{T} N \nu_j + \delta_{ij}$$

对于所有 i，其中 δ_{ij} 表示克罗内克δ函数。向量 ν_i' 和 ν_i 包含非负整数，因此我们可以将拼接后的向量 $\mathsf{h}\nu_i', \nu_i\mathsf{i}$ 视为 $\mathbb{Q}^m$ 的向量，其中 $m \in \mathbb{N}$ 是 G' 的状态数与我们假设能够识别 $[\![G']\!] + 1_L$ 的加权树自动机的状态数之和。由于 $\mathbb{Q}^m$ 是有限维的 $\mathbb{Q}$-向量空间，由族 $(\mathsf{h}\nu_i', \nu_i\mathsf{i})_{i \in \mathbb{N}}$ 张成的 $\mathbb{Q}$-向量空间也是有限维的。因此，我们可以从 $(\mathsf{h}\nu_i', \nu_i\mathsf{i})_{i \in \mathbb{N}}$ 中选取一个有限生成集。为简单起见，我们假设 $\mathsf{h}\nu_1', \nu_1\mathsf{i} \ldots, \mathsf{h}\nu_K', \nu_K\mathsf{i}$ 构成这样的生成集。因此，存在 $a_1, \ldots, a_K \in \mathbb{Q}$，使得 $\mathsf{h}\nu_{K+1}', \nu_{K+1}\mathsf{i} = \sum_{i \in [K]} a_i \mathsf{h}\nu_i', \nu_i\mathsf{i}$。应用矩阵乘法通常的分配律，我们如下导出矛盾。

$$\begin{aligned}\big([\![G']\!] + 1_L\big)\big(\sigma(t_{K+1}, t_{K+1})\big) &= (\nu_{K+1}')^\mathrm{T} N' \nu_{K+1}' = \sum_{i \in [K]} a_i (\nu_i')^\mathrm{T} N' \nu_{K+1}' \\ &= \sum_{i \in [K]} a_i \nu_i^\mathrm{T} N \nu_{K+1} = \nu_{K+1}^\mathrm{T} N \nu_{K+1} = [\![G']\!]\big(\sigma(t_{K+1}, t_{K+1})\big)\end{aligned}$$

对于一般情况，我们不想假设 $[\![G_2]\!]$ 是正则的，因此不能像上文对 $[\![G']\!]$ 那样假设存在矩阵表示。为了让我们的思路行得通，我们找出一组树，使得 $[\![G_1]\!] + [\![G_2]\!]$ 的行为类似于 $[\![G']\!] + 1_L$；更准确地说，我们构造一个上下文 C 和一个由两两不同的树组成的序列 $t_0,\ t_1,\ t_2,$ .....

使得对于所有 $i, j \in \mathbb{N}$，均有 $(\llbracket G_1 \rrbracket + \llbracket G_2 \rrbracket)(C(t_i, t_j)) = \nu_i^{(1)} N \nu_j^{(2)} + \delta_{ij}\mu_i$，此外，对于所有 $i \in \mathbb{N}$，均有 $\mu_i > 0$。遗憾的是，如果 G_1 使用形如 $\{v = v', v' = v''\}$ 的约束，其中两个以上的位置被约束为两两等价，那么仅使用二元上下文 C 可能并不足够。因此，我们必须考虑更一般的 n-元上下文 C，然后确定一个树序列，使得上述方程在 $C(t_i, t_j, t_j, \ldots, t_j)$ 上成立。

现在我们已准备好陈述最后一个定理。为此，我们将使用以下版本的 Ramsey 定理 [19]。对于集合 X，我们用 $\binom{X}{2}$ 表示 X 中所有大小为 2 的子集构成的集合。

▶ **定理16**。设 $k \geq 1$ 为一个整数，且 $f\colon \binom{\mathbb{N}}{2} \to [k]$ 为一个映射。存在一个无限子集 $E \subseteq \mathbb{N}$，使得对于某个 $i \in [k]$，$f|_{\binom{E}{2}} \equiv i$。

▶ **定理17**。设 $G = (Q \cup \{\bot\}, \Sigma, F, P, \mathrm{wt})$ 为一个修剪*WTGh*。若 G 满足大复制性质，则 $\llbracket G \rrbracket$ 不是正则的。

