arXiv:2603.10589v1 [math.LO] 11 Mar 2026

自然数的及时标准模型与非标准模型

NIKOLAY BAZHENOV，IVAN GEORGIEV，DARIUSZ KALOCI´NSKI，STEFAN VATEV，以及 MICHAŁ WROC-LAWSKI

Abstract. 计算的抽象模型通常将自然数集 $\mathbb{N}$上的后继函数 S 视为原始运算，尽管其底层实现对应于作用于特定数值表示的复杂程序。无需诉诸记号，也可以通过将标准解释（$\mathbb{N}$， S）替换为一个同构副本 $\mathcal{A}=$（$\mathbb{N}$， $S^{\mathcal{A}}$）来分析这种行为；在该副本中， $S^{\mathcal{A}}$ 不再能通过单条指令计算。虽然只要 $S^{\mathcal{A}}$ 可计算，自然数集 $\mathcal{A}$ 上的可计算函数类就是标准的，但已有结果表明，这种不变性在原始递归层面并不成立。我们考察哪些运算集具有这样的性质：若这些运算在 $\mathcal{A}$ 上是原始递归的，那么 $\mathcal{A}$ 上的原始递归函数类仍保持标准。我们将这样的运算集称为 *bases for punctual standardness*。我们给出一系列非基结果，表明由此诱导的 $\mathcal{A}$上的原始递归函数类可能与标准函数类有显著差异。具体而言，我们证明，范围广泛的自然运算——包括 Skolem 和 Levitz 研究的原始递归函数大类——都不能构成这样的基。另一方面，我们给出了用于点标准性的自然有限基。我们的结果回答了 Grabmayr 最近提出的一个问题，并确立了某些自然有限生成结构的准时范畴性。

1. Introduction

引入复杂性类时，有时会将（$\mathbb{N}$， S）——也就是带有后继函数的自然数——视为一个基本概念。例如，用于原始递归的循环程序 [16], 以及用于部分递归函数的 while程序（参见例如 [17], 定义 I.5.5）和寄存器机（参见例如 [25]）都使用指令 $X:=X+1$，用该指令将寄存器 X 中存储的数替换为它的后继数。

相比之下，图灵机、Post机或字符串重写系统等符号计算模型，并不将后继运算视为基本操作，而是要求对其进行定义。例如，对十进制数码进行后继运算，需要一个能够计算该运算的非平凡程序。若采用不同的记数法，就需要为后继运算编写不同的程序。因此，后继运算可以通过许多不同的方式定义，每种定义都会引出一组独特的对象序列，用来表示数的递增。

同样的观察也适用于更抽象的层面，即考察自然数时：将后继运算视为原始概念，可以替换为一个其指称产生同构行为的程序。更直接地说，也可以仅仅改变 $X:=X+1$ 的语义。无论哪种情况，重新定义后继都会导出一个新结构 $\mathcal{A}=(\mathbb{N},\ S^{\mathcal{A}})$，它与 $(\mathbb{N},\ S)$ 同构。这会影响我们对后继（它现在变成了 $S^{\mathcal{A}}$）的理解，并由此影响我们对前驱、加法及所有其他函数的理解。

这一思想可以精确表述如下。设 $c_{\mathcal{A}}:(\mathbb{N},S)\to\mathcal{A}$ 是一个同构。当结构可由上下文确定时，我们省略下标 $\mathcal{A}$。一元函数 f 在 $\mathcal{A}$ 中的表示，是相对于 c而言 f的同构像，即 $c\circ f\circ c^{-1}$，简记为 $f^{\mathcal{A}}$。这一约定自然可以推广到非一元函数和关系。重要的是，随着这种重新诠释，复杂性类的概念也可能发生变化。下述概念源自可计算结构理论：

2020 *Mathematics Subject Classification.* 03C57, 03D20.*Key words and phrases.* 准点结构；原始递归；初等函数；可接受性。
N. Bazhenov 的工作得到阿卡杰姆戈罗多克数学中心根据与俄罗斯联邦科学与高等教育部签订的协议 No. 075- 15-2025-349 的支持。
I. Georgiev 和 S. Vatev 的工作得到 NextGenerationEU 通过保加利亚 NRRP 项目 No. BG-RRP-2.004-0008 - C01-70/123/195 的支持。
D. Kaloci´nski 和 M. Wroc-lawski 的工作得到波兰国家科学中心根据协议 No. 2023/49/B/HS1/03930 的支持。

定义 1. 如果 $f^{\mathcal{A}}$ 是可计算的（分别为原始递归的），我们就称函数 f 在 $\mathcal{A}$ 上是可计算的（分别为原始递归的）。

令 $\mathcal{C}^{\mathcal{A}}$ 表示在 $\mathcal{A}$上可计算的函数类。众所周知，而且不难证明，等式 $\mathcal{C} = \mathcal{C}^{\mathcal{A}}$ 成立，当且仅当 $S^{\mathcal{A}}$ 是可计算的（例如参见 [23]）。

等式 $\mathcal{C} = \mathcal{C}^{\mathcal{A}}$ 之所以理想，原因如下：只要我们希望与丘奇–图灵论题保持一致，改变指令$X := X+1$ 的语义就不应改变可计算函数类。因此，条件 $\mathcal{C} = \mathcal{C}^{\mathcal{A}}$ 将 $(\mathbb{N}, S)$ 的那些副本单独区分出来，我们称之为标准或可接受。这一概念从多个角度得到研究，包括可计算性理论（Rogers [20]意义下的可接受编号）、理论计算机科学（Riccardi [19] 和 Royer [22]意义下的可接受程序系统）、可计算结构理论 [6], 以及哲学 [23,24]。

本文从原始递归的视角出发，运用点式结构理论的方法，继续探讨这一问题 [12, 4, 9, 2]。尽管原始递归函数的范围非常广，远超实际可计算性的范畴，但它仍然标示出一条重要的理论边界：这条边界区分了一般可计算性与仅使用有界循环的算法所实现的可计算性 [16]。这条边界对计算机科学而言并不陌生。一些自然且经过充分研究的问题，其可判定性早在数十年前就已为人所知，但后来人们发现，其计算复杂度超出了原始递归的范畴。一个典型的例子是向量加法系统的可达性问题 [7]。

本文的核心出发点见于点式结构理论的奠基论文。显然，对于 $(\mathbb{N},\ S)$ 的一个原始递归副本 $\mathcal{A}$，同构映射 $c_{\mathcal{A}}$ （从标准副本到 $\mathcal{A}$）是原始递归的；然而，可以证明如下结论：

命题 2 ([4], 例 4.1(3)，第 86 页)。 存在 $\mathcal{A} = (\mathbb{N}, S^{\mathcal{A}})$，它是 $\mathcal{S} = (\mathbb{N}, S)$ 的一个同构副本，使得 $S^{\mathcal{A}}$ 是原始递归的，但 $c_{\mathcal{A}}^{-1}$ 不是。

命题 $\mathcal{A}$ 中的结构 2 是可接受的，即 $\mathcal{C} = \mathcal{C}^{\mathcal{A}}$，因为 $S^{\mathcal{A}}$ 是可计算的。然而，根据观察 5（下文将证明），上述命题有如下推论：存在一个原始递归函数 f，使得其表示 $f^{\mathcal{A}}$ 不是原始递归的。若将 $\mathcal{PR}$ 和 $\mathcal{PR}^{\mathcal{A}}$分别记为原始递归函数类以及在 $\mathcal{A}$上原始递归的函数类，则可得 $\mathcal{PR} \neq \mathcal{PR}^{\mathcal{A}}$。换言之，在 $\mathcal{A}$ 上原始递归的函数类并不等同于标准的原始递归函数类。

从原始递归计算的角度来看，上述情况令人不安。特别是，如果我们在循环程序中改变指令 $X := X+1$ 的语义，将其替换为 $S^{\mathcal{A}}$，那么尽管 $S^{\mathcal{A}}$ 本身是原始递归的[1]，并且使用的显然是有界循环，我们仍会得到一个非标准的原始递归函数类 $\mathcal{PR}^{\mathcal{A}}$。这与受限版本的丘奇–图灵论题相悖；该论题是一种隐含的准形式化假设，在 [4] 第 1.6 节中作了明确表述：一个函数是原始递归的，当且仅当它可以由只使用有界循环的算法描述。

因此，我们有理由引入以下概念。我们将定义域为 $\mathbb{N}$，且其签名中的关系和运算均为原始递归的结构称为点态结构。

定义 3 (点标准性)。 我们称点态副本 $\mathcal{A}$ 属于 $\mathcal{S} = (\mathbb{N},\ S)$ 是逐点标准的，当且仅当 $\mathcal{PR} = \mathcal{PR}^{\mathcal{A}}$。否则，我们称 $\mathcal{A}$ 是点非标准的。

点标准性的充要条件可以用与标准副本 $\mathcal{S}$ 之间满足适当约束的同构来表述。

定义 4 ([12])。 准点结构 $\mathcal{A}$ 和 $\mathcal{B}$ 逐点同构，若存在一个原始递归同构 f 将二者联系起来，且其逆映射 f^{-1} 也是原始递归的。我们称这样的同构 f 点态的。

观察5。 结构 $\mathcal{A}$ 是 $\mathcal{S}$ 的定义域为 $\mathbb{N}$ 的一个副本，当且仅当 $\mathcal{A}$ 逐点标准时，它与 $\mathcal{S}$ 逐点同构。

证明。 如果 $\mathcal{A}$与 $\mathcal{S}$逐点同构，那么 $c_{\mathcal{A}}$ 和 $c_{\mathcal{A}}^{-1}$ 是原始递归双射，因此 $\mathcal{PR}^{\mathcal{A}} = \{c_{\mathcal{A}} \circ f \circ c_{\mathcal{A}}^{-1} \mid f \in \mathcal{PR}\} = \mathcal{PR}$ 和 $\mathcal{A}$ 是准时标准的。反过来，设 $\mathcal{A}$ 是准时标准的。对于任意 x，我们考察数 $c_{\mathcal{A}}(0)$ $c_{\mathcal{A}}(1) \ldots, c_{\mathcal{A}}(x)$。由于 $c_{\mathcal{A}}$ 是单射，至少有一个

[1]事实上，在 在这个特定情况下，可以在线性时间内计算出 $S^{\mathcal{A}}$ 。

它们是 $\geq x$。我们定义 $f(x) = \mu_{t\leq x}[c_{\mathcal{A}}(t) \geq x]$。显然，函数 f 是原始递归的，并且对于所有 x，$c_{\mathcal{A}}(f(x)) \geq x$。但这也意味着，对于所有 x，$c_{\mathcal{A}}(f(c_{\mathcal{A}}^{-1}(x))) \geq c_{\mathcal{A}}^{-1}(x)$。由于 $\mathcal{A}$是准时标准的，函数 $c_{\mathcal{A}} \circ f \circ c_{\mathcal{A}}^{-1}$ 是原始递归的。由此可得，$c_{\mathcal{A}}^{-1}$ 也是原始递归的，且 $\mathcal{A}$ 与 $\mathcal{S}$逐点同构。□

用与标准副本的及时同构来刻画点标准性，虽然在形式上是正确的，但存在一个缺点：它是相对于自然数的一种特定表示，即 $\mathcal{S}$，来评估点标准性。在某种意义上，$\mathcal{S}$ 根据定义就是及时标准的，而这一选择可能会被视为任意的。因此，将这种方法视为外部方法是合理的。相反，我们主张寻求一种更一般且更稳健的内部刻画。

这种刻画的指导已经可以从现有的可接受性方法中获得。例如，Rogers 将部分递归函数的编号的可接受性视为相对于典范编号 [20], 但后来，这种方法被通过利用控制结构从内部刻画可接受性的方法所取代 [19] 或者通过自然数算术运算的可实现性来刻画 [23, 6]。遵循第二种方法，我们旨在确定一组尽可能小且自然的基本算术运算，使得它们在 $\mathcal{A}$ 上的原始递归性等价于 $\mathcal{A}$的点标准性。Grabmayr 最近也考虑了类似的方向（[10], 第 10 页）。

1.1. 贡献。

本文提出的主要新工作概念如下。

定义 6（点标准性的基）。 函数类 $\mathcal{F} \subseteq \mathcal{PR}$ 是点标准性的基，当且仅当对于 $\mathcal{S}$ 的每个副本 $\mathcal{A}$，以下条件成立：若 $\mathcal{F}$ 中的元素在 $\mathcal{A}$上是原始递归的，则 $\mathcal{A}$ 是点标准的。

请注意，上述概念可以推广到自然数上函数的任意复杂性类。特别地，根据 Shapiro 的结果 [23], ，类 $\{S\}$ 是可接受性的基，因为 S 是可计算的，并且对于 $\mathcal{S}$的每个副本 $\mathcal{A}$，如果 S 在 $\mathcal{A}$ 上可计算，那么 $\mathcal{C} = \mathcal{C}^{\mathcal{A}}$。

我们的贡献有两个方面。

首先，我们给出了若干非基结果——一些无法构成点标准性之基的自然类的例子。其中，我们观察到两种现象。一方面，$\mathcal{S}$ 的某些点副本似乎与 $\mathcal{S}$ 大不相同：令人惊讶的是，$\mathcal{PR}$中相当大的自然子类在这些副本中无法保持原始递归性（贡献1a）；这些例子是通过利用阿克曼函数 [1]的性质构造出来的。另一方面，我们表明，$\mathcal{S}$的点副本即使保留了 $\mathcal{PR}$ 中较大自然子类的原始递归性，仍可能是点非标准的（贡献1b）；这些例子通过对点式结构理论中已知的岛屿技术进行精细调整，再将其应用于 Skolem 和 Levitz 此前研究过的原始递归函数大类而得到。总之，贡献1a和贡献1b表明，寻找点标准性的基并非易事。

其次，我们给出自然的准时标准性基（贡献2）。其自然性源于对熟悉的算术运算的使用。该结果通过对Kalimullin 等人的构造进行复杂性理论分析 [12] 并应用已知的初等函数代换基而得到。这解决了Grabmayr最近提出的一个技术问题 [10], ，我们将在后文更详细地讨论。此外，我们的准时标准性基立即产生自然的有限生成点式范畴的结构，即其点副本始终逐点同构的结构。此前，点式范畴的结构要么通过不自然的例子为人所知，要么出现在非有限生成的情形中。

下面我们更详细地描述各项贡献。

贡献*1a*。 第一组否定结果给出了 $\mathcal{S}$ 的点副本，其上某些自然运算是原始递归的，然而大类原始递归函数在这些副本上却不再保持原始递归。这些证明依赖于一种新方法，该方法利用阿克曼函数的性质——阿克曼函数是可计算但非原始递归函数的典型例子。这一方法使我们能够系统研究所得模型；特别是，它使我们能够证明，某些广泛的函数类或关系类在给定副本上不是原始递归的。

下列定理在第3 节中得到证明（见命题16 和引理19）。该构造的最初目标是产生一个点态副本 $\mathcal{A}$，即 $\mathcal{S}$ 的副本，使得前驱函数不是原始

递归的，在 $\mathcal{A}$上。在命题 16 中，我们证明：事实上，还有更多自然的原始递归函数在 $\mathcal{A}$上不是原始递归的。

定理7. 存在 $\mathcal{S}$ 的一个点态副本 $\mathcal{A}$ ，使得序关系在 $\mathcal{A}$上是原始递归的，并且每当 $f : \mathbb{N} \to \mathbb{N}$ 是一个对所有充分大的 x都大于 $2x$ 的原始递归函数时， f 在 $\mathcal{A}$上不是原始递归的。

下一个定理在第3.4 节中得到证明（见命题27），它展示了类似的现象：我们可以使加法成为原始递归的，但仍有大量自然的原始递归函数在某个副本上可能不是原始递归的。

定理8. 存在 $\mathcal{S}$ 的一个点态副本 $\mathcal{B}$ ，使得序关系和加法在 $\mathcal{B}$上是原始递归的，但每当 $f : \mathbb{N} \to \mathbb{N}$ 是一个对所有充分大的 x都大于 x^2 的原始递归函数时， f 在 $\mathcal{B}$上不是原始递归的。

利用同样的技术，我们可以将这一现象推进到加法和乘法（见命题29）：

定理9. 存在 $\mathcal{S}$ 的一个点态副本 $\mathcal{C}$ ，使得序关系、加法与乘法在 $\mathcal{C}$上是原始递归的，但每当 f： $\mathbb{N} \to \mathbb{N}$ 是原始递归的，且对所有充分大的 x都大于 $x^{\log_2 x}$ 时， f 在 $\mathcal{C}$上不是原始递归的。

结构 $\mathcal{B}$ 与 $\mathcal{C}$ 是由结构 $\mathcal{A}$ 通过引入一种处理二进制展开的一般方法而得到的，该方法用于产生 $\mathcal{S}$ 的新点副本（由已构造的副本得到）。我们证明，无法用同一方法将这一系列定理推广到指数函数的层面（见命题30）。相比之下，大陆–岛屿技巧能够成功地保留所有常底的指数函数（见贡献1b）。

定理9 具有如下基本推论，我们认为这令人惊讶，因为 S、 $+$、 $\times$、与$<$ 构成自然数典范公理化中的核心原始概念，并且构成普通算术中最基本的运算。

推论 10. 后继 S、加法 $+$、乘法 $\times$和序关系$\leq$ 这些标准算术运算并不构成*punctual*标准性的基。

在本文中，我们将看到这一结果的三个证明。除定理9外，它还可以通过不同方法，利用定理1.3（来自 [11] ）以及观察5导出。另一个证明将从定理12中提取。

所有技术细节，包括证明，均可在第3节中找到。

贡献*1b.* 贡献1a的新颖技术对于精细分析某些特定的（快速增长的）原始递归函数非常有用。然而，为了将我们的方法论应用于熟悉的大类函数，我们采用已经确立的大陆–岛屿方法 [12, 5]。这一构造方法最初在 [12], 中引入，用于构造一个及时的同构副本 $\mathcal{A}$ ，作为给定可计算结构 $\mathcal{M}$的副本。该构造保证某些函数 $f^{\mathcal{A}}$ （通常是后继，也可能包括其他函数）的原始递归性得以保持。同时，通过对角化确保另一个函数（或者是 $\mathcal{A}$的定义域中元素之间的函数，或者是从 $\mathcal{A}$ 到 $\mathcal{M}$上的同构）不是原始递归的。

本文通过证明以下结果展示了岛屿技术。除非 P是单元素集，否则我们无法使用在贡献1a中用于证明所述定理的方法证明这一结果：

定理 11. 对于任意有限素数集 P，存在 $\mathcal{B}$—— $\mathcal{S}$的一个点态副本——使得在 $\mathcal{B}$ 上原始递归的线性函数仅为 $f(x) = ax + b$，其中 a 的所有素因子都属于P。

为了拓展大陆–岛屿技巧的适用范围，我们证明定理 12 （见下文），该定理给出了通过此方法构造结构 $\mathcal{A}$ 的点态非标准副本的充分条件 $\mathcal{S} =$ （$\mathbb{N}$， S）。直观地说，可以将定理 12视为一条元定理：定理12提炼出一组技术条件，这些条件既能保持自然数集 $\mathcal{F}$ 中一大类函数 $\mathcal{A}$的原始递归性，又能确保前驱函数 Pred 的像$^{\mathcal{A}}$ 不是经典意义上的原始递归函数。这些条件本质上

结合了先前对大陆–岛屿方法的应用（例如，见 [12, 5, 11, 8]）以及 Levitz 研究的函数类的递归论性质 [14]。

在定理 12中， $\mathcal{Q}$ 是一致的原始递归函数族，而 $\mathcal{F}$ 是一致的一元原始递归函数族。所有其他技术条件均在第4节中作了充分说明。

定理12。 如果一个 L-族 $(\mathcal{Q}, \mathcal{F})$ 具有范式集，那么族 $\mathcal{F}$ 就不是点标准性的基。

定理 12带来了两个重要应用。首先，它为推论10提供了独立的论证。其次，它可用于推导一个强得多的结果。Skolem [26], Levitz [14]在相关研究的基础上，考察了以下原始递归函数子类，其中尤其包含所有指数多项式。

