arXiv:2603.08624v1 [cs.FL] 9 Mar 2026

# 上下文无关树

Jan Philipp Wächter*

曼彻斯特大学数学系，英国曼彻斯特牛津路，邮编 M13 9PL

2026年3月10日

Muller 和 Schupp 提出了上下文无关图的概念（该概念源自上下文无关群的凯莱图）。这类图总是类树的（即与一棵树拟等距）。本文研究其中真正的上下文无关树这一子类。我们证明，这类树可以用多边非确定有限自动机进行有限状态描述；对于确定性图，这种描述可特化为某些部分确定有限自动机。我们还从算法角度研究了这种编码方式，并证明确定性上下文无关树的同构问题在有根和无根情形下都是非确定性对数空间完全的。

**关键词：** 上下文无关图；同构问题；非确定性对数空间完全；Schützenberger图
**2010年数学主题分类：** 68Q17,68R10, 05C62,68Q45

## 1 引言

上下文无关图这一概念由 Muller 和 Schupp 提出 [13]。研究这些图的动机，源于对字问题为上下文无关语言的群的凯莱图的研究（这类群简称为上下文无关群）。在一项著名成果中，同一批作者此前（几乎）证明了上下文无关群恰好就是几乎自由群 [12]（另见 [4]）。他们的证明最初还需要一个额外假设，后来 Dunwoody 证明该假设总是成立 [5]。许多群上的算法问题都与凯莱图（或相关图）的性质直接相关，这促使人们对上下文无关图进行算法研究。早在此时，Muller 和 Schupp 就基于 Rabin树定理 [17] 得出了一个重要结果：任意上下文无关图的单子二阶理论都是可判定的。后来，Kuske 和 Lohrey [9] 基本上证明了反方向，从而揭示了上下文无关图与单子二阶理论可判定性之间的紧密联系。

尽管这种方法功能强大，却不足以回答有关上下文无关图之间关系的问题（例如，它们是否同构）。这类问题出现在

*本研究得到 EPSRC 项目 EP/Y008626/1“特殊逆幺半群：几何、结构与算法”的资助。

组合代数的另一个分支：逆半群和幺半群的研究。在这一领域，凯莱图的作用由所谓的 *Schützenberger* 图取代。围绕它们已经发展出一套非常丰富的理论；其中，Stephen 的开创性工作 [19] 尤为值得一提。与群的情形相比，根本区别在于，我们现在必须处理多个 Schützenberger 图，而非单个凯莱图。因此，我们需要回答有关多个图的问题。当然，这些图一般并非上下文无关图（群的情形甚至也并非如此）。不过，对于有限表示的类树逆幺半群，它们是 [7],这使我们再次面临对这类（多个）图设计算法的需求。

尽管现有关于上下文无关图的许多研究（例如参见 [2, 9, 10, 16, 18]）都源于其前述起源及其在算法半群群论中的应用，但上下文无关图本身也值得作为独立研究对象。事实上，虽然本文的研究动机当然也来自算法逆半群群论，但完全可以脱离该领域的深厚背景来理解。

算法研究上下文无关图面临的首要障碍，是如何将它们有限地编码为算法的输入。Muller 和 Schupp 最初的定义基于图的端锥，或者更宽泛地说，基于它们“在无穷远处”的行为。尽管这似乎无法引出对这些图的有限描述，但他们证明，上下文无关图也可以作为下推自动机（PDA）的配置图出现（更多细节请直接参见 [13] ）。然而，下推自动机在算法处理方面本身就颇具挑战，因此更简单的描述会更有用。

本文将重点研究上下文无关树这一子类。这一研究动机来自两个方面：首先，这类树会作为 Schützenberger 图出现，例如在自由逆幺半群中表现为有限的 *Munn* 树 [14] ，也会在某些其他有限呈示的逆幺半群中表现为无限树 [11] （其中 [7]）。其次，所有上下文无关图都是类树的，也就是说，它们与树拟等距，等价地，它们具有强树分解 [1, 7]。从某种意义上说，上下文无关图似乎由其底层的树“支配”，因此进一步研究这一点似乎很有意义。

我们证明，上下文无关树可以用多边非确定有限自动机进行有限描述。这种描述的有限状态特性使其适合算法处理，并且有理由认为，它事实上是这类对象的自然编码。我们进一步证明，对于确定性上下文无关树，这种描述是一个部分确定有限自动机（pDFA）（其中找不到任何由 $aa^{-1}$ 标记的路径）。从上述代数背景来看，确定性情形尤其值得关注，因为作为 Cayley 图或 Schützenberger 图出现的上下文无关图总是确定性的。我们证明，利用这种 pDFA 描述，确定性上下文无关树的同构问题是可判定的，而且实际上是 NL 完全的。这不仅适用于有根同构，也适用于更复杂的无根情形。

本文的研究结果很可能可作为起点，用于研究更一般的上下文无关图的有限状态描述，并探讨同构问题之外的其他算法问题。当前结果还可为另一个方向的研究铺路，即计算上下文无关树（或更一般的图）的同构（

，因而也包括自同构）的描述。此类描述将很有用，因为 Schützenberger 图的自同构群恰好是某个逆幺半群的极大子群 [19] （这是一个正在积极研究的领域，参见例如 [6]）。

## 2 预备知识

**复杂性理论**。 我们需要了解复杂性理论中的基本概念。特别是，我们需要了解复杂性类 NL，即可通过非确定性对数空间求解的问题类。我们还需要了解 NL 完全的概念（这里使用由 LogSpace 转换器计算的多对一 LogSpace 归约）。有关这些内容的背景知识，请读者参阅该主题的任何标准教材（例如 [15]）。

**字母表与单词**。 字母表 是由 $A$ 个字母组成的集合。对合字母表 是配备了一个对合 $\cdot^{-1}\colon A \to A$ 的字母表 $A$ （其中 $a^{-1} = a$ 是可能的）。对于每个（普通）字母表 $A$，我们可以附加上对合字母表 $A^{\pm 1} = A \uplus A^{-1}$，其中 $A^{-1} = \{a^{-1} \mid a \in A\}$ 是一个不相交副本，且对合为 $a^{-1} = a$ 和 $(a^{-1})^{-1} = a$。

字母表 $A$（是否带有对合）上的单词 $w$ 是由字母组成的有限序列 $w = a_1 \dots a_\ell$ $a_1, \dots\dots, a_n \in A$。其长度 为 $\ell$，而空 单词（即长度为 0 的唯一单词）记作 $\varepsilon$。 $A$ 上的所有单词构成的集合记作 $A^*$，该集合具有一种自然的单词连接运算。我们用 $A^\ell$ 表示 $A$上长度恰为 $\ell$ 的单词集合，并采用这种记法的自然变体，例如用 $A^{\leq \ell}$ 表示长度至多为 $\ell$ 的单词集合。若存在某个 $v$ 使得 $w = uv$，则单词 $u$ 是另一个单词 $w$ 的前缀 。最后，（形式）语言 是某个 $A$ 的 $A^*$ 子集。

**图**。 设 $A$ 为一个字母表。一个 $A$-图 是一个有向的、边由 $A$ 标记的图 $\Gamma = (V, E)$，其节点集为 $V$，边集为 $E \subseteq V \times A \times V$。在讨论一条边时，我们记作$u \xrightarrow{a} v$，也记作 $v \xleftarrow{a} u$，表示元素 $(u, a, v) \in V \times A \times V$。这样的边起始于 $u$，终止于 $v$ ，并由 $a$标记 。一个顶点的出度 （入度）是以该顶点为起点（终点）的边集的基数。若存在一个有限数，能够界定一个 $A$-图中每个顶点的出度（入度），则称该图的出度（入度）一致有界 。

一条长度为 $\ell \in \mathbb{N}$ 的路径 在 $A$-图 $\Gamma = (V, E)$中是由边 $e_1 \dots e_\ell$ 构成的有限序列，$\dots\dots$，$e_\ell \in E$，满足 $e_i$ 的终点是 $e_{i+1}$的起点。该路径从与 $v_0$ 相同的顶点开始 $e_1$ ，并在与 $v_\ell$ 相同的顶点结束 $e_\ell$；这是一条从 $v_0$ 到 $v_\ell$的路径。如果这些边分别标记为 $a_1, \dots\dots, a_\ell$，则该路径的标签为 $a_1 \dots a_\ell \in A^*$。

两个 $A$-图（或者，如果要强调我们考虑的是 $A$-图，则称为 $A$-同构）之间的同构 $\Gamma = (V, E)$和 $\Delta = (W, F)$是一个双射函数$\varphi\colon V \to W$，满足当且仅当存在一条边 $v_1 \xrightarrow{a} v_2 \in E$ 时，才有一条边$\varphi(v_1) \xrightarrow{a} \varphi v_2 \in F(\{\})$。

有时我们将一个特殊节点称为 $A$-图的根 ，并称其为有根的$A$-图。有根 $A$-图之间的同构必须将根映射到另一个图的根。

我们还将考虑带节点标签的 $A$-图，为其配备一个函数 $q\colon V \to Q$，该函数将节点集映射到某个字母表 $Q$。如果 $\Gamma = (V, E)$ 是一个带有节点标记$p\colon V \to P$的 $A$-图，而 $\Delta = (W, F)$ 是一个带有节点标记 $q\colon W \to Q$的 $A$-图，那么如果存在一个 $A$-同构 $\iota\colon V \to W$ 和一个双射 $\beta\colon P \to Q$，使得 $\beta(p(v)) = q(\iota(v))$，则 $\Gamma$ 和 $\Delta$作为带节点标签的 $A$-图是同构的。