证明。根据引理 14，存在两个修剪WTGh $G_1 = (Q_1 \cup \{\bot\}, \Sigma, F_1, P_1, \mathrm{wt}_1)$ 和 $G_2 = (Q_2 \cup \{\bot\}, \Sigma, F_2, P_2, \mathrm{wt}_2)$，使得对于所有 $t \in T_\Sigma$，$\llbracket G \rrbracket(t) = \llbracket G_1 \rrbracket(t) + \llbracket G_2 \rrbracket(t)$。此外，存在 $f \in Q_1$，使得对于所有 $q \in Q_1 \setminus \{f\}$，$F_1(f) \neq 0$ 且 $F_1(q) = 0$；并且恰好存在一个产生式 $p_\mathrm{f} = (\ell_\mathrm{f} \xrightarrow{E_\mathrm{f}} f) \in P_1$，其目标状态为 f。最后，对于该产生式 p_f，存在 $(u, v) \in E_\mathrm{f}$，使得 $\ell_\mathrm{f}(u) \neq \ell_\mathrm{f}(v) = \bot$，并且存在由 $t_0, t_1, t_2, \ldots \in T_\Sigma$ 构成的无限序列，其中各树两两不同，且对于所有 $i \in \mathbb{N}$，$D^{\ell_\mathrm{f}}(u)_{G_1}(t_i) \neq \emptyset$。

令 $t \in T_\Sigma$ 满足 $D^f_{G_1}(t) \neq \emptyset$，并令 $w_1, \ldots, w_r$ 为所有通过 E_f 与 u 满足等式约束的位置的枚举，其中假设 $w_1 = u$。我们定义上下文 $C = t[\Box]_{w_1} \cdots [\Box]_{w_r}$。那么，$\llbracket G_1 \rrbracket(C(t_i, t_j, t_j, \ldots, t_j)) > 0$，当且仅当 $i = j$。

我们先引入一些额外的记号。令 $k, h \in \mathbb{N}$，并假设存在 $q \in Q_2$，使得 $F_2(q) \neq 0$，且 $d = (p_1, w_1) \cdots (p_m, w_m) \in D^q_{G_2}(C(t_k, t_h, t_h, \ldots, t_h))$。对每个 $i \in [m]$，令 $p_i = \ell_i \xrightarrow{E_i} q_i$；对于集合 $X \subseteq \mathrm{pos}(C(t_k, t_h, t_h, \ldots, t_h))$，令 $i_1 < \cdots < i_n$ 满足：$w_{i_1}, \ldots, w_{i_n}$ 是 $\{w_1, \ldots, w_m\} \cap X$ 的一个枚举；也就是说，这是 X 中所有 d 应用产生式的位置。我们定义 $d|_X = (p_{i_1}, w_{i_1}) \cdots (p_{i_n}, w_{i_n})$，$\mathrm{wt}_2(d|_X) = \prod_{j \in [n]} \mathrm{wt}_2(p_{i_j})$，以及 $D_{kh} = \{d'|_{\mathrm{pos}(C)} \mid \exists q' \in Q_2\colon F_2(q') \neq 0, d' \in D^{q'}_{G_2}(C(t_k, t_h, t_h, \ldots, t_h))\}$。

现在，我们按以下方式应用 Ramsey 定理。对于 $k, h \in \mathbb{N}$ 其中 $k < h$，我们考虑映射 $\{k, h\} \mapsto D_{kh}$。由于每个 D_{kh} 都是字母表 $P_2 \times \mathrm{pos}(C)$ 上长度至多为 $|\mathrm{pos}(C)|$ 的有限词集合，因此该映射的值域有限。于是，根据 Ramsey 定理，我们得到一个子序列 $(t_{i_j})_{j \in \mathbb{N}}$，其中 $D_{i_k i_h} = D_<$ 对所有 $k, h \in \mathbb{N}$ 均成立，且存在某个集合 $D_<$。为简便起见，我们假设对于所有 $k, h \in \mathbb{N}$，只要 $k < h$，就有 $D_{kh} = D_<$。类似地，我们再选取一个子序列，并假设对于所有 $k, h \in \mathbb{N}$，只要 $k > h$，就有 $D_{kh} = D_>$。最后，映射 $k \mapsto D_{kk}$ 的值域也有限，因此根据鸽巢原理，我们可以再选取一个子序列，并假设对于所有 $k \in \mathbb{N}$，都有 $D_{kk} = D_=$，且存在某个集合 $D_=$。下文将证明 $D_< = D_> \subseteq D_=$。