定义 13。 Levitz 类 $\mathcal{L}$ 是最小的函数类 $f\colon \mathbb{N} \to \mathbb{N}$，包含常函数 0、1 和恒等函数 $x \mapsto x$，并满足以下条件：若 f、$g \in \mathcal{L}$ 和 $n \in \mathbb{N}^+$，则函数

(a) $x \mapsto f(x) + g(x)$,
(b) $x \mapsto f(x) \cdot g(x)$,
(c) $x \mapsto n^{f(x)}$，以及
(d) $x \mapsto x^{f(x)}$

也属于 $\mathcal{L}$。

如果从 *Levitz* 类的定义中省略第 *(d)* 项，那么所得的函数类将记为 $\mathcal{L}_c$。注意， $\mathcal{L}_c \subset \mathcal{L}$，并且类 $\mathcal{L}_c$ 对复合封闭。

推论 14. 类 $\mathcal{L}_c$ 不是点标准性的基。

总而言之，迄今讨论的结果表明，寻找点标准性的基并非易事。最自然的候选——经典自然数公理化描述中出现的算术运算——无法提供这样的基。同样，将潜在的基扩展到某些自然的快速增长函数（包括 $\mathcal{L}_c$ ），也无法解决这一问题。

虽然我们在本文中并未证明这一点，但我们指出，可以直接构造$(\mathbb{N}, S, \mathrm{Pred}, +, \times, <)$的一个点式非标准副本。这缓解了这样一种潜在担忧：点非标准性可能仅仅源于试图只保留某些非递减函数。此外， $\mathcal{L}_c \cup \{\mathrm{Pred}\}$ 似乎也不是punctual标准性的基，这一点看起来是合理的。

所有技术细节，包括证明，均可在第4节找到。

贡献*2.* 为了获得自然的准时标准性基，我们首先对Kalimullin、Melnikov和Ng的构造（[12], 命题 4.2）进行复杂性理论分析，该构造产生一个刚性的（即仅有一个自同构）点范畴结构 $\mathcal{B} = (\mathbb{N}, s, c, o)$。结构 $\mathcal{B}$采用人为词汇以及同样人为的解释，专门设计用来满足预期的要求。关于细节，我们请读者参阅原文；在此，我们仅指出 o是一个常数， $o, s(0)\ ss(o) \ldots$ 是一条 ω-链，并且 c是一个将定义域划分为有限的两两不交循环的一元函数。我们的分析表明， $\mathcal{B}$ 在Kalm´ar[13] 的意义上是初等的（等价地， $\mathcal{E}^3$ 位于Grzegorczyk层级中）。由此可知，$(\mathbb{N}, S, s, c, o)$是一个初等的、刚性的点范畴结构。

下一步是利用有关初等函数类的代换基的现有结果。函数类 $\mathcal{F}$ （在此情形下为初等函数）的代入基，是由 $\mathcal{F}$ 中的函数组成的有限集合，通过结合投影和常量应用替换算子，即可生成整个函数类。以下是这样一个基的例子： $x + y$, x 模 y, x^2,2^x （例如，参见 [15]中的推论4.8）。我们证明

定理15. 初等函数类的每个代换基都是及时标准性的一个基。

根据观察5，这一结果提供了自然的有限生成的点式范畴结构的例子。此前，这类结构要么是通过上述人为构造的有限生成结构$\mathcal{B} = (\mathbb{N},\ s,\ c,\ o)$，要么是通过非有限生成的结构（例如完全图或阿贝尔 p群 $\bigoplus_{i \in \omega} \mathbb{Z}_p$）而为人所知。

我们的结果还可以说提供了点标准性的一种句法刻画。事实上，我们以纯粹内在的方式描述这一概念：对哪些函数被假定为原始递归函数作出要求，而无需援引与某个固定标准副本之间的任何相似性判据。

定理15 回答了 Grabmayr 提出的问题（参见 [10], 第 10 页）：他希望确定足以唯一确定点式同构类型的点式数系的复杂度约束。在 Grabmayr 的论文语境中，复杂度约束的含义如下：在一个数系（也就是用我们的术语来说，一个 $\mathcal{S}$的点副本）中，除了后继之外，还需要哪些运算是原始递归的，才能使该数系成为点式范畴的？这里，额外的复杂度约束通过条件“ $\mathcal{F}$ 的成员在 $\mathcal{A}$上是原始递归的”来形式化（参见定义6）。值得注意的是，给定 $\mathcal{E}^3$的一个代入基，定理15 不仅确定了唯一的点式同构类型——而且确定了正确的点式同构类型，也就是说：这类系统的所有点副本都是准时标准的。

贡献2的技术部分见第5节。

结论与开放问题。本文探讨了Grabmayr最近（且独立地）提出的以下问题 [10]：在（ $\mathcal{A}$ $\mathbb{N}$， S）的一个副本上，必须要求哪些运算是原始递归的，才能保证 $\mathcal{A}$ 上原始递归的函数恰好与标准原始递归函数一致？根据观察5，这等价于要求刻画（$\mathbb{N}$， S）的点式同构类型。我们将这类运算集称为点标准性的基（定义6）。

我们得出了一些否定性结果。特别是，标准算术运算 $S, +, \times, <$ 虽然在自然数的典范公理化刻画中占有重要地位，却并非点标准性的基。此时，识别这类基的任务变得非同寻常，因为乍看之下，很少有函数比 $S, +, \times, <$ 更自然。此外，本文所呈现的非基结果范围表明，某些相对自然且经过深入研究的原始递归函数类无法构成这类基。为解决这个问题，间接策略取得了成功：我们从点式结构理论中取用一个范畴性结果，分析其底层构造的复杂度，随后利用已有的关于初等函数类的代换基结果。

定理 15 为点标准性提供了一个自然的结构刻画。例如，使用马赞蒂引入的一个基 [15], ，点态副本 $\mathcal{A}$（$\mathbb{N}$， S）当且仅当函数 $x + y$， x 模 y， x^2以及 2^x 在 $\mathcal{A}$上是原始递归的时，才是点标准的。我们的结果也可以表述如下。设 $\mathcal{F}$ 是初等函数类的一个代入基。则对（ $\mathcal{A} =$ ，$\mathbb{N}$）的每一个同构表示 $\mathcal{F}^{\mathcal{A}}$（$\mathbb{N}$， $\mathcal{F}$），以下等价关系成立： $\mathcal{A}$ 是原始递归的，当且仅当 $\mathcal{A}$ 是点标准的。这表明，点标准性的概念既自然又概念上优雅。

定理15 还具有一个进一步的根本性推论：如果所有初等函数在 $\mathcal{A}$上都是原始递归的，那么结构 $\mathcal{A}$ 仍然是点标准的。等价地，任何点标准性的失败，都必然已经在初等函数层面表现出来。更强地说，函数 $x + y$， x 模 y， x^2或 2^x 中至少有一个必须不是 $\mathcal{A}$上的原始递归函数。这进一步支持了这样的观点：带有后继的自然数的点非标准副本是极不自然的对象。

最后，将这一图景与我们的否定性结果，尤其是推论 14，并置起来颇具启发性。该推论表明，一个庞大且自然的函数类 $\mathcal{L}_c$，不足以保证点标准性。尽管 $\mathcal{L}_c$ 包含在初等函数类中，但它缺少某些关键的初等运算；这些运算的缺失会使其他初等函数变得复杂到难以处理。这再次凸显了：一组运算若要成为点标准性的基，所需的微妙平衡。

由此引出了几个尚未解决的问题。

虽然否定性结果存在局限，但在许多情况下，这一局限究竟在哪里并不明确。特别是，我们不知道定理 12 还能在多大程度上进一步推广，以及整个 Levitz 类是否构成 punctual标准性的基。我们猜想并非如此。其他大型原始递归函数类也是值得进一步研究的自然方向。例如，Barra 和 Gerhardy 在[3], 中研究了使用迭代幂运算定义的函数。

我们还猜想，甚至 Grzegorczyk 的第二类 $\mathcal{E}^2$ 也是 punctual标准性的基。这将需要对 [12] 中关于一个点分类刚性结构的构造进行实质性修改。

2. Preliminaries

关于原始递归和 Grzegorczyk层级的详细背景，请参阅专著 [18] 。读者也可以查阅有关次递归层级的教科书 [21]。

设 f 和 g 是全函数。若 f 可由 g 和初始初等函数通过复合与（有界）原始递归得到，则记作 $f \leq_{PR} g$ ($f \leq_E g$)。

照例， $f \equiv_E g \Leftrightarrow f \leq_E g$& $g \leq_E f$和 $f \equiv_{PR} g \Leftrightarrow f \leq_{PR} g$& $g \leq_{PR} f$。
若函数 f 满足 $f \leq_{PR} 0$ ($f \leq_E 0$)，则称其为原始递归的（初等的）。
一个关系 R 是原始递归的（初等的），如果其特征函数是原始递归的（初等的）。

我们假定一个固定的（满射的）有序对编码： $\langle x, y\rangle$ 是有序对(x, y)的编码，其中我们记$x = (\langle x, y\rangle)_0$ 以及 $y = (\langle x, y\rangle)_1$。照例，函数$(x, y) \mapsto \langle x, y\rangle$, $z \mapsto (z)0$, $z \mapsto (z)_1$ 被选为初等的。此外， $\langle x, y\rangle$ 关于 x, y两者都严格递增，这意味着对所有非零的 z，有$(z)_0 < z$ 以及$(z)_1 < z$。

设 $\mathcal{T}_1$ 为初等的Kleene谓词，从而任一部分可计算一元函数都具有形式$\lambda x.(\mu s \mathcal{T}_1(e, x, s))_0$。我们可以利用阿克曼函数 $A(x, y)$来定义所有一元原始递归函数这一类的可计算枚举：

$$\psi_e(x) = (\mu s \leq A((e)_1, x)\ \mathcal{T}_1((e)_0, x, s))_0.$$

我们还定义

$$\psi_{e,s}(x)\downarrow \iff \mathcal{T}_1((e)_0, x, s),$$

这表示计算 $\psi_e(x)$ 需要 $\leq s$ 步。我们将经常用 $\psi_{e,\ s}(x)\uparrow$ 表示 $\psi_{e,\ s}(x)\downarrow$ 的否定。我们可以假设下述性质：

$$\psi_{e,s}(x)\downarrow \implies s > \max(e, x, \psi_e(x)). \tag{1}$$

设 $a : \mathbb{N} \to \mathbb{N}$ 是一个具有以下性质的序列：

1. a 严格递增；2. 关系 $a_n = k$ 是初等的；3. 对任意一元原始递归函数 f，当 n 充分大时， $f(a_n) < a_{n+1}$。

众所周知， $a_n = A(n,\ n)$ 满足这三个性质。
以下观察显而易见：

（i）性质 3. 意味着，对于任意一元原始递归函数 f， $f(n) < a_n$ 对所有充分大的 n都成立，因此 a 不是原始递归函数。

证明。 取 $g(n) = \max_{k \leq n+1} f(k)$。那么 g 是原始递归的，并且

$$f(n+1) \leq g(n) \leq g(a_n) < a_{n+1}$$

除有限多个 n 外，对所有 n 都成立。

（ii）性质1. 和 2. 意味着存在一个初等函数

$$a^{-1} : \mathbb{N} \to \mathbb{N} \quad \text{with} \quad a^{-1}(a_n) = n. \tag{$*$}$$

证明。 给定输入 k，搜索满足 $a_n = k$ 的 $n \leq k$。若搜索成功，则给出输出 $a^{-1}(k) = n$。否则，给出输出 0。

（iii）作为一元关系， $Ran(a)$ 是初等的：

$$k \in Ran(a) \iff \exists n \leq k\ a_n = k.$$

(iv) 唯一的严格递增一元函数 h 满足 $ran(h) = \mathbb{N} \setminus Ran(a)$ 是初等的。 证明。性质 3. 意味着，对于所有充分大的 x， $\{x+1, x+2\} \cap (\mathbb{N} \setminus Ran(a)) \neq \emptyset$ ($Ran(a)$ 是一个非常稀疏的集合）。因此 $h(x) \leq 3x$ 对除有限多个 x之外的所有情况都成立。

3. Examples for Punctually Nonstandard Structures

我们提出一种构造点态副本 $\mathcal{A} = (\mathbb{N}, S^{\mathcal{A}})$ 的新思路，使得前驱函数在 $\mathcal{A}$ 上不是原始递归的。与 [12] 和 [4]，中的构造不同，这一思路能够对所构造的副本 $\mathcal{A}$ 上的原始递归函数和关系进行更精细的分析。

我们通过 $S^{\mathcal{A}}(a_n) = h(\langle n,\ 0\rangle)$ 来开始定义 $\mathcal{A}$。由于 a 不是原始递归的，我们无法通过原始递归算法从 $h(\langle n,\ 0\rangle)$ 计算出 a_n ；但由 $(*)$ 可知，给定 a_n 后，我们可以初等计算 $h(\langle n,\ 0\rangle) = h(\langle a^{-1}(a_n)\ 0\rangle)$。现在我们要以初等方式将所有这些元素连接起来。以下是这一思路的示意：

$$a_n \longrightarrow h(\langle n,0\rangle) \longrightarrow h(\langle n,1\rangle) \longrightarrow \ldots \longrightarrow h(\langle n,a_{n+1}\rangle) \longrightarrow a_{n+1}.$$

未被使用的自然数属于集合

$$F = \{h(\langle n,i\rangle) \mid i > a_{n+1}\}.$$

显然， F 是初等的，因为 h和 a 的图都是初等的。满足 $Ran(free) = F$的唯一的严格递增一元函数 $free$ 是原始递归的：如果我们已经计算出 $h(\langle n,\ i\rangle) \in F$，那么 F中下一个元素的值至多为 $h(\langle n,\ i+1\rangle)$。

为了补全我们的模型，我们必须将 F 中的元素插入上图（每个自然数都必须恰好出现一次）。我们可以按以下方式进行：

$$a_n \longrightarrow free(n) \longrightarrow h(\langle n,0\rangle).$$

更正式地说，我们通过以下原始递归算法定义 $S^{\mathcal{A}}$ 。

给定输入 x:

1. 如果存在 $n \le x$，使得 $a_n = x$, $S^{\mathcal{A}}(x) = free(n)$。2

· 如果存在 ists $n \le x$, 后继h that $free(n) = x$, $S^{\mathcal{A}}(x) = h(\langle n,0\rangle)$.

3. 否则，存在 $k \le x$，使得 $h(k) = x$。我们计算 n，i其中 $k = \langle n,\ i\rangle$。如果 $i = a_{n+1}$，则$S^{\mathcal{A}}(x) = i$，否则 $S^{\mathcal{A}}(x) = h(\langle n,\ i+1\rangle)$。

Not请注意，在子句 3 中，我们有 $i \le a_{n+1}$，因为情形 $i > a_{n+1}$ 已由子句 2.

令 c为从 $(\mathbb{N},\ S)$ 到 $(\mathbb{N},\ S^{\mathcal{A}})$ 的唯一同构。

对于任意 $x \in \mathbb{N}$，我们将 *the $\mathcal{A}$-number of* x 称为满足 $x = c(p)$ 的唯一 $p \in \mathbb{N}$，因此 x 的 $\mathcal{A}$-number 是 $c^{-1}(x)$。

对 n 进行归纳法可轻易证明，在上述模型中 $\mathcal{A}$:

$\mathcal{A}$-数 a_n 是 $3n + \sum_{k=1}^{n} a_k$,

$\mathcal{A}$-数 $free(n)$ 是 $3n + 1 + \sum_{k=1}^{n} a_k$ ，并且

$\mathcal{A}$-数 $h(\langle n,\ i\rangle)$ 对于 $i \le a_{n+1}$是 $3n + i + 2 + \sum_{k=1}^{n} a_k$。

一方面，同构 c 是原始递归的，因为 $S^{\mathcal{A}}$ 是原始递归的，并且 $c(p) = (S^{\mathcal{A}})^p(a_0)$。另一方面， c^{-1} 不是原始递归的，因为对于所有 n， $c^{-1}(free(n)) > a_n$ 。事实上，很容易证明 $c^{-1} \equiv_E a$。

给定 p，我们可以 用原始递归计算元素 $x = c(p)$，其 $\mathcal{A}$-数为 p；但给定 x w e无法 p原始递 归地计算 $\mathcal{A}$-数 $p$of x。

还需注意，对于函数 $f\colon \mathbb{N} \to \mathbb{N}$， $f^{\mathcal{A}}(x)$ 的 $\mathcal{A}$-数为 $f(p)$，其中 x的 $\mathcal{A}$-数为 p。

3.1. 探索模型中函数的像 $\mathcal{A}$。我们来探究所构造模型 $\mathcal{A}$ 中函数像的复杂性。为简洁起见，我们省略 $\mathcal{A}$ ，用“数”代替“$\mathcal{A}$-数”；同样，在同构的记号中，用 c 代替 $c_{\mathcal{A}}$ 。

我们的第一个例子是前驱函数的像 $\mathrm{Pred}^{\mathcal{A}}$ ，它不是原始递归的。否则，$\mathrm{Pred}^{\mathcal{A}}(free(n)) = a_n$ 就会是原始递归的。

结果发现，许多其他原始递归函数 f 在所构造的模型 $\mathcal{A}$中并非原始递归。

命题 16. 设 $f\colon \mathbb{N} \to \mathbb{N}$ 是一个原始递归函数，使得对所有足够大的 x，都有 $f(x) \ge 2x$ 。那么 $f^{\mathcal{A}}$ 不是原始递归的。

证明。 假设 $f^{\mathcal{A}}$是原始递归的。那么 $f^{\mathcal{A}}(free(n))$是关于 n的原始递归函数。$f^{\mathcal{A}}(free(n))$的数量为 $f(p)$，其中 p是 $free(n)$的数量。因此， $f^{\mathcal{A}}(free(n))$的数量为 $f(3n+1+\sum_{k=1}^{n} a_k)$。

性质 3. 蕴含：对于所有充分大的 n，我们有

$$f\left(3n+1+\sum_{k=1}^{n}a_k\right) < a_{n+1},$$

因此， $f^{\mathcal{A}}(free(n))$ 的数目严格介于 $free(n)$ 和 a_{n+1} 的数目之间，并且根据构造，$f^{\mathcal{A}}(free(n)) = h(\langle n, i\rangle)$ 对某个 $i \le a_{n+1}$ 成立。但 $h(\langle n, i\rangle)$ 的数目是 $3n+i+2+\sum_{k=1}^{n}a_k$，因此：

$$3n+i+2+\sum_{k=1}^{n}a_k = f\left(3n+1+\sum_{k=1}^{n}a_k\right) \ge 2\left(3n+1+\sum_{k=1}^{n}a_k\right),$$

这显然意味着，对于所有充分大的 n，不等式 $i \ge a_n$ 成立。但这产生了矛盾，因为可以利用等式 $f^{\mathcal{A}}(free(n)) = h(\langle n, i\rangle)$，从 n 通过原始递归计算出 i。□

作为推论，唯一使得 $p^{\mathcal{A}}$ 为原始递归的非恒定一元多项式 p，是形如 $p(x) = x + b$ 的那些（对于这样的 p， $p^{\mathcal{A}}(x) = (S^{\mathcal{A}})^b(x)$）。

似乎很少有原始递归函数 f 在 $\mathcal{A}$上仍保持原始递归。但只要同构 c 是原始递归的，那么对于任意原始递归的 f，函数 $f \circ c$ 在 $\mathcal{A}$ 上仍保持原始递归，因为$(f \circ c)^{\mathcal{A}} = c \circ f \circ c \circ c^{-1} = c \circ f$。因此从某种意义上说，在那些在 $\mathcal{A}$上仍保持原始递归的函数之中，驻留着全部原始递归函数的一个完整副本。但从另一种意义上说，这些函数相当不自然（例如与多项式相比），因为它们的定义关键依赖于同构 c。