**对合图与上下文无关图**。若一个 $A$-图 $\Gamma = (V, E)$ 的字母表 $A$ 是对合字母表，且满足 $u \overset{a}{\triangleright} v \in E \iff u \overset{a^{-1}}{\triangleleft} v \in E$，则称它为对合图。因此，在对合图中，出度和入度相等，这使我们可以讨论节点的度。我们可以将对合扩展到边集：称边 $e^{-1} = u \overset{a^{-1}}{\triangleleft} v$ 为边 $e = u \overset{a}{\triangleright} v$ 的逆。

通常，我们可以定义一个 $A$-图 $\Gamma$ 的对合闭包 $\Gamma^{\pm 1}$：如果 $A$ 不是对合字母表，我们就改为考虑字母表 $A^{\pm 1}$ 上的 $\Gamma$。在对合字母表上，$\Gamma = (V, E)$ 的对合闭包就是满足 $E^{-1} = \{e^{-1} \mid e \in E\}$ 的 $A$-图 $\Gamma^{\pm 1} = (V, E \cup E^{-1})$。通常，绘制对合图时，我们只为每个对合对画一条边。

边上的对合甚至可以延拓为路径上的对合：从 $u$ 到 $v$ 的路径$\pi = e_1 \dots e_\ell$的逆是从 $v$ 到 $u$ 的、长度相同的路径 $\pi^{-1} = e_\ell^{-1} \dots e_1^{-1}$。若路径 $\pi$ 不能对任意边 $e$ 以及路径 $\varrho_1, \varrho_2$写成 $\pi = \varrho_1 e e^{-1} \varrho_2$，则称其为约化的。

对合图是连通的，如果对任意两个节点 $u$ 和 $v$，都存在从$u$ 到 $v$的路径。由此，我们可以将连通对合图中节点之间的距离度量定义为从一个节点到另一个节点的最短路径的长度。

若我们处理的是这样一个有根的对合 $A$-图 $\Gamma$，则令节点$v$ 的层 $\ell(v)$为其到根的距离。注意，边只能存在于层数相差至多1的节点之间（即连向上一层、同一层或下一层节点的边）。半径为 $n \in \mathbb{N}$ 的圆盘 $\Gamma^{\leq n}$（以根为中心）由层数至多为 $n$的节点组成。我们也可以把这些节点之间的边包含进来，并将其视为一个子图。尽管$\Gamma$ 是连通的，图 $\Gamma_{n+1} = \Gamma \setminus \Gamma^{\leq n}$ 却未必连通。为保持一致，我们也用 $\Gamma_0$ 表示 $\Gamma$ 本身。对于层 $n$上的节点 $v$，我们用 $\Gamma(v)$表示 $\Gamma_n$ 中包含 $v$ 的连通分支（即 $\Gamma(v)$恰好包含那些仅使用层数大于或等于 $n$ 的节点之间的边、从$v$ 可达的节点）。这样的连通分支$\Gamma(v)$称为 $\Gamma$ 的一个端锥，其中层数恰好为 $n$ 的节点是它的前沿点。末端同构是两个端锥 $\Gamma(u)$与 $\Gamma(v)$之间的一个 $A$-同构，它把 $\Gamma(u)$的前沿点映到 $\Gamma(v)$的前沿点上。

一个上下文无关图是一个连通的对合 $A$-图，其字母表为有限对合字母表$A$，并具有一致有界度，且存在一个根，使其端同构类有限。事实证明，这一性质不取决于所选的根 [13, 2。7. 推论]；因此，我们可以将任意顶点用作上下文无关图的根。

**树**。如果每个顶点都是一条从根出发的唯一约化路径的终点，那么有根对合图就是一棵树。我们称这样的图为对合 $A$树，或简称对合树。事实上，这一性质与根无关，因此即使没有指定根，我们也可以谈论对合树。如果一个 $A$-图的对合闭包是一棵树，那么它本身也是一棵树。

在树中，同一层的节点之间不能有边（否则会产生多条从根出发的约化路径）。因此，边只能连接前一层或后一层。同样，我们有：

**事实 2.1.** 在有根对合树 $\Gamma$中，节点 $v$ 是端锥 $\Gamma(v)$的唯一前沿点。

证明。 假设有两个位于同一层 $n$ 的节点 $u$ 和 $v$，它们由一条路径 $\pi_{uv}$连接，且该路径只经过层级为 $n$或更高的节点。不失一般性，我们可以假设这条路径是约化路径。此外，从根到 $u$和 $v$分别存在两条唯一约化路径 $\pi_u$ 和 $\pi_v$。它们的长度都必须为 $n$。 $\pi_u$ 的最后一条边连接 $u$ 与层级为 $n-1$ 的一个节点，因此必定与 $\pi_{uv}$的第一条边不同。换句话说， $\pi_u\pi_{uv}$ 是另一条从 $u$ 到 $v$ 的约化路径，它与 $\pi_v$ 不同（因为长度不同）。 □

由前述事实可立即得出，我们可以将对合树中的端锥视为有根图，而末端同构与有根同构是相同的概念。

# 3 上下文无关树

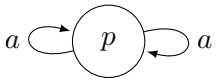
## 3.1 一般情形

**多边非确定有限自动机。** 一个多边非确定有限自动机（简称$mNFA$ ）是一个有限的 $A$-边标记多重图。形式上，我们将其视为一个元组 $\mathcal{A}=(Q,\ A,\ T,\ \alpha,\ \lambda,\ \omega)$，其中 $Q$ 是一个有限状态集， $A$ 是一个有限字母表， $T$是一个有限转移集， $\alpha,\ \omega\colon T\to Q$ 是函数，分别给出转移 $\tau\in T$的起始$\alpha(\tau)$和结束 $\omega(\tau)$，而 $\lambda\colon T\to A$是描述转移 $\tau\in T$ 标签 $\lambda(\tau)$的函数。

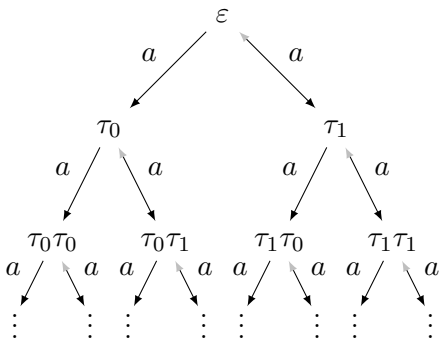
对于具有 $\alpha(\tau)=p$、 $\lambda(\tau)=a$和 $\omega(\tau)=q$的转移 $\tau$，我们也使用图示记法 $\tau=p\ \overset{a}{\blacktriangleright}\ q$。此外，我们采用绘制有限自动机的常见方式，但需注意，任意两个状态之间都可能有多条带 $a$标签的边。

mNFA $\mathcal{A}$的一次运行 是一个转移序列 $\varrho=\tau_1\dots\tau_\ell\ \tau_1,\ \dots\dots,\ \tau_\ell\in T$，其中$\omega(\tau_i)=\alpha(\tau_{i+1})$（对所有 $i\le i<\ell$均成立）。它从 $\alpha(\varrho)=\alpha(\tau_1)$开始，在 $\omega(\varrho)=\omega(\tau_\ell)$结束，其长度 为 $\ell$，并且标签 为 $\lambda(\varrho)=\lambda(\tau_1)\dots\lambda(\tau_\ell)$。如果存在一次运行从状态 $p$ 开始并在 $q$结束，那么状态 $q$就称为从状态 $p$ 是可达 或可访问的。如果所有状态都能从某个状态到达，那么该状态称为$\mathcal{A}$ 的根 。

对于 mNFA $\mathcal{A}=(Q,\ A,\ T,\ \alpha,\ \lambda,\ \omega)$ 的每个状态 $p\in Q$，我们都可以关联一棵从 $p$ 开始的计算/运行树 $\mathcal{A}$。形式上，我们考虑一个 $A$-图，其节点集 $V$ 是从 $p$ 开始的 $\mathcal{A}$ 的运行集合。若对于某个转移 $\tau\in T$， $\pi=\varrho\tau$，则从 $\varrho$ 到 $\pi$ 有一条边。该边的标签为 $\lambda$（$\tau$）。我们将空路径 $\varepsilon$ 视为该图的根，并注意到所有边都从该根向外定向。我们可以假设 $A$ 是一个对合字母表（必要时转换为 $A^{\pm1}$），并通过添加所有逆边得到一个对合 $A$-图。容易验证，这个对合 $A$-图是一棵树，我们将其记为 $\Gamma$（$p$）。 $\Gamma$（$p$）中的度一致有界于 $|T|$。




(a) 一个非常简单的 mNFA，只有一个状态和两个自环 $\tau_0$ 和 $\tau_1$。




(b) 伴随树 $\Gamma(p)$。所有节点均标记为 $p$。每对对合对中只绘制一条边；方向为黑色方向。

图1：带有关联树的 mNFA 示例。

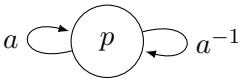
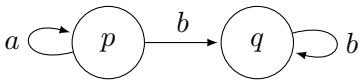
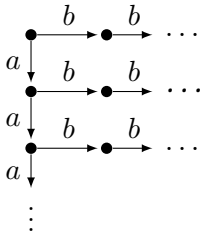