目前，我们假设 $D_< \neq \emptyset$，令 $(p_1, w_1) \cdots (p_m, w_m) \in D_<$，并令每个 $i \in [m]$ 都有 $p_i = \ell_i \xrightarrow{E_i} q_i$。此外，对于 $k, h \in \mathbb{N}$，我们定义 $C_{kh} = C(t_k, t_h, t_h, \ldots, t_h)$、$C_{k\Box} = C(t_k, \Box, \Box, \ldots, \Box)$ 和 $C_{\Box h} = C(\Box, t_h, t_h, \ldots, t_h)$。我们证明，来自每个 E_i 的每个约束，在所有满足 $k, h \geq 1$ 的 C_{kh} 上都成立，而不仅仅对 $k < h$ 成立。更确切地说，令 $i \in [m]$、$(u', v') \in E_i$，以及 $(u, v) = (w_i u', w_i v')$。我们证明，对所有 $k, h \geq 1$，$C_{kh}|_u = C_{kh}|_v$ 成立。注意，根据假设，对于所有满足 $k < h$ 的 $k, h \in \mathbb{N}$，$C_{kh}|_u = C_{kh}|_v$ 都为真。我们通过分类讨论来证明这一结论，分类依据是 u 和 v 相对于位置 $w_1, \ldots, w_r$ 的位置。

1. 如果 u 和 v 均与 w_1 平行，则 $C_{ij}|_u$ 和 $C_{ij}|_v$ 不依赖于 i。因此，$C_{0j}|_u = C_{0j}|_v$ 对于所有 $j \geq 1$ 均可推出该命题。
2. 如果 u 与 w_1 且 v 与 w_1 平行，则 $C_{ij}|_v$ 不依赖于 i。如果 $u \leq w_1$，那么根据我们的假设 $(t_i)_{i \in \mathbb{N}}$ 两两不同，我们得到

矛盾 $C_{02}|_v = C_{02}|_u \neq C_{12}|_u = C_{12}|_v$，其中应满足 $C_{02}|_v = C_{12}|_v$。因此，我们有 $w_1 \leq u$，尤其是，$C_{ij}|_u$ 不依赖于 j。因此，对于所有 i，$j \geq 1$，我们得到 $C_{ij}|_u = C_{i,\,i+1}|_u = C_{i,\,i+1}|_v = C_{0,\,i+1}|_v = C_{0,\,i+1}|_u = C_{0j}|_u = C_{0j}|_v = C_{ij}|_v$。如果 v 与 w_1 存在前缀关系，且 u 与 w_1 平行，那么通过在此论证中形式上交换 u 和 v，我们可以得出相同的结论。

3. 如果 u 和 v 都与 w_1具有前缀关系，那么 u 和 v 彼此平行，就意味着 $w_1 \leq u$和 $w_1 \leq v$。特别是，u和 v都与所有 $w_2, \ldots, w_m$平行。因此，与第一种情况一样，我们可以得出，$C_{ij}|_u$ 和 $C_{ij}|_v$ 不依赖于 j，而由 $C_{i,i+1}|_u = C_{i,i+1}|_v$ 对所有 $i \in \mathbb{N}$成立即可推出该结论。

设 k、$h \geq 1$ 和 $d_C \in D_<$，并设 $q \in Q_2$、d_k、$_{k+1} \in D^q_{G_2}(C_k$、$_{k+1})$ 以及 d_{h-1}、$_h \in D^q_{G_2}($ C_{h-1}、$_h)$ 满足 $d_C = d_k$、$_{k+1}|_{\mathrm{pos}(C)} = d_{h-1}$、$_h|_{\mathrm{pos}(C)}$。那么，对于 $d_k = d_k$、$_{k+1}|_{\mathrm{pos}(C_{k,k+1})}$ \pos($C_{\square,k+1}$) 和 $d_h = d_{h-1}$、$_h|_{\mathrm{pos}(C_{h-1,h})}$\pos($C_{h-1,\square}$)，我们可以将 $d = d_k d_h d_C$ 重新排序为 C_{kh} 的 G_2 的完全最左推导，因为根据关于 d_k、$_{k+1}$的假设，d_k 中的所有等式约束均得到满足；根据关于d_{h-1}、$_h$的假设，d_h 中的所有等式约束均得到满足；而 d_C 中的所有等式约束则由我们的分类讨论得到满足。考虑特殊情形 $k = 2$、$h = 1$和 $k = h = 1$，以及 $D_>$ 和 $D_=$的定义，我们得到 $d_C \in D_{21} = D_>$ 和 $d_C \in D_{11} = D_=$，进而得到 $D_< \subseteq D_>$ 和 $D_< \subseteq D_=$。

反向包含关系 $D_> \subseteq D_<$ 可通过类似的推理得到。综上，我们得到 $D_< = D_> \subseteq D_=$。根据上述推理，我们先前排除的情形 $D_< = \emptyset$ 只有在同时满足 $D_> = \emptyset$时才可能发生；在这种情况下，我们再次得到 $D_< = D_> \subseteq D_=$。