我们已经看到了关于原始递归 f 的例子，其中 $f^{\mathcal{A}}$ 仍保持原始递归，或者变得不再是原始递归的。在下一个命题中，我们将看到一族例子，其中 f 不是原始递归的，但 $f^{\mathcal{A}}$ 变为原始递归的。

命题17. 对于任意原始递归的 $v\colon \mathbb{N} \to \mathbb{N}$，使得 $v(x) > x$，存在 $u\colon \mathbb{N} \to \mathbb{N}$，使得 $u \equiv_E v$、 $u = f^{\mathcal{A}}$，且 f 不是原始递归的。

证明。 证明的思路是：在标准模型中，从 F 中的一个元素原始递归地跳到 F中的下一个元素很容易；但在模型 $\mathcal{A}$ 中， F 中的元素变得零散，因此无法再进行这些跳跃。

给定一个满足 $v(x) > x$的原始递归 v，我们按如下方式定义 u：

$$u(x) = \begin{cases} h(\langle m, v(i)\rangle), & \text{if } x = h(\langle m, i\rangle), \\ 0, & \text{otherwise.} \end{cases}$$

由于 h 是初等的，我们有 $u \le_E v$，也有 $v \le_E u$，因为可以由 $u(h(\langle 0, i\rangle)) = h(\langle 0, v(i)\rangle)$计算出 $v(i)$。当然，u是原始递归的，因此我们取 $f = c^{-1} \circ u \circ c$，使得 $f^{\mathcal{A}} = u$。假设f 是原始递归的。我们将展示一种原始递归算法，可以从 a_s计算出 a_{s+1}，但这由性质3可知不可能。

给定输入 a_s:
1. 计算 y，使得 $c(y) = free(s)$。
2. 计算 $f(y)$ 并给出输出 a_{s+1}。
我们可以在步骤 1 中计算 y，因为 $y = 3s+1+\sum_{k=1}^{s}a_k$，并且通过 a 的图，我们可以获得所有 $k \le s$ 对应的 a_k。设 $free(s) = h(\langle m, i\rangle)$。由于 $v(i) > i$，且 $h(\langle m, i\rangle) \in F$，因此我们有 $u(h(\langle m, i\rangle)) = h(\langle m, v(i)\rangle) \in F$。所以，对于 $t \ge s+1$， $u(c(y)) = u(free(s)) = free(t)$，由此我们得出，$u(c(y))$ 的数量至少与 $free(s+1)$ 的数量一样多，也就是说 $f(y) = c^{-1}(u(c(y))) \ge 3s+4+\sum_{k=1}^{s+1}a_k$。特别地， $a_{s+1} \le f(y)$，我们可以在步骤 2 中使用值 $f(y)$ 和 a 的图来计算 a_{s+1}。□

事实证明，对于函数 u，还有一些采用更自然选择的类似例子。

命题18。 设 g 为函数，且 $g^{\mathcal{A}}$ 是前驱函数。那么 g 不是原始递归的。

证明。 假设 g 是原始递归的。与前一个命题中一样，我们将证明 $a_s \mapsto a_{s+1}$ 是原始递归的，从而导出矛盾。

We 可以假设 s 足够大，使得 $free(s+1) < a_s$ 且 $h(\langle s,\ a_s\rangle) < a_{s+1}$.

给定输入 a_s:
尝试 t he valUes $x = free(s+1)+1,\ free(s+1)+2,\ \ldots,\ h(\langle s, free(s+1)+1\rangle)$ 连续的 l
y: 1. 若 $x \notin Ran(h)$，则终止。2. 若 $x \in Ran(h)$，计算 m, i，使得 $x = h(\langle m, i\rangle)$。
2.1. 若 $x \notin F$ 且 $m \leq s$，则终止。2.2. 否则， $x = x+1$ 并继续到 1。3. 计算 x 的个数 p，
然后利用 $g(p)$ 计算 a_{s+1}。

首先注意到，对 x 的搜索总是会终止，因为最后一个值 $x = h(\langle s, free(s+1)+1\rangle)$ 满足 2.1 中的条件。

考虑终止步骤时 x 的值。若 $x \notin Ran(h)$，则 $x = a_k$，其中 $k \leq s$ （上述不等式表明 $k > s$ 不可能）。若 $x \in Ran(h)$，则 $x = h(\langle m, i\rangle)$，其中 $m \leq s$ 且 $i \leq a_{m+1}$。在这两种情形下，我们都能计算 x 的个数 p，因为我们可以访问 $a_0, a_1, \ldots, a_s$。现在考虑上一步的数 $x-1$。要么 $x-1 = free(s+1)$（在第一步终止），要么 $x-1 \in F$，要么 $x-1 = h(\langle m, i\rangle)$，其中 $m \geq s+1$ （在后续步骤终止）。 x 的值只增不减，因此对 $t \geq s+1$ 有 $x-1 = free(t)$，或者 $x-1 = h(\langle m, i\rangle)$，其中 $m \geq s+1$ 且 $i \leq a_{m+1}$。在这两种情形下， $x-1$ 的个数显然大于 a_{s+1}。根据 g 的定义， $x-1$ 的个数是 $g(p)$，因此我们可以利用 $g(p)$ 以及 a 的图来计算 a_{s+1}。□

3.2. **探究模型中关系的像** $\mathcal{A}$。设 R 为一元关系。其像 $R^{\mathcal{A}}$ 定义为： $R(x) \Leftrightarrow R^{\mathcal{A}}(c(x))$，或等价地， $R^{\mathcal{A}}(x) \Leftrightarrow R(c^{-1}(x))$。

引理 19。标准序的像 $<^{\mathcal{A}}$ 是原始递归的。

证明。给定 $x,\ y$，我们要判定 $c^{-1}(x) < c^{-1}(y)$ 还是 $c^{-1}(x) > c^{-1}(y)$（当然， $c^{-1}(x) = c^{-1}(y)$ 当且仅当 $x = y$）。我们不能使用 c^{-1}，但可以计算 m，使得 x 位于模型 $\mathcal{A}$ 中介于 a_m 与 a_{m+1} 之间的部分（以及对应于 y的类似 n ）。为此，我们首先描述原始递归函数 pos:

给定输入 x: 1. 若 $\exists\, m \leq x$ 且 $a_m = x$，则令 $i = 0$。2. 若 $\exists\, m \leq x$
且 $free(m) = x$，则令 $i = 1$。3. 否则，计算 $m,\ i'$，使得 $x = h(\langle m,$
$i'\rangle)$，并令 $i = i'+2$。4. 返回答案 $pos(x) = \langle m,\ i\rangle$。

现在给定 $x,\ y$我们计算 $pos(x) = \langle m,\ i\rangle$ 以及 $pos(y) = \langle n,\ j\rangle$。若
$m < n \vee (m = n\ \&\ i < j)$，则 $x <^{\mathcal{A}} y$。若 $m > n \vee (m = n\ \&\ i > j)$，
则 $y <^{\mathcal{A}} x$。 □

显然， $R \leq_{PR} R^{\mathcal{A}}$。因此，若 $R^{\mathcal{A}}$ 是原始递归的，则 R 也原始递归。此外，对于任意原始递归的 S，都存在一个原始递归的 R 使得 $R^{\mathcal{A}} = S$。事实上，我们可以取$R(x) \Leftrightarrow S(c(x))$。

模型 $\mathcal{A}$ 中，原始递归关系的一个子类在像下构成的类，正是所有原始递归关系组成的类。我们将通过给出一些例子来说明这个子类是真子类：在这些例子中， R 是原始递归的，但 $R^{\mathcal{A}}$ 不是。

将 a_n 替换为 max($a_n,\ A(n,\ n)$) 后，我们可以假设对所有 n，都有 $A(n,\ n) \leq a_n$。
这意味着我们可以选择一个原始递归函数 ϕ，使得 $\psi_n(n) = \phi(a_n)$ 对所有 n（回忆一下， ψ 是定义于第2节中的一元原始递归函数的枚举）。
现在考虑关系 S，其定义如下算法 .
给定输入 x: 1. 计算满足 $a_k \leq c^{-1}(x)$ 的最大 a_k。2. 如果 ϕ
$(a_k) = 0$，则返回输出 1；如果 $\phi(a_k) \neq 0$，则返回输出 0。

命题 20。由 $R(x) \Leftrightarrow S(c(x))$ 定义的关系 R 是原始递归的，但关系 $R^{\mathcal{A}} = S$ 不是原始递归的。

证明。 首先我们论证 R 是原始递归的。给定输入 $c(x)$，输入到 S，利用 a 的图，我们可以确定满足 $a_k \leq c^{-1}(c(x)) = x$ 的最大 a_k，然后使用原始递归函数 ϕ 返回输出。

显然， $R^{\mathcal{A}} = S$，因此只需证明 S 不是原始递归的。

给定 n，我们可以取输入 $x = h(\langle n,\ 0\rangle)$。于是， $c^{-1}(x)$是 x的数量，也就是说 $c^{-1}(x) = 3n + 2 + \sum_{k=1}^{n} a_k$。对于足够大的 n，满足 $a_k \leq c^{-1}(x)$的最大 a_k 是 a_n。因此， $\phi(a_k) = \phi(a_n) = \psi_n(n)$。我们得到，对于几乎所有的 n， $S(h(\langle n,\ 0\rangle))$都会返回输出，而这与 $\psi_n(n)$不同，因此 S不是原始递归的。□

我们可以用同样的思路得到许多其他例子。

对于任意原始递归函数 f，若 $f(x) \geq x$，令关系 R_f定义为$R_f(x) \Leftrightarrow S(c(f(x)))$。基本相同的证明表明， R_f是原始递归的，而 $R_f^{\mathcal{A}}$不是原始递归的。给定输入 $x = h(\langle n,\ 0\rangle)$， $R_f^{\mathcal{A}}(x) \Leftrightarrow S(y)$，其中 $y = c(f(c^{-1}(x)))$。因此，对于足够大的 n，满足 $=f(3n + 2 + \sum_{k=1}^{n} a_k)$的最大 $a_k \leq c^{-1}(y)$仍然是 a_n。

3.3. **确保线性函数的像是原始递归的**。 接下来，自然要构造一个非标准精确副本 $\mathcal{D}$，使得倍增函数的像是原始递归的。我们取点态副本 $\mathcal{A}$，将其定义域替换为偶数，并把它的第 n 个元素放在新副本 $\mathcal{D}$ 的位置 2^n。这样一来， $\mathcal{D}$ 中倍增函数在 $\mathcal{A}$ 的元素上的限制，其像就与 $\mathcal{A}$ 中后继的像一致。此外，我们必须设法用奇数填满所有其他位置，使得倍增函数在这些位置上的像也是原始递归的。

更具体地说：我们取模型 $\mathcal{A} = (\mathbb{N},\ S^{\mathcal{A}})$，并将 $\mathbb{N}$ 替换为 $2\mathbb{N}$，使奇数空出来。取两个元素 $2x$ 和 $2y$，使得 $S^{\mathcal{A}}(x) = y$，并假设我们已经插入了位于 $2x$ 之前的元素，因此 $2x$ 的新数值为 p'，其中 $p' \geq 2$。然后，我们在 $2x$ 和 $2y$ 之间插入以下奇数：

$$2x \ \to\ odd'(x) \ \to\ 2\langle x, 0\rangle + 1 \ \to\ 2\langle x, 1\rangle + 1 \ \to\ \ldots \ \to\ 2\langle x, k_x - 1\rangle + 1 \ \to\ 2y,$$

使得新的自然数集 $2y$ 的数目为 $2p'$，而 odd' 是集合 $\{2\langle x, i\rangle + 1 \mid i \geq k_x\}$ 的严格递增枚举。我们称这个新的副本为 $\mathcal{D} = (\mathbb{N},\ S^{\mathcal{D}})$。它的开头如下：

$$2c(0) \ \to\ odd'(c(0)) \ \to\ 2c(1) \ \to\ odd'(c(1)) \ \to$$
$$2c(2) \ \to\ odd'(c(2)) \ \to\ 2\langle c(2), 0\rangle + 1 \ \to\ 2\langle c(2), 1\rangle + 1 \ \to\ 2c(3) \ \to\ \ldots$$

我们有 $k_{c(0)} = 0$，并且对于 $p \geq 1$， $k_{c(p)} = 2^p - 2$。 $2x$ 的新数 p' 是 2^p，其中 $p = c^{-1}(x)$ 是 x 的旧数。因此，关系 $k_x = i$ 是原始递归的： $k_x = i \Leftrightarrow \exists p \leq i + 1\ [x = c(p)\ \&\ \max(2^p,\ 2) = i + 2]$。

所以，集合 $Ran(odd')$ 及其枚举 odd' 也是原始递归的。

命题 21。 后继 $S^{\mathcal{D}}$ 是原始递归的。

证明。 该算法是对模型 $\mathcal{D}$ 的定义进行的直接形式化。我们将用到 $\mathcal{S}^{\mathcal{A}}$ 是原始递归的这一事实，以及关系 $k_x = i$ 是原始递归的这一事实；这意味着关系 $k_x < i$ 和 $k_x > i$ 也都是原始递归的。

给定输入 n：1. 如果存在 $t \leq n$，使得 $2t + 1 = n$：计算 x, i，使得 $t = \langle x, i\rangle$。1.1. 如果 $k_x > 0\ \&\ i + 1 = k_x$，则 $S^{\mathcal{D}}(n) = 2S^{\mathcal{A}}(x)$。1.2. 如果 $k_x > 0\ \&\ i + 1 < k_x$，则 $S^{\mathcal{D}}(n) = 2\langle x, i + 1\rangle + 1$。1.3. 如果 $k_x = 0$ 或 $(k_x > 0\ \&\ i + 1 > k_x)$，则：计算 m，使得 $odd'(m) = n$。1.3.1. 如果 $k_m = 0$，则 $S^{\mathcal{D}}(n) = 2S^{\mathcal{A}}(m)$。1.3.2. 否则， $S^{\mathcal{D}}(n) = 2\langle m, 0\rangle + 1$。2. 否则， $S^{\mathcal{D}}(n) = odd'(\lfloor \frac{n}{2} \rfloor)$。

– □

命题 22. 函数 $d(x) = 2x$ 的像 $d^{\mathcal{D}}$ 是原始递归的。

证明。 思路如下：如果输入是 $2x$，那么我们可以利用旧后继 $S^{\mathcal{A}}$跳到 $d^{\mathcal{D}}(2x) = 2y$ 。如果输入是 $odd'(x)$ 或 $2\langle x, i\rangle + 1$，我们可以计算出 x，像第一种情况那样跳到 $2y$，然后相应地向右移动两次或 $2i + 4$ 次。这一点至关重要地依赖于差值$d(p + 1) - d(p)$为常数这一事实。

给定输入 n:
1. 若存在 $t \le n$，使得 $2t + 1 = n$：计算 x，i，使得 $t = \langle x, i\rangle$。
1.1. 若 ($k_x > 0$ & $i + 1 \le k_x$)，则 $d^{\mathcal{D}}(n) = 2\langle 2S^{\mathcal{A}}(x)\ 2i + 2\rangle + 1$。
1.2. 否则：计算 m，使得 $odd'(m) = n$ 且 $y = S^{\mathcal{A}}(m)$。若 $k_y > 0$，则 $d^{\mathcal{D}}(n) = 2\langle y,\ 0\rangle + 1$，否则 $d^{\mathcal{D}}(n) = 2y$。2. 否则：若 $n = 2c(0)$，则 $d^{\mathcal{D}}(n) = n$，否则 $d^{\mathcal{D}}(n) = 2S^{\mathcal{A}}(\lfloor \frac{n}{2} \rfloor)$。

– □

命题23. 在模型 $\mathcal{D} = (\mathbb{N}, S^{\mathcal{D}})$中，像为原始递归的线性函数仅有形如 $2^a x + b$ 的函数。

证明. 显然，函数 $\lambda x.2^a x + b$ 的像是 $\lambda x.((S^{\mathcal{D}})^b(d^{\mathcal{D}})^a(x))$，因此它是原始递归的。现在考虑线性函数 $f(x) = mx + b$，其中 $2^a < m < 2^{a+1}$ 对某个 a 成立。我们将证明 $c^{-1} \le_{PR} f^{\mathcal{D}}$，因此 $f^{\mathcal{D}}$ 不可能是原始递归的。

给定输入 x:
1. 如果 $c^{-1}(x) \le b$，则用表给出答案。 2. 否则计算 $f^{\mathcal{D}}(2x)$，然后计算 x'，i，使得 $f^{\mathcal{D}}(2x) = 2\langle x',\ i\rangle + 1$。 3. 搜索满足 $c(p) = x$ 的 $p \le i + 1$，并给出答案 p。

令 $p = c^{-1}(x)$ 为 x 的 $\mathcal{A}$-数。在步骤 2 中，$p > b$，且 $2x$ 的 $\mathcal{D}$-数为 $p' = 2^p \ge 2$，因此 $f^{\mathcal{D}}(2x)$ 的 $\mathcal{D}$-数为 $mp' + b$。现在我们有（这些是相应元素的 $\mathcal{D}$-数）

$$2^a p' + 2 \ \le\ (m-1)p' + 2 \ \le\ mp' \ \le\ mp' + b \ <\ mp' + p \ <\ mp' + p' \ \le\ 2^{a+1}p'.$$

根据模型 $\mathcal{D}$ 的构造，我们有 $f^{\mathcal{D}}(2x) = 2\langle x',\ i\rangle + 1$，其中 $i = (mp' + b) - (2^a p' + 2) \ge p' - 2$（$2x'$ 的 $\mathcal{D}$-数为 $2^a p'$，但我们不需要这一事实）。我们得到 $p' \le i + 2$，也就是 $2^p \le i + 2$，这意味着 $p \le i + 1$，并且第 3 步中的搜索成功。
□

命题 24。 令 $f:\ \mathbb{N} \to \mathbb{N}$ 为一个原始递归函数，使得对于所有充分大的 x，$f(x) \ge x^2$。那么 $f^{\mathcal{D}}$ 不是原始递归的。

证明。 几乎对所有 x，令 $f(x) \ge x^2$，并假设 f 和 $f^{\mathcal{D}}$ 是原始递归的。

我们将定义一个新函数 $g^{\mathcal{A}}$，然后论证它的原像 g 是原始递归的，并且$g(p) \ge 2p$ 对几乎所有 p都成立；根据命题 16，这是不可能的。

为了计算 $g^{\mathcal{A}}(x)$：计算 $f^{\mathcal{D}}(2x)$，并给出答案 y，其中 $2y$ 是最小偶数元素，其 $\mathcal{D}$ 数大于 $f^{\mathcal{D}}(2x)$ 的 $\mathcal{D}$ 数。

注意，$f^{\mathcal{D}}(2x)$ 的形式为 $2x'$、$odd(x')$ 或 $2\langle x',\ i\rangle + 1$，其中某个 x'，i；因此 $y = S^{\mathcal{A}}(x')$，从而 $g^{\mathcal{A}}$ 是原始递归的。

如果 x 的 $\mathcal{A}$ 数 p 足够大，那么 $2x$ 的 $\mathcal{D}$ 数为 2^p，因此 $f^{\mathcal{D}}(2x)$ 的 $\mathcal{D}$ 数为 $f(2^p) \ge 2^{2p}$，所以 $2y$ 的 $\mathcal{D}$ 数也大于 2^{2p}。由此可知，y 的 $\mathcal{A}$ 数大于 $2p$，也就是说 $g(p) \ge 2p$。

此外，y 的 $\mathcal{A}$-数 $g(p)$可以由 $f^{\mathcal{D}}(2x)$及其 $\mathcal{D}$-数 $f(2^p)$通过原始递归计算得出。由于 f 是原始递归的，g 也具有原始递归性。因此，证明完成。□

