arXiv:2301.07544v1 [math.LO] 18 Jan 2023

从 GTC 到重置：从循环证明系统生成重置证明系统

Graham E. Leigh，哥德堡大学；
Dominik Wehr，哥德堡大学

2023年1月19日

我们研究循环证明系统，其中推导是图而非树。此类系统通常附带一个条件，用来判定哪些推导可被接纳为“证明”，这称为可靠性条件。该可靠性条件通常采用全局迹条件或重置条件的形式：前者是依赖于证明图中所有无限路径的性质；后者则是一种“局部”条件，仅依赖于简单环，因此通常在更多证明变换下保持稳定。

在本文中，我们提出一种通用方法，用于从具有全局迹条件的循环证明系统构造具有重置条件的循环证明系统。与以往的方法不同，这种生成方法完全独立于逻辑语义，只依赖于“迹”和“进展”概念的组合方面。我们运用此方法，为文献中的三个循环证明系统提出重置证明系统：循环算术 [22], 循环 G¨odel 的 T [9] 以及模态 μ-演算 [16]的循环表列。

目录

1 引言 2

2 文章概述 3

3 预备知识 5
3.1 循环证明系统.5 3.2 轨迹范畴.7 3.3 自动机理论.10

4 Safra 棋盘 11 5 重置证明系统 22 5.1 可靠性.24 5.2 完备性与证明搜索.27 6 推导具体重置证明系统 30 6.1 皮亚诺算术.30 6.1.1 循环算术.30 6.1.2 重置算术.32

本研究得到克努特和爱丽丝·瓦伦堡基金会 [2020.0199, 2015.0179] 的支持。

6.2 G¨odel 的 T.37 6.2.1 循环 G¨odel 的 T.37 6.2.2 重置 G¨odel 的 T.39 6.3 模态 μ-演算.41 6.3.1 循环模态 μ-演算.42 6.3.2 $\mathbb{F}$-重置模态 μ-演算.44 6.3.3 $\mathbb{B}$-重置模态 μ-演算.49 6.3.4 Jungteerapanich-Stirling 系统.52

7 结论 55

1 引言

在循环证明中，叶子可以标注为对先前推导步骤的“递归”引用，而非公理，因此推导呈现为有限图的形状，而不只是（有限）树。允许此类证明的证明系统尤其适用于包含不动点或（余）归纳定义的概念的逻辑（参见例如 [8, 16, 22, 23]）。由于循环证明可能存在无限分支，其可靠性通常无法归结为系统公理的可靠性及其规则的真值保持。在这种情况下，必须施加额外的可靠性条件。最常见的此类条件称为全局轨迹条件：若循环推导中的每条无限分支都包含一条无限进展的轨迹，则该推导是可靠的。轨迹和进展的确切定义因证明系统而异。在 Simpson 的循环算术 [22] 中，轨迹是沿推导分支延伸的项序列；当项的值递减时，便认为轨迹取得了进展。相比之下，在具有不动点的逻辑所对应的循环证明系统中，轨迹通常是公式序列，而进展则基于不动点展开。

全局轨迹条件在文献中十分常见，因为它通常很容易适配新的循环证明系统，并且可以直接证明其可靠性。然而，它也有一些缺点。验证一个循环推导是否满足全局轨迹条件，从而确认它是否构成证明，一般来说是 PSPACE 完全的 [6]。此外，全局轨迹条件的“全局”性质——即证明中的“局部”变化可能会影响“全局”可靠性——往往使其不适用于证明理论研究。事实上，除了可靠性和完备性之外，循环证明理论的大多数结果都是针对采用其他可靠性条件的系统得出的，例如重置证明 [3–5, 15], 、归纳序 [24] 和弹跳线程 [7]。

重置证明是循环证明，它们不采用全局轨迹条件，而是通过为序列式添加标注，并使用一种用于标记“进展”的特定“重置”推理规则来实现所需机制。这种重置条件局限于证明底层图中的每个简单环。这既简化了证明检查，将问题降至多项式时间（通常甚至是线性时间），也使得推理证明变换更加容易。文献中最早出现的重置证明系统

是 Jungteerapanich [12]提出的模态 μ-演算表列系统。此后，研究者为模态 μ-演算的无交替片段 [15],、一阶 μ-演算 [1],、带有主模态的模态逻辑 [20] 以及完全计算树逻辑 CTL* [2]设计了类似的证明系统[1] 。尽管已有研究指出，可以通过使用 Safra 自动机为序列式添加标注，从基于全局轨迹的循环证明系统构造重置证明系统 [21] （例如参见 [12, 15]），但上述所有重置证明系统都是作为 Jungteerapanich 系统的变体设计的，而非“直接”通过 Safra 构造得到。

本文将 Safra 自动机与重置证明系统之间的联系形式化。通过采用关于迹条件及其所诱导的循环证明系统的恰当抽象概念，我们证明，任何由全局轨迹条件给出的循环证明系统都会自然地诱导出一个等价的重置证明系统。这一结果不仅停留在抽象层面，还提供了一套“配方”，可从任何合适的循环证明系统推导出相应的重置证明系统，即使该系统与模态 μ-演算无关也是如此。我们采用的迹条件和循环证明系统的抽象概念足够一般，能够涵盖循环证明系统相关文献中研究的大多数证明系统。我们将此方法应用于文献中的两个循环证明系统：循环算术 [22] 和循环 G¨odel 的 T [9],，并在每种情况下得到一个等价的重置式证明系统。将这一构造应用于模态 μ-演算的循环证明时，根据所采用的迹的概念，会诱导出不同的重置证明系统。后一类系统有助于说明我们的方法与 Jungteerapanich 和 Stirling 所采用方法之间的区别。

2 文章概述

在第3节中，我们给出支撑并引出本文其余内容的定义。我们依赖的两个核心概念是循环证明系统的一种抽象表述（第3.1节），以及一种用所谓的激活代数来描述循环证明系统全局轨迹条件的方法（第3.2节）。在第3.3节中，我们回顾了一些关于无穷字自动机和树自动机的定义与结果，本文始终会用到这些内容。

循环证明系统由两个部分组成：一组推导规则和一个可靠性条件。给定一组推导规则，通过在一个“树形”的循环图上标注这些规则的实例，即可得到一个预证明。可靠性条件是针对这类预证明设定的条件，用于区分循环证明系统中的证明与单纯的预证明。另一个重要概念是循环证明系统同态：给定两个循环证明系统 R 和 S，同态 $f\colon \mathrm{R} \to \mathrm{S}$ 大致包含一种将 R-序列式转换为 S-序列式的方式，使得经过这种转换后，R 的推导规则在 S 中可容许。关键的是，这样的同态 $f\colon \mathrm{R} \to \mathrm{S}$ 可以将 R-证明转换为 S-证明，从而建立两个证明系统之间的联系。本文的核心构想是，对于某些循环证明系统 R，构造一个

[1]严格来说，并非所有这些系统都是我们所说的重置证明系统。有关更多细节，请参见第7节中关于相关工作的讨论。

与 R“相对应”的重置证明系统 R(R)。我们通过构造合适的双向同态，将这种“对应”形式化。

本文的第二个核心概念——一种为抽象循环证明规定全局迹条件的方法——将在第 3.2 节回顾。预证明上的迹条件会在每个序列式中确定一组迹对象，并在循环证明的每条无限分支上，确定一组连接这些迹对象的迹。沿着这些迹，存在一种累积的“进展”概念，并通过激活代数加以规定。如果每条无限分支上都存在一条无限次取得进展的迹，则该预证明被视为证明。文献中绝大多数循环证明系统的可靠性条件，都是这种全局迹条件。

本文将证明，任何其可靠性条件可实现为此类全局迹条件的循环证明系统，都可以关联到一个重置证明系统，即其可靠性条件完全由简单环决定的循环证明系统。给定一个可靠性条件由激活代数 A 规定的循环证明系统 R，重置系统 R(R) 的构造方法是：为循环证明系统的推导元素（即“序列式”）添加*Safra* 棋盘标注。Safra 棋盘将在第 4 节介绍，其灵感源自用于确定化无限字自动机的 Safra 构造 [21]，具体采用了 [14] 中的表述。大致而言，对于含有迹对象 X 的给定序列式，其 Safra 棋盘由若干“方格”$(x, a) \in X \times \mathrm{A}$ 组成，方格上放置着游戏筹码的栈，用以“跟踪”迹值的进展。给定一个以 Γ 为结论、以 Γ' 为其某个前提的 R-推导规则，相关规则会说明如何移动并扩展棋盘上的栈，以便从 Γ 的棋盘得到 Γ' 的棋盘，同时考虑从 Γ 到 Γ' 的迹步骤中所取得的“进展”。此外，还可以对这些 Safra 棋盘执行某些簿记操作，包括重置操作，它会“重置”Safra 棋盘上累计的部分进展。借助这一机制，迹条件得以简化：如果每条无限分支上都有无穷多个重置步骤发生，则该预证明就是证明，这表明某个迹值沿该分支无限次取得进展。与全局迹条件不同，只需验证每个预证明的单个简单环都包含适当的重置，即可确立这一性质，从而得到一种“局部”可靠性条件。

对于带有以激活代数 A 定义的全局轨迹条件的循环证明系统 R，重置系统 R(R) 在第 5 节中定义。我们证明了 R 与 R(R) 之间的两个关键性质：**可靠性** 和**完备性**。可靠性指出，在重置系统 R(R) 中可证明的任何带标注序列式，在原系统 R 中都可以不带标注地证明。反过来，完备性是指在 R 中可证明的任何序列式，在 R(R) 中也可证明。这两个性质的名称恰如其分，因为它们使得相对于某种语义而言，R 的可靠性和完备性能够“提升”到 R(R)。这两个结果均通过构造合适的循环证明系统同态得以确立。在第 5.1 节中，我们考察转换 *strip*：$\mathrm{R(R)} \to \mathrm{R}$，它会从 R(R) 证明中剥除 Safra 棋盘标注，并删除与各种 Safra 棋盘簿记操作对应的推导步骤。通过证明这是一个同态，我们可以得出结论：每个 R(R) 证明都会自然地导出一个相应的 R 证明，证明“相同”的序列式，从而得到可靠性。第 5.2 节中的完备性证明则

不那么直接：对于 R 的每个有限子系统 F，我们定义 R(R) 的一个有限子系统 S(F)，它具有证明搜索性质：对于每个可通过循环证明 Π 在 F 中证明的序列式 Γ，S(F) 中都存在 Π 的一个带标注的有限展开，它是 Γ 的证明。由于 S(F) 可以通过同态*embed*：S(F) $\to$ R(R) 嵌入 R(R)，因此这就给出了在重置系统中对 Γ 的证明。

第6节将上述结果应用于文献中的多种循环证明系统，以构造相应的重置系统：皮亚诺算术（第6.1节）、G¨ Gödel 的 T（第6.2节）和模态 μ-演算（第6.3节）。虽然系统 R(R) 对任何合适的循环证明系统都具有可靠性和完备性，但它往往不太容易“使用”。通常，只需进行一些人体工学调整，就能缓解这种情况。这正是我们在第6节所做的事：对于上述每个具体的循环系统 R，我们都设计了一个定制的重置系统 S，它以 R(R) 为“灵感”。借助一对同态嵌入 : S $\to$ R(R) 和扩展 : S(F) $\to$ S，可以得到 S 的可靠性和完备性。重要的是，依托前几节的结果，构造这些定制系统 S 以及同态嵌入 和扩展 几乎不费什么功夫。我们希望这些示例足够有启发性，让读者能够将同样的方法应用于自己选择的任何合适的循环证明系统。

我们将在第7节以简短的结论、相关工作的概述以及对未来研究的展望作结。

3 预备知识

3.1 循环证明系统

我们首先对循环证明系统给出一个适当的抽象描述。由于我们采用了非常宽泛的可靠性条件概念，因此我们所了解的每个循环证明系统都是这种循环证明系统概念的一个实例。

循环证明和预证明是某些有限图，其节点依据某个推导系统标记为序列式。我们不使用一般图，而是使用循环树作为循环证明概念底层的数据结构。在某些定义和证明中，循环树已被证明稍微更加方便。树是一个非空集合$T \subseteq \omega^*$，并且该集合对取前缀封闭。每个 $t \in T$称为一个节点，而 Chld(t) $:=$ {$ti \in T \mid i \in \omega$}中的节点称为其子节点。如果 Chld($t$) $= \emptyset$，则节点 t是 T的叶节点；否则是内部节点。循环树是一个二元组（T， β），由有限树 T和偏函数 β: Leaf (T) ¤$\to$ Inner(T)组成，该偏函数将 T的某些叶节点映射到 T的内部节点，并且对于每个 $t \in$ dom(β)，$\beta(t) < t$满足前缀序。如果 $t \in$ dom(β)，则称其为芽，而 $\beta(t)$称为其伴随节点。

定义 3.1 一个推导系统是一个三元组 (Seq，R ρ)，由一个序列式对 Seq、一个推导规则集合 R 以及一个规则解释 ρ: R $\to$ Seq* 组成，使得对于每个 $R \in$ R，$\rho(R) = (\Gamma, \Delta_1, \ldots\ldots, \Delta_{n-1}) \in$ Seq^n 对于 $n > 0$。序列式 Γ 是 R 的结论，而 Δ_i 是其前提。下文中，我们将推导系统 (Seq，R ρ) 简称为 R。

R-预证明 是一个三元组 $\Pi = (C, \lambda, \delta)$，由一棵循环树 $C = (T, \beta)$ 以及一个

标记 λ: $T \to$Seq，使得对于每个 $t \in$dom(β)，都有 $\lambda(t) = \lambda(\beta(t))$；以及一个偏函数 δ: ($T \setminus$ dom(β)) ⇀R，使得对于每个 $t \in T \setminus$ dom(β)

- 要么 $t \in$dom(δ) 满足 $\rho(\delta(t)) = (\Gamma, \Delta_1, \ldots\ldots, \Delta_n)$ 且 $\lambda(t) = \Gamma$，并且 Chld(t) ={$t1, \ldots\ldots, tn$} 且 $\lambda(ti) = \Delta_i$
- 要么 $t \in$Leaf (T)。

用 Pp(R) 表示 R-预证明的集合。序列式 $\lambda(\varepsilon)$ 称为 Π的末序列。每个叶节点 $o \in$Leaf (T) \ dom(δ) 称为开放的，其关联的序列式 $\lambda(o)$ 是 Π的一个假设。

一个循环证明系统是一个三元组 (Seq，R ρ，Pfs)，由一个推导系统 (Seq，R ρ) 以及集合 Pfs $\subseteq$ Pp(R) 构成；后者是没有假设的 R-预证明，称为 R-证明。任何以 Γ 为末序列的 Π ∈ Pfs 都称为 Γ的一个证明。这样的预证明称为满足 R 的可靠性条件。我们将推导系统的命名约定扩展到循环推导系统，将 (Seq，R ρ，Pfs) 记为 R。⌟

循环证明系统之间的证明态射与预证明态射在本文的结果中发挥着关键作用。本质上，预证明态射 f: R $\to$ R′ 表明 R 的推导规则在 R′ 中是可容许的。由此可得到一种将 R-预证明转换为 R′-预证明的方法：只需将 R-证明中每次应用推导规则的步骤替换为见证其可容许性的 R′-预证明，即可得到一个 R′-预证明。如果该方法能将所有 R-证明转换为 R′-证明，f: R $\to$ R′ 就称为证明态射。

定义 3.2 设 (Seq, R ρ, Pfs) 和 (Seq′, R′, ρ', Pfs′) 为循环证明系统。预证明态射 f: R $\to$ R′ 由一个函数 f_0: Seq $\to$ Seq′ 和一个函数 f_1: R $\to$ Pp(R′) 组成；前者将 R-序列映射到 R′-序列，后者为每条 R-规则指定一个 R′-预证明。此外，这两个函数必须相互协调：对于 $R \in$R，其中 $\rho(R) = (\Gamma, \Delta_1, \ldots, \Delta_n)$，预证明 $f_1(R)$ 必须以 $f_0(\Gamma)$ 为末序列，并以 $f_0(\Delta_1) \ldots, f_0(\Delta_n)$ 为假设。下文中，我们将 f_0: Seq $\to$ Seq′ 和 f_1: R $\to$ Pp(R′) 均记作 f。⌟

用预证明复合来形式化描述将 R-预证明转换为由 f 诱导的 R′-预证明（即 R $\to$ R′）的方法，最为简便。因此，假设 R 是一个循环推导系统，并且 $\Pi = ((C, \beta)\ \lambda, \delta)$ 是一个带有开放叶节点 $o_1, \ldots, o_n$ 的 R-预证明。此外，假设存在 R-预证明 $\Pi_1 = ((T_1, \beta_1)\ \lambda_1, \rho_1) \ldots, \Pi_n = ((T_n, \beta_n)\ \lambda_n, \rho_n)$，使得 Π_i 的末序列是 $\lambda(o_i)$，其假设为 $\Xi^i_1, \ldots, \Xi^i_{m_i}$。那么，可以将这些材料复合成一个末序列为 Γ、假设为以下内容的预证明 $\Pi[\Pi_1, \ldots, \Pi_n] = ((T_c, \beta_c)\ \lambda_c, \delta_c)$

$\Xi_1^1, \ldots \Xi_{m_1}^1, \Xi_1^2, \ldots, \Xi_{m_2}^2, \ldots, \Xi_1^n, \ldots, \Xi_{m_n}^n$ 如下：

$$T_c \coloneqq T \cup \bigcup_{i=1}^{n} \{l_i t \mid t \in T_i\}$$

$$\lambda_c(t) \coloneqq \begin{cases} \lambda(t) & t \in T \\ \lambda_i(s) & t = o_i s \text{ for } s \in T_i \end{cases}$$

$$\delta_c(t) \coloneqq \begin{cases} \delta(t) & t \in \mathrm{dom}(\delta) \\ \delta_i(s) & t = o_i s \text{ for } s \in T_i \text{ and } s \in \mathrm{dom}(\delta_i) \end{cases}$$

$$\beta_c(t) \coloneqq \begin{cases} \beta(t) & t \in \mathrm{dom}(\beta) \\ o_i \beta_i(s) & t = o_i s \text{ for } s \in T_i \text{ and } s \in \mathrm{dom}(\beta_i) \end{cases}$$

假设存在一个预证明态射 $f\colon \mathrm{R} \to \mathrm{R}'$，以及一个 R-预证明 $\Pi = (T, \beta)\ \lambda, \delta)$，其末序列为 Γ，假设为 $\Delta_1, \ldots, \Delta_n$。由此可导出一个 R′-预证明 $f(\Pi)$，其末序列为 $f(\Gamma)$，假设为 $f(\Delta_1) \ldots, f(\Delta_n)$。该预证明通过递归定义在 T 上：为每个节点 $t \in T$ 关联一个预证明 Π_t，其末序列为 $\lambda(t)$。首先，对每个 $t \in \mathrm{Leaf}(T) \setminus \mathrm{dom}(\delta)$，设置预证明 $\Pi_t \coloneqq (\{\varepsilon\}\ \varepsilon \mapsto \lambda(t)\ \emptyset)$，即将 $\lambda(t)$ 推导为一个开放叶节点。现在，对于每个 $t \in \mathrm{dom}(\delta)$，若所有 $\{t_1, \ldots, t_n\} = \mathrm{Chld}(t)$ 都关联了预证明 Π_{t_i}，则定义 $\Pi'_t \coloneqq \Pi_\delta(t)\ [\Pi_{t_1}, \ldots, \Pi_{t_n}]$（其中 $\Pi_\delta(t)$ 由态射 $f\colon \mathrm{R} \to \mathrm{R}'$ 给出）。若 $t \notin \mathrm{im}(\beta)$，则 $\Pi_t = \Pi'_t$。否则，Π_t 由 Π'_t 得到：将对应于 $\beta^{-1}(t)$ 中叶节点的 Π'_t 各开放叶节点上的 β-循环添加到 ε。然后 $f(\Pi) \coloneqq \Pi_\varepsilon$。

定义 3.3 设 (Seq、R ρ、Pfs) 和 (Seq′、R′、ρ'、Pfs′) 是循环证明系统。预证明态射 $f\colon \mathrm{R} \to \mathrm{R}'$ 如果保持 R 的可靠性条件，则称为证明态射。也就是说，对于每个 $\Pi \in \mathrm{Pfs}$，都有 $f(\Pi) \in \mathrm{Pfs}'$。⌟

3.2 轨迹范畴

在上一节中，我们对可靠性条件的描述相当模糊。本节将介绍一种可靠性条件：全局轨迹条件。更具体地说，我们将用某些范畴来描述一种指定全局轨迹条件的通用方法。随后，我们将定义一族这样的范畴，它们足以指定文献中大多数的全局轨迹条件。本节给出的定义改编自 [6] 和 [26]。

记由 ω 按 $\leq$（由 ω 按 $<$）诱导的 ω 预序范畴（半范畴）为 $n < m$，并用 $n < m$ 表示它们的（非恒等）态射。固定一个范畴 T。穿过 T 的路径是一个函子 $P\colon \omega \to \mathrm{T}$。给定路径 $P, P'\colon \omega \to \mathrm{T}$，若存在一个半函子 $S\colon \omega \to \omega$（即严格单调映射 $S\colon \omega \to \omega$），使得 $P = P' \circ S$，则称 P 是 P' 的子路径，记作 $P \subseteq P'$。换言之，P' 的子路径 P 可以（a）“丢弃” P' 的一个有限前缀（例如 $P(0) = P'(k)$），以及（b）合并 P' 的多个“步骤”（例如 $P(i < i+1) = P'(j+n-1 < j+n) \circ \ldots \circ P'(j < j+1)$）。$\subseteq$ 的传递对称闭包记作 $\sim$。

迹范畴是一个带有路径条件的范畴，且该条件在子路径下保持不变。这一一般概念涵盖了循环证明理论文献中发现的大多数“轨迹”概念。

定义 3.4 迹范畴是一个范畴 T，连同路径上的一个条件 $C_{\mathcal{T}}$，称为迹条件，在 $\sim$ 下保持不变；也就是说，若 $P \sim P'$，则 $C_{\mathcal{T}}(P)$ 当且仅当 $C_{\mathcal{T}}(P')$ 成立。」

迹解释用迹范畴 T 来刻画推导系统 R 中迹和进展的概念。R 的序列式被视为 T 的对象，而 R 的推导规则中从结论到前提的每个“步骤”，都与一个态射相关联，该态射连接分别对应于该结论和前提的对象。在这种解释下，穿过预证明的每条分支都会诱导出一条路径 $\omega \to \mathrm{T}$，从而可以根据 T 的迹条件，对全局轨迹条件作出一般性规定。文献中针对循环证明系统的此类迹解释的多个示例，可见第6节和 [26, 第 6]章。

定义 3.5 设 $(\mathrm{Seq}, \mathrm{R}\ \rho)$ 为一个推导系统，T 为一个迹范畴。迹解释 $\iota\colon \mathrm{R} \to \mathrm{T}$ 由一个映射 $\iota\colon \mathrm{Seq} \to \mathrm{Ob}(\mathrm{T})$ 构成，并且对于每条规则 $r \in \mathrm{R}$，若 $\rho(r) = (\Gamma, \Delta_1, \ldots, \Delta_n)$，则对每个 $1 \le i \le n$，都有态射 $r_i\colon \iota(\Gamma) \to \iota(\Delta_i)$。」

