arXiv:2601.21862v1 [cs.FL] 29 Jan 2026

元胞自动机可归约性作为无限字的复杂度度量

Markel Zubia[1] 与Herman Geuvers[2]

[1] 波鸿鲁尔大学，德国，markel.zubia@ruhr-uni-bochum.de[2] 奈梅亨拉德堡德大学，荷兰，herman@cs.ru.nl

摘要。 无限字，亦称流，在计算机科学和数学中具有重要意义，由此自然引出应如何度量其复杂度的问题。我们引入元胞自动机可归约性 作为流复杂度的度量：当存在一个将 σ 映射到 τ的元胞自动机时， σ 至少与τ 同样复杂。这使得可以将流划分为复杂度的度，类似于可计算性理论中的图灵度。我们研究由度的偏序所产生的层级的代数性质，表明该层级不是良基的，也不是稠密的；最终周期流按其周期的整除关系排序；稀疏流是原子；极大流具有最大子字复杂度；流集合的上确界一般并不存在。我们还给出一个伪算法，用于在该可约性意义下对流进行分类。

关键词： 元胞自动机、流、转换器、无限序列

1 引言

复杂度研究是理论计算机科学和数学诸多分支中的核心主题。举例来说，计算复杂性理论量化了判定问题和算法的难度 [2]。类似地，可计算性理论根据问题的（不可）判定程度对其进行分类 [35]。在这一广阔领域中，流被定义为有限字母表上的无限单边序列，发挥着重要作用。它们自然地产生，并在许多领域得到广泛研究，包括形式语言理论 [1], 自动机理论 [32], 数论 [5,17], 和动力系统 [22,27]。鉴于它们无处不在，探讨流本身的复杂度便显得很有意义。

衡量流复杂度的一种方法是研究通过计算模型实现的变换。具体来说，考虑一个自动机 M ，它以流 σ 为输入，并返回流 τ 作为输出。由此可以说， σ 包含足够的信息来重构 τ；换言之， σ 的复杂度至少与 τ一样高。图灵机虽然功能强大，但在此情境下存在困难，因为所有可计算流都会归入同一个等价类。因此，人们探索了更弱的模型，例如 Mealy机 [33,3]和有限状态转换器（FST） [13,15,14,24]。这些方法促使人们进一步研究通过其他模型实现的流变换。

本文旨在建立一个框架，以基于元胞自动机这一计算模型比较流的复杂度。元胞自动机是一类离散动力系统，其中按空间排列的细胞会随时间更新。每个细胞都具有内容，通常称为该细胞的状态；每个时间步，该状态都会根据局部规则进行更新，而该规则仅取决于该细胞及其相邻元胞的状态 [41]。本文通过将流视为由细胞组成的无限单边数组，引入一种基于一维元胞自动机（1CA）的流复杂度度量。这为比较流提供了一种数学上自然的定义，并且与元胞自动机的经典定义相一致 [41]。

例 *1.1. </style}*设 和 是如下所示的二进制流。则 *1CA* 可以按照以下规则将 变换为 ： 若一个细胞的状态为 *0*，且其左侧有一个状态为 *1* 的细胞，或者其左右邻居的状态均为 *0*，则该细胞变为 *1*；否则，该细胞变为 *0*。因此，我们称 的复杂度至少与 相同，或者简而言之，称 可归约为 τ。

$$\sigma = 10100100010000100000\ldots$$

$$\tau = 01010011001110011110\ldots$$

使用一维元胞自动机进行流变换，相较于 Mealy机和有限状态转换器有两大优势：

1. 一维元胞自动机形成了更细致的层级，能够更微妙地区分不同的流。例如，有限状态转换器将所有周期流归为同一个度，而一维元胞自动机能够区分它们。同样，著名的倍周期（PD）和 Thue-Morse（TM）序列（在例 3.1 和 3.2 中有形式化定义）对于有限状态转换器而言是等价的，而我们推测，对于 1CA 可约性，PD 的复杂度严格高于 TM。2. 1CA 归约简单，因此能够高效地进行流分类（见第 6 节），这也促使其在实际应用中得到使用。相比之下，有限状态转导器综合在计算上难以处理 [18,16]。

有限状态转换器与一维元胞自动机变换流的方式有着根本区别。有限状态转换器会从左到右依次扫描，并使用固定大小（尽管可以任意大）的记忆。相比之下，一维元胞自动机对所有元胞全局应用统一的局部规则，不涉及顺序处理。因此，尽管本文探讨了此前在其他类型自动机所诱导的层级中研究过的结构 [14,4,33], ，我们对这些结构的处理方式本质上有所不同。例如，关于有限状态转换器的研究常常依赖类似抽引引理的论证，而这类推理对于一维元胞自动机并无意义。相反，我们引入了新颖的技术，包括对角化论证（仿照康托尔）以及具有精心构造的局部邻域的流。

贡献 我们在第 4 节根据元胞自动机可归约性 的概念定义了流上的元胞自动机变换，并在第 4.1 节将其与有限状态转导器可约性进行比较。在第 5 节中，我们引入度 的概念，将其定义为由 1CA 可约性所诱导的等价类，即能够相互归约的一类流。我们在后续的第 5.1 至 5.3 节证明了流的度层级的关键结构性质，包括：

– 由于存在所谓的“原子”度，因此它并不稠密；– 由于存在无限下降链，因此它不是良基的；– 存在无限上升链；– 最终周期流按整除关系排序；– 稀疏流要么是原子，要么最终常值；– 极大流必须具有最大子词复杂度；– 流集合的上确界通常不存在。

第 6 节介绍一种用于检验一个流是否可约到另一个流的算法，并证明其正确性。第 7 节中，我们将考察其他定义，并将其与主要方法进行比较。我们先在第 2 节回顾相关工作。

2 相关工作

柯尔莫哥洛夫复杂度 柯尔莫哥洛夫复杂度以生成一个字的最小图灵机的大小来衡量该字的信息含量 [26]。尽管对无限字的研究已十分广泛 [38], ，但即使对于有限词，该复杂度也是不可判定的，而且在实践中难以近似计算 [43]。一种替代方案是根据流之间的图灵可转导性对其分类，但这种方法会将所有可计算流归为同一类，因此促使人们研究通过较弱计算模型定义的可约性。

*Mealy*机与有限状态转换器 Mealy机是最早用于研究流可约性的较弱模型之一，最初见于 [33] ，之后见于 [3]。这些自动机针对每个输入字母输出一个字母，从而生成对应的输出流。后续研究转向有限状态转换器（FSTs），这类转换器更加灵活，可针对每个输入字母输出一个有限字 [13,15,14,24]。这些研究考察了由通过 Mealy 机和 FSTs 进行归约所诱导的流层级的代数性质，并特别关注极值、链、反链、稠密性和原子度（极小非平凡度）。

子词复杂度子词复杂度根据流中有限子串的多样性来衡量其复杂度 [11,4]。由于这种度量不施加计算限制，因此对于计算复杂性很高的流，包括不可判定流，也可能赋予相对较低的复杂度值，这限制了它在计算层面的实用性。

ω-语言、*Wadge* 预序与 *Wagner* 层级 ω-语言是由自动机识别的一组流。最突出的一类是 ω-正则语言，由 Büchi 自动机识别 [39]。更广泛的类别包括上下文无关语言 [39,19] 和递归 ω-语言 [20]。与之密切相关的是，Wadge 预序通过连续函数归约 [40], 来分类 ω-语言，而 Wagner 层级则根据 ω-正则语言在 Wadge 层级中的位置对其进行分类 [31]。与这些侧重于语言族的方法不同，我们的研究探讨单个流的复杂度。

可计算性与复杂性理论归约 可计算性理论引入了若干可约性概念，可用于判定问题的分类，例如多对一归约和图灵归约 [34,35]。将这些归约限制在多项式运行时间内，即可得到它们的直接类比，并在复杂性理论中得到广泛研究 [2]。我们的 1CA 可约性概念与这些经典归约有相似之处，但专门用于流的比较。

3 预备知识

我们采用冯·诺依曼序数，其中 $0 := \emptyset$，且 $n+1 := n \cup \{n\}$。这意味着 $1 = \{0\}$、$2 = \{0,1\}$，依此类推。第一个极限序数是 $\omega = \{0,1,2,\dots\}$。我们用 $n \mid m$ 表示 n 整除 m。模 m 的同余记作 $a \equiv b \pmod{m}$。n 与 m 的最大公约数和最小公倍数分别记为 $\gcd(n,m)$ 和 $\operatorname{lcm}(n,m)$。给定集合 X 与 Y，定义域为 X、陪域为 Y 的函数记作 $f\colon X \to Y$，此类函数的集合记为 Y^X。函数复合由 $(g \circ f)(x) = g(f(x))$ 给出。部分函数 $f\colon X \rightharpoonup Y$ 是一个函数 $f\colon X \to Y \cup \{\bot\}$，其中 $\bot$ 表示未定义值。我们在函数定义中使用下划线“_”作为占位符或通配符；例如，$f(_,_,x) = x$ 定义了忽略前两个参数并返回第三个参数的函数。

3.1 有限词与无限词

字母表是一个至少包含两个元素的有限集合，其成员称为字母。给定字母表 Σ，长度为 $n \in \omega$ 的词集合是 Σ^n。有限词 $w \in \Sigma^n$ 的长度为 $|w| = n$。空词记为 ε，其中 $\Sigma^0 = \{\varepsilon\}$。$\Sigma$ 上所有有限词的集合由 Kleene 闭包 $\Sigma^* = \bigcup_{n\in\omega} \Sigma^n$ 给出。我们还定义 $\Sigma^{\leq n} = \{w \in \Sigma^* \mid |w| \leq n\}$。对于 $w \in \Sigma^*$ 和 $a \in \Sigma$，我们用 $a{:}w$ 表示词扩展，意为将 a 前置于 w。词 w_1 和 w_2 的连接记为 $w_1 \cdot w_2$，其递归定义为 $\varepsilon \cdot w = w$ 和 $(a{:}w_1)\cdot w_2 = a{:}(w_1 \cdot w_2)$。重复连接记为 w^n，其中 $w^0 = \varepsilon$ 且 $w^{n+1} = w \cdot w^n$。对于二进制词 $w \in 2^{=n}$，我们将按位补码 $w \in 2^{=n}$ 定义为 $\overline{w}(i) = 1 \Leftrightarrow w(i) = 0$。我们令 $w(n\colon m) = w(n) \cdot w(n+1) \cdots w(m)$。我们将经常提到词的“移位”：对于 $w \in \Sigma^n$ 和 $i < n$，i-移位是循环置换 $w(i+1,\dots,n-1) \cdot w(0,\dots,i)$。

Σ 上的无限流是一个函数 $\sigma : \omega \to \Sigma$，也可视为一个元素 $\sigma \in \Sigma^\omega$。流可以分解为 $\sigma = a{:}\sigma'$，其中 $a \in \Sigma$ 是头部，$\sigma' \in \Sigma^\omega$ 是尾部。我们用 $2 = \{0, 1\}$ 表示双字母字母表，二进制流的集合为 2^ω，称为康托集。流也可以通过拼接无穷多个有限词来构造；对于由 Σ 上的词组成的序列 $\{w_i\}_{i\in\omega}$，记作 $\prod_{i=1}^{\infty} w_i$。

3.2 流的构造

构造流有多种方法，包括自动序列 [1], 态射词 [29], 林登迈尔系统 [28], 和托普利茨词 [23,7]。我们通过递归方程以余归纳方式定义流，这种方法表达力强且简洁，适合构造流。

例 *3.1.* 图厄-莫尔斯序列在在线整数序列百科全书（OEIS）中的代码为 [A010060]，其定义为 $\mathsf{TM} = 0{:}X$，其中 $X = 1{:}\mathsf{zip}(X, Y)$，$Y = 0{:}\mathsf{zip}(Y, X)$，且 $\mathsf{zip}(x{:}\sigma, \tau) = x{:}\mathsf{zip}(\tau, \sigma)$。展开这个余归纳定义，

$$\begin{aligned}\mathsf{TM} &= 0{:}X\\ &= 0{:}1{:}\mathsf{zip}(X, Y)\\ &= 0{:}1{:}\mathsf{zip}(1{:}\mathsf{zip}(X, Y), Y)\\ &= 0{:}1{:}1{:}\mathsf{zip}(Y, \mathsf{zip}(X, Y))\\ &= 0110100110010110\ldots\end{aligned}$$

例 *3.2.* 倍周期序列在 OEIS 中的代码为 [A096268]，其定义为 $\mathsf{PD} = \mathsf{zip}(\mathsf{ones}, \mathsf{inv}(\mathsf{PD}))$，其中 $\mathsf{ones} = 1{:}\mathsf{ones}$，且 $\mathsf{inv}(x{:}\sigma) = 1{:}\mathsf{inv}(\sigma)$（若 $x = 0$），否则 $0{:}\mathsf{inv}(\sigma)$。展开此定义可得 $\mathsf{PD} = 1011101010111011\ldots$

3.3 有限状态转换器

有限状态转换器（FST）是能够将一个流转换为另一个流的有限自动机。它们通过从左到右遍历输入流中的每个字母，并为每个字母生成一个有限字作为输出来实现这一点。有关有限状态转换器的详细介绍，请参阅 [1]。

定义 3.3。有限状态转换器（*FST*）表示为一个元组 $(Q, \Sigma, q_0, T, \lambda)$，其中 Q 是有限的状态集合，Σ 是一个至少包含两个元素的有限集合，称为字母表，$q_0 \in Q$ 是初始状态，$T\colon \Sigma \times Q \to Q$ 是转移函数，而 $\lambda\colon \Sigma \times Q \to \Sigma^*$ 是输出函数。

定义 3.4。给定一个 *FST* $M = (Q, \Sigma, q_0, T, \lambda)$，对应于 M 的流变换 $f_M : Q \times \Sigma^\omega \rightharpoonup \Sigma^\omega$ 为

$$f_M(q, a{:}\sigma) = \lambda(q, a) \cdot f_M(T(q, a), \sigma),$$

并且我们使用简写记法 $f_M(\sigma) = f_M(q_0, \sigma)$。

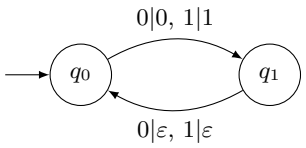


图 1：一个将 zip(σ, τ) 归约到σ 的有限状态转导器，适用于任意 σ, $\tau \in \Sigma^\omega$。

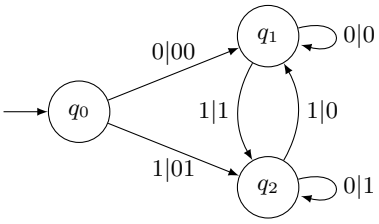


图 2：一个将倍周期序列归约到图灵机的有限状态转导器。

这些变换通过偏函数建模，因为有限状态转换器可以无限次输出空词，使结果成为有限序列，因此不再是流。

有限状态转换器为比较流的复杂度提供了一种结构化方法 [13,3,33]。如果一个流 σ 可以通过有限状态转换器转换为另一个流 τ，那么 σ 被认为至少与 τ一样复杂。这引出了下面关于有限状态转导器归约的形式化定义。

定义 3.5。有限状态转导器归约 是一种关系 $\triangleright^{\mathrm{F}} \subseteq \Sigma^\omega \times \Sigma^\omega$，对于所有$\sigma$和$\tau \in \Sigma^\omega$, 当且仅当存在一个有限状态转导器 M 使得$f_M(\sigma) = \tau$时，$\sigma \triangleright^{\mathrm{F}} \tau$ 成立。

Additionally, we let $\sigma \triangleleft^{\mathrm{F}} \tau \Leftrightarrow \tau \triangleright^{\mathrm{F}} \sigma$, *and* $\diamond^{\mathrm{F}} = \triangleright^{\mathrm{F}} \cap \triangleleft^{\mathrm{F}}$.