3.4. 在模型中强制原始递归加法和乘法。上一节的思路无法用于证明加倍函数和三倍函数的像同时具有原始递归性。例如，在模型 $\mathcal{D}$ 中，位于位置 $3 \cdot 2^n = 2^{n+1} + 2^n$的元素处于 n个元素中的第 n个与第 $(n + 1)$ 个元素之间，因此若不知道 n，就无法访问它。

克服这一问题的一种方法是使用大陆-岛屿构造。这在定理11的证明（第4节）中完成。

为寻求另一种方法，我们观察到，可以不使用原始副本 $\mathcal{A}$中的单个元素，而使用此类元素的列表，从而对位置介于 2^n 与 2^{n+1} 之间的元素施加层级结构。例如，长度为1的列表 $\langle c(n)\rangle$ 对所有 n取得位置 2^n 。要使位置加倍，只需对列表中的元素应用后继 $S^{\mathcal{A}}$ 。要使 $\langle c(n)\rangle$的位置变为三倍，我们将元素 $\langle c(n)\ c(n+1)\rangle$ 放在位置 $3\cdot 2^n$。于是我们逐渐想到，可以取该位置的二进制展开的指数，并在该处放入 $\mathcal{A}$中$\mathcal{A}$-数恰好为这些指数的元素所组成的列表。

这个想法富有成效，因为在所得的副本中，加法函数的像是原始递归的。此外，将这一构造迭代两次，就能得到一个副本，其中乘法函数的像也是原始递归的。

因此，很自然地会期待再迭代一次，就能使指数函数的像成为原始递归的。但事实并非如此：在所得的副本中，函数 2^x 的像是原始递归的，当且仅当它在原始副本中已经是原始递归的。 $\mathcal{A}$

更形式化地说：给定任意点态副本 $\mathcal{A}=(\mathbb{N},S^{\mathcal{A}})$，它是 $(\mathbb{N},S)$ 的副本，且其唯一同构 c 从 $(\mathbb{N},S)$ 到$\mathcal{A}$ ，我们通过以下同构 $\tilde{c}$定义一个新的副本 $\mathcal{A}$：

$$\tilde{c}(0)\ =\ 0,\quad \tilde{c}(2^{i_k}+2^{i_{k-1}}+\ldots+2^{i_0})\ =\ 2^{c(i_k)}+2^{c(i_{k-1})}+\ldots+2^{c(i_0)}.$$

因此， $\mathcal{A}$ 中的个体被视为具有 $\tilde{c}(n)$ 的形式，其中 $n\in\mathbb{N}$ ，而 $\mathcal{A}$-数 $\tilde{c}(n)$ 为 n。因此，后继 $S^{\mathcal{A}}$ 满足 $S^{\mathcal{A}}(\tilde{c}(n))=\tilde{c}(n+1)$。

引理 25。由等式定义的函数 add： $\mathbb{N}\times\mathbb{N}\to\mathbb{N}$

$$add(x,u)\ =\ \tilde{c}(n+2^j),$$

其中 $x=\tilde{c}$（n）和 $u=c$（j）是原始递归的。

证明。显然，函数 add在模型 $\tilde{\mathcal{A}}$ 中实现了 x 与 $2^{c(j)}$ 的加法。证明的思路是在 $\mathcal{A}$ 中模拟二进制加法。我们需要初等关系 bit，使得当且仅当 x 的二进制表示中的第 i 位为 1 时， $bit(x,\ i)=true$ （等价地，2^i 属于 x 的二进制表示）。

给定输入 $x,\ u$:
1. 搜索满足 $bit(x,\ (S^{\mathcal{A}})^k(u))=false$ 的最小 $k\le x$。2. 对于 $i=0,$
1 、 ..... 、 k，令 $u_i=(S^{\mathcal{A}})^i(u)$。3. 返回答案 $x+2^{u_k}-\sum_{i<k}2^{u_i}$。

注意，如果我们将 $S^{\mathcal{A}}$ 替换为标准的 S （在其被使用的所有位置），我们便得到 x 与 2^u的标准加法。事实上，我们可以扫描 x的二进制表示，寻找 u 右侧的第一个0数字，然后将所有已扫描的1改为0。 k 的最大可能值是 x的二进制表示中1的个数，因此 $k\le x$。

由于该算法在使用标准 S时能正确运行，因此它在模型$\mathcal{A}$ 中使用后继 $S^{\mathcal{A}}$时也能正确运行。□

命题 26. 模型 $\tilde{\mathcal{A}}$ 中后继与加法的像 $S^{\tilde{\mathcal{A}}}$ 和 $+^{\tilde{\mathcal{A}}}$ 是原始递归的。

证明。 我们有等式$S^{\mathcal{A}}(x)=add(x,c(0))$，因此 $S^{\mathcal{A}}$ 是原始递归的。至于和 $x+^{\mathcal{A}}y$，我们将 y 表示为如下形式

$$y=2^{u_k}+2^{u_{k-1}}+\ldots+2^{u_0}$$

（清楚地 y $x+^{\mathcal{A}}y=x$ 当 $y=0$）。然后我们可以按以下方式迭代函数 add ：

$$x+^{\tilde{\mathcal{A}}}y\ =\ add(\ldots add(add(x,u_0),u_1)\ldots u_k).$$

迭代次数至多为（ y 的二进制长度）。 □

通过非常类似的分析，可以证明，只要序关系 $<^{\mathcal{A}}$ 是原始递归的，序关系 也 $<^{\mathcal{A}}$ 是原始递归的。

命题 27. 设 $\mathcal{A}$为第 3 节开头构造的模型。那么 $S^{\mathcal{A}}$ 和 $+^{\mathcal{A}}$ 是原始递归的，但对于任意原始递归的 f，只要 $f(x) \geq x^2$ 对所有充分大的 x都成立，其像 $f^{\mathcal{A}}$ 就不是原始递归的。

证明。 由于 $S^{\mathcal{A}}$ 是原始递归的，由命题 26 可知 $S^{\mathcal{A}}$ 与 $+^{\mathcal{A}}$ 是原始递归的。现设 f是原始递归的，使得 $f(x) \geq x^2$对几乎所有 x成立，并假设 $f^{\mathcal{A}}$也是原始递归的。该证明沿用与命题 24中相同的论证。

我们将定义一个新函数 $g^{\mathcal{A}}$，然后论证其原像 g 是原始递归的，并且$g(p) \geq 2p$ 对几乎所有 p成立，而这由命题 16可知是不可能的。

为了计算 $g(p)$，给出答案 $c(n)$，其中 $2^{c(n)}$ 属于 y 的二进制表示，且 n 是极大的。

注意，我们可以写出 $y = 2^{u_k} + 2^{u_{k-1}} + \ldots + 2^{u_0}$，并且可以确定 $c(n)$，使 n 极大，这可利用关系 $<^{\mathcal{A}}$完成；根据引理 19，该关系是原始递归的。因此 $g^{\mathcal{A}}$ 是原始递归的。

如果 x 具有足够大的 $\mathcal{A}$数 p，则 2^x 具有 $\mathcal{A}$数 2^p，因此 $y = f^{\mathcal{A}}(2^x)$具有 $\mathcal{A}$数$f(2^p) \geq 2^{2p}$。因此，使得 $2^{c(n)}$ 属于 y 的二进制表示的最大 n必须大于 $2p$。换言之，$g^{\mathcal{A}}(x) = c(n)$具有 $\mathcal{A}$数 $n \geq 2p$，即 $g(p) \geq 2p$。

此外，$f^{\mathcal{C}}(2^x)$的 $\mathcal{A}$数 $f(2^p)$ 计算得到以由于 f 是原始递归的，g 也是原始递归的。□

现在我们转向乘法。

命题 28. 设原始模型 $\mathcal{A}$ 中加法的像 $+^{\mathcal{A}}$ 是原始递归的。则模型 $\widetilde{\mathcal{A}}$ 中乘法的像 $\cdot^{\widetilde{\mathcal{A}}}$ 也是原始递归的。

证明。 与加法类似，我们首先定义函数 $prod\colon \mathbb{N} \times \mathbb{N} \to \mathbb{N}$，使得 $prod(x,\ u) = \widetilde{c}(n \cdot 2^j)$，其中 $x = \widetilde{c}(n)$ 且 $u = c(j)$。我们表示

$$n = 2^{i_k} + 2^{i_{k-1}} + \ldots + 2^{i_0}.$$

于是 $n \cdot 2^j = 2^{i_k+j} + 2^{i_{k-1}+j} + \ldots + 2^{i_0+j}$ 且

$$\begin{aligned} prod(x,u) &= 2^{c(i_k+j)} + 2^{c(i_{k-1}+j)} + \ldots + 2^{c(i_0+j)} \\ &= 2^{c(i_k)+^{\mathcal{A}}c(j)} + 2^{c(i_{k-1})+^{\mathcal{A}}c(j)} + \ldots + 2^{c(i_0)+^{\mathcal{A}}c(j)} \end{aligned}$$

当然，这意味着可以利用 $+^{\mathcal{A}}$ 是原始递归函数这一事实，从 $x = 2^{c(i_k)} + 2^{c(i_{k-1})} + \ldots + 2^{c(i_0)}$ 和 $u = c(j)$ 原始递归地计算 $prod(x,\ u)$。

现在，定义函数 f（它需要是原始递归的）。令$y = \widetilde{2^{u_k} + 2^{u_{k-1}} + \ldots + 2^{u_0}}$。于是

$$x \cdot^{\widetilde{\mathcal{A}}} y \ = \ prod(x, u_0) +^{\widetilde{\mathcal{A}}} prod(x, u_1) +^{\widetilde{\mathcal{A}}} \ldots +^{\widetilde{\mathcal{A}}} prod(x, u_k)$$

且显然 $x \cdot^{\mathcal{A}} 0 = 0$。 □

接下来的命题中，我们将把该构造迭代两次。

命题 29。 设 $\mathcal{A}$为在第 3 节开头构造的模型。那么 $+^{\mathcal{A}}$ 和 $\cdot^{\mathcal{A}}$ 都是原始递归的，但对于任意原始递归的 f，若满足 $f(x) \geq x^{\log_2 x}$对所有充分大的 x都成立，则其像 $f^{\mathcal{A}}$不是原始递归的。

证明。 记 $\mathcal{B} = \widetilde{\mathcal{A}}$ 和 $\mathcal{C} = \widetilde{\mathcal{B}} = \widetilde{A}$。我们知道 $S^{\mathcal{B}}$ 和 $+^{\mathcal{B}}$ 都是原始递归的，因此由命题 26 和 28可知，$S^{\mathcal{C}}$、$+^{\mathcal{C}}$ 和 $\cdot^{\mathcal{C}}$ 都是原始递归的。

现在设 f 是原始递归的，且对几乎所有 x，都有 $f(x) \geq x^{\log_2 x}$，并假设 $f^{\mathcal{C}}$ 也是原始递归的。

我们将定义 $g^{\mathcal{B}}$，然后说明它的原像 g 是原始递归的，而 $g(p) \geq p^2$，这与命题 27矛盾。

计算 $g^{\mathcal{B}}(x)$：取 $y = f^{\mathcal{C}}(2^x)$ 并给出答案 $\widetilde{c}(i_k + \ldots + i_0)$，其中 $y = 2^{\widetilde{c}(i_k)} + \ldots 2^{\widetilde{c}(i_0)}$.

注意到 $\widetilde{c}(i_k + \ldots + i_0) = \widetilde{c}(i_k) +^{\mathcal{B}} \ldots +^{\mathcal{B}} \widetilde{c}(i_k)$，它可以根据 y 的二进制表示和原始递归函数 $+^{\mathcal{B}}$计算得出。因此，$g^{\mathcal{B}}$ 是原始递归的。

如果 x 有足够大的 $\mathcal{B}$-数 p，那么 2^x 有 $\mathcal{C}$-数 2^p，因此 $y = f^{\mathcal{C}}(2^x)$ 有 $\mathcal{C}$-数$f(2^p) \geq 2^{p^2}$。所以， $f(2^p)$的二进制展开中指数的和 $i_k + \ldots + i_0$大于 p^2。换句话说， $g^{\mathcal{B}}(x) = \widetilde{c}(i_k + \ldots + i_0)$ 有 $\mathcal{B}$-数 $i_k + \ldots + i_0 \geq p^2$，也就是说 $g(p) \geq p^2$。

此外， $\mathcal{B}$-数 $g(p) = i_k + \ldots + i_0$ 可以从 $\mathcal{C}$-数 $f(2^p)$ of$y = f^{\mathcal{C}}(2^x)$ 直接计算出来。由于 f 是原始递归的， g 也是原始递归的。□

We 最后证明同样的思路不适用于乘方运算，从而结束本节 .

命题 30. 设原始模型中加法的像 $+^{\mathcal{A}}$ 是原始递归的。设 $p(x) = 2^x$.~那么， $p^{\mathcal{A}}$ 是原始递归的，当且仅当 $p^{\mathcal{A}}$ 是原始递归的。

证明。设 $p^{\widetilde{\mathcal{A}}}$ 是原始递归的。我们有等式 $p^{\widetilde{\mathcal{A}}}(2^{c(x)}) = 2^{c(2^x)}$，因为 $\widetilde{\mathcal{A}}$-数 of$2^{c(x)}$ 是 2^x ，而 $\widetilde{\mathcal{A}}$-数 of $2^{c(2^x)}$ 是 $p(2^x)$。因此，给定 $c(x)$，我们可以用原始递归方式计算 $c(2^x)$，这意味着 $p^{\mathcal{A}}$ 是原始递归的。

反过来，设 $p^{\mathcal{A}}$ 是原始递归的。那么对于 $x = 2^{c(i_k)} + \ldots + 2^{c(i_0)}$，我们有

$$\begin{aligned} p^{\widetilde{\mathcal{A}}}(x) &= 2^{c(2^{i_k} + \ldots + 2^{i_0})} \\ &= 2^{\left(c(2^{i_k}) +^{\mathcal{A}} \ldots +^{\mathcal{A}} c(2^{i_0})\right)} \\ &= 2^{\left(p^{\mathcal{A}}(c(i_k)) +^{\mathcal{A}} \ldots +^{\mathcal{A}} p^{\mathcal{A}}(c(i_0))\right)}. \end{aligned}$$

我们可以根据 x 的二进制展开计算 $c(i_k) \ldots, c(i_0)$，而且由于 $+^{\mathcal{A}}$ 和 $p^{\mathcal{A}}$ 是原始递归的，因此 $p^{\mathcal{A}}$ 也是原始递归的。□

4. Mainland–Island Technique and the Metatheorem

4.1. 展示大陆–岛屿技巧。

定理 11. 对于任意有限素数集 P，都存在 $\mathcal{B}$—— $\mathcal{S}$ 的一个态副本——使得在 $\mathcal{B}$ 上原始递归的线性函数只有 $f(x) = ax + b$，其中 a 在 P 之外没有素因子。

证明。为简单起见，这里给出针对素数集 $\{2, 3\}$ 的证明。推广到任意有限素数集很容易——稍后我们会简要讨论。

我们构造一个准时模型 $\mathcal{B}$ 和一个同构 $c : (\mathbb{N}, S) \to \mathcal{B}$。在每个阶段， $\mathcal{B}$ 中的每个元素都有唯一的标记——一个线性（可能为常值）函数，形式为 $dx + e$，其中 $d = 0$ 或 $d = 2^u 3^v$，对某些 u、 v、 $e \in \mathbb{N}$ 成立。 c 当前的定义域是标记为常函数的元素集合；这些元素始终构成 $\mathbb{N}$ 的一个初始段。我们保持结构 $(range(c)\ S^{\mathcal{B}})$ 为一个称作大陆的$S^{\mathcal{B}}$-链，并且 c 是从 $(dom(c)\ S)$ 到 $(range(c)\ S^{\mathcal{B}})$ 的唯一同构。 $\mathcal{B} \setminus range(c)$ 中的元素排列在互不相交的 $S^{\mathcal{B}}$-链中（参见图1），称为岛。岛的初始元素始终以 $2^u 3^v x$ 为标记，其中 u、 $v \in \mathbb{N}$ 成立。

令 R_0、 R_1、.....为所有三元组编码 $\langle a、\ b、\ n\rangle$ 的原始递归枚举，使得 a 有一个素因子 $p \notin \{2, 3\}$，且 b、 $n \in \mathbb{N}$。每个 $R_i = \langle a、\ b、\ n\rangle$ 表示如下要求：$(ax + b)^{\mathcal{B}} \neq \psi_i(x)$，其中 ψ_i 是根据某个固定的此类函数的可计算枚举得到的第 i 个原始递归一元函数。我们将逐一满足这些要求。每个要求依次经历以下状态：未激活、激活和已满足。当 $R_i = \langle a、\ b、\ n\rangle$ 激活时，会为其关联一个新的见证 w，它将在 $\mathcal{B}$ 中开辟自己的岛。目标是使 $(ax + b)^{\mathcal{B}}(w) \neq \psi_n(w)$。我们等待一个阶段 s，使得 $\psi_{n、\ s}(w)\downarrow$；每次都迅速扩展当前结构，使 S、 $2x$、 $3x$ 在 $\mathcal{B}$ 中为原始递归。这些扩展会产生不断增多的岛。一旦找到这样的 s，我们就运行“连接”过程，通过适当合并所有岛与大陆来满足 R_i。详情如下。

当我们向 $\mathcal{B}$ 中添加新元素时，它总是当前不在 $\mathcal{B}$ 中的最小数（即所谓的新鲜数）。

构造。在阶段 0， $\mathcal{B}$ 由带有标记 0 的 0 组成， $S^{\mathcal{B}} = \emptyset$，并且 $c(0) = 0$，因此大陆只包含一个元素 0。所有要求均未激活。

令 $s > 0$。首先，我们运行 *Extend* （见下文），并得到 $\mathcal{B}'$。

如果没有任何要求处于激活状态，就令 i 为使 R_i 处于未激活状态的最小数。 R_i 变为激活状态。我们向 $\mathcal{B}'$ 中添加一个新鲜数 w。 w 成为 R_i 的见证，并获得标记 x。我们令 $\mathcal{B}_s = \mathcal{B}'$，然后进入下一阶段。

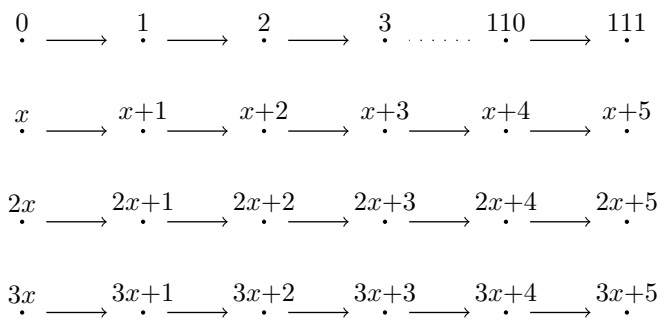


FIGURE 1. 大陆（顶部链）和三个带有相应标记的岛的示意图。箭头对应于 $S^{\mathcal{B}}$。

如果有一项要求处于激活状态，就令 $R_i = \langle a、\ b、\ n\rangle$ 为该要求，并令 w 为它的见证。如果 $\psi_{n、\ s}(w) \uparrow$，我们就令 $\mathcal{B}_s = \mathcal{B}'$，然后进入下一阶段。否则，我们执行 连接（$\mathcal{B}'、\ \psi_{n、\ s}(w)$）。这会产生一个变为 $\mathcal{B}_s$ 的模型。我们进入下一阶段。

Extend。给定 $\mathcal{B}$，c，$M(\mathcal{B})$。我们需要 $\mathcal{B}' \supset \mathcal{B}$、$c' \supset c$ 和 $M(\mathcal{B}')$，使其满足以下性质：

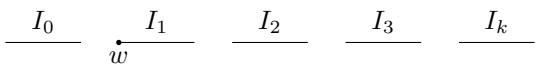
1. $M(\mathcal{B}') \supset M(\mathcal{B})$；2. 若 $dx + e \in M(\mathcal{B})$，则 $dx + e + 1$，$2(dx + e)$，$3(dx + e)$ $\in M(\mathcal{B}')$ 3. 若 $dx + e$，$dx + e' \in M(\mathcal{B}')$ 且 $e < e'$，则 $dx + e'' \in M(\mathcal{B}')$，对于每个 满足 $e < e'' < e'$ 的 $e'' \in \mathbb{N}$。