固定一棵循环树 $C = (T, \beta)$。序列 $\pi \in T^\omega$ 是一条经过 C 的分支，当且仅当 $\pi_0 = \varepsilon$，并且对每个索引 $i \in \omega$ 都满足以下性质：(a) 若 $\pi_i \notin \mathrm{Leaf}(T)$，则 $\pi_{i+1} \in \mathrm{Chld}(\pi_i)$；(b) 若 $\pi_i \in \mathrm{Leaf}(T)$，则 $\pi_i \in$ 定义域(β) 且 $\pi_{i+1} = \beta(\pi_i)$。

定义 3.6 设 $(\mathrm{Seq}, \mathrm{R}\ \rho)$ 是一个带有轨迹解释 $\iota\colon \mathrm{R} \to \mathrm{T}$ 的推导系统。设 $\Pi = (C, \lambda, \delta)$ 是一个预证明，且 π 是一条经过 C 的路径。这将导出一条路径 $\hat{}\pi\colon \omega \to \mathrm{T}$，其定义如下：

$$\hat{\pi}(i) := \iota(\lambda(\pi_i)) \qquad \hat{\pi}(i < i{+}1) := \begin{cases} \delta(\pi_i)_j\colon \iota(\lambda(\pi_i)) \to \iota(\lambda(\pi_{i+1})) & \pi_i \notin \mathrm{Leaf}(T) \text{ and } \pi_{i+1} = \pi_{i+1} j \\ 1_{\hat{\pi}(i)} & \pi_i \in \mathrm{dom}(\beta) \end{cases}$$

这将导出一个循环证明系统 $\iota(\mathrm{R}) := (\mathrm{Seq}, \mathrm{R}\ \rho, \mathrm{Pfs})$，其中

$$\mathrm{Pfs} := \{\Pi \in \mathrm{Pp}(\mathrm{R}) \mid \text{对于每条经过 } \Pi \text{ 的路径 } \pi, \text{ 迹条件 } C_{\mathcal{T}}(\hat{}\pi) \text{ 均成立}\}$$ 」

前一定义中描述的这类可靠性条件是一种全局轨迹条件：乍看之下，需要检查 T 的每一条无限分支——在非平凡情形下，这样的分支有连续统多个——是否都满足迹条件。在“足够有限”的实例中（循环证明理论文献中的实例几乎全都属于此类），借助拉姆齐定理，验证范围可以缩小到有限个周期路径（例如参见 [6, 定理 3]）。即便如此，这仍是一个复杂的验证过程，其复杂度很快就会超出人类检查证明的能力。重置证明系统是采用另一种可靠性条件的循环证明系统。虽然我们尚未发现与定义 3.6 类似的、对重置可靠性条件的抽象论述，但重置证明系统仍具有一些显著特征。重置条件通常通过注释机制来“跟踪进展”，以用于序列式

。该条件要求在每个简单环上都应用重置规则，以重置所跟踪的部分进展；也就是说，在预证明中，对于 β定义域中的每个 $s \in$，都要沿着 $\beta(s)$ 与 s 之间的路径应用该规则。因此，要验证预证明是否满足这类重置条件，只需分别分析每个这样的简单环，由此得到一种“局部”可靠性条件，与上文定义的全局轨迹条件相对。然而，与采用全局轨迹条件的证明系统相比，重置证明系统往往需要大得多的证明（关于这一点较为一般的论述，见引理 4.21）。

接下来，我们给出一族由激活代数诱导的迹范畴 $\mathrm{T}_{\mathcal{A}}$。文献中大多数全局迹条件都可以用适当代数 A 对应的 $\mathrm{T}_{\mathcal{A}}$ 来描述。

定义 3.7 激活代数 $\mathrm{A} = (A, \leq, \vee, 0, \alpha)$ 是一个有限半格 $(A, \leq, \vee, 0)$，并带有一个固定的激活元素 $\alpha \in A$，其中 $0 \neq \alpha$。

A-激活迹范畴 $\mathrm{T}_{\mathcal{A}}$ 以有限集作为对象。集合X, Y之间的态射是所有关系 $R \subseteq X \times \mathrm{A} \times Y$。给定态射 $R\colon X \to Y, R'\colon Y \to Z$，它们的复合定义为

$$(x, c, z) \in R' \circ R \qquad \text{iff} \qquad \exists y \in Y. \exists a, b \in A.\ (x, a, y) \in R,\ (y, b, z) \in R' \text{ and } a \vee b = c$$

恒等　　态射为 $1_X := \{(x, 0, x) \mid x \in X\}$。我们常写 $xR^a y$表示 $(x, a, y) \in R$　　.

$\mathrm{T}_{\mathcal{A}}$ 的迹条件定义如下：路径 $P\colon \omega \to \mathrm{T}_{\mathcal{A}}$ 满足迹条件，当且仅当存在子路径 $P' \subseteq P$以及一个无穷序列 σ，使得对沿该路径的每个 $i \in \omega$，都有 $\sigma(i) \in P'(i)$，并且对所有 $i \in \omega$，都有 $\sigma_i P' (i < i+1)^{\alpha} \sigma_{i+1}$。⌟

例 3.8 布尔值 $\mathbb{B} := \{0,1\}$ 构成一个激活代数，其并运算为通常的连接运算，并且 $\alpha := 1$。它们对应于文献中最常见的全局轨迹条件形式：轨迹具有进展点（在 $\mathrm{T}_{\mathbb{B}}$ 的映射中由三元组 $(x, 1, y)$ 表示），若一条路径包含无穷多个进展点，则满足迹条件。很容易验证，$\mathrm{T}_{\mathbb{B}}$ 的迹条件恰好就是这一条件。

激活代数的另一个例子是三值失败代数 $\mathbb{F} := \{0,1,2\}$，其并运算为最大值，并且 $\alpha := 1$。$\mathrm{T}_{\mathbb{F}}$ 的迹条件对应于这样的全局轨迹条件：若轨迹包含无穷多个进展点（在 $\mathrm{T}_{\mathbb{F}}$ 的映射中由三元组 $(x, 1, y)$ 表示）且不包含失败点（在 $\mathrm{T}_{\mathbb{F}}$ 的映射中由三元组 $(x, 2, y)$ 表示），则满足迹条件。同样，$\mathrm{T}_{\mathbb{F}}$ 的迹条件恰好确保了这一条件。失败代数在文献中是模态 μ-演算的一种常见轨迹条件（参见定义 6.16）。

轨迹范畴 $\mathrm{T}_{\mathcal{A}}$ 是研究循环证明理论的天然媒介。它们足够抽象，能够涵盖文献中的许多轨迹条件；同时又足够具体，因此可以为其推导出循环证明理论中的各种定理，例如下面的可判定性结果。本文将展示，如何为轨迹解释 $\iota\colon \mathrm{R} \to \mathrm{T}_{\mathcal{A}}$ 的循环证明系统 $\iota(\mathrm{R})$ 精确构造重置证明系统。

命题 3.9 固定一个推导系统 R 和一个 R-预证明 Π。给定一个轨迹解释$\iota\colon \mathrm{R} \to \mathrm{T}_{\mathcal{A}}$，可以判定 Π 是否是 $\iota(\mathrm{R})$ 中的证明。

证明 证明这一点有多种方法。例如，可以借助无穷字自动机 [26, 定理 4.4]，也可以借助拉姆齐定理 [26, 定理 4.14]。■

3.3 自动机理论

无限字自动机和树自动机理论一直是循环证明理论中的重要工具。本文也不例外：作为我们构造重置系统的核心，Safra 棋盘的概念（第 4 节）基于 Safra 树构造 [14]，该构造旨在高效地对某些无限字自动机进行确定化。本文给出的完备性证明也极为依赖一条关于无限树自动机所描述语言的可居住性的定理（命题 3.12）。

我们先回顾一些关于无限字自动机的概念。B¨uchi 自动机是一个元组$\mathfrak{B} = (Q, \Sigma, \Delta, S, F)$，其中 Q 是有限状态集，Σ 是有限字母表，$S \subseteq Q$ 是初始状态集，关系 $\Delta \subseteq Q \times \Sigma \times Q$ 是转移关系，而 $F \subseteq Q$ 是接受条件。给定一个字 $\sigma \in \Sigma^\omega$，若序列 $\rho \in Q^\omega$ 满足 $\rho_0 \in S$，且对每个 $i \in \omega$ 都有 $(\rho_i, \sigma_i, \rho_{i+1}) \in \Delta$，则称其为 $\mathfrak{B}$ 在 σ 上的一次运行。若存在某个 $q \in F$ 使得 $\rho_i = q$ 无穷多次出现，则运行 ρ 是接受运行。若存在 $\mathfrak{B}$ 在 σ 上的一次接受运行，则称字 σ 被 $\mathfrak{B}$ 接受。集合$L(\mathfrak{B}) := \{\sigma \in \Sigma^\omega \mid \sigma$ 被 $\mathfrak{B}$ 接受$\}$ 是 $\mathfrak{B}$ 的语言。

连接循环证明理论与无限字自动机理论的一个重要结果是，文献中许多循环证明系统中满足轨迹条件的分支，都可以由某些无限字自动机识别。固定任意一个迹范畴 T。若对于 T 的任意有限态射集 M，都存在一个 B¨uchi 自动机 $\mathfrak{B}$，使得 $L(\mathfrak{B})$ 是下文的集合 $T(M)$，则称其迹条件为*B¨uchi* 可识别的。

$$T(M) := \{\tau \in M^\omega \mid P(i < i+1) := \tau_i \text{是有效路径，并满足 T 的迹条件} \}$$

在 $\mathrm{T}_{\mathcal{A}}$ 的框架下，可以给出一种用于构造此类识别自动机的一般方法。该方法可用于证明命题 3.9。下面的构造是对 [26, 命题 5.11] 中所给构造的一种变体，该命题也给出了命题 3.11 的证明。

定义 3.10 设 A 为一个激活代数，M 为 $\mathrm{T}_{\mathcal{A}}$ 的态射有限集，并固定$O := \text{Đ}_{\tau \in M}\{\mathrm{dom}(\tau)、\mathrm{cod}(\tau)\}$。B¨uchi-自动机 $\mathfrak{B}(\mathrm{A}\ M) = (M,\ Q,\ \Delta,\ O,\ F)$ 定义如下，

固定某个任意的 $0^* \notin \mathrm{A}$。

$$
\begin{aligned}
Q &:= O \cup \{(X, x, a) \mid X \in O, x \in X, a \in \mathcal{A} \cup \{0^*\}\} \\
\Delta &:= \{(X, R\colon X \to Y, Y) \mid R\colon X \to Y \in M\} \\
&\quad \cup \{(X, R\colon X \to Y, (Y, y, 0)) \mid Y \in O, x \in Y\} \\
&\quad \cup \{((X, x, a), R\colon X \to Y, (Y, y, a \vee b)) \mid xR^b y, a \vee b \neq \alpha, R \in M\} \\
&\quad \cup \{((X, x, a), R\colon X \to Y, (Y, y, 0^*)) \mid xR^b y, a \vee b = \alpha, R \in M\} \\
&\quad \cup \{((X, x, 0^*), R\colon X \to Y, (Y, y, a)) \mid xR^a y, R \in M, a \neq \alpha\} \\
&\quad \cup \{((X, x, 0^*), R\colon X \to Y, (Y, y, 0^*)) \mid xR^a y, R \in M, a = \alpha\} \\
F &:= \{(X, x, 0^*) \mid X \in O, x \in X\}
\end{aligned}
$$

⌟

命题 3.11 对于任意 A，$\mathrm{T}_{\mathcal{A}}$ 的迹条件都可由 B¨uchi 自动机识别。此外，对于 $\mathrm{T}_{\mathcal{A}}$ 的任意一组态射 M，都有 $L(\mathfrak{B}(\mathrm{A}\ M)) = T(M)$。

对于字母表 Σ，一个 Σ标记的树是一个有序对 $(T, \lambda\colon T \to \Sigma)$，其中 T是一棵可能无限的树。如果一个Σ标记的树 (T, λ) 是 Σ标记树 $(T'\ \lambda')$ 的子树，那么它就是 T'的一个“后缀”；也就是说，存在某个$t \in T'$，使得 $T = \{ts \in T' \mid s \in T'\}$ 且 $\lambda(s) = \lambda'(ts)$。Rabin 树自动机是一个元组$\mathfrak{A} = (\Sigma, Q, \Delta, s, R)$，由一个有限字母表 Σ、一个状态集合 Q、一个转移集合 $\Delta \subseteq Q \times \Sigma \times Q^*$、一个初始状态 $s \in Q$ 以及一个接受条件 $R = \{(G_0, B_0) \ldots (G_n, B_n)\}$ 组成，其中$G_i \cap B_i = \emptyset$且 $G_i \cup B_i \subseteq Q$。设 (T, λ) 是一棵 Σ标记的树。$\mathfrak{A}$在 (T, λ) 上的运行是一个 Q标记$\rho\colon T \to Q$，它满足 $\rho(\varepsilon) = s$；并且对于每个满足 $\mathrm{Chld}(t) = \{t0, \ldots, tn\}$ 的 $t \in T$，转移 $(\rho(t)\ \lambda(t)\ \rho(t0) \ldots, \rho(tn)) \in \Delta$。如果对于 T的每条无限分支 $b \in T^\omega$，都存在 $(G, B) \in R$，使得 $\rho(b_i) \in G$对无穷多个 $i \in \omega$成立，且 $\rho(b_i) \in B$仅对有限多个 $i \in \omega$成立，那么该运行就是接受运行。如果存在 $\mathfrak{A}$ 在其上的接受运行，那么 Σ标记的树 (T, λ) 就被 $\mathfrak{A}$ 接受。集合 $L(\mathfrak{A}) := \{(T, \lambda\colon T \to \Sigma) \mid (T, \lambda)$ 被 $\mathfrak{A}$接受$\}$ 是 $\mathfrak{A}$的语言。

以下是 Rabin 博弈无记忆确定性的一个推论；例如，参见 [17].

命题 3.12 若 $\mathfrak{A}$ 是一个语言非空的 Rabin 树自动机，则 $\mathfrak{A}$ 通过正则运行接受一棵正则树。

4 Safra 棋盘

本节介绍 Safra 棋盘，它是 Safra [21]提出的树构造的一种变体，用于确定化 B¨uchi 自动机。我们专门针对自动机 $\mathfrak{B}(\mathrm{A}\ M)$，或等价地，针对 $\mathrm{T}_{\mathcal{A}}$ 的迹条件，对 Safra 棋盘的呈现方式进行了调整。受 Kozen 对 Safra 自动机的论述 [14], 启发，我们用上面放置筹码堆的棋盘来描述这一构造，而不是用树。Safra 棋盘能够识别态射序列 τ 是否为满足迹条件的路径，这与自动机 $\mathfrak{B}(\mathrm{A}\ M)$ 类似。它们是第 5 节所述抽象循环重置证明的构建模块。

对于 以下定义固定一个包含 $\omega \subseteq \mathrm{C}$的某个可数集，我们称其为筹码集 *s.*

定义 4.1 在激活代数 A 和集合 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 上的一个*Safra* 棋盘是一个元组 (Θ, σ)，由一个控制 Θ、集合 $\Theta \subset \mathrm{C}$ 上的一个有限线性序 $(\Theta, \leq)$，以及一个映射 $\sigma\colon X \times \mathrm{A} \to \mathrm{P}(\mathrm{P}(\Theta))$ 组成。此外，要求对于每个 $\gamma \in \Theta$，都存在 $a \in \mathrm{A}$ 和 $x \in X$，使得$\gamma \in S \in \sigma(x, a)$。$\Theta$中的元素称为筹码。集合 $S \in \sigma(x, a)$ 表示筹码栈，其中按 $\leq$最小的元素位于底部，按 $\leq$最大的元素位于顶部。

如果对于所有的 $x \in X$ 和 $a \in A$，γ 都不位于任何 $S \in \sigma(x, a)$ 的顶部，那么筹码 $\gamma \in \Theta$ 就是被覆盖的。

任意给定控制 Θ 中的筹码堆，都按照关系 $S <_{\Theta} S'$ 线性有序；当且仅当 S 包含对称差 $S \Delta S' = (S \setminus S') \cup (S' \setminus S)$ 中的最小元素时，该关系成立。

我们用 $\mathcal{SB}(\mathrm{A}, X)$ 表示 A 上 Safra 棋盘的集合，用 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 表示..... ⌟

与自动机 $\mathfrak{B}(\mathrm{A}, M)$ 类似，Safra 棋盘会导出一个状态转移系统，其中定义了“接受运行”的概念，用来识别描述经过 $\mathrm{T}_{\mathcal{A}}$ 的路径且满足迹条件的序列 $\tau \in M^{\omega}$。转移的形式为 $(\Theta, \sigma) \overset{X}{\rightsquigarrow} (\Theta', \sigma')$：从一个 Safra 棋盘转移到另一个 Safra 棋盘。这里，字母 X 表示转移类型，共有五种： τ-后继、弱化、稀疏化、 γ-重置和填充。下面我们逐一介绍每种转移。

定义 4.2 设 (Θ, σ) 是 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 上的一个 Safra 棋盘，并设 $\tau\colon X \to Y$ 是 $\mathrm{T}_{\mathcal{A}}$ 的一个态射。(Θ, σ) 的 τ-后继 是 Y 上的一个 Safra 棋盘 (Θ', σ')，它通过以下两个步骤得到：

移动 根据 τ 移动棋盘上的所有栈，得到如下的 Y 上的中间板 (Θ^*, σ^*)：

$$\sigma^*(y, a) \coloneqq \{S \in \sigma(x, b) \mid \exists c \in \mathcal{A}.\ (x, c, y) \in \tau \text{ and } a = b \vee c\}$$

其中， $\Theta^* \coloneqq \{\gamma \in \Theta \mid \exists y \in Y, a \in \mathrm{A}\ S \in \sigma^*(y, a)\ \mid \gamma \in S\}$ 是 Θ 的一个子序。

覆盖 覆盖所有落在 α 上的栈。首先，固定一个线性有序集合 $\Theta^\circ \subset \mathrm{C} \setminus \Theta$ 和双射 $\iota\colon \{y \in Y \mid \sigma^*(y, \alpha) \neq \emptyset\} \simeq \Theta^\circ$。然后令

$$\sigma'(y, a) \coloneqq \begin{cases} \emptyset & a = \alpha \\ \sigma^*(y, a) \cup \{S \cup \{\iota(y)\} \mid S \in \sigma^*(y, \alpha)\} & a = 0 \\ \sigma^*(y, a) & \text{otherwise} \end{cases}$$

现在固定 $\Theta' \coloneqq \Theta^* \oplus \Theta^\circ$，其中 $\oplus$ 表示线性序的拼接。

我们用 $(\Theta, \sigma) \overset{\tau}{\rightsquigarrow} (\Theta', \sigma')$ 表示 (Θ', σ') 是 (Θ, σ) 的 τ-后继。 ⌟

例 4.3 举例来说，记 $\mathbb{F}$ 三值故障代数（$\{0, 1, 2\} \leq, \vee, 0, 1$）以及集合 $\{w, x, y, z\} \in \mathrm{Ob}(\mathrm{T}_{\mathbb{F}})$。$\mathrm{Sb}(\mathbb{F}, X)$ 中的 Safra 棋盘可以看作一个方形游戏棋盘，类似于国际象棋棋盘，如图 1 所示。事实上，图 1 给出了以下内容的一个τ-后继转移： $\tau \coloneqq \{(x, 1, x), (x, 1, y), (y, 0, y), (y, 1, y)$ $(z, 2, z)\}$ 和 $\Theta \coloneqq \{a, b, c, d, e\}$（按字母顺序排列），以及

$$\sigma(w, 0) \coloneqq \{\{a\}\} \quad \sigma(x, 0) \coloneqq \{\{b\}\} \quad \sigma(y, 0) \coloneqq \{\{c, d\}\} \quad \sigma(z, 0) \coloneqq \{\{e\}\}$$

其中 $\sigma(u,v) := \emptyset$适用于 $u \in X$和 $v \in \{1,2\}$，如图 1a 所示。要得到它的一个 τ-后继，首先需要执行移动步骤，按照 τ 移动棋盘上的栈。此步骤得到的棋盘 (Θ^*, σ^*) 如图 1b 所示。请注意，来自 $(y,0)$ 的栈既被移动到了 $(y,1)$，也留在了 $(y,0)$；此外，位于 $(w,0)$ 的栈被移除了，因为在 τ中没有关于 w的轨迹三元组。还要注意，对于固定的棋盘 (Θ,σ) 和态射 τ，移动步骤完全确定。要得到 (Θ',σ')，需要执行覆盖步骤：将 (Θ^*,σ^*) 中每个 (u,α) 上的栈移回 $(u,0)$。为标记已达到 α，在每个移回的栈上放置一个新的芯片（该芯片此前不在 Θ中）。如果有多个栈从 (v,α) 移到 $(v,0)$，则在每个栈顶部放置同一个芯片。在本例中，由于 $\alpha = 1$，需要移动位于 $(x,1)$ 和 $(y,1)$ 上的栈，并引入新的芯片 g 和 h。

$X \setminus A$	0	1	2
w	a		
x	b		
y	cd		
z	e		

(a) 棋盘 (Θ,σ)

$X \setminus A$	0	1	2
w			
x		b	
y	c	b, cd	
z			e

(b) 棋盘 (Θ^*,σ^*)

$X \setminus A$	0	1	2
w			
x	bg		
y	c, bh, cdh		
z			e

(c) 棋盘 (Θ',σ')

图 1：$(\Theta,\sigma) \overset{\tau}{\rightsquigarrow} (\Theta',\sigma')$ 的示例

定义 4.4 设 (Θ,σ) 是 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 上的 Safra 棋盘。另一个位于 X 上的 Safra 棋盘 (Θ',σ') 是 (Θ,σ) 的弱化，如果对于每个 $x \in X$ 和 $a \in \mathrm{A}$，都有 $\sigma'(x,a) \subseteq \sigma(x,a)$。此外，还要求 $\Theta' \subseteq \Theta$ 满足：每个 $\gamma \in \Theta'$ 都出现在某个 $x \in X, a \in \mathrm{A}$ 对应的 $S \in \sigma'(x,a)$ 中。我们用 $(\Theta,\sigma) \overset{W}{\rightsquigarrow} (\Theta',\sigma')$ 表示后一个棋盘是前一个棋盘的弱化。

(Θ,σ) 的稀疏化 是由 $\sigma\prime(x,a) := \{\min_{<_\Theta} \sigma(x,a)\}$（若 $\sigma(x,a) \neq \emptyset$）以及其他情况下的 $\sigma\prime(x,a) = \emptyset$ 所诱导的特殊弱化 (Θ',σ')。我们用 $(\Theta,\sigma) \overset{T}{\rightsquigarrow} (\Theta',\sigma')$ 表示后一个棋盘是前一个棋盘的稀疏化。」

例 4.5 图 2 展示了从例 4.3 的结果出发进行一次稀疏化转移后的结果。当棋盘上的某个位置存在多个栈时，稀疏化会移除除 $<_\Theta$ 字典序最小的栈之外的所有栈。因此，在图 2a 的棋盘上，稀疏化只会修改 $\sigma(y,\ 0)$。注意，$c <_\Theta cdh$当 $\{c\}\Delta\{c,\ d,\ h\} = \{d,\ h\}$ 时，这意味着 cdh 包含对称差中字典序最小的元素。事实上，每当 $S \subset S'$ 对于 Θ 上的两个栈成立时，就有 $S\prime <_\Theta S$。其次，$bh <_\Theta cdh$当 bh 包含 b 而 cdh 不包含时。

定义 4.6 设 (Θ,σ) 是 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 上的 Safra 棋盘，且 $\gamma \in \Theta$ 被覆盖。栈 S的γ-重置 $S \upharpoonright \gamma$定义为

$$S \upharpoonright \gamma := \begin{cases} \{z \in S \mid z \leq \gamma\} & \text{if } \gamma \in S \\ S & \text{otherwise} \end{cases}$$