设 $d_1, \ldots, d_n$是 $D_<$、$i \in [n]$和 $k \in \mathbb{N}$的一个枚举。我们定义集合

$$D^{(1)}_{i,k} = \left\{ d|_{\mathrm{pos}(C_{k,k+1}) \setminus \mathrm{pos}(C_{\square,k+1})} \mid d \in D^q_{G_2}(C_{k,k+1}),\, d_i = d|_{\mathrm{pos}(C)},\, q \in Q_2 \right\}$$

$$D^{(2)}_{i,k} = \left\{ d|_{\mathrm{pos}(C_{k+1,k}) \setminus \mathrm{pos}(C_{k+1,\square})} \mid d \in D^q_{G_2}(C_{k+1,k}),\, d_i = d|_{\mathrm{pos}(C)},\, q \in Q_2 \right\}$$

以及相应的权重 $\nu^{(1)}_i$、$_k = \sum_{d \in D^{(1)}_i}$、$_k$ wt$_2$(d)和 $\nu^{(2)}_i$、$_k = \sum_{d \in D^{(2)}_i}$、$_k$ wt$_2$(d)。最后，令 q_i 为 d_i 中最后一个产生式的目标状态，并定义 $\nu_i = F_2(q_i) \cdot \mathrm{wt}_2(d_i)$。于是，对于所有 k、$h \in \mathbb{N}$，我们都有 $[\![G_2]\!](C_{kh}) = \sum_{i \in [n]} (\nu^{(1)}_{i\,k}, \cdot\, \nu_i \cdot\, \nu^{(2)}_{i\,h},) + \delta_{kh}\mu_k$，其中非负整数权重($\mu_j$)$_{j \in \mathbb{N}}$源于 $D_= \setminus D_< \neq \emptyset$ 可能成立这一事实。我们将权重 $\nu^{(1)}_{i\,k}$ 排列成行向量 $\nu^{(1)}_k$，将权重 $\nu^{(2)}_{i\,h}$ 排列成列向量 $\nu^{(2)}_h$，并将权重 ν_i 排列成对角矩阵 N，使得 $[\![G_2]\!](C_{kh}) = \nu^{(1)}_k N \nu^{(2)}_h + \delta_{kh}\mu_k$。

回忆一下，当且仅当对所有 $k, h \in \mathbb{N}$ 都有 $k = h$ 时，$[\![G_1]\!](C_{kh}) > 0$。因此，我们可以修改权重 μ_k，从而得到 $[\![G]\!](C_{kh}) = [\![G_2]\!](C_{kh}) + [\![G_1]\!](C_{kh}) = \nu^{(1)}_k N \nu^{(2)}_h + \delta_{kh}\mu_k$，其中对所有 $k \in \mathbb{N}$ 都有 $\mu_k > 0$。如果 $[\![G]\!]$ 是正则的，我们可以假设表示为 $[\![G]\!](C_{kh}) = g(\kappa_k, \kappa_h, \kappa_h, \ldots, \kappa_h)$，对所有 $k, h \in \mathbb{N}$ 成立，其中 κ_h 是 $\mathbb{N}$ 上的权重有限向量，其每个条目对应于 t_h 的所有推导到加权树自动机的特定状态的总和，并且 g 是一个多线性映射，用于编码 $C(\square, \square, \ldots, \square)$ 的推导权重，这些权重取决于 $\square$-节点处的具体输入状态以及根 ε 处的目标状态。我们选择 K，使得拼接向量 $\mathbf{h}\kappa_1, \nu^{(1)}_1 \mathbf{i} \ldots, \mathbf{h}\kappa_K, \nu^{(1)}_K \mathbf{i}$ 构成由 $(\mathbf{h}\kappa_i, \nu^{(1)}_i \mathbf{i})_{i \in \mathbb{N}}$ 张成的 $\mathbb{Q}$-向量空间的生成集。于是，存在系数 $a_1, \ldots, a_K \in \mathbb{Q}$，满足 $\kappa_{K+1} = \sum_{i \in [K]} a_i \kappa_i$ 和 $\nu^{(1)}_{K+1} = \sum_{i \in [K]} a_i \nu^{(1)}_i$。因此，我们有

$$\begin{aligned} \nu^{(1)}_{K+1} N \nu^{(2)}_{K+1} + \mu_{K+1} = g(\kappa_{K+1}, \kappa_{K+1}, \ldots, \kappa_{K+1}) &= \sum_{i \in [K]} a_i g(\kappa_i, \kappa_{K+1}, \ldots, \kappa_{K+1}) \\ &= \sum_{i \in [K]} a_i \nu^{(1)}_i N \nu^{(2)}_{K+1} = \nu^{(1)}_{K+1} N \nu^{(2)}_{K+1} \end{aligned}$$

