arXiv:2512.13967v1 [math.GR] 16 Dec 2025

自由群中潜在正元素的增长与语言复杂性

EMMA DINOWITZ、LUCY KOCH-HYDE、SIOBH´AN O'CONNOR 和 ´EAMONN OLIVE

Abstract. 自由群中的一个词若与某个仅由正指数书写的元素在自同构下等价，则称其为"潜在正"。我们将开发自动机来分析潜在正词的性质，并利用它们为含有 2 至 7 个生成元的自由群中潜在正元素的渐近增长给出新的界限。我们证明，对于 F_2 ，这些界限是紧的，从而将增长函数确定到一个常数倍。我们还使用相同的工具证明，某些受限自动机无法识别潜在正元素集合。

1. Introduction

自由群 $F_r = F(x_1、\ldots\ldots、x_r)$ 中的一个词，如果能表示为生成元 $x_1、\ldots\ldots、x_r$ 的乘积且不含逆元，则称为**正**词。

可用单个正关系词给出表示的群称为**正单关系群**。关于这类群，已有若干结果：

- 文献 [1] 证明了正单关系群是剩余可解的。[11] 进一步证明了正单关系群是自由群的可解扩张。[1]
- 文献 [20]证明了满足 $C'(1/6)$ 小消去条件的正单关系群是剩余有限的。

遗憾的是，随着长度增加，随机选取的给定长度的词为正词的概率呈指数级趋于零；长度为 n 的正词数量为 r^n ，而长度为 n 的词总数为 $2r(2r-1)^{n-1}$。

然而，对群表示中的关系词施加自由群自同构，会得到一个同构群。因此，若一个词与某个正词在自同构下对应，则其关联的单关系群具有类似于上述性质。这样的词称为**潜在正**词，最早由 [2]引入。

这引出了我们主要的研究主题：

问题 1.1. F_r中长度为 n 的潜在正词有多少个？也就是说，潜在正词的集合如何增长？

已有研究计算了其他类似词类增长的渐近估计。我们将与某个生成元自同构等价的词称为**本原**词。 [13]给出了 F_2 中本原词的增长为 $\sqrt{3}^n$，而 [12] 给出了当 $r > 2$时， F_r 中本原词的增长为$(2r-3)^n$ 。（这些都是渐近估计。我们将在后文正式定义这类估计

[1][11]还表明，无挠正单关系词群满足一个更强的性质。它们是自由群扩张（$\mathbb{Q}$-可解群）。

是什么。）由于本原词也是潜在正词，因此它们为潜在正词的增长提供了一个初始下界。这也说明，除秩 2 外，潜在正元素的增长明显快于正元素。事实上，随着规模增大，正元素在潜在正元素中的密度趋于零。

本文的主要结果回答了针对 F_2 的问题 1.1。更具体地说，我们证明了以下定理。

定理 1.2。 F_2 中长度为 n 的潜在正词数量为 $\Theta(\lambda_1^n)$，其中 $\lambda_1 \approx 2.50506841362147$ 是多项式的最大根

$$\lambda^4 - 3\lambda^3 + \lambda^2 + \lambda - 1 \tag{1}$$

由此计算我们看到，潜在正词在 F_2 中所有约化词的集合中确实呈指数级消失，但其速率比正词更慢。

我们还给出了 F_r 中 $3 \le r \le 7$ 的潜在正词增长的新下界。据我们所知，对于大于 7 的秩，最好的下界仍然来自 [12]。要进一步推进问题 1.1，可能需要一种能够识别 F_r 中潜在正词的算法；我们在 F_2 中的答案依赖于一种算法，该算法最初由 Goldstein 在 [5], 中针对 F_2 提出，并在 [9] 中得到改进。对于 $r > 2$，是否存在这样的算法尚不得而知，因此我们提出：

问题 1.3. 判断一个词是否具有潜在正性有多复杂？

特别是，我们希望确定哪些类型的自动机能够识别潜在正性。现有算法表明，对于秩为二的情形，任何至少与图灵机同样强大的机器都能判定这一问题。事实上， [9] 给出了一种算法，其最坏情况下使用线性空间，因此秩为二的情形可以由线性有界自动机判定。

本文将从相反的方向探讨这个问题。我们将证明，某些计算模型（包括有限状态机和下推自动机）不足以判定 F_r 对于 $r \ge 2$。

2. Background

2.1. 增长与密度。

本文主要讨论词集的增长。

定义 2.1. 设 S 是有限字母表 Σ上的一个词集。我们将**增长函数** $\mathfrak{g}_n(S)$定义为 S 中长度为 n的词的数量。

我们将定义几种渐近方式来考察这些函数的增长。我们将使用的最精细的增长概念是渐近等价。

定义 2.2。 我们称两个函数 $f, g\colon \mathbb{N} \to \mathbb{N}$ 是**渐近等价**的，记作 $f \sim g$，当

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = 1$$

较粗略的增长概念也将很有用。

定义 2.3. 我们将介绍以下几类 $\mathbb{N} \to \mathbb{N}$ 函数：

- $O(g) = \{f \mid \exists C, m : \forall n \ge m : Cg(n) > f(n)\}$

- $\Omega(g) = \{f \mid g \in O(f)\}$
- $\Theta(g) = \Omega(g) \cap O(g)$

这些概念分别与 $\leq$、 $\geq$和 $=$类似，并构成一个偏序。这些概念中最广为人知的无疑是第一个，即所谓的“大 O”，它通常用于算法分析。我们将使用其中每一种概念。此外，我们也会少量使用与 $<$ 和 $>$类似的类：

- $o(g) = O(g) \setminus \Theta(g)$
- $\omega(g) = \Omega(g) \setminus \Theta(g)$

定义 2.4。集合 $B \subseteq A \subseteq \Sigma^*$ 具有一个**密度函数** 定义为

$$n \mapsto \frac{\mathfrak{g}_n(B) + 1}{\mathfrak{g}_n(A) + 1}$$

定义 2.5。集合 $B \subseteq A$ 在 A 中是**可忽略的**，当密度的极限为 0 时。或者，当密度的极限为 1 时， B 在 A 中是**泛型的**。

一个更强的条件也会有用。

定义 2.6。 设集合 $B \subseteq A$ 的密度函数为 d，当对于某个常数 $C > 1$， $d(n) \in O(C^{-n})$ 时，集合 $B \subseteq A$ 在 A 中是**指数可忽略**的。同样，当其相对补集 $A \setminus B$是指数可忽略的时， B 是**指数泛型**的。

2.2。 **自由群**。 为了增强字母之间的视觉对比，我们通常使用彩色图形来表示自由群的生成元（ 、 、 和 ）。我们用相应图形的轮廓来表示生成元的逆元（ 、 、 ， 分别为 ）

我们将通过仅展示受影响生成元上的作用来表示自同构。例如，在 $F($ 、 $)$ 中，我们使用 $\mapsto$ 来表示执行该映射并在 上保持恒等的自同构。

如果一个词的第一个字母和最后一个字母不是彼此的逆元，我们就称该词是**循环约化**的。同样，我们可以通过不断删除一个词的第一个和最后一个字母，直到其成为循环约化的，来得到该词的**循环约化**。由于循环约化始终是一次共轭，一个词的循环约化始终与该词处于同一个自同构轨道中。

我们将在下面的结果中看到，求出循环约化词的增长率，通常足以确定所有词的增长率。事实上，如果一个词集的增长速度快于其中一个词的共轭类，那么循环约化词的增长就与整个词集的增长相同。我们将在推论 2.8 中看到这一点，但首先需要下面这个技术引理。

引理 2.7。 设有两个函数 f 和 g，以及某个 $b > a > 1$ ，使得f（n） $\in O$（a^n）且 g（n） $\in \Omega(b^n)$ 。粗略地说， g 的指数部分的底大于 f 的指数部分的底。函数 $x \mapsto \sum_{i=0}^{x} f$（$i$）$g$（$x - i$） 属于 Θ（g）。

证明。 下界显然成立，因此我们将继续证明上界。

请注意以下内容。

$$
\begin{aligned}
(1) \qquad O\left(\sum_{i=0}^{x} f(i)g(x-i)\right) &\subseteq O\left(\sum_{i=1}^{x} \frac{f(i)g(x)}{x b^i a^i}\right) \\
&\subseteq O\left(g(x) \sum_{i=1}^{\infty} \frac{a^i}{b^i}\right) \\
(2) \qquad &\subseteq O\left(g(x) \sum_{i=1}^{\infty} \frac{a^i}{b^i}\right) \\
(3) \qquad &= O\left(g\right)
\end{aligned}
$$

(1) 成立，因为 $g(n) \in \Omega(b^n)$ 意味着 $bg(x-1) \leq g(x)$，从而 $g(x-i) \leq b^{-1}g(x)$。(2) 成立，因为 $f(n) \in O(a^n)$。(3) 成立，因为 $b > a$，因此该级数收敛至 $(1-b/a)^{-1}$，这是一个常数。□