$X \setminus \mathrm{A}$	0	1	2
w			
x	bg		
y	c, bh, cdh		
z			e

(a) 棋盘 $(\Theta,\ \sigma)$

$X \setminus \mathrm{A}$	0	1	2
w			
x	bg		
y	bh		
z			e

(b) 棋盘 $(\Theta',\ \sigma')$

图 2：$(\Theta,\ \sigma) \overset{T}{\rightsquigarrow} (\Theta',\ \sigma')$ 示例

那么，(Θ, σ) 的 γ-重置 为 (Θ', σ_γ)，其中

$$\sigma'(x,a) \coloneqq \{S \restriction \gamma \mid S \in \sigma(x,a)\} \qquad \Theta' \coloneqq \{\gamma \in \Theta \mid \exists x \in X, a \in \mathcal{A}, S \in \sigma'(x,a).\ \gamma \in S\}$$

我们将 $(\Theta, \sigma) \overset{R_\gamma}{\rightsquigarrow} (\Theta', \sigma')$ 记作此表达式。 ⌟

例 4.7 图 3 展示了对例 4.5 中所得棋盘应用 a b-重置后的结果。请注意，在该棋盘上无法对 e, g 或 h 进行重置，因为它们都没有被覆盖。

$X \setminus \mathrm{A}$	0	1	2
w			
x	bg		
y	bh		
z			e

(a) 棋盘 (Θ, σ)

$X \setminus \mathrm{A}$	0	1	2
w			
x	b		
y	b		
z			e

(b) 棋盘 $(\Theta',\ \sigma')$

图 3：$(\Theta,\ \sigma) \overset{R_b}{\rightsquigarrow} (\Theta',\ \sigma')$ 的示例

定义 4.8 设 (Θ, σ) 是 $X \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 上的 Safra 棋盘。若对于每个 $x \in X$，都有 $\sigma(x, 0) \subseteq \sigma'(x, 0) \subseteq \sigma(x, 0) \cup \{\emptyset\}$，且对于所有 $a \in \mathrm{A} \setminus \{0\}$，都有 $\sigma'(x, a) = \sigma(x, a)$，则称棋盘 (Θ, σ') 是 (Θ, σ) 的填充，记作 $(\Theta, \sigma) \overset{P}{\rightsquigarrow} (\Theta, \sigma')$。⌟

例 4.9 图 4 展示了对例 4.7 所得棋盘进行种群转移后的结果。这里，新芯片 f 已被添加到 $(w,\ 0)$ 和 $(x,\ 0)$ 中。也可以将它另外添加到 $(y,\ 0)$ 和 $(z,\ 0)$ 中。

Safra 棋盘运行是由我们定义的不同类型的转移组成的序列。重要的是，要将这样的序列视为在某个 $\tau \in M^\omega$ 上的运行，它必须“消耗” τ中的所有“字母”。

定义 4.10 固定一个由 $\mathrm{T}_{\mathcal{A}}$的态射组成的集合 M以及某个 $\tau \in M^\omega$。若存在一个严格单调函数 ι：$\omega \to \omega$，且对于每个 $i \in \omega$，满足以下任一条件，则 Safra 棋盘序列 $(\Theta_i, \sigma_i)_{i \in \omega}$ 称为 τ 的运行

$X \setminus A$	0	1	2	
w				
x	b			
y	b			
z				e

(a) 棋盘 (Θ, σ)

$X \setminus A$	0	1	2
w	f		
x	b, f		
y	b		
z			e

(b) 棋盘 (Θ', σ')

图 4：$(\Theta, \sigma) \overset{P}{\rightsquigarrow} (\Theta', \sigma')$ 的示例

- $i = \iota(n)$，其中 $n \in \omega$，且 $(\Theta_i, \sigma_i) \overset{\tau_n}{\rightsquigarrow} (\Theta_{i+im}, \sigma_{i+1})$；或者
- $i \neq \iota(n)$ 和 $(\Theta_i,\ \sigma_i) \overset{W}{\rightsquigarrow} (\Theta_{i+1},\ \sigma_{i+1})$ 或 $(\Theta_i,\ \sigma_i) \overset{P}{\rightsquigarrow} (\Theta_{i+1},\ \sigma_{i+1})$ 或 $(\Theta_i,\ \sigma_i) \overset{R_\gamma}{\rightsquigarrow} (\Theta_{i+1},\ \sigma_{i+1})$ 对某个 $\gamma \in \Theta_i$

如果存在某个 N 和某个 $\gamma \in \tilde{\mathrm{N}}_{N \leq n} \Theta_n$，使得沿着 $(\Theta_i,\ \sigma_i)_{i \in \omega}$ 发生无穷多次 γ-重置，则运行 $(\Theta_i,\ \sigma_i)_{i \in \omega}$ 是接受的。⌟

构造 Safra 棋盘运行时有很大的灵活性，因为在任意时刻都可以采取许多不同类型的转移（例如，总是可以采取使 σ 保持不变的弱化转移）。对于本文中的某些证明，严格规定运行中转移的序关系会很有用。这可以通过贪心运行的概念实现：对于给定的 τ，这类运行的转移序关系是确定性的。之所以称这类运行为贪心运行，是因为可以证明，只要存在一个接受的 Safra 棋盘运行 τ，τ 的贪心运行也同样是接受的。因此，在许多情况下，只需关注贪心运行即可。反过来，在构造运行时，也总是可以遵循贪心构造策略。贪心运行的概念还与确定化 $\mathfrak{B}(\mathrm{A}\ M)$ 上的运行密切相关（见定义 4.19）。

定义 4.11 固定 $X, Y \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$、某个态射 $\tau\colon X \to Y$ 以及其上的一个 Safra 棋盘 (Θ, σ)。那么 (Θ', σ') 是从 (Θ, σ) 经过一次贪心 τ-转移 得到的结果，记作 $(\Theta, \sigma) \overset{\tau}{\rightsquigarrow}_g (\Theta', \sigma')$，如果

$$(\Theta, \sigma) = (\Theta_0, \sigma_0) \overset{R_{\gamma_k}}{\rightsquigarrow} \ldots \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_k, \sigma_k) \overset{P}{\rightsquigarrow} (\Theta_{k+1}, \sigma_{k+1}) \overset{\tau}{\rightsquigarrow} (\Theta_{k+2}, \sigma_{k+2}) \overset{T}{\rightsquigarrow} (\Theta_{k+3}, \sigma_{k+3}) = (\Theta', \sigma')$$

它是按照以下说明从第 1 步开始执行所产生的转移序列。

1. 如果 (Θ_0, σ_0) 中存在被覆盖的筹码 $\gamma_1 < \ldots < \gamma_k$，则按降序执行 γ_i-重置，即：

$$(\Theta_0, \sigma_0) \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_1, \sigma_1) \overset{R_{\gamma_{k-1}}}{\rightsquigarrow} \ldots \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_k, \sigma_k)$$

   然后继续执行第 2 步。

2. 继续进行填充 $(\Theta_k,\ \sigma_k) \overset{P}{\rightsquigarrow} (\Theta_{k+1},\ \sigma_{k+1})$，使每个 $\sigma_i(x,\ 0) = \emptyset$ 都被填充为 $|\sigma_{i+1}(x,\ 0)| = 1$，而所有其他 $\sigma_{i+1}(x,\ a)$ 保持不变。继续执行第 3 步。

3. 执行 τ-转移（Θ_{k+1}， σ_{k+1}） $\overset{\tau}{\rightsquigarrow}$ （Θ_{k+1}， σ_{k+1}），然后继续执行第 4 步。4. 执行稀疏化（Θ_{k+2}， σ_{k+2}） $\overset{T}{\rightsquigarrow}$ （Θ_{k+3}， σ_{k+3}）。

我们用 $(\Theta,\ \sigma)\ \overset{\tau}{\rightsquigarrow}_g(\Theta',\ \sigma')$ 表示上文所述的完整转移序列 . ⌟

定义 4.12 某个 τ 的运行 $(\Theta_i,\ \sigma_i)_{i\in\omega}$ 若满足贪心条件，则称其为贪心运行： $\Theta_0 = \emptyset$ 且 $\sigma_0(x,\ a) = \emptyset$，此外，该运行还是由贪心 τ_i-转移构成的序列，即：

$$(\Theta_0, \sigma_0) \overset{\tau_0}{\rightsquigarrow}_g (\Theta_{\iota(0)+2}, \sigma_{\iota(0)+2}) \overset{\tau_1}{\rightsquigarrow}_g (\Theta_{\iota(1)+2}, \sigma_{\iota(1)+2}) \overset{\tau_2}{\rightsquigarrow}_g \cdots$$ ⌟

事实 4.13 如果 $\tau \in M^\omega$ 描述了一条经过 $\mathrm{T}_{\mathcal{A}}$ 的路径，则存在一个 τ 的贪心运行；除 Θ_i的筹码选择外，该运行是唯一的。如果 τ 未描述这样的路径，则不存在 τ 的贪心运行。

证明 要证明贪心运行的存在，只需注意，定义 4.11 的条款 1、2 和 4 所规定的转移总是可以执行的。因此，构造此类运行可能失败的唯一原因，是某个规定的 τ_i-后继转移无法执行。出现这种情况的唯一原因，是当前 Safra 棋盘所处的集合不同于 $\mathrm{dom}(\tau_i)$。然而，如果 $(\Theta_0,\ \sigma_0) \in \mathrm{Sb}(\mathrm{A},\ \mathrm{dom}(\tau_0))$，很容易看出，由于假设 τ 描述了一条经过 $\mathrm{T}_{\mathcal{A}}$ 的路径，这个问题不会出现。因此可以构造出一个贪心运行，而且它确实是一个运行，因为 τ 的所有“字母”最终都会被读取。由于定义 4.11 的条款总是规定唯一一个接下来要执行的转移，因此 τ 的贪心运行中的转移顺序总是固定的，这意味着它们只能在筹码选择上有所不同，正如所声称的那样。

对于第二个断言，注意到如果 τ 不能描述一条路径，那么必定存在 τ_i 和τ_{i+1}，使得 $\mathrm{cod}(\tau_i) \neq \mathrm{dom}(\tau_{i+1})$。在这种情况下，如上所述，无法执行 τ_{i+1}-后继步骤，这意味着 τ 上不存在运行（因而也不存在贪心运行）。■

接下来，我们将证明上述定义在以下意义下是正确的：任意 $\tau \in M^\omega$ 描述一条满足迹条件的路径，当且仅当 τ上存在一个接受的 Safra 棋盘运行。我们的论证依赖于 Safra 棋盘运行与 $\mathfrak{B}$上的运行之间的对应关系（$\mathrm{A}\ M$）。因此，这些结果仅在 A 有限的 $\mathrm{T}_{\mathcal{A}}$ 中成立。

引理 4.14 设 M 是某个有限 A 上 $\mathrm{T}_{\mathcal{A}}$ 的有限态射集。如果 $\tau \in M^\omega$ 描述一条满足迹条件的路径，那么 τ 上的贪心运行存在且是接受的。

证明 根据事实 4.13，贪心运行 $(\Theta_i, \sigma_i)_{i\in\omega}$存在。回想一下，存在一个函数 ι： $\omega \to \omega$，用于指示读取 τ_i时的索引，即 $(\Theta_{\iota(i)}, \sigma_{\iota(i)}) \overset{\tau_i}{\rightsquigarrow} (\Theta_{\iota(i)+1}, \sigma_{\iota(i)+1})$。此外，注意到由于 τ 满足迹条件，因此存在一个接受运行 $\rho \in Q^\omega$，它在 $\mathfrak{B}(\mathrm{A}\ M)$ 上运行于 τ。由于运行 ρ 是接受的，从某个时刻 R 开始，它必定沿着 τ 追踪一条轨迹，经过状态 $\Sigma X \in O.\ X \times (\mathrm{A} \cup 0^*) \subseteq Q$。这样的状态 (X, x, a) 对应于 X上的 Safra 棋盘中的位置 (x, a)，这一联系对本证明至关重要。因此，对于任意 $R \leq i$，我们记 $\sigma_\iota(i)\ (\rho_i)$，表示 $\sigma_{\iota(i)}(x, a)$，其中 $\rho_i = (X, x, a)$，并将 0^* 视为 $0 \in \mathrm{A}$（容易看出，定义 $\Theta_{\iota(i)}$所依托的对象$X_\iota(i) \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 必定始终与 X相同）。

我们对贪心运行中、读取 τ的下一个字母之前的 Safra 棋盘作几点观察，即棋盘 $(\Theta_{\iota(i)}, \sigma_{\iota(i)})$。为此，固定 $|X| := \sup_{i\in\omega} |X_i|$；由于 M是有限个态射构成的有限集，因此只有有限多个不同的 X_i。

1. $|\sigma_{\iota(i)}(x, a)| \leq 1$在每个 $\iota(i)$ 处：这是通过 τ_{i-1}-后继之后的稀疏化来保证的（或者因为贪心运行从空棋盘开始）。因此，我们将 $\sigma_{\iota(i)}$ 视为函数$X_i \times \mathrm{A} \to \mathrm{P}(\Theta_{\iota(i)})$，其中 $\sigma_{\iota(i)}(x, a) = \emptyset$当且仅当在原始解释下 $\sigma_\iota(i)(x, a) = \emptyset$。2. $|\Theta_{\iota(i)}| \leq |X| \cdot |\mathrm{A}|$：如果筹码数超过 $|X| \cdot |\mathrm{A}|$，那么必有一个筹码在 $(\Theta_\iota(i)_{-2}, \sigma_{\iota(i)-2})$ 上被覆盖（该棋盘是最近一次重置后得到的，而这次重置属于 $(\Theta_\iota(_{i-1})_{+2}, \sigma_\iota(_{i-1})_{+2}) \overset{\tau_i}{\rightsquigarrow}_g (\Theta_\iota(i)_{+2}, \sigma_{\iota(i)+2})$），因为每个 (x, a) 上只有一个最顶部的筹码；这与最近一次重置是在 $(\Theta_{\iota(i)-3}, \sigma_\iota(i)_{-3})$ 处应用的事实矛盾。3. 对任意 $R \leq i$，$\sigma_{\iota(i)}(\rho_i) \neq \emptyset$：我们对 i 使用归纳法。首先，假设 $R = i$：那么对于某个 $x \in X_i$，有 $\rho_i = (X_i, x, 0)$。由于在贪心运行中，萨弗拉棋盘上任何没有栈的点 $(x, 0)$ 都会在读入下一个态射之前被填入，因此 $\sigma_\iota(i)(\rho_i) = \sigma_\iota(i)(x, 0)$ 非空。现在，由于 $\sigma_\iota(i)(\rho_i)$ 非空，很容易看出，在计算 τ_i-后继时，$\sigma_\iota(i)(\rho_i)$ 上的栈会被移到 $\sigma_\iota(i)_{+1}(\rho_{i+1})$ 上。由于在 $(\Theta_{\iota(i)+1}, \sigma_{\iota(i)+1})$ 与 $(\Theta_\iota(_{i+1}) \ \sigma_\iota(_{i+1}))$ 之间可能发生的任何步骤，都不会清除棋盘上任何至少有一个栈的空间中的所有栈，因此可得 $\sigma_{\iota(i+1)}(\rho_{i+1}) \neq \emptyset$（尽管其上唯一的栈可能并非从 $\sigma_{\iota(i)}(\rho_i)$ 移来的栈，因为可能发生了稀疏化步骤）。4. 必定存在一个最大高度 $1 \leq k \leq |X| \cdot |\mathrm{A}|$，使得从某个 $R < K$ 起，对于每个 $\iota(i) > K$，$|\sigma_\iota(i)(\rho_i)| \geq k$ 的高度不超过 ρ_i 上栈的高度：这是因为 $|\sigma_\iota(i)(\rho_i)| \leq |X| \cdot |\mathrm{A}|$（由 2. 得出），且$|\sigma_\iota(i)(\rho_i)| \geq 1$（由 3. 得出）。5. 从某个 $K < N$ 起，所有满足 $N \leq i$ 的 $\sigma_{\iota(i)}(\rho_i)$ 的第 k 个筹码必须保持不变：由于 $|\sigma_{\iota(i)}(\rho_i)|$ 之后不再小于 k，这意味着第 k 个筹码不会再作为重置的一部分被清除；因此，第 k 个筹码的颜色若要改变，唯一可能的情况就是在稀疏化过程中，$\sigma_{\iota(i)}(\rho_i)$ 上的栈被一个更小的栈“替换”，而这个栈的第 k 个筹码颜色不同。这样的栈按照前 k 个元素上的字典序也总是更小。但在任意有限线性序上，这种字典序都是良基的，因为它总能嵌入良基的 ω^k。因此，这种替换只能发生有限次。

因此，对于任意 $N \leq i$，$\sigma_\iota(i)(\rho_i)$ 的第 k 个值（记作 γ）都保持不变，这也意味着对于所有满足 $N \leq i$ 的 i，$\gamma \in \Theta_i$。只需证明会发生无穷多次 γ-重置，即可得出运行 $(\Theta_i, \sigma_i)_{i\in\omega}$ 是接受运行。由于 ρ 是接受运行，它会无穷多次经过状态 $(x, 0^*) \in F$。注意，每当运行进入 $(x, 0^*)$ 时，它所沿循的轨迹便已达到 α，也就是说，在 $\sigma_\iota(i)(\rho_i)$ 的栈顶放置了一个新筹码，而该栈从 N 开始包含 γ。由于 k 被选为无穷多次出现的最大栈高度，因此也可得出$|\sigma_\iota(i)(\rho_i)| = k$，从而 $\max \sigma_\iota(i)(\rho_i) = \gamma$ 会无穷多次成立。在 N 之后，只有通过一次 γ-重置，才能移除 γ 上方由达到 α 的、ρ 所追踪的轨迹添加的新筹码。因此，沿着 $(\Theta_i, \sigma_i)_{i\in\omega}$ 必须发生无穷多次 γ-重置，使其成为 τ 上的接受运行。

■

引理 4.15 固定一个由 M 个 $\mathrm{T}_{\mathcal{A}}$ 态射组成的有限集，其中 A 有限。现在假设某个 $\tau \in M^{\omega}$ 有一个接受运行 $(\Theta_i,\ \sigma_i)_{i\in\omega}$。那么 τ 描述了 $\mathrm{T}_{\mathcal{A}}$ 中一条满足 $\mathrm{T}_{\mathcal{A}}$ 迹条件的路径。

证明 我们通过证明 τ 在 $\mathfrak{B}(\mathrm{A}\ M)$ 上必定存在接受运行来证明这一点。由于运行 $(\Theta_i, \sigma_i)_{i\in\omega}$ 是接受运行，因此存在 $N \in \omega$ 和 $\gamma \in \tilde{\mathrm{N}}_{N\le i}\Theta_i$，使得沿该运行发生无穷多次 γ-重置。现在，将 $\overline{\tau}[i, j] \in M^{\omega}$ 记为在索引 i 和 j 之间读取的 τ 的字母；也就是说，若 $\iota(k-1) < i \le \iota(k) < \iota(k+n) < j \le \iota(k+n+1)$，则 $\overline{\tau}[i, j] = \tau_k\tau_{k+1}\ldots\tau_{k+n}$。我们先证明一个关键事实：对于 $N \le i \le j$，若 $\gamma \in \mathrm{D}\,\sigma_j(x, a)$，则必定存在 $x' \in X_i$ 和 $b \in \mathrm{A}$，使得 $\gamma \in \mathrm{D}\,\sigma_i(x', b)$，并且 $(X_i, x'\,b) \xrightarrow{\overline{\tau}[i,j]} (X_j, x, a)$ 在 $\mathfrak{B}(\mathrm{A}\ M)$ 上（为此，我们将 0 与 0^* 视为相同）。我们对 j 使用归纳法证明这一点。显然，若 $i = j$，则可以将 $x' := x$ 选为 $b := a$，并令其在 $\mathfrak{B}(\mathrm{A}\ M)$ 中为 $(X_i, x, a) \xrightarrow{\varepsilon} (X_i, x, a)$。对于归纳步骤，我们按 (Θ_j, σ_j) 与 $(\Theta_{j+1}, \sigma_{j+1})$ 之间的转移步骤进行分类讨论：

- $(\Theta_j,\ \sigma_j) \overset{\tau_k}{\rightsquigarrow} (\Theta_{j+1},\ \sigma_{j+1})$：假设 $\gamma \in \mathrm{D}\,\sigma_{j+i}$（$x,\ a$）。如 $\gamma \in \Theta_j$，不难看出，包含 γ 的栈只有在前一次转移将其“移动”到 γ 时，才可能到达那里。更确切地说，这意味着必须存在一个 $x' \in X_j$ 和 $b \in \mathrm{A}$，使得 $\gamma \in \mathrm{D}\,\sigma_j$（$x'\,b$），且（$X_j,\ x'\,b$）$\xrightarrow{\tau_k}$（$X_{j+1},\ x,\ a$）。根据归纳假设，还必须存在 $x'' \in X_i$ 和 $c \in \mathrm{A}$，使得 $\gamma \in \mathrm{D}\,\sigma_i$（$x''\,c$），且（$X_i,\ x''\,c$）$\xrightarrow{\overline{\tau}[i,j]}$（$X_j,\ x'\,b$）。如 $\overline{\tau}[i, j+1] = \overline{\tau}[i, j]\tau_k$，由此可得（$X_i,\ x''\,c$）$\xrightarrow{\overline{\tau}[i,j+1]}$（$X_{j+1},\ x,\ a$），证毕。

- $(\Theta_j,\ \sigma_j) \overset{W}{\rightsquigarrow} (\Theta_{j+1},\ \sigma_{j+1})$：如果 $\gamma \in \mathrm{D}\,\sigma_{j+1}(x,\ a)$，那么 $\gamma \in \mathrm{D}\,\sigma_j(x,\ a)$ 也成立，因为弱化只能移除栈。由此可根据归纳假设直接得出结论，因为$\overline{\tau}[i, j+1] = \overline{\tau}[i, j]$。

- $(\Theta_j,\ \sigma_j) \overset{R_\gamma}{\rightsquigarrow} (\Theta_{j+1},\ \sigma_{j+1})$：由于这种重置只会从某些栈中移除筹码，$\gamma \in \mathrm{D}\,\sigma_{j+1}(x,\ a)$ 意味着 $\gamma \in \mathrm{D}\,\sigma_j(x,\ a)$ 也成立。因此，只需按照归纳假设继续即可。
- $(\Theta_j,\ \sigma_j) \overset{P}{\rightsquigarrow} (\Theta_{j+1},\ \sigma_{j+1})$：同样，$\gamma \in \mathrm{D}\,\sigma_{j+1}(x,\ a)$ 蕴含 $\gamma \in \mathrm{D}\,\sigma_j(x,\ a)$，因为 σ_{j+1} 与 σ_j 的区别仅在于增加了一些空栈（因此其中不可能含有 γ）。按照归纳假设继续即可。