例 *3.6*。如第 3.2 节所定义，交织函数会交错两个流。图 1 展示了一个将 zip(σ, τ) 归约到 σ的有限状态转导器，这意味着 zip(σ, τ)$\triangleright^{\mathrm{F}}\sigma$。每条转移边都标记为 $a|w$，表示读取字母 a，并将单词 w 作为输出返回。

例 *3.7*。图 2 展示了一个有限状态转导器，它读取倍周期序列并输出图厄-莫尔斯序列；这两个序列分别在例 3.1 和例 3.2 中介绍。

4 元胞自动机可归约性

本节通过一维元胞自动机（一维元胞自动机）介绍流可约性的概念，并分析其一些基本代数性质。我们将流视为由细胞组成的一维构型，这些构型可以通过全局应用局部规则来更新，从而生成后续的流。此处介绍的定义与经典研究 [41],中探讨的一维元胞自动机的一般概念相一致，其中局部规则映射到单个字母。关于元胞自动机可约性的其他定义，请参见第 7 节。

定义 4.1. 一维元胞自动机（*1CA*）是一个三元组 (Σ, N, δ)，其中 Σ是一个至少包含两个元素的有限集合，称为字母表；$N \in \mathbb{N}$ 是邻域半径；$\delta : (\Sigma \cup \{\bot\})^{2N+1} \to \Sigma$ 是局部规则。

定义 4.2. 给定一个流 $\sigma \in \Sigma^\omega$ 和一个一维元胞自动机 $M = (\Sigma, N, \delta)$，其（局部）邻域函数是映射

$$\mathcal{V}_{\sigma,N}(i) = (\sigma(i-N), \ldots, \sigma(i), \ldots, \sigma(i+N)),$$

其中，对于任意 $j < 0$，我们令 $\sigma(j) = \perp$。M 的全局更新规则 是一个流变换 $f_M\colon\ \Sigma^\omega \to \Sigma^\omega$ 定义为

$$f_M(\sigma) = \delta \circ \mathcal{V}_{\sigma,N}.$$

例 *4.3*。固定一个二元字母表 $\Sigma = 2 = \{0,1\}$，我们可以定义一个一维元胞自动机 M，使得

$$f_M(x{:}y{:}\sigma) = (x \oplus y){:}f_M(y{:}\sigma),$$

其中“$\oplus$”表示模二加法（即异或）。由于每个细胞只需要查看其右侧的紧邻单元，因此只需令 $N = 1$ 并定义$\delta(_,\ x,\ y) = x \oplus y$即可。有趣的是，这一操作将众所周知的序列 TM $\triangleright^{\mathrm{C}}$ 倍周期序列 [13,12]进行了约化。

例 *4.4*。给定 $\Sigma = \{a, b, c\}$，我们可以定义一个一维元胞自动机 $M = (\Sigma,\ N,\ \delta)$，使得$f_M(\sigma) = ab \cdot \sigma$，令 $N = 2$ 并定义

$$\delta(x, y, z, _, _) = \begin{cases} a & \text{if } x = y = \perp, \\ b & \text{else if } x = \perp, \\ x & \text{else.} \end{cases}$$

通过以下定义，我们引入 1CA 可约性的概念，即在一次全局更新中将一个流转换为另一个流的能力。

定义 4.5。*1CA* 归约是一种关系 $\triangleright^{\mathrm{C}} \subseteq \Sigma^\omega \times \Sigma^\omega$，对于所有$\sigma$、$\tau \in \Sigma^\omega$，当且仅当存在一个 *1CA* M使得 $f_M(\sigma) = \tau$时，关系成立。
Additionally, we let $\sigma \triangleleft^{\mathrm{C}} \tau \Leftrightarrow \tau \triangleright^{\mathrm{C}} \sigma$, *and* $\diamond^{\mathrm{C}} = \triangleright^{\mathrm{C}} \cap \triangleleft^{\mathrm{C}}$.

我们将在下一个命题中证明其传递性。因此，尽管 1CA 可约性表示单次更新，我们仍可以将其视为经过有限多个步骤。

命题 4.6。关系 $\triangleright^{\mathrm{C}}$ 是一个预序。

证明。令 σ、τ、$\rho \in \Sigma^\omega$。

– 通过恒等 1CA，我们得到 $\sigma \triangleright^{\mathrm{C}} \sigma$，$\delta(x) = x$。– 假设 $\sigma \triangleright^{\mathrm{C}} \tau$ 通过 $M_1 = (\Sigma, N_1, \delta_1)$，且 $\tau \triangleright^{\mathrm{C}} \rho$ 通过 $M_2 = (\Sigma, N_2, \delta_2)$。那么，存在 $M_3 = (\Sigma, N_1 + N_2, \delta_3)$，使得令.....后 $\sigma \triangleright^{\mathrm{C}} \rho$ 通过 M_3

$$\delta_3(x_0, \ldots, x_{2(N_1+N_2)}) := \delta_2(\delta_1(v_0), \ldots, \delta(v_{2N_2})),$$

其中 $v_i :=\ (x_i, \ldots, x_{i+2N_1})$。 □

推论 4.7。 $\diamond^{\mathrm{C}}$ 是一个等价关系。 □

例 4.4 展示了如何利用一维元胞自动机将字符串 ab 前置到流中。这一具体情形引出了一个更一般的结果：一维元胞自动机可以插入或移除有限字符串，如下述命题所述。

命题 4.8。 对于任意 $\sigma \in \Sigma^{\omega}$、 $w \in \Sigma^{*}$，我们有 $w \cdot \sigma \diamond^{\mathrm{C}} \sigma$。

证明。 给定字母 $a \in \Sigma$ 和流 $\sigma \in \Sigma^{\omega}$，我们可以用 $M = (\Sigma, \mathbf{1}, \delta)$ 将 a 附加到 σ，其中

$$\delta(x, y, _) = \begin{cases} a & \text{if } x = \bot, \\ x & \text{else.} \end{cases}$$

由于 $\triangleright^{\mathrm{C}}$ 具有传递性，一维元胞自动机可以插入或前置有限词。

对称地，由于可以用一维元胞自动机计算流的尾部，其中 $\delta(_, _, x) = x$，根据传递性可以移除有限个字母。□

推论 4.9. 关系 $\triangleright^{\mathrm{C}}$ 在有限变异下封闭；换言之，对所有 σ、 $\tau \in \Sigma^{\omega}$，若集合 $\{i \in \mathbb{N} \mid \sigma(i) \neq \tau(i)\}$ 是有限的，则$\sigma \triangleright^{\mathrm{C}} \tau$成立。 □

接下来，我们证明任意字母表都可以用二进制编码，且保持 1CA 可约性。对于该证明，必须考虑到并非每一种编码都足够。为说明这一点，考虑将字母表 $\Sigma' = \{\mathsf{A}, \mathsf{B}, \mathsf{C}\}$ 朴素地编码为 $\psi(\mathsf{A}) = 00$、 $\psi(\mathsf{B}) = 01$ 和 $\psi(\mathsf{C}) = 10$。我们对任意字 σ 重载 $\psi(\sigma) = \prod_{i=0}^{\infty} \psi(\sigma(i))$。于是，流 B^{ω} 被编码为 $(01)^{\omega} = 0101\ldots$，而 C^{ω} 被编码为 $(10)^{\omega} = 1010\ldots$，它们最终对任何一维元胞自动机规则而言看起来彼此无法区分。虽然这不会立即导致矛盾，因为 $B^{\omega} \diamond^{\mathrm{C}} C^{\omega}$ 并且 $(01)^{\omega} \diamond^{\mathrm{C}} (10)^{\omega}$ 也是如此，但很容易构造一个该朴素编码失效的例子。例如，若我们令

$$\sigma = \prod_{i=0}^{\infty} B^i C^i, \qquad \tau = \prod_{i=0}^{\infty} C^i B^i,$$

则 $\sigma \diamond^{\mathrm{C}} \tau$，而 $\psi(\sigma) \not\triangleright^{\mathrm{C}} \psi(\tau)$，因为对任意一维元胞自动机 (Σ, δ, N)， B^{N+1} 与 C^{N+1} 的局部邻域看起来无法区分，即

$$\mathcal{V}_{B^{\omega}, N}(N) = \mathcal{V}_{C^{\omega}, N}(N + 2),$$

然而， δ 必须将其中一个映射为$\mathbf{0}$，将另一个映射为$\mathbf{1}$。在下述结果中，我们构造一种合适的保持可约性的编码，以避免上述问题。

引理 4.10. 设 Σ 是一个字母表，即一个至少含有两个元素的有限集合。那么，存在一个映射 $\phi : \Sigma \to 2^{*}$，使得

$$\forall \sigma, \tau \in \Sigma^{\omega}, \sigma \triangleright^{\mathrm{C}} \tau \Leftrightarrow \phi(\sigma) \triangleright^{\mathrm{C}} \phi(\tau).$$

证明. 为构造一个合适的编码，令 $d = \lceil \log_2(|\Sigma|) \rceil$，并考虑一个单射 $\gamma\colon \Sigma \to 2^d$。那么，该映射

$$\phi(a) := 1 \cdot \gamma(a) \cdot 1 \cdot 0^{2^d+1}$$

满足上述双向蕴含：

($\Rightarrow$) 假设 $\sigma \triangleright^{\mathrm{C}} \tau$ 满足 $M = (\Sigma, N, \delta)$，也就是说，对于所有 $i \in \mathbb{N}$, $\delta(\mathcal{V}_{\sigma,N}(i)) = \tau(i)$。为了构造一个将 $\phi(\sigma)$ 约化为 $\triangleright^{\mathrm{C}} \phi(\tau)$ 的一维元胞自动机，我们定义 $M' = (2, N', \delta')$，使其满足 $N' = N \cdot (2^{d+1} + 3)$。注意， $\phi(\sigma)$ 的任意长度为 N' 的子串都是如下形式的一个字的移位

$$1 \cdot w_1 \cdot 1 \cdot 0^{2^d+1} \cdot 1 \cdot w_2 \cdot 1 \cdot 0^{2^d+1} \cdots 1 \cdot w_N \cdot 1 \cdot 0^{2^d+1}.$$

将上述字的移位记为 w'。接着，按如下方式定义 δ'，即可得到所需的约化：如果 w'.....中间位置的字母

– .....属于一个长度为 $2^d + 1$ 的零块，则返回 0。

– .....是一个紧邻长度为 $2^d + 1$ 的零块的 1，则返回 1。

– .....属于某个 w_{i_0}。由于 γ 是单射，我们可以考虑词 $\prod_{j=1}^{N} \gamma^{-1}(w_j)$，并对其进行移位，使 $\gamma^{-1}(w_i)$ 位于中心；将其记为 v。然后，返回 $\phi(\delta(v))$ 中间的字母，并将其移位，使其与 w' 对齐，因为 $\delta(v)$ 会通过 δ，借助 $\sigma \triangleright^{\mathrm{C}} \tau$，在 τ 中产生相应的字母。

因此，根据构造，我们有

$$f_{M'}(\phi(\sigma)) = \prod_{j=0}^{\infty} 1 \cdot \gamma(\tau(j)) \cdot 1 \cdot 0^{2^d+1} = \phi(\tau).$$

($\Leftarrow$) 假设 $\phi(\sigma) \triangleright^{\mathrm{C}} \phi(\tau)$，其中 $M = (2, N, \delta)$，即

$$\phi(\sigma) = \prod_{i=0}^{\infty} 1 \cdot \gamma(\sigma(i)) \cdot 1 \cdot 0^{2^d+1}$$

可约化为

$$\phi(\tau) = \prod_{i=0}^{\infty} 1 \cdot \gamma(\tau(i)) \cdot 1 \cdot 0^{2^d+1}.$$

从高层次来看，由于与 ($\Rightarrow$) 方向具有对称性，我们令一维元胞自动机 $M' = (2, N', \delta')$，其中 $N' = \lceil N/(2^{d+1} + 3) \rceil$，通过从 δ 中提取 δ' 得到。其想法是，如果邻域中的词 $w_0 \in 2^d$

$$w = \prod_{i=-N'}^{N'} 1 \cdot \gamma(w_i) \cdot 1 \cdot 0^{2^d+1}$$

被映射为某个词 $v \in 2^d$ （尽管 δ 输出单个比特，但可以将它们拼接起来以得到 v），我们只需令

$$\delta'(\gamma^{-1}(w_{-N'}), \ldots, \gamma^{-1}(w_{N'})) := \gamma^{-1}(v). \qquad \square$$