推论 2.8。设 S 是 F_r 的一个子集，其中只包含循环约化词。设 S 为循环约化后属于 S的词集。如果 S 中长度为 ℓ 的词的增长为 $g \in \Omega\left(b^{\ell}\right)$，其中 $b > \sqrt{2r-1}$ 为某个数，那么 S 中长度为 ℓ 的词的增长为 $\Theta(g)$。

证明。考虑 S 中长度为 ℓ 且循环约化长度为 ℓ'的词。这些词的形式为 vwv^{-1}，其中 w 属于 S，并且v 与 w 之间以及 w 与 v^{-1}之间均不存在消去。由此可得 $|w| = \ell'$ 和 $|v| = (\ell - \ell')/2$。根据我们的假设，S 中长度为 ℓ' 的词的增长是某个函数 g。只要其最后一个字母不与 w 或 w^{-1}的第一个字母相消，v就可以是长度为$(\ell - \ell')/2$ 的任意词。这类词的增长是某个函数$f(\ell - \ell') \in \Theta\left((2r-1)^{(\ell-\ell')/2}\right) = \Theta\left(\sqrt{2r-1}^{\ell-\ell'}\right)$。因此，这种形式的词的增长为 $g(\ell')f(\ell-\ell')$。于是，S中所有长度为 ℓ 的词集，就是对介于 0 和 ℓ之间的每个 ℓ' 求和。因此，根据引理 2.7，这就是 $\Theta(g)$。□

这一推论适用于本文讨论的所有词集。幸运的是，我们的增长总是显著快于共轭类的增长。不过，也可以用类似的论证证明下面的结论。

推论 2.9. 设 S 是 F_r 的子集，且只包含循环约化词。设 S 是这样一个词集：其中的词经过循环约化后属于 S。如果 S 中长度为 ℓ 的词的增长为 $g \in O\left(b^{\ell}\right)$，其中某个 $b < \sqrt{2r-1}$，那么 $\bar{S}$ 中长度为 ℓ 的词的增长为 $\Theta\left(\sqrt{2r-1}^{\ell}\right)$。

F_2 中的循环约化本原词按多项式增长，因此整个词集的增长率为 $\sqrt{3}$，与一个共轭类的增长率相同。

2.3. **潜在正性**。我们需要一些关于潜在正性的事实，尤其是在 F_2中。首先，我们介绍一个基本但非常有用的事实，它适用于任意秩的自由群中的潜在正性。

命题 2.10. 假设词 w *in* $F_r = F(x_1, \ldots, x_r)$，$r \geq 2$ 中有 $r-1$ 个生成元只以正指数出现。那么 w 是潜在正的。

证明。 不妨假设除 x_r 外的每个生成元都只以正指数出现。于是， x_r^m 中每个 m 为负的音节，都出现在另一个生成元的正幂音节之后。设 w 中 x_r 上出现的负指数的绝对值最大者为 k 。由于 $x_1, \dots, x_{r-1}$ 都只以正指数出现，对每个 $i \in \{1, 2, \dots, r-1\}$ 应用自同构 $x_i \mapsto x_i x_r^{-k}$，都会将 w 映射为一个正词。□

我们将大量运用 Goldstein 的以下定理 [5] （另一种证明见 [9]）。

定理 2.11 (Goldstein 判据)。 设 w 是循环约化且潜在正的。若将 w 视为循环词，则存在 x， $y \in \{\circ, \bullet, \triangle, \blacktriangle\}$ 使得每次出现的 x 前后都有 y。

我们还将引用 [9] 中的以下相关技术引理。

引理 2.12。 若 $w \in F_2$ 是潜在正的，并且满足 *Goldstein* 判据，且 $x =$ 和 $y =$ ，将 $\mapsto$ 作用于 w 会将其映射为一个词，该词满足 *Goldstein* 判据，且 $x =$ 和 $y =$ 或 $x =$ 和 $y =$ 。由对称性，若w 是潜在正的，并且满足 *Goldstein* 判据，且 $x =$ 和 $y =$ ，将 $\mapsto$ 作用于 w 会将其映射为一个词，该词满足 *Goldstein* 判据，且x 和 $y =$ 或 $x =$ 和 $y =$ 。

最后，为简洁起见，我们用简写 PP_r 表示 in F_r中循环约化潜在正词的集合。

3. Automata and matrices

3.1. 自动机与矩阵。 我们构造下界的方法是给出能够生成潜在正词的自动机（即由潜在正词构成的正则语言），并根据自动机计算其增长。我们将自动机 A视为一个有向图，其节点标记为自由群中的字母（即生成元及其逆元）。我们将此标记函数称为 η_A。 η_A 将有向图中的路径映射为自由群中的词。此处**路径** 定义为一列节点（除非另有说明，该序列为有限序列），其中任意两个相邻节点 x 和 y之间都有一条从 x到y的边。将路径视为节点序列后，对序列中的每个节点应用 η_A ，即可得到一个词。

因此，当有向图中没有边连接某个节点与带有其逆标签的节点时，我们称该自动机为**约化的**。如果自动机是约化的，那么任意路径的像都是约化词。

我们尤其关注闭路径及其像。如果存在一条从 y 到 x的边，我们就称从节点 x 到节点 y 的路径是**闭合的** 。我们也将空路径（即不含节点的路径）视为当然闭合。约化自动机中的闭路径必然提升为循环约化词，因为我们要求最后一个字母与第一个字母之间存在一条边。

我们还将定义一种**混合**[2] **自动机**。

定义 3.1。 如果存在某个 N，使得对于任意一对有序节点 x 和 y，以及任意$n \geq N$，都存在一条从 x 到 y、长度为 n 的路径，那么我们称有向图（或更广义地说，自动机）是**混合**的。

[2]我们借用了符号动力系统中的这一术语。参见 [10, 定义 4.5。9]以作比较。

这些自动机对我们很有用，因为下面的定理可以帮助我们计算闭路径的数量：

引理 3.2. 设 G 是一个混合索菲移位，其邻接矩阵的最大特征值为 λ。G 中长度为 n 的闭路径数量为 $\Theta(\lambda^n)$

证明。不可约索菲移位具有右解析表示 $\mathcal{G} = (G', L)$（[10, 定理 3.3.11]），而对于右解析图，长度为 n 的路径 $|L_n|$ 数量满足 $\frac{1}{k}|L_n(G')| \leq |L_n(\mathcal{G})| \leq |L_n(G')|$ ([10, 定理 4.1.13])，其中 k 是 G'中的状态数。不可约图具有 $C\lambda^n \leq |L_n| \leq C'\lambda^n$。长度为 n 的闭路径 P_n 数量具有 $|P_n| \leq |L_n| \leq C'\lambda^n$。由于 G 是混合的，因此存在一个 N，使得每一对节点 v、v' 之间都有一条从 v到 v'、长度为 N的路径。因此，通过添加一条长度为 N 、从 w 到该词的路径，以及一条长度为N、从该词到 w 的路径，可以将 L_n 嵌入$P_{n+2N,w}$ （围绕节点 w 的闭路径）中。因此，对所有 $n \geq 2N$都有$C\lambda^{-2N}\lambda^n \leq |L_{n-2N}| \leq |P_{n,w}| \leq |P_n|$。□

实际上，这里讨论的几乎所有内容都将是混合的，不过为完整起见，我们可以给出以下推论：

推论 3.3。设 G 是有限个混合有向图之和。G 中长度为 n 的闭路径数量为 $\Theta(\lambda^n)$，其中 λ 是 G 的邻接矩阵的最大特征值。

证明。有向图之和的邻接矩阵是一个分块矩阵，其分块分别由该和的各个分量构成。因此，λ 是这些分块的最大特征值。于是，如果 G 有 k 个分量，则闭路径的数量为 $O(k\lambda^n) = O(\lambda^n)$。

由于还有一个分量的最大特征值为 λ，因此该分块中的路径以 $\Theta(\lambda^n)$ 的速率增长。因此，G 中的闭路径数量为 $\Omega(\lambda^n)$。□

因此，我们的策略是使用推论 3.3 来计算该有向图中的闭路径数量。如果我们的标记映射是一一对应的，那么这将给出机器生成的（循环约化）词集合的 Θ-类。然而，对于我们想要完成的工作而言，单射性要求过强。

定义 3.4. 如果存在常数 c，使得对任意循环约化词 w，像为 w的闭路径至多有 c 条，则称自动机的节点标记是一**对常数** 的。

为了更容易判断自动机的标记是否一对常数，我们给出以下有用的引理：

引理 3.5. 混合自动机的节点标记是一对常数，当且仅当不存在起点、终点和像都相同的不同路径。

证明。假设有两条不同的路径，p_0 和 p_1，从 x 到 y，且满足 $\eta_A(p_0) =\eta_A(p_1)$。现在任取一条从 y 到 x的路径 q。由于自动机是混合的，必定存在 q。令 $p_0 \circ q$ 表示先沿着 p_0、再沿着 q所形成的路径。我们有 $\eta_A(p_0 \circ q) = \eta_A(p_1 \circ q)$。设 $w = \eta_A(p_0 \circ q)$。由于从 x 到其自身至少有两种方式，且它们的像为 w，因此从 x 到其自身至少有 2^n 种方式，且像为 w^n。由于长度 w^n 在n 上呈线性增长，而 2^n 在 n上呈指数增长，因此产生的重复词数量不存在多项式界，更不用说常数界了。