图2：一个 mNFA，适用于图1b并带有灰色定向。




(a) 一个具有两个状态的 mNFA。




(b) 图 $\Gamma(p)$。左侧的节点标记为 $p$，所有其他节点标记为 $q$。

图3：另一个带有关联树的 mNFA 示例。

我们可以为该图赋予节点标签：令 $\varrho \in T^*$ 为 $\mathcal{A}$ 从 $p$开始的一次运行（即 $\Gamma(p)$ 的一个节点）。该次运行最终到达某个状态 $q$，我们将此状态用作 $q(\varrho) = q$ 在 $\varrho$ 中 $\Gamma(p)$ 的标签。读者可在 图1、图2和图3中看到两个 mNFA 及其对应关联树的示例。

有时，使用 $\Gamma(p)$ 的递归描述会更方便。对于一个带有 $V \subseteq T^*$ 和 $\tau \in T$ 的 $A$-图 $\Gamma = (V, E)$，令 $\tau\Gamma$ 为节点集为 $\tau V = \{\tau v \mid v \in V \subseteq T^*\}$、边集为 $\tau E = \{\tau u \overset{a}{\blacktriangleright} \tau v \mid u \overset{a}{\blacktriangleright} v \in E\}$ 的 $A$-图。若 $\Gamma$ 有一个根 $r$，我们将 $\tau r$ 记作 $\tau\Gamma$ 的根；对于节点标记 $q\colon V \to Q$，令 $\tau q\colon \tau V \to Q$，其中 $\tau q(\tau v) = q(v)$。显然，作为一个（有根/带节点标签的） $A$-图，$\Gamma$ 与 $\tau\Gamma$ 同构（通过 $A$-同构 $v \mapsto \tau v$）。

令 $\{\tau_1, \dots, \tau_n\}$ 为从 $p$开始的转移集合。我们可以写作 $\tau_i = p \overset{a_i}{\blacktriangleright} q_i$(对于 $i = 1$、.....、 $n$)，并注意到 $\Gamma(p)$ 是由并集 $\bigcup_{i=1}^{n} \tau_i\Gamma(q_i)$ 加上额外节点 $\varepsilon$ 以及额外边 $\{\varepsilon \overset{a_i}{\blacktriangleright} \tau_i \mid i = 1, \dots, n\}$ 及其逆边（其中 $\varepsilon$ 标记为 $p$）所产生的 $A$-图。

**正则树**。借助 mNFA，我们可以定义什么是正则树。

**定义 3.1**。如果一个有根、带节点标签的对合 $A$-树同构于某个 mNFA（其字母表为$A$）中某个状态 $p$ 的 $\Gamma(p)$（作为有根、带节点标签的 $A$-图），则称它为正则。如果某个有根对合 $A$-树对于某种节点标记是正则的，则称其为正则；如果某个对合 $A$-树在选取某个顶点作为根后是正则的，则称其为正则。

事实证明，正则性与所选的根无关，因此对于正则的对合 $A$-树，我们可以选择任意节点作为根：

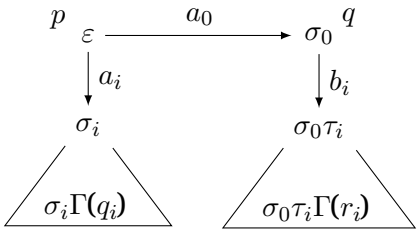
**命题 3.2.** 设 $\Gamma$ 是一棵对合 $A$-树，$u$ 和 $v$ 是其中的两个节点。那么，$\Gamma$ 以 $u$ 为根时是正则的，当且仅当以 $v$ 为根时也是正则的。

证明。只需证明距离为 1 的节点 $u$ 和 $v$ 的情形（其余情形随后可由归纳法推出）。

不妨设 $\Gamma$ 等于某个 mNFA $\mathcal{A} = (Q, A, T, \alpha, \lambda, \omega)$ 中某个状态 $p$ 的 $\Gamma(p)$。设 $\sigma_0$ 是 $\Gamma(p)$ 中根 $\varepsilon$ 的一个邻点；也就是说，我们有 $T \ni \overline{\sigma_0} = p \xrightarrow{a_0} q_0$。我们需要证明，如果将根改为 $\sigma_0$，那么所得的图作为有根的 $A$-图，与某个（可能不同的）mNFA $\mathcal{A}'$ 中某个状态 $q'$ 的 $\Gamma(q')$ 同构（这也会产生新的节点标记）。

令 $\sigma_0$、……、$\sigma_m$ 表示从 $p$ 开始的 $\mathcal{A}$ 的转移，并将 $\sigma_i = p \xrightarrow{a_i} q_i$ 写作（对于 $i = 0$、……、$m$）。此外，令 $\tau_1$、……、$\tau_n$ 表示从 $q_0$ 开始的转移，并将 $\tau_i = q_0 \xrightarrow{b_i} r_i$ 写作（对于 $i = 1$、……、$m$）。

根据 $\Gamma(p)$ 的递归描述，我们现在可以按如下方式描绘它（其中每个对合边对只画一条边）：

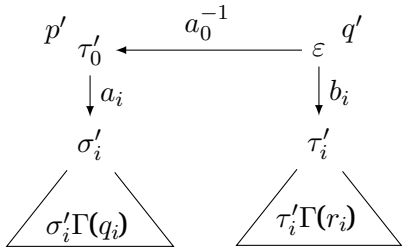



为了构造 $\mathcal{A}'$，我们从 $\mathcal{A}$ 开始，添加新状态 $p'$ 和 $q'$，以及以下两两不相交的转移：

$$\sigma'_i = p' \xrightarrow{a_i} q_i \text{ for } i = 1, \ldots, m \text{ (but not for } i = 0),$$
$$\tau'_0 = q' \xrightarrow{a_0^{-1}} p' \text{ and}$$
$$\tau'_i = q' \xrightarrow{b_i} r_i \text{ for } i = 1, \ldots, n$$

实际上，我们将状态 $p$ 复制为 $p'$，保留了它几乎所有的转移，但排除了 $\sigma_0$。类似地，我们将 $q_0$ 复制为 $q'$，并向 $p'$ 添加了一条额外的 $a_0^{-1}$-转移。

现在我们得到如下的 $\Gamma(q')$（同样，每个对合对只画一条边）：




作为一个对合的有根 $A$图，它同构于 $\Gamma(p)$，只要将那里的根改为 $\sigma_0$。因此，将根改为一个邻点后，$\Gamma(p)$仍然是正则的。 □

事实上，我们得到，对于树而言，正则与上下文无关是等价的：

**定理 3.3.** 对合树是正则的，当且仅当它是上下文无关的。

证明。设 $\Gamma$ 是一棵对合 $A$ 树。作为一棵树，$\Gamma$ 是连通的。

首先，假设它是正则的，这意味着，对于某个根和某种节点标记，它同构于 $\Gamma(q_0)$（作为一张有根、带节点标签的 $A$ 图），其中 $q_0$ 是字母表为 $A$ 的某个 mNFA 的状态。只需证明 $\Gamma(q_0)$ 是上下文无关的。作为 mNFA 的字母表，$A$ 是有限的，并且 $\Gamma(q_0)$ 中任意节点的度显然受 mNFA 中转移数（有限）的限制。因此，只需证明 $\Gamma(q_0)$ 具有有限个端同构类。然而，对于节点 $u$，$\Gamma(u)$ 的端同构类由 $u$ 的节点标签唯一确定。具体而言，我们声称，若 $u$ 标记为状态 $p$，那么 $\Gamma(u) = u\Gamma(p)$，且 $u$ 是这个端锥唯一的前沿点。由于 $u\Gamma(p)$ 作为有根图同构于 $\Gamma(p)$，这便给出了一个末端同构。

该命题本身可以通过对 $\Gamma(q_0)$的递归描述使用归纳法来证明。由定义，该陈述对根 $u = \varepsilon$（其标签为 $q_0$）成立。假设我们已经证明了 $\Gamma(u) = u\Gamma(p)$，其中 $p$ 是 $u$ 的标签，且 $u$ 是唯一的前沿点。我们需要证明，对任意转移$\tau = p \xrightarrow{a} q$（其中 $a \in A$），有 $\Gamma(u\tau) = u\tau\Gamma(q)$。我们从 $\Gamma(u)$得到 $\Gamma(u\tau)$，方法是从中移除与 $u$ 处于同一层的所有节点（这些是前沿点；即我们只移除 $u$），并选取包含 $u\tau$的连通分支。由于 $v \mapsto uv$ 是从 $\Gamma(p)$到 $\Gamma(u) = u\Gamma(p)$的 $A$-同构，我们可以应用 $\Gamma(p)$的递归描述。设 $\tau_1$，……，$\tau_d$ 为从 $p$ 出发的转移，并记$\tau_i = p \xrightarrow{a_i} q_i$（对于 $i = 1$，……，$d$）。我们得到 $\Gamma(u)$等于 $\bigcup_{i=1}^{d} u\tau_i\Gamma(q_i)$，并附加节点 $u$，该节点通过一条以 $a_i$为标签的边（及其逆）与每个 $u\tau_i$ 相连。特别地，所有在不同的 $u\tau_i\Gamma(q_i)$中起始并结束的路径都必须经过$u$，并且根节点 $u\tau_i$ 是唯一与 $u$ 直接连通的节点（即下一层的节点）。这就证明了该命题。

对于逆向推导，设 $\Gamma$ 是上下文无关的。特别地，$A$ 是有限的，并且存在一个有限数 $D$，它是任意节点度的上界。我们还可以选取一个根 $r$，使其具有有限个端同构类。对于这些类，我们可以选取 $\Gamma(r_1), \ldots, \Gamma(r_n)$作为代表元。不失一般性，假设 $r = r_1$。我们为 $\Gamma$ 赋予节点标记，将节点 $v$映射到 $i \in \{1, \ldots, n\}$，其中 $\Gamma(v)$ 与 $\Gamma(r_i)$ 端同构。