引理 4.10 说明将字母表限制为 $2 = \{0, 1\}$ 是合理的，因为任何其他字母表上的层级中出现的代数结构，也会出现在 2 上的层级中。因此，在示例中我们主要考虑二进制流，尽管我们的定理可能仍涉及用 Σ表示的一般字母表。值得注意的是，虽然可以为有限状态转换器证明一个类似的引理 [13]„ 但其方法截然不同：有限状态转导器通过从左到右扫描来处理流，这意味着流的开头可以充当参照系，因此，简单的编码就足以证明可约性得以保留。

4.1 比较有限状态转导器归约与 1CA 归约

本节比较第 3.3 节中介绍的有限状态转导器可约性与 1CA 可约性。关键区别在于，有限状态转换器具有记忆，而一维元胞自动机没有记忆。我们利用这一差异，在经典分离结果 [32]的精神下，确立这两个族之间的分离。

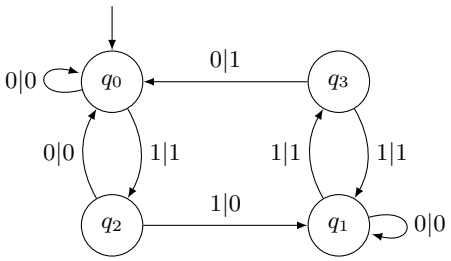


图 3：将 σ 归约为 τ 的有限状态转导器的状态转移图，如命题 4.11 所定义。

命题 4.11。 $\triangleright^{\mathrm{F}} \not\subseteq \triangleright^{\mathrm{C}}$。

证明。 令 $\xi =(001)^{\omega}$，并定义以下两个流：

$$\sigma := \prod_{i\in\mathbb{N}} 0^{i+1} \cdot 1 \cdot \xi(i+1), \quad \text{and} \quad \tau := \prod_{i\in\mathbb{N}} 0^{i+1} \cdot 1 \cdot \xi(i).$$

可以按如下方式构造一个将 σ 约化为 τ 的有限状态转导器：

1. 跳过所有填充 0。2. 遇到 1 时，下一个字母属于 ξ。用一个状态记住它，并将其替换为之前读到的字母。3. 返回步骤 1。

该有限状态转导器的显式状态转移图如图 3 所示。

然而，元胞自动机无法执行这种约简。为说明这一点，假设存在一个一维元胞自动机 $M=(\Sigma,\ N,\ \delta)$，其中 $f_M(\sigma)=\tau$。根据 σ 的构造，存在一个 $j_0\in\mathbb{N}$，使得对于所有 $j\geq j_0$，都存在 $k\in\mathbb{N}$，其中

$$\mathcal{V}_{\sigma,N}(k)=0^{N-1}\cdot 1\cdot\xi(j+1)\cdot 0^N$$

（分量 $\xi(j+1)$ 位于邻域的中间）。任取满足 $j\equiv 0$ 模 3 和 $j\equiv 2$ 模 3 的此类 $j,\ j'\geq j_0$，并取相应的 $k,\ k'\in\mathbb{N}$。由于 $\xi(j+1)=\xi(j'+1)=0$，我们得到 $\mathcal{V}_{\sigma,\ N}(k)=\mathcal{V}_{\sigma,\ N}(k')$。然而，根据假设 $f_M(\sigma)=\tau$，我们有

$$0=\xi(j)=\delta(\mathcal{V}_{\sigma,N}(k))=\delta(\mathcal{V}_{\sigma,N}(k'))=\xi(j')=1,$$

矛盾。 □

命题 4.12. $\triangleright^{\mathrm{C}}\subseteq\triangleright^{\mathrm{F}}$。

证明。 假设 $\sigma\triangleright^{\mathrm{C}}\tau$ 可由一维元胞自动机 $M=(\Sigma,\ N,\ \delta)$ 实现。我们构造一个有限状态转导器 $M'=(Q,\ \Sigma,\ q_0,\ T,\ \lambda)$ 来实现 $\sigma\triangleright^{\mathrm{F}}\tau$，令 $Q=(\Sigma\cup\{\perp\})^{2N+1}$、$q_0=(\perp,\ \ldots\ldots,\ \perp)$，并定义 T 和 λ 如下：

1. 从 $q_0=(\perp,\ \ldots,\ \perp)$ 开始，并在任意时刻以先进先出方式记录当前状态下下一个读取的字母，并忘记最早读取的字母：

$$T(q,a)=(q(1),\ldots,q(2N),a).$$

2. 读取前 N 个字母时，σ 中第一个字母的完整邻域仍未读到。因此，对于任意满足 $\perp$ 至少出现在前 $N+1$ 个位置中的 q，我们令 $\lambda(q,\ a)=\varepsilon$。

3. 读取第 $N+1$ 个字母后，第一个细胞的整个邻域已编码在 q 中，因此对于任意少于 $N+1$ $\perp$ 个字母的 q，我们可以定义 $\lambda(q,\ a)=\delta(q)$。

因此，由 M' 的定义可知

$$f_{M'}(q_0,\sigma)=\prod_{i=0}^{\infty}\delta(\mathcal{V}_{\sigma,N}(i))=f_M(\sigma)=\tau,$$

从而 $\sigma\triangleright^{\mathrm{F}}\tau$。 □

先前的研究引入了 Δ-轨道的概念：通过反复应用异或有限状态转换器 [12,13] 得到的轨道。我们可以检验两个流的轨道是否重叠，这可能揭示一些著名流之间有趣的联系，否则这些联系很难被发现。鉴于异或运算可以通过初等元胞自动机实现（如例 4.3 中的构造），且一个Δ-轨道不过是它的时空图，从一维元胞自动机的角度来看待这一技术，或许能带来更多见解，正如我们在 B 节中所讨论的那样。

5 流的度

在可计算性理论中，利用图灵机比较自然数集合。如果两个集合图灵等价，它们就属于同一个图灵度；随后可以比较这些度，从而形成一个层级 [34,35]。本节介绍与可计算性理论中的定义相对应的元胞自动机归约相关定义。我们首先在以下定义中形式化 $\diamond^{\mathrm{C}}$-度的概念。

定义 5.1. $\diamond^{\mathrm{C}}$-度的层级 是商集 $\mathfrak{D} = \Sigma^{\omega}/\diamond^{\mathrm{C}}$，我们将其成员称为 $\diamond^{\mathrm{C}}$-度。 $\sigma \in \Sigma^{\omega}$ 的度记作 $[\sigma]_{\diamond^{\mathrm{C}}}$。

接下来，我们将 1CA 归约的概念扩展到用于比较流的度。

定义 5.2. 流可约性 是一个关系 $\leq^{\mathrm{C}} \subseteq \mathfrak{D} \times \mathfrak{D}$，满足对于所有$X$、$Y \in \mathfrak{D}$，都有 $X \leq^{\mathrm{C}} Y \Leftrightarrow \forall \sigma \in X$、$\forall \tau \in Y$、 $\tau \triangleright^{\mathrm{C}} \sigma$。

根据命题 4.6，该层级构成一个偏序集，且具有 $\leq^{C}$ 。

引理 5.3. $(\mathfrak{D}, \leq^{\mathrm{C}})$ 是一个偏序集。 □

类似于图灵度， $\diamond^{\mathrm{C}}$-度是可数的，因为每个一维元胞自动机都有有限描述，这意味着一维元胞自动机只有可数多个。此外，每个 $\diamond^{\mathrm{C}}$-度都是无限的，因为由推论 4.9，这些度在有限变异下是封闭的。

引理 5.4. 每个 $\diamond^{\mathrm{C}}$-度都是可数无限的。

证明。 设 $\sigma \in \Sigma^{\omega}$ 并考虑其度 $[\sigma]_{\diamond^{\mathrm{C}}}$。对每个 $\tau \in [\sigma]_{\diamond^{\mathrm{C}}}$，存在一维元胞自动机 M，使得 $f_M(\sigma) = \tau$。因此，若我们定义 $g(\tau) = M$，则 g是单射的，因为 $g(\tau) = g(\tau')$蕴含

$$\tau = f_{g(\tau)}(\sigma) = f_{g(\tau')}(\sigma) = \tau'.$$

因此，由于一维元胞自动机的集合是可数的，我们得到从 $[\sigma]_{\diamond^{\mathrm{C}}}$ 到 $\mathbb{N}$的一个单射。 □

推论 5.5。存在 $2^{\aleph_0}$ 多个 $\diamond^{\mathrm{C}}$-度。 □

$\diamond^{\mathrm{C}}$-度的流的层级结构可以看作一个包含不可数多个度的“宇宙”，这些度可以通过 $\leq^{\mathrm{C}}$进行比较；每个度都是可数无限的，从而形成一个偏序集结构。

5.1 最终 k-周期流

最终周期流（即最终会变为周期流的流）由于其普遍性和简单性而值得研究。这类流属于 Mealy机和有限状态转换器 [3,13]层级结构中的最低度。本节将这些流置于一维元胞自动机层次结构中。为完整起见，我们首先简要定义流周期性；这是一个经典概念，关于最终周期序列和最小周期的背景知识，请参见例如 [1] 。

定义 5.6。 设 $\sigma \in \Sigma^{\omega}$。若存在 $n \in \mathbb{N}$，使得对所有 $i \geq n$， $\sigma(i+p) = \sigma(i)$，则称数 $p \in \mathbb{N}_{>0}$ 为 σ 的周期。

定义 5.7。 σ 的周期集合记为 $\mathrm{Per}(\sigma)$。若 $\mathrm{Per}(\sigma) \neq \emptyset$，则称 σ 是最终周期的。我们称 $\mathrm{Per}(\sigma)$ 中的最小元素为 σ 的（最小）周期，并将其记作$\mathrm{per}(\sigma)$。

定义 5.8。 对于 $k \in \mathbb{N}^{>0}$，最终 k-周期流的集合为

$$\mathcal{T}_k = \{\, \sigma \in \Sigma^{\omega} \mid \sigma \textit{ ultimately periodic and } \mathrm{per}(\sigma) = k \,\}.$$

定义 5.9。 最终周期流的集合为

$$\mathcal{T} = \bigcup_{k \in \mathbb{N}_{>0}} \mathcal{T}_k.$$

例 *5.10.* 考虑以下二进制流:

$$\sigma_2 = 1010101010\ldots \quad \text{and} \quad \sigma_6 = 011101\,011101\,\ldots.$$

于是 $\mathrm{per}(\sigma_2) = 2$ 且 $\mathrm{per}(\sigma_6) = 6$，因此 $\sigma_2 \in \mathcal{T}_2$ 且 $\sigma_6 \in \mathcal{T}_6$。虽然 6 也是 σ_2的一个周期，但分类只取决于最小周期 $\mathrm{per}(\sigma_2)$。

命题 5.11. $\mathcal{T}_1$ 对 Σ^{ω}是 $\triangleright^{\mathrm{C}}$-最小的，换言之，

$$\forall \sigma \in \Sigma^{\omega}, \forall \kappa \in \mathcal{T}_1, \sigma \triangleright^{\mathrm{C}} \kappa.$$

证明。设 $\sigma \in \Sigma^{\omega}$ 和 $\kappa = x \cdot y^{\omega} \in \mathcal{T}_1$ 满足 $|y| = 1$。我们有 $\sigma \triangleright^{\mathrm{C}} y^{\omega}$，这是由 $\delta(_) = y$得出的；此外，根据推论 4.9，我们有 $y^{\omega} \triangleright^{\mathrm{C}} \kappa$ 。因此，由传递性可得 $\sigma \triangleright^{\mathrm{C}} \kappa$。□

此外，由命题 4.12 可知 $\triangleright^{\mathrm{C}} \subseteq \triangleright^{\mathrm{F}}$ ，又由于 $\mathcal{T}$在 $\triangleright^{\mathrm{F}}$ [13]下向下封闭，因此 $\mathcal{T}$ 在 $\triangleright^{\mathrm{C}}$下向下封闭。

推论 5.12。 $\mathcal{T}$ 在 $\triangleright^{\mathrm{C}}$ 下向下封闭，即:

$$\forall \sigma \in \mathcal{T}, \tau \in \Sigma^{\omega},\ \textit{if } \sigma \triangleright^{\mathrm{C}} \tau \textit{ then } \tau \in \mathcal{T}. \qquad \square$$

尽管我们通过关于有限状态转换器的一个事实得到了推论 5.12，但直观上，对周期流中的所有字母应用相同的局部规则，会得到另一个周期流。接下来，我们将证明辅助引理 5.13 和 5.15，以便稍后在定理 5.16 中展示： $\mathcal{T}_m$ 可约化为 $\mathcal{T}_n$ 当且仅当 n 整除 m。

引理 5.13. $\forall \tau,\ \tau' \in \mathcal{T}_k,\ \tau \diamond^{\mathrm{C}} \tau'$。

证明。设 $\tau,\ \tau' \in \mathcal{T}_k$。根据 $\diamond^{\mathrm{C}}$ 在推论 4.9 下对有限变异封闭，我们可以假设 τ 和 τ' 都是周期的，即 $\tau = w^\omega$ 和 $\tau' = w'^\omega$。那么，映射

$$(w(i-k \mod k), \ldots, w(i), \ldots, w(i+k \mod k)) \mapsto w'(i)$$

可以通过一维元胞自动机实现，因为 τ 和 τ' 没有子周期，即对于 $k' < k$，$\tau, \tau' \notin \mathcal{T}_{k'}$，因此局部邻域不会与其自身的（非平凡）移位重叠。□

例 *5.14*。让我们回顾一下例 5.10 中的 $\sigma_2 = 10101010\cdots \in \mathcal{T}_2$ 和 $\sigma_6 = 011101\cdots \in \mathcal{T}_6$。注意，下面这个带有 $N = 5$ 的规则可以约化 $\sigma_6 \triangleright^{\mathrm{C}} \sigma_2$:

$$\begin{aligned}
&\delta(_,_,_,_,_,0,1,1,1,0,1) = 1, \qquad \delta(_,_,_,_,_,1,1,1,0,1,0) = 0,\\
&\delta(_,_,_,_,_,1,1,0,1,0,1) = 1, \qquad \delta(_,_,_,_,_,1,0,1,0,1,1) = 0,\\
&\delta(_,_,_,_,_,0,1,0,1,1,1) = 1, \qquad \delta(_,_,_,_,_,1,0,1,1,1,0) = 0.
\end{aligned}$$