现在假设不存在这样的路径对。对于任意一对映射到同一个词的闭路径，我们可以证明它们必定完全不相交。如果它们不相交，就会共享某个节点 x。这样，我们便可以从 x出发，沿着这两条闭路径行进，从而构造出两条不同的、从 x 到其自身且映射到同一个词的路径。由于任何一组重复词都必须由一组两两不相交的闭路径诱导，因此重复词的总数不超过节点数，也就是一个常数。□

如下：

引理 3.6. 设 A 是一个标记为一对常数的混合自动机。令$p(n)$ 表示 A中长度为 n 的闭路径数。令 $w(n)$表示由 A生成的长度为 n 的循环词数。 $w \in \Theta(p)$。

证明。 由于标记为一对常数且保持长度， $\frac{1}{c}p(n) \leq w(n) \leq p(n)$。因此 $w \in \Theta(p)$。

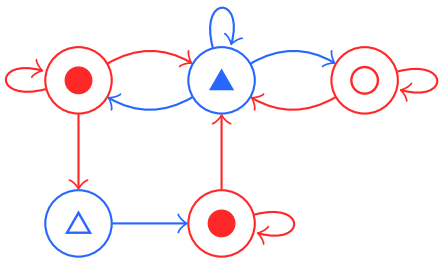
□

由此，我们已经具备证明下界所需的一切。我们将给出若干机器，并证明它们是约化的、混合的、单到常数的，而且只生成潜在正词。它们的特征值便给出了下界。

不过，我们认为这样做可能会有些神秘。我们构造自动机的方法或许会有用。对于一个固定的自动机 A和自同构 φ ，可以构造一个自动机，使其恰好生成由 A 生成的词在 φ作用下的像。接下来，我们的策略是从生成正词的自动机出发，应用自同构来增大最大特征值。我们构造的自动机都是将一个无限自同构族取极限所得。

4. Lower bounds

Figure 4.1. 一个只生成潜在正词的机器



4.1. **秩 2**。

定理 4.2. 图 *4.1* 中的机器生成的所有词都是潜在正的。此外，长度为 n 的路径通过该机器的增长为 $\Theta(\lambda_1^n)$，其中 $\lambda_1 \approx 2.50506841362147$ 是下列多项式的最大根：

$$(1-\lambda)\left(\lambda^4 - 3\lambda^3 + \lambda^2 + \lambda - 1\right).$$

证明. 固定一个由机器生成的词 w。考虑 $\mapsto$ 作用于 w的效果。 w 中的所有音节两侧都被夹住，因此 $\mapsto$ 会使它们的长度各减少 1。由于每个 前面都有 n，它引入的任何内容都会与一个 s 抵消，而 会引入一个新的来替代被抵消的那个。因此，所得的词仍由该机器生成，而且 s 的数量严格减少（除非原本就没有 s）。根据归纳法，反复应用这个自同构，最终会得到一个不含 s 的词。由于 F_2 中只在某个生成元上具有正指数的所有词都是潜在正的，这个词也是潜在正的。

该机器有两个标记为 的节点，不过我们可以确定不存在重复路径，因为其中一个节点的入边仅来自自身和 ，另一个节点的入边则仅来自自身和。

图 4.1 中机器的转移矩阵的特征多项式为

$$(1-\lambda)\left(\lambda^4-3\lambda^3+\lambda^2+\lambda-1\right).$$

其最大根为 $\lambda_1 \approx 2.50506841362147$，因此长度为 n 的路径增长为 $\Theta(\lambda_1^n)$。□

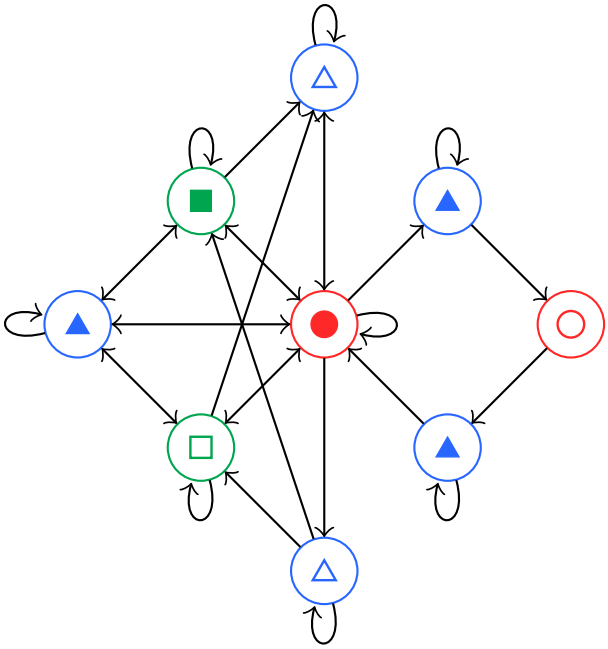


FIGURE 4.3. 该机器只生成潜在正词。

4.2. 秩 3 至 7。 为了得到更高秩的下界，我们将提出一种通用策略。为此，参考图 4.3 和图 4.4 中的机器可能会有所帮助。

具体来看图 4.4，将由此产生的词变为正词的策略是，从 向两个方向发出，然后发送 in

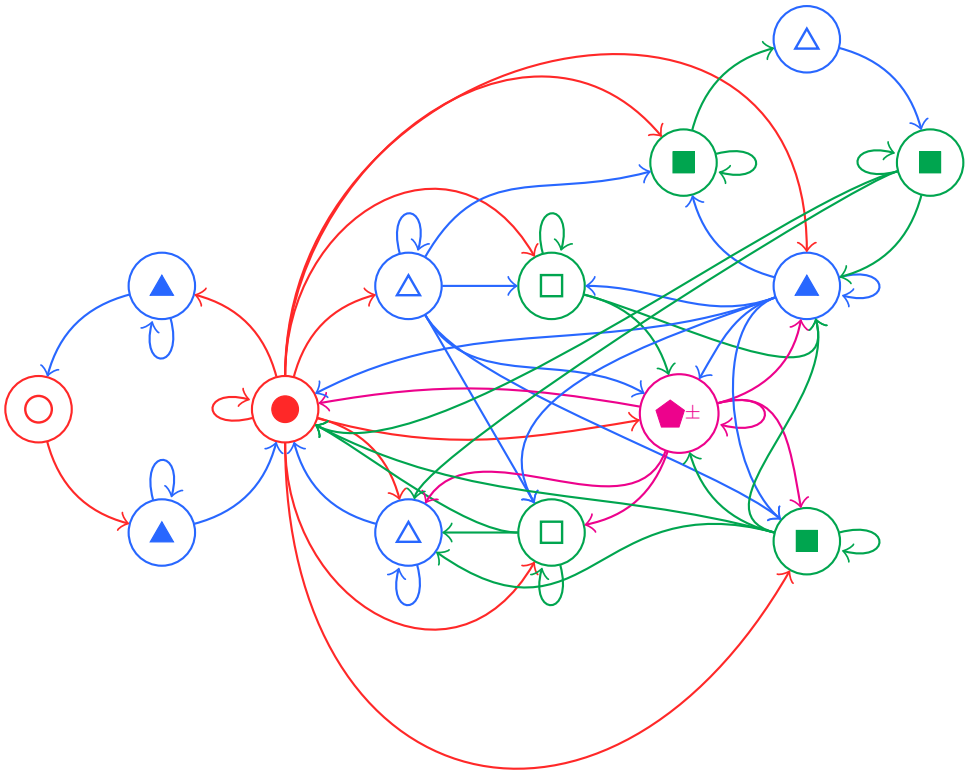


FIGURE 4.4.秩4情形中，(2)所述的机器只生成潜在正词。在该机器中，标记为 和 的节点具有完全相同的入边和出边，唯一的区别是它们各自都有一个自环，但彼此之间没有边。为简化图示，我们将这两个行为几乎相同的节点合并为一个节点。此节点标记为 ±，表示沿着从另一个节点指向该节点的入箭头前进时，机器可以生成 或 。然而，沿着从 ± 指向自身的入边前进时，机器无法切换符号。计算特征值时，可以使用原矩阵，也可以使用上方的矩阵，前提是指向标记为 ± 的节点的边（自环除外）权重设为 2（代表两种可能性）。

从 向两个方向发出。机器左右两侧的环形部分会将 s 和 s 隔离开，直到中间的 s 和 s 被消除。之所以可行，是因为当我们从 向两个方向发出时，所有能够指向右侧节点的元素最终都会“变成”一个 。最后，我们可以利用 s 消除剩余的 ，再利用 s 消除剩余的 s。然后我们可以使用命题 2.10 来完成证明。在更高秩的情形下，我们迭代这一策略。下面将更详细地给出一般描述。