我们构造一个字母表为 $A$、状态集为 $\{1, \ldots, n\}$ 的 mNFA，使得 $\Gamma(i)$ 与 $\Gamma(r_i)$ 作为有根、带节点标签的 $A$-图同构（即端同构）。特别地，$\Gamma(1)$ 与 $\Gamma(r_1) = \Gamma(r) = \Gamma$ 同构，因此它是正则的。

对于转移，考虑某个 $i \in \{1, \ldots, n\}$ 对应的 $\Gamma(r_i)$。设 $e_1$、……、$e_d$ 是从 $r_i$ 出发、位于 $\Gamma(r_i)$ 内部且带有 $e_k = r_i \xrightarrow{a_k} v_k$ 的边。此外，设对于每个 $k = 1$、……、$d$，$j_k$ 是 $v_k$ 的标签（即 $\Gamma(v_k)$ 与 $\Gamma(r_{j_k})$ 端同构）。对于每条这样的边，我们都会向 mNFA 添加一条转移 $\tau_k = i \xrightarrow{a_k} j_k$（如果有多条边带有某个 $a$ 标签且指向带有某个 $j$-标签的节点，则可能会产生多重边）。

这样构造出的 mNFA 具有上述性质（我们可以通过对 $\ell$归纳证明：对于所有 $\ell \geq 0$和 $i \in \{1, \ldots, n\}$，$\Gamma(r_i)$中半径为 $\ell$的圆盘与 $\Gamma^{\leq \ell}(i)$同构；证明依据是 mNFA 的构造以及 $\Gamma(i)$的递归描述）。 □

注 3.4. 定理 3.3特别指出，任意上下文无关对合树都与 $\Gamma(p)$ 同构，其中 $p$ 是某个 mNFA 的状态。这为上下文无关树提供了一种有限描述；此外，我们还可以假设 $p$ 是 mNFA 的根（因为可以删去任何不可达状态，而这不会改变 $\Gamma(p)$）。有人可能会认为，这确实是自然的有限描述上下文无关树的方式；每当我们讨论算法时，都将采用这种对上下文无关对合树的描述。

## 3.2 确定性情形

确定性 $A$-图和部分确定有限自动机。若 $\Gamma = (V, E)$ 是一个 $A$-图，则称其为确定性的，如果任意一对边 $u \xrightarrow{a} v \in E$ 和 $u \xrightarrow{a} v' \in E$ 都蕴含 $v = v'$。部分确定有限自动机（简称 pDFA）是某个有限字母表 $A$ 上的有限确定性 $A$-图；见下文的例 3.8。pDFA 的节点通常称为状态，$A$ 称为其字母表，其边称为转移。在 pDFA 的语境中，$\mathcal{A}$ 中的有限路径也称为 $\mathcal{A}$ 的一次运行。

对于 pDFA $\mathcal{D} = (Q, T)$ 中的任意状态 $p$，若其字母表为 $A$，我们将 out $p = \{a \mid p \xrightarrow{a} q \in T\} \subseteq A$ 定义为可从 $p$ 读出的字母集合，并令 $p \cdot a \in Q$ 为从 $p$ 读取一个 $a \in$ out $p$ 后到达的唯一状态。

我们可以将部分确定有限自动机 $\mathcal{D}$ 视为 mNFA 的特殊情形：令 $\mathcal{A} = (Q, A, T, \alpha, \lambda, \omega)$，其中 $\alpha(p \xrightarrow{a} q) = p$、$\lambda(p \xrightarrow{a} q) = a$ 以及 $\omega(p \xrightarrow{a} q) = q$。这给出了属于部分确定有限自动机的 mNFA 的描述：

**事实 3.5。** *一个 mNFA $\mathcal{A} = (Q, A, T, \alpha, \lambda, \omega)$ 来自部分确定有限自动机，当且仅当对于所有 $\tau, \tau' \in T$ 满足 $\alpha(\tau) = \alpha(\tau')$ 以及 $\lambda(\tau) = \lambda(\tau')$ 时，都有 $\tau = \tau'$。*

*证明。* 刚刚定义的将部分确定有限自动机解释为 mNFA 的方式显然满足所需条件。反过来，$T$ 通过以下方式单射映射到 $Q \times A \times Q$：$\tau \mapsto \alpha(\tau)\ \omega(\tau)^{\lambda(\tau)}$。令 $T'$ 为该映射的像，并注意到 $(Q, T')$ 不仅是一个 $A$-图，甚至还是一个确定性的图。该部分确定有限自动机恰好产生原来的 mNFA（根据上述解释）。 □

对于部分确定有限自动机（在 $A$ 上）的任意状态 $p \in Q$，我们可以定义其 语言 $\mathscr{L}(p)$ $p$ a s

$$\mathscr{L}(p) = \{w \in A^* \mid w \text{ labels a run starting in } p\}$$

并注意到：

**事实 3.6。** $\mathscr{L}(p) = \{\varepsilon\} \cup \bigcup_{a \in \text{输出 } p} a\mathscr{L}(p \cdot a)$

对于一般的 mNFA，我们必须考虑所有运行，才能描述属于其状态的树 $\Gamma(p)$。对于部分确定有限自动机，起始于状态 $p$ 的转移与 out $p$ 一一对应，因此我们可以将 $\Gamma(p)$ 写成 $a\Gamma(p \cdot a)$ 的并集的对合闭包，其中 $a \in$ out $p$，并加入额外的节点 $\varepsilon$（标记为 $p$）和额外的边 $\{\varepsilon \xrightarrow{a} a \mid a \in \text{out } p\}$（其中 $a\Gamma(p \cdot a)$ 的定义与 mNFA 情形下 $\tau\Gamma(q_i)$ 的定义类似）。在由此得到的树中，任意节点都来自 $A^*$，其层由它作为 $\Gamma$ 上单词的长度决定

。边 $u \xrightarrow{a} v$ 只存在于长度恰好相差一的节点 $u$ 和 $v$ 之间。两个节点中较短的一个是另一个节点的父节点，较长的一个则是子节点。对这些关系进行迭代，便可得到祖先和后代的概念：当且仅当 $u$ 是 $v$ 的前缀时，$u$ 是 $v$ 的祖先；反过来，当且仅当 $v$ 是 $u$ 的后代时，情况也是如此。$v$ 在 $\Gamma(p)$ 中的端锥由其中来自 $vA^*$ 的节点构成。稍后我们将用 $\Gamma(p) \setminus vA^*$ 表示从 $\Gamma(p)$ 中移除属于 $v$ 端锥的所有节点（以及与它们关联的边）后得到的图。

由此不难看出，$\Gamma(p)$ 完全由 $\mathscr{L}(p)$ 描述：

**事实 3.7**。设 $p$ 是部分确定有限自动机 $\mathcal{A}$ 的一个状态，$q$ 是部分确定有限自动机 $\mathcal{B}$的一个状态。那么，有根对合图 $\Gamma(p)$ 与 $\Gamma(q)$ 同构（不考虑节点标签），当且仅当 $\mathscr{L}(p) = \mathscr{L}(q)$。

证明。我们用 $\Gamma \check{} = \Delta$ 表示 $\Gamma$ 和 $\Delta$ 作为有根对合图（不带节点标签）同构。我们还用 $\Gamma^{\leq \ell}(p)$（$\Gamma^{\leq \ell}(q)$）表示 $\Gamma(p)$（$\Gamma(q)$）的半径为 $\ell$的圆盘，并证明对所有 $\ell$，$\Gamma^{\leq \ell}(p) \check{} = \Gamma^{\leq \ell}(q)$ 当且仅当 $\mathscr{L}(p) \cap A^{\leq \ell} = \mathscr{L}(q) \cap A^{\leq \ell}$，此结论由归纳法得出。

对于 $\ell = 0$，$\Gamma^{\leq 0}(p)$ 和 $\Gamma^{\leq 0}(q)$ 分别只由 $p$ 和 $q$组成，因此二者同构。同理，我们有 $\mathscr{L}(p) \cap A^{\leq 0} = \{\varepsilon\} = \mathscr{L}(q) \cap A^{\leq 0}$。

对于从 $\ell$ 到 $\ell + 1$ 的归纳步骤，我们注意到，$\Gamma^{\leq \ell+1}(p)$ 由所有 $a \in$ 出 $p$ 对应的并集 $a\Gamma^{\leq \ell}(p \cdot a)$ 以及带有边 $\{\varepsilon \xrightarrow{a} a \mid a \in$ 出 $p\}$ 的附加节点 $\varepsilon$ 组成（并且对于$\Gamma^{\leq \ell+1}(q)$ 也有类似的结论）。因此，$\Gamma^{\leq \ell+1}(p) \check{} = \Gamma^{\leq \ell+1}(q)$ 当且仅当 $\mathsf{out}\ p = \mathsf{out}\ q$，且对于所有 $a \in$ 出 $p = \mathsf{out}\ q$，$a\Gamma^{\leq \ell}(p \cdot a) \check{} = \Gamma^{\leq \ell}(p \cdot a) \check{} = \Gamma^{\leq \ell}(q \cdot a) \check{} = a\Gamma^{\leq \ell}(q \cdot a)$。由归纳法可得，$\Gamma^{\leq \ell}(p \cdot a) \check{} = \Gamma^{\leq \ell}(q \cdot a)$ 当且仅当$\mathscr{L}(p \cdot a) \cap A^{\leq \ell} = \mathscr{L}(q \cdot a) \cap A^{\leq \ell}$（其中 $a \in$ 出$p \cap \mathsf{out}\ q$），而这当且仅当 $a\mathscr{L}(p \cdot a) \cap A^{\leq \ell+1} = a\mathscr{L}(q \cdot a) \cap A^{\leq \ell+1}$。