我们取满足上述条件的最小标记集合——这就是 $M(\mathcal{B}')$。我们将 $\mathcal{B}$ 扩展为 $\mathcal{B}'$，使得 $|\mathcal{B}'| = |M(\mathcal{B}')|$，并规定对于每个 b，$b' \in \mathcal{B}'$, $S^{\mathcal{B}'}(b) = b'$ 当且仅当，对于某个 d，$e \in \mathbb{N}$，b的标记是$dx + e$，且 b' 的标记是 $dx + e + 1$。

Connect($\mathcal{B}'$，m)。。假设当前要求已激活了 t 个阶段 s_0，s_1，.....，s_t，其中 s_t是当前阶段。$\mathcal{B}'$中的岛的数量至多为 $\Sigma_{j=0}^{t}(j+1)$，因为在阶段 s_j，我们会添加带有标记 $2^u 3^v x$ 的新岛，使得 $u + v = j$（参见 *Extend*）。显然，这为任意给定阶段的岛的数量提供了一个统一的原始递归上界。设 $\mathcal{B}'$ 中的岛的数量为 k。

使用原始递归算法将这些岛排列成线性序 I_1，I_2，.. . ，I_k，规则如下：标记为 dx 的岛先于标记为 $d'x$ 的岛，当且仅当 $d < d'$。令 I_0 为大陆。

我们的目标是满足 R_i，方法是谨慎地将所有岛连接起来



并连接到同一块大陆



使各岛互不重叠，且 $ac^{-1}(w) + b \neq c^{-1}(m)$。

我们需要找到 $c^{-1}(w)$的合适取值。为简便起见，将其记为 q。找到 q 后，我们就可以执行下文段落中所述的步骤。

Finish(q)。我们扩展 $S^{\mathcal{B}'}$，并在必要时扩展其定义域，使得 $[0, |\mathcal{B}'|)$ 是 $\mathbb{N}$ 的最小初始段，并且 $\mathcal{B}'$ 是一个单一的 $S^{\mathcal{B}'}$-链；此外，对于每个 $dx + e \in M(\mathcal{B}')$，$c$ 都将 $dq + e$ 映射到标记为 $dx + e$的元素，其中 c 是从 ($[0, |\mathcal{B}'|)$ S) 到 $\mathcal{B}'$的唯一同构。注意，扩展后的 c 将 q 映射到 w，因为 w的标记为 x。我们称 $\mathcal{B}_s$ 为所得结构，c 为上文所述的唯一同构，并且$M(\mathcal{B}_s) = dom(c)$，也就是说，每个元素的标记都是一个常值函数。

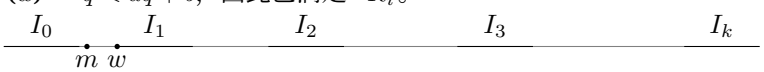
让我们找到 q。令 $d_i x$ 和 $d_i x + e_i$ 分别为 I_i 的第一个和最后一个元素的标记。

设 $q' := |I_0|$。对 $i = 1,2, \ldots, k-1$ 执行有界循环： $q' := max\{q', \lceil \frac{e_i}{d_{i+1} - d_i} \rceil + 1\}$。 $\frac{e_i}{d_{i+1} - d_i}$ 正是 $d_i x + e_i = d_{i+1} x$ 的解。如果我们执行 $Finish(q')$，就能确保各岛不会重叠，因为对于 $i = 1, 2, \ldots, k-1$，q' 满足不等式组 $d_i x + e_i < d_{i+1} x$。为了满足 R_i，在某些情况下我们可能需要更改 q'。

1. 情况 $m \in I_0$。执行 $Finish(q')$。由于 $c^{-1}(m) < c^{-1}(w) = q < aq +$，因此已满足 R_i b.



2. 情况 $m \notin I_0 \cup I_1 \cup \cdots \cup I_k$。设置 $c(|I_0|) = m$ 和 $q = max(q', |I_0|)$。调用 $Finish(q)$。由于 $c^{-1}(m) < c^{-1}(w) = q < aq + b$，因此已满足 R_i。



3. 情形 $m \in I_j, 1 \leq j \leq k, a < d_j$。令 m 的标记为 $d_j x + e$。计算 $q = max(q', \lceil \frac{b-e}{d_j - a} \rceil + 1)$。注意，$\lceil \frac{b-e}{d_j-a} \rceil$ 是满足 ~~$ax +$~~ $b \leq d_j x + e$ 的最小整数。称为 $Finish(q)$。 R_i 已满足，因为 $aq + b < d_j q + e = c^{-1}(m)$。

4. 情形 $m \in I_j, 1 \leq j \leq k, a > d_j$。令 m 的标记为 $d_j x + e$。计算 $q = max(q', \lceil \frac{e-b}{a-d_j} \rceil + 1)$。注意，$\lceil \frac{e-b}{a-d_j} \rceil$ 是满足 ~~$ax +$~~ $b \geq d_j x + e$ 的最小整数。称为 $Finish(q)$。 R_i 已满足，因为 $aq + b > d_j q + e = c^{-1}(m)$。

请注意，$m \in I_j$，$1 \leq j \leq k$，$a = d_j$ 这种情况是不可能的，因为 a 有一个素因子 $p \notin \{2, 3\}$，而 $d_j = 2^u 3^v$，其中某个 u，$v \in \mathbb{N}$。

验证。我们来说明 $\mathcal{B}$ 是一个及时的结构。显然，$\mathbb{N}$ 是 $\mathcal{B}$ 的定义域，因为在每个阶段，我们都会通过加入当前 $\mathcal{B}$ 中不存在的最小数，适当地扩展当前的 $\mathcal{B}$。要计算 $S^{\mathcal{B}}(a)$，其中 $a \in \mathbb{N}$，我们需要计算 $\mathcal{B}_{a+1}$，因为 a 已存在于 $\mathcal{B}_a$ 中；如果 $S^{\mathcal{B}}(a)$ 在 $\mathcal{B}_a$ 中未定义，它就会在 $\mathcal{B}_{a+1}$ 中被定义（参见 $Extend$）。 $\mathcal{B}_{a+1}$ 可以通过一个以 $a+1$ 为界的循环来计算，该循环依次计算 $\mathcal{B}_0$，$\mathcal{B}_1$、……、$\mathcal{B}_{a+1}$。我们已经在构造部分解释过，在给定阶段执行的运算由原始递归算法完成。

以类似的方式，我们论证 $2x$ 和 $3x$ 在 $\mathcal{B}$ 中的像是 p.r.。令 $a \in \mathbb{N}$。假设我们想计算 $(2x)^{\mathcal{B}}(a)$。我们计算 $\mathcal{B}_{a+1}$。显然，$a \in \mathcal{B}_{a+1}$。令 $dx + e$ 为 a 在 $\mathcal{B}_{a+1}$ 中的标记。在阶段 $a+1$ 计算 $\mathcal{B}'$（即调用 $Extend$）。令 b 为标记为 $2(dx + e)$ 的元素。我们断言 b 就是所需的值，即 $(2x)^{\mathcal{B}}(a)$。

假设在阶段 $a+1$ 开始时，a 位于大陆中，即其标记实际上是 e $(d = 0)$。在 $Extend$ 中，如果标记 $2e$ 尚不存在，我们就添加它。根据 $\mathcal{B}'$ 的定义，b 的数量必定是 a 数量的两倍。现在假设在阶段 $a+1$ 开始时，a 不在大陆中。在某个阶段，我们将执行 $Connect$。在那里，我们设置 $c(dq + e) = a$ 以及 $c(2(dq + e)) = b$，因此 b 正如所要求的那样。对 $3x$ 也可作类似论证。□

4.2. **元定理**。首先，我们介绍表述本小节主要结果（下文定理 12）所需的技术性定义。该结果给出了构造 $\mathcal{S} = (\mathbb{N}, S)$ 的点态非标准副本，同时保持函数类 $\mathcal{F}$ 的原始递归性所需的充分条件。直观地说，通过仔细分析大陆–岛屿技巧以及 Levitz 类的特殊性质，可以提炼出这些条件（回顾定义 13）。更具体地说，对 Levitz 的论文 [14] 的分析（将在第 4.4 小节中详述）表明：每个函数 $f \in \mathcal{L}$ 都具有一种“良好”的正规形，而我们将在下文的定义 33 中刻画这一特征。

设 f 和 g 是一元原始递归函数。若存在 $n_0 \in \mathbb{N}$ 使得 g **最终支配** f（记作 $f \preccurlyeq g$），则称 g 最终支配 f。

$$(\forall x \geq n_0)(f(x) \leq g(x)). \tag{2}$$

我们称满足等式（2）的任意数 n_0 为 (f, g)-支配见证。若 n_0 还满足（$\forall x \geq n_0$）（$f(x) < g(x)$），则称 n_0 为严格的 (f, g)-支配见证。

设 $\mathcal{Q} = \{q_i(x_1, x_2, \ldots\ldots, x_{r_i})\}_{i \in \mathbb{N}}$ 是原始递归函数的一致族。令 $T_1(\mathcal{Q})$ 为属于 $\mathcal{Q} \cup \{I_m^n$ 的复合闭包的所有一元函数构成的集合：$1 \leq m \leq n\}$，其中 $I_m^n(x_1, \ldots\ldots, x_n) = x_m$。

我们称函数集 $\mathcal{F} \subseteq T_1(\mathcal{Q})$ 是可列举的，如果存在一个签名 $\mathcal{Q}$ 中项的可计算列表 $(t_i(x))_{i \in \mathbb{N}}$，使得 $\mathcal{F}$ 等于由项 $t_i(x)$、$i \in \mathbb{N}$ 定义的所有函数构成的集合。

注 31。请注意，一般而言，两个不同的项可以定义同一个函数：例如，对于 $\mathcal{Q} = \{+, \cdot\}$，考虑项 $t_1(x) = (x + x) \cdot x$ 和 $t_2(x) = x \cdot x + x \cdot x$。

在下面的定义中，“L”同时表示“可列举的”和“线性有序的”。

定义 32. 若一对 $(\mathcal{Q}, \mathcal{F})$ 满足以下条件，则称其为一个 L-族：

(1) $\mathcal{F} \subseteq T_1(\mathcal{Q})$ 是一个可列举的一元函数族。*(2)* 该族 $\mathcal{F}$ 包含后继 $S(x)$，且 $I^1_1(x) = x$。*(3)* 族 $\mathcal{F}$ 关于复合封闭，即如果 $f(x)$ 和 $g(x)$ 都属于 $\mathcal{F}$，那么 $g(f(x)) \in \mathcal{F}$。*(4)* 函数集 $\mathcal{F}$ 关于支配序 $\preccurlyeq$ 是线性有序的。*(5)* $\mathcal{F}$ 中的每个函数 $f(x)$ 都是严格递增的：即，如果 $x < y$，那么 $f(x) < f(y)$。

定义 33。 设 $(\mathcal{Q}, \mathcal{F})$ 是一个 L-族。如果原始递归集 $T \subseteq \mathbb{N}$ 满足以下条件，我们称其为 关于 $(\mathcal{Q}, \mathcal{F})$ 的范式集：

(1) 给定任意定义从 $\mathcal{F}$ 到某个函数的项 $t(x)$（在签名 $\mathcal{Q}$ 中），可以原始递归地计算出值 $N(t) \in T$。*(2)* 在 T 上存在一个原始递归线性序 $\leq^T$，使得对于定义从 $\mathcal{F}$ 到某个函数的任意项 $t_1(x)$ 和 $t_2(x)$，都有：*(a)* 当且仅当 $N(t_1) \leq^T N(t_2)$ 时，$t_1(x) \preccurlyeq t_2(x)$。*(b)* 如果 $t_1(x) \prec t_2(x)$，则可以原始递归地找到一个严格的 (t_1, t_2) -支配见证。

本节的主要结果如下：

定理12。 如果一个 L-族 $(\mathcal{Q}, \mathcal{F})$ 具有范式集，那么该族 $\mathcal{F}$ 不是 *punctual*标准性的基。

证明。 我们将构造一个点结构 $\mathcal{A}$，它与 $\mathcal{S} = (\mathbb{N}, S)$ 同构，并具有以下性质：

- 对于每个函数 $f \in \mathcal{F}$，其表示 $f^{\mathcal{A}}$ 都是原始递归的，
- 序关系 $\leq^{\mathcal{A}}$ 是原始递归的，
- 前驱函数 $\mathrm{Pred}^{\mathcal{A}}$ 的表示不是原始递归的（因此，结构 $(\mathbb{N}, S^{\mathcal{A}})$ 是点非标准的）。

我们将满足以下要求：对于 $e \in \mathbb{N}$，

$\mathcal{R}_e$**::** 谓词 $\mathcal{A} \neq \psi_e$，其中 $\psi_e(x)$ 是第 e 个一元原始递归函数。

首先，我们来证明一个关于可列举族的有用事实：

引理 34. 设 $\mathcal{F} \subseteq T_1(\mathcal{Q})$ 是一个可列举族，且 $(t_i(x))_{i\in\mathbb{N}}$ 是一个列出 $\mathcal{F}$ 的项的可计算列表。那么存在一个项的原始递归列表 $(t'_s(x))_{s\in\mathbb{N}}$，使得 $(t'_s(x))_{s\in\mathbb{N}}$ 列出 $\mathcal{F}$。

证明。 对于 $s \in \mathbb{N}$，我们通过递归定义项 $t'_s(x)$ 和辅助值 $p(s)$。令 $t'_0(x) = t_0(x)$，并令 $p(0) = 1$。

对于 $s \geq 1$，如果项 $t_{p(s-1)}(x)$ 可以在至多 s 个计算步内计算出来，那么我们令 $t'_s(x) = t_{p(s-1)}(x)$，且令 $p(s) = p(s-1) + 1$。否则，令 $t'_s(x) = t_0(x)$，且令 $p(s) = p(s-1)$。□

根据引理34，我们选取一个项的原始递归列表 $(t_i(x))_{i\in\mathbb{N}}$，该列表列出了我们的族 $\mathcal{F}$。

对于给定的 e，$s \in \mathbb{N}$ 我们可以用原始递归方式计算出以下有限部分结构 $\mathcal{M}[e, s]$。$\mathcal{M}[e, s]$ 的签名为 $\sigma_e = \{S, \leq, t_0, t_1, \ldots, t_e\}$。

我们令 $\mathcal{M}[e, 0] = \{0\}$。直观地说，$\mathcal{M}[e, s + 1]$ 是通过应用来自 σ_e 的所有函数，并在 $\mathbb{N}$ 中取向下闭包，从 $\mathcal{M}[e, s]$ 得到的。更正式地说，部分结构 $\mathcal{M}[e, s + 1]$定义如下：

- $\mathcal{M}[e, s+1]$ 的定义域包含所有满足 $a \leq f(b)$ 的数 $a \in \mathbb{N}$，其中 $b \in \mathrm{dom}(\mathcal{M}[e, s])$ 且 $f \in \sigma_e$。
- $\leq^{\mathcal{M}[e,s+1]}$ 是由 $\mathbb{N}$ 导出的标准序。
- 对于函数 $f \in \sigma_e$ 和 $x \in \mathrm{dom}(\mathcal{M}[e, s])$，有 $f^{\mathcal{M}[e,s+1]}(x) = f(x)$。
- 对于 $f \in \sigma_e$ 和 $x \in \mathrm{dom}(\mathcal{M}[e, s + 1]) \setminus \mathrm{dom}(\mathcal{M}[e, s])$，$f^{\mathcal{M}[e,s+1]}(x)$ 的值未定义。

所需的及时结构 $\mathcal{A} \cong (\mathbb{N}, S)$ 分阶段构造：在阶段 s，我们构造一个有限结构 $\mathcal{A}_s$。我们将有 $\mathrm{card}(\mathcal{A}_{s+1}) > \mathrm{card}(\mathcal{A}_s) \geq s + 2$。我们假设 $\mathcal{A}_0$ 包含数字 0 和 1，且 $S^{\mathcal{A}}(0) = 1$。

在给定的阶段 s，令 $e(s)$ 为尚未满足的要求中指标最小的那个 $\mathcal{R}_{e(s)}$。我们还将定义一个辅助值 $\theta(s) \in \mathbb{N}$，使得 $\theta(s + 1) > \theta(s)$。

我们将构造一个同构嵌入 $c[s]$，将签名约化 $\mathcal{M}[e(s)\ \theta(s)] \upharpoonright \{S\}$ 嵌入所构造的有限结构 $\mathcal{A}_s$ 中。有时我们会滥用记号，将约化 $\mathcal{M}[e(s)\ \theta(s)] \upharpoonright \{S\}$ 与 $\mathcal{M}[e(s)\ \theta(s)]$ 视为相同。

约定 35. 子结构 $\widehat{\mathcal{M}}[s] = (\mathrm{range}(c[s])\ S^{\mathcal{A}_s})$ 称为（阶段 s 的）大陆，而 $\widehat{\ }$ 子结构 $\mathrm{Ar}[s] = \mathcal{A}_s \setminus \mathcal{M}[s]$ 称为群岛。

我们始终假设，大陆中的每个元素 x 都严格小于群岛中的所有元素 $<^{\mathcal{A}}$。

我们注意到大陆具有以下性质：若 $x \in \widehat{\mathcal{M}}[s]$，则元素 $S^{\mathcal{A}}(x)$ 属于 $\widehat{\mathcal{M}}[s+1]$。

我们将确保以下条件：若 $x \in \mathcal{A}_s$，则存在一个阶段 $s' \geq s$，使得 $x \in \widehat{\mathcal{M}[s']}$。也就是说，每个元素 x 最终都会加入大陆。这一性质将保证所构造的结构 $\mathcal{A} = \bigcup_{s\in\mathbb{N}} \mathcal{A}_s$ 通过映射 $c = \bigcup_{s\in\mathbb{N}} c[s]$ 同构于 $(\mathbb{N}, S)$。

每当将新元素 x 加入 $\mathcal{A}_s$ 时，x 都会被选为当前不属于 $\mathcal{A}_s$ 的最小元素。要求 $\mathcal{R}_e$ 的策略。假设 $\mathcal{R}_e$ 在阶段 $s_0 \geq 1$ 开始运行。

首先，选择一个新的见证 w。见证 w 成为当前群岛中的一个元素，因此在阶段 s_0，它没有通过函数 $S^{\mathcal{A}}$ 与大陆连接。

等待一个满足条件的阶段 $s' > s_0$：$\psi_{e,\ s'}(w)\downarrow$。等待期间，我们通过下文所述的扩展过程 Extend$(e,\ s)$ 构造 $\mathcal{A}_s$。这里要注意，w 仍未与大陆通过 $S^{\mathcal{A}}$ 连接。

在过程 Extend$(e,\ s)$ 中，我们定义一个辅助参数 $\beta(s) \in \mathbb{N}$。这个参数将帮助我们处理群岛的增长（下文将详细说明）。我们令 $\beta(s_0) = s_0$。

假设 $\psi_{e,\ s'}(w)\downarrow = y$。那么由式（1），可得 $y < s'$，因此 $y \in \mathcal{A}_{s'-2}$：事实上，注意到基数$(\mathcal{A}_{s'-2}) \geq s'$，于是 $\{0, 1, \ldots, s'-1\} \subseteq$ 定义域$(\mathcal{A}_{s'-2})$。因此 $S^{\mathcal{A}}(y) \in \mathcal{A}_{s'-1}$ 且 $S^{\mathcal{A}}(y) \neq w$。事实上，这是以下性质的推论：