这意味着 $\mu_{K+1} = 0$，从而得出了我们所期望的矛盾。 ◀

参考文献

1 Symeon Bozapalidis 和 George Rahonis。可识别树级数在树同态下的封闭性。J. Autom. Lang. Comb., 10(2–3):185–202, 2005。2 H. Comon、M. Dauchet、R. Gilleron、C. Löding、F. Jacquemard、D. Lugiez、S. Tison 和 M. Tommasi。树自动机——技术与应用，2007。3 Carles Creus、Adrià Gascón、Guillem Godoy 和 Lander Ramos。HOM 问题是 EXPTIME 完全的。载于：2012 年第 27 届 IEEE 逻辑与计算机科学年度研讨会论文集，第 255–264 页。IEEE，2012。4 J ohn Doner。树接受器及其若干应用。J. Comput. System Sci., 4(5):406–451, 1970。5 Frank Drewes。文法图像生成：一种基于树的方法。Springer，2006。6 Zoltán Ésik 和 Werner Kuich。形式树级数。J. Autom. Lang. Comb., 8(2):219–285, 2003。7 Zoltán Fülöp、Andreas Maletti 和 Heiko Vogler。同步树代换文法的可识别性保持。载于：树自动机在自然语言处理中的应用研讨会论文集，第 1–9 页。ACL，2010。8 Zoltán Fülöp、Andreas Maletti 和 Heiko Vogler。加权扩展树变换器。Fundam. Inform., 111(2):163–202, 2011。9 Zoltán Fülöp 和 Heiko Vogler。加权树自动机与树变换器。载于：《加权自动机手册》，第 9 章，第 313–403 页。Springer，2009。10 Ferenc Gécseg 和 Magnus Steinby。树自动机。技术报告 1509.06233，arXiv，2015。11 Guillem Godoy 和 Omer Giménez。HOM 问题是可判定的。J. ACM, 60(4):1–44, 2013。12 Guillem Godoy、Omer Giménez、Lander Ramos 和 Carme Àlvarez。HOM 问题是可判定的。载于：第 42 届 ACM 计算理论研讨会论文集，第 485–494 页。ACM，2010。13 Jonathan S. Golan。《半环及其应用》。Kluwer Academic，Dordrecht，1999。14 Udo Hebisch 和 Hanns J. Weinert。《半环——计算机科学中的代数理论与应用》。World Scientific，1998。15 Dan Jurafsky 和 James H. Martin。《语音与语言处理》。Prentice Hall，第 2 版，2008。16 Andreas Maletti 和 Andreea-Teodora Nász。带约束的加权树自动机，2023。URL: https://arxiv.org/abs/2302.03434, doi:10.48550/ARXIV.2302.03434。17 J. Mongy-Steen。可识别树核的变换。Forêts RATEG。博士论文，Université de Lille，1981。18 Dominique Perrin。自动机与无限词的最新成果。载于：第 11 届数学基础国际研讨会论文集，LNCS 第 176 卷，第 134–148 页。Springer，1984。19 F. P. Ramsey。形式逻辑中的一个问题。Proc. London Math. Soc, 30, 1930。20 Arto Salomaa 和 Matti Soittola。形式幂级数的自动机理论方面。Springer，1978。21 Marcel Paul Schützenberger。关于一类自动机的定义。Inform. and Control, 4(2–3):245–270, 1961。22 James W. Thatcher。通过有限自动机理论的推广刻画上下文无关文法的推导树。J. Comput. Syst. Sci., 1(4):317–322, 1967。23 James W. Thatcher 和 Jesse B. Wright。广义有限自动机理论及其在二阶逻辑判定问题中的应用。Math. Systems Theory, 2(1):57–81, 1968。24 Reinhard Wilhelm、Helmut Seidl 和 Sebastian Hack。《编译器设计》。Springer，2013。

附录

▶ 定理 *2*（见 [16, 定理 5]）。设 $G = (Q, \Sigma, F, P, wt)$ 是一个修剪加权树自动机，且 $h\colon T_\Sigma \to T_\Gamma$ 是一个非删除、非擦除的树同态。那么存在一个修剪加权树文法 G'，满足 $[\![G']\!] = h_{[\![G]\!]}$。此外，$size(G') \in O\big(size(G) \cdot size(h)\big)$，且 $ht(G') \in O\big(size(h)\big)$。

证明。 我们构造一个 WTGc G' 用于 $h_{[\![G]\!]}$，分两个阶段进行。首先，我们构造 WTGc

$$G'' = (Q \cup \{\bot\}, \Delta \cup \Delta \times P, F'', P'', \mathrm{wt}'')$$