综上可得，$\Gamma^{\leq \ell+1}(p) \check{} = \Gamma^{\leq \ell+1}(q)$ 当且仅当 $\mathsf{out}\ p = \mathsf{out}\, q \,\mathsf{and}\, a\mathscr{L}(p \cdot a) \cap A^{\leq \ell+1} = a\mathscr{L}(q \cdot a) \cap A^{\leq \ell+1}$ 对所有 $a \in \mathsf{out}\ p = \mathsf{out}\, q$ 都成立。

如果两个条件都成立，则可推出

$$\begin{aligned}
\mathscr{L}(p) \cap A^{\leq \ell+1} &= \{\varepsilon\} \cup \bigcup_{a \in \mathsf{out}\, p} \left( a\mathscr{L}(p \cdot a) \cap A^{\leq \ell+1} \right) \\
&= \{\varepsilon\} \cup \bigcup_{a \in \mathsf{out}\, q} \left( a\mathscr{L}(q \cdot a) \cap A^{\leq \ell+1} \right) \\
&= \mathscr{L}(q) \cap A^{\leq \ell+1}.
\end{aligned}$$

反过来，假设 $\mathscr{L}(p) \cap A^{\leq \ell+1} = \mathscr{L}(q) \cap A^{\leq \ell+1}$。特别地，这意味着 $\mathsf{out}\, p = \mathscr{L}(p) \cap A = \mathscr{L}(q) \cap A = \mathsf{out}\ q$。通过分别将 $\mathscr{L}(p)$ 和 $\mathscr{L}(q)$ 与 $aA^{\leq \ell} \subseteq A^{\leq \ell+1}$ 相交，我们得到对于每个 $a \in \mathsf{out}\ p = \mathsf{out}\, q$，都有 $a\mathscr{L}(p \cdot a) \cap A^{\leq \ell+1} = a\mathscr{L}(q \cdot a) \cap A^{\leq \ell+1}$。□

**约化部分确定有限自动机**。人们很容易以为，部分确定有限自动机的状态 $p$ 对应的图 $\Gamma(p)$ 恰好属于确定性正则树。然而，$\Gamma(p)$ 的对合性要求部分确定有限自动机还满足另一项性质。

考虑一个以有限字母表 $A$ 为字母表的部分确定有限自动机。我们总可以假设 $A$ 是对合的（必要时通过转到 $A^{\pm 1}$），并称某个状态的语言是约化的，如果它不包含任何以 $aa^{-1}$（对任意 $a \in A$）为因子的单词 $w$。若一个部分确定有限自动机的所有状态的语言都是约化的，则称该部分确定有限自动机是约化的；这等价于它不包含任何标记为 $aa^{-1}$（对任意 $a \in A$）的路径。

注释。请注意，在约化的部分确定有限自动机中，不可能存在标记为 $aa^{-1}$（或 $aa^{-1}$）的路径，但完全可能存在一个具有两条不同入向 $a$-转移的状态！因此，作为部分确定有限自动机是约化的，与具有确定性对合闭包并不相同。

**例 3.8**。图1 中的自动机 不是确定性的，因此不是部分确定有限自动机；与之关联的树也不是确定性的。图2 中的自动机 是确定性的，但并未约化（由于对合闭包，与之关联的树仍不是确定性的）。图3 中的自动机 最终是一个约化的部分确定有限自动机，而 $\Gamma(p)$ 是一个确定性对合图。

**定理3.9**。设 $\Gamma$ 是一棵对合树。那么，$\Gamma$ 是确定性的且上下文无关，当且仅当 $\Gamma$ 同构于（作为一棵有根的、无节点标签的 $A$-图）某个约化的部分确定有限自动机中状态 $p$ 对应的 $\Gamma(p)$。

证明。首先，假设 $\Gamma$ 是确定的上下文无关的。根据定理 3.3，$\Gamma$ 于是同构于（作为一个有根的、对合的 $A$-图）$\Gamma(p)$，其中 $p$ 是一个 mNFRA $\mathcal{A}$在有限对合字母表 $A$上。不失一般性，我们可以假设 $\mathcal{A}$ 的所有状态都从 $p$ 可达。

我们首先利用事实 3.5证明 $\mathcal{A}$ 来自一个部分确定有限自动机。考虑 $\mathcal{A}$中的两个转移 $\tau$和 $\tau'$，其中 $\alpha(\tau) = \alpha(\tau') = q$，且 $\lambda(\tau) = \lambda(\tau') = a$。我们需要证明 $\tau = \tau'$。根据我们的假设，状态 $q$ 可通过某个运行 $\varrho$从 $p$ 到达。这意味着，在 $\Gamma(p)$中存在一条从 $\varepsilon$ 到 $\varrho$ 的路径（标记为 $\lambda(\varrho)$）。此外，在 $\Gamma(p)$中还有边$\varrho \xrightarrow{a} \varrho\tau$和$\varrho \xrightarrow{a} \varrho\tau'$。由 $\Gamma(p)$的确定性可得 $\varrho\tau = \varrho\tau'$，因此 $\tau = \tau'$。

还需证明该部分确定有限自动机是约化的（对于证明的这个方向）。反设存在转移 $\tau_1 = q_1 \xrightarrow{a} q_2$和 $\tau_2 = q_2 \xrightarrow{a^{-1}} q_3$（其中 $a$ 取自对合字母表 $A$）。由于我们可以通过运行 $\varrho$从 $p$到达 $q_1$，因此在 $\Gamma(p)$ 中可以得到一条从 $\varepsilon$ 到 $\varrho$ 的路径，以及边 $\varrho \xrightarrow{a} \varrho\tau_1$和 $\varrho\tau_1 \xrightarrow{a^{-1}} \varrho\tau_1\tau_2$。由于 $\Gamma(p)$ 是对合的，我们还必须在 $\Gamma(p)$ 中有边 $\varrho\tau_1 \xrightarrow{a^{-1}} \varrho$，这就产生矛盾，因为 $\varrho \neq \varrho\tau_1\tau_2$ 和 $\Gamma(p)$，因此不可能是确定的。

反过来，考虑约化部分确定有限自动机 $\mathcal{A}$ 中的状态 $p$，其状态集为 $Q$。由于可以将部分确定有限自动机视为 mNFA 的一种特殊情况，由定理 3.3 可知 $\Gamma(p)$ 是上下文无关的，接下来只需证明它是确定的。设 $\Gamma(p)$ 中有边 $v \xleftarrow{a} u$和 $u \xrightarrow{a} v'$。我们需要证明 $v = v'$。有两种可能的情况：$u$ 是 $v$ 和 $v'$的父节点；或者 $v$ 是 $u$ 的父节点，而 $u$ 又是 $v'$ 的父节点（交换 $v$ 和 $v'$ 的情况是对称的）。

在第一种情况下，我们有 $v = ua = v'$（根据为部分确定有限自动机构造 $\Gamma(p)$ 的方式），因此证明完毕。在第二种情况下，设 $q$ 为 $v$的标签，$r$ 为 $u$ 的标签，$q'$ 为 $v'$的标签。由于构造 $\Gamma(p)$ 时存在边 $v \xrightarrow{a^{-1}} v'$（因为 $v$ 是 $u$的父节点），因此必有边 $v \xleftarrow{a} u$。这条边只能由 $\mathcal{A}$中的转移 $q \xrightarrow{a^{-1}} r$ 产生。

类似地，边 $u \xrightarrow{a} v$只能由 $\mathcal{A}$中的转移 $r \xrightarrow{a} q'$产生。这意味着$a^{-1}a \in \mathscr{L}(q)$，但这是不可能的，因为 $\mathcal{A}$ 是约化的。 □

注记3.10。正如 mNFA 的根是对一般上下文无关对合图的一种自然的有限描述，我们可以从定理3.9得知，约化部分确定有限自动机的根是描述确定性上下文无关树的一种有限形式（我们要求该状态是一个根，因为部分确定有限自动机中不可达的部分不会影响生成树）。从某种意义上说，与描述一般上下文无关图的 mNFA 相比，这种描述甚至更加自然，因为它确实涵盖了树具有确定性这一事实。因此，每当我们需要以算法方式编码确定性上下文无关树时，都会采用这种描述。

# 4 同构问题

## 4.1 有根情形

我们首先证明，确定性上下文无关树的有根同构问题[1]

**输入：** 一个以 $\check{}p$ 为根的约化部分确定有限自动机 $\mathcal{A}$，
以及一个以 $\check{}q$为根的约化部分确定有限自动机 $\mathcal{B}$

**问题：** $\Gamma(\check{}p)$ 和 $\Gamma(\check{}q)$ 是否作为有根对合图同构?

是非确定性对数空间完全的。事实上，我们证明其补是非确定性对数空间完全的，由此该陈述可由Immerman-Szelepcsényi定理 [8, 20] 得出（另见例如 [15,定理7.6]）。根据事实3.7，这等价于证明检验 $\mathscr{L}(\check{}p) \neq \mathscr{L}(\check{}q)$对同一输入是非确定性对数空间完全的。