现在令 $(r_n)_{n\in\omega}$ 为 γ-重置在 N 之后的索引序列，即一个单调递增的序列，满足 $N < r_0$ 且 $(\Theta_{r_n}, \sigma_{r_n}) \overset{R_\gamma}{\rightsquigarrow} (\Theta_{r_n+1}, \sigma_{r_n+1})$。定义集合 $S_n := \{(x, a) \mid \gamma \in \mathrm{D}\,\sigma_{r_n}(x, a)\}$。前述结果意味着，对于任意 $(x, a) \in S_{n+1}$，都存在 $(x', b) \in S_n$，使得 $(X_{r_n}, x', b) \xrightarrow{\overline{\tau}[r_n, r_{n+1}]} (X_{r_{n+1}}, x, a)$ 属于 $\mathfrak{B}(\mathrm{A}\ M)$（其中我们将 $0 \in \mathrm{A}$ 与自动机状态中的 0^* 视为相同）。应用 K¨onig 引理可得一个序列 $((x_n, a_n) \in S_n)_{n\in\omega}$，使得对于每个 $n \in \omega$，都有 $(X_{r_n}, x_n, a_n) \xrightarrow{\overline{\tau}[r_n, r_{n+1}]} (X_{r_{n+1}}, x_{n+1}, a_{n+1})$。值得注意的是，每个运行片段 $(X_{r_n}, x_n, a_n) \xrightarrow{\overline{\tau}[r_n, r_{n+1}]} (X_{r_{n+1}}, x_{n+1}, a_{n+1})$ 都至少一次经过 $\mathfrak{B}(\mathrm{A}\ M)$ 的接受状态集合 F：在 $(\Theta_{r_n+1}, \sigma_{n+1})$ 中，每个 γ 实例都是其各自栈中最顶部的筹码。另一方面，在 $(\Theta_{r_{n+1}}, \sigma_{r_{n+1}})$ 中，每个 γ 实例都被覆盖。这意味着，每个栈 $\gamma \in S \in \sigma(x, a)$（其中 $(x, a) \in S_{n+1}$）都必须在 r_n 与 r_{n+1} 之间至少一次“达到 α”。在 $\mathfrak{B}(\mathrm{A}\ M)$ 中，

这对应于经过 F。因此，运行片段 $(X_{r_n}, x_n, a_n) \xrightarrow{\overline{\tau}[r_n, r_{n+1}]} (X_{r_{n+1}}, x_{n+1}, a_{n+1})$ 已经提供了 τ 上某个接受运行的后缀，因为 F 被无限次经过。剩下的就是证明存在一个运行片段 $X_0 \xrightarrow{\overline{\tau}[0, r_0]} (X_{r_0}, x_0, a_0)$，以便在 $\mathfrak{B}(\mathrm{A}\ M)$ 上组装 τ 的接受运行。再次应用前述结果可知，必定存在一个 $x \in X_N$ 和一个 $a \in \mathrm{A}$，使得 $(X_N, x, a) \xrightarrow{\overline{\tau}[N, r_0]} (X_{r_0}, x_0, a_0)$。现在考察步骤 $(\Theta_{N-1}, \sigma_{N-1}) \rightsquigarrow (\Theta_N, \sigma_N)$，不失一般性地假设该步骤将筹码 γ 引入 Θ_N，即 $\gamma \notin \Theta_{N-1}$。新筹码只能由 τ_k-步骤的覆盖阶段引入。因此，新筹码只能出现在 $(x, 0)$ 上，这意味着上面的运行片段实际上是 $(X_n, x, 0) \xrightarrow{\overline{\tau}[N, r_0]} (X_{r_0}, x_0, a_0)$。最后，注意到运行 $(\Theta_i, \sigma_i)_{i \in \omega}$ 的存在已保证 $\mathrm{cod}(\tau_k) = \mathrm{dom}(\tau_{k+1})$，因为对于每个 $k \in \omega$，都有 $(\Theta_\iota(k+1)\ \sigma_\iota(k+1)) \overset{\tau_{k+1}}{\rightsquigarrow} (\Theta_\iota(k+1)+1, \sigma_\iota(k+1)+1)$。这意味着 $X_0 \xrightarrow{\overline{\tau}[0, N]} (X_n, x, 0)$ 是 $\mathfrak{B}(\mathrm{A}\ M)$ 上的运行片段。因此，可以组装下方图示的 τ 在 $\mathfrak{B}(\mathrm{A}\ M)$ 上的接受运行，并得出结论：τ 确实描述了 $\mathrm{T}_{\mathcal{A}}$ 中一条满足迹条件的路径。

$$X_0 \xrightarrow{\overline{\tau}[0,N]} (X_n, x, 0) \xrightarrow{\overline{\tau}[N,r_0]} (X_{r_0}, x_0, a_0) \xrightarrow{\overline{\tau}[r_0,r_1]} (X_{r_1}, x_1, a_1) \xrightarrow{\overline{\tau}[r_1,r_2]} \dots$$ ∎

定理 4.16 固定某个有限集合 M，其中包含一些 $\mathrm{T}_{\mathcal{A}}$ 的态射，且 A 有限。存在一条在 $\tau \in M^\omega$ 上的接受 Safra 棋盘运行，当且仅当 τ 描述了一条经过 $\mathrm{T}_{\mathcal{A}}$ 且满足迹条件的路径。

最后，我们通过 Safra 构造 [21] 来说明 Safra 棋盘与 B¨uchi 自动机的确定化（更具体地说，是 $\mathfrak{B}(\mathrm{A}\ M)$ 的确定化）之间的联系。为此，我们根据 Safra 棋盘定义 $\mathfrak{B}(\mathrm{A}\ M)$ 的一个确定化变体。

为确保我们构造的自动机具有有限状态空间，我们首先证明，在执行贪心转移步骤时，只需有限的筹码供应即可。引理 4.18 中陈述的最后一个条件对于所构造自动机的接受条件至关重要。

定义 4.17 固定一个有限的 A、$X \in \mathrm{T}_{\mathcal{A}}$ 和一个数 $K \geq |X|$。固定筹码供应 $K = \{n \in \omega \mid n < K \cdot (|\overline{\mathrm{A}}| + 1)\}$，将 $\mathrm{Sb}(\mathrm{A}\ X,\ K) \subseteq \mathrm{Sb}(\mathrm{A}\ X)$ 记为 K-稀疏 Safra 棋盘的集合。若棋盘 $(\Theta,\ \sigma)$ 是 K-稀疏的，则

- $\Theta \subseteq \overline{K}$ 和 $|\Theta| \leq K \cdot |\mathrm{A}|$
- 在 $(\Theta,\ \sigma)$ 中，每个棋盘位置上至多有一个栈
- 在 $(\Theta,\ \sigma)$ 上的任何位置 $(x,\ \alpha)$ 都没有栈 ⌟

引理 4.18 对于 $X, Y \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$，设 $(\Theta_0, \sigma_0) \in \mathrm{Sb}(\mathrm{A}\ X, K)$ 是一个 $K \geq |X|, |Y|$。那么，对于任意 $\tau\colon X \to Y$，都存在一个 Safra 棋盘 $(\Theta_n, \sigma_n) \in \mathrm{Sb}(\mathrm{A}\ Y, K)$，以及一个转移序列 $(\Theta_0, \sigma_0) ? \rightsquigarrow \dots ? \rightsquigarrow (\Theta_n, \sigma_n)$，使得 $(\Theta_0, \sigma_0) \overset{\tau}{\rightsquigarrow}_g (\Theta_n, \sigma_n)$。此外，$\Theta_n \cap \Theta_0 = \tilde{\mathrm{N}}_{i \leq n} \Theta_i$。

证明 对于棋盘 $(\Theta,\ \sigma)$，定义 $|\sigma| \coloneqq |\{(x,\ a) \in \mathrm{dom}(\sigma) \mid \sigma(x,\ a) \neq \emptyset\}|$，即棋盘位置上具有栈的数量。我们将说明，可以从 $(\Theta_0,\ \sigma_0)$ 依次执行定义 4.11 中的步骤 1 至 4，使所得序列 $(\Theta_i,\ \sigma_i)_{i \leq n}$ 满足 $\Theta_i \subseteq K$。为确保 $\Theta_n \cap \Theta_0 = \tilde{\mathrm{N}}_{i \leq n} \Theta_i$，必须使转移仅引入来自 $\Theta' \coloneqq \overline{K} \setminus \Theta_0$ 的筹码。注意，根据 $(\Theta_0,\ \sigma_0)$ 的第一个稀疏条件，可得 $|\Theta_0| \leq K \cdot |\mathrm{A}|$，因此 $K \leq |\Theta'|$。

1. 假设 $\gamma_1 < \ldots < \gamma_n \in \Theta_0$ 在（$\Theta_0,\ \sigma_0$）中被覆盖。我们首先证明，该序列中的每个转移

$$(\Theta_0, \sigma_0) \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_1, \sigma_1) \overset{R_{\gamma_{k-1}}}{\rightsquigarrow} \ldots \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_k, \sigma_k)$$

都可以执行。沿此序列，可能导致转移（$\Theta_i,\ \sigma_i$）$\overset{R_{\gamma_{k-i}}}{\rightsquigarrow}$（$\Theta_{i+1},\ \sigma_{i+1}$）不合法的唯一原因，是 γ_{k-i} 未在（$\Theta_i,\ \sigma_i$）中被覆盖。只有当此序列中某个更早的重置“取消覆盖”了 γ_{k-i} 时，才会发生这种情况。但这不可能发生：假设 $\gamma > \gamma'$，并考虑某个栈，其中 $S \ni \gamma'$ 和 γ' 均被覆盖，即 $\max S \neq \gamma'$。那么 $S \restriction \gamma$ 有两种可能：若 $\gamma \notin S$，则 $S \restriction \gamma = S$，因此 γ' 仍被覆盖。若 $\gamma \in S$，则 $\gamma \in S \restriction \gamma$，这意味着芯片 $\gamma > \gamma'$ 仍留在 S 中，且 γ' 仍被覆盖。因此，对某个 $\gamma_j > \gamma_{k-i}$ 执行重置不会取消对 γ_{k-i} 的覆盖，这意味着可以执行 γ_{k-i}-重置。

由于重置只会移除筹码，很容易看出 $\Theta_k \subseteq \Theta_0 \subseteq K$。由于重置从不添加任何新栈，因此可得 $|\sigma_k| = |\sigma_0|$。现在注意以下几点：如果一个筹码在（$\Theta_k,\ \sigma_k$）中被覆盖，那么它在（$\Theta_0,\ \sigma_0$）中也被覆盖，因为重置只会移除栈顶部的筹码。因此，每个筹码 $\gamma \in \Theta_k$ 都必须位于（$\Theta_k,\ \sigma_k$）中至少一个栈的顶部：反设（$\Theta_k,\ \sigma_k$）中有一个被覆盖的筹码。但这样一来，它在（$\Theta_0,\ \sigma_0$）中也会被覆盖，这意味着它属于 γ_i，并且会在这一串重置操作中被重置。但它不可能在（$\Theta_k,\ \sigma_k$）中被覆盖，因为任何被重置的筹码在重置后都会变为未覆盖状态。如果 Θ_k 中的每个筹码都必须出现在至少一个栈的顶部，那么很容易看出 $|\Theta_k| \leq |\sigma_k|$。

2. 执行填充（$\Theta_k,\ \sigma_k$）$\overset{P}{\rightsquigarrow}$（$\Theta_{k+1},\ \sigma_{k+1}$）。这意味着，如果 σ_k（$x,\ 0$）$= \emptyset$，那么 σ_{k+1}（$x,\ 0$）$= \{\emptyset\}$。由于 $\Theta_{k+1} = \Theta_k$，因此可得 $\overline{\Theta}_{k+1} \subseteq K$。

3. 现在，使用 τ 执行（$\Theta_{k+1},\ \sigma_{k+1}$）$\overset{\tau}{\rightsquigarrow}$（$\Theta_{k+2},\ \sigma_{k+2}$）：$X \to Y$。在变换过程中，栈根据 τ 从位置（$x,\ a$）$\in X \times \mathrm{A}$ 移至位置（$y,\ a'$）$\in Y \times \mathrm{A}$。如果某个栈落在（$y,\ \alpha$）上，则会将其移至（$y,\ 0$），并在栈顶添加一枚新筹码。如果多个栈落在（$y,\ \alpha$）上，则会为它们全部添加相同的筹码。在此步骤中，最多只有 $|\{(y,\ \alpha) \mid y \in Y\}| = |Y|$ 枚新筹码被放入 Θ_{k+2}，这意味着筹码供应 Θ' 充足。因此，$\Theta_{l+1} \subseteq K$。

4. 在这一步（$\Theta_{k+2},\ \sigma_{k+2}$）$\overset{T}{\rightsquigarrow}$（$\Theta_{k+3},\ \sigma_{k+3}$）中，会移除一些栈，以及其中可能曾经包含的一些筹码，并且不会添加新的筹码。因此 $\Theta_{k+3} \subseteq K$。

不难看出，在第 4 步之后，$(\Theta_{k+3}, \sigma_{k+3})$ 的每个位置上至多有一个栈。此外，在第 3 步之后，位置 $(y,\ \alpha)$ 上没有栈，而这一事实在第 4 步后仍然成立。接下来只需证明 $|\Theta_{k+3}| \subseteq K \cdot |\mathrm{A}|$。我们已证明，在第 1 步之后，$|\Theta_k| = |\sigma_0|$。

由于在 (Θ_0, σ_0) 中，(x, α) 上没有栈，这意味着 $|\Theta_k| \leq |X| \cdot (|\mathrm{A}| - 1) \leq K \cdot (|\mathrm{A}| - 1)$。不难看出，在第 2 步之后，$|\Theta_{k+1}| = |\Theta_k|$。如前所述，第 3 步至多添加 $|Y| \leq K$ 个新筹码，这意味着 $|\Theta_{k+2}| \leq |\Theta_{k+1}| + K \leq K \cdot |\mathrm{A}|$。由于第 4 步只会移除筹码，因此这意味着$|\Theta_{k+3}| \leq |\Theta_{k+2}| \leq K \cdot |\mathrm{A}|$，如所需。■

因此，我们可以构造一个确定性 Rabin 自动机，用来识别 $\mathrm{T}_{\mathcal{A}}$ 的迹条件，类似于非确定性 Büchi 自动机 $\mathfrak{B}(\mathrm{A}\ M)$。对于自动机的构造，以及第 5.2 节中的一些论证，如果对于每个 $(\Theta, \sigma) \in \mathrm{Sb}(\mathrm{A}\ X, K)$ 和 τ，都有唯一的 $(\Theta', \sigma') \in \mathrm{Sb}(\mathrm{A}\ Y, K)$，使得 $(\Theta, \sigma) \overset{\tau}{\rightsquigarrow}_g (\Theta', \sigma')$，将会很有帮助。因此，我们直接假设，对于每个这样的 (Θ, σ) 以及适用的 τ，都已通过应用选择公理或其他方式选定了一个这样的对象；并且对于适当的 K，将 $\overset{\tau}{\rightsquigarrow}_g$ 视为定义在 K- 稀疏 (Θ, σ) 上的单射函数，使得 $(\Theta, \sigma) \overset{\tau}{\rightsquigarrow}_g (\Theta', \sigma')$ 始终依据引理 4.18 推导得出。

定义 4.19 设 A 有限，并固定 $\mathrm{T}_{\mathcal{A}}$ 中对象的有限集 $O \subset \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 以及态射集 $M \subseteq \text{Đ}_{X,Y \in O} \mathrm{Hom}(X, Y)$。此外，令 $K := \max_{X \in O} |X|$。以起始对象 $S \in O$ 为起点的 ***Safra*** 自动机$\mathfrak{S}(\mathrm{A}\ S, M)$ 是 Rabin 自动机 $(M, Q, \delta, (S, (\Theta_0, \sigma_0))\ R)$，其中

$$Q := \Sigma X \in O.\ \mathrm{SB}(\mathcal{A}, X, K)$$

$$\delta := ((X, (\Theta, \sigma)), \tau \colon X \to Y) \mapsto \text{some } (Y, (\Theta', \sigma')) \overset{\tau}{\rightsquigarrow}_g (X, (\Theta, \sigma))$$

$$R := \{(\{(X, (\Theta, \sigma)) \in Q \mid \gamma \in \Theta, \gamma \text{ covered}\}, \{(X, (\Theta, \sigma)) \in Q \mid \gamma \notin \Theta\}) \mid \gamma \in \overline{K}\}$$

和 $(\Theta_0,\ \sigma_0) = (\emptyset,\ (s,\ a) \mapsto \emptyset)$。 ⌟

引理 4.20 对于有限的 A、$O \subset \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$ 以及 $M \subseteq \text{Đ}_{X,Y \in O} \mathrm{Hom}(X, Y)$，Safra 自动机$\mathfrak{S}(\mathrm{A}\ S, M)$ 接受序列 $\tau \in M^\omega$，当且仅当 $\mathrm{dom}(\tau_0) = S$，并且 τ描述一条经过 $\mathrm{T}_{\mathcal{A}}$ 的、满足迹条件的路径。

证明 首先，考虑 $\tau \in M^\omega$ 在 $\mathfrak{S}(\mathrm{A}\ S, M)$ 上的任意一次运行 $(S, (\Theta_0, \sigma_0)) \xrightarrow{\tau_0} (X_1, (\Theta_1, \sigma_1)) \xrightarrow{\tau_1} \ldots$。每个状态转移 $(X_i, (\Theta_i, \sigma_i)) \xrightarrow{\tau_i} (X_{i+1}, (\Theta_{i+1}, \sigma_{i+1}))$ 都对应于根据引理 4.18 得到的一个贪心转移 $(\Theta_i, \sigma_i) \overset{\tau_i}{\rightsquigarrow}_g (\Theta_{i+1}, \sigma_{i+1})$。因此，可以将该运行“expand”为如下形式的 Safra 棋盘运行

$$(\Theta_0, \sigma_0) \overset{?}{\rightsquigarrow} (\Theta_0^1, \sigma_0^1) \overset{?}{\rightsquigarrow} \ldots \overset{?}{\rightsquigarrow} (\Theta_0^{n_0}, \sigma_0^{n_0}) \overset{?}{\rightsquigarrow} (\Theta_1, \sigma_1) \overset{?}{\rightsquigarrow} (\Theta_1^1, \sigma_1^1) \overset{?}{\rightsquigarrow} \ldots \overset{?}{\rightsquigarrow} (\Theta_1^{n_1}, \sigma_1^{n_1}) \overset{?}{\rightsquigarrow} \ldots$$

通过比较定义 4.11 和 4.12，很容易看出，上述运行必定是贪心运行。与事实 4.13 类似，当且仅当 $\tau \in M^\omega$ 描述了一条经过 $\mathrm{T}_{\mathcal{A}}$ 的路径，且其定义域 $\mathrm{dom}(\tau_0) = S$时，这个贪心 Safra 棋盘运行，以及由此延伸出的 $\mathfrak{S}(\mathrm{A}\ S,\ M)$上的运行才存在。后一个条件源于这样一个事实：$(\Theta_0,\ \sigma_0)$ 始终是 $S \in \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$上的 Safra 棋盘。由引理 4.14 和 4.15 可知，当且仅当 τ 满足迹条件时，上述 Safra 棋盘运行才是接受的。因此，只需论证 Rabin 条件 R在 $\mathfrak{S}(\mathrm{A}\ S,\ M)$上的运行中成立，当且仅当扩展的 Safra 棋盘运行是接受的。为此，注意到：如果存在某个芯片 $\gamma \in K$和某个 $N \in \omega$，使得 $\gamma \in \Theta_i$对所有 $N \leq i$都成立，且 γ被覆盖无穷

多个次 $(\Theta_i,\ \sigma_i)$，那么γ 从某个时刻起会出现在贪心 Safra 棋盘运行沿途的每个 Safra 棋盘上；仔细考察引理 4.18 即可看出，前一个条件正是这个意思。定义 4.11 规定，每当 γ 在 $(\Theta_i,\ \sigma_i)$中被覆盖时，贪心运行中的 $(\Theta_i,\ \sigma_i)$与 $(\Theta_{i+1},\ \sigma_{i+1})$之间就会发生一次 γ-重置。因此，Rabin 条件与 Safra 棋盘运行的接受条件完全类似：当且仅当存在某个芯片 γ，它最终不再被移除，且被无穷多次重置时，该条件才成立。■

我们通过推导 $\mathfrak{S}(\mathrm{A}\ S,\ M)$ 各个组成部分大小的一些界，来结束本节。请注意，这些并非 Safra 构造的最优界。读者可参阅 [18],，例如其中介绍了一种空间高效构造。

引理 4.21 设 A 为有限集，$O \subset \mathrm{Ob}(\mathrm{T}_{\mathcal{A}})$，且 $M \subseteq \text{Đ}_{X,\ Y \in O} \mathrm{Hom}(X,\ Y)$。将 Safra 自动机记为 $\mathfrak{S}(\mathrm{A}\ S,\ M) = (M,\ Q,\ \delta,\ s,\ R)$，并令 $K \coloneqq \max_{X \in O} |X|$。

1. 对任意 $X \in O$，有 $|\mathrm{Sb}(\mathrm{A}\ X,\ K)| \leq \text{Í}_{K\ |A|} \binom{K}{C} \cdot (|\mathrm{A}|+1)^C \cdot C! \cdot 2^{C \cdot |X| \cdot (|\mathrm{A}|-1)} = O(K!)$。2. $|Q| \leq \text{Í}_{X \in O} |\mathrm{Sb}(\mathrm{A}\ X,\ K)| \overset{C=1}{=} O(|O| \cdot K!)$。3. $|R| = K \cdot (|\mathrm{A}| + 1) = O(K)$。

证明 2. 可由 1. 直接推出。此外，3. 成立，因为 $|R| = |\overline{K}| = K \cdot (|\mathrm{A}| + 1)$。要理解 1. 中的界，注意一个满足 $|\Theta| = C$ 的棋盘 $(\underline{\Theta}, \sigma) \in \mathrm{Sb}(\mathrm{A}\ X, K)$ 由三个部分组成：一个选择 $\Theta \subseteq K$（若 $|\Theta| = C$，则有 $\binom{\overline{K}}{C} = \binom{K \cdot (|\mathrm{A}|+1)}{C}$ 种不同选择）、施加于 Θ 的一个线性序（共有 $|\Theta|! = C!$ 种），以及每个位置 (x, a) 上的一个栈 $S \subseteq \mathrm{P}(\Theta)$，其中 $a \neq \alpha$，即一个函数 $\sigma\colon X \times (\mathrm{A} \setminus \{\alpha\}) \to \mathrm{P}(\Theta)$（共有 $\left(2^C\right)^{|X| \cdot (|\mathrm{A}|-1)} = 2^{C \cdot |X| \cdot (|\mathrm{A}|-1)}$ 种）。此外，考虑到 $1 \leq |\Theta| = C \leq K \cdot |\mathrm{A}|$，即可得到 1. 中所述的界。■

5 重置证明系统

固定一个激活代数 A 和一个激活迹范畴 $\mathrm{T}_{\mathcal{A}}$。本节将证明，由映射到 $\mathrm{T}_{\mathcal{A}}$ 的迹解释所诱导的每个循环证明系统，都能导出一个可靠性条件基于 Safra 棋盘的循环证明系统。这为基于具体循环证明系统推导具体的基于 Reset 的证明系统提供了起点，正如我们在第6节所做的那样。

给定一个由迹解释 ι 诱导的循环证明系统 $\mathrm{R}\colon \mathrm{R} \to \mathrm{T}_{\mathcal{A}}$，重置证明系统 R(R) 通过给 R 中的序列式标注 A-Safra 棋盘而得到。具体而言，R-序列式 Γ 会标注为一个 Safra 棋盘 $(\Theta, \sigma) \in \mathrm{Sb}(\mathrm{A}\ \iota(\Gamma))$。R 中的每条推导规则$R \in$ 都会“提升”为 R(R) 中相应的推导规则，其中标注第 i个前提的 Safra 棋盘，是迹解释映射 r_i 在结论的 Safra 棋盘上进行转移后的结果。此外，系统 R(R) 还包含与 Safra 棋盘上三种“记账转换”对应的结构规则：弱化、重置和填充。R(R) 的可靠性条件要求每个简单环 $\pi(t)$，即芽 t 与其伴随节点之间的路径，都包含一次重置规则的应用；该规则会留下一个不变量$_{\mathcal{A}}$，它是控制的一个前缀，并且沿整个简单环保持不变。