使得对于每个产生式 $p = \sigma(q_1, \dots, q_k) \to q \in P$ 以及 $h(\sigma) = u = \delta(u_1, \dots, u_n)$，

$$p'' = \Big(\langle \delta, p\rangle(u_1, \dots, u_n)[\![q_1, \dots, q_k]\!] \xrightarrow{E} q\Big) \in P'' \quad \text{with} \quad E = \bigcup_{i \in [k]} \mathrm{pos}_{x_i}(u)^2$$

其中，代换 $\mathrm{h}\delta, p\mathrm{i}(u_1, \dots, u_n)[\![q_1, \dots, q_k]\!]$ 对每个 $i \in [k]$，仅将 $\mathrm{h}\delta, p\mathrm{i}(u_1, \dots, u_n)$ 中 x_i 最左侧的出现替换为 q_i，并将所有其他出现替换为 $\bot$。此外，$\mathrm{wt}''(p'') = \mathrm{wt}(p)$。另外，我们令 $p''_\delta = \delta(\bot, \dots, \bot) \to \bot \in P''$，其中对于每个 $k \in \mathbb{N}$ 和 $\delta \in \Delta_k \cup \Delta_k \times P$，$\mathrm{wt}''(p''_\delta) = 1$。$P''$ 中没有其他产生式。最后，对于所有 $q \in Q$，我们令 $F''(q) = F(q)$，并令 $F''(\bot) = 0$。

现在我们可以删除注释。首先，我们移除所有指向 $\bot$ 且带有来自 $\Delta \times P$ 的符号标签的产生式。其次，我们使用确定性重标记来移除 $\Delta \times P$ 的标签的第二个分量。因此，我们最终得到一个 WTGc G'，使得 $[\![G']\!] = h_{[\![G]\!]}$。显然，$\mathrm{size}(G') \in \mathcal{O}\big(\mathrm{size}(G) \cdot \mathrm{size}(h)\big)$，并且 $\mathrm{ht}(G) \in \mathcal{O}\big(\mathrm{size}(h)\big)$。

注释的唯一目的，是在计算 $h_{[\![G]\!]}$ 的和之前，建立 G 的有效推导与 G'' 的有效推导之间的一一对应。这使构造正确性的理解更为简单，但除此之外并无必要，为提高效率可以省略。◀

▶ **引理 4.** 设 $G = (Q \cup \{\bot\}\ \Sigma, F, P, \mathrm{wt})$ 是一个加权树文法 *An equivalent, trim WTGh* G' *can be constructed in polynomial time.*

证明。 首先，回顾一下，我们可以假设 $\mathrm{wt}(p) \neq 0$ 对每个 $p \in P$ 都成立，因为 $_G(d) = 0$ 对每个包含产生式 d 的 G 推导都成立，而该产生式 p 满足 $\mathrm{wt}(p) = 0$。证明中，我们采用一个简单的可达性算法。对于每个 $n \in \mathbb{N}$ 和 $U \subseteq Q$，令

$$Q_0 = \emptyset \qquad Q_{n+1} = Q_n \cup \bigcup_{\substack{(\ell \xrightarrow{E} q) \in P \\ \ell \in T_\Sigma(Q_n)}} \{q\} \qquad \Pi_U = \bigcup_{\substack{(\ell \xrightarrow{E} q) \in P \\ \ell \in T_\Sigma(U)}} \big\{(q, q') \in U^2 \mid \mathrm{pos}_{q'}(\ell) \neq \emptyset\big\}\ .$$

由于 Q 是有限的，存在 N 使得 $Q_N = Q_{N+1}$。令 $Q' = Q_N$。一个直接的证明表明，$q \in Q'$ 当且仅当存在某个 $t \in T_\Sigma$，使得存在 $d \in D^q_G(t)$，且 $\mathrm{wt}_G(d) \neq 0$。为确保终态的可达性，我们令 $\lhd$ 为 Q' 上包含 $\Pi_{Q'}$ 的最小自反传递关系。于是 $P' = \{\ell \xrightarrow{E} q \in P \mid q \in Q',\ \exists q_\mathrm{f} \in Q'\colon F(q_\mathrm{f}) \neq 0,\ q_\mathrm{f} \lhd q\}$，而所需的加权树文法只需取为 $G' = (Q \cup \{\bot\}\ \Sigma,\ F,\ P',\ \mathrm{wt}\,|_{P'})$。◀