然而，反过来则不成立，也就是说，$\sigma_2 \not\triangleright^{\mathrm{C}} \sigma_6$。这一思想在下文的引理 5.15 和定理 5.16 中得到了正式阐述。

引理 5.15。对于 $\tau \in \mathcal{T}_m$，如果 $\tau \triangleright^{\mathrm{C}} \sigma$，那么 σ 的最终周期大小为 m。

证明。只需证明当 $\tau = (1\cdot 0^{m-1})^\omega$ 时该性质成立，因为对于任意其他的 $\tau' \in \mathcal{T}_m$，若 $\tau' \triangleright^{\mathrm{C}} \sigma$，则根据引理 5.13，$\tau \triangleright^{\mathrm{C}} \sigma$。因此，假设 $\tau \triangleright^{\mathrm{C}} \sigma$通过 $M = (2,\ N,\ \delta)$。由于对所有 $j \geq N+1$，都有 $\mathcal{V}_{\tau,\ N}(j) = \mathcal{V}_{\tau,\ N}(j+m)$，于是得到

$$\sigma(j) = \delta(\mathcal{V}_{\tau,N}(j)) = \delta(\mathcal{V}_{\tau,N}(j+m)) = \sigma(j+m),$$

换言之，σ 的最终周期为 m。 □

定理 5.16。 $\mathcal{T}_n \leq^{\mathrm{C}} \mathcal{T}_m$ 当且仅当 $n \mid m$。

证明。回忆一下，我们用 $n \mid m$ 表示“n 整除 m”。由于引理 5.15意味着 $\mathcal{T}_n \leq^{\mathrm{C}} \mathcal{T}_m \Rightarrow n \mid m$，我们只需证明（⇐）方向。假设 $n \mid m$，根据引理 5.13，我们只需考虑 $\tau_m = (1\cdot 0^{m-1})^\omega$ 和$\tau_n = (1\cdot 0^{n-1})^\omega$。由于 $n \mid m$，对于某个 k，有 $|w_m| = k \cdot |w_n|$。因此，对于$i = 0$、.....、$m-1$，我们可以仅根据大小为 $m-1$ 的有限邻域，将 $w_m(i)$与 $w_n(i \bmod n)$匹配，这与例 5.14中构造局部规则的方法类似，因为 w_m 没有子周期。□

推论 5.17。 以下结论由定理 *5.16*推出

1. 每个 $\mathcal{T}_k$ 都是一个 $\diamond^{\mathrm{C}}$-度。 *2.* $\mathcal{T}_1$ 是最低的 $\diamond^{\mathrm{C}}$-度。 *3.* $(\{\mathcal{T}_k\}_{k\in\mathbb{N}}, <^{\mathrm{C}})$ 是良基的。 *4.* 对所有 $k>1$，序列 $\mathcal{T}_1 <^{\mathrm{C}} \mathcal{T}_k <^{\mathrm{C}} \mathcal{T}_{k^2} <^{\mathrm{C}} \mathcal{T}_{k^3} <^{\mathrm{C}} \dots$ 都是无限上升链。□

经典文献将度 P 定义为一个原子（也称为素度），条件是 P 与底度 $\mathcal{T}_1$之间不存在严格介于其间的度。以往研究特别关注原子 [13,24], ，因为原子的存在意味着层级并非稠密，而且可以用来构造有趣的流，例如极值。

定义 5.18。设 $P\in\mathfrak{D}$ 和 $P\neq\mathcal{T}_1$。 P 是一个原子，当且仅当对所有$X\in\mathfrak{D}$，都有 $\mathcal{T}_1\le^{\mathrm{C}} X\le^{\mathrm{C}} P\Rightarrow X=P$ 或 $X=\mathcal{T}_1$。

推论 5.19。 度 $\mathcal{T}_p$ 是原子，当且仅当 $p\in\mathbb{N}$ 是素数。 □

$\mathfrak{D}$ 中存在无限下降链，意味着该层级不是良基的。虽然最终 k-周期的流不会形成这样的链，但我们可以利用它们构造一条链。思路是将每个$\mathcal{T}_k$ 中的信息编码到一个流中，我们可以抽象地将其表示为 $\langle\mathcal{T}_2,\ \mathcal{T}_3,\ \mathcal{T}_4,\ \mathcal{T}_5,\ \dots\rangle$。如果成功，我们就能将其严格约化为 $\langle\mathcal{T}_1,\ \mathcal{T}_3,\ \mathcal{T}_4,\ \mathcal{T}_5,\ \dots\rangle$，再将其约化为 $\langle\mathcal{T}_1,\ \mathcal{T}_1,\ \mathcal{T}_4,\ \mathcal{T}_5,\ \dots\rangle$，依此类推。尽管这个元组本身不是流，但我们可以通过拼接越来越长的有限前缀，达到同样的效果。令$\tau_k=(1\cdot 0^{k-1})^\omega\in\mathcal{T}_k$。将下降链中的第一个流定义为

$$\mu_1=\tau_2^{\le 1}\cdot\tau_2^{\le 2}\cdot\tau_3^{\le 3}\cdot\tau_2^{\le 4}\cdot\tau_3^{\le 5}\cdot\tau_4^{\le 6}\cdot\tau_2^{\le 7}\cdot\tau_3^{\le 8}\cdot\tau_4^{\le 9}\cdot\tau_5^{\le 10}\cdot\dots,$$

其中， $\tau_k^{\le l}$ 表示长度为 l 的 τ_k 前缀。于是，我们得到如下无限下降链：

命题 5.20. 存在一条无限下降链

$$\begin{aligned}
\mu_1 &= \tau_2^{\le 1}\cdot\tau_2^{\le 2}\cdot\tau_3^{\le 3}\cdot\tau_2^{\le 4}\cdot\tau_3^{\le 5}\cdot\tau_4^{\le 6}\cdot\tau_2^{\le 7}\cdot\tau_3^{\le 8}\dots\\
\trianglerighteq^{\mathrm{C}}\ \mu_2 &= \tau_1^{\le 1}\cdot\tau_1^{\le 2}\cdot\tau_3^{\le 3}\cdot\tau_1^{\le 4}\cdot\tau_3^{\le 5}\cdot\tau_4^{\le 6}\cdot\tau_1^{\le 7}\cdot\tau_3^{\le 8}\dots\\
\trianglerighteq^{\mathrm{C}}\ \mu_3 &= \tau_1^{\le 1}\cdot\tau_1^{\le 2}\cdot\tau_1^{\le 3}\cdot\tau_1^{\le 4}\cdot\tau_1^{\le 5}\cdot\tau_4^{\le 6}\cdot\tau_1^{\le 7}\cdot\tau_1^{\le 8}\dots\\
\trianglerighteq^{\mathrm{C}}\ &\dots,
\end{aligned}$$

其中“ $\trianglerighteq^{\mathrm{C}}$ ”表示严格可约性。

证明。 对于所有 $i \in \mathbb{N}$，我们需要证明 $\mu_i \not\rhd^{\mathrm{C}}_{=} \mu_{i+1}$。首先通过反证法证明 $\mu_{i+1} \not\rhd^{\mathrm{C}} \mu_i$：假设 $\mu_{i+1} \rhd^{\mathrm{C}} \mu_i$ 经由 $M = (\Sigma,\ N,\ \delta)$。根据 μ_i 和 μ_{i+1}的构造，τ_1 中越来越长的子串在μ_{i+1} 中被映射到 τ_{i+1} 中等长的子串，而这些子串位于 μ_i中。由于这些子串的长度超过邻域大小 $2N+1$，因此有 $f_M(\tau_1) = \tau_{i+1}$，这与定理 5.16 矛盾。

我们通过提供一个合适的一维元胞自动机 $M = (\Sigma,\ N,\ \delta)$ 来证明 $\mu_i \rhd^{\mathrm{C}} \mu_{i+1}$。规则 δ 必须判断 a0 是否属于 τ_{i+1} 的某个子串，并且仅当属于时才将其变为 1。我们将这样的规则定义如下：

– 如果中心的字母是 1，则返回 1。– 如果它是在一段 $k \leq i$ 个零中的一个零，则当且仅当邻域具有 $\ldots 1 \cdot 1 \cdot 0^k \cdot 1 \ldots$ 或 $\ldots 1 \cdot 0^i \cdot 1 \cdot 0^k \cdot 1 \ldots$ 这两种形式之一时返回 1。– 其他情况下，返回 0。

此操作可以通过一条具有 $N = 2i+1$ 的规则来执行。请注意，由于 μ_i 由越来越长的周期子串组成，M “最终正确地约化为 μ_{i+1}”，也就是说，

$$\exists j \in \mathbb{N}, \forall j' \geq j, f_M(\mu_i)(j') = \mu_{i+1}(j').$$

因此，根据推论 4.9，$f_M(\mu_i) \diamond^{\mathrm{C}} \mu_{i+1}$，这意味着 $\mu_i \rhd^{\mathrm{C}} \mu_{i+1}$。 □

例 *5.21*。 如命题 5.20 所示，流 μ_3可通过具有以下规则的 $N = 5$ 的一维元胞自动机归约到 μ_2：

$$\begin{aligned}
&\delta(\ _,\ _,\ _,\ _,\ _,\ 1,\ _,\ _,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ _,\ _,\ _,\ 1,\ 1,\ 0,\ 1,\ _,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ _,\ _,\ _,\ 1,\ 1,\ 0,\ 0,\ 1,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ _,\ _,\ 1,\ 1,\ 0,\ 0,\ 1,\ _,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ _,\ 1,\ 0,\ 0,\ 1,\ 0,\ 1,\ _,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ _,\ 1,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1,\ _,\ _,\ _,\ _) = 1,\\
&\delta(\ 1,\ 0,\ 0,\ 1,\ 0,\ 0,\ 1,\ _,\ _,\ _,\ _,\ _) = 1,
\end{aligned}$$

并且 δ 对任何其他输入都返回 0。

5.2 稀疏流

康托尔对角线论证可用于构造一个不可判定集合 $K \subseteq \mathbb{N}$；对于每一台图灵机 φ_i，若 $\varphi_i(i)$ 停机，则令 $i \in K$，否则令 $i \notin K$。类似的方法可用于构造图灵不可比较集合 [25,21,30], ，这启发我们构造非周期的 $\rhd^{\mathrm{C}}$不可比较流。

注意，半径为 0 的一维元胞自动机无法将字符串 00 映射为字符串 01，因为第一个 0 被映射为 0，而第二个 0 被映射为 1。因此，若令 $\sigma_0 = 001$ 和 $\tau_0 = 010$，这两个字符串无法通过任何半径为 0 的一维元胞自动机进行比较

。类似地，令 $\sigma_1 = 000\ 000\ 010$ 和 $\tau_1 = 000\ 010\ 000$，不存在一个满足 $N = 1$ 、能够比较 σ 和 τ的一维元胞自动机。容易看出，对于 σ_i 和 τ_i，一般规律也是如此。因此，通过分别拼接所有 σ_i 和 τ_i 得到的流 σ 和 τ 是 $\triangleright^{\mathrm{C}}$-不可比较的。

请注意，刚刚构造的不可比较流大多由 0 组成。直觉告诉我们，一般来说，含有许多 0 的流必须针对邻域半径进行“对角化”，因此很难通过一维元胞自动机进行比较。这促使我们引入流稀疏性的概念。

定义 5.22. 流 $\sigma \in \Sigma^{\omega}$ 是稀疏的 当且仅当存在一个填充字母 $p \in \Sigma$，使得 $\forall d \in \mathbb{N}, \exists n_0 \in \mathbb{N}, \forall i \geq n_0, \sigma(i) \neq p \Rightarrow \sigma(i+j) = p$，对于 $j = 1, \ldots, d$.