对于属于非确定性对数空间，我们首先观察到，可以假设两个输入自动机具有公共字母表 $A$，因为每个在 $A$ 上的部分确定有限自动机特别地也是一个在$A \cup B$上的自动机。于是，属于非确定性对数空间便很容易看出：通过逐字母猜测两个语言的对称差中的一个见证。为此，我们只需存储当前状态 $p$ 于 $\mathcal{A}$ 中以及当前状态 $q$ 于 $\mathcal{B}$ 中，并将它们推进到 $p \cdot a$ 和 $q \cdot a$，针对每个猜测的字母 $a \in A$（若可能）。如果我们到达只能在其中一个（而不能在另一个）中读取 $a$ 的情形，我们就找到了对称差中的一个单词并接受。

对于 NL-困难性，我们给出一个对数空间-归约，将其归约自图可访问性问题的一个变体。这里的主要问题是，我们希望所有状态都可访问；也就是说，输入 $\check{}p$ 和 $\check{}q$ 应当是各自自动机的根（参见 备注 3.10）。为此，我们采用了与 [3, 引理 2。2] 类似的方法。

**命题 4.1**。 确定性上下文无关树的有根同构问题是 *NL*-困难的。

证明。 我们将问题 2GAP 归约

**输入：** 一个有限无标号有向图 $\Gamma$，其节点集为 $V = \{0, \ldots, n-1 \}$
其中每个节点至多有两条出边

**问题：** 在 $\Gamma$ 中，是否存在一条从 0 到 $n-1$ 的有向路径?

[1]回忆一下，自动机的根是这样一种状态：从该状态可以到达所有其他状态。

归约到同构问题的补。可以看出，2GAP 是 非确定性对数空间难的。所用的主归约与经典归约非常相似，后者用于证明有向图上的一般可达性问题是非确定性对数空间难的（例如参见 [15, 定理 16。2]）：我们不必像经典做法那样从任意非确定性对数空间机器出发，再使用它的配置图，而可以假设该机器至多有两个后继配置。显然，与一般的非确定性相比，这并未构成限制，因为只需通过多个步骤即可实现更多的非确定性分支。

对于该归约，考虑输入中给定的某个 $\Gamma$，用于 2GAP。我们不失一般性地假设 $n = 2^\ell$ 是 2 的幂（例如，可以通过添加编号从 1 到所需数量的状态来实现）。我们还可以假设 0 没有入边（因为删除这些边不会影响是否存在从 0 到 $n-1$ 的路径），并且 $n-1$ 没有任何出边（理由相同）。

我们分两步进行归约。首先，通过为所有边添加标签，将 $\Gamma$ 扩展为一个定义在二元字母表 $A = \{0,1\}$ 上的部分确定有限自动机 $\mathcal{A}'$。如果状态 $i$ 有两条出边，分别指向状态 $j$ 和 $k$，其标签分别为 $j \le k$，我们就将指向 $j$ 的边标记为 0，将指向 $k$ 的边标记为 1。如果它只有一条出边，我们就将其复制，并将其中一条标记为 0，另一条标记为 1。如果节点 $i < n-1$ 没有任何出边，我们就添加两个自环 $i \xrightarrow{0} i$ 和 $i \xrightarrow{1} i$。

注意，从 0 开始并在 $n-1$ in $\mathcal{A}'$ 结束的运行存在，当且仅当存在一条从 0 到 $n-1$ in $\Gamma$ 的路径。进一步注意，除了 $n-1$（在该状态下两者都无法读取）以外，在所有状态下都可以读取 0 和 1。因此，$\mathscr{L}(0) = \{0,1\}^*$ 成立，当且仅当不存在从 0 到 $n-1$ in $\Gamma$的路径。

注意，$\mathcal{A}'$ 只使用来自 $A$ 的正字母，因此是约化的（作为对合字母表 $A^{\pm 1}$上的部分确定有限自动机）。另请注意，$\mathcal{A}'$ 显然可以由确定性对数空间转导器计算。

现在，我们可以将 $\mathcal{A}'$ 用作第一个部分确定有限自动机，其状态为 0，并构造一个只有一个状态、带有分别对应 0 和 1 的两个自环的第二个部分确定有限自动机来进行归约；但这样做有个问题：在 $\mathcal{A}'$ 中，$n-1$ 并非总是从 0 可访问，因此 0 并非总是根（而我们要求同构问题的输入满足这一点）。为了解决这个问题，我们进行归约的第二步：首先，将 $\mathcal{A}'$ 中的每个状态 $i$ 重命名为 $i$ 的二进制表示 $\mathsf{bin}_\ell i$，并在前面补前导零，使其长度达到 $\ell$。对于每个 $w \in \{0,1\}^{<\ell}$，我们添加一个新状态 $w$ 以及转移 $w \xrightarrow{0} w0$ 和 $w \xrightarrow{1} w1$。将如此构造的部分确定有限自动机记为 $\mathcal{A}$。注意，$\mathcal{A}$ 的每个状态都可从 $\varepsilon$ 访问，并且 $\mathcal{A}$ 可以在对数空间内构造。还要注意，$\mathcal{A}$ 仍然是约化的。

对于 $\mathcal{B}$，我们使用 $\mathcal{A}$ 的一个副本，并将状态 $\mathsf{bin}_\ell 0 = 0^\ell$ 替换为另一个状态 $f$，其转移为 $f \xrightarrow{0} f$ 和 $f \xrightarrow{1} f$。由手—$\mathcal{A}$ 中的状态为 $p$，而 $\mathcal{B}$ 中的状态为 $q$，两种情况下我们都使用 $\varepsilon$。

现在请注意，语言 $\mathscr{L}(p)$ 和 $\mathscr{L}(q)$ 之间的差异只能来自分别从 $0^\ell$ 和 $f$ 开始的可能运行（它们分别是通过从 $p$ 和 $q$ 读取 $0^\ell$ 到达的）。根据构造，$f$ 的语言是 $\{0,1\}^*$，而 $0^\ell$ 的语言是 $\{0,1\}^*$，当且仅当在 $\Gamma$ 中不存在从 0 到 $n-1$ 的路径。因此，$\mathscr{L}(p) \neq \mathscr{L}(q)$ 当且仅当不存在这样的路径，由此证明得证。 □

| 例程 何时返回真？$U(p,\ q)$ 在 $\Gamma(\check{}q)$ 中存在一个标记为 $q$ 的节点 $v$，且满足 $\Gamma(p)\check{}$ |
|---|
| $=\Gamma(\check{}q,\ v)$。$U(p,\ q,\ b_0)$ 在 $\Gamma(\check{}q)$ 中存在一个标记为 $q$ 的节点 $v$，且满足 $\Gamma(p)\check{}=\Gamma$ $(\check{}q,\ v)\setminus vb_0A^*$。 |

表1：子程序 $U(p,\ q)$ 和 $U(p,\ q,\ b_0)$ 的语义

## 4.2 无根情形

接下来我们将证明，确定性上下文无关树的无根同构问题

> **输入：** 一个以状态 $\check{}p$ 为状态的约化部分确定有限自动机 $\mathcal{A}$，
> 以及一个以状态 $\check{}q$ 为状态的约化部分确定有限自动机 $\mathcal{B}$
> **问题：** $\Gamma(\check{}p)$ 和 $\Gamma(\check{}q)$ 是否作为无根对合图同构?

同样是 NL 完全的。即使额外假设各自自动机的所有状态都可从 $\check{}p$ 和 $\check{}q$ 到达，结论也不会改变。

**NL 算法。** 为描述 NL 算法，我们固定两个约化部分确定有限自动机$\mathcal{A}=(P,\ S)$ 和 $\mathcal{B}=(Q,\ T)$（其中 $P\cap Q=\emptyset$），它们定义在同一个有限字母表 $A$ 上（不失一般性；见上文），状态分别为 $\check{}p\in P$ 和 $\check{}q\in Q$。我们还引入符号 $\check{}=$，表示两个 $A$-图作为有根图同构。此外，如果 $v$ 是某个部分确定有限自动机的状态 $s$ 对应的 $\Gamma(s)$ 中的一个节点，我们用 $\Gamma(s,\ v)$ 表示这样一棵（带节点标签的）定根对合树：将根改为 $v$。另外请回忆，$\Gamma(s)\setminus vA^*$ 表示从 $\Gamma(s)$ 中移除 $v$ 的端锥后得到的图（可将其视为位于 $v$“下方”的子树）。

为了在非确定性对数空间中测试 $\Gamma(\check{}p)$ 是否与 $\Gamma(\check{}q)$ 作为非定根图同构，我们引入两个子程序 $U(p,q)$ 和 $U(p,q,b_0)$，其参数为 $p\in P$、$q\in Q$ 和 $b_0\in A$。我们将把这些子程序实现为非确定性对数空间算法，其语义如表1所示（另见图 4）。特别地，如果对于某个新的根 $v$（其标签为 $q\in Q$），$\Gamma(\check{}p)$ 与 $\Gamma(\check{}q)$ 作为非定根树同构，当且仅当 $\Gamma(\check{}p)$ 与 $\Gamma(\check{}q,v)$ 作为有根树同构。因此，在证明表1中的语义之后，我们可知 $\Gamma(\check{}p)$ 与 $\Gamma(\check{}q)$ 作为非定根树同构，当且仅当对于某个 $q\in Q$，$U(\check{}p,q)$ 返回真。由于我们可以非确定性地猜测 $q\in Q$，只要将 $U(p,q)$ 实现为具有所需语义的非确定性对数空间算法，就解决了这个问题。