伴随节点 $\beta(t)$ 之间的路径，都包含一次重置规则的应用；该规则会留下一个不变量 θ，它是控制的一个前缀，并且沿整个简单环保持不变。

定义 5.1 固定一个由迹解释 ι 到 $\mathrm{T}_{\mathcal{A}}$所诱导的循环证明系统 (Seq, R ρ,Pfs)。R 的重置证明系统 是循环证明系统 R(R) :=(R(Seq), R(R), R(ρ), R(Pfs))，其定义如下。R(R) 中的序列式是形如 Γ 的表达式；(Θ, σ)，其中 $\Gamma \in$Seq 是一个 R-序列式，而 $(\Theta, \sigma) \in$Sb(A $\iota(\Gamma)$) 是 Γ 的迹对象 $\iota(\Gamma)$ 上的 Safra 棋盘。次序 Θ 称为 Γ；(Θ, σ) 的控制 。R(R) 的推导规则由 R 中规则的标注版本以及额外的结构规则组成。结构规则由以下三个规则模式给出。

$$\textsc{Weak}\ \frac{\Gamma; (\Theta', \sigma') \text{ where } (\Theta, \sigma) \overset{W}{\rightsquigarrow} (\Theta', \sigma')}{\Gamma; (\Theta, \sigma)} \qquad \textsc{Reset}_{\gamma}\ \frac{\Gamma; (\Theta', \sigma') \text{ where } (\Theta, \sigma) \overset{R_\gamma}{\rightsquigarrow} (\Theta', \sigma')}{\Gamma; (\Theta, \sigma)}$$

$$\textsc{Pop}\ \frac{\Gamma; (\Theta', \sigma') \text{ where } (\Theta, \sigma) \overset{P}{\rightsquigarrow} (\Theta', \sigma')}{\Gamma; (\Theta, \sigma)}$$

对于每条规则 $R \in$R，其中 $\rho(R) = (\Gamma, \Delta_1, \ldots, \Delta_n)$，以及迹解释给出的映射 r_i：$\iota(\Gamma) \to \iota(\Delta_i)$，以下规则模式针对每个 $(\Theta, \sigma) \in$Sb(A $\iota(\Gamma)$) 给出规则：

$$R\ \frac{\Delta_1; (\Theta_1, \sigma_1) \text{ where } (\Theta, \sigma) \overset{r_1}{\rightsquigarrow} (\Theta_1, \sigma_1) \quad \ldots \quad \Delta_n; (\Theta_n, \sigma_n) \text{ where } (\Theta, \sigma) \overset{r_n}{\rightsquigarrow} (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

设 $D = (C, \lambda, \delta)$ 是 R(R) 的一个预证明。取某个 $t \in \mathrm{dom}(\beta)$，并令 $\pi(t) = (\Gamma_i; (\Theta_i, \sigma_i))_{i<n}$ 为从 $\beta(t)$ 到 t 的路径上的序列式。令 Θ 为所有 Θ_i 的最长公共前缀。 $\pi(t)$ 的一个不变量是 Θ 的任意前缀 θ，满足在 $\beta(t)$ 与 t 之间发生了一次 max(θ)-重置。有时我们会谈到 $\pi(t)$ 的不变量，此时指的是满足条件的最长前缀。R(R)- 预证明满足可靠性条件 R(Pfs)，当且仅当对于每个 $t \in \mathrm{dom}(\beta)$，$\pi(t)$ 中 $\beta(t)$ 与 t 之间的路径都有一个不变量。」

R 的一个 重置证明 是 R(R) 中的循环证明。这本质上是 R 中的循环证明，只是额外加入了以注释形式呈现的结构。任何与 $R \in$ R 对应的规则应用都会直接影响穿过预证明的轨迹，而结构规则则为控制Θ、 σ进行“记账”。这种直觉可以更形式化地表述：存在一个从 R(R) 到 R 的证明态射，它由去除注释而得（Θ、 σ）。

固定一个由 $\mathrm{T}_{\mathcal{A}}$ 上的轨迹解释所诱导的循环证明系统 R。函数*strip*：R(Seq) $\to$Seq 定义为 *strip*(Γ；（Θ， σ）) := Γ 在序列式上。

引理 5.2 对于一个由 $\mathrm{T}_{\mathcal{A}}$ 上的轨迹解释所诱导的循环证明系统 R，函数*strip*：R(Seq) $\to$Seq 可扩展为预证明态射*strip*：R(R) $\to$R。

^^**证明** 我们需要为每条规则 $R \in$R(R)指定 R 中一个对应的预证明 *strip*(R)。只需考虑两种情况：

- $\hat{R}$ 是结构规则：那么 R 具有如下形式

$$\hat{R}\frac{\Gamma;(\Theta',\sigma') \text{ where } (\Theta,\sigma) \overset{X}{\rightsquigarrow} (\Theta',\sigma')}{\Gamma;(\Theta,\sigma)}$$

其中 X 是 W， P 或 R_γ，对某个 $\gamma \in \Theta$ 成立。无论如何，我们需要找到一个预证明，其假设为 $strip(\Gamma;(\Theta,\ \sigma))\ =\Gamma$，前提为 $strip(\Gamma;(\Theta,\ \sigma))\ =\Gamma$。这样的预证明由 Γ 的恒等预证明给出，即三元组 $(\{\varepsilon\}\ \varepsilon \mapsto \Gamma,\ \emptyset)$。

- $\hat{R}$ 对应于一条规则 $R \in \mathrm{R}$：也就是说，存在 $R \in \mathrm{R}$，使得 $\rho(R)\ =(\Gamma,\ \Delta_1、\ \ldots\ldots、\ \Delta_n)$，并且有映射 r_i： $\iota(\Gamma)\ \rightarrow \iota(\Delta_i)$，由轨迹解释给出，而 $\hat{}R$ 的形式为

$$\hat{R}\frac{\Delta_1;(\Theta_1,\sigma_1) \text{ where } (\Theta,\sigma) \overset{r_1}{\rightsquigarrow} (\Theta_1,\sigma_1) \quad \ldots \quad \Delta_n;(\Theta_n,\sigma_n) \text{ where } (\Theta,\sigma) \overset{r_n}{\rightsquigarrow} (\Theta_n,\sigma_n)}{\Gamma;(\Theta,\sigma)}$$

那么，与第一种情况类似，我们需要找到一个证明，其开放叶节点为 Γ，在 R 中有 Δ_1、……、Δ_n。仅由一次 R 应用构成的预证明正合要求。■

引理 5.2 仅确立了 *strip* 是一个预证明态射，而非证明态射。证明后者则更为复杂。*strip* 构成 R(R) 与 R 之间的证明态射，这可以理解为一个相对可靠性结果：假设 R 是可靠的，即该系统只证明真的序列式。由于 *strip* 是一个证明态射，如果序列式 $\Gamma;(\Theta,\sigma)$ 可在 R(R) 中证明，那么通过该态射，就能在 R 中得到 $strip(\Gamma;(\Theta,\sigma))$ 的循环证明，因此 R(R) 是可靠的。

下文第 5.1 节将说明 *strip* 是一个证明态射。在第 5.2 节中，我们将证明 R(R) 相对于 R 的完备性定理：如果在 R 中存在 Π 的 Γ 的循环证明，那么就存在一个 $\Gamma;(\emptyset,(s,a)\ \mapsto \emptyset)$ 的循环证明 $\hat{}\Pi$。此外，$strip(\hat{}\Pi)$ 是 Π 的有限展开。

5.1 可靠性

可靠性证明依赖于循环证明中连通子图的概念。在循环证明中，每个连通子图都可以对应于一个子集 $\eta \subseteq \mathrm{dom}(\beta)$，我们称之为连通循环。给定一个处于循环标准形的循环树 C，连通循环是一个芽节点集合 $\eta \subseteq \mathrm{dom}(\beta)$，满足

(i) 存在某个基元素 $b(\eta)\ \in \eta$，使得对于每个 $t \in \eta$， $\beta(b(\eta))\ \leq \beta(t)$；(ii) 对于每个 $t_0\ \in \eta$，存在 $t_1,\ \ldots,\ t_n \in \eta$（其中 $n\ =\ 0$ 可能成立），使得对于每个 $i\ <\ n$， $\beta(t_i)\ \leq t_{i+1}$ 且 $t_n = b(\eta)$

对于循环树 $C\ =(T,\ \beta)$，子树是一个集合 $T\prime \subseteq T$，满足：若 $s,\ t \in T\prime$ 且 $s\ <\ u\ <\ t$ 依据前缀序，则 $u \in T\prime$；此外，若 $s \in T\prime \cap \mathrm{dom}(\beta)$，则 $\beta(s)\ \in T\prime$。C 的任意连通循环 η 都描述了 C 的一个子树 $C[\eta]\ =\{s \in T \mid \exists t \in \eta.\ \beta(t)\ \leq s \leq t\}$。

预证明的连通循环与其无限分支密切相关：对于穿过循环树的任意无限路径，该路径无限次经过的节点构成一个由以下内容描述的子树

连通循环。下文中，我们用序列 $\pi \in T^\omega$ 表示穿过循环树 $C = (T,\ \beta)$ 的无限路径，并要求对每个 $i \in \omega$，都有 $\pi_{i+1} \in \mathrm{Chld}(\pi_i)$ 或 $\pi_{i+1} = \beta(\pi_i)$。此外，记 $\mathrm{Inf}(\pi)$ 为 $:= \{s \in T \mid \pi_i = s$无限次出现$\}$，记 $\mathrm{Occ}(\pi)$ 为 $:= \{s \in T \mid \pi_i = s$对于某个 $i \in \omega\}$，并用 $s <_+ t$ 表示 $t \in \mathrm{Chld}(s)$。

引理 5.3 设 $\pi \in T^\omega$ 是一棵循环树 $C = (T,\ \beta)$（处于循环标准形）中的一条无限路径。则存在一个 C 的连通循环 η，使得 $\mathrm{Inf}(\pi) = C[\eta]$。

证明 不失一般性，假设 $\mathrm{Occ}(\pi) = \mathrm{Inf}(\pi)$。现在考虑 $\eta := \mathrm{Occ}(\pi) \cap \mathrm{dom}(\beta)$。我们将通过多个中间步骤证明，$\eta$ 是一个连通循环，并且确实有 $\mathrm{Inf}(\pi) = C[\eta]$。

1. 对于 $t \in \beta$，如果 $s \le t$并且 $s \notin \beta(t)$ 那么 $\beta(t) < s$：这是因为 $\le$ 是前缀关系。
2. 设 $s\alpha$ 是一条有限路径。则对于每个 α_i，可推出 $s \le \alpha_i$，或者存在某个 α_j，使得$j < i$，$\beta(\alpha_j) \le \alpha_i$并且 $\beta(\alpha_j) < s$。根据 $|\alpha|$进行归纳法证明。如果 $|\alpha| = 1$，那么 $s <_+ \alpha_0$ 或 $s \in \mathrm{dom}(\beta)$ 且 $\alpha_0 = \beta(s)$，这意味着 $\alpha_0 < s$，正如所需。现在令 $\beta(\alpha) = n + 1$，需要考虑四种情况

   - $s \le \alpha_{n-1}$ 和 $\alpha_{n-1} <_+ \alpha_n$：那么 $s \le \alpha_n$，显然成立。
   - $s \le \alpha_{n-1}$和 $\alpha_n = \beta(\alpha_{n-1})$：那么假设 $s \notin \beta(\alpha_{n-1})$。但这意味着 $\beta(\alpha_{n-1}) < s$必然成立，满足第二个条件。
   - $\beta(\alpha_j) \le \alpha_{n-1}$和 $\alpha_{n-1} <_+ \alpha_n$：同样，第二个条件对 α_n显然成立。
   - $\beta(\alpha_j) \le \alpha_{n-1}$和 $\alpha_n = \beta(\alpha_{n-1})$：如果 $\beta(\alpha_j) \notin \alpha_n$，那么 $\alpha_n = \beta(\alpha_{n-1}) < \beta(\alpha_j) < s$。

3. η 有一个基元素 $b(\eta)$：我们证明，若 $X \subseteq \eta$ 满足存在一个 $b \in X$ 使得$\beta(b) \le \eta \backslash X$（即对所有 $t \in \eta\backslash X$，$\beta(b) \le t$），则由 $|X|$上的归纳法可得 $b(\eta) \in X$。若 $|X| = 1$，则显然 $X = \{b(\eta)\}$。现在令 $|X| > 1$，取 π的一个有限片段 $\beta(b)\alpha$，使得 $\mathrm{Occ}(\alpha) =$无穷(π)。根据前一个结果，要么对所有 $t \in \eta$，都有 $\beta(b) \le \beta(t)$，即 $b = b(\eta)$；要么存在某个 $b' \in \eta$，使得 $\beta(b') < \beta(b)$。对于后一种情况，由于否则 $\beta(b) \le \beta(b')$，所以 $b' \in X$，并由传递性得 $\beta(b') \le \eta\backslash X$。然后用 $X' := X\backslash\{b\}$ 和 $b' \in X'$ 继续论证，同时注意到 $|X'| = |X| - 1$。

4. 对于每个 $t \in \eta$都存在 $l_0 \dots l_n$满足 $l_0 = t$，$l_n = b(\eta)$ 且 $\beta(l_i) < l_{i+1}$对所有 $i < n$：直接观察可知，对于每个 $l_0 \in \eta$，都存在一条有限子路径 $l_0 \alpha b(\eta)$，它属于 π，并描述这样一个叶子序列。

5. 无穷$(\pi) \subseteq C[\eta]$：对于每个 $s \in$无穷(π)，我们必须找到一个 $t \in \eta$满足 $\beta(t) \le s \le t$。首先，如果 $s \in$无穷(π)，那么必定存在某个 $s \le t \in$定义域$(\beta) \cap$无穷$(\pi) = \eta$，否则 π便不可能从s 无限延续下去。现在假设对于所有满足 $s \le t$ 的 $t \in \eta$，都有 $s < \beta(t)$。一旦 π越过 s，它就再也无法“跳回”到 s以下：它能到达的“最低”点是某个 $s \le t$对应的 $\beta(t)$。但这样一来，s不可能被到达多次，这与 $s \in$无穷(π) 矛盾。

6. 设 $s\alpha$ 是一条有限路径，$s < t$并且 $t \notin \mathrm{Occ}(\alpha)$ 则 $\mathrm{Occ}(\alpha) \cap \mathrm{Up}(t) = \emptyset$：对$|\alpha|$应用归纳法进行证明。若 $|\alpha| = 1$，则 $s <_+ \alpha_0$，因为 $s \notin \mathrm{dom}(\beta)$，而 $s < t$。在这种情况下，$\alpha_0 \in \mathrm{Up}(t)$ 只有在 $\alpha_0 = t$ 时才可能成立，这与假设矛盾。若 $|\alpha| = n + 1$，假设 $\alpha_n \in \mathrm{Up}(t)$。根据与 $|\alpha| = 1$相同的论证，这意味着 $\alpha_{n-1} \not<_+ \alpha_n$。因此 $\alpha_n = \beta(\alpha_{n-1})$ 且 $t \le \alpha_n \le \alpha_{n-1}$，这与归纳假设矛盾。

7. $C[\eta] \subseteq \mathrm{Inf}(\pi)$：设 $\beta(t) \leq s \leq t$，其中 $t \in \eta$。存在无穷多个有限的“子片段” $\beta(t)\alpha t$，它们属于 π。于是 $s \in \mathrm{Occ}(\alpha)$，因为根据前一个结果，$t$无法以其他方式从 $\beta(t)$到达。■

可靠性论证背后的思路相当简单：对于 R(R)证明 的每个连通循环 η，都可以找到一个所有循环 η共有的“共享不变量”。这类不变量的性质使我们能够得出结论：从证明中的任意一条无限路径上读出控制（Θ，σ），若该路径恰好无限次访问 $C[\eta]$，那么所得运行必定是一个接受的 Safra 棋盘运行，因此底层轨迹必定满足迹条件。论证中最复杂的一步是证明这类共享不变量的存在性。

在本节余下部分，固定一个由迹解释ι诱导的循环证明系统 $\mathrm{R}: \mathrm{R} \to \mathrm{T}_{\mathcal{A}}$。

命题 5.4 设 (C, λ, δ) 是一个 R(R)证明，且 η是 C的一个连通循环。那么存在某个 $t \in \eta$，使得 $\beta(t) < t$的不变量 θ是每个带有 $s \in \eta$的 $\beta(s) < s$的不变量的前缀。

证明：注意，可以在 $\sqsubset$ 上施加一个线性序 η，使得对于任意 $s_0 \in \eta$，都可以通过取 $s_1, \ldots, s_n$，并使它们都小于$\sqsubset s_0$，来满足连通循环定义中的条件 (ii)。显然，对于 $s \in \eta$，每个下集 $\mathrm{Down}_{\sqsubseteq}(s)$ 都是一个连通循环。我们按照 $\sqsubset$-序进行归纳，证明对于每个 $s \in \eta$，连通循环 $\mathrm{Down}_{\sqsubseteq}(s)$ 都包含一个循环$\beta(t) < t$，其不变量是 $\mathrm{Down}_{\sqsubseteq}(s)$ 中所有不变量的前缀。于是，取 $\eta = \mathrm{Down}_{\sqsubseteq}(\max_{\sqsubset} \eta)$ 即可得出结论。对于 $\sqsubset$-最小元素，结论显然成立。因此，取某个 $s \neq b(\eta)$，并考察 $\eta' := \mathrm{Down}_{\sqsubseteq}(s) \setminus \{s\}$。显然，对于某个 s'，有 $\eta' = \mathrm{Down}_{\sqsubseteq}(s')$，因此其中有一个元素$t' \in \eta'$，其不变量 θ'是 η' 中所有循环的不变量的前缀。我们首先证明路径 $\beta(s) \in C[\eta']$：由于 $\mathrm{Down}_{\sqsubseteq}(s)$ 是一个连通循环，因此必定存在一个最短序列 $s_1, \ldots, s_n \in \eta'$，满足 $0 < n$，且 $\beta(s_i) \leq s_{i+1}$ 和 $s_n = b(\eta)$。于是，必有 $\beta(s_{i+1}) < \beta(s_i)$，否则就可以避开经过 s_{i+1} 的“绕行”，从而缩短该序列。这意味着 $\beta(s_1) < \beta(s) \leq s_1$，也就是说，$\beta(s)$ 出现在路径 $\beta(s_1) \leq s_1$ 上。由于 θ' 是 $\beta(s_1) \leq s_1$ 的不变量的前缀，它也必定是 $\beta(s)$ 处控制的前缀。因此，$\beta(s) \leq s$ 的不变量 θ只有两种可能：要么 θ' 是它的前缀，要么它是 θ'的前缀。前一种情况下，t' 仍然是 $\mathrm{Down}_{\sqsubseteq}(s)$ 中不变量为所有其他不变量前缀的元素；后一种情况下，根据前缀关系的传递性，s 便是这个新的元素。■

定理 5.5 函数 $strip: \mathrm{R}(\mathrm{Seq}) \to \mathrm{Seq}$ 是一个循环证明系统态射。

证明 该断言的部分内容已在引理 5.2 中证明。只需证明：如果Π 是 R(R) 中的一个证明，那么 $strip(\Pi)$ 就是 R 中的一个证明。

设 $\Pi = (C, \lambda, \delta)$ 是 R(R) 中 Γ; (Θ, σ) 的一个循环证明。为此，只需证明：每条由路径 π' 经 $\mathrm{strip}(\Pi) = (C', \lambda', \delta')$ 所诱导的路径 $\mathrm{b}\pi': \omega \to \mathrm{T}_{\mathcal{A}}$，都满足 $\mathrm{T}_{\mathcal{A}}$ 的迹条件。必定存在一条路径 π，沿着 π'“跟随” Π。设 $(\Theta_i, \sigma_i)_{i \in \omega}$ 满足 $\lambda(\pi' i) = \mathrm{b}\Gamma_i; (\Theta_i, \sigma_i)$。显然，$(\Theta_i, \sigma_i)_{i \in \omega}$ 是 τ 的一条 Safra 棋盘运行，且 $\tau_i := \pi'(i < i + 1)$。由引理 4.15，要证明 P 满足全局轨迹

条件。由引理 5.3，存在 C 的一个连通循环 η，使得 $\mathrm{Inf}(\pi) = C[\eta]$。因此，从某个位置开始，$\pi$ 始终处于 $C[\eta]$ 中，例如从索引 N 开始。进一步由命题 5.4，存在 $t \in \eta$，使得 $\beta(t) < t$ 的不变量 θ 是 $C[\eta]$ 中各注释里所有控制的前缀。这意味着对所有 $i \geq N$，$\theta \leq \Theta_i$。考虑芯片 $\gamma := \max(\theta)$：由前面的观察可知，对所有 $i \geq N$，$\gamma \in \Theta_i$。此外，γ 在路径 $\beta(t) < t$ 上被重置，因为 θ 是该路径的不变量。因此，沿着 $(\Theta_i, \sigma_i)_{i\in\omega}$ 会发生无限多次 γ-重置，使其成为所需的接受运行。■

推论 5.6（可靠性）若 Π 是 $\Gamma; (\Theta, \sigma)$ 在 R(R) 中的一个证明，则 $strip(\Pi)$ 是 Γ 在 R 中的一个证明。

5.2 完备性与证明搜索

本节证明 R(R) 相对于 R 的完备性，即任何可在 R 中证明的序列式，都可在 R(R) 中证明。我们将证明，如果 R 只有有限个推导规则，就可以在 R(R) 中进行证明搜索。因此，设 R 满足推导规则集合 F 为有限集，且其可靠性条件由 $\mathrm{T}_{\mathcal{A}}$ 中的迹解释所诱导。回顾一下，$\mathrm{T}_{\mathcal{A}}$ 的对象是有限集。

我们首先为 R 构造一个证明搜索系统 S(R)。与 R(R) 类似，S(R) 的序列式是带有 Safra 棋盘标注的 R-序列。不过，S(R) 的标注受到限制，必须是 K-稀疏的，其中 K 取适当的值。关键在于，如果 R 的推导规则数量有限，系统 S(R) 的推导规则数量也有限；这与 R(R) 不同，因此更便于进行证明搜索。更具体地说，S(R) 的每条规则都通过以下方式构造：取 R 的一条规则 $R \in$，用一个 K-稀疏的 Safra 棋盘标注其结论，并用经由迹解释映射 r_i 进行贪心转移后得到的 K-稀疏 Safra 棋盘标注第 i 个前提。可靠性条件是定义 4.19 中 Safra 自动机接受条件的一个“全局变体”。

定义 5.7 固定 $K := \max\{|\iota(\Gamma)| \mid \Gamma \in \mathrm{Seq}\}$。R 的证明搜索系统是下文定义的系统 $\mathrm{S(R)} = (\mathrm{S(Seq)}, \mathrm{S(R)}, \mathrm{S}(\rho), \mathrm{S(Pfs)})$。S(R) 的序列式是表达式 $\Gamma; (\Theta, \sigma)$，其中 $\Gamma \in \mathrm{Seq}$ 是一个 R-序列式，且 $(\Theta, \sigma) \in \mathrm{Sb}(\mathrm{A}\ \iota(\Gamma)\ K)$ 是 K-稀疏的、定义在 $\iota(\Gamma)$ 上的 Safra 棋盘。S(R) 的规则包括：对于每个满足 $\rho(R) = (\Gamma, \Delta_1, \ldots, \Delta_n)$ 的 $R \in \mathrm{R}$、映射 $r_i : \iota(\Gamma) \to \iota(\Delta_i)$，以及每个 $(\Theta, \sigma) \in \mathrm{Sb}(\mathrm{A}\ \iota(\Gamma)\ K)$，都有以下规则：

$$\mathrm{R}(\Theta,\sigma)\ \frac{\Delta_1; (\Theta_1, \sigma_1) \text{ where } (\Theta, \sigma) \overset{r_1}{\rightsquigarrow_g} (\Theta_1, \sigma_1) \quad \ldots \quad \Delta_n; (\Theta_n, \sigma_n) \text{ where } (\Theta, \sigma) \overset{r_n}{\rightsquigarrow_g} (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