换言之，如果对于每个 $d \in \mathbb{N}$，都存在 $n_0 \in \mathbb{N}$，使得在 n_0 之后，每个非-p 字母后面至少跟着连续 d 个 p，那么该流就是稀疏的。

例 *5.23.* 下面的流在 $p = 0$ 下是稀疏的：

$$\prod_{i=0}^{\infty} 1 \cdot 0^i = 110100100010000\ldots$$

为简单起见，我们将讨论范围限定为 $\Sigma = 2$ （即 $\{0, 1\}$）；此外，若考虑稀疏流，我们假设 0 是填充字母。尽管如此，本节的结果可以直接推广到任意稀疏流。

很难通过一维元胞自动机比较两个稀疏流，因为最终几乎整个邻域都只包含填充字母 0。事实上，接下来我们将证明，非最终恒定的稀疏流都是原子。

定理 5.24。 任意稀疏流的 $\diamond^{\mathrm{C}}$-度要么是原子，要么是 $\mathcal{T}_1$。

证明。 设 $\zeta \in 2^{\omega}$ 是稀疏的。若 $\zeta \in \mathcal{T}_1$，则结论成立。因此，设 $\zeta \notin \mathcal{T}_1$，也就是说，它具有如下形式

$$\zeta = \prod_{i=0}^{\infty} 1 \cdot 0^{c(i)},$$

其中 $c : \mathbb{N} \to \mathbb{N}$ 单调无界。现在，假设 $\zeta \triangleright^{\mathrm{C}} \sigma$ 通过 $M = (\Sigma, N, \delta)$。不失一般性，我们可以假设 $\delta(0, \ldots\ldots, 0) = 0$。最终，每个邻域至多包含一个 1，因此我们可以专注于这些情形，并通过推论 4.9 处理有限差异。定义

$$w := \prod_{i=0}^{2N} \delta(0^{2N-i} \cdot 1 \cdot 0^i);$$

那么，鉴于 $\zeta \triangleright^{\mathrm{C}} \sigma$ 和 ζ 是稀疏的，我们有 σ 最终会呈现“w 后面跟着越来越多的零，并不断重复”的形式。更正式地说，以下

流

$$\rho := \prod_{i=0}^{\infty} w \cdot 0^{c(i)-|w|},$$

其中我们令 $0^n = \varepsilon$ 对应于 $n < 0$，它具有如下性质：从某一点 $j \in \mathbb{N}$ 起，$\sigma(j) = \rho(j)$，从而由推论 4.9 可得 $\sigma \diamond^{\mathrm{C}} \rho$。有两种情形：

1. $w = 0^{2N+1}$，此时我们得到 $\rho = 0^{\omega}$，这意味着 $\sigma \in \mathcal{T}_1$，证明即告完成。

2. $w \neq 0^{2N+1}$，此时我们可借助该规则从 ρ 中还原 ζ（除有限多个字母外）

$$\delta(x) = \begin{cases} 1 & \text{if } x = w, \\ 0 & \text{else.} \end{cases}$$

由此可得 $\sigma \diamond^{\mathrm{C}} \zeta$。

我们已经证明 $\forall \sigma \in 2^{\omega}$、$\zeta \triangleright^{\mathrm{C}} \sigma \Rightarrow \sigma \in \mathcal{T}_1$ 或 $\sigma \diamond^{\mathrm{C}} \zeta$，这意味着根据定义 5.18，$[\zeta]_{\diamond^{\mathrm{C}}}$ 是一个原子。□

推论 5.25. 存在 $2^{\aleph_0}$ 多个原子 $\diamond^{\mathrm{C}}$-度。

证明。 存在 $2^{\aleph_0}$ 多个稀疏流，这意味着存在 $2^{\aleph_0}$ 个互不相同的稀疏流的度，因为由引理 5.4，每个度都是可数的，而定理 5.24 表明它们都是原子。□

自然的下一步是确定稀疏流所归约到的流。为此，我们引入流同余的概念。

定义 5.26. 流 σ、$\tau \in 2^{\omega}$ 是同余的[3]，当存在 n、$m \in \mathbb{N}$ 使得 $\forall i \in \mathbb{N}$、$\sigma(n+i) = \tau(m+i)$。我们将同余记为 $\sigma \cong \tau$。

显然，$\sigma \cong \tau$ 蕴含 $\sigma \diamond^{\mathrm{C}} \tau$（推论 4.9）。

命题 5.27. 对于每个稀疏 σ，$\tau \in \Sigma^{\omega}$、$\sigma \diamond^{\mathrm{C}} \tau$ 当且仅当 $\sigma \cong \tau$。

证明。 只需证明 (⇒) 方向。令 $\sigma \triangleright^{\mathrm{C}} \tau$ 经过一维元胞自动机 $M = (\Sigma, N, \delta)$。那么，以下两点成立

– $\delta(0, \ldots, 0) = 0$。

– 在所有恰好包含一个 1 的邻域中，必须恰有一个邻域输出 1，而其余邻域均输出 0。

[3] 请注意，对于任意字母表的情形，为使证明能够沿用，这一同余概念应扩展到字母重命名（双射 $\phi\colon \Sigma \to \Sigma$）的情况。

因此，在一段初始部分中，$\tau(i)$ 的值由包含 $\perp$ 个或多个 1 的邻域上的 δ 决定；之后，我们会在 τ 中发现 1 的副本，它们与 σ 中对应的 1 之间的距离固定。由此，我们得到一一映射ϕ：$\Sigma \to \Sigma$，其中 $\phi(p_1) = p_2$，且 $\phi(x) = y$。此外，根据给定的 $\tau \triangleright^{\mathrm{C}} \sigma$，通过对称推理可得到 ϕ^{-1}，因此 ϕ 是一个双射，并且 $\sigma \cong \tau$。□

本文所定义的稀疏性，与文献 [1] 中引入的较弱稀疏性概念相比，是一种相当严格的性质。例如，在其他语境下，诸如 $\prod_{i\in\mathbb{N}} 11 \cdot 0^i$ 这样的流会被视为稀疏流，但按照本文目前研究的意义，它并不稀疏。受此启发，下面我们引入并研究一种较弱的概念。

定义 5.28。流 $\sigma \in \Sigma^\omega$ 是弱稀疏的，当且仅当存在一个填充字母 $p \in \Sigma$，使得

$$\lim_{n\to\infty} \frac{|\{i \le n \mid \sigma(i) = p\}|}{n} = 1.$$

例 *5.29*。对于任意有限字 $w \in 2^*$，流 $\prod_{i\in\mathbb{N}} w \cdot 0^i$ 是弱稀疏的，但除非 w 至多包含一个 1，否则它并不稀疏。

例 *5.30*。素数流是 $\pi \in 2^\omega$，其中 $\pi(i) = 1$ 当且仅当 i 是素数。根据素数定理，流 π 是弱稀疏的；而张益唐定理表明，对于某个常数 $k \in \mathbb{N}$ [44]，存在无穷多个大小为 k 的素数间隙，这意味着 π 并不稀疏。

我们再次将范围限制为 $\Sigma = 2 = \{0, 1\}$，并假设 0 是填充字母。接下来，我们证明稀疏比弱稀疏要求更严格：每个稀疏流都是弱稀疏的，但反过来则不成立，正如例 5.29 和 5.30 所示。

引理 5.31。$\forall \sigma \in \Sigma^\omega$，如果 σ 是稀疏的，那么 σ 也是弱稀疏的。

证明。设流 σ 是稀疏的。将弱稀疏定义中的极限展开后，我们需要证明的性质可改写如下：

$$\forall \gamma \in \mathbb{R}^{>0}, \exists n_0 \in \mathbb{N}, \forall n \ge n_0,\ \frac{|\{i \le n \mid \sigma(i) = p\}|}{n} \ge 1 - \gamma.$$

这种改写与稀疏性的定义明显重叠。为给出明确的证明，令 $\gamma > 0$，并取任意 $d \in \mathbb{N}$。于是，根据稀疏性，存在 $N \in \mathbb{N}$，使得对所有 $i \ge N$，若 $\sigma(i) \neq p$，则 $\sigma(i+1), \ldots, \sigma(i+d)$ 都等于 p。因此，在 N 之后任意连续的 $d+1$ 个位置中，至多有一个不是 p。

因此，对所有 $n \ge N$，

$$|\{N < i \le n \mid \sigma(i) \neq p\}| \le \frac{n - N}{d + 1},$$

因此 $|\{i \le n \mid \sigma(i) \neq p\}| \le \frac{n-N}{d+1} + N$。两边除以 n 得到

$$\frac{|\{i \le n \mid \sigma(i) \neq p\}|}{n} \le \frac{1}{d+1} + \frac{N}{n}.$$

取补集,

$$\frac{|\{i \le n \mid \sigma(i) = p\}|}{n} \ge 1 - \frac{1}{d+1} - \frac{N}{n}.$$

为完成证明，选取 d 使得 $\frac{1}{d+1} \le \frac{\gamma}{2}$，并选取 $n_0 \ge N$ 使得对所有 $n \ge n_0$均有 $\frac{N}{n} \le \frac{\gamma}{2}$。□

引理 5.32. 若 σ 是稀疏的，则每个 $\tau \in [\sigma]_{\diamond^{\mathrm{C}}}$ 都是弱稀疏的。

证明。 回顾定理 5.24 中的情形 1 与情形 2。在情形 1 中，我们有 $\sigma,\ \tau \in \mathcal{T}_1$，因此它是弱稀疏的。在情形 2 中，除去有限多个字母后，τ 最终具有如下形式

$$\prod_{i=0}^{\infty} w \cdot 0^{c(i)-|w|},$$

其中当 $n < 0$时为 $0^n = \varepsilon$。回顾 $c\colon\ \mathbb{N} \to \mathbb{N}$ 是单调无界的，这意味着 τ 是弱稀疏的。□

给定一个流 $\sigma \in \Sigma^{\omega}$，我们可以通过将其与越来越多的填充字母交错，构造一个包含来自 σ 的信息的弱稀疏流。以下结果利用这一原理，将周期流的良基结构编码到一族弱稀疏流中。

命题 5.33. 定义如下弱稀疏流：

$$\varsigma_i = \prod_{j=0}^{\infty} (1 \cdot 0^{i-1})^j \cdot 0^{2^j},$$

于是，对任意 n、$m \in \mathbb{N}$，我们有 $\varsigma_m \triangleright^{\mathrm{C}} \varsigma_n$ 当且仅当 $n \mid m$。

证明. 显然 ς_i 是弱稀疏的，因为对常数 i 有 $O(2^j) \gg O(j \cdot i)$。因此我们着重证明该双向蕴含。

（⇒）假设借助机器 $M = (\Sigma, N, \delta)$ 有 $\varsigma_m \triangleright^{\mathrm{C}} \varsigma_n$。由 ς_m 与 ς_n 的构造，$(1 \cdot 0^{m-1})^{\omega} \in \mathcal{T}_m$ 的任意大子串（甚至大于 M 的局部邻域）都会出现在 ς_m 中；由于 $f_M(\varsigma_m) = \varsigma_n$，这些子串被 M 映射为 ς_n 中 $(1 \cdot 0^{n-1}) \in \mathcal{T}_n$ 的子串。因此我们有 $f_m(\tau_m) \diamond^{\mathrm{C}} \tau_n$，从而由定理 5.16 可得 $n \mid m$。

（⇐）假设 $n \mid m$，即 $\exists k \in \mathbb{N},\ m = n \cdot k$。那么，我们构造一个一维元胞自动机 $M = (\Sigma,\ N,\ \delta)$，通过将输入流中的每个 $1 \cdot 0^m$ 映射为输出中的 $(1 \cdot 0^n)^k$，来归约 $\varsigma_m \triangleright^{\mathrm{C}} \varsigma_n$。□

5.3 极大度与上确界

至此，我们已经证明了存在一个最小（底部）度。接下来自然会问，极大度是否存在。这个问题向来很难：对于有限状态转换器 [13]，，它仍未解决；在我们的设定中也同样悬而未决。不过，我们可以在这个方向上得到一些部分结果，首先给出以下定理：

定理 5.34。对于任意 $\sigma \in \Sigma^\omega$，如果存在一个有限字 $v \in \Sigma^*$，它在 σ 中只出现有限次，那么 σ 就不是极大的。

证明。假设 $v \in \Sigma^*$ 在 σ 中只出现有限次。构造一个包含 v 的字 $\hat{}v$，使它与自身的移位不发生（非平凡）重叠。例如，可以取 $\hat{}v := 1 \cdot v \cdot 1 \cdot 0^{|v|+1}$。由于 v 在 σ 中只出现有限次，$\hat{}v$ 也只出现有限次。根据鸽巢原理，存在一个长度为 $\lceil v \rceil$ 的字 $w \in \Sigma^*$，它在 σ 中出现无限次。考虑通过将 w 的这些无限次出现中任意选取的若干处替换为 $\hat{}v$，来修改 σ 的过程。每次这样的修改都会产生一个新的无限流 τ。显然有以下两点：1. 将所有 $\hat{}v$ 替换为 w（因为 $\hat{}v$ 不会与自身的移位重叠），再通过有限变异补偿 σ 中原本已有的有限个 $\hat{}v$，即可将每个这样的 τ 约简回 σ。2. 选择替换哪些 w 副本的方式有 $2^{\aleph_0}$ 种。因此，σ 不可能约简到所有这些流，这意味着存在严格高于 σ 的流。因此，σ 不是极大的。□

根据定理 5.34，极大流必须包含每个有限子词无穷多次，也就是说，它必须具有极大子词复杂度 [11]。令 $\mathcal{S} \subseteq \Sigma^\omega$ 为这类流的集合。现在我们证明，$\mathcal{S}$ 中并非所有流对于 1CA 可约性都是极大的。

引理 5.35。$\mathcal{S}$ 中存在非极大流。

证明。取 $\sigma \in \mathcal{S}$，并令 M 为例 4.3 中的异或一维元胞自动机。令 $f_M(\sigma) = \tau$，我们断言 $\tau \in \mathcal{S}$，但 $f_M(\sigma)$ 并非极大。

– 由于满射性，$f_M(\sigma)$ 中每个有限字都会出现无穷多次：例如，若要判断 011 是否在 $f_M(\sigma)$ 中出现无穷多次，答案是肯定的，因为它的原像 1101 在 σ 中出现无穷多次。因此，$f_M(\sigma) \in \mathcal{S}$。

– 另一方面，由于异或自动机的非单射性，$f_M(\sigma)$ 无法约化回 σ。为证明这一点，反设 $f_M(\sigma) \triangleright^{\mathrm{C}} \sigma$ 可通过 $M' = (2, N, \delta)$ 得到。对于任意字 $w \in 2^{2N+1}$，由于 $f_M(\sigma) \in \mathcal{S}$，$f_M(\sigma)$ 中必须同时出现字 $0 \cdot w$ 和 $1 \cdot w$。具体而言，在 $f_M(\sigma)$ 中存在索引 i 和 j，使得

$$\mathcal{V}_{f_M(\sigma),N}(i) = w = \mathcal{V}_{f_M(\sigma),N}(j),$$

此外，$\mathcal{V}_{f_M(\sigma)\,N}(i)$ 左侧的字母为 0（即 $f_M(\sigma)(i-N-1) = 0$），而 $\mathcal{V}_{f_M(\sigma)\,N}(j)$ 左侧的字母为 1（即 $f_M(\sigma)(j-N-1) = 1$）。

因此，由于一种情况下 w 左侧为 0，而另一种情况下为 1， M' 输出的奇偶性发生了翻转（因为它撤销了异或运算）。因此，

$$\delta(\mathcal{V}_{f_M(\sigma),N}(i)) \neq \delta(\mathcal{V}_{f_M(\sigma),N}(j)),$$

矛盾。 □

Mealy机与有限状态转换器的层次构成半格，其中 zip 算子（见例 3.6）给出两个流的最小上界 [3,13]。对于 1CA 归约，在限制为最终周期流或稀疏流时可以构造上确界。然而，如后文引理 5.38 所示，流对的上界一般并不存在。

推论 5.36。 $\mathcal{T}_{\mathrm{lcm}(n,m)}$ 是 $\{\mathcal{T}_n, \mathcal{T}_m\}$的上确界。