其实现见算法 1。两个算法非常相似，因此我们将它们列在同一份伪代码清单中。仅取黑色部分，即可得到 $U(p,\ q)$ 的实现；再添加灰色部分，即可得到 $U(p,\ q,\ b_0)$的实现。我们使用两条特殊指令来处理算法中的非确定性：对于有限集 $X$，使用`guess(`$X$`)`以非确定性地猜测一个元素 $x\in X$ (即，对于每个 $x\in X$，都会产生一个新的非确定性计算分支)。使用断言 时，所有不满足其后条件的非确定性分支都会失败。该条件尤其也可以是另一个非确定性对数空间算法。在这种情况下，内层算法终止并接受的分支上的计算将继续。

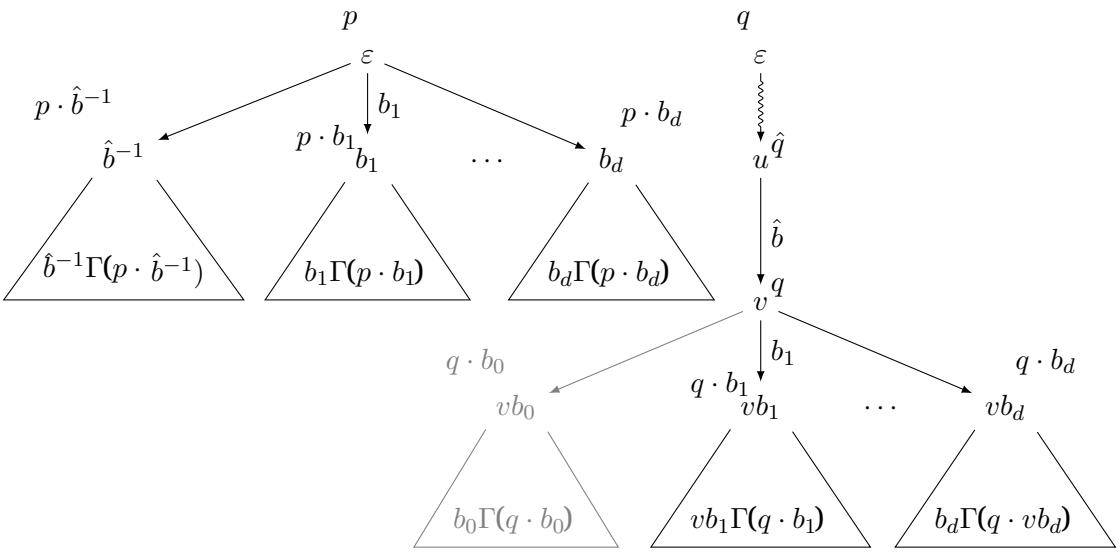



图 4： $U(p,q)$ 和 $U(p,q,b_0)$ 语义的示意图：通过将 $\varepsilon$ 映射到 $v$，左侧的树必须与右侧的树同构。对于 $U(p,q,b_0)$，此同构中忽略灰色子树。

**算法 1** 将 $U(p,q)$ 和 $U(p,q,b_0)$ 实现为非确定性对数空间算法

```
1  fun U(p ∈ P, q ∈ Q, b_0 ∈ A):
2    if q = ε and 猜测({true, false}):                    ▷ 在根节点处，我们没有前驱
3      assert 出p = 出q \ {b_0}                           ▷ 确保 p 处的出边相匹配
4      for a ∈ 出p:
5        assert Γ(p · a) ≐ Γ(q · a)                       ▷确保所有子节点下方的子树都同构
6      return true
7    else:
8      (q̂, b̂) ← 猜测（{(q̂, b̂) | q̂ --b̂--> q 在 T}）          ▷ 猜测 q 的前驱
9      assert 输出p = {b̂^{-1}} ∪ （输出q \ {b_0}）          ▷确保 p 处的出边相匹配
10     for b ∈ （出边q \ {b_0}）:
11       assert Γ(p · b) ≐ Γ(q · b)                     ▷ 确保子节点下方的所有子树都同构
12     return U(p · b̂^{-1}, q̂, b̂)                        ▷ 这是对 b̂^{-1} 下方子树的尾递归
```

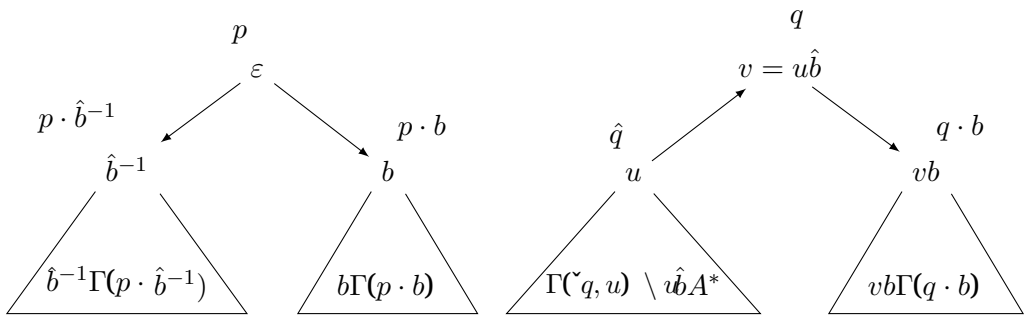



图 5： $\Gamma(p)$ 和 $\Gamma(\check{q},\ v)$ 的绘制。我们拥有所有 $b \in \mathsf{out}\, q$ 的边和子树

在深入探讨算法之前，我们要指出，在任何时刻，我们只需要存储来自 $P$ 和 $Q$以及来自 $A$的数量为常数的状态和字母。特别地，第 12 行 中的递归是尾递归，在此处我们不需要存储任何先前的变量值。由于我们已经知道，检查 $\Gamma(p) \check{=} \Gamma(q)$ 可以通过非确定性对数空间 在第 4.1 小节中实现，这表明该算法只需要对数空间。

理解该算法的最佳方式是查看其正确性证明。为此，我们将计算分支的递归深度定义为该分支上执行第 12 行 的次数。我们给出表1中语义的一个参数化版本：

- 在 $\Gamma(\check{q})$ 中的第 $\ell$ 层上，存在一个标签为 $q$ 的节点 $v$，使得 $\Gamma(p) \check{=} \Gamma(\check{q},\ v)$ 当且仅当 $U(p,\ q)$ 在某个递归深度为 $\ell$ 的计算分支上接受。
- 在 $\Gamma(\check{q})$ 中的第 $\ell$ 层上，存在一个标签为 $q$ 的节点 $v$，使得 $\Gamma(p) \check{=} \Gamma(\check{q},\ v) \setminus vb_0A^*$当且仅当 $U(p,\ q,\ b_0)$ 在某个递归深度为 $\ell$ 的计算分支上接受。

我们首先通过对 $\ell$作归纳法，并行证明两个命题的“仅当”部分。首先，令 $\ell = 0$。假设对于某个 $v$，在 $\Gamma(\check{q})$ 中第 0 层上存在标签为 $q$ 的节点，并且 $\Gamma(p) \check{=} \Gamma(\check{q}, v)$。这意味着 $v = \varepsilon$（即 $\Gamma(\check{q})$ 的根）必须具有标签 $q = \check{q}$，因此 $\Gamma(p) \check{=} \Gamma(\check{q})$。所以，在某个计算分支上，我们进入第 2 行的 if 部分。由于 $\Gamma(p)$ 和 $\Gamma(\check{q})$ 是同构的有根树，因此我们有 $\mathsf{out}\, p = \mathsf{out}\, q$，并且对于每个 $a \in \mathsf{out}\, p = \mathsf{out}\ q$，$\Gamma(p)$ 中的子树 $aA^*$ 与 $\Gamma(\check{q})$ 中的子树同构（作为有根树）。换句话说，两个断言语句均成立，最终我们返回 true（不在第 12 行进行递归）。对于 $U(p, q, b_0)$ 的证明相同，唯一的区别是我们需要忽略 $\Gamma(\check{q})$ 中的子树 $b_0A^*$。

对于从 $\ell - 1$到 $\ell$的归纳步骤，假设对于 $\Gamma(\check{q})$中第 $\ell$层、标签为 $q$的节点 $v$，有 $\Gamma(p) \check{=} \Gamma(\check{q}, v)$。令 $u$ 为 $\Gamma(\check{q})$中 $v$ 的父节点，并设它通过边 $\hat{}u \xrightarrow{b} v$与 $v$ 连通；这意味着 $\overline{v} = u\hat{}b$。它位于第 $\ell - 1$ 层，标签为某个 $\hat{}q$，使得其中存在 $\mathcal{B}$（即 $T$）中的转移 $\hat{}q \xrightarrow{b} q$。在某个计算分支中，我们进入第 2 行 if 语句的 else 分支，并恰好猜出这一组合 $\hat{}q, b$。我们可以如图 5 所示画出 $\Gamma(p)$ 和 $\Gamma(\check{q}, v)$（包括节点标签）。在绘制 $\Gamma(\check{q}, v)$ 时，注意 $u$ 上方和右侧的部分恰好是子树 $vA^* = ubA^*$。树的其余部分则作为子树存在于 $u$下方。对于 $\Gamma(p)$ 的绘制，注意：我们