- 在结构 $\mathcal{A}_{s'-1}$ 中，元素 w 没有与大陆连接，因此，对于所有 $\frown z \in \mathcal{M}[s'-1]$，$w \neq S^{\mathcal{A}}(z)$。$\frown$
- 若 $z \in \mathrm{Ar}[s'-1] = \mathcal{A}_{s'-1} \setminus \mathcal{M}[s'-1]$，那么（根据下文对扩展过程 $\mathtt{Extend}(e, s)$ 的描述）存在某个项 $q(x$，它属于签名 σ_e，使得 $z = q^{\mathcal{A}}(w)$ 且 $q(x) \in \mathcal{F}$。由于族 $\mathcal{F}$ 中的每个函数都严格递增（定义 32 的条目（5））$，$我们推得$w \leq^{\mathcal{A}} z$ 且 $w \neq S^{\mathcal{A}}(z)$。

因此，我们有 $S^{\mathcal{A}}(\psi_e(w)) \neq w$，从而推得 $\psi_e(w) \neq \mathrm{Pred}^{\mathcal{A}}(w)$。因此，要求$\mathcal{R}_e$ 已（永远）满足。

在阶段 s' 结束时，我们通过下文所述的连接过程 Connect$(e,\ s')$ 将当前群岛连接到大陆。之后，我们可以放心地转到下一个尚未满足的要求。Extend过程$(e,\ s)$。首先，我们放入 $\theta(s) = \theta(s-1)+1$，并使用新鲜数生成以下扩展：

- 大陆 $\widehat{\mathcal{M}[s-1]}$ 扩展为新的大陆 $\widehat{\mathcal{M}[s]}$，它与 $\widehat{\mathcal{M}[e, \theta(s-1)+1]}$ 同构$\frown$
- 同构 $c[s-1]$（从 $\mathcal{M}[e, \theta(s-1)]$ 映射到 $\mathcal{M}[s-1]$）扩展为一个同构$\frown c[s]$，从 $\mathcal{M}[e, \theta(s-1)\ +1]$ 映射到 $\mathcal{M}[s]$。

在构造过程中，群岛中的每个元素 z 都有自己的标签 $q_z(w)$，其中 $q_z(x)$ 是签名 $\sigma_e = \{S, t_0, t_1, \ldots, t_e\}$ 中的一个项，使得 $q_z(x)$ 定义了从族 $\mathcal{F}$到某处的函数。该标签满足 $q_z^{\mathcal{A}}(w) = z$。

我们假设元素 w 的标签为 w（回想一下，函数 I_1^1 属于 $\mathcal{F}$）。

我们固定范式集 T，它来自定义 33（以及相应的对象 $N(t)$、$\leq^T$）。

假设我们已经为每个元素 $z \in \mathrm{Ar}[s-1]$ 定义了标签 $q_z(w)$。那么新的群岛 $\mathrm{Ar}[s] \supset \mathrm{Ar}[s-1]$ 按如下方式构造。

依次对每个函数 $f \in \sigma_e$ 及其对应于 $z \in \mathrm{Ar}[s-1]$ 的每个标签 $q_z(w)$，执行以下操作：

- 考虑项 $u(x) = f(q_z(x))$。根据定义 32的第（2）–（3）项，$u(x)$ 定义了一个从族 $\mathcal{F}$到 $N(u)$ 的函数。因此，我们可以用原始递归计算其值。
- 如果存在一个来自当前群岛的元素 a，使得 $N(q_a) = N(u)$（也就是说，$q_a(x)$ 和$u(x)$ 定义了同一个函数），那么我们就声明 $f^{\mathcal{A}}(z) = a$。这一声明定义良好，

因为我们必定有

$$a = q_a^{\mathcal{A}}(w) = u^{\mathcal{A}}(w) = f^{\mathcal{A}}(q_z^{\mathcal{A}}(w)) = f^{\mathcal{A}}(z).$$

• 否则，对于当前群岛中的所有 a，都有 $N(q_a) \neq N(u)$。由于 $\leq^T$ 是 T 上的线性序（并且集合 $\mathcal{F}$ 关于支配序 $\preccurlyeq$ 是线性有序的），这意味着对于每个 a，要么 $q_a(x) \prec u(x)$，要么 $u(x) \prec q_a(x)$。

我们向群岛中添加一个新元素 a^*，并定义其标签 $q_{a^*}(w) = u(w)$。我们令 $f^{\mathcal{A}}(z) = a^*$。此外，对于当前群岛中的每个元素 $b \neq a^*$，我们执行以下操作：

– 如果 $N(q_b) <^T N(u)$，那么我们计算一个严格的 (q_b, u)-支配见证 d_b。我们还声明 $b <^{\mathcal{A}} a^*$。– 如果 $N(u) <^T N(q_b)$，那么我们计算一个严格的 (u, q_b)-支配见证 d_b。我们声明 $a^* <^{\mathcal{A}} b$。

最后，值 $\beta(s)$定义为 $1+$（ $\beta(s-1)$与上述计算中出现的所有严格支配见证中的最大值d_b ）。因此，若 $x \geq \beta(s)$且 $a,\ b \in \mathrm{Ar}[s]$ 且 $a <^{\mathcal{A}} b$，则有 $q_a(x) < q_b(x)$。

Connect过程$(e,\ s)$。我们定义值 $\Theta = \max(\theta(s-1)\ \beta(s-1))$。上文给出的值$\beta(s-1)$的计算确保参数 Θ具有以下性质：

（†）：假设在阶段 s 开始时，我们有两个来自群岛$\mathrm{Ar}[s-1]$的元素 $a \neq b$。那么相应的标签 $q_a(w)$和 $q_b(w)$ 满足以下条件：若 $x \geq \Theta$ 且 $a <^{\mathcal{A}} b$，则$q_a(x) < q_b(x)$。

我们计算值 $D' = \max(\mathrm{dom}(\mathcal{M}[e, \theta(s-1)]))$和 $D = c(D')$。

如果 $D' < \Theta$，那么首先我们使用新数，将 $\mathcal{M}[s-1]$ 扩展为 $\mathcal{M}[e, \Theta]$ 的同构副本（并以适当的方式扩展同构 $c[s-1]$ ）。因此，不失一般性，我们可以假设 $D' \geq \Theta$。

注意，值 $S^{\mathcal{A}_{s-1}}(D)$ 未定义。因此，我们的连接过程（用于构造 $\mathcal{A}_s$）安排如下：

(a) 我们令 $S^{\mathcal{A}}(D) = w$，并令 $c(D'+1) = w$。
(b) 对每个满足 $a \neq w$ 的 $a \in \mathrm{Ar}[s-1]$，我们找到标签 $q_a(w)$，并定义 $c(q_a^{\mathcal{S}}(D'+1)) = a$，其中值 $q_a^{\mathcal{S}}(D'+1)$ 在标准副本 $\mathcal{S} = (\mathbb{N},\ S)$ 中求得。

由于每个函数 $q_a(x) \in \mathcal{F}$ 都是严格递增的，因此我们有 $q_a^{\mathcal{S}}(D'+1) \geq D'+1$，从而 $q_a^{\mathcal{S}}(D'+1)$ 大于 $\mathcal{M}[e, \theta(s-1)]$ 中的所有元素。

由性质（†）可推得，所有 $a, b \in \mathrm{Ar}[s-1]$ 都满足：

$$q_a^{\mathcal{S}}(D'+1) \leq q_b^{\mathcal{S}}(D'+1) \iff a \leq^{\mathcal{A}} b.$$

特别地，$c^{-1} \restriction \mathrm{Ar}[s-1]$ 是良定义映射，也是线性序（$\mathrm{Ar}[s-1], \leq^{\mathcal{A}}$）与（$c^{-1}(\mathrm{Ar}[s-1])\ \leq$）之间的同构。

由此我们得到（目前）有

$$x \leq y \ \Leftrightarrow\ c(x) \leq^{\mathcal{A}} c(y), \quad \text{for all } x, y \in \mathrm{dom}(c[s]).$$

（c）我们计算 $\Theta' = \max\{c^{-1}(a) :\ a \in \mathrm{Ar}[s-1]\}$。注意 $\Theta' \geq D'+1 > \Theta \geq \theta(s-1)$。因此，$\Theta' \geq \theta(s-1)+1$。我们令 $\theta(s) = \Theta'$，并使用新数将 $\mathcal{M}[s-1]$ 扩展到 $\mathcal{M}[s] \cong \mathcal{M}[e, \theta(s)]$。我们还将 $c[s-1]$ 扩展为一个同构 $c[s]$，使其从 $\mathcal{M}[e, \theta(s)]$ 映射到 $\mathcal{M}[s]$。

构造。按照上述策略，我们逐一满足要求 $\mathcal{R}_e$、 $e \in \mathbb{N}$：$\mathcal{R}_0$，$\mathcal{R}_1$，$\mathcal{R}_2$，.....。

如果在阶段 s 结束时，我们需要从要求 $\mathcal{R}_e$ 转到下一个要求 $\mathcal{R}_{e+1}$，那么（如有需要），我们就使用新数将当前大陆 $\mathcal{M}[s]$ （其形式类似于 $\mathcal{M}[e, \theta(s)]$）扩展为 $\mathcal{M}[e+1, \theta(s)]$的同构副本。我们还会以适当的方式扩展同构 $c[s]$。

验证。显然，构造的结构 $\mathcal{A}=\bigcup_{s\in\mathbb{N}}\mathcal{A}_s$ 的定义域是 $\mathbb{N}$。注意，对于每个 $x\in\mathbb{N}$，都有 $x\in\mathcal{A}_x$。

由于所述构造从未使用无界搜索过程，结构 $\mathcal{A}=(\mathbb{N},\ S^{\mathcal{A}})^{\frown}$是及时的。此外，每个数 $x\in\mathbb{N}$最终都会加入大陆 $\mathcal{M}[s]$，并且到每个阶段 $^{\frown}$结束时，结构 s 都与签名约化 $\mathcal{M}[s]\upharpoonright(\mathcal{M}[e]s(\theta)s[]\upharpoonright\{S\})$同构。因此，我们有$\curvearrowright\mathcal{A}=\bigcup_{s\in\mathbb{N}}\mathcal{M}[s]\cong(\mathbb{N},\ S)$。

序关系 $\leq^{\mathcal{A}}$ 是原始递归的。事实上，以下原始递归过程为此提供了见证。给定数 $x\neq y$，考虑值 $s=\max(x,\ y)$和有限结构 $\mathcal{A}_s$:

- 如果 $x,\ y$ 都属于大陆 $\widehat{\mathcal{M}[s]}$，那么我们计算 $c^{-1}(x)$ 和 $c^{-1}(y)$。这里有$x<^{\mathcal{A}}y\iff c^{-1}(x)<c^{-1}(y)$。
- 若 $x,\ y$ 均属于群岛$\mathrm{Ar}[s]$，则我们有不等式 $x<^{\mathcal{A}}y$，当且仅当该不等式已在阶段 s结束前由Extend过程明确声明。$\widehat{}$
- 否则，其中一个数，例如 x 属于 $\mathcal{M}[s]$，而另一个数 y 属于$\mathrm{Ar}[s]$。于是我们有 $x<^{\mathcal{A}}y$。

对于每个函数 $f\in\mathcal{F}$，存在一个指标 e，使得 $f(x)=t_e(x)$。我们证明函数$f^{\mathcal{A}}(x)=t_e^{\mathcal{A}}(x)$是原始递归的。

我们非一致地固定阶段 $s^*\geq 1$，使得我们的构造在阶段 s^*开始处理要求 $\mathcal{R}_e$。给定 $x\in\mathbb{N}$，我们描述一个用于计算 $t_e^{\mathcal{A}}(x)$的原始递归过程。我们考虑$s=\max(s^*,\ x)$以及有限结构 $\mathcal{A}_s\subset\mathcal{A}_{s+1}$。注意 $x\in\mathcal{A}_s$。

- 如果 x 属于大陆 $\mathcal{M}[s]$，那么使$\widehat{\text{用}}$函数 $c[s+1]$，我们就能恢复值 $c(t_e(c^{-1}(x)))$，它等于 $t_e^{\mathcal{A}}(x)$。
- 否则，x 属于群岛 $\mathrm{Ar}[s]$，并且到阶段 $s+1$结束时，Extend过程已明确声明了值 $t_e^{\mathcal{A}}(x)$。

我们得到，对于每个 $f\in\mathcal{F}$，函数 $f^{\mathcal{A}}$ 都是原始递归的。

在描述要求 $\mathcal{R}_e$ 的策略时，我们已经证明 $\mathcal{R}_e$ 最终会得到满足。因此，函数 $\mathrm{Pred}^{\mathcal{A}}(x)$ 不是原始递归的。因此，我们得出结论，一致族 $\mathcal{F}$ 不是 punctual标准性的基。定理 12 得证。□

4.3. **定理 12 的推论**。对定理 12 证明的分析使我们能够得到以下技术性结果:

推论 36。假设一个 L-族 $(\mathcal{Q},\mathcal{F})$ 具有范式集。设 $\mathcal{U}=\{u_j(x_1,x_2,\ldots,^{x_{m_j}})\}_{j\in\mathbb{N}}$是一致的原始递归函数族，使得 $\mathcal{U}\subseteq\mathcal{Q}$ 和 $\mathcal{U}$ 满足以下性质：$(\ddagger)$**：** 任取一个 $j\in\mathbb{N}$和一个任意元组 $\vec{r}=(r_1(x)\ r_2(x)\ldots,^{r_{m_j}(x)})$，使得每个 $r_i(x)$要么是来自 $\mathcal{F}$ 的函数，要么是来自 $\mathbb{N}$ 的常数。那么函数

$$u(x)=u_j(r_1(x),r_2(x),\ldots,r_{m_j}(x))$$

要么属于 $\mathcal{F}$，要么是一个常数，并且可以（对 j 和 $\vec{r}$一致地）原始递归地找到一个表达式 $t'(x)$（它要么是签名 $\mathcal{Q}$中的一个项 $t(x)$，要么是一个常数 $d\in\mathbb{N}$），该表达式定义$u(x)$。

那么，族 $\mathcal{F}\cup\mathcal{U}$ 不是*punctual*标准性的基。

证明概要。本质上，需要把所有函数 $u_j\in\mathcal{U}$ 纳入结构 $\mathcal{A}$ 的构造中，而该结构由定理12给出：也就是说，我们还必须额外确保，对每一个 $j\in\mathbb{N}$，函数 $u_j^{\mathcal{A}}$都是原始递归的。

构造的关键修改包括以下内容。

(1) 有限结构 $\mathcal{M}[e,s+1]$ 的定义域必须包含所有数 a，使得对某些 $j\leq e$ 以及 $b_1,\ b_2,\ \ldots\ldots,\ b_{m_j}\in$定义域$(\mathcal{M}[e,s])$，有 $a\leq u_j(b_1,\ b_2,\ \ldots\ldots,\ b_{m_j})$。非正式地说，这一约定确保人们能够在构造的结构 $\mathcal{A}$的大陆部分上“及时地”计算函数 $u_j^{\mathcal{A}}$。

（2）我们必须修改用于扩展当前群岛的过程 $\mathrm{Extend}(e,s)$。在处理 $z\in\mathrm{Ar}[s-1]$ 的标签 $q_z(w)$ 时，我们不仅需要对它应用函数 $f\in\sigma_e$，还需要应用来自性质 $(\ddagger)$ 的所有函数 $f'(x)=u_j(r_1(x)\ r_2(x)\ldots,^{r_{m_j}(x)})$，其中 $j\leq e$，并且参数 $r_i(x)$ 的选取方式如下:

- 要么是当前大陆 $\mathcal{M}[s]$ 中的一个常数，
- 对于函数 $q_a(x)$，其中标签 $q_a(w)$ 属于群岛 $\mathrm{Ar}[s-1]$。

非正式地说，这使我们能够及时计算值 $u_j^{\mathcal{A}}(b_1,\ b_2,\ \ldots\ldots,\ b_{m_j})$，前提是某个$b_i$ 首次出现在 $\mathcal{A}$ 的群岛部分中。

证明的其余部分与定理 12的证明类似，只需作相应调整。 □

我们说明如何应用推论 36。回想一下

推论 10。 标准的后继运算 S、加法 $+$、乘法 $\times$和序关系 $\leq$ 并不构成点标准性的基。

为了证明推论 10，我们选取以下对象：

- 原始递归函数族 $\mathcal{Q}$等于 $\{S,+,\times\}$，
- 一元函数族 $\mathcal{F}$ 选为所有非恒定多项式 $p(x)$组成的函数族，其系数取自 $\mathbb{N}$。

容易验证，数对 $(\mathcal{Q},\ \mathcal{F})$ 是一个 L-族。此外，族 $\mathcal{U} := \mathcal{Q}$ 具有推论 36 中的性质(‡)。

现在我们需要为 $(\mathcal{Q},\ \mathcal{F})$ 引入一个范式集 T（回顾定义 33）。给定一个非恒定多项式

$$t(x) = a_k x^k + a_{k-1}x^{k-1} + \cdots + a_1 x + a_0, \quad a_i \in \mathbb{N},\ a_k \neq 0,$$

我们令 $N(t) = p_0^{a_0} \cdot p_1^{a_1} \cdot \ldots \cdot p_{k-1}^{a_{k-1}} \cdot p_k^{a_k}$，其中 p_i是第 i 个素数。我们定义 $T = \{N(t):\ t \in \mathcal{F}\}$.