S(R)-预证明 Π 满足可靠性条件 S(Pfs)，当且仅当对于经过 Π 的每条无限路径 $(\Gamma_i; (\Theta_i, \sigma_i))_{i\in\omega}$，都存在某个 $N \in \omega$ 和 $\gamma \in \tilde{\mathrm{N}}_{N\leq i} \Theta_i$，使得 γ 被覆盖无穷多次。

⌟

由于我们固定 $\overset{\tau}{\rightsquigarrow_g}$ 为定义在 K-稀疏 Safra 棋盘上的单射函数，因此 R 和 (Θ, σ) 的选择唯一确定了规则 $\mathrm{R}(\Theta, \sigma)$。

引理 5.8 函数 $expand$: S(Seq) $\to$ R(Seq)，满足 $expand(\Gamma; (\Theta, \sigma)) := \Gamma; (\Theta, \sigma)$，可以扩展为一个证明态射。

证明 为证明这一断言，首先选取某个 $R(\Theta, \sigma) \in S(R)$，其排列如下

$$R(\Theta,\sigma)\frac{\Delta_1; (\Theta_1, \sigma_1) \quad \dots \quad \Delta_n; (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

于是存在 $R \in R$，其中 $\rho(R) = (\Gamma, \Delta_1, \dots, \Delta_n)$，以及由轨迹解释给出的态射 r_i: $\iota(\Gamma) \to \iota(\Delta_i)$。我们需要在 R(R) 中找到与之对应的预证明 $expand(R(\Theta, \sigma))$。注意，对每个 $i \le n$，都存在 $(\Theta, \sigma) \overset{r_i}{\rightsquigarrow}_g (\Theta_i, \sigma_i)$，其中展开后的序列为展开后的序列

$$(\Theta, \sigma) \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_r^1, \sigma_r^1) \dots \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_r^k, \sigma_r^k) \overset{P}{\rightsquigarrow} (\Theta_p, \sigma_p) \overset{\tau_i^{\Delta_i}}{\rightsquigarrow} (\Theta_i^*, \sigma_i^*) \overset{T}{\rightsquigarrow} (\Theta_i, \sigma_i)$$

其中，初始的 R_γ-步和 P-步由所有 $i \le n$ 共享（见引理 4.18）。于是我们可以如下推导 $expand(R(\Theta, \sigma))$:

$$\textsc{Weak}\,\frac{\textsc{R}\,\dfrac{\textsc{Weak}\,\dfrac{\Delta_1; (\Theta_1, \sigma_1)}{\Delta_1; (\Theta_1^*, \sigma_1^*)} \quad \dots \quad \textsc{Weak}\,\dfrac{\Delta_n; (\Theta_n, \sigma_n)}{\Delta_n; (\Theta_n^*, \sigma_n^*)}}{\textsc{Pop}\,\dfrac{\Gamma; (\Theta_p, \sigma_p)}{\Gamma; (\Theta_r^k, \sigma_r^k)}}}{\textsc{Reset}_{\gamma_2}\,\dfrac{\vdots}{\textsc{Reset}_{\gamma_1}\,\dfrac{}{\Gamma; (\Theta_r^1, \sigma_r^1)}}\ \Gamma; (\Theta_0, \sigma_0)}$$

为证明 expand 保持可靠性条件，设 $\Pi = (C, \lambda, \delta)$ 是一个 S(R)证明，且 expand $(\Pi) = (C', \lambda', \delta')$ 是其 expand 翻译。现在考虑某个 $t' \in \mathrm{dom}(\beta')$，以及与之关联的路径 $(\Gamma_i'; (\Theta_i', \sigma_i'))_{i \le n}$，该路径位于 expand (Π) 中的 $\beta'(t')$ 与 t' 之间。存在一条相应的路径 $(\Gamma_i; (\Theta_i, \sigma_i))_{i \le m}$，经过 Π。根据 S(R) 的可靠性条件，从根节点出发并在 $(\Gamma_i; (\Theta_i, \sigma_i))_{i \le m}$ 上无限循环的路径 Π 必须有某个 $\gamma \in \tilde{\mathrm{N}}_{i \le m} \Theta_i$，在 $(\Gamma_i; (\Theta_i, \sigma_i))_{i \le m}$ 上的某处被覆盖。由引理 4.18 可知，$\gamma \in \tilde{\mathrm{N}}_{i \le n} \Theta_i'$ 也成立；并且 γ 在 $(\Gamma_i; (\Theta_i, \sigma_i))_{i \le m}$ 上的某处被覆盖这一事实意味着，沿着 $(\Gamma_i'; (\Theta_i', \sigma_i'))_{i \le n}$ 的某处必须发生一次 γ- 重置。剩下要证明的是，该循环具有一个不变量 θ，满足 $\max(\theta) = \gamma$。取 $\theta := \{\gamma' \in \Theta \mid \gamma' \le \gamma\}$ 后，剩下要证明的是，循环过程中 θ 内没有任何芯片消失。但这不可能发生，因为在到达芽之前，它不可能在“γ 下方被替换”，因为这需要先从控制中移除 γ。因此，expand (Π) 中的每个循环都有一个不变量 θ，并伴有一次 $\max(\theta)$-重置，这意味着 expand (Π) 是一个 R(R)证明。 ∎

我们采用证明搜索系统的概念，证明 R(R) 相对于 R 的完备性：每个在 R 中可证的序列式，在 R(R) 中也可证。由于该证明基于证明搜索过程，我们得到的结果甚至更强：R(R)证明本质上是 R-证明的展开

。固定一个预证明（属于任意推导系统 S） $\Pi = ((T,\ \beta)\ \lambda,\ \delta)$，在芽处对其定义域进行展开 Π 于芽处$t \in \mathrm{dom}(\beta)$，得到预证明 $\Pi' := ((T', \beta')\ \lambda',\ \delta')$，其

$$T' := T \cup \{tu \mid \beta(t)u \in T\} \qquad \lambda'(s) := \begin{cases} \lambda(s) & s \in T \\ \lambda(\beta(t)u) & s = tu \end{cases}$$

δ'以类似方式定义为 λ'和 $\beta' := \beta \setminus \{(t, \beta(t))\} \cup \{(tu, c)\}$，其中 $t = \beta(t)u$，且$c = t$或 $c = \beta(t)$。Π' 是 Π 的一个预证明，当且仅当它是 Π 的一个展开，也就是说，可以通过反复展开 Π得到 Π'。

定理 5.9（完备性）设 $(\mathrm{Seq}, \mathrm{R}\ \rho, \mathrm{Pfs})$ 是由 $\mathrm{T}_{\mathcal{A}}$上的轨迹解释所诱导的循环证明系统。如果在 R 中存在 Seq 的一个循环证明 $\Pi\ \ \Gamma \in$，那么在 R(R) 中存在Γ的一个证明 Π'；$(\emptyset, (x, a) \mapsto \emptyset)$。此外，$strip(\Pi')$ 是 Π的一个展开。

证明设 $\Pi = (C, \lambda, \delta)$ 是 R 中 Γ 的一个证明。考虑 R 的子系统 $\mathrm{R}' := (\mathrm{im}(\lambda), \mathrm{im}(\delta)\ \rho\ \restriction \mathrm{R}', C \cap \mathrm{Pp}(\mathrm{R}'))$。由于 Π 是有限的，R′也是有限的。现在固定

$$M := \{r_i : \iota(\Gamma) \to \iota(\Delta_i) \mid R \in \mathcal{R}', \rho(R) = (\Gamma, \Delta_1, \ldots, \Delta_n), r_i \text{ given by the trace interpretation}\}$$

并根据定义 4.19 构造 Safra 自动机 $\mathfrak{S}(\mathrm{A}\ i(\Gamma)\ M) = (M,\ Q,\ s,\ \delta_{\mathfrak{S}},\ R_{\mathfrak{S}})$。我们构造 Rabin 树自动机 $\mathfrak{A} = (\mathrm{Seq}',\ Q', \Delta,\ s',\ R')$，其中

$$\begin{aligned}
Q' &:= \{(s, (\Theta, \sigma)) \mid s \in C \setminus \mathrm{dom}(\beta), \Gamma := \lambda(s), (\iota(\Gamma), (\Theta, \sigma)) \in Q\} \\
\Delta' &:= \{((s, (\Theta, \sigma)), (\Delta_1, \ldots, \Delta_n), ((t_1, (\Theta_1, \sigma_n)), \ldots, (t_n, (\Theta_n, \sigma_n)))) \mid \\
&\qquad \text{if } s \in T \setminus \mathrm{Leaf}(T), \mathrm{Chld}(s) = \{t_1, \ldots, t_n\}, \rho(\delta(s)) = (\Gamma, \Delta_1, \ldots, \Delta_n), \\
&\qquad r_i : \iota(\Gamma) \to \iota(\Delta_i) \text{ given by the trace interpretation} \\
&\qquad \text{and } \delta_{\mathfrak{S}}((\iota(\Gamma), (\Theta, \sigma)), r_i) = (\iota(\Delta_i), (\Theta_i, \sigma_i))\} \\
s' &:= (\varepsilon, (\Theta_0, \sigma_0)) \text{ where } s = (\iota(\lambda(\varepsilon)), (\Theta_0, \sigma_0)) \\
R' &:= \{(\{(s, (\Theta, \sigma)) \mid (\iota(\lambda(s)), (\Theta, \sigma)) \in G\}, \{(s, (\Theta, \sigma)) \mid (\iota(\lambda(s)), (\Theta, \sigma)) \in B\}) \mid (G, B) \in R_{\mathfrak{S}}\}
\end{aligned}$$

很容易看出，$L(\mathfrak{A})$ 恰好包含了 Π 的“无限展开”，而 $\mathrm{Sb}(\mathrm{A}\ i(\Gamma)\ K)$ 沿着其各分支的路径 $P: \omega \to \mathrm{T}_{\mathcal{A}}$ 存在一次成功运行。换言之，$L(\mathfrak{A})$ 中唯一的树对应于 Π 的展开。根据命题 3.12，$L(\mathfrak{A})$ 中存在一棵正则树，它在 $\mathfrak{A}$ 上有一次正则运行。将每个与规则 $R := \delta(s)$ 对应的步骤 $(\ (s, (\Theta, \sigma)), \mathrm{h}\Delta_1, \ldots, \Delta_n\mathrm{i}, (\ (t_1, (\Theta_1, \sigma_n)) \ldots, (t_n, (\Theta_n, \sigma_n))))$ 替换为相应的 $\mathrm{S}(\mathrm{R}')$-规则 $\mathrm{R}(\Theta, \sigma)$，即可将该运行转化为一个 $\mathrm{S}(\mathrm{R}')$-预证明 Π'：

$$\mathrm{R}(\Theta,\sigma)\frac{\Delta_1; (\Theta_1, \sigma_1) \quad \ldots \quad \Delta_n; (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

由于该运行满足 Rabin 条件 R'，预证明 Π' 满足 $\mathrm{S}(\mathrm{R}')$ 的可靠性条件。Π'的结论是 Γ；$(\emptyset,\ (x,\ a) \mapsto \emptyset)$，因为这对应于 A 的初始状态 s'。现在，根据引理 5.8，$expand(\Pi')$ 是一个 $\mathrm{R}(\mathrm{R}')$证明（因此也是一个 $\mathrm{R}(\mathrm{R})$证明）。由于 $\mathfrak{A}$的状态由 C的节点标记，因此 $\mathfrak{A}$ 上的正则运行必定对应于 Π的一次展开。因此，$strip(\Pi')$ 也必定是 Π 的一次展开。∎

6 推导具体重置证明系统

本节中，我们将应用相关结果，为文献中的 3 个循环证明系统推导出重置证明系统：第 6.1 节的循环算术、第 6.2 节的循环 G¨odel 的 T，以及第 6.3 节的模态 μ- 演算。对于每个系统，抽象重置系统 R(R) 都将作为起点。不过，我们推导出的重置系统与“朴素”的系统 R(R) 均有几处“人体工学调整”，并采用了更偏句法的注释机制。在接下来的各节中，我们首先回顾以全局轨迹条件表述的原始循环证明系统，然后定义我们提出的重置证明系统。

6.1 皮亚诺算术

循环算术最初由 Alex Simpson 在 [22] 中提出。它是一种循环证明系统，能够证明与皮亚诺算术相同的定理。

6.1.1 循环算术

循环算术 的项语言和公式语言如下所示。该公式语言采用非标准形式，将不等式 $s < t$ 视为原始概念，而非定义出来的概念。如下文将会清楚说明的，这简化了循环算术全局迹条件的定义。

$$
\begin{aligned}
s, t \in \textsc{Term} &::= x \mid 0 \mid Ss \mid s + t \mid s \cdot t \\
\varphi, \psi \in \textsc{Form} &::= s = t \mid s < t \mid \bot \mid \varphi \wedge \psi \mid \varphi \vee \psi \mid \varphi \to \psi \mid \forall x.\varphi \mid \exists x.\varphi
\end{aligned}
$$

用 $[t/x]$ 表示通常的代入操作，即将项 t 代入项或公式中变量 x 的所有自由出现处。这是一个偏操作：当 t 中的自由变量与 φ 中的约束变量不互异时，$\varphi[t/x]$ 无定义。下文中，书写 $\varphi[t/x]$ 也将同时表示所得公式有定义这一断言。

定义 6.1 循环算术的序列式 是形如 $\Gamma \Rightarrow \Delta$ 的表达式，其中 Γ, Δ 是由公式构成的有限集。循环算术序列式的集合记作 Seq_{CA}。记 Γ, φ 表示 $\Gamma \cup \{\varphi\}$，记 Γ, Γ' 表示 $\Gamma \cup \Gamma'$。循环算术的推导规则 由以下一阶逻辑标准规则的选取构成，

$$\mathrm{Ax}\frac{}{\Gamma, \varphi \Rightarrow \varphi, \Delta} \qquad \rightarrow\mathrm{L}\frac{\Gamma, \varphi \Rightarrow \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma, \varphi \rightarrow \psi \Rightarrow \Delta} \qquad \rightarrow\mathrm{R}\frac{\Gamma, \varphi \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \varphi \rightarrow \psi, \Delta}$$

$$\wedge\mathrm{L}\frac{\Gamma, \varphi, \psi \Rightarrow \Delta}{\Gamma, \varphi \wedge \psi \Rightarrow \Delta} \qquad \wedge\mathrm{R}\frac{\Gamma \Rightarrow \varphi, \Delta \quad \Gamma \Rightarrow \psi, \Delta}{\Gamma \Rightarrow \varphi \wedge \psi, \Delta} \qquad \vee\mathrm{L}\frac{\Gamma, \varphi \Rightarrow \Delta \quad \Gamma, \psi \Rightarrow \Delta}{\Gamma, \varphi \vee \psi \Rightarrow \Delta}$$

$$\vee\mathrm{R}\frac{\Gamma \Rightarrow \varphi, \psi, \Delta}{\Gamma \Rightarrow \varphi \vee \psi, \Delta} \qquad \forall\mathrm{L}\frac{\Gamma, \varphi[t/x] \Rightarrow \Delta}{\Gamma, \forall x.\varphi \Rightarrow \Delta} \qquad \forall\mathrm{R}\frac{\Gamma \Rightarrow \varphi, \Delta \quad x \notin \mathrm{FV}(\Gamma, \Delta)}{\Gamma \Rightarrow \forall x.\varphi, \Delta}$$

$$\exists\mathrm{L}\frac{\Gamma, \varphi \Rightarrow \Delta \quad x \notin \mathrm{FV}(\Gamma, \Delta)}{\Gamma, \exists x.\varphi \Rightarrow \Delta} \qquad \exists\mathrm{R}\frac{\Gamma \Rightarrow \varphi[t/x], \Delta}{\Gamma \Rightarrow \exists x.\varphi, \Delta} \qquad \bot\mathrm{L}\frac{}{\Gamma, \bot \Rightarrow \Delta}$$

$$=\mathrm{L}\frac{\Gamma[t/x, s/y] \Rightarrow \Delta[t/x, s/y] \quad x, y \notin \mathrm{FV}(s, t)}{\Gamma[s/x, t/y], s = t \Rightarrow \Delta[s/x, t/y]} \qquad =\mathrm{R}\frac{}{\Gamma \Rightarrow t = t, \Delta}$$

以及以下结构规则，

$$\mathrm{Wk}\frac{\Gamma \Rightarrow \Delta}{\Gamma, \Gamma' \Rightarrow \Delta, \Delta'} \qquad \mathrm{Cut}\frac{\Gamma, \varphi \Rightarrow \Delta \quad \Gamma \Rightarrow \varphi, \Delta}{\Gamma \Rightarrow \Delta} \qquad \mathrm{Sub}\frac{\Gamma \Rightarrow \Delta}{\Gamma[s/x] \Rightarrow \Delta[s/x]}$$

以下算术专用公理

$$\begin{array}{lll} s < t, t < u \Rightarrow s < u & s < t \Rightarrow Ss < St & \Rightarrow s + St = \mathrm{S}(s + t) \\ s < t, t < s \Rightarrow & \Rightarrow s < t, s = t, t < s & \Rightarrow t \cdot 0 = 0 \\ s < t, t < Ss \Rightarrow & \Rightarrow t < St & \Rightarrow s \cdot St = (s \cdot t) + s \\ t < 0 \Rightarrow & \Rightarrow t + 0 = t & \end{array}$$

以及算术专用推导规则

$$\mathrm{S}\frac{\Gamma, t = Sx \Rightarrow \Delta \quad x \text{ fresh}}{\Gamma, 0 < t \Rightarrow \Delta}$$

⌟

请注意，使用循环算术的规则和归纳模式$(\forall x.(\forall y.y < x \rightarrow \varphi[y/x]) \rightarrow \varphi) \rightarrow \forall x.\varphi$构成的无假设、非循环预证明，恰好证明了皮亚诺算术的所有定理。循环算术也恰好证明了皮亚诺算术的所有定理，只是以全局轨迹条件取代了归纳模式（证明见 [22, 定理 6] ）。

若一个项 t 出现在序列式 $\Gamma \Rightarrow \Delta$ 中，指它出现在 Γ 或 Δ中的某个公式内，也可以是作为另一项的子项出现。用 $\mathrm{Term}(\Gamma \Rightarrow \Delta)$ 表示出现在 $\Gamma \Rightarrow \Delta$中的项的集合。令 $R \in$ 循环算术满足 $\rho(R) = (\Gamma \Rightarrow \Delta, \Gamma_1 \Rightarrow \Delta_1, \ldots\ldots, \Gamma_n \Rightarrow \Delta_n)$，即以 $\Gamma \Rightarrow \Delta$ 为其结论，且 $\Gamma_i \Rightarrow \Delta_i$为其某个前提。固定 $t \in \mathrm{Term}(\Gamma \Rightarrow \Delta)$ 和 $t' \in \mathrm{Term}(\Gamma_i \Rightarrow \Delta_i)$。若满足以下三个条件之一，则称项 t'为 t的前驱，记作 $t' \leftarrow_R^i t$：

- R是（代换规则）的一个实例，而 $\Gamma = \Gamma'[s/x]$，$\Delta = \Delta'[s/x]$ 和 $t = t'[s/x]$;

• 或者 R 是除 （代换规则）以外某条规则的一个实例，并且 $t = t_\prime$；

• 或 R 是 (=L) 的一个实例，且 $\Gamma = \Gamma_\prime[s/x,\ t/y]$，$\Gamma_\prime = \Gamma_{\prime\prime}[t/x,\ s/y]$；对于 Δ，情况类似，并且存在一个项 $t_{\prime\prime}$ 使得 $t = t_{\prime\prime}[s/x,\ t/y]$ 且 $t_\prime = t_{\prime\prime}[t/x,\ s/y]$。

回顾一下，布尔值 $\mathbb{B} = \{0, 1\}$ 配上通常的并运算和 $\alpha := 1$，构成一个激活代数。这是为 循环算术 指定全局轨迹条件时最自然的激活代数。

定义 6.2 轨迹解释 ι: 循环算术 $\to \mathbb{T}_\mathbb{B}$ 定义为 $\iota(\Gamma \Rightarrow \Delta) := \mathrm{Term}(\Gamma \Rightarrow \Delta)$；对于任意满足 $\rho(R) = (\Gamma \Rightarrow \Delta,\ \Gamma_1 \Rightarrow \Delta_1,\ \ldots,\ \Gamma_n \Rightarrow \Delta_n)$ 的 $R \in$ 循环算术，迹映射 r_i: $\mathrm{Term}(\Gamma \Rightarrow \Delta) \to \mathrm{Term}(\Gamma_i \Rightarrow \Delta_i)$ 定义为

$$
\begin{aligned}
r_i := &\{(t, 0, t') \mid t \in \textsc{Term}(\Gamma \Rightarrow \Delta), t' \in \textsc{Term}(\Gamma_i \Rightarrow \Delta_i) \text{ and } t' \leftarrow^i_R t\} \cup \\
&\{(t, 1, s) \mid t \in \textsc{Term}(\Gamma \Rightarrow \Delta), t', s \in \textsc{Term}(\Gamma_i \Rightarrow \Delta_i) \text{ and } t \leftarrow^i_R t' \text{ and } s < t' \in \Gamma_i\}
\end{aligned}
$$

这条轨迹 解释导出了循环算术 的可靠性条件，如定义 3.6 所述。 ⌟

6.1.2 重置算术

我们提出一种称为重置算术 RA的皮亚诺算术循环证明系统。它基于循环算术 所导出的重置系统 R(循环算术)，并作了一些细微修改。

重置算术 的序列式是形如 $\Theta; \sigma \mid \Gamma \Rightarrow \Delta$ 的表达式，其中 $\Gamma \Rightarrow \Delta$ 是一个循环算术序列式，Θ 是由互不相同的字符组成、称为控制 的序列，σ 是一个赋值 $t \mapsto u$的有限集，其中 t 是Γ, Δ 中的项，u 是 $u \sqsubseteq \Theta$的子序列。重置算术序列式的集合记作 Seq_{RA}。对于一个赋值$t \mapsto u$，记 $(t \mapsto u)[s/x]$ 为 $t[s/x] \mapsto u$。此记法可推广至赋值集 $\sigma[s/x]$。控制为空的序列式 $\varepsilon; \emptyset \mid \Gamma \Rightarrow \Delta$ 与循环算术序列式 $\Gamma \Rightarrow \Delta$ 等同。

定义 6.3 重置算术的推导规则 如下所列。在每条规则中，$\Theta'; \sigma'$表示先从σ 中移除所有赋给未在前提中出现的项的赋值，然后移除 Θ 中未赋给至少一个项的所有字母。RA的规则包含经过调整以“妥善处理”控制 $\Theta; \sigma$的循环算术 规则。请注意，Wk 也