下面以分块矩阵的形式给出我们关于 F_r 的自动机的一般邻接矩阵。其中， U 是一个矩阵，其对角线上方的所有元素均为 1，且对角线及其下方的所有元素均为 0。 L 是 $U^{\intercal}$，而 I 是单位矩阵。所有这些分块都是方形（$r-2$）$\times$（$r-2$）矩阵。

我们还将为每个节点命名并标注。节点名称采用与分块数组所在行对应的希腊字母，并带有一个下标，该下标与标签的下标相同（例如，不存在 ε_1 节点，因为分块矩阵的 ε行中没有标记为 x_1^{-1}的节点）。

需要注意的是，在所有情况下，两个节点集合之间的 U 表示存在一条从源节点到目标节点的转移，且源节点的索引严格大于目标节点。 L 则表示相反的含义，即源节点的索引严格小于目标节点。

$$
\begin{array}{cc}
 & \\
\alpha_1 & \{x_1 \\
\beta_i & \begin{cases} x_2 \\ \vdots \\ x_{r-1} \end{cases} \\
\gamma_r & \{x_r \\
\delta_r & \{x_r^{-1} \\
\varepsilon_i & \begin{cases} x_2^{-1} \\ \vdots \\ x_{r-1}^{-1} \end{cases} \\
\zeta_i & \begin{cases} x_2^{-1} \\ \vdots \\ x_{r-1}^{-1} \end{cases} \\
\eta_i & \begin{cases} x_2 \\ \vdots \\ x_{r-1} \end{cases} \\
\theta_i & \begin{cases} x_1^{-1} \\ \vdots \\ x_{r-2}^{-1} \end{cases} \\
\iota_i & \begin{cases} x_2 \\ \vdots \\ x_{r-1} \end{cases}
\end{array}
\begin{bmatrix}
1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\
1 & 1 & 1 & 1 & U+L & U & 0 & 0 & U \\
1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & L & 0 & 0 & L+I & 0 & 0 & 0 & 0 \\
1 & U+L & 1 & 1 & L & U+I & 0 & 0 & U \\
1 & L & 0 & 0 & L & 0 & I & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & I & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & I & I
\end{bmatrix}
\tag{2}
$$

定理 4.5。由 *(2)* 生成的所有词都是潜在正的。

证明。 设有一个由（2）中的机器生成的词 w 。我们将明确构造一列自同构，将 w 变为正词。首先，我们将应用自同构 $x_1 \mapsto x_2^{n_2} x_1 x_2^{n_2}$，其中 n_2 比最大 x_2^{-1}音节的指数大 1，也就是说，它是连续x_2^{-1}的最长连续串的长度。机器中有三个节点标记为 x_2^{-1}。注意，其中两个，即ε_2 和 ζ_2，必须始终出现在与 x_1相邻的音节中。具体而言， ε_2 必须始终紧跟在 x_1 之后，而 ζ_2 必须始终位于 x_1之前。因此，应用自同构 $x_1 \mapsto x_2^{n_2} x_1 x_2^{n_2}$会将由 ε_2 和 ζ_2 生成的所有音节替换为

正 x_2 音节。还要注意，x_1^{-1}只能出现在形式为 $x_1x_2^ix_1^{-1}x_2^jx_1$ 的子串中（其中 i和 j为正数）。因此，通过应用此自同构，从x_1^{-1}音节引入的 $x_2^{-n_2}$音节会立即抵消。

接下来我们应用 $x_2 \mapsto x_3^{n_3}x_2x_3^{n_3}$，其中 n_3 比最大 x_3^{-1} 音节上的指数大 1。这会消除由 ε_3 和 ζ_3 产生的所有 x_3^{-1} 音节。注意，由 θ_i 产生的 x_2^{-1} 音节可能出现在初始词 w 的各种子串中，但应用 $x_1 \mapsto x_2^{n_2}x_1x_2^{n_2}$ 后，它们总是出现在形如 $x_2x_3^ix_2^{-1}x_3^jx_2$ 的子串中。因此，应用这第二个自同构不会引入新的 x_3^{-1} 音节。

我们继续这一过程：让每个生成元在两个方向上都生成直到 x_{r-2} 的后续生成元，直到消除所有源自 ε_i 和 ζ_i 的音节。

此时，每个负符号音节都属于以下情况之一：
(1) 出现在形如 $x_{r-1}x_i^{-1}x_{r-1}$ for $i < r-1$ 的子词中。（这些源自 θ_i）(2) 一个 x_r^{-1} 音节。（这些源自 δ_i）

我们先处理 (1)。注意，此时所有 x_{r-1} 音节都是正的。我们首先应用自同构 $x_{r-1} \mapsto x_{r-1}x_{r-2}$。这会消去所有剩余的负符号 x_{r-2} 音节，因为它们都与一个 x_{r-1} 相邻。现在，每个先前被 (1) 覆盖的音节都与一个 x_{r-2} 相邻，因此我们应用自同构 $x_{r-2} \mapsto x_{r-2}x_{r-3}$，并沿着这种递降模式继续，直到应用 $x_2 \mapsto x_2x_1$，消去 (1) 类型的最后一个负符号音节。

现在剩下的负符号音节都形如 x_r^{-k}，因此由命题 2.10 可知，证明完毕。

□

5. Upper bound

我们希望得到 F_2 中潜在正词增长率的一个上界，因此将从使用潜在正性的一个必要条件开始。

Figure 5.1.恰好生成那些满足 Goldstein 判据的词的机器，其中 $y =$ 和 $x =$

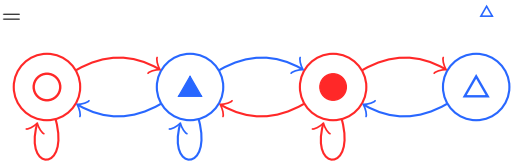


图5.1中的机器生成所有满足 Goldstein 判据的词，其中 x 和 y 取特定值。容易看出，所有满足 Goldstein 判据的词具有相同的增长率，因为有 8 种有意义的 x，y 组合，且每种组合都会将图5.1中的机器重新标记。

因此，我们有如下结论：

引理5.2. PP_2 的增长为 $\Omega(\lambda_1^n)$，其中 $\lambda_1 \approx 2.53209$ 是该多项式的最大根：

$$\lambda^3 - 3\lambda^2 + 3 \tag{3}$$

证明. 由图5.1可得以下转移矩阵:

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{4}$$

我们可以计算出（4）的特征多项式为

$$\lambda\left(\lambda^3 - 3\lambda^2 + 3\right)$$

由于每个节点都有唯一的标签，将路径分配给词的函数显然是单射。因此，图 5.1 中机器的增长率是（3）的最大根。根据定理 2.11，每个潜在正词都由该机器生成，因此潜在正词的增长率为 $\Omega(\lambda_1^n)$。□

我们将使用记号 γ 表示自同构$\leftrightarrow$ 。我们已经看到，潜在正词的数量受满足 Goldstein 判据的词的数量的上界限制。通过考察满足 Goldstein 判据、且在应用自同构后仍满足该判据的词，我们可以进一步降低这个上界。根据引理 2.12，如果一个潜在正词以某个 x 和 y满足该判据，那么在应用某个自同构后，它要么以相同的 x 和 y 满足该判据， γ要么以x（ γ）和y（ x ）满足该判据。于是，我们可以将应用引理 2.12 中的自同构后仍满足 Goldstein 判据的潜在正词集合分成两组：一组对应于 y和 γ，另一组对应于x（ γ）和y（ Θ）。应用该自同构后满足判据的词的 Θ类等于这两组中较大者的 F_2类，因此我们可以分别考察它们。此外，如果我们再次应用引理 2.12 所指定的自同构，那么这两组中的每一组都可以按相同方式进一步分成两个情形。重复这一过程，就得到一棵无限二叉树（图 5.4）。树的每一层由词在固定次数的过程中满足 Goldstein 判据的所有方式组成。每一层中增长最快的集合给出了潜在正词数量的上界。我们通过以下定义将这一思想形式化。

定义 5.3。 定义 G为满足 Goldstein 判据且具有 $x =$ 和 $y =$ 的词集。令 φ 为引理 2.12 中描述的自同构 $\mapsto$ ，并令 γ 为之前的自同构 $\leftrightarrow$ 。递归定义：

$$R(S) = \{w : w \in G, \varphi(w) \in S\}$$
$$L(S) = \{w : w \in G, \gamma(\varphi(w)) \in S\}$$

我们将采用一种方便的简写方式，省略括号和末尾的 G 。因此，例如，我们写作 L^2R 表示 $L^2(R(G))$。

我们对定理 1.2 的证明依赖于以下关于图 5.4 中这棵树的三个事实。

命题 5.5。 对于树中的每个词集 S， $\mathfrak{g}(SL)$ $\mathfrak{g}(SR) \in O\left(\mathfrak{g}(S)\right)$。换言之， S 的子项的增长率至多为 S的增长率。