设 $q(x)$ 和 $\neq t(x)$ 是来自 $\mathcal{F}$ 的两个多项式。当且仅当多项式 $u(x) := q(x) - t(x)$ 的首项系数为正时，$q(x)$ 最终支配 $t(x)$。假设 $u(x)$ 等于 $b_k x^k + \cdots + b_1 x + b_0$，其中 $b_k \in \mathbb{N}^+$和 $b_i \in \mathbb{Z}$。那么可以选取 $n_0 := k \cdot (1 + \max\{|b_j|:\ j \leq k\})$ 作为严格的（t，q）-支配见证。

利用上述论证，很容易证明 T确实是 $(\mathcal{Q},\ \mathcal{F})$ 的范式集。因此，应用推论 36，我们得到推论 10。

4.4. Levitz 类的一个应用。

回顾 Levitz 类的概念：

定义 13。 Levitz 类 $\mathcal{L}$ 是包含常函数 $f\colon\ \mathbb{N} \to \mathbb{N}$ 和恒等函数 $x \mapsto x$，并满足以下条件的最小函数类：若 f、$g \in \mathcal{L}$和 $n \in \mathbb{N}^+$，则函数

(a) $x \mapsto f(x) + g(x)$,

(b) $x \mapsto f(x) \cdot g(x)$,

(c) $x \mapsto n^{f(x)}$，以及

(d) $x \mapsto x^{f(x)}$

也属于 $\mathcal{L}$。

如果从 *Levitz* 类的定义中省略条件 *(d)*，则所得函数类记作 $\mathcal{L}_c$。注意，$\mathcal{L}_c \subset \mathcal{L}$，且类 $\mathcal{L}_c$ 对复合封闭。

下面证明：

推论 14。 类 $\mathcal{L}_c$ 不是点标准性的基。

证明。 注意，对于来自 Levitz 类 $\mathcal{L}$ 的函数 f，只有当 $x = 0$ 或 $f = 0$ 时，$= 0\ (x)$ 才有可能。因此，从 $\mathcal{L}$ 到 $\mathbb{N}^+$ 的函数的限制恰好是第 3 节中来自 [14], 的类 $\mathcal{L}'$ 中的函数。特别地，按照（最终）支配序 $\preccurlyeq$，它们是良序的。

为了应用定理 12，我们选择：

- $\mathcal{Q} = \{S,+,\times\} \cup \{n^x:\ n \in \mathbb{N}^+\}$,
- $\mathcal{F}$ 是类 $\mathcal{L}_c$ 中所有非常值函数构成的族。

显然，$\mathcal{F}$ 中的每个函数都是严格递增的。由此可得，$(\mathcal{Q},\mathcal{F})$ 是一个 L-族。现在我们需要为 $(\mathcal{Q},\mathcal{F})$ 引入一个范式集 T。

我们假设项 $f \in \mathcal{L}$采用以下编码：

$$\ulcorner 0 \urcorner = 0, \quad \ulcorner 1 \urcorner = 1, \quad \ulcorner f + g \urcorner = 2^{\ulcorner f \urcorner + 1} \cdot 3^{\ulcorner g \urcorner + 1},$$
$$\ulcorner f \cdot g \urcorner = 5^{\ulcorner f \urcorner + 1} \cdot 7^{\ulcorner g \urcorner + 1}, \quad \ulcorner x^f \urcorner = 11^{\ulcorner f \urcorner + 1}, \quad \ulcorner n^f \urcorner = 13^n \cdot 17^{\ulcorner f \urcorner + 1}.$$

注意，关系 $f = 0$ 和 $f = 1$ 在 $\ulcorner f \urcorner$中是原始递归的。

我们定义

$$T = \{\ulcorner f \urcorner : f \in \mathcal{F}\} \text{ and } \widehat{T} = \{\ulcorner f \urcorner : f \in \mathcal{L}\} \tag{3}$$

注意到， T 和 T 都是原始递归集合。

在证明的其余部分，我们证明较大的集合 T 满足定义33中关于 对 $(\mathcal{Q},\ \mathcal{L})$ 的条件 2，其中 $\mathcal{Q} = \mathcal{Q} \cup \{x^y\}$。这一事实意味着， T是 $(\mathcal{Q},\ \mathcal{F})$ 的范式集。

我们在此指出为 Lemma 12 证明的“较大”的对 $(\mathcal{Q},\ \mathcal{L})$ 能直接将定理 12 复用到 T 上。

我们介绍分析 T 与 $\mathcal{L}$所必需的背景。按照 [14]，若 $f = g + h$ 蕴含 $g = 0$ 或 $h = 0$，则称 $f \neq 0$ 为加法素元。

$f \neq 1$ 的乘法正规形 （乘法正规形）是一种表示 $f = u_1^{f_1} u_2^{f_2} \ldots u_k^{f_k}$，其中：

1. 每个 f_i 均为加法素元；2. 每个 u_i 都属于 $\mathbb{N} \setminus \{0,1\} \cup \{x\}$；3. 若 $i \neq j$ 且 u_i, $u_j \in \mathbb{N}$，则 $f_i \neq f_j$；4. 若 $u_i \in \mathbb{N}$，则 $f_i \neq 1$；5. $u_k^{f_k} \preccurlyeq \ldots \preccurlyeq u_2^{f_2} \preccurlyeq u_1^{f_1}$。

通过省略第3项与第5项，我们得到 f的集合乘法正规形 概念。

每个加法素元 $f \neq 1$都有唯一的乘法正规形，记作 $MNF(f)$。我们假设$MNF(1) = 1$。

Th [14] 中的定理 3.2 表明，对于 f，g采用乘法正规形时，$f = u_1^{f_1} u_2^{f_2} \ldots u_k^{f_k}$和 $g = v_1^{g_1} v_2^{g_2} \ldots v_\ell^{g_\ell}$，

$$\begin{aligned} f \prec g \ \text{ iff } \ \exists i \ [\forall j < i \ (u_j^{f_j} = v_j^{g_j}) \ \& \\ (i = k+1 \leq \ell \ \vee \ f_i \prec g_i \ \vee \ (f_i = g_i \ \& \ u_i \prec v_i))]. \end{aligned} \tag{4}$$

一个加 法正规形 (ANF) 对于 $f \neq 0$是一个表示 $f = p_1 + p_2 + \ldots + p_k$，其中：

1. 每个 p_i 都是加法素元；
2. $p_k \preccurlyeq \ldots \preccurlyeq p_2 \preccurlyeq p_1$。

通过放宽要求 2.，我们得到 f 的集合*ANF* 这一概念。

每个 $f \neq 0$ 都有唯一的 ANF，记作 $ANF(f)$。我们规定 $ANF(0) = 0$。

定理 3.3 在 [14] 中表明，对于 f，g 的 ANF， $f = p_1 + p_2 + \ldots p_k$ 且 $g = q_1 + q_2 + \ldots q_\ell$

$$f \prec g \ \text{ iff } \ \exists i \ [\forall j < i \ (p_j = q_j) \ \& \ (i = k+1 \leq \ell \ \vee \ p_i \prec q_i)]. \tag{5}$$

作为推论，MNF 中的任意 $f = u_1^{f_1} u_2^{f_2} \ldots u_k^{f_k}$ 都是加法素元。特别地，加性素元的乘积也是加法素元。

直观地说，接下来一系列技术引理确立了定义 33 的条件 2.(a)：最终支配的序关系 $f \preccurlyeq g$可由相应的编码原始递归地计算 $\ulcorner f \urcorner$，$\ulcorner g \urcorner$。

引理 37。 存在函数 $\boldsymbol{sa}$、 $\boldsymbol{sm}$： $\mathcal{L} \to \mathcal{L}$，使得：

1. 对任意 $f \in \mathcal{L}$、 $\boldsymbol{sa}(f) = p_1 + p_2 + \ldots p_k$是 f的集合*ANF*，且 $\ulcorner p_i \urcorner \leq \ulcorner f \urcorner$对所有 i均成立；*2.* 对任意加法素元 $f \in \mathcal{L}$、 $\boldsymbol{sm}(f) = u_1^{f_1} u_2^{f_2} \ldots u_k^{f_k}$是 f的集合乘法正规形，且 $\ulcorner f_i \urcorner < \ulcorner f \urcorner$对所有 i均成立。

此外，$\ulcorner \boldsymbol{sa}(f) \urcorner$和 $\ulcorner \boldsymbol{sm}(f) \urcorner$关于 $\ulcorner f \urcorner$是原始递归的。

证明。 对 f的构造进行归纳。对于 $f \in \{0,1\}$，我们取 $\boldsymbol{sa}(f) = \boldsymbol{sm}(f) = f$。

现在假设我们已经定义了 $\boldsymbol{sa}(f') = p_1 + p_2 + \ldots + p_k$和 $\boldsymbol{sa}(f'') = q_1 + q_2 + \ldots + q_\ell$以及 $\boldsymbol{sm}(f')$、 $\boldsymbol{sm}(f'')$。

对于 $f = f' + f''$,我们定义 $\boldsymbol{sa}(f) = p_1 + p_2 + \ldots + p_k + q_1 + q_2 + \ldots + q_\ell$。显然， $\ulcorner p_i \urcorner \leq \ulcorner f' \urcorner < \ulcorner f \urcorner$ 并且 $\ulcorner q_j \urcorner \leq \ulcorner f \urcorner$。如果 $f \neq f' + f''$ 或 f' 是一个加法素元 , 那么 $f' = 0 (k = 0)$，并且我们定义 $\boldsymbol{sm}(f) = \boldsymbol{sm}(f'')$, 或 $f'' = 0 (\ell = 0)$ 并定义 $\boldsymbol{sm}(f) = \boldsymbol{sm}(f')$。关于指数编码的条件显然已满足，已满足。

对于 $f = f' \cdot f''$，我们定义 $\boldsymbol{sa}(f) = \sum_{i=1}^{k} \sum_{j=1}^{\ell} p_i \cdot q_j$。由于 $p_i \cdot q_j$ 是加法素元，因此该定义是正确的。对于所有 i， j，我们有：

$$\ulcorner p_i \cdot q_j \urcorner \ = \ 5^{\ulcorner p_i \urcorner + 1} \cdot 7^{\ulcorner q_j \urcorner + 1} \ \leq \ 5^{\ulcorner f' \urcorner + 1} \cdot 7^{\ulcorner f'' \urcorner + 1} \ = \ \ulcorner f \urcorner.$$

如果 $f = f' \cdot f''$ 是加法素元，那么 f' 和 f'' 也是加法素元（$k = \ell = 1$），并且我们定义$\boldsymbol{sm}(f) = \boldsymbol{sm}(f') \cdot \boldsymbol{sm}(f'')$。显然，两个集合乘法正规形的乘积仍然是集合乘法正规形。对于 $\boldsymbol{sm}(f)$ 的每个因子 u^p，我们有 $\ulcorner p \urcorner < \max(\ulcorner f' \urcorner, \ulcorner f'' \urcorner) \leq \ulcorner f \urcorner$。

对于 $f = x^{f'}$，我们有 $f = x^{p_1} \cdot x^{p_2} \cdot \ldots \cdot x^{p_k}$。这意味着 f 是一个加法素元，因此我们可以取 $\boldsymbol{sa}(f) = f$ 和 $\boldsymbol{sm}(f) = x^{p_1} \cdot x^{p_2} \cdot \ldots \cdot x^{p_k}$。当然，对于所有 i，我们都有 $\ulcorner p_i \urcorner \leq \ulcorner f' \urcorner < \ulcorner f \urcorner$。

最后，如果 $n \neq 1$，我们定义 $\boldsymbol{sa}(f) = \boldsymbol{sm}(f) = 1$。对于 $n \geq 2$，我们有 $f = n^{p_1} \cdot n^{p_2} \cdot \ldots \cdot n^{p_k}$。如果 $p_i \neq 1$，那么 n^{p_i} 是一个加法素元，但对于某个 i，我们可能有 $p_i = 1$。为此，令 $I = \{i \mid p_i = 1\}$ 和 $p = n^{\sum_{i \notin I} p_i}$。那么 p 是一个加法素元，因为它是若干加性素元的乘积。我们取

$$\boldsymbol{sa}(f) = \underbrace{p + p + \ldots + p}_{n^{|I|}\ times}.$$

注意到 $\ulcorner p \urcorner \leq \ulcorner n^{f'} \urcorner = \ulcorner f \urcorner$。

如果 f 是加法素元（$I = \emptyset$），那么我们定义 $\boldsymbol{sm}(f) = \prod_{i=1}^{k} n^{p_i}$。这个定义中的指数 p_i 显然满足 $\ulcorner p_i \urcorner \leq \ulcorner f' \urcorner < \ulcorner f \urcorner$。

在所有情况下，如果 f 不是加法素元，我们就定义 $\boldsymbol{sm}(f) = 0$。 □

引理 38。对于任意加法素元 f，$\ulcorner MNF(f) \urcorner$关于 $\ulcorner f \urcorner$是原始递归的。对于任意加性素元g、f，关系 $g \preccurlyeq f$关于 $\ulcorner g \urcorner$、$\ulcorner f \urcorner$是原始递归的。

证明。我们对 $\ulcorner f \urcorner$以及所有满足 $\ulcorner g \urcorner < \ulcorner f \urcorner$的 g同时进行归纳。

设 f是一个加法素元。利用引理 37，我们得到集合乘法正规形 $\boldsymbol{sm}(f) = u_1^{f_1} u_2^{f_2} \ldots u_k^{f_k}$，其中$\ulcorner f_i \urcorner < \ulcorner f \urcorner$对所有 i。为了构造 $MNF(f)$:

(1) 计算 $MNF(f_1)$ $MNF(f_2) \ldots$, $MNF(f_k)$；(2) 通过将形如 n^{f_i} 和 m^{f_i} 的因子合并为一个因子 $(nm)^{f_i}$，落实 MNF 定义中的第 3 条；(3) 通过按照 $\preccurlyeq$对因子排序，落实 MNF 定义中的第 5 条。

要比较 $u_i^{f_i}$和 $u_j^{f_j}$ 关于 $\preccurlyeq$，我们可以使用等式（4），并对 f_i和 f_j应用归纳假设。显然，整个算法都可以在编码上用原始递归方式完成。

现在设 g，f是加法素元，且满足 $\ulcorner g \urcorner < \ulcorner f \urcorner$。我们使用上述算法计算 $MNF(f)$ 和 $MNF(g)$。然后可以利用等式（4）以及两个乘法正规形中的所有指数的编码都小于 $\ulcorner f \urcorner$这一事实来比较它们。 □

引理39。对于任意 $f \in \mathcal{L}$，$\ulcorner ANF(f) \urcorner$是关于 $\ulcorner f \urcorner$的原始递归。关系 $g \preccurlyeq f$是原始$\widehat{\ }$递归的，关于 $\ulcorner g \urcorner$, $\ulcorner f \urcorner$。因此，式（*3*）中的集合满足定义 *33* 中关于$\widehat{\ }$对 $(\mathcal{Q}, \mathcal{L})$的条件 *2.(a)*。

证明。给定 $f \in \mathcal{L}$，首先利用引理 $\boldsymbol{sa}(f) = p_1 + p_2 + \ldots + p_k$构造集合ANF 37。然后计算$MNF(p_1)$ $MNF(p_2) \ldots, MNF(p_k)$，并利用引理 38，按 排序。该算法从 $\ulcorner ANF(f) \urcorner$原始递归地生成 $\ulcorner f \urcorner$。

现在给定 f，$g \in \mathcal{L}$，首先利用上述算法计算 $ANF(f)$ 和 $ANF(g)$，然后应用（5）以及引理 37 的第二部分来比较各个加项，从而对它们进行比较。该算法关于编码同样是原始递归的。 □

现在我们需要证明定义 33 的条件 2.(b)：也就是说，我们需要说明如何用原始递归找到适当的严格支配见证。

我们假设下文所有对数均以 2 为底。

引理 40。对于任意 $k \geq 5$，都有：$x \geq k^2 \Rightarrow \log_x x < \frac{1}{k}$。

证明。函数 $\log_x x$ 在 $x \geq 3$ 上严格递减。因此，$k \geq 5$ 和 $x \geq k^2$ 推出

$$\frac{\log x}{x} \leq \frac{\log k^2}{k^2} < \frac{1}{k},$$

因为最后一个不等式等价于 $k^2 < 2^k$，而这对任意 $k \geq 5$ 都成立。 □

引理 41. 对于任意 u、$v \in \mathbb{N}$，使得 $2 \leq u < v$：$u^{\frac{1}{(v^2-1)}} < \frac{1}{2}\left(1 - \frac{\log u}{v}\right)$。

证明。对于任意 $n>0$，我们有：

$$\frac{1}{2}\left(1-\frac{\log u}{\log v}\right)>\frac{1}{n} \;\Leftrightarrow\; n-\frac{\log u}{\log v}n>2 \;\Leftrightarrow\; \left(\frac{v}{u}\right)^n>v^2.$$

根据一个众所周知的初等不等式：$\left(\frac{v}{u}\right)^n\geq 1+n(\frac{v}{u}-1)$。我们还有，$1+n(\frac{v}{u}-1)\geq v^2 \Leftrightarrow n\geq u(v^2-1)$ $\cdot$对于 $n=u(v^2-1)$，最后一个不等式显然成立，因为 $v-u\geq 1$。□

通过对 $\max(\ulcorner MNF(f)\urcorner, \ulcorner MNF(g)\urcorner)$ 进行同时原始递归，我们将定义二元函数ω，ν：$\mathcal{L}^2\to\mathbb{N}$，使得：

$(*)$：对于所有加性素元 f，g，若 $f\prec g$，则 $\omega(f,g)$是一个严格的（f，g）-支配见证，而 $\nu(f,g)$是一个（$f\cdot x$，g）-支配见证。

如果 $f=g=1$，我们定义 $\omega(f,g)=\nu(f,g)=0$。

现在令 $MNF(f)=u_1^{f_1}u_2^{f_2}\dots u_k^{f_k}$且 $MNF(g)=v_1^{g_1}v_2^{g_2}\dots v_\ell^{g_\ell}$。

为了定义 $\omega(f,g)$，我们遵循定理 3.2 的证明第一部分中的情形，见 [14]。

情形 1： $f_1\prec g_1$。对每个 $i\geq 2$，我们有 $u_i^{f_i}\preceq u_1^{f_1}$，并且在情形 $u_i^{f_i}\prec u_1^{f_1}$ 中，$\omega(u_i^{f_i},u_1^{f_1})$ 是相应的严格支配见证。此外，$f_1\prec g_1$ 蕴含 $f_1x\preceq g_1$，并且 $\nu(f_1,g_1)$ 是支配见证。对任意大于前述所有见证的 x，我们有：

$$f\preceq u_1^{kf_1},\quad \frac{\log f}{\log v_1^{g_1}}\;\leq\;\frac{k\log x}{x}\left(\frac{1}{\log v_1}\right).$$

由引理 40，我们有：

$$x\geq(k+5)^2\;\Rightarrow\;\frac{\log x}{x}<\frac{1}{k+5}\;\Rightarrow\;\frac{k\log x}{x}\left(\frac{1}{\log v_1}\right)<1,$$

因此我们可以定义

$$\omega(f,g)=\max(\omega(u_2^{f_2},u_1^{f_1}),\dots,\omega(u_k^{f_k},u_1^{f_1}),\nu(f_1,g_1),(k+5)^2).$$

情形 2：： $f_1=g_1$ & $u_1\prec v_1$。对每个 $i\geq 2$，我们有 $f_i\prec f_1$，并令 $\nu(f_i,f_1)$ 为 (f_ix,f_1)-支配见证。我们必须分析以下各项中的被加数：$\frac{f}{\log v_1^{g_1}}=\sum_{i=1}^k\frac{f_i}{\log\log f_1}\frac{u_i}{\log v_1}$。让我们 ~~————~~

take $\epsilon=\frac{1}{2}\left(1-\frac{\log u_1}{\log(u_1+1)}\right)$.