允许对赋值进行“弱化” σ。

$$\mathrm{Ax}\frac{}{\Theta;\sigma \mid \Gamma, \varphi \Rightarrow \varphi, \Delta} \qquad \rightarrow\mathrm{L}\frac{\Theta';\sigma' \mid \Gamma, \varphi \Rightarrow \Delta \quad \Theta';\sigma' \mid \Gamma \Rightarrow \psi, \Delta}{\Theta;\sigma \mid \Gamma, \varphi \to \psi \Rightarrow \Delta}$$

$$\rightarrow\mathrm{R}\frac{\Theta;\sigma \mid \Gamma, \varphi \Rightarrow \psi, \Delta}{\Theta;\sigma \mid \Gamma \Rightarrow \varphi \to \psi, \Delta} \qquad \wedge\mathrm{L}\frac{\Theta;\sigma \mid \Gamma, \varphi, \psi \Rightarrow \Delta}{\Theta;\sigma \mid \Gamma, \varphi \wedge \psi \Rightarrow \Delta}$$

$$\wedge\mathrm{R}\frac{\Theta';\sigma' \mid \Gamma \Rightarrow \varphi, \Delta \quad \Theta';\sigma' \mid \Gamma \Rightarrow \psi, \Delta}{\Theta;\sigma \mid \Gamma \Rightarrow \varphi \wedge \psi, \Delta} \qquad \vee\mathrm{L}\frac{\Theta';\sigma' \mid \Gamma, \varphi \Rightarrow \Delta \quad \Theta';\sigma' \mid \Gamma, \psi \Rightarrow \Delta}{\Theta;\sigma \mid \Gamma, \varphi \vee \psi \Rightarrow \Delta}$$

$$\vee\mathrm{R}\frac{\Theta;\sigma \mid \Gamma \Rightarrow \varphi, \psi, \Delta}{\Theta;\sigma \mid \Gamma \Rightarrow \varphi \vee \psi, \Delta} \qquad \forall\mathrm{L}\frac{\Theta;\sigma \mid \Gamma, \varphi[t/x] \Rightarrow \Delta}{\Theta;\sigma \mid \Gamma, \forall x.\varphi \Rightarrow \Delta}$$

$$\forall\mathrm{R}\frac{\Theta;\sigma \mid \Gamma \Rightarrow \varphi, \Delta \quad x \notin \mathrm{FV}(\Gamma, \Delta)}{\Theta;\sigma \mid \Gamma \Rightarrow \forall x.\varphi, \Delta} \qquad \exists\mathrm{L}\frac{\Theta;\sigma \mid \Gamma, \varphi \Rightarrow \Delta \quad x \notin \mathrm{FV}(\Gamma, \Delta)}{\Theta;\sigma \mid \Gamma, \exists x.\varphi \Rightarrow \Delta}$$

$$\exists\mathrm{R}\frac{\Theta';\sigma' \mid \Gamma \Rightarrow \varphi[t/x], \Delta}{\Theta;\sigma \mid \Gamma \Rightarrow \exists x.\varphi, \Delta} \qquad \bot\mathrm{L}\frac{}{\Theta;\sigma \mid \Gamma, \bot \Rightarrow \Delta}$$

$$=\mathrm{L}\frac{\Theta;\sigma[t/x, s/y] \mid \Gamma[t/x, s/y] \Rightarrow \Delta[t/x, s/y] \quad x, y \notin \mathrm{FV}(s, t)}{\Theta;\sigma[s/x, t/y] \mid \Gamma[s/x, t/y], s = t \Rightarrow \Delta[s/x, t/y]} \qquad =\mathrm{R}\frac{}{\Theta;\sigma \mid \Gamma \Rightarrow t = t, \Delta}$$

$$\textsc{Wk}\frac{\Theta';\sigma \mid \Gamma \Rightarrow \Delta}{\Theta;\sigma, \sigma^* \mid \Gamma, \Gamma^* \Rightarrow \Delta, \Delta^*} \qquad \textsc{Cut}\frac{\Theta;\sigma \mid \Gamma, \varphi \Rightarrow \Delta \quad \Theta;\sigma \mid \Gamma \Rightarrow \varphi, \Delta}{\Theta;\sigma \mid \Gamma \Rightarrow \Delta}$$

$$\textsc{Sub}\frac{\Theta;\sigma[s/x] \mid \Gamma \Rightarrow \Delta}{\Theta;\sigma \mid \Gamma[s/x] \Rightarrow \Delta[s/x]} \qquad S\frac{\Theta;\sigma \mid \Gamma, t = Sx \Rightarrow \Delta \quad x \text{ fresh}}{\Theta;\sigma \mid \Gamma, 0 < t \Rightarrow \Delta}$$

RA 的公理是下文列出的循环算术算术公理。这意味着 $\Theta;\sigma \mid \Gamma \Rightarrow \Delta$ 对于下文的任意循环算术序列式 $\Gamma \Rightarrow \Delta$ 以及任意控制 $\Theta;\sigma$，它都是*RA*的一条公理。

$$\begin{array}{lll}
s < t, t < u \Rightarrow s < u & s < t \Rightarrow Ss < St & \Rightarrow s + St = \mathrm{S}(s + t) \\
s < t, t < s \Rightarrow & \Rightarrow s < t, s = t, t < s & \Rightarrow t \cdot 0 = 0 \\
s < t, t < Ss \Rightarrow & \Rightarrow t < St & \Rightarrow s \cdot St = (s \cdot t) + s \\
t < 0 \Rightarrow & \Rightarrow t + 0 = t &
\end{array}$$

最后，*RA* 包含三条推导规则，而循环算术中没有与之对应的规则。

$$\textsc{Focus}\frac{\Theta;\sigma,(t\mapsto\varepsilon)\mid\Gamma\Rightarrow\Delta\quad t\in\textsc{Term}(\Gamma,\Delta)}{\Theta;\sigma\mid\Gamma\Rightarrow\Delta}$$

$$\textsc{Reset}_a\frac{\Theta';\sigma,(t_1\mapsto ua),\dots,(t_n\mapsto ua)\mid\Gamma\Rightarrow\Delta\quad a\text{ does not occur in }\sigma}{\Theta;\sigma,(t_1\mapsto uau_1),\dots,(t_n\mapsto uau_n)\mid\Gamma\Rightarrow\Delta}$$

$$<\text{L}\frac{\Theta a;\sigma,(s\mapsto ua)\mid\Gamma\Rightarrow\Delta\quad a\text{ fresh}}{\Theta;\sigma,(t\mapsto u)\mid\Gamma,s<t\Rightarrow\Delta}$$

一个*RA*-预证明满足*RA* 的可靠性条件，如果每一对芽 $t\in$ 定义域(β)和伴随节点 $\beta(t)$都具有一个不变量：存在一个字母 a，使得 a出现在 t与 $\beta(t)$之间的所有控制中，a的前缀在这些控制中保持不变，并且重置$_a$规则在 t与 $\beta(t)$之间应用。若一个*RA*-证明是*CA*-序列的证明 $\Gamma\Rightarrow\Delta$，则其根节点标记为 ε；$\emptyset\mid\Gamma\Rightarrow\Delta$。如果存在 $\Gamma\Rightarrow\Delta$ 的*RA* 证明，则记作$RA\vdash\Gamma\Rightarrow\Delta$。」

证明系统*RA* 有一项“人体工学调整”，使其区别于“naïve”重置系统 R(*CA*)。在 R(*CA*) 中，与*CA*-规则对应的规则会在取得进展时向控制中添加新筹码（Θ,σ），也就是说，当假设 Γ 中存在不等式 $s<t$ 时就会添加。这很快就会变得难以控制，使控制的处理相当棘手。在*RA*中，*CA*对应规则绝不会向控制中添加筹码，只会在筹码不再使用时将其移除。相反，<L 规则允许证明者添加与 Γ中某个不等式 $s<t$ 所体现的进展相对应的筹码。

我们通过构造证明态射$embed\colon RA\ \to$R(循环算术)，证明*RA* 相对于*PA* 的可靠性。回顾一下，R(循环算术)是重置证明系统

引理6.4 存在一个函数$embed\colon \mathsf{Seq}_{RA}\to\mathsf{R}(\mathsf{Seq}_{CA})$，其定义如下：

$$embed(\Theta;\sigma\mid\Gamma\Rightarrow\Delta):=\Gamma\Rightarrow\Delta;(\overline{\Theta},\overline{\sigma})$$

其中，$\overline{\Theta}$是集合$\{a\mid a\in\Theta\}$，其顺序按照Θ中字母的位置排列，且$\sigma(t)\ :=\{\{x\mid x\overline{\in}u\}\mid t\mapsto u\in\sigma\}$。它可以扩展为证明态射$embed\colon RA\to$R(循环算术)。

证明 为了翻译与 CA规则 对应的规则，我们需要考虑 RA 与 R(CA) 对上文所述 $\Rightarrow$ 左侧不等式的处理方式差异。选取一条 $R\in$ RA 规则，它有对应的 CA规则（即 R 不是 <L、Wk、Focus 或 Reset 的实例）。它被翻译为如下的 R(CA) 预证明。这里，我们用 u_i 表示由 R σ_i'“错误地”添加的筹码，并用 σ_i' 表示“错误的”栈。为此，我们采用记号 $\sigma\cup\sigma'$ 来表示函数 $(x,a)\mapsto\sigma(x,a)\cup\sigma'(x,a)$。注意，对于每个 $s<t\in\Gamma_i$，决定 Safra 棋盘转移 $(\Theta,\sigma)\overset{r_i}{\rightsquigarrow}(\Theta_i\oplus u_i,\overline{\sigma_i}\,\overline{\cup}\,\overline{\sigma_i'})$ 的迹映射 r_i 包含两个涉及 t 的前驱 $t'\leftarrow_R^i t$ 的转移：$(t,0,t')$ 和 $(t,1,s)$。后者导致了筹码被“错误地”添加。

前者确保 t 上的栈不会从 σ_i 中移除，而只是被“重新分配”给 t'，正如 *RA* 中所做的那样。这保证了 σ_i 属于每个前提中的控制。

$$\mathrm{R}\frac{\Theta_1;\sigma_1 \mid \Gamma_1 \Rightarrow \Delta_1 \quad \cdots \quad \Theta_1;\sigma_1 \mid \Gamma_1 \Rightarrow \Delta_1}{\Theta;\sigma \mid \Gamma \Rightarrow \Delta} \quad \overset{embed}{\rightsquigarrow}$$

$$\mathrm{R}\frac{\mathrm{Weak}\dfrac{\Gamma_1 \Rightarrow \Delta_1;(\overline{\Theta_1},\overline{\sigma_1})}{\Gamma_1 \Rightarrow \Delta_1;(\overline{\Theta_1}\oplus u_1,\overline{\sigma_1}\cup\sigma_1')} \quad \cdots \quad \mathrm{Weak}\dfrac{\Gamma_n \Rightarrow \Delta_n;(\overline{\Theta_n},\overline{\sigma_n})}{\Gamma_n \Rightarrow \Delta_n;(\overline{\Theta_n}\oplus u_n,\overline{\sigma_n}\cup\sigma_n')}}{\Gamma \Rightarrow \Delta;(\overline{\Theta},\overline{\sigma})}$$

重置规则和聚焦规则分别对应于 R(循环算术) 中重置规则和 Pop 规则的应用。Wk 被转换为循环算术 的 Wk-规则与 R(循环算术) 的 Weak-规则的组合：前者用于在 $\Gamma \Rightarrow \Delta$ 中进行“弱化”（如上所述，处理“错误”的芯片），后者用于在 σ 中进行“弱化”。通过 *embed* 进行转换时，唯一仍未解决的规则是 <L。这一转换通过应用（可能是空应用的）循环算术中的 Wk-规则来实现，以“模拟”移除不等式 $s < t$；随后应用 R(循环算术) 中的 Weak-规则，移除所有“错误”添加的芯片和栈，即除不等式 $s < t$所诱导的那个之外的全部芯片和栈。

$$<\mathrm{L}\frac{\Theta a;\sigma,(s \mapsto ua) \mid \Gamma \Rightarrow \Delta \quad a \text{ fresh}}{\Theta;\sigma,(t \mapsto u) \mid \Gamma, s < t \Rightarrow \Delta} \quad \overset{embed}{\rightsquigarrow}$$

$$\mathrm{Wk}\frac{\mathrm{Weak}\dfrac{\Gamma \Rightarrow \Delta;(\overline{\Theta a},\overline{\sigma,(s,0) \mapsto ua})}{\Gamma \Rightarrow \Delta;(\overline{\Theta a v},\overline{\sigma,(t \mapsto u)}\cup\overline{(s,0) \mapsto ua}\cup\sigma')}}{\Gamma, s < t \Rightarrow \Delta;(\overline{\Theta},\overline{\sigma,(t \mapsto u)})}$$

通过比较*RA* 和 R(循环算术) 的可靠性条件，可以很容易地观察到，*embed* 保持了*RA*的可靠性条件。■

推论6.5（可靠性） 若 $RA \vdash \Gamma \Rightarrow \Delta$ 则 $PA \vdash \Gamma \Rightarrow \Delta$。

证明 若 Π是 *RA* 中的一个证明，证明 $\varepsilon;\ \emptyset \mid \Gamma \Rightarrow \Delta$，那么 $embed(\Pi)$ 就是 R(循环算术) 中 $\Gamma \Rightarrow \Delta;(\emptyset, ((x,a) \mapsto \emptyset))$ 的一个证明，因此 $strip(embed(\Pi))$ 是 循环算术中 $\Gamma \Rightarrow \Delta$ 的一个证明。由于 循环算术 证明的序列式与 *PA* 相同（见 [22, 定理 6]），因此 *PA*中也必定存在 $\Gamma \Rightarrow \Delta$ 的一个证明。■

设 F 是循环算术的一个有限片段。为了得出*RA* 相对于*PA*的完备性，我们构造一个证明态射*search*: $\mathsf{S}(\mathsf{F}) \to RA$。为此，回顾一下，S(F) 是由 F 中的推导规则以及循环算术的轨迹解释所诱导的“证明搜索系统”。

引理6.6 存在一个函数*search*：$\mathsf{S}(\mathsf{Seq}_{\mathcal{F}}) \to \mathsf{Seq}_{RA}$，满足

$$search(\Gamma \Rightarrow \Delta;(\Theta,\sigma)) := \widehat{\Theta};\widehat{\sigma} \vdash \Gamma \Rightarrow \Delta$$

b其中 $S \in \Theta^*$ 对于 $S \subseteq \Theta$，是长度为 $|\Theta|$ 的无重复序列，且按照 Θ 严格排序；如果 $(\Theta,\ \sigma)$是 X上的一个 K-稀疏 Safra 棋盘，那么

$$\widehat{\sigma} \coloneqq \{s \mapsto \widehat{S} \mid x \in X, s \in \textsc{Term}(\Gamma, \Delta) \text{ and } \sigma(s, 0) = \{S\}\}$$

该函数可以扩展为证明态射 *search*: $\mathsf{S}(\mathsf{F}) \to RA$。

证明 为证明这一主张，首先选取某个 $\mathrm{R}(\Theta, \sigma) \in \mathsf{S}(\mathsf{F})$，其排列如下

$$\mathrm{R}(\Theta,\sigma)\frac{\Gamma_1 \Rightarrow \Delta_1; (\Theta_1, \sigma_1) \quad \dots \quad \Gamma_n \Rightarrow \Delta_n; (\Theta_n, \sigma_n)}{\Gamma \Rightarrow \Delta; (\Theta, \sigma)}$$

那么存在 $R \in \mathrm{F}$，使得 $\rho(R) = (\Gamma \Rightarrow \Delta, \Gamma_1 \Rightarrow \Delta_1, \dots, \Gamma_n \Rightarrow \Delta_n)$，且由轨迹解释给出态射 $r_i\colon \iota(\Gamma \Rightarrow \Delta) \to \iota(\Gamma_i \Rightarrow \Delta_i)$。于是对每个 $i \leq n$，存在 $(\Theta, \sigma) \overset{r_i}{\rightsquigarrow}_g (\Theta_i, \sigma_i)$，其中展开后的序列是展开后的序列

$$(\Theta, \sigma) \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_r^1, \sigma_r^1) \dots \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_r^k, \sigma_r^k) \overset{P}{\rightsquigarrow} (\Theta_p, \sigma_p) \overset{r_i}{\rightsquigarrow} (\Theta_i^*, \sigma_i^*) \overset{T}{\rightsquigarrow} (\Theta_i, \sigma_i)$$

其中初始的 R_γ-步和 P-步由所有 $i \leq n$共享（见引理 4.18）。与引理 5.8 类似，我们可以在 RA中推导出以下内容：

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\widehat{\Theta_1}; \widehat{\sigma_1} \mid \Gamma_1 \Rightarrow \Delta_1}{\widehat{\Theta_1'}; \widehat{\sigma_1'} \mid \Gamma_1 \Rightarrow \Delta_1}\textsc{Wk} \quad \dfrac{\widehat{\Theta_n}; \widehat{\sigma_n} \mid \Gamma_n \Rightarrow \Delta_n}{\widehat{\Theta_n'}; \widehat{\sigma_n'} \mid \Gamma_n \Rightarrow \Delta_n}\textsc{Wk}}{\vdots}\ {<}\mathrm{L} \quad \dfrac{\vdots}{\widehat{\Theta_p}; \widehat{\sigma_p} \mid \Gamma_1 \Rightarrow \Delta_1} \quad \dots \quad \dfrac{\vdots}{\widehat{\Theta_p}; \widehat{\sigma_p} \mid \Gamma_n \Rightarrow \Delta_n}\ {<}\mathrm{L}}{\widehat{\Theta_p}; \widehat{\sigma_p} \mid \Gamma \Rightarrow \Delta}\mathrm{R}}{\vdots}}{\widehat{\Theta_r^k}; \widehat{\sigma_r^k} \mid \Gamma \Rightarrow \Delta}\textsc{Focus}}{\vdots}}{\widehat{\Theta_0}; \widehat{\sigma_0} \mid \Gamma \Rightarrow \Delta}\textsc{Reset}_a$$

也就是说，首先从 Θ_0 最大的项开始，应用所有可能的重置$_a$-规则。然后对每个没有 $t \mapsto u \in \sigma_r^k$ 的 $ct \in$ 项 $(\Gamma \Rightarrow \Delta)$ 应用 Focus。应用与 $R \in \mathrm{F}$ 相对应的 RA- 规则后，按照下一段所述，谨慎地应用 <L 的各种实例。通过应用 Wk 关闭预证明的每个分支，并从 bσ_i 中移除所有多余的注释。

应用 <L-实例需要多加考虑：若 $s < t$，$s < t' \in \Gamma_i$且 $t \neq t'$，那么 $\sigma_i^*(s,\ 0)$ 将包含标注 $\sigma_P(t,\ 0)\,\sigma_P(t',\ 0)$，并由同一个$\gamma \in \Theta_i^*$扩展。另一方面，在 RA 中，t和 t' 的标注只能通过分别应用 <L-规则来扩展，这意味着标注将分别由两个不同的筹码γ，γ'扩展。不过请注意，经过稀疏化步骤 $(\Theta_i^*,\ \sigma_i^*)$ 后，$\overset{T}{\rightsquigarrow}(\Theta_i,\ \sigma_i)$ 中只有一个标注保留在 $\sigma_1(s,\ 0)$ 中。因此，上图所示的预证明只应用了 <L-实例，从而得到 s的这个“存续”标注，并带有“正确”的筹码 γ。如上方的预证明所示，这导致

序列式 Θ_1' 中的 c b c b；$\sigma_1' \mid \Gamma_1 \Rightarrow \Delta_1$ 而不是“天真”的序列式 Θ_1^*；$\sigma_1^* \mid \Gamma_1 \Rightarrow \Delta_1$。然而，应用 Wk 后，叶节点处便会得到所需的序列式。

仿照引理 5.8 中对 *expand* 所作论证，可知 *search* 保持可靠性条件。■

推论 6.7（完备性） 若 $PA \vdash \Gamma \Rightarrow \Delta$ 则 $RA \vdash \Gamma \Rightarrow \Delta$。

证明 假设在*PA*中存在一个关于 $\Gamma \Rightarrow \Delta$ 的证明。根据 [22, 定理 6]，同一序列式在*CA*中也有证明。事实上，由于循环证明是有限树，该证明是在*CA*的某个有限片段 F 中构造的。由定理 5.9，存在一个关于 $\Gamma \Rightarrow \Delta$的证明 Π；$(\emptyset,\ (x,\ a) \mapsto \emptyset)$，它属于 S(F)，因此*search*(Π) 是*RA*中关于 ε；$\emptyset \mid \Gamma \Rightarrow \Delta$ 的证明。■

6.2 G¨odel 的 T

G¨odel 的 T [10] 是对简单类型 λ演算的一种扩展，包含自然数类型 N 以及类型任意的原始递归。循环 G¨odel 的 T 是 G¨odel 的 T 的循环变体，最早由 Das [9]提出。这意味着 G¨odel 的 T 作为推导系统，其导出的不是“证明”，而是内禀类型项（见 [19, 第 15]章）。相应地，循环 G¨odel 的 T（*CGT*）的预证明是内禀类型余项，而证明则是满足某种良定义条件的此类余项。尽管如此，为与文章其他部分的术语保持一致，我们仍分别称它们为*CGT*的预证明和证明。

（重置）G¨odel 的 T 示例展示了本文所提方法的“理想路径”：只需对 R(*CGT*) 做少许调整，即可得到重置系统 *RGT*。因此，只需稍加努力，便可证明*RGT* 与 *CGT* 之间的可靠性和完备性结果。

6.2.1 循环 G¨odel 的 T

G¨odel 的 T 的类型 由以下文法给出

$$A \in \textsc{Type} ::= N \mid A \to A$$

CGT 的序列式 是表达式 $\Gamma \Rightarrow A$，其中 Γ 是有限类型序列。G¨odel 的 T 中序列式的集合记为 Seq_{CGT}。

定义 6.8 循环 G¨odel 的 T *CGT* 的推导规则 如下：

$$\textsc{Id}\frac{}{A \Rightarrow A} \qquad 0\frac{}{\Rightarrow N} \qquad S\frac{}{N \Rightarrow N} \qquad L\frac{\Gamma \Rightarrow \rho \quad \Gamma, A \Rightarrow B}{\Gamma, \rho \to A \Rightarrow B}$$

$$R\frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow A \to B} \qquad \textsc{Cond}\frac{\Gamma \Rightarrow A \quad \Gamma, N \Rightarrow A}{\Gamma, N \Rightarrow A} \qquad \textsc{Ex}\frac{\Gamma_0, A, B, \Gamma_1 \Rightarrow C}{\Gamma_0, B, A, \Gamma_1 \Rightarrow C}$$

$$\textsc{Wk}\frac{\Gamma \Rightarrow B}{\Gamma, A \Rightarrow B} \qquad \textsc{Ctr}\frac{\Gamma, A, A \Rightarrow B}{\Gamma, A \Rightarrow B} \qquad \textsc{Cut}\frac{\Gamma \Rightarrow B \quad \Gamma, B \Rightarrow A}{\Gamma \Rightarrow A}$$

将*CGT* 呈现为序列演算，打破了以自然演绎风格呈现类型推导（或等价地，内禀类型项）的传统。请注意，*CGT* 的无假设非循环预证明也可以使用以下原始递归规则

$$\textsc{Rec}\frac{\Gamma \Rightarrow A \quad \Gamma, A, \Rightarrow A}{\Gamma, N \Rightarrow A}$$

对应的正是 G¨odel 的 T 中内禀类型项（以序列式风格呈现）。可以证明，*CGT* 的证明定义的泛函与普通 G¨odel 的 T 定义的泛函完全相同（见 [9, 定理 94]）。

接下来给出 *CGT* 的可靠性条件。给定一个类型序列 Γ，定义 $|\Gamma|_N$ 为基类型 N 在 Γ 中的出现次数:

$$|\Gamma|_N \coloneqq \begin{cases} 0 & \Gamma = \varepsilon \\ 1 + |\Gamma'| & \Gamma = N, \Gamma' \\ |\Gamma'| & \Gamma = A, \Gamma' \text{ with } \tau \neq N \end{cases}$$

定义 6.9 迹解释 $\iota\colon CGT \to \mathbb{T}_{\mathbb{B}}$ 定义为 $\iota(\Gamma \Rightarrow \tau) \coloneqq \{1, \ldots, |\Gamma|_N\}$；对于任意满足 $\rho(R) = (\Gamma \Rightarrow \tau, \Gamma_1 \Rightarrow \tau_1, \ldots, \Gamma_n \Rightarrow \tau_n)$ 的 $R \in \mathsf{R}$，迹映射 $r_i\colon \iota(\Gamma \Rightarrow \tau) \to \iota(\Gamma_i \Rightarrow \tau_i)$ 定义如下:

- 若 R 是 Ex 的一个实例，它交换 N 的两个实例，即 $\Gamma = \Gamma_0, N, N, \Gamma_1$，则

$$r_i \coloneqq \{(j, 0, j) \mid j \leq k \text{ or } k + 2 < j\} \cup \{(k + 1, 0, k + 2), (k + 2, 0, k + 1)\}$$

  其中 $k \coloneqq |\Gamma_0|_N$。
- 若 $\Gamma = \Gamma'$，N 且 R 是 Wk 的一个实例，或 R 是 Cond 的一个实例，且 $i = 1$，则

$$r_i \coloneqq \{(j, 0, j) \mid j < |\Gamma|_N\}.$$

- 若 R 是 Cond 的一个实例，且 $i = 2$，则

$$r_i \coloneqq \{(j, 0, j) \mid j < k\} \cup \{(k, 1, k)\}$$

  其中 $k \coloneqq |\Gamma|_N$。
- 若 R 是 Ctr 的一个实例，则

$$r_i \coloneqq \{(j, 0, j) \mid j < k\} \cup \{(k - 1, 0, k)\}$$

  其中 $k \coloneqq |\Gamma|_N$。
- 否则 $r_i \coloneqq \{(j, 0, j) \mid j \leq |\Gamma|_N\}$.