我们将图 4.1 R^∞ 中机器生成的词集称为.....；结合下面的命题，这一记法便不难理解：

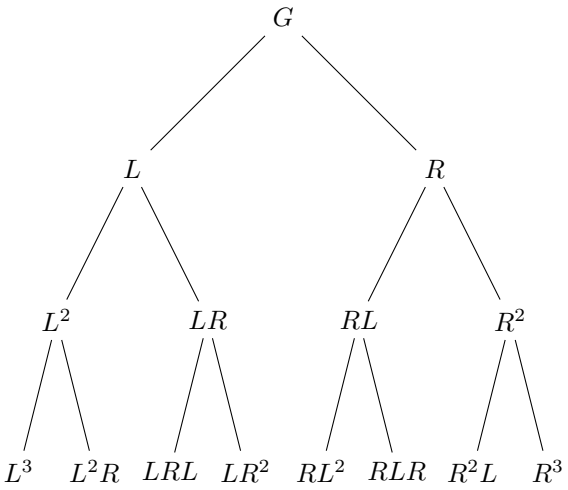


FIGURE 5.4. 以树形绘制的前四层集合

命题 5.6。

$$\bigcap_{n=0}^{\infty} R^n = R^\infty$$

命题 5.7。对于所有 $n \in \mathbb{N}$,

$$\mathfrak{g}(R^n L) \in o(\mathfrak{g}(R^\infty))$$

暂且将这些命题视为已知，我们可以证明主要结果：

定理 *1.2* 的证明。 我们将树的第 n 层生成的词集称为 Y_n。可以将其视为所有在经过 Goldstein算法 n 步后不会到达非正词的词。

我们可以看到， $\mathfrak{g}(Y_n)$ 给出了 PPr 增长的逐步上界，并且随着 n 趋于无穷而变得越来越紧。我们将证明 $\mathfrak{g}(Y_n) \sim \mathfrak{g}(R^n)$，因此，当 n 趋于无穷时， $\mathfrak{g}(R^n)$ 的逐点极限是一个上界。

由于 $R^n L$ 的增长严格慢于 R^∞ （命题 5.7），根据归纳法，树中名称包含任何 L 的所有集合，其增长都严格慢于 R^∞。由于 R^∞ 是 R^n 的子集，它的增长必然更慢（$\mathfrak{g}(R^n) \in \Omega(\mathfrak{g}(R^\infty))$），因此对于树中任何包含 L 的节点 S， $\mathfrak{g}(S) \in o(\mathfrak{g}(R^n))$。由于 Y_n 是 R^n 与树中其他包含 L 的集合的并集， $\mathfrak{g}(Y_n) \sim \mathfrak{g}(R^n)$。

由于 $\mathfrak{g}(R^n)$收敛至不超过 $\mathfrak{g}(R^\infty)$，这意味着我们的上界收敛至 $O(\mathfrak{g}_k(R^\infty)) = O(\lambda_1^k)$。根据定理 4.2，我们有 $\mathfrak{g}_k(\mathrm{PP}_2) \in \Omega(\lambda_1^k)$，因此$\mathfrak{g}_k(\mathrm{PP}_2) \in \Theta(\lambda_1^k)$。□

剩下的就是证明这些命题。命题 5.5 的证明最短。

命题 *5.5* 的证明。 我们只证明 $\mathfrak{g}(SL) \in O(\mathfrak{g}(S))$。另一种情形的证明方法相同。我们对树中的所有路径进行归纳。根据定义 5.3，我们有 $L \subseteq G$。设 Z_0 和 Z_1 是满足 $Z_0 \subseteq Z_1$的集合。因此， $R(Z_0) \subseteq R(Z_1)$，同样地， $L(Z_0) \subseteq L(Z_1)$。于是，由归纳法可得$SL \subseteq S$，因此 $\mathfrak{g}(SL) \in O(\mathfrak{g}(S))$。□

现在，我们将用禁子词（在 [10]中称为“禁块”）来描述语言 R^n 和 R^nL。这一描述将有助于证明另外两个命题。

引理 5.8。集合 R^n 是所有不包含以下子词的约化词：

-
-
- 所有 ^{k} 其中 $0 \le k \le n$

证明。首先，我们将证明这些子词均不会出现。由 Goldstein 判据可知，，以及 都不可能出现。集合 R^n 可表示为：

$$R^n = \{w \mid w \in G, (\mapsto {}^n)(w) \in G\} \tag{5}$$

假设 $w \in R^n$ 包含一个子词 ^{k}。它在 $\mapsto {}^n$ 下的像必定包含 $^{k-n}$，因为右侧会在其左侧产生 $^{-n}$，而在只引入新的 和字母的自同构下，两者都无法抵消。若 $k < n$，则该像包含 ，这在 G中是禁止的。若 $k = n$，则该像包含 ，这在 G中也同样是禁止的。因此 k 必须大于 n。

设 w 是一个不包含 、 或 的循环词。那么 在 w 中只能紧邻 出现，因此 $w \in G = R^0$。我们将对 n 使用归纳法。设对于任意 $0 \le k \le n$，w 均不包含 、 或 ^{k}。我们将证明，若 φ 是自同构 $\mapsto$，则对于任意 $0 \le k \le n-1$，$\varphi(w)$ 不包含 、 或 ^{k}。反设它确实包含这几类子词之一。若 $\varphi(w)$ 包含 ，则 φ 必定加入了一个未被约掉的 ，因此 w 必定含有形如 或 的子词，这与我们的假设矛盾。类似地，若 $\varphi(w)$ 包含 ，则 要么原本就在 w 中，要么是由 φ 加入的。因此，w 必定含有形如 或 的子词，这与我们的假设矛盾。最后，若 $\varphi(w)$ 包含形如 ^{k} 的子词，其中 $0 \le k \le n-1$，则 w 必定含有形如 $^{k+1}$ 的子词，再次与我们的假设矛盾。□

引理 5.9. R^nL 是 R^n 中所有不包含形如^k （其中 $k > n+2$）的子词的约化词集合。

证明。令 φ为自同构 $\mapsto$。首先，注意到集合 R^nL 可表示为

$$\begin{aligned} R^nL &= \{w \mid w \in G, \varphi^n(w) \in L\} \\ &= \{w \mid w \in G, \varphi^n(w) \in G, \gamma(\varphi^{n+1}(w)) \in G\}. \end{aligned}$$

这立即表明，如果 w 不属于 R^n，那么它也不属于 R^nL。

接下来，假设 w 包含一个形如 ^{k} 的子词，其中 $k > n+2$。令ψ 为自同构 $\gamma \circ \varphi^{n+1}$。那么 $\psi(\ {}^k)$ 包含 $^{k-n-1}$ 作为子词。由于 $k > n+2$，子词的像中的指数必定至少为二，这意味着 $\psi(w)$ 包含其作为子词，因此不属于 G。因此，如果 w 包含形如 ^{k} 的子词，则不属于 R^nL。

另一方面，假设 $w \in R^n$ 和 w 不包含形如 ^{k} 其中 $k > n+2$的子词。为了证明 w 属于 R^nL，我们只需证明 $\gamma(\varphi^{n+1}(w)) \in G$。这等价于证明 , , 和不会作为子词出现在 $\varphi^{n+1}(w)$中。在 $\varphi^{n+1}(w)$ 中出现的任何 都必定已经在 w中。事实上，任何新的 都必须由一个产生，然而每个

两侧都必须紧邻 音节，并且根据引理 5.8，这些音节上的指数必须大于 n。由于 $w \in G$，它不能包含 ^{m} 或 ^{m} 作为子词，因此 在 w 中的任何一次出现都必须位于形如 ^{k} 的子词中，其中 $k > 0$。因此，只要对于 $k \leq n+2$ 的任何此类子词都有 w，就不可能是 $\varphi^{n+1}(w)$ 的子词，因此 w 属于 $R^n L$。□

命题 *5.6* 的证明。由引理 5.8 可知，所有 R^n 都是不包含 、或 ^{k} 的词，其中任意 $k \geq 0$。

从图 4.1 可以清楚看出，这些子词都不可能出现在 R^∞ 中。

假设 w 不包含 , 、或 ^{k} ，对于任意 $k \geq 0$。我们将证明，图 4.1 中的机器会生成 w。根据我们的假设，中的每个 w 前后都有 ，并且 不可能出现在 之前或之后。最后，出现在一个 之后的 不能紧跟一个 。因此，我们看到确实 w 是机器中的一条路径。□

命题 *5.7* 的证明。考虑 $R^n L$ 中的循环词。我们将定义一个函数 f，它取 $R^n L$ 的一个泛型子集，并将其中每个词映射到 R^∞ 中长度相同的一个词。这将表明 $\mathfrak{g}_k(R^n L) \in O(\mathfrak{g}_k(R^\infty))$，因为这样的编码表明所选子集以 R^∞ 的速度增长，而其余部分可忽略，因此不会影响渐近增长。

机器 $R^n L$ 和 R^∞ 有两处不同：

- $R^n L$ 禁止所有形如 $^{n+2+k}$ 的子词，其中 k 是正整数；而 R^∞ 则不禁止。
- R^∞ 禁止所有形如 $^{n+k}$ 的子词，其中 k 是正整数；而 $R^n L$ 则不禁止。