在两种情形 $v_1\in\mathbb{N}$ 与 $v_1=x$ 中，我们有：

$$x>u_1\;\Rightarrow\;\frac{\log u_1}{\log v_1}\;\leq\;\frac{\log u_1}{\log(u_1+1)}.$$

对每个 $i\geq 2$ 与 $x\geq\nu(f_i,f_1)$ $x\geq u_i$，我们有：

$$\frac{f_i\log u_i}{f_1\log v_1}\;\leq\;\frac{\log u_i}{x\log v_1}\;\leq\;\frac{\log x}{x}.$$

由引理 41：$u_1\;\frac{1}{u_1+1((^2-1)<\epsilon}$。因此由引理 40：

$$x\geq\left(ku_1((u_1+1)^2-1)+5\right)^2\;\Rightarrow\;\frac{\log x}{x}\;<\;\frac{1}{ku_1((u_1+1)^2-1)+5}\;<\;\frac{\epsilon}{k}.$$

对 $i=2$，.....，k 求和，我们得到：$\sum_{i=2}^k\frac{f_i}{\log f_1}\frac{u_i}{\log v_1}<\epsilon$。最后，由 ϵ的选取：

$$\sum_{i=1}^k\frac{f_i\log u_i}{f_1\log v_1}\;=\;\frac{\log u_1}{\log v_1}+\sum_{i=2}^k\frac{f_i\log u_i}{f_1\log v_1}\;<\;\frac{\log u_1}{\log(u_1+1)}+\epsilon\;<\;1.$$

我们定义：

$$\omega(f,g)=\max(\nu(f_2,f_1),\dots,\nu(f_k,f_1),u_1+1,u_{max},\left(ku_1((u_1+1)^2-1)+5\right)^2),$$

其中 u_{max} 是集合 $\{u_i\mid 2\leq i\leq k\ \&\ u_i\neq x\}$ 的上确界。

情形 3:: $f_1=g_1$ & $u_1=v_1$。显然，我们定义：

$$\omega(f,g)=\omega(u_2^{f_2}\dots u_k^{f_k},v_2^{g_2}\dots v_\ell^{g_\ell}).$$

其余情况都是对称的（ f 和 g的角色互换）。

现在定义 $\nu(f,\ g)$ 时，我们遵循 [14] 中定理 3.2 证明的第三部分。我们可以假设 $f \prec g$($g \prec f$；另一种情况是对称的，我们定义 $\nu(f,\ f) = 0$)。

情形 1： $f = 1$。然后 $x \preceq g$，我们取 $\nu(f,\ g) = 0$。

情形 2： $f \neq 1$。然后 $x \preceq f$。令 w为 $MNF(f)$ 和 $MNF(g)$ 的公共部分。因此，我们有 $f = f'w$和 $g = g'w$，其中 f' 和 g' 是加性素元，它们的乘法正规形中没有公共项。我们将证明 $f'x \preceq g'$，并由此提取 $(fx,\ g)$- 支配见证。

情形 2.1： $f' = 1$。然后 $g' \neq 1$，因此 $x \preceq g'$。我们定义 $\nu(f,\ g) = 0$。

情形2.2： $f' \neq 1$。我们有 $MNF(f') = s_1^{t_1} s_2^{t_2} \dots s_m^{t_m}$，对于 $m \geq 1$，以及 $MNF(g') = r_1^{q_1} r_2^{q_2} \dots r_p^{q_p}$，对于 $p \geq 1$. 我们比较 $s_1^{t_1}$ 与 $r_1^{q_1}$。

情形2.2.1： $t_1 \prec q_1$。则第一部分中的情形1表明 $f'x \prec g'$，即 $s_1^{t_1} s_2^{t_2} \dots s_m^{t_m} x \prec r_1^{q_1} r_2^{q_2} \dots r_p^{q_p}$。我们定义：

$$\begin{aligned}\nu(f,g) &= \omega(s_1^{t_1} s_2^{t_2} \dots s_m^{t_m} x, r_1^{q_1} r_2^{q_2} \dots r_p^{q_p}) \\ &= \max(\omega(s_2^{t_2}, s_1^{t_1}), \dots, \omega(s_m^{t_m}, s_1^{t_1}), \nu(t_1, q_1), (m+6)^2).\end{aligned}$$

情形2.2.2： $t_1 = q_1$ 且 $s_1 \prec r_1$。则第一部分中的情形2表明 $f'x \prec g'$，即 $s_1^{t_1} s_2^{t_2} \dots s_m^{t_m} x \prec r_1^{q_1} r_2^{q_2} \dots r_p^{q_p}$。我们定义：

$$\begin{aligned}\nu(f,g) &= \omega(s_1^{t_1} s_2^{t_2} \dots s_m^{t_m} x, r_1^{q_1} r_2^{q_2} \dots r_p^{q_p}) \\ &= \max(\nu(t_2, t_1), \dots, \nu(t_m, t_1), s_1 + 1, s_{max}, \big((m+1)s_1((s_1+1)^2 - 1) + 5\big)^2).\end{aligned}$$

根据假设，情形 $t_1 = q_1$ & $s_1 = r_1$ 不可能发生。

最后，我们将定义 $d\colon \mathcal{L}^2 \to \mathbb{N}$，使得对于任意 $f, g \in \mathcal{L}$, $f \prec g$，均可推出 $d(f,g)$ 是一个严格的 (f,g)-支配见证。

给定 $f, g \in \mathcal{L}$，我们首先计算 $ANF(f) = p_1 + p_2 + \dots p_k$ 和 $ANF(g) = q_1 + q_2 + \dots + q_m$。如果 $ANF(f)$ 是 $ANF(g)$ 的前缀，我们定义 $d(f,g) = 0$。否则，令 $i \leq k$ 为第一个满足 $p_i \neq q_i$ 的指标。

情形 1:: $p_i \prec q_i$。那么对于 $x \geq \nu(p_i, q_i)$，有 $p_i x \leq q_i$。我们有 $p_i + \dots + p_k \leq k \cdot p_i \leq p_i x \leq q_i + \dots + q_k$ 对于 $x \geq k$。我们定义 $d(f,g) = \max(\nu(p_i, q_i)\ k)$。

情形 2:: $q_i \prec p_i$。对称，$d(f,g) = \max(\nu(q_i, p_i)\ m)$。

当然，d在编码上是原始递归的。我们由此得出，来自等式（3）的集合 T 满足定义33 对偶（$\mathcal{Q}$, $\mathcal{L}$）的要求。这意味着 T 是 L族（$\mathcal{Q}$, $\mathcal{F}$）的范式集。根据定理12，集合 $\mathcal{F}$ 和 $\mathcal{L}_c = \mathcal{F} \cup \{\text{常函数}\}$ 都不是点标准性的基。推论14 得证。□

5. Sufficient Condition for Punctual Standardness

本节在关于 [12] 中准时范畴性的结果基础上，建立点标准性的充分条件。回顾一下，$(\psi_e)_{e \in \mathbb{N}}$ 是对一元原始递归函数类的可计算枚举。为方便起见，我们将 $Nf(e,x,y,s)$ 记作这样的初等关系：当且仅当 $\psi_{e,s}(x)\downarrow$ 且 $\psi_e(x) = y$ 时，该关系成立。

目前，我们将语言限制为一个常量和两个一元函数。令 $\mathcal{B}_n$ 为该语言中的第 n 个及时结构，由 $\mathcal{B}_n = (\mathbb{N},\ o_n,\ s_n,\ c_n)$ 定义，其中 n 是一个三元组的编码，$o_n = (n)_0$，$s_n = \psi_{(n)_1}$，$c_n = \psi_{(n)_2}$。

在计算下面两个引理中的步骤数时，我们只考虑调用关系 Nf 的计算部分。

引理42。 关系 Cyc 是初等的，其中 $Cyc(n、\ y、\ l、\ s)$ 成立，当且仅当 y 具有 c_n 个循环长度l，且这一验证需要 $\leq s$ 步。

证明。非正式地说，当序列 y、$c_n(y)$ $c_n^2(y)$.....、$c_n^{l-1}(y)$ 由互不相同的元素组成，满足 $c_n^l(y) = y$，并且所有计算均需 $\leq s$ 步时，$Cyc(n,\ y,\ l,\ s)$ 成立。

因此，当且仅当存在一个列表 $L = \langle L_0,\ L_1,\ \dots,\ L_l \rangle$，满足以下条件时，$Cyc(n,\ y,\ l,\ s)$ 成立：(1) $L_0 = L_l = y$; (2) $\forall i < l,\ Nf((n)_2,\ L_i,\ L_{i+1},\ s)$; (3) $\forall i < j < l,\ L_i \neq L_j$。

此外，L 的编码以 $\langle y,\ s,\ \dots\dots,$ 长度为 $l+1$的 $s \rangle$ 为界，它在 y, s, l中是初等的。□

引理 43。其余　　关系 $PatCyc$ 是初等的，其中 $PatCyc$（n，i，x，s）恰好当该元素　　s

$$y_0 = s_n^i(x),\ y_1 = s_n^{2^{n+1}}(y_0),\ \ldots,\ y_x = s_n^{2^{n+1}}(y_{x-1}),\ y_{x+1} = s_n^{2^{n+1}}(y_x)$$

分别具有 c_n-个周期长度 $2n+1$、$2n+2$、.....、$2n+2$、$2n+1$，且这些计算恰好需要 s步。

证明。我们以相同含义定义 $PatCyc'(n,i,x,s)$，但执行 $\leq s$ 步，然后我们 s　　et

$$PatCyc(n,i,x,s) \ \Leftrightarrow\ PatCyc'(n,i,x,s)\ \&\ \neg PatCyc'(n,i,x,s-1).$$

因此，当且仅当存在一个列表 $L = \langle L_0, L_1, \ldots, L_l\rangle$，使得:

(1) $l = i + (x+1)\cdot 2^{n+1}$, (2) $L_0 = x$, (3) $\forall i < l$, $Nf((n)_1, L_i, L_{i+1}, s)$, (4) $Cyc(n, L_i, 2n+1, s)$, (5) $\forall j < x$, $Cyc(n, L_{i+(j+1)\cdot 2^{n+1}}, 2n+2, s)$, (6) $Cyc(n, L_{i+(x+1)\cdot 2^{n+1}}, 2n+1, s)$.

与前一个引理中一样，L 的编码有上界 $\langle x, s, \ldots, s\rangle$，其长度为 $l+1$，它在n, i, x, s中是初等的。□

定理44。函数 f，定义于 *[12]*, 命题*4.2* 中，是初等的。

证明。首先我们证明，关系 $u_x = y$ 在 n、x、y 中是初等的，其中 u_x 是截至阶段 x 的步数。如果相应的搜索在阶段 x 失败，我们约定 $u_x = u_{x-1}$，并规定$u_{-1} = 0$（在 [12]中，u_x 在这种情况下未定义）。

关系 $u_x = y$ 可以通过以下递归算法判定:

(1) 如果 $x = 0$，则令 $y' = 0$。(2) 如果 $x > 0$: (a) 搜索满足 $u_{x-1} = y'$ 的 $y' \leq y$。（b）如果搜索失败，则返回 假。(3) 搜索最小的 $i \in [1;\ \langle n, y'\rangle + (x+1)2^{n+1}]$，使得对于某个（唯一的）$s \leq y$, $PatCyc(n, i, x, s)$ 成立。(4) 如果搜索成功，则返回 $y = y' + s$。(5) 如果搜索不成功，则返回 $y = y'$。

更正式地说，如果 $P(n,\ x,\ y)$ 是真值列表

$$u_x = 0,\ u_x = 1,\ \ldots,\ u_x = y,$$

那么上述算法通过对 x 进行有界原始递归，计算出 P 的编码。因此，P 是初等的，关系 $u_x = y$ 因而也是初等的。

请注意，如果对某个 x 的搜索在第 3 步失败，对某个 $x' > x$ 的搜索仍可能成功。在 [12]，中，构造在第一次尝试失败后不会再进行搜索，但从我们的角度看，这一区别无关紧要。

现在，给定输入 t，我们使用以下初等算法计算 $f(t)$:

(1) 计算 $n,\ m$，使得 $t = \langle n,\ m\rangle$。(2) 如果 $u_0 = 0$ 或 $m \leq u_0$, 则返回 $2n+1$。(3) 搜索满足 $m \leq u_{x+1}$ 或 $u_{x+1} = u_x$ 的最小 $x < m$。(4) 搜索满足 $u_x = v$ 的 $v < m$。(5) 如果 $m \equiv v+1$ 模 $(x+2)$, 则返回 $2n+1$。(6) 如果 $m \not\equiv v+1$ 模 $(x+2)$，则返回 $2n+2$。

注意，关系 $u_{x+1} = u_x$不是初等的，但在 3. 中使用它时，我们有 $u_0 < \ldots < u_x < m$，因此 $u_{x+1} = u_x$ 可以替换为 $\exists v' < m(u_x = v'\ \&\ u_{x+1} = v')$。□

接下来我们定义结构 $\mathcal{A}_f = (\mathbb{N},\ 0,\ s_K,\ c_K)$，其中第 n 个周期有 $f(n)$ 个元素，从 $2n$ 开始，而周期的其他元素依次填入奇数。

定理 45。函数 s_K 和 c_K 是初等的。

证明。 周期 0，1，……， n 包含 $p(n) = \sum_{s=0}^{n}(f(s) - 1)$ 个奇数。因此，对于 $n > 0$，第 n 个周期中的第一个奇数是 $2p(n-1)+1$。

给定输入 x，我们先确定包含 x 的周期的编号 n，然后确定指标 i，使得如果 x 为奇数，那么 x 就是其所在周期中的第 i 个奇数项。

(1) 如果 x 为偶数，那么 $n = \lfloor \frac{x}{2} \rfloor$。(2) 如果 $x = 1$，那么 $i = 0$ 和 $n = 2 - f(0)$。(3) 如果 $x > 1$ 为奇数，我们就寻找满足 $x = 2p(n-1) + 2i + 1$ 的 $n < x$ 和 $i < x$。

注意，对于任意 s，不可能有 $f(s) = f(s+1) = 1$。这说明了第 2 节中的输出以及界 $n < x$。

最后，我们计算 $s_K(x) = 2n + 2$，并且

$$c_K(x) = \begin{cases} x & \text{if } x \text{ is even and } f(n) = 1, \\ 2p(n-1)+1 & \text{if } x \text{ is even and } f(n) > 1, \\ x+2 & \text{if } x \text{ is odd and } i+2 \neq f(n), \\ 2n & \text{if } x \text{ is odd and } i+2 = f(n). \end{cases}$$

□

定理46。 集合 $\{s_K, c_K\}$ 是点标准性的基。

证明。固定一个副本 $\mathcal{A}$，使得 $s_K^{\mathcal{A}}$ 和 $c_K^{\mathcal{A}}$ 是原始递归的。令 $c_{\mathcal{A}}$ 为从$\mathcal{S}$到 $\mathcal{A}$的唯一同构。由于 $s_K^{\mathcal{A}}$和 $c_K^{\mathcal{A}}$是原始递归的，我们可以选取 n，使得 $\mathcal{B}_n = (\mathbb{N}, 0^{\mathcal{A}}, s_K^{\mathcal{A}}, c_K^{\mathcal{A}})$。显然，$\mathcal{A}_f = (\mathbb{N}, 0, s_K, c_K)$通过 $c_{\mathcal{A}}$与 $\mathcal{B}_n$同构。根据 f的构造，这意味着 $\mathcal{B}_n$和 $\mathcal{A}_f$逐点同构。因此， $c_{\mathcal{A}}^{-1}$ 也是原始递归的。由观察5可知， $\mathcal{A}$是准时标准的。□

定理15。 初等函数类的每个代入基都是点标准性的基。

证明。 任取初等函数的一个代入基 B 以及一个 $\mathcal{A}$ 的副本 $\mathcal{S}$，使得 B 中的每个函数在 $\mathcal{A}$上都是原始递归的。于是，s_K 和 c_K 在 $\mathcal{A}$ 上也都是原始递归的， B 因为它们可以通过代入从定理46 我们得出 $\mathcal{A}$ 是准时标准的。□

References

[1] 威廉·阿克曼。《关于希尔伯特对实数的构造》。载于：《数学年刊》99（1928年），第118–133页。 DOI: 10.1007/BF01459088。[2] Matthew Askes 和 Rod Downey。《在线、可计算与点态结构理论》。载于：IGPL逻辑期刊 31.6（2022年），第1251–1293页。 DOI: 10.1093/jigpal/jzac065。[3] Mathias Barra 和 Philipp Gerhardy。《Skolem 超运算是良序的》。载于：《数学理论与计算实践》。第5635卷。《计算机科学讲义》。柏林、海德堡：施普林格，2009年，第11–20页。 DOI: 10.1007/978-3-642-03073-4_2。[4] Nikolay Bazhenov、Rod Downey、Iskander Kalimullin 和 Alexander Melnikov。《在线结构理论基础》。载于：《符号逻辑通报》25.2（2019年），第141–181页。 DOI: 10.1017/bsl.2019.20。[5] Nikolay Bazhenov、Iskander Kalimullin、Alexander Melnikov 和 Keng Meng Ng。《有限生成结构的在线表示》。载于：《理论计算机科学》844（2020年），第195–216页。 DOI: 10.1016/j.tcs.2020.08.021。[6] Nikolay Bazhenov 和 Dariusz Kaloci´nski。《度谱与记号的相对可接受性》。载于：第31届EACSL计算机科学逻辑年会（CSL 2023），波兰华沙，2023年2月13日至16日。Bartek Klin 和 Elaine Pimentel 编。第252卷。LIPIcs。Schloss Dagstuhl - 莱布尼茨信息学中心，2023年，11:1–11:20。 DOI: 10.4230/LIPICS.CSL.2023.11。[7] Wojciech Czerwinski 和 Lukasz Orlikowski。《向量加法系统中的可达性问题是阿克曼完备的》。载于：第62届IEEE计算机科学基础年会（FOCS 2021），美国科罗拉多州丹佛，2022年2月7日至10日。IEEE，2021年，第1229–1240页。 DOI: 10.1109/FOCS52979.2021. 00120。

[8] Marina Dorzhieva 和 Ellen Hammatt。《格嵌入与点线性序》。载于：《符号逻辑杂志》（2025年），在线发表。 DOI: 10.1017/jsl.2025.10106。[9] Rod Downey、Alexander Melnikov 和 Keng Meng Ng。《在线结构理论基础第二卷：算子方法》。载于：《计算机科学中的逻辑方法》17.3（2021年7月），6:1–6:35。 ISSN: 1860-5974。 DOI: 10.46298/lmcs-17(3:6)2021。[10] Balthasar Grabmayr。《结构与计算》。载于：Noˆus（2025年）。 DOI: 10.1111/nous.70023。[11] Iskander Kalimullin、Alexander Melnikov 和 Maxim Zubkov。《点次数与格嵌入》。载于：《计算与自动机理论及其应用专题》。新加坡：世界科学出版社，2024年，第315–334页。 DOI: 10.1142/9789811278631_0012。[12] Iskander Sh. Kalimullin、Alexander G. Melnikov 和 Keng Meng Ng。《可即时计算的代数结构》。载于：《理论计算机科学》674（2017年），第73–98页。 DOI: 10.1016/J.TCS.2017. 01.029。[13] L´aszl´o Kalm´ar。《一个简单的不可判定算术问题示例》。载于：《数学与物理学报》50（1943年），第1–23页。[14] Hilbert Levitz。《表示最小 ε数的算术函数有序集》。载于：《数学逻辑与数学基础杂志》21（1975年），第115–120页。[15] Stefano Mazzanti。《原始递归函数类的朴素基》。载于：《数学逻辑季刊》48.1（2002年），第93–104页。[16] Albert R. Meyer 和 Dennis M. Ritchie。《循环程序的复杂性》。载于：美国计算机协会第22届全国会议论文集，1967年，华盛顿特区。Solomon Rosenthal 编。美国计算机协会，1967年，第465–469页。 DOI: 10.1145/800196.806014。[17] Piergiorgio Odifreddi。《经典递归论：自然数函数与集合理论》。第125卷。《逻辑与数学基础研究》。阿姆斯特丹：北荷兰出版社，1989年。 ISBN: 978-0-444-87295-1。[18] Piergiorgio Odifreddi。《经典递归论：第二卷》。第143卷。《逻辑与数学基础研究》。阿姆斯特丹：北荷兰出版社，1999年。[19] Gregory A. Riccardi。《抽象程序系统中控制结构的独立性》。载于：《计算机与系统科学杂志》22.2（1981年），第107–143页。 DOI: 10.1016/0022-0000(81) 90024-6。[20] Hartley Rogers。《递归函数与有效可计算性理论》。马萨诸塞州剑桥：麻省理工学院出版社，1987年。[21] H. E. Rose。《次递归：函数与层级》。纽约：牛津大学出版社，1984年。[22] James S. Royer。《程序结构的内涵理论》。第273卷。《计算机科学讲义》。柏林：施普林格，1987年。 DOI: 10.1007/3-540-18253-5。[23] Stewart Shapiro。《可接受记号》。英语。载于：《圣母大学形式逻辑杂志》23.1（1982年），第14–20页。 ISSN: 0029-4527。 DOI: 10.1305/ndjfl/1093883561。[24] Stewart Shapiro、Eric Snyder 和 Richard Samuels。《可计算性、记号与数的实在知识》。载于：《哲学》7.1（2022年）。 DOI: 10.3390/philosophies7010020。[25] John C. Shepherdson 和 Howard E. Sturgis。《递归函数的可计算性》。载于：《美国计算机协会杂志》10.2（1963年），第217–255页。 DOI: 10.1145/321160.321170。[26] Thoralf Skolem。《表示最小 ε数的算术函数有序集》。载于：《挪威皇家科学学会会报》29（1956年），第54–59页。

(BAZHENOV) Novosibirsk State University, Novosibirsk, Russia and Innopolis University, Innopolis, Russia*Email address*: bazhenov@math.nsc.ru*URL*: https://bazhenov.droppages.site(GEORGIEV) Faculty of Mathematics and Informatics, Sofia University, Bulgaria*Email address*: ivandg@fmi.uni-sofia.bg(Kaloci´nski) Institute of Computer Science, Polish Academy of Sciences, Poland*Email address*: dariusz.kalocinski@gmail.com*URL*: https://www.dariuszkalocinski.com

(VATEV) Faculty of Mathematics and Informatics, Sofia University, Bulgaria
Email address: stefanv@fmi.uni-sofia.bg*URL*:
https://fmi.uni-sofia.bg/fmi/logic/stefanv/

（弗罗茨瓦夫斯基） Faculty of Philosophy, University of Warsaw,
Poland*Email address*: m.wroclawski@uw.edu.pl