证明。 $\mathcal{T}_{\mathrm{lcm}(n,m)}$ 是 $\mathcal{T}_n$的上界, $\mathcal{T}_m$，由定理 5.16 可知。为证明它是上确界，设 $X \in \mathfrak{D}$ 使得 $X >^{\mathrm{C}} \mathcal{T}_n, \mathcal{T}_m$ 且 $\mathcal{T}_{\mathrm{lcm}(n,m)} >^{\mathrm{C}} X$。于是我们得到 $X \in \mathcal{T}$，因为由推论 5.12， $\mathcal{T}$在 $\triangleright^{\mathrm{C}}$ 下封闭，从而由定理 5.16 得到 $X \diamond^{\mathrm{C}} \mathcal{T}_{\mathrm{lcm}(n,m)}$。 □

构造稀疏流的上确界要稍微复杂一些，但仍然是可行的。

引理 5.37. 设 σ、 τ为稀疏的。则 $\{\sigma, \tau\}$ 具有一个上确界。

证明。 给定 2^ω中的稀疏 σ 与 τ ，我们将把 σ 和 τ 编码进一个新的流 ξ，以构成一个上确界。这可以通过谨慎避免那些若在左右两侧用0填充便可能被混淆的模式来实现。以下是这样一种编码的一个例子:

- If $\sigma(i) = 1$ and $\tau(i) = 1$, let $\xi(i, i+1, i+2, i+3) = 1000$,
- if $\sigma(i) = 1$ and $\tau(i+1) = 1$, let $\xi(i, i+1, i+2, i+3) = 1100$,
- if $\sigma(i) = 1$ and $\tau(i+2) = 1$, let $\xi(i, i+1, i+2, i+3) = 1010$,
- if $\sigma(i) = 1$ and $\tau(i+3) = 1$, let $\xi(i, i+1, i+2, i+3) = 1110$,
- if $\tau(i) = 1$ and $\sigma(i+1) = 1$, let $\xi(i, i+1, i+2, i+3) = 1001$,
- if $\tau(i) = 1$ and $\sigma(i+2) = 1$, let $\xi(i, i+1, i+2, i+3) = 1101$,
- if $\tau(i) = 1$ and $\sigma(i+3) = 1$, let $\xi(i, i+1, i+2, i+3) = 1011$,
- for any yet undefined index i, let $\xi(i) = 0$.

从某一点 n 起， （ σ 和 τ中的）每一个1后面都至少跟着七个0，这确保上述模式不会重叠。点 n 之前出现的任何内容都可以忽略，因为它可以通过有限变异来处理。因此， $\xi \triangleright^{\mathrm{C}} \sigma, \tau$。为了证明这个上界同时也是上确界，考虑 σ和 τ的任意上界 ξ'，以及相应的机器 $M = (2, N, \delta)$和

$M' = (2, N', \delta')$。然后，在较高层面上，可以直截了当地得到从 ξ' 到 ξ 的归约，因为在上述方案中，$\xi(i)$ 是相对于 $\sigma(i)$ 和 $\tau(i)$ 定义的，而这些分别由 $\delta(\mathcal{V}_{\xi',N}(i))$ 和 $\delta'(\mathcal{V}_{\xi',N'}(i))$ 所标识。□

注意，按引理 5.37 构造的 ξ 并不稀疏（而是弱稀疏），因此无法通过归纳应用上述方案来求稀疏流有限集的上确界。不过，对于任意有限个稀疏流，类似的论证可以给出合适的编码，尽管这种编码要复杂得多。

推论 5.36 和引理 5.37 中上确界的构造之所以可行，非正式地说，是因为其中包含的信息不算太多，因此可以将两个流压缩成一个。相比之下，我们接下来要构造一些包含过多信息、因而不存在上界的流；事实上，它们属于 $\mathcal{S}$。

引理 5.38. 存在 σ、$\tau \in \Sigma^\omega$，它们没有上界。

证明。构造 σ 和 τ，使得对于每个 k 以及每一对 w_1、$w_2 \in \Sigma^k$，块 w_1 都出现在 σ 中，并与 w_2 在 τ 中对齐。反设 ξ 可通过机器 $M_1 =$（Σ、N_1、δ_1）和 $M_2 =$（Σ、N_2、δ_2）分别约化到 σ 和 τ。令 $N =$ 为 $\max(N_1, N_2)$，并为方便起见，将 δ_i 扩展为：$(\Sigma \cup \{\perp\})^{2N+1} \to \Sigma$，其中包含 $i = 1$、2。注意，σ（或 τ）中长度为 l 的块只取决于 ξ 中长度为 $2N + l$ 的块。此外，这样一个 ξ 的块需要同时编码 σ 和 τ 中长度为 l 的块，而根据构造，$l > 2N$ 无法做到这一点。更形式化地说，我们定义如下形式的映射 g、h：$\Sigma^{2N+l} \to \Sigma^l$：

$$g(x_1, \ldots, x_{2N+l}) := \prod_{j=1}^{l} \delta_1(x_j, \ldots, x_{2N+j}),$$
$$h(x_1, \ldots, x_{2N+l}) := \prod_{j=1}^{l} \delta_2(x_j, \ldots, x_{2N+j}).$$

那么，令 $r(x) := g(x) \cdot h(x)$，可得

$$r(\xi(i-N), \ldots, \xi(i+N+l-1)) = \sigma(i : i+l-1) \cdot \tau(i : i+l-1),$$

鉴于 $\xi \triangleright^{\mathrm{C}} \sigma, \tau$ 通过 δ_1 和 δ_2。根据 σ 和 τ 的构造，函数 r：$\Sigma^{2N+l} \to \Sigma^{l+l}$ 必为满射，因为它们包含所有长度为 l 的子词对。然而，当 $l > 2N$ 时，这是不可能的；又一个通过对角化得出的证明。□

6 确定可约性的一种算法

我们可以用一句简洁而贴切的口号来重新表述 1CA 可约性：

$$\boxed{\sigma \triangleright^{\mathrm{C}} \tau \text{ iff } \exists N \in \mathbb{N}, \forall i, i' \in \mathbb{N}, \mathcal{V}_{\sigma,N}(i) = \mathcal{V}_{\sigma,N}(i') \Rightarrow \tau(i) = \tau(i').}$$

算法1 用于近似 1CA 可约性的算法。

输入： $\sigma,\ \tau \in \Sigma^{\omega},\ c_{\max} \in \mathbb{N},\ \alpha \in \mathbb{Q}^{>0}$
输出： 对 $\sigma \triangleright^{\mathrm{C}} \tau$ 是否成立的估计
初始化查询计数器 $c = 0$
对于 $N = 0$ 到 ∞ **执行**
  初始化缓存 $V_N = \{\}$
  对于 $i = N$ 至 ∞ **执行**
    $c \leftarrow c + 1$
    如果 $c > c_{\max}$ **then**
      如果 $i \geq \alpha|\Sigma|^{2N+1}$ **则**
        返回 “是”
      else
        返回“否”
      结束如果
    结束如果
    如果 $\mathcal{V}_{\sigma,N}(i) \notin V_N$ **则**
      $V_N \leftarrow V_N \cup \{\mathcal{V}_{\sigma,N}(i) \mapsto \tau(i)\}$
    else
      如果 $V_N(\mathcal{V}_{\sigma,N}(i)) \neq \tau(i)$ **则**
        跳出内层 for 循环。
      结束 if
    结束 if
  结束 for
结束 for

这句口号启发了一种蛮力算法：遍历所有邻域半径并检查是否存在不匹配，从而判定 1CA 可约性。我们提出算法1，其输入为两个流 $\sigma,\ \tau \in \Sigma^{\omega}$，并将它们视为预言机，此外还包括最大查询预算 $c_{\max} \in \mathbb{N}$，用于限制可访问 σ 和 τ 中符号的数量。它还使用一个可调超参数 $\alpha \in \mathbb{Q}^{>0}$，用于设定返回“是”之前所需的置信度。

该算法通过遍历递增的半径值 $N = 0$，1，2，.....来执行。对于每个 N，它都会检查流中的位置 $i \geq N$ 。在每个索引 i处，它从 σ中提取邻域 $\mathcal{V}_{\sigma,\ N}(i)$。如果此前未见过该邻域，就将其连同相应的值 $\tau(i)$一起添加到缓存 V_N 中。如果此前见过该邻域，算法会验证 $\tau(i)$是否与先前存储的值一致；如果不一致，则表明当前半径 N 不足，因此算法会跳出内层循环并递增 N。算法会持续跟踪查询用量；一旦达到查询上限 $c_{\max}$ ，便作出判断。如果针对当前 N已检查至少 $\alpha|\Sigma|^{2N+1}$ 个不同索引，则返回“Yes”；否则返回“No”。

事实上，算法1无法在有限时间内判定一个流是否约化为另一个流，因为需要检查的符号有无穷多个。实际上，如果我们将流编码为图灵机，那么该判定问题不可判定也就不足为奇了（见附录 A）。然而，它具有“渐近正确”的行为： σ 约化

为 τ，当且仅当该算法最终总是输出“Yes”。我们在定理 6.1 中将其形式化，其中 $P(\sigma,\ \tau,\ c_{\max},\ \alpha) \in \{\text{“Yes”、“No”}\}$ 表示使用参数 $\sigma,\ \tau,\ c_{\max},\ \alpha$运行该算法。

定理 6.1. 对于 σ、$\tau \in \Sigma^{\omega}$ 和 $\alpha > 0,\ \sigma \triangleright^{\mathrm{C}} \tau$ 当且仅当 $\exists c_0 \in \mathbb{N}$，其中 $\forall c_{max} \geq c_0$、$P(\sigma、\ \tau、\ c_{max}、\ \alpha) =$“是”。

证明。（$\Rightarrow$）设 σ、$\tau \in \Sigma^{\omega}$，并设 N为使得 $\sigma \triangleright^{\mathrm{C}} \tau$成立的最小半径，且该结果由半径为 N 的一维元胞自动机给出。于是，$\forall N' < N$、$\exists i_{N'}$、$i'_{N'}$，其中 $\mathcal{V}_{\sigma、\ N'}(i_{N'}) = \mathcal{V}_{\sigma、\ N'}(i'_{N'})$ 且$\tau(i_{N'}) \neq \tau(i'_{N'})$。通过令 $c_0 = \sum_{N'<N} \max(i_{N'}, i'_{N'})$，我们有：若传入的 $c_{\max}$ 值大于 c_0，则算法1将排除所有小于 N 的半径。因此，对于任意 $\alpha \in \mathbb{R}^{>0}$，我们可以令 $c'_0 = c_0 + \alpha\lceil\Sigma^{2N+1}\rceil$，这意味着对于任意$c_{\max} > c'_0$，我们得到 $P(\sigma, \tau, c_{\max}, \alpha) =$“是”。

($\Leftarrow$) 令 $\sigma, \tau \in \Sigma^{\omega}, \alpha > 0$，并假设 $\exists c_0 \in \mathbb{N}$，其中 $\forall c_{\max} \geq c_0$,$P(\sigma, \tau, c_{\max}, \alpha) =$“是”。那么，对于某个 N，不存在 $i, i' \in \mathbb{N}$，其中 $\mathcal{V}_{\sigma,N}(i) = \mathcal{V}_{\sigma,N}(i')$ 且 $\tau(i) \neq \tau(i')$；否则，算法最终会找到这一对并输出“否”。因此，$\sigma \triangleright^{\mathrm{C}} \tau$。□

在下面的推论中，我们通过逆否命题来论证否定情形。

推论 6.2。对于任意 $\sigma, \tau \in \Sigma^{\omega}$ 和 $\alpha > 0$，当且仅当存在无限多个 $c_{max} \in \mathbb{N}$，其中 $P(\sigma, \tau, c_{max}, \alpha) =$“否”。□

然而，推论 6.2 并不意味着：如果无法进行归约，那么算法最终总会输出“No”。当然，通过对角化，可以构造两个流，使算法无限多次（但并非总是）输出“Yes”，尽管正确答案应为“No”。这是由于前文提到的难题：必须在有限时间内检查无限多个符号。我们将在引理 6.3 中构造这样的流。

引理 6.3。$\exists \sigma, \tau \in \Sigma^{\omega}, \alpha > 0$，使得 $\sigma \not\triangleright^{\mathrm{C}} \tau$ 尽管存在无限多个 $c_{max} \in \mathbb{N}$ 使得 $P(\sigma, \tau, c_{max}, \alpha) =$“*Yes*”。

证明。设 $\sigma = 0^{\omega}$ 和 $\tau := \prod_{i\in\mathbb{N}} 0^{\lceil \alpha|\Sigma|^{2i+1}\rceil} \cdot 1$。那么，根据推论 5.12，$\sigma \not\triangleright^{\mathrm{C}} \tau$，但存在无限多个 $c_{\max} \geq c_0$ 使得 $P(\sigma,\ \tau,\ c_{\max},\ \alpha) =$“Yes”。□

在引理 6.3 中，流 τ 看起来像 σ，持续指数级长的时间（约 $\alpha|\Sigma|^{2i+1}$ 个符号），之后为了满足 $\sigma \not\triangleright^{\mathrm{C}} \tau$，它看起来会有所不同。一般而言，任何使该引理成立的反例也必须包含这样的指数级长片段，在这些片段中，两个流看起来相似。因此，对于不含这些指数级片段的任意一对流，该算法最终总能给出正确的输出；对于许多（如果不是大多数）具有普遍意义的流而言，情况正是如此。例如，最终周期流最终总能被正确分类。

7 可约性的替代定义

本节考察对前文引入的 1CA 可约性概念的扩展。我们为混合元胞自动机和有限字元胞自动机给出初步结果，同时也简要介绍一些替代方案。这些系统严格来说可能不是元胞自动机，但仍具有元胞自动机的特征。

7.1 混合元胞自动机

密码学中的一些方法会使用混合元胞自动机，将多个局部规则交替组合，以平衡密码学鲁棒性的各种指标 [8]。在接下来的定义中，我们将引入混合元胞自动机可约性及其度，并证明一些初步性质。