所有其他子词（不包含上述任何子词）要么在两台机器中都不合法，要么在两台机器中都合法。

接下来，我们的函数 f 将替换以下形式的片段：

$$^{k_0}\,^{n+k_1}\,^{n+k_2}\ldots\,^{n+k_{s-1}}\,^{k_s}$$

其中 $s \geq 2$ 和 $k_0, \ldots, k_s$ 为正整数，满足以下条件：

$$^{k_0-1}\,^{n+k_1+2}\,^{n+k_2}\ldots\,^{n+k_{s-1}}\,^{k_s-1}$$

我们给出一些观察结果。

- 我们要替换的片段在 $R^n L$ 中是合法的，但在 R^∞中是不合法的。
- 第一种形式的片段不可能发生重叠，因此替换方案不存在歧义。
- 只要 $R^n L$中的一个词包含 ，它在 R^∞中所有不合法的子词都必须属于这种形式。
- 在 f 中进行替换后得到的像是一个在 R^∞中合法、但在 $R^n L$中不合法的字符串。
- 该线段及其像长度相同。

如果我们考虑 $R^n L$ 中不包含 的词的子集，可以看出它们也必定不包含 ，因此它们的增长最多为 $O\left(2^\ell\right)$。与总词集 $R^n L$ 相比，它们可忽略不计，而总词集的增长显然更快。因此，如果令 $R^n L'$ 为属于 $R^n L$ 且包含 的词集，那么 $R^n L'$ 的增长与 $R^n L$渐近等价。

函数 $f\colon R^n L' \to R^\infty$ 是一个保持输入长度的单射。因此我们有

$$\mathfrak{g}(R^nL) \sim \mathfrak{g}(R^nL') \in O(\mathfrak{g}(R^\infty))$$

此外，子词 $v = \bullet\blacktriangle\circ^n\blacktriangle\circ^{n+3}\blacktriangle$不能出现在任何$w \in R^nL'$ 在 f下的像中。这是因为 v 在 R^nL中不合法，但不能作为替换片段的子词出现，因为这类片段的第一个音节的指数总是 $> n+2$，而每个其他音节前面都必须有 （而这前面有 ）。

由于 R^∞ 是由不可约混合自动机生成的语言，因此它具有“增长敏感性”（见 [3]），所以排除任意子词后所得的子集是指数可忽略的。因此，对于某个 $\varepsilon < 1$， $\mathfrak{g}_k(R^nL') \in O(\varepsilon^k\mathfrak{g}_k(R^\infty))$，于是

$$\mathfrak{g}(R^nL) \sim \mathfrak{g}(R^nL') \in o(\mathfrak{g}(R^\infty))$$

结论得证。 □

这就给出了定理 1.2 中较弱的结论；现在我们将证明一个稍强的版本。命题 5.7 的一个加强形式如下：

引理 5.10。 R^∞ 在PP_2中不可忽略

证明。 更确切地说，我们将证明 R^∞ 在$\bigcup_{n=0}^\infty R^nL$中不是可忽略的。这是充分的，因为 $R^nL \supsetneq \mathrm{PP}_2$。与命题 5.7 的证明一样，我们将舍弃所有不含 a 的词，因为根据相同的推理，这些词在总量中可忽略。

我们的策略是提供一个单射 $f\colon (\bigcup_{n=0}^\infty R^nL' \setminus R^\infty) \rightarrowtail R^\infty$，使长度增加不超过某个常数上界。更正式地说，存在一个常数 c，使得对于定义域中的每个 w，$|w| + c \geq |f(w)|$。这就足够了，因为 R^∞ 呈指数增长，即 $\mathfrak{g}_n(R^\infty) \in \Theta(\lambda^k)$。因此

$$\mathfrak{g}_{k+c}(R^\infty) \sim \lambda^c\mathfrak{g}_k(R^\infty) \in \Theta(\lambda^k)$$

为了定义 f，考虑定义域中的某个词 w。 w 属于某个 R^nL，其中 n。如果 w 属于多个 R^nL，我们就取可能的最大值作为 n。选定 n后， R^∞ 中被禁止的子词形式如下

$$\blacktriangle\bullet^{k_0}\triangle\bullet^{n+k_1}\triangle\bullet^{n+k_2}\triangle\ldots\triangle\bullet^{n+k_{s-1}}\triangle\bullet^{k_s}\blacktriangle$$

其中 $s \geq 2$ 且每个 k_i 均为正。我们将任意选取其中一个子词作为“信号段”。其余这些禁子词都将替换为：

$$\blacktriangle\bullet^{k_0-1}\blacktriangle\circ^{n+k_1+2}\blacktriangle\circ^{k_2}\blacktriangle\ldots\blacktriangle\circ^{k_{s-1}}\blacktriangle\bullet^{k_s-1}\blacktriangle$$

这种替换不会增加词的长度。对于信号段，我们将其替换为以下形式：

$$\blacktriangle\bullet^{k_0-1}\blacktriangle\circ\blacktriangle\circ^{n+3}\blacktriangle\circ^{k_1}\blacktriangle\circ^{k_2}\blacktriangle\ldots\blacktriangle\circ^{k_{s-1}}\blacktriangle\bullet^{k_s-1}\blacktriangle$$

这种替换可能会增加长度，但最多增加 4（当 $s = 2$ 时会出现这种情况）。

为了证明这是一个单射，我们将给出一种从像中恢复词的方法w。首先，我们可以利用信号恢复编码中使用的 n。我们找出所有形如i $m+3$ 的子词，其中 i为非负数，且 m。这样的子词中必有一个来自编码后的信号，而其他编码后的禁用词都不会产生这样的子词，因为它们都要求第一个音节的指数至少为 3。可能存在多个这样的子词，但除了

信号一定存在于初始词中，因此我们的信号就是指数最大的那个。一旦恢复了 n，反转信号段上的编码就很简单了；我们也可以确定所有非信号片段的位置，并反转它们的编码。每个编码子词都必须包含一个形如$^{n+k+2}$ 的词，其中 k为正数，而这样的片段在$R^n L$中是禁用的，因此不会出现在其他位置。由此，我们可以反转编码，并且 f 是单射。

由于唯一可能增加长度的替换是信号编码，而它恰好只发生一次，因此长度最多增加 6。□

这个问题自然延伸出的一个问题是， R^∞ 是否是 PP_2中的泛型。在我们给出的编码中，信号可以对应不同的片段。这表明可能存在一定的灵活空间，暗示 R^∞ 可能是泛型。然而，存在选择余地的情况需要多个禁用片段，而这种情况实际上相当罕见。事实上，与最初的直觉相反，计算机模拟表明， R^∞ 不是泛型，其密度大致固定在 $1/5$。（均匀抽样潜在正词并不容易，尤其是对于较长的词。为此，我们抽取了约 100 个长度约为 300 的词。）

目前，我们不提供任何方向的证明，但会提供证据，说明 R^∞是研究这个问题时应考虑的正确集合。

定理 5.11. 如果存在一个有限强连通自动机，能够生成 PP_2 的一个泛型子集， R^∞ 则能生成PP_2 的一个泛型子集。

证明. 设 A 是一个能够生成 PP_2 的泛型子集的强连通自动机。反设 R^∞ 在 PP_2，中不是泛型的，因此根据引理 5.10， R^∞ 的密度收敛到 0 与 1 之间的某个常数。

A 不能完全包含 R^∞，因为根据 [3, 主要结果] A 的增长速度会快于 R^∞。根据引理 5.10，R^∞ 的增长速度与 PP_2 相同，因此这意味着 A的增长速度会快于它声称包含自身的集合。

现在考虑交集 $P = A \cap R^\infty$。由于 R^∞ 不可忽略，而 A是泛型的，因此 P 是 R^∞的一个泛型真子集。然而，由强连通自动机生成的两个集合的交集也由一个强连通自动机生成，因此根据 [3, 主要结果]，该交集不可能是R^∞中的泛型集合。□

6. An automaton for the potentially positive words

在定理 5.11 的证明中，我们给出了一个论证，表明不存在能恰好生成潜在正词的有限强连通自动机。

鉴于此，我们可以尝试使用一种更一般类型的自动机来生成 PP_2。最直接的做法或许是放宽要求，不再要求词由闭环生成，而是指定起始状态和结束状态，并考虑从起始状态到结束状态的所有路径的像。（通过为每个节点创建一份自动机副本，并将每份副本中的一个节点设为起始状态和结束状态，就可以将闭环要求转化为这种自动机。）这种自动机称为“有限状态机”（FSM），由有限状态机生成的语言称为“正则语言”。

问题 6.1。 对于任意 $r \geq 2$， PP_r 是正则语言吗？

对于那些主要具有动力系统背景的人而言，这等价于询问潜在正词的移位是否是$sofic$。

如果这些问题的答案是“是”，就意味着我们可以实现一种算法，用最坏情况时间复杂度为 $O(n)$、空间复杂度为常数的方式，识别长度为 n 的循环约化潜在正词。这确实是一个诱人的前景。