▶ 引理 *14.* 设 $G = (Q \cup \{\bot\}\ \Sigma, F, P, wt)$ 是满足大复制性质的修剪 *WTGh*。那么存在两个修剪 *WTGh* $G_1 = (Q_1 \cup \{\bot\}\ \Sigma, F_1, P_1, wt_1)$ 和 $G_2 = (Q_2 \cup \{\bot\}\ \Sigma, F_2, P_2, wt_2)$，使得 $[\![G]\!] = [\![G_1]\!] + [\![G_2]\!]$，并且对于某个 $f \in Q_1$，有 $F_1(f) \neq 0$ 且 $F_1(q) = 0$，对所有 $q \in Q_1 \setminus \{f\}$ 均成立，并且

- 恰好存在一个产生式 $p_\mathrm{f} = (\ell_\mathrm{f} \xrightarrow{E_\mathrm{f}} f) \in P_1$，其目标状态为 f；对于该产生式，存在 $(u、v) \in E_\mathrm{f}$，其中 $\ell_\mathrm{f}(u) \neq \ell_\mathrm{f}(v) = \bot$，并且存在一个由两两不同的树组成的无限序列 $t_0、t_1、t_2、\ldots \in T_\Sigma$，使得 $D^{\ell_\mathrm{f}\ (u)\ G_1}(t_i) \neq \emptyset$ 对所有 $i \in \mathbb{N}$ 均成立。

证明。令 $p = (\ell \xrightarrow{E} q) \in P$ 为大复制性质中的一个产生式。由于G 是修剪的，存在一棵树 $t' \in T_\Sigma$、一个终态 $q_\mathrm{f} \in Q$，使得 $F(q_\mathrm{f}) \neq 0$，以及一个推导 $d = (p_1, w_1) \cdots (p_m, w_m) \in D_G^{q_\mathrm{f}}(t')$ 和 $i \in [m]$，使得 $p_i = p$。换言之，存在一个使用产生式 p的推导。对于每个 $j \in [m]$，令 $p_j = \ell_j \xrightarrow{E_j} q_j$；并令 $w_{i_1} > \cdots > w_{i_k}$为 w_i在位置 {w_1，...，w_m}中的前缀按前缀序严格降序排列所得的序列。特别地，我们有 $w_{i_1} = w_i$和 $w_{i_k} = \varepsilon$。

对于位置 w和约束集合 E'，我们定义 $wE' = \{(wu, wv) \mid (u, v) \in E'\}$。我们希望将产生式 $p_{i_1}, \ldots, p_{i_k}$的左部合并为一个新的产生式 $\ell_{i_k}[\ell_{i_{k-1}}]_{w_{i_{k-1}}} \cdots [\ell_{i_1}]_{w_{i_1}} \xrightarrow{E_\mathrm{f}} q_\mathrm{f}$，使其满足 $E_\mathrm{f} = \bigcup_{j\in[k]} w_{i_j} E_{i_j}$。然而，我们需要确保 $w_{i_1}, \ldots, w_{i_k}$不会出现在 E_f中。因此，我们假设上文的 p、t'、q_f、d和 i 的选取方式使得 w_i 在所有可能的选取中具有最小长度。接下来我们证明 $w_{i_1}, \ldots, w_{i_k}$不会出现在 E_f中。

设 $(u, v) \in E$，其中 $\ell(u) \neq \ell(v) = \bot$，且 $t \in T_\Sigma$ 满足 $\mathsf{ht}(t) > \mathsf{ht}(G)$，并且 $D_G^{\ell(u)}(t) \neq \emptyset$。假设存在 $j \in [k]$，使得 w_{i_j} 出现在 E_f中。那么存在 $(u', v') \in E_{i_{j+1}}$，使其满足 $w_{i_j} = w_{i_{j+1}}u'$。树 $t'[\![t]\!]_{w_i u|_{w_{i_j}}}$表明，$p_{i_{j+1}}$ 也是一个符合大复制性质的产生式，但 $|w_{i_{j+1}}| < |w_i|$，因此 w_i 并非最小长度。