定义 7.1。 一个混合 1CA 是一个元组 $M = (\Sigma, N, \{\delta_1, \ldots, \delta_K\})$，其定义方式与标准的一维元胞自动机相同，但全局规则 $f_M(\sigma)(i) = \delta_k(\mathcal{V}_{\sigma,N}(i))$ 满足 $k \equiv i$(模 K)。

定义 7.2。 一个混合 1CA 归约 是关系 $\triangleright_h^{\mathrm{C}} \subseteq \Sigma^\omega \times \Sigma^\omega$，其中 $\sigma \triangleright_h^{\mathrm{C}} \tau$当且仅当存在一个混合 *1CA* M使得 $f_M(\sigma) = \tau$。此外，我们令$\sigma \triangleleft_h^{\mathrm{C}} \tau \Leftrightarrow \tau \triangleright_h^{\mathrm{C}} \sigma$，且 $\diamond_h^{\mathrm{C}} = \triangleright_h^{\mathrm{C}} \cap \triangleleft_h^{\mathrm{C}}$。

定义 7.3. $\diamond_h^{\mathrm{C}}$-度 是商集 $\mathfrak{D}_h = \Sigma^\omega / \diamond^{\mathrm{C}}$中的一个元素，我们将 $\mathfrak{D}_h$ 称为 $\diamond_h^{\mathrm{C}}$-度的集合。关系 $\leq_h^{\mathrm{C}}$ 对所有X、$Y \in \mathfrak{D}_h$ 定义如下： $Y \leq_h^{\mathrm{C}} X \Leftrightarrow \forall \sigma \in X$、$\forall \tau \in Y$、 $\sigma \triangleright_h^{\mathrm{C}} \tau$。

命题 7.4. 关系 $\triangleright_h^{\mathrm{C}}$ 与 *FST*-归约和 *1CA*-归约的比较如下：

1. $\triangleright^{\mathrm{C}} \subsetneq \triangleright_h^{\mathrm{C}}$,
2. $\triangleright_h^{\mathrm{C}} \subsetneq \triangleright^{\mathrm{F}}$。

这意味着 $\triangleright_h^{\mathrm{C}}$ 在有限变异下封闭。

证明。
1. 由任意 1CA $M = (\sigma, N, \delta)$ 都有一个等价的 h1CA$M' = (\sigma, N, \{\delta\})$，可知 $\triangleright^{\mathrm{C}} \subseteq \triangleright_h^{\mathrm{C}}$ 成立；而 $\triangleright_h^{\mathrm{C}} \not\subseteq \triangleright^{\mathrm{C}}$，因为 $0^\omega \not\triangleright^{\mathrm{C}} (01)^\omega$，但 $0^\omega \triangleright_h^{\mathrm{C}} (01)^\omega$通过 $M'' = (\Sigma, 0, \{\delta_1, \delta_2\})$，其中 $\delta_1(_) = 0$ 且 $\delta_2(_) = 1$。2. 这可以像命题 4.12 中那样用 FST 实现，区别在于它必须在每一步模拟不同的 δ_k ；只要令 $Q = (\Sigma \cup \{\bot\})^{2N+1} \times \{1, \ldots, K\}$即可做到。□

命题 7.5. $(\Sigma^\omega, \triangleright_h^{\mathrm{C}})$ 是一个预序。

证明。自反性显然成立。任取 σ，τ，$\rho \in \Sigma^\omega$，并假设 $\sigma \triangleright_h^{\mathrm{C}} \tau$ 和 $\tau \triangleright_h^{\mathrm{C}} \rho$ 通过 $M_i = (\Sigma,\ N_i,\ \{\delta_1^{(i)}, \ldots, \delta_{K_i}^{(i)}\})$ 成立，其中 $i = 1,\ 2$。那么，存在

$$M_3 = (\Sigma, N_1 + N_2, \{\delta_1^{(3)}, \ldots, \delta_{K_1 \cdot K_2}^{(3)}\})$$

使得 $\sigma \triangleright_h^{\mathrm{C}} \rho$，它可以用类似命题 4.6 的方法构造。采用简写记号 $\delta_i^{(3)}{}_{,\ j} := \delta^{(3)}{}_{(i-1)\cdot K_2 + j}$，令

$$\delta_{i,j}^{(3)}(x_0, \ldots, x_{2(N_1+N_2)}) := \delta_j^{(2)}(\delta_i^{(1)}(v_0), \ldots, \delta(v_{2N_2})),$$

其中 $v_k := (x_k, \ldots, x_{k+2N_1})$。 □

引理 7.6. *$\mathcal{T}$ 对于 Σ^ω 是 $\triangleright_h^{\mathrm{C}}$-极小的，即 $\forall \sigma \in \Sigma^\omega, \forall \tau \in \mathcal{T}, \sigma \triangleright_h^{\mathrm{C}} \tau$。*

证明。设 $\sigma \in \Sigma^\omega$ 和 $\tau = x \cdot y^\omega \in \mathcal{T}$，其中 $x,\ y \in \Sigma^*$。那么，$\sigma \triangleright_h^{\mathrm{C}} y^\omega$ 与 $M = (\Sigma,\ N,\ \{\delta_1, \ldots, \delta_K\})$，其中 $N = 0$、$K = |y|$，且 $\delta_i(_) = y(i)$。由于该关系在有限变异下封闭，我们得到 $y^\omega \triangleright_h^{\mathrm{C}} \tau$；由传递性可得 $\sigma \triangleright_h^{\mathrm{C}} \tau$。□

由于 $\triangleright_h^{\mathrm{C}} \subseteq \triangleright^{\mathrm{F}}$ 根据命题 7.4，$\mathcal{T}$ 必须在 $\diamond_h^{\mathrm{C}}$ 下封闭，因此也必须是一个度。因此，根据引理 7.6，$\mathcal{T}$ 是最小的 $\triangleright_h^{\mathrm{C}}$-度。

推论 7.7。 $\mathcal{T}$ 是最小的 $\diamond_h^{\mathrm{C}}$-度。 □

总之，混合 1CA 所产生的流层级具有有趣的代数性质，而且它们在密码学等实际应用中的存在，为进一步探索这一概念提供了更多动力。

7.2 有限字输出的元胞自动机

我们考虑一种一维元胞自动机，其中元胞可以输出多个字母，这可以理解为将每个元胞拆分成多个元胞。

定义 7.8。 有限字 *1CA* 是一个三元组 (Σ, N, δ)，其定义方式与标准一维元胞自动机相同，不同之处在于 $\delta : (\Sigma \cup \{\perp\})^{2N+1} \to \Sigma^*$，且全局规则 $f_M : \Sigma^\omega \rightharpoonup \Sigma^\omega$ 是 $f_M(\sigma) = \prod_{i \in \mathbb{N}} \delta(\mathcal{V}_{\sigma,N}(i))$。

定义 7.9。 有限字一维元胞自动机归约 是一个关系 $\triangleright_*^{\mathrm{C}} \subseteq \Sigma^\omega \times \Sigma^\omega$，其中 $\sigma \triangleright_*^{\mathrm{C}} \tau$ 当且仅当存在一个有限字 *1CA* M，使得 $f_M(\sigma) = \tau$。此外，我们令 $\sigma \triangleleft_*^{\mathrm{C}} \tau \Leftrightarrow \tau \triangleright_*^{\mathrm{C}} \sigma$，并令 $\diamond_*^{\mathrm{C}} = \triangleright_*^{\mathrm{C}} \cap \triangleleft_*^{\mathrm{C}}$。

命题 7.10。 $\triangleright_*^{\mathrm{C}}$ 与其他概念的比较如下：

1. $\triangleright^{\mathrm{C}} \subsetneq \triangleright_*^{\mathrm{C}}$，
2. $\triangleright_*^{\mathrm{C}} \subsetneq \triangleright^{\mathrm{F}}$，

3. $\triangleright_h^{\mathrm{C}} \not\subseteq \triangleright_*^{\mathrm{C}}$, *and* $\triangleright_*^{\mathrm{C}} \not\subseteq \triangleright_h^{\mathrm{C}}$.

证明。1. 和 2. 的证明与命题 7.4 中的证明类似。

为证明 $\triangleright_*^{\mathrm{C}} \not\subseteq \triangleright_h^{\mathrm{C}}$，任取 $\xi \in \Sigma^\omega$，并令 $\xi \triangleright_*^{\mathrm{C}} f_M(\xi)$，其中有限字 1CA$M = (\Sigma, 0, \delta)$，且 $\delta(x) = xx$。一般而言，混合 1CA 无法执行这种约简，例如当 ξ 不可计算时。因此，我们得出结论 $\triangleright_h^{\mathrm{C}} \subsetneq \triangleright_*^{\mathrm{C}}$。为证明$\triangleright_h^{\mathrm{C}} \not\subseteq \triangleright_*^{\mathrm{C}}$，定义

$$\sigma := \prod_{i=0}^{\infty} 0^{2^i} \cdot 1 \quad \text{and} \quad \tau := \prod_{i=0}^{\infty} 0^{3 \cdot 2^{2i}+1} \cdot 1.$$

我们通过 h1CA $\delta_1(x) = x$ 和 $\delta_2(_) = 0$，将 σ 中每个偶数索引处的字母 1 替换为 0，从而得到 $\sigma \triangleright_h^{\mathrm{C}} \tau$，如 $3 \cdot 2^{2i} + 1 = 2^{2i} + 1 + 2^{2i+1}$所示。相比之下，$\sigma \not\triangleright_*^{\mathrm{C}} \tau$，因为对于任意 N，σ 中形如 $0^N \cdot 1 \cdot 0^N$的局部邻域必须映射到 τ中的 0 和 1。□

作为推论 4.9 和命题 7.10 的推论，$\triangleright_*^{\mathrm{C}}$ 在有限变异下封闭。

引理 7.11. $\mathcal{T}$ 对于 Σ^ω是 $\triangleright_*^{\mathrm{C}}$-极小的，即

$$\forall \sigma \in \Sigma^\omega, \forall \tau \in \mathcal{T}, \sigma \triangleright_*^{\mathrm{C}} \tau.$$

证明. 设 $\sigma \in \Sigma^\omega$ 和 $\tau = x \cdot y^\omega \in \mathcal{T}$满足 $x, y \in \Sigma^\omega$。那么，通过规则为下式的一维元胞自动机，我们得到 $\sigma \triangleright_*^{\mathrm{C}} \tau$

$$\delta(c, _, _) = \begin{cases} x & \text{if} \quad c = \bot \\ y & \text{else.} \end{cases}$$ □

然而，正如我们接下来所示，这种归约不具有传递性：

命题 7.12. 关系 $\triangleright_*^{\mathrm{C}}$ 在 Σ^ω上不具有传递性。

证明. 固定 $\Sigma = 3 = \{0, 1, 2\}$，并令 $\xi \in \{0, 1\}^\omega$ 任意。然后，定义流

$$\rho = \prod_{i \in \mathbb{N}} \xi(i) \cdot 2^i = \xi(0) \cdot \xi(1) \cdot 2 \cdot \xi(2) \cdot 22 \cdot \xi(3) \cdot 222 \cdot \xi(4) \cdot 2222 \cdot \ldots,$$

并令 M 为计算两两异或的一维元胞自动机（参见例 4.3）：$f_M(x{:}y{:}\sigma) = (x \oplus y) \cdot f_M(y{:}\sigma)$。如果 $x = 2$，则我们可以用 $\delta(x) = \varepsilon$ 将 $\rho \triangleright_*^{\mathrm{C}} \xi$ 约简；否则用 $\delta(x) = x$。此外，$\xi \triangleright_*^{\mathrm{C}} f_M(\xi)$，而对于大多数 $\xi \in \Sigma^\omega$，$\rho \not\triangleright_*^{\mathrm{C}} f_M(\xi)$，因为 $\xi(i)$ 之间的间隔不断增大。□

由于（Σ^ω，$\triangleright_*^{\mathrm{C}}$）不是预序，因此这个模型不如我们的 1CA 归约有趣。不过，未来的工作可以研究 $\triangleright_*^{\mathrm{C}}$ 的传递闭包，将其作为衡量流复杂度的一种方法。

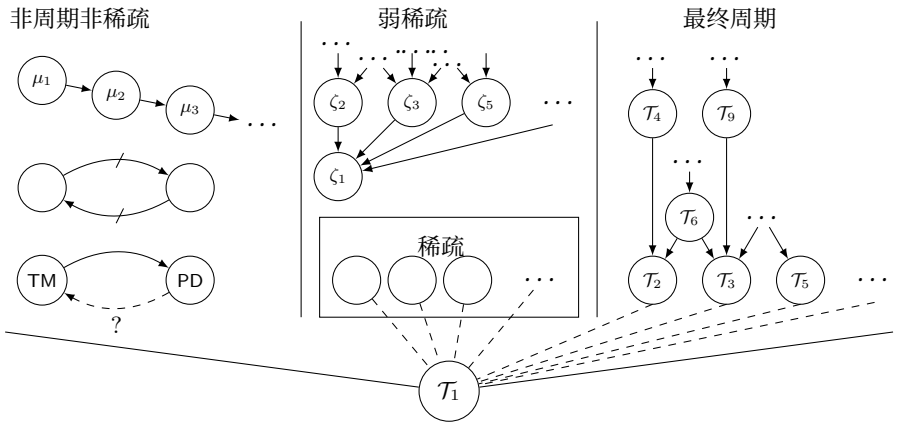


图4：一维元胞自动机度层级的示意图。每个圆圈表示一个度，箭头表示可约性。图中包括无限下降链、不可比较度、$\mathrm{TM}_{\triangleright^C}$倍周期序列、稀疏度和弱稀疏度，以及最终周期度。

7.3 其他可约性概念

极限可约性 允许元胞自动机规则无限期执行，从而产生极限集 [42],，这促使人们研究极限 *1CA* 可约性。在这种可约性中，流通过元胞可能无限期运行的元胞自动机进行转换。由于初等元胞自动机是图灵完备的， [9], 极限 1CA 归约可能与图灵归约有相似之处。