读者或许已经猜到，既然这里提出的是一个问题而非命题，这个梦想就不会实现。我们将证明这种自动机并不存在。事实上，我们将证明一个更一般的结果。

有限状态机等价于没有存储带的图灵机。因此，它们被称为有限 状态机，是一种内存有限（即受常数限制）的计算模型。我们已经从 [9] 得知，确定性图灵机能够识别 PP_2。因此，不受限制的内存足以完成识别。事实上，所给出的算法只使用线性空间（DSPACE(n)），所以图灵机可以在内存空间受线性上界限制的情况下识别 PP_2。这种自动机称为“线性有界自动机”，而能够由这类自动机识别的语言称为“上下文有关语言”。在线性有界自动机和有限状态机之间还存在其他类别的自动机。例如，上下文无关语言就是能够由下推自动机识别的语言。

定理 6.2。 对于 $r \geq 2$，PP_r 不是良嵌套多重上下文无关语言。

良嵌套多重上下文无关语言（良嵌套 MCFL）是包含所有正则语言的一类语言。良嵌套 MCFL 并不是广为人知的语言类别，因此本文的读者很可能对其并不熟悉。我们不会试图改变这一点。 [7], 中给出了定义，但我们的证明不会用到定义的技术细节。由于正则语言包含于良嵌套 MCFL 中，定理 6.2 将表明，问题 6.1 的答案是“否”。事实上，由于我们避开了技术细节，转而使用已有结果，读者可以将“良嵌套 MCFL”替换为“正则语言”，仍能完全理解证明。实际上，其他广为人知的语言类，例如上下文无关语言、ETOL 语言和树邻接语言，也包含于良嵌套 MCFL 中，同样可以这样替换。

我们将证明关于良嵌套 MCFL 的两个重要引理:

引理 6.3 (良嵌套 MCFL 的抽引引理 [7])。 若 $L \subseteq \Sigma^*$ 是良嵌套 *MCFL*，则存在某个 $k \geq 1$ （称为 L的抽引维数）和某个 $p \geq 1$ （称为 L的抽引长度），使得每个满足$|s|_\Sigma \geq p$ 的字符串 $s \in L$ 都可以写成

$$s = v_0\, w_1\, v_1\, w_2\, v_2\, \ldots\, w_k\, v_k$$

其中:

- $\{w_i \mid 1 \leq i \leq k\} \neq \{\varepsilon\}$
- $\forall n \in \mathbb{Z}_{\geq 0} : v_0\, w_1^n\, v_1\, w_2^n\, v_2\, \ldots\, w_k^n\, v_k \in L$

自然，这个引理适用于所有更弱的语言类。不过，正则语言有一个简单得多的泵引理。参见 [6, 定理 4.1].[3] 我们将采用相同的技巧，利用这两种方法将 PP_2 排除在相应的语言类之外。

[3]关于上下文无关语言，参见 [6, 定理 7.18]。关于树邻接语言，参见 [19, 定理 4.7]。

引理 6.4 ([16, 命题 6][4])。对于每个良嵌套多重上下文无关语言 $L_1 \subseteq \Sigma^*$，都存在一个 k，使得对于每个正则语言 $L_2 \subseteq \Sigma^*$，$L_1 \cap L_2$ 都是一个抽引维数为 k 的良嵌套多重上下文无关语言。

关于正则语言中的对应结果，参见 [15, 定理 2.7].[5]

由此，定理 6.2 将成为以下结论的一个相当简单的推论：

引理 6.5。PP_2 不是良嵌套多重上下文无关语言。

证明。反设 PP_2 是泵浦维数为 k 的良嵌套多重上下文无关语言。考虑由以下形式的词组成的语言 Λ_k：

$$\triangle\bullet^{n_1}\triangle\bullet^{n_2}\triangle\bullet^{n_2}\triangle\ldots\triangle\bullet^{n_k}\triangle\bullet\blacktriangle\circ^{m}\blacktriangle\bullet$$

其中 n_1、.....、n_k、$m \geq 1$。由于 Λ_k 是正则语言，根据引理 6.4，只有当 PP_2 是良嵌套多重上下文无关语言时，PP_2 才是良嵌套多重上下文无关语言。我们首先证明，若词 $w \in \Lambda_k$ 是潜在正的，则 $\forall i$： $n_i \geq m-1$。

考虑一个词 s，它属于语言 Λ_k，但含有一些 $n_i < m-1$。注意，s 满足 Goldstein 判据，其中 $x =$ 且 $y =$。我们将证明，$\triangle$应用自同构$\mapsto \blacktriangle\bullet^{n}$ 后，该词不再满足 Goldstein 判据，其中 n 是 n_i 中最小的 s。应用该自同构后，音节 n 变为空，从而产生一个 子词，因此 $x \neq$ 根据引理 2.12，唯一的另一种可能是 $x =$。然而，必定会作为子词出现，因为m在原词中出现过，而我们只从指数中移除了 $n \leq m-2$。由于 s 与一个不满足 Goldstein 判据的词自同构，因此它不是潜在正的。

现在令 $L_k \cap \mathrm{PP}_2$ 的抽引长度为 p。我们考虑这个词

$$s = \triangle\bullet^{p}\triangle\bullet^{p}\triangle\ldots\triangle\bullet^{p}\triangle\bullet\blacktriangle\circ^{p+1}\blacktriangle\bullet \in L_k.$$

这个词是潜在正的，我们可以证明：

$$\begin{aligned} s &= \triangle\bullet^{p}\triangle\bullet^{p}\triangle\ldots\triangle\bullet^{p}\triangle\bullet\blacktriangle\circ^{p+1}\blacktriangle\bullet \\ (\blacktriangle \mapsto \blacktriangle\bullet^{p}) \quad &\mapsto \triangle^{k+1}\bullet\blacktriangle\circ\blacktriangle\bullet \\ (\bullet \mapsto \blacktriangle^{k+1}\bullet) \quad &\mapsto \bullet\blacktriangle\circ\blacktriangle\bullet \\ (\blacktriangle \mapsto \blacktriangle\bullet) \quad &\mapsto \bullet\blacktriangle\blacktriangle\bullet\bullet \end{aligned}$$

因此 $s \in L_k \cap \mathrm{PP}_2$。

$|s| = (k+1)(p+1)+5 \geq p$ 因此，引理 6.3 应适用于这个词。Now_i 可以包含以下任何带下划线的部分，因为删去这些部分（即 $v_0\, w_1^0\, v_1\, w_2^0\, v_2\, \ldots\, w_k^0\, v_k$）会使这个词不再是 L_k：

$$\underline{\triangle}\bullet^{p}\underline{\triangle}\bullet^{p}\underline{\triangle}\ldots\underline{\triangle}\bullet^{p}\underline{\triangle\bullet\blacktriangle}\circ^{p+1}\blacktriangle\bullet$$

还剩下 $k+1$ 个独立片段，因此不可能每个片段都包含一个 w_i。每个片段要么是 a，要么是音节。根据上述论证，w_i 不可能包含剩余音节的任何部分，因为如果增大 上的幂，而不同时增大其他所有音节上的幂，该词就不再是潜在正的。

---

[4] [16] 中的表述与此处的表述有显著差异。可以参阅 [7, 第 323] 页，其中以更协调的方式阐述了该结果。[5]关于上下文无关语言，见 [6, 定理 7。27], 关于树邻接语言，见 [19, 定理 4。5]。

此外，如果某个非空 w_i 包含了剩余音节的一部分，移除 w_i 子词就会降低该音节上的指数，使其小于 $m-1=p$。 m 无法降低，因为该音节不包含任何w_j。因此，该词就不再是潜在正的。

必定存在一个非空的 w_i，但任何部分中都不包含 s，因此我们得出矛盾，$_2$ 不可能是良嵌套多重上下文无关语言。□

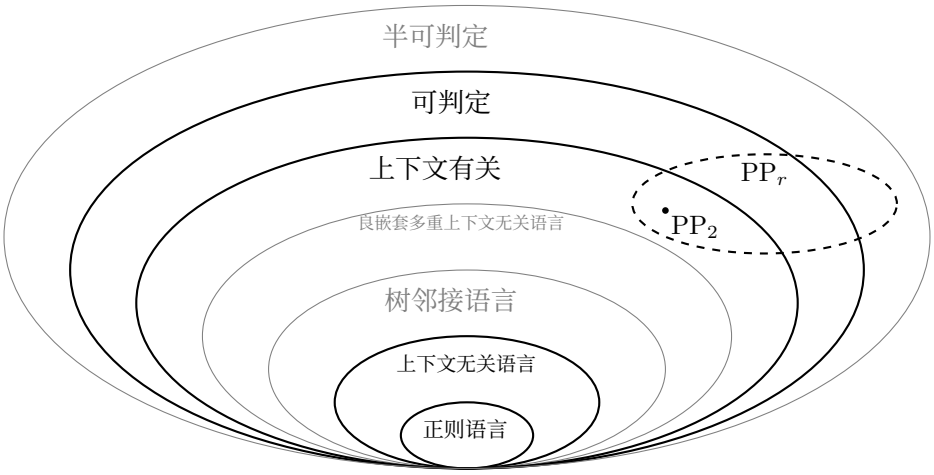
现在，定理 6.2 将很容易得到：

定理 *6.2* 的证明。根据 [4], $\mathrm{PP}_r \cap F_n = \mathrm{PP}_n$ 在 $r \geq n$ 时的主要结果。由于 F_r 是正则语言，我们可以应用引理 6.4，看到对于 $r \geq 2$，$_r$ 是良嵌套多重上下文无关语言，当且仅当 $_2$ 是良嵌套多重上下文无关语言。由于 $_2$ 不是良嵌套多重上下文无关语言，更高的秩也都不是。□