我们将 $G_1 = (Q_1 \cup \{\bot\}\ \Sigma, F_1, P_1, \mathsf{wt}_1)$ 定义如下。令 $f \notin Q \cup \{\bot\}$ 为一个新状态。我们设定 $Q_1 = Q \cup \{f\}$、$F_1(f) = F(q_\mathrm{f})$，以及对于所有 $q' \in Q$，$F_1(q') = 0$。对于满足 $E_\mathrm{f} = \bigcup_{j\in[k]} w_{i_j} E_{i_j}$ 的产生式 $p_\mathrm{f} = (\ell_{i_k}[\ell_{i_{k-1}}]_{w_{i_{k-1}}} \cdots [\ell_{i_1}]_{w_{i_1}} \xrightarrow{E_\mathrm{f}} f)$，我们令 $P_1 = P \cup \{p_\mathrm{f}\}$、$\mathsf{wt}_1(p_\mathrm{f}) = \prod_{j\in[k]} \mathsf{wt}(p_{i_j})$，以及对于所有 $p' \in P$，$\mathsf{wt}_1(p') = \mathsf{wt}(p')$。于是，$G_1$ 分别在位置 w_{i_1}、...、w_{i_k} 处模拟了 G 中使用产生式 p_{i_1}、...、p_{i_k} 的所有推导。对于无限树序列的存在性，令 $(u, v) \in E$，其中 $\ell(u) \neq \ell(v) = \bot$，并且 $t \in T_\Sigma$ 满足 $\mathsf{ht}(t) > \mathsf{ht}(G)$ 且 $D_G^{\ell(u)}(t) \neq \emptyset$。根据引理 7，存在一个两两不同的树组成的无限序列 $t_0, t_1, t_2, \ldots \in T_\Sigma$，使得对于所有 $i \in \mathbb{N}$，$D_G^{\ell(u)}(t_i) \neq \emptyset$。由于对于所有 $i \in \mathbb{N}$，$D_G^{\ell(u)}(t_i) \subseteq D_{G_1}^{\ell(u)}(t_i)$，这就是所需的序列。我们通过指出 $(w_i u, w_i v) \in E_\mathrm{f}$，并且 p_f 的左部 ℓ_f 满足 $\ell_\mathrm{f}(w_i u) = \ell(u)$，来完成 G_1 的定义。

接下来，我们构造 G_2，使其按以下意义模拟 G 的所有其余推导。如果 d 是从 G 推导到一个不同于 q_f的状态的推导，那么它也是 G_2的一个推导。如果 d 是从 G 推导到 q_f 的推导，但其最后一个产生式不是 p_{i_k}，那么它会由一个将 G_2 推导到新状态 f的推导来模拟。如果 d 是 G 的一个推导，且其最后一个产生式是 p_{i_k}，但 $w_{i_{k-1}}$ 处的产生式不是 $p_{i_{k-1}}$，那么它同样会由一个将 G_2 推导到 q_f的推导来模拟，依此类推。为了更紧凑地定义 G_2，我们用符号 $\square$ 表示高度为 0 的树，而对于 $j = k$，项 $\square[\ell_{i_k}]_{w_{i_k}} \cdots [\ell_{i_{j+1}}]_{w_{i_{j+1}}}[\ell']_{w_{i_j}}$应读作 $\square[\ell']_{w_{i_j}}$。我们令 $f \notin Q \cup \{\bot\}$ 为一个新状态，并将 $G_2 = (Q_2 \cup \{\bot\}\ \Sigma, F_2, P_2, \mathsf{wt}_2)$ 定义为 $Q_2 = Q \cup \{f\}$,$F_2(q_\mathrm{f}) = 0$, $F_2(f) = F(q_\mathrm{f})$，以及对所有 $q' \in Q \setminus \{q_\mathrm{f}\}$，$F_2(q') = F(q')$。此外，我们令

$$P_2 = P \cup \bigcup_{j\in[k]} \left\{ \square[\ell_{i_k}]_{w_{i_k}} \cdots [\ell_{i_{j+1}}]_{w_{i_{j+1}}}[\ell']_{w_{i_j}} \xrightarrow{E_f} f \;\middle|\; p' = (\ell' \xrightarrow{E'} q_{i_j}) \in P \setminus \{p_{i_j}\},\ E_f = w_{i_j}E' \cup \bigcup_{j'=j+1}^{k} w_{i_{j'}} E_{i_{j'}} \right\}.$$

对于由 p'构造的产生式 $p_f = \square[\ell_{i_k}]_{w_{i_k}} \cdots [\ell_{i_{j+1}}]_{w_{i_{j+1}}}[\ell']_{w_{i_j}} \xrightarrow{E_f} f$，如上所述，我们令 $\mathsf{wt}_2(p_f) = \mathsf{wt}(p') \cdot \prod_{j'=j+1}^{k} \mathsf{wt}(p_{i_{j'}})$，并对每个 $p' \in P$，令 $\mathsf{wt}_2(p') = \mathsf{wt}(p')$。那么，对于每个 $t \in T_\Sigma$，都有 $[\![G]\!](t) = [\![G_1]\!](t) + [\![G_2]\!](t)$。注意，修剪 G_1 和 G_2 不会移除任何新添加的产生式，前提是 G 是修剪的。◀