双侧元胞自动机 有限状态转导器可约性可以自然地扩展到双侧流 [6], ，也可以用一维元胞自动机以类似方式实现。与单侧一维元胞自动机不同，双侧一维元胞自动机在有限变异下并不封闭，并且可能具有有限度（例如，$\ldots 0000 \ldots$ 的度是有限的），尽管关于周期流和稀疏流的结果很可能仍然适用。

8 结论与未来工作

总而言之，$(\mathfrak{D}, \leq^{\mathrm{C}})$ 是一个偏序集，其中归约在对流进行有限改变后保持不变。我们完整地分类了最终周期度，并确定了 $\mathcal{T}_n \leq^{\mathrm{C}} \mathcal{T}_m$ 当且仅当 $n \mid m$。此外，我们通过有策略地将最终周期流压缩为单个流，找到了无限下降链，这表明该层级不是良基的。稀疏流通常是原子，其度只包含弱稀疏流，而弱稀疏流可以在其中编码周期流。极大流必须将每个有限字都包含无限多次，而流集的上确界通常不存在。图 4 展示了该层级的示意图。

未来工作 如第 5.3 节所述，极大度是否存在仍是一个开放问题。此外，我们没有对一些特定的知名流进行分类，因为我们认为这与本文的研究方向正交。我们猜想 TM $\not\subset^{\mathrm{C}}\mathrm{PD}$，这或许可以通过证明异或运算不可逆来证实；证明思路类似于引理 5.35 中的方法。最后，第 7 节介绍了可约性的其他定义，它们可能成为未来研究的方向。

致谢。 部分经由 ERC Starting Grant 101077178 (DEUCE) 资助。

利益披露。 作者声明不存在与本文内容相关的利益冲突。

参考文献

1. Allouche, J., Shallit, J.O.：《自动序列——理论、应用与推广》。剑桥大学出版社（2003） 2. Arora, S., Barak, B.：《计算复杂性：现代方法》。剑桥大学出版社（2009） 3. Belovs, A.：无限字机器偏序集的一些代数性质。RAIRO Theor. Informatics Appl. **42**(3), 451–466（2008） 4. Belovs, A., Buls, J.：无限字机器不变集构成的完全分配格。Discussiones Mathematicae - General Algebra and Applications**27**(1), 109–121（2007） 5. Berthé, V., Rigo, M.：《组合学、自动机与数论》。第135卷，剑桥大学出版社（2010） 6. Bosma, W., Zantema, H.：用置换转换器对序列排序。Indagationes Mathematicae **28**(1), 38–54（2017） 7. Cassaigne, J., Karhumäki, J.：托普利茨词、广义周期性与周期迭代态射。European Journal of Combinatorics**18**(5), 497–510（1997） 8. Chakraborty, K., Chowdhury, D.R.：CSHR：适用于密码学的混合元胞自动机规则的选择。载于：ACRI。Lecture Notes in Computer Science，第7495卷，第591–600页。Springer（2012） 9. Cook, M.：初等元胞自动机中的普适性。Complex Syst. **15**(1)（2004） 10. Coombes, S.：海贝的几何形态与色素分布。诺丁汉：诺丁汉大学数学科学系（2009） 11. Ehrenfeucht, A., Lee, K.P., Rozenberg, G.：无相互作用的各类确定性发育语言的子词复杂度。《理论计算机科学》**1**(1), 59–75（1975） 12. Endrullis, J., Hendriks, D., Klop, J.W.：让我们来制造差异！CoRR**abs/0911.1004**（2009） 13. Endrullis, J., Hendriks, D., Klop, J.W.：流的度。Integers **11B**, A6（2011） 14. Endrullis, J., Karhumäki, J., Klop, J.W., Saarela, A.：无限字、多项式与原子的度。Int. J. Found. Comput. Sci. **29**(5), 825–843（2018） 15. Endrullis, J., Klop, J.W., Saarela, A., Whiteland, M.A.：转导器度。载于：WORDS。Lecture Notes in Computer Science，第9304卷，第1–13页。Springer（2015）

16. Exibard, L., Filiot, E., Jecker, I.：基于多序列规范的转导器综合复杂度。载于：MFCS。LIPIcs，第117卷，第46:1–46:16页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik（2018） 17. Fernando, Q.：《p 进数：导论》。Springer International Publishing（2020） 18. Filiot, E., Jecker, I., Löding, C., Winter, S.：有限转导器的等价性与一致化问题。载于：ICALP。LIPIcs，第55卷，第125:1–125:14页。Schloss Dagstuhl - Leibniz-Zentrum für Informatik（2016） 19. Finkel, O.：博雷尔层级与 ω 上下文无关语言。《理论计算机科学》**290**(3), 1385–1405（2003） 20. Finkel, O.：图灵机的 ω-语言的歧义性。《计算机科学中的逻辑方法》**10** （2014） 21. Friedberg, R.M.：两个不可比较递归可枚举集的不可解度（波斯特问题的解答，1944）。《美国国家科学院院刊》**43**(2), 236–238（1957） 22. Hirsch, M.W., Smale, S., Devaney, R.L.：《微分方程、动力系统与混沌导论》。Academic Press（2013） 23. Jacobs, K., Keane, M.：托普利茨型 0-1 序列。Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete **13**, 123–131（1969） 24. Kaufmann, N.：分类所有低于 $N3$ 的转导器度。《逻辑与计算杂志》**33**(5), 970–983（2023） 25. Kleene, S.C., Post, E.L.：递归不可解度的上半格。《数学年刊》**59**(3), 379–407（1954） 26. Li, M., Vitányi, P.M.B.：《柯尔莫哥洛夫复杂度及其应用导论》，第4版。计算机科学丛书，Springer（2019） 27. Lind, D., Marcus, B.：《符号动力系统与编码导论》。剑桥大学出版社（2021） 28. Lindenmayer, A.：发育过程中细胞相互作用的数学模型 I：具有单侧输入的丝状体。《理论生物学杂志》**18**(3), 280–299（1968） 29. Lothaire, M.：《应用词组合学》。剑桥大学出版社（2005） 30. Muchnik, A.A.：关于算法理论中可约性问题的不可解性。载于：苏联科学院通报，第108卷，第1页（1956） 31. Perrin, D., Pin, J.：《无限字——自动机、半群、逻辑与博弈》，纯粹与应用数学丛书，第141卷。Elsevier Morgan Kaufmann（2004） 32. Pettorossi, A.：《自动机理论与形式语言：基本概念、定理与技术》。Springer Nature（2022） 33. Rayna, G.：有限状态可转换性的度。《信息与控制》**24**(2), 144–154（1974） 34. Shapiro, N.：可计算度。《美国数学会汇刊》**82**(2), 281–299（1956） 35. Shoenfield, J.R.：《不可解度》，北荷兰数学研究丛书，第 2 卷。North-Holland（1971） 36. Sloane, N.J.A.：在线整数序列百科全书。https://oeis.org（2025 2025 37 18 ），访问日期：September . Soare, R.：《递归可枚举集与度：可计算函数和可计算生成集研究》。数学逻辑展望丛书，Springer（1987） 38. Staiger, L.：《无限字的柯尔莫哥洛夫复杂度》。《理论计算机科学》**383**(2-3), 187–199（2007）

39. Thomas, W.：无限对象上的自动机。载于：形式模型与语义，第 133–191 页。Elsevier（1990）40. Van Wesep, R., Kechris, A.S., Löwe, B., Steel, J.R.：Wadge 度与描述集合论。载于：Wadge 度与射影序数。Cabal 研讨会。第 2 卷，第 24–42 页（2012）41. Wolfram, S.：元胞自动机的统计力学。《现代物理评论》**55**（3），601（1983）42. Wolfram, S.：元胞自动机的计算理论。《数学物理通讯》**96**（1），15–57（1984）43. Zenil, H.：算法复杂度估算方法综述：选项、挑战与新方向。《熵》**22**（6），612（2020）44. Zhang, Y.：素数间的有界间隔。《数学年刊》，第 1121–1174 页（2014）

第 6 节算术层级分类的证明

在第 6 节中，我们介绍了一种算法，它以两个流作为输入，并估计一个流是否可约为另一个流。在全文中，输入流都被视为预言机，不指定它们的编码方式。如果用图灵机表示流，我们就可以形式化一个具体的判定问题，并将其置于算术层级中。鉴于对图灵机进行推理的难度，在这种设定下，该问题确实不可判定。

首先，我们假设（与通常做法一样）部分可计算函数 $\{\varphi_i\}_{i\in\mathbb{N}}$ 的有效枚举。接着，如果 $\sigma \in \Sigma^\omega$ 满足可计算$\exists i \in \mathbb{N},\ \forall j \in \mathbb{N},\ \varphi_i(j) = \sigma(j)$，我们就称该流为可计算。这使我们能够将元胞自动机可约性表述为一个算术性质：

定义 A.1。 可计算 1CA 可约性是如下集合：

$$\triangleright^{\mathrm{C}}_{\mathrm{comp}} = \{\langle s,t\rangle \in \mathbb{N} \mid \varphi_s, \varphi_t \ are\ total\ and exists N \in \mathbb{N}, \forall i, i' \in \mathbb{N}, \\ \mathcal{V}_{\varphi_s,N}(i) = \mathcal{V}_{\varphi_s,N}(i') \Rightarrow \varphi_t(i) = \varphi_t(i')\}.$$

引理 A.2。$\triangleright^{\mathrm{C}}_{\mathrm{comp}} \in \Delta^0_3$。

证明。 这是定义 A.1 的一个推论。 □

引理 A.3. $\triangleright^{\mathrm{C}}_{\mathrm{comp}} \in \Sigma^0_2$-困难。

证明。 只需证明集合 $\mathrm{Fin} = \{e \in \mathbb{N} \mid \mathrm{dom}(\varphi_e) \mathrm{is finite}\}$ 能多对一归约到 $\triangleright^{\mathrm{C}}_{\mathrm{comp}}$；这是一个众所周知的 Σ^0_2-完全问题 [37], 。首先，我们引入下面的函数：

$$\varphi_e(k)_n = \begin{cases} 0 & \text{if } \varphi_e(k) \text{ has halted precisely at step } n, \\ 1 & \text{otherwise.} \end{cases}$$

令 $\psi_e : \mathbb{N} \to \{0,1\}$ 为通过对所有 $\varphi_e(k)_n$ 进行交错搜索而得到的全可计算函数。也就是说，给定一个双射 $\pi : \mathbb{N} \to \mathbb{N}^2$，我们令 $\psi_e(i) = \varphi_e(x)_y$，其中 $\pi(i) = (x, y)$。于是，我们有 $\varphi_e \in \mathsf{Fin} \Leftrightarrow 1^\omega \triangleright^{\mathrm{C}}_{\mathrm{comp}} \psi_e$，这意味着 $\mathsf{Fin} \leq_m \triangleright^{\mathrm{C}}_{\mathrm{comp}}$，因此 $\triangleright^{\mathrm{C}}_{\mathrm{comp}}$ 是 Σ^0_2-难的。□

引理 A.4。 $\triangleright^{\mathrm{C}}_{\mathrm{comp}} \in \Pi^0_2$-难。

证明。 集合 $\mathsf{Tot} = \{e \in \mathbb{N} \mid \varphi_e \text{ 是全的}\}$ 是一个著名的 Π^0_2-完全问题 [37],，并且 Tot 多一归约到 $\triangleright^{\mathrm{C}}_{\mathrm{comp}}$，经由 $f(n) = \langle n, n\rangle$。□

B Δ-轨道与初等元胞自动机

一阶差分算子 定义为 $\Delta(x : y : \sigma) = (x \oplus y) : \Delta(y : \sigma)$，其中$\oplus$ 表示模 2 加法（异或）。众所周知， $\Delta(\mathrm{TM}) = \mathrm{PD}$，如例 4.3 所述。受此结果启发， [12]就一阶差分算子提出一个本质问题：反复迭代 Δ会遇到什么？为回答这一问题，他们引入如下概念：给定一个流 σ，其 Δ-轨道 是序列 σ， $\Delta(\sigma)$ $\Delta^2(\sigma)$ $\Delta^3(\sigma)$ $\ldots$ Δ-轨道使得能够对流进行“图形化”分析，其中轨道序列中的每个流都显示为图像中的一行像素。也就是说，图像中的第一行像素对应于 σ，第二行对应于 $\Delta(\sigma)$，依此类推。通过采用这种“图形化”评估技术， [12] 将谢尔宾斯基序列 $\mathrm{S} = 00111100011000011\ldots$ 与《梅菲斯托圆舞曲》序列 $\mathrm{W} = 0010011100010011\ldots$ 的 Δ-轨道可视化，从而作出一个奇特的发现：$\Delta^2(\mathrm{S}) = \Delta^3(\mathrm{W})$。这些 Δ-轨道如图 5 所示。由于这两个序列的定义看似彼此无关，若不使用 Δ-轨道，这一惊人事实将很难被注意到。

如例 4.3 所述，一维元胞自动机可以执行函数 Δ。这个元胞自动机是一种初等元胞自动机（ECA），即规则 102，它与规则 30 [41] 一同被归入初等元胞自动机的第 3 类。此外，一个序列的 Δ-轨道恰好是它在规则 102 下的时空图。有趣的是，自然结构与过程常常被拿来与第 3 [10] 类初等元胞自动机的行为相比较。顺理成章的下一步是分析其他初等元胞自动机所产生的轨道，以寻找与其他著名序列之间的联系。人们甚至可以用类似于算法1的算法将这一过程自动化。

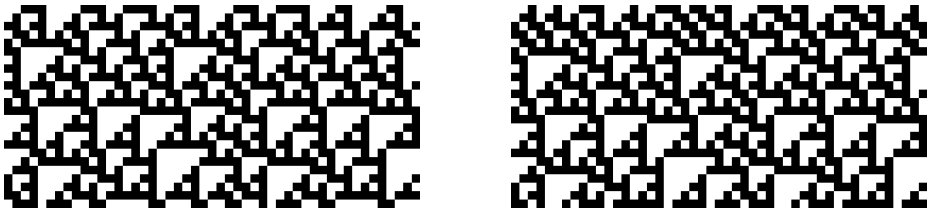


图 5：谢尔宾斯基序列（左）与梅菲斯托圆舞曲序列（右）的 Δ-轨道。