有 $\Gamma(p)\check{} = \Gamma(\check{}q, v)$，因此 $\hat{}b^{-1} \in$ 出自 $p$（因为作为对合图，$v$ 有一条指向 $u$ 的出边 $\hat{}b^{-1}$），甚至还有出 $p = \{\hat{}b^{-1}\} \cup$ 出 $q$（因为作为同构图，它们在根处必须具有相同的出边）。这也是我们在下一步算法中验证的内容。由于这些树同构，它们各自的子树也必定同构，因此，特别地，对于所有 $q$的出 $b \in$，都有 $b\Gamma(p \cdot b)\check{} = vb\Gamma(q \cdot b)$；算法也会对此进行断言。最后，出于同样的原因，我们还有 $\Gamma(p \cdot \hat{}b^{-1})\check{} = \hat{}b^{-1}\Gamma(p \cdot \hat{}b^{-1})\check{} = \Gamma(\check{}q, u) \setminus u\hat{}bA^*$。由于 $u$ 在 $\Gamma(\check{}q)$ 中位于第 $\ell - 1$ 层，因此根据归纳法，$U(p \cdot \hat{}b^{-1}, \hat{}q, \hat{}b)$ 会在递归深度为 $\ell - 1$的某个计算分支上接受，由此得到 $U(p, q)$ 的整体计算分支，其递归深度为 $\ell$。同样，$U(p, q, b_0)$ 的证明也相同，只是我们需要在所有论证中忽略 $v$ 下方的子树 $vb_0A^*$，因此考虑集合 $\mathsf{out}\, q \setminus \{b_0\}$，而不是 $\mathsf{out}\, q$ 中的所有元素（算法所做的正是这一点）。

另一方向的证明方式相同：假设我们有一条接受的计算分支，其递归深度为 0，针对 $U(p, q)$。这只有在我们进入第 2 行的 if 分支时才会发生。这意味着必须有 $q = \check{}q$，并且我们断言 $\Gamma(p) = \Gamma(\check{}q)$（即，所求节点 $v$ 是根 $\varepsilon$）。确实，$\Gamma(p)$ 中根节点的出边与 $\Gamma(\check{}q)$ 根节点的出边相同，因为计算终止意味着 $\mathsf{out}\, p = \mathsf{out}\ q$。这些边分别到达的子树也同构（作为有根树），因为根据算法的终止条件，对所有 $a \in \mathsf{out}\, p$，都必须有 $\Gamma(p \cdot a)\check{} = \Gamma(\check{}q \cdot a)$。这就证明了所断言的同构关系。

对于归纳步骤，考虑针对 $U(p, q)$ 的一条接受的计算分支，其递归深度为 $\ell > 0$。此时，我们必须进入第 2 行 if 语句的 else 分支。令 $\hat{}q$ 为计算分支上所选的 $q$ 在 $\mathcal{B}$ 中的前驱（转移为 $\hat{}q \xrightarrow{b} q \in T$）。由于有 $\mathsf{out}\, p = \{\hat{}b^{-1}\} \cup \mathsf{out}\ q$，我们可以如图 5 那样画出 $\Gamma(p)$。根据归纳法（使用第 12 行），在 $\Gamma(\check{}q)$ 中，第 $\ell - 1$ 层上存在某个标签为 $\hat{}q$ 的 $u$，满足 $\hat{}b^{-1}\Gamma(p \cdot \hat{}b^{-1})\check{} = \Gamma(p \cdot \hat{}b^{-1})\check{} = \Gamma(\check{}q, u) \setminus u\hat{}bA^*$。我们有 $\hat{}b \in \mathsf{out}\, \hat{}q$，因此，在 $\Gamma(\check{}q)$ 中，存在一个标签为 $q$ 的节点 $v = u\hat{b}$，它是 $u$ 的子节点。注意，$v$ 位于 $\Gamma(\check{}q)$ 的第 $\ell$ 层。根据构造，$v$ 的子节点是节点 $vb$，其对应的子树为 $vb\Gamma(q \cdot b)$，对所有 $b \in \mathsf{out}\, q$ 均如此。换句话说，我们也可以如图 5 那样画出 $\Gamma(\check{}q, v)$。根据归纳法，我们已经得到 $\hat{}b^{-1}\Gamma(p \cdot \hat{}b^{-1})\check{} = \Gamma(\check{}q, u) \setminus u\hat{}bA^*$。此外，根据算法中相应的断言，对于所有 $b \in \mathsf{out}\, q$，我们还有 $b\Gamma(p \cdot b)\check{} = \Gamma(p \cdot b)\check{} = \Gamma(q \cdot b) = vb\Gamma(q \cdot b)$。这表明 $\Gamma(p)$ 与 $\Gamma(\check{}q, v)$ 同构为有根树（如所需）。同样，从 $U(p, q, b_0)$ 的一条终止计算分支中，我们可以得到证明 $\Gamma(p)\check{} = \Gamma(\check{}q, v) \setminus vb_0A^*$ 所需的全部条件，从而完成证明。

NL 困难性。我们通过对第 4.1 小节中同构问题的有根版本进行简单归约，得到 NL 困难性。对于该归约，取状态为 $\check{}p$ 的部分确定有限自动机 $\mathcal{A}$ 和状态为 $\check{}q$ 的 $\mathcal{B}$，并首先向它们的（不失一般性）公共字母表 $A$ 中再添加一个新字母 $\top$（若将其视为对合字母表，则 $\top$ 的逆与 $\top$ 不同）。然后，我们分别向 $\mathcal{A}$ 和 $\mathcal{B}$ 添加新状态 $\check{}p'$ 和 $\check{}q'$，并带有转移 $\check{}p' \xrightarrow{\top} p$ 和 $\check{}p' \xrightarrow{\check{}\top} p$。我们将如此修改后的自动机连同状态 $\check{}p'$ 和 $\check{}q'$ 用作无根同构问题的输入（并注意到它们各自自动机的所有状态均可从这些状态可达）。这

显然可在对数空间内计算。注意，$\Gamma(\check{}p')$ 和 $\Gamma(\check{}q')$ 的根节点分别是其图中唯一具有出 $\top$-边的节点。因此，任何（非定根图的）同构都必须将它们相互映射，从而必定是有根图的同构，证明至此结束。

**致谢**。作者谨此感谢 Mark Kambites，就本文涉及的多个主题进行了讨论。

## 参考资料

[1] Yago Antolín。几乎自由群的凯莱图。Groups Complexity Cryptology，3(2):301–327，2011。doi:10.1515/gcc.2011.012。[2] Tullio Ceccherini-Silberstein 和 Wolfgang Woess。上下文无关群对 I：上下文无关群对与图。European Journal of Combinatorics，33(7):1449–1466，2012。《群、图与语言》。doi:10.1016/j.ejc.2012.03.011。[3]Sang Cho 和 Dung T. Huynh。有限状态自动机问题的并行复杂度。Information and Computation，97(1):1–22，1992。doi:10.1016/0890-5401(92)90002-W。[4] Volker Diekert 和 Armin Weiß。上下文无关群与 Bass-Serre 理论。载于 Juan González-Meneses、Martin Lustig 和 Enric Ventura 编，《自由群与自同构周边的算法及几何主题》，《数学高级课程——巴塞罗那 CRM》，第43–110页。Birkhäuser，2017。doi:10.1007/978-3-319-60940-9_2。[5] Martin J. Dunwoody。有限表示群的可达性。Inventiones mathematicae，81:449–457，1985。doi:10.1007/BF01388581。[6] Robert D. Gray 和 Mark Kambites。有限表示特殊逆幺半群的极大子群。Journal of the European Mathematical Society，2025。doi:10.4171/JEMS/1585。[7] Robert D. Gray、Pedro V. Silva 和 Nóra Szakács。具有双曲和类树 Schützenberger 图的逆幺半群的算法性质。Journal of Algebra，611:651–687，2022。doi:10.1016/j.jalgebra.2022.07.029。[8] Neil Immerman。非确定性空间在补运算下封闭。SIAM J. Comput.，17(5):935–938，1988。doi:10.1137/0217058。[9] Dietrich Kuske 和 Markus Lohrey。凯莱图的逻辑方面：群的情形。Annals of Pure and Applied Logic，131(1):263–286，2005。doi:10.1016/j.apal.2004.06.002。[10] Christian Lindorfer 和 Wolfgang Woess。自避游走的语言。Combinatorica，40:691–720，2020。doi:10.1007/s00493-020-4184-z。

[11] Stuart W. Margolis 和 John C. Meakin。逆幺半群、树与上下文无关语言。美国数学会汇刊，335:259–276，1993。doi:10.1090/S0002-9947-1993-1073775-X。[12] David E. Muller 和 Paul E. Schupp。群、端理论与上下文无关语言。计算机与系统科学杂志，26(3):295–310，1983。doi:10.1016/0022-0000(83)90003-X。[13] David E. Muller 和 Paul E. Schupp。端理论、下推自动机与二阶逻辑。理论计算机科学，37:51–75，1985。doi:10.1016/0304-3975(85)90087-8。[14] W. Douglas Munn。自由逆半群。伦敦数学会会刊，s3-29(3):385–404，1974。doi:10.1112/plms/s3-29.3.385。[15] Christos M. Papadimitriou。《计算复杂性》。Addison-Wesley，1994。[16] Laurent Pélecq。上下文无关图的自同构群。理论计算机科学，165(2):275–293，1996。doi:10.1016/0304-3975(96)00222-8。[17] Michael O. Rabin。二阶理论与无限树上自动机的可判定性。美国数学会汇刊，141:1–35，1969。doi:10.1090/S0002-9947-1969-0246760-1。[18] Emanuele Rodaro。Muller–Schupp 定理的推广与类树逆图。伦敦数学会杂志，109(5):e12903，2024。doi:10.1112/jlms.12903。[19] Joseph B. Stephen。逆幺半群的表示。纯粹与应用代数杂志，63(1):81–112，1990。doi:10.1016/0022-4049(90)90057-O。[20] Róbert Szelepcsényi。用于非确定性自动机的强制枚举法。信息学学报，26:279–284，1988。doi:10.1007/BF00299636。