推论 6.6。不存在能够使用常数空间判定循环约化的潜在正词的算法。

这可由复杂性理论中的一个经典结果推出（见 [17]）：该结果指出，如果一个集合可以在常数空间内判定，那么它就是正则语言。

FIGURE 6.7. 这幅欧拉图展示了潜在正性的分布，以及它与各种形式语言类别之间的关系。经典乔姆斯基层级中的类别以更粗、更深的线条绘制。虚线表示根据我们目前的理解，PP_r 对于 $r \geq 2$ 所能覆盖的最大范围。



7. TOWARDS NON-TRIVIAL UPPER BOUNDS

为 F_3 寻找上界已被证明更加困难。虽然在 F_2中，我们能够使用 [5] 和 [9] 中的算法来排除某些词并得出一个上界，但目前尚无适用于更高秩的算法。事实上，在 F_3 中已知并非潜在正的词相当有限。

（1）换位子群中的词 $[F_3, F_3]$。

（2）在 F_2 的自同构作用下不是潜在正的词 F_2 （见 [4]），以及与它们自同构等价的词。

[8] 中的定理3.9可用于证明（2）是指数可忽略的。对该定理稍作收窄，使其适用于本文：

定理7.1 （定理3.9）。 如果 $w \in F_r$ 是词 $v \in F_{r-1}$ 轨道中的一个循环约化词，那么 w 不包含本原性阻碍子词。

根据 [3, 例 1]，如果自由群的某个子集存在一个禁用子词，那么该集合是可忽略的。

那么，（1）似乎最有希望。回顾一下 Cohen-Grigorchuk 定理：

定理 7.2 (Cohen-Grigorchuk)。 设 $\varphi\colon F_r \to G$ 是一个满射群同态。 G 是可和的，当且仅当

$$\lim_{n\to\infty} \sqrt[n]{\mathfrak{g}_n(\ker(\varphi))} = 2r-1$$

证明见 [18] 。这里的推论是，由于 F_r 的阿贝尔化是可和的，换位子群的增长率粗略地说（相差一个次指数因子）与 $(2r-1)^n$ 相同。这为我们提供了一个非平凡上界的起点：

推论7.3。 F_r 中潜在正词的集合在所有词的集合中不是指数泛型的。

证明。 所有词的数量增长为 $\Theta((2r-1)^n)$。如果潜在正词的集合是指数泛型的，那么它的补集（其中包含换位子群）的增长率必为某个 $\varepsilon < 1$ 对应的 $O(\varepsilon^n(2r-1)^n)$。

Thus

$$\lim_{n\to\infty} \sqrt[n]{\mathfrak{g}_n([F_r, F_r])} \le \varepsilon(2r-1) < 2r-1$$

这与定理 7.2 矛盾。 □

$[F_r, F_r]$ 的增长可得到更精确的渐近估计，即 [14]

$$\frac{1}{\sqrt{n}^r}(2r-1)^n \tag{6}$$

这说明换位子群实际上是可忽略的（只是并非指数级地可忽略）。

我们在下表中总结当前情况：

	pos.	原始。	潜在正。[6]	all
F_2	$\Theta(2^n)$	$\Theta\left(\sqrt{3}^n\right)$	$\Theta(2.505^n)$	$\Theta(3^n)$
F_3	$\Theta(3^n)$	$\Theta(n3^n)$	$\Omega(4.024^n)$	$\Theta(5^n)$
F_4	$\Theta(4^n)$	$\Theta(n5^n)$	$\Omega(5.746^n)$	$\Theta(7^n)$
F_5	$\Theta(5^n)$	$\Theta(n7^n)$	$\Omega(7.509^n)$	$\Theta(9^n)$
F_6	$\Theta(6^n)$	$\Theta(n9^n)$	$\Omega(9.290^n)$	$\Theta(11^n)$
F_7	$\Theta(7^n)$	$\Theta(n11^n)$	$\Omega(11.083^n)$	$\Theta(13^n)$
F_r	$\Theta(r^n)$	$\Theta(n(2r-3)^n)$	$\Omega((2r-3)^n)$	$\Theta((2r-1)^n)$

References

[1] Gilbert Baumslag, *Positive one-relator groups*, 《美国数学会汇刊》156（1971年5月）（英文）。[2] Gilbert Baumslag、Alexei G. Myasnikov 和 Vladimir Shpilrain, *Open problems in combinatorial group theory* 1999, 1 27 , 《当代数学》, 第 – 页（英文）。[3] Tullio Ceccherini-Silberstein 和 Wolfgang Woess, *Growth and ergodicity of context-free languages* **354** 2002 11, 4597 4625 1926891 , 《美国数学会汇刊》（），第 – 期。MR[4] Aaron Clark 和 Richard Goldstein, *Stability of numerical invariants in free groups*, 《代数通讯》33（2005年10月），第11期，4097–4104（英文）。[5] Richard Goldstein, *An algorithm for potentially positive words in F_2*, 《当代数学》421（2006），157–168（英文）。[6] John E. Hopcroft、Rajeev Motwani 和 Jeffrey D. Ullman, *Introduction to automata theory languages and computation* 3 2007 , 第版，培生, 波士顿、旧金山、慕尼黑, （英文）。[7] Makoto Kanazawa, *The pumping lemma for well-nested multiple context-free languages*, 《语言理论发展》，2009年，第312–325页（英文）。丛书名：计算机科学讲义。[8] Lucy Koch-Hyde、Siobhan O'Connor、Eamonn Olive 和 Vladimir Shpilrain, *Orbit-blocking words in free groups* 2025 , 《群论杂志》（）。[9] Lucy Koch-Hyde、Siobhan O'Connor 和 Eamonn Olive, *Complexity of recognizing potentially positive elements in the free group of rank 2* 2025 , 《代数与计算国际期刊》（）（英文）。[10] Douglas A. Lind 和 Brian Marcus, *An introduction to symbolic dynamics and coding*, 第二版，剑桥大学出版社，剑桥，2021年（英文）。[11] Marco Linton, *Residually rationally solvable one-relator groups*, 《伦敦数学会通报》（2025年10月），blms.70210（英文）。[12] D. Puder 和 C. Wu, *Growth of primitive elements in free groups*, 《伦敦数学会杂志》 **90** 201405 1, 89 104, https://academic.oup.com/jlms/article-pdf/90/1/89/9682782/jdu009.pdf （），第 – 期，见。[13] Igor Rivin, *A remark on 'Counting primitive elements in free groups' (by J. Burillo and E. Ventura)* **107** 2004 1, 99 100 , 《几何献刊》（8月），第 – 期。[14] *Growth in free groups (and other stories)—twelve years later*, 《伊利诺伊数学杂志》54（2010），第1期，327–370。[15] Arto Salomaa, *Jewels of formal language theory*, 计算机科学出版社，罗克维尔，马里兰州，1981年（英文）。[16] H. Seki 和 Y. Kato, *On the generative power of multiple context-free grammars and macro grammars* **91** 2008 , 《电子信息通信学会信息与系统汇刊》E-D（2月），第2期，209–221（英文）。[17] R. E. Stearns、J. Hartmanis 和 P. M. Lewis, *Hierarchies of memory-limited computations*, 第6届开关电路理论与逻辑设计年度研讨会（SWCT 1965），1965年，第179–190页（英文）。[18] Ryszard Szwarc, *A short proof of the Grigorchuk-Cohen cogrowth theorem*, 《美国数学会杂志》106（1989年7月），663–665（英文）。[19] K. Vijayshanker、David J. Weir 和 Aravind K. Joshi, *Characterizing structural descriptionsproduced by various grammatical formalisms*, 《计算语言学协会第25届年会论文集》-，1987年，第104–111页（英文）。[20] Daniel T. Wise, *The residual finiteness of positive one-relator groups*, 《瑞士数学评论》76（2001年6月），第2期，314–338（英文）。

Department of Mathematics, CUNY Graduate Center, New York, NY 10016
Email address: emmad4867@gmail.com

Department of Mathematics, CUNY Graduate Center, New York, NY 10016
Email address: lhyde@gradcenter.cuny.edu

Department of Mathematics, CUNY Graduate Center, New York, NY 10016
Email address: doconnor@gradcenter.cuny.edu

Email address: ejolive97@gmail.com