这引出了定义 3.6 中所述的循环 G¨odel's T 的可靠性条件 C。如果$\Gamma \Rightarrow \tau$在 *CGT* 中可证明，则记作 $CGT \vdash \Gamma \Rightarrow \tau$。⌟

6.2.2 重置 G¨odel 的 T

我们引入了一个与 *CGT* 对应的基于重置的证明系统，称为**重置 *G¨odel's T***（*RGT*）。它基于由 *CGT* 诱导的重置系统 R(*CGT*)，并进行了少量修改。

***RGT* 的序列式** 是表达式 $\Theta \mid \Gamma \Rightarrow A$，其中**控制** Θ 是由互不相同的字符组成的序列，A 是未标注的类型，而**上下文** Γ 是一个类型列表 B^u，并标注有子序列 $u \sqsubseteq \Theta$；仅当 $B = N$ 时，该子序列才可以非空。*RGT*序列式的集合记作 Seq_{RGT}。

定义 6.10 重置 G¨odel 的 T 的推导规则如下。用 Θ' 表示从中移除了上下文中任何标注里都未出现的所有字母后的控制。

$$\textsc{Id}\frac{}{\Theta \mid A^u \Rightarrow A} \qquad 0\frac{}{u \mid \Rightarrow N} \qquad S\frac{}{\Theta \mid N^u \Rightarrow N} \qquad L\frac{\Theta' \mid \Gamma \Rightarrow A \quad \Theta \mid \Gamma, B^u \Rightarrow C}{\Theta \mid \Gamma, (A \to B)^u \Rightarrow C}$$

$$R\frac{\Theta \mid \Gamma, A^\varepsilon \Rightarrow B}{\Theta \mid \Gamma \Rightarrow A \to B} \qquad \textsc{Cond}\frac{a \text{ fresh in } \Theta \quad \Theta' \mid \Gamma \Rightarrow A \quad \Theta a \mid \Gamma, N^{ua} \Rightarrow A}{\Theta \mid \Gamma, N^u \Rightarrow A}$$

$$\textsc{Ex}\frac{\Theta \mid \Gamma, A^v, B^u, \Theta \mid \Gamma' \Rightarrow C}{\Theta \mid \Gamma, B^u, A^v, \Theta \mid \Gamma' \Rightarrow C} \qquad \textsc{Wk}\frac{\Theta' \mid \Gamma \Rightarrow B}{\Theta \mid \Gamma, A^u \Rightarrow B} \qquad \textsc{Ctr}\frac{\Theta \mid \Gamma, A^u, A^u \Rightarrow B}{\Theta \mid \Gamma, A^u \Rightarrow B}$$

$$\textsc{Cut}\frac{\Theta \mid \Gamma \Rightarrow B \quad \Theta \mid \Gamma, B^\varepsilon \Rightarrow A}{\Theta \mid \Gamma \Rightarrow A}$$

$$\textsc{Reset}_a\frac{\Theta' \mid \Gamma, N^{ua}, \ldots, N^{ua} \Rightarrow B \quad a \text{ does not occur in } \Gamma}{\Theta \mid \Gamma, N^{uau_1}, \ldots, N^{uau_n} \Rightarrow B}$$

一个*RGT*-预证明是一个**证明**，其中每一对芽 $t \in \mathrm{dom}(\beta)$ 和伴随节点 $\beta(t)$ 都满足不变量：存在一个字母 a，使得 a 出现在 t 与 $\beta(t)$ 之间的所有控制 Θ 中，a 在控制 Θ 中的前缀保持不变，并且重置$_a$规则在 t 与 $\beta(t)$ 之间应用。若证明Π 的根节点标记为 $\varepsilon \mid \Gamma^\varepsilon \Rightarrow A$，则称其为 $\Gamma \Rightarrow A$ 的**证明**。若存在 $\varepsilon \mid \Gamma^\varepsilon \Rightarrow A$ 在*RGT* 中的证明，则记作 $RGT \vdash \Gamma \Rightarrow A$。⌟

系统 *RGT* 与 R(*CGT*) 的一个区别在于：上下文中的每个类型始终都带有标注。在 R(*CGT*) 中，则需要通过 Pop 规则手动添加标注。此外，为了让系统的定义稍微简单一些，*RGT* 在 Γ 中的每个类型上都使用标注，而不只是对 N 的实例使用标注。不过，在这两个系统中，只有类型 N 的实例才可能具有非空标注。

我们通过构造证明态射 *RGT* 相对于 *CGT* 的可靠性，来证明 *embed*: $RGT \to$R(*CGT*)。

引理 6.11 存在一个函数 $embed\colon \mathsf{Seq}_{RGT} \to \mathrm{R}(\mathsf{Seq}_{CGT})$，定义如下：

$$embed(\Theta \mid \Gamma \Rightarrow A) \coloneqq \overline{\Gamma} \Rightarrow A; (\overline{\Theta}, \sigma(i) \mapsto \{\{x \text{ occurs in } u\} \mid \Gamma@i = N^u\})$$

其中，$\overline{\Theta}$ 是按字母在 Θ中的位置排序的{u 在 Θ中出现}集合，Γ 是从类型中移除所有标注后的 Γ，而 $\Gamma@i$ 是递归定义的偏运算，定义如下：

$$\Gamma, N^u@0 \coloneqq N^u \qquad \Gamma, N^u@i+1 \coloneqq \Gamma@i \qquad \Gamma, A^u@i \coloneqq \Gamma@i \text{ (where } A \neq N)$$

该函数可扩展为一个证明态射$embed\colon RGT \to \mathrm{R}(CGT)$。

证明 重置 G¨odel 的 T 中的大多数规则，只需转换为循环 G¨odel 的 T 所对应的诱导重置系统中的规则。唯一的例外是带有 $B = N$的规则 R 和 Cut，它们会在 R(CGT) 中生成“无注释实例”。这些情况可通过额外应用 Pop 来处理，如下方 R 的情况所示：

$$\mathrm{R}\frac{\Theta \mid \Gamma, N^{\varepsilon} \Rightarrow A}{\Theta \mid \Gamma \Rightarrow N \to A} \quad \overset{embed}{\rightsquigarrow} \quad \mathrm{R}\,\frac{\mathrm{Pop}\,\dfrac{\Gamma, N \Rightarrow A; (\overline{\Theta}, \overline{\sigma} \cup \{(k+1, \{\emptyset\})\})}{\Gamma, N \Rightarrow A; (\overline{\Theta}, \overline{\sigma} \cup \{(k+1, \emptyset)\})}}{\Gamma \Rightarrow N \to A; (\overline{\Theta}, \overline{\sigma})}$$

通过比较RGT 和 R(CGT)的可靠性条件，可以很容易地看出，$embed$保留了RGT的可靠性条件。

■

请注意，在下面的可靠性定理中，重要的是，与RGT 证明相对应的CGT 证明不仅推导出相同的序列式 $\Gamma \Rightarrow A$，而且还具有相同的计算内容。很容易看出，$strip$ 和$embed$ 都不会改变相关证明的计算内容。

推论 6.12（可靠性）如果 $RGT \vdash \Gamma \Rightarrow A$ 通过 Π 那么 $CGT \vdash \Gamma \Rightarrow A$ 通过$strip$（$embed$（Π）。

设 F 是 CGT 的有限片段。为了得出相对于 CGT 的 RGT 完备性，我们构造一个证明态射 $search$：$\mathrm{S}(\mathrm{F}) \to CGT$。为此，回顾一下，S(F) 是由 F 中的推导规则以及 CGT 的轨迹解释所诱导的“证明搜索系统”。

引理 6.13 函数 $search\colon \mathrm{S}(\mathsf{Seq}_{\mathcal{F}}) \to \mathsf{Seq}_{RGT}$ 定义为

$$search(\Gamma \Rightarrow A; (\Theta, \sigma)) \coloneqq \widehat{\Theta} \mid \Gamma^{\sigma} \Rightarrow A$$

b，其中对于任意 $S \subseteq \Theta, S \in \Theta^*$ 是长度为 $|S|$的无重复序列，由 S 的元素组成，并按照 Θ 严格排序。回想一下，(Θ, σ) 是 K-稀疏的，记号 $\Gamma^{\sigma} \coloneqq \Gamma_1^{\sigma}$ 递归定义为

$$\Gamma_i^{\sigma} \coloneqq \begin{cases} A^{\varepsilon}, \Gamma_i'^{\sigma} & \text{if } \Gamma = A, \Gamma' \text{ with } A \neq N \\ N^{\hat{S}}, \Gamma_{i+1}'^{\sigma} & \text{if } \Gamma = N, \Gamma' \text{ with } \sigma(i) = \{S\} \\ N^{\varepsilon}, \Gamma_i'^{\sigma} & \text{if } \Gamma = N, \Gamma' \text{ with } \sigma(i) = \emptyset \\ \varepsilon & \text{if } \Gamma = \varepsilon \end{cases}$$

该函数可以扩展为一个证明态射 $search\colon \mathrm{S}(\mathrm{F}) \to RGT$.

证明 为了得到这一结论，首先选取一个按如下方式排列的 $\mathrm{R}(\Theta,\sigma) \in \mathrm{S}(\mathrm{F})$

$$\mathrm{R}(\Theta,\sigma)\frac{\Gamma_1 \Rightarrow B_1;(\Theta_1,\sigma_1) \quad \ldots \quad \Gamma_n \Rightarrow B_n;(\Theta_n,\sigma_n)}{\Gamma \Rightarrow A;(\Theta,\sigma)}$$

那么，存在 $R \in \mathrm{F}$，其中 $\rho(R) = (\Gamma \Rightarrow A,\ \Gamma_1 \Rightarrow B_1,\ \ldots\ldots,\ \Gamma_n \Rightarrow B_n)$，并且态射 $r_i\colon \iota(\Gamma \Rightarrow A) \to \iota(\Gamma_i \Rightarrow B_i)$ 由轨迹解释给出。于是，对于每个 $i \le n$，存在 $(\Theta,\ \sigma) \overset{r_i}{\rightsquigarrow}_g (\Theta_i,\ \sigma_i)$，其中扩展后的序列即为扩展后的序列

$$(\Theta,\sigma) \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_r^1,\sigma_r^1) \ldots \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_r^k,\sigma_r^k) \overset{P}{\rightsquigarrow} (\Theta_p,\sigma_p) \overset{r_i}{\rightsquigarrow} (\Theta_i,\sigma_i) \overset{T}{\rightsquigarrow} (\Theta_i,\sigma_i)$$

其中初始的 R_γ-步骤和 P-步骤由所有 $i \le n$ 共享（见引理 4.18）。请注意，由于循环 G¨odel's T 的轨迹解释具有特定结构，扩展后的序列中 Safra 棋盘的同一字段上绝不会出现两个栈，这意味着稀疏化不会改变 Θ_i 和 σ_i。与引理 5.8 类似，我们可以在重置 G¨odel's T 中推导出以下结论：

$$\mathrm{R}\,\frac{\widehat{\Theta_1} \mid \Gamma_1^{\sigma_1} \Rightarrow B_1 \qquad \ldots \qquad \widehat{\Theta_n} \mid \Gamma_n^{\sigma_n} \Rightarrow B_n}{\begin{array}{c}\widehat{\Theta_r^k} \mid \Gamma^{\sigma_r^k} \Rightarrow A \\ \vdots \\ \widehat{\Theta_0} \mid \Gamma^{\sigma_0} \Rightarrow A\end{array}}\,\textsc{Reset}_a$$

也就是说，首先从 Θ_0-最大者开始，应用所有可能的 Reset_a-规则。由于 Γ^σ 用 N^ε 标注了 N 的实例，而 σ 并未为这些实例“分配”栈，因此无需在预证明中将总体化步骤复制为 $\Gamma^{\sigma_r^k} = \Gamma^{\sigma_P}$。应用与 $R \in \mathrm{F}$ 对应的规则，完成预证明。请注意，重置 G¨odel's T 的规则同样会用 N^ε 标注 N 的实例，而 σ_i 并未为这些实例分配栈，这意味着前提确实是 $\mathrm{b}\ \Theta_i \vdash \Gamma_i^{\sigma_i} \Rightarrow B_i$。由于稀疏化转移不会改变 σ_i，因此无需在预证明中复制该转移。

与引理 5.8 中针对 *expand* 所给出的论证类似，可以证明 *search* 保持可靠性条件。■

与可靠性类似，对于完备性而言，证明的计算内容保持不变同样至关重要。

定理 6.14（完备性） 若 $CGT \vdash \Gamma \Rightarrow A$ 通过 Π，则 $RGT \vdash \Gamma \Rightarrow A$ 通过证明 Π'，使得 $strip(embed(\Pi'))$ 是 Π 的一个展开。

6.3 模态 μ-演算

模态 μ-演算（$K\mu$）扩展了经典模态逻辑 K，并加入不动点量词 $\nu x.\varphi$ 和 $\mu x.\varphi$，分别表示最大不动点和最小不动点。μ-演算一直是循环证明理论领域的核心：最早的循环证明系统是为 K_μ [16] 提出的，而 $K\mu$（以及诸如 μMALL 和高阶 μ-演算等变体）则是循环证明理论领域研究最多的对象。

本节将为模态 μ-演算构造两个重置证明系统，分别称为 $RK_\mu^{\mathbb{F}}$（第6节.3.2）和 $RK_\mu^{\mathbb{B}}$（第6节.3.3）。它们对应于对 $K\mu$ 的迹条件的两种不同而自然的表述：一种基于布尔值 $\mathbb{B}$，另一种基于失败代数 $\mathbb{F}$。这也表明，如果一个推导系统存在多种合理的轨迹解释，那么同一个推导系统可以导出多个截然不同的重置证明系统。

第一个重置证明系统由 Jungteerapanich 于 [12] 提出，用于 $K\mu$ 的可满足性；随后，Stirling [25] 将其改写为用于有效性的重置证明系统。后一系统在文献中通常称为 Jungteerapanich-Stirling（$\mathcal{JS}$）系统。在第6.3.4节中，我们回顾系统 $\mathcal{JS}$，并将其与系统 $RK_\mu^{\mathbb{F}}$ 和 $RK_\mu^{\mathbb{B}}$进行比较。这一比较凸显了我们方法可能最大的缺点：本文中的重置证明系统仅依据迹条件构造，而未深入考察相关逻辑的语义。

6.3.1 循环模态 μ-演算

我们对 $K\mu$ 的介绍基于 [3]中的介绍。给定一个命题字母集合 Prop 和一个可数变量集合 Var，μ-公式由以下文法给出：

$$\varphi \in \textsc{Form} ::= p \mid \neg p \mid x \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \Box\varphi \mid \Diamond\varphi \mid \mu x.\varphi \mid \nu x.\varphi \quad p \in \mathrm{Prop}, x \in \mathrm{Var}$$

若 $x, y \in$Var 出现在 φ中，且 x涵盖 y，记作 $x <_\varphi y$，当且仅当对某个 $\sigma \in \{\mu, \nu\}$ 和 ψ，$\sigma y.\psi$作为子公式出现在 φ中，并且 x在 $\sigma y.\psi$中自由出现。若关系 $<_\varphi$是严格预序，则称 φ 良命名。下文只考虑序列式 Γ，其中 Γ 是良命名公式组成的集合。

定义 6.15 $K\mu$ 的序列式是良好命名的 μ-公式的有限集 Γ。记 Γ, φ 表示 $\Gamma \cup \{\varphi\}$，记 Γ, Γ' 表示 $\Gamma \cup \Gamma'$。$K\mu$-序列式的集合记为 $\mathsf{Seq}^{K\mu}$。$K\mu$ 的推导规则如下：

$$\textsc{Ax}\frac{}{p, \neg p} \qquad \textsc{Wk}\frac{\Gamma}{\Gamma, \varphi} \qquad \vee\frac{\Gamma, \varphi, \psi}{\Gamma, \varphi \vee \psi} \qquad \wedge\frac{\Gamma, \varphi \qquad \Gamma, \psi}{\Gamma, \varphi \wedge \psi} \qquad \textsc{Mod}\frac{\Gamma, \varphi}{\Diamond\Gamma, \Box\varphi}$$

$$\mu\frac{\Gamma, \varphi[\mu x.\varphi/x]}{\Gamma, \mu x.\varphi} \qquad \nu\frac{\Gamma, \varphi[\nu x.\varphi/x]}{\Gamma, \nu x.\varphi}$$

在上述规则中，$\varphi[\psi/x]$ 表示将公式 φ 中变量 x 的所有实例替换为公式 ψ 后得到的公式。这是一个偏运算，仅当 x 未被 φ 中的某个不动点量词 νx 或 μx 约束时才有定义。写作 $\varphi[\psi/x]$，即默认该运算对这些参数有定义。

如前所述，模态 μ-演算 $K\mu$ 至少有两种自然的迹解释：一种基于布尔值$\mathbb{B}$，另一种基于失败代数 $\mathbb{F}$。我们用 $\iota_{\mathcal{A}}$ 来区分这两种迹解释，其中 A 是相关解释所定义其上的激活代数。解释 $\iota_{\mathbb{F}}$ 对应于全局轨迹条件的「通常」表述

：将 μ-演算的全局轨迹条件表述为蕴含层级 $<_\varphi$上的奇偶条件。事实上，这一全局轨迹条件正是激活代数 [6]这一概念最初的提出动机。

对于推导规则 $R \in K\mu$，其中 $\rho(R) = (\Gamma, \Delta_1, \ldots\ldots, \Delta_n)$，称 $\varphi' \in \Delta_i$为 $\varphi \in \Gamma$的前驱，记作 $\varphi' \leftarrow^i_R \varphi$，如果 φ是 R_i中的主公式，即 φ被 R_i“改变”，并且 φ'是由 φ经 R得到的公式出现之一；或者 φ不是 R_i中的主公式，且 $\varphi = \varphi'$。

定义 6.16 记 $V_\nu(\varphi) := \{x \mid x$在 φ中由 ν约束$\}$，轨迹解释$\iota_{\mathbb{F}}$: $K\mu \to \mathbb{T}_{\mathbb{F}}$ 定义如下

$$\iota_{\mathbb{F}}(\Gamma) := \{(\varphi, x) \mid \varphi \in \Gamma \text{ and } x \in V_\nu(\varphi)\}$$

并且对于任意 $R \in K\mu$，其中 $\rho(R) = (\Gamma, \Delta_1, \ldots\ldots, \Delta_n)$，轨迹映射 r_i: $\iota_{\mathbb{F}}(\Gamma) \to \iota_{\mathbb{F}}(\Delta)$ 定义为$r_i := \{((\varphi, x)\ a^*, (\varphi', x)) \mid \varphi' \leftarrow^i_R \varphi\}$，其中 a^*定义为

$$a^* := \begin{cases} 2, & \text{if } R_i \text{ instance of } \mu, \varphi = \mu y.\theta, \varphi' = \theta[\mu y.\theta/y] \text{ and } y <_\varphi x, \\ 1, & \text{if } R_i \text{ instance of } \nu, \varphi = \nu x.\theta, \varphi' = \theta[\nu x.\theta/x], \\ 0, & \text{otherwise.} \end{cases}$$

⌟

轨迹解释 $\iota_{\mathbb{B}}$ 对于$K\mu$ 的定义方式是：在序列式中“追踪”各个不动点量词实例，并找出一个被无穷次展开且从未被更高层量词的展开“抹除”的最大不动点量词。称序列 $a \in \mathbb{B}^*$ 为一个子公式地址，并定义部分寻址函数 $\varphi @ a$ 如下：

$$\varphi @ \varepsilon := \varphi \qquad \varphi_0 \bullet \varphi_1 @ ia := \varphi_i @ a \qquad \bigcirc \varphi @ 0a := \varphi @ a$$

其中 $\bullet \in \{\wedge, \vee\}$和 $\bigcirc \in \{\neg, 2, 3\} \cup \{\mu x, \nu x \mid x \in \mathrm{Var}\}$。将公式的 ν-地址集合定义为 $N(\varphi) := \{a \in \mathbb{B}^* \mid \varphi @ a = \nu x.\psi\}$。给定 $x \in \mathrm{Var}$，将 x中开放的地址集合定义为 $O_x(\varphi) := \{a \in \mathbb{B}^* \mid \varphi @ a = x$和 $\forall a' < a, \psi. \varphi @ a' \neq \nu x.\psi \wedge \varphi @ a' \neq \mu x.\psi\}$。

定义 6.17 迹解释 $\iota_{\mathbb{B}}$: $K\mu \to \mathbb{T}_{\mathbb{B}}$ 定义为 $\iota_{\mathbb{B}}(\Gamma) := \{(\varphi, a) \mid \varphi \in \Gamma \wedge a \in N(\varphi)\}$。对于每条非不动点规则 $R \in K\mu$，其形式为 $\rho(R) = (\Gamma, \Delta_1, \ldots, \Delta_n)$，轨迹映射 r_i: $\iota_{\mathbb{B}}(\Gamma) \to \iota_{\mathbb{B}}(\Delta_i)$ 只需“追踪”不动点实例。例如，若 R 是以下 $\vee$-规则的实例

$$\vee \frac{\Gamma, \varphi_0, \varphi_1}{\Gamma, \varphi_0 \vee \varphi_1}$$

那么 $r_1 := \{((\psi, a), 0, (\psi, a)) \mid (\psi, a) \in \iota_{\mathbb{B}}(\Gamma)\} \cup \{((\varphi_0 \vee \varphi_1, ia), 0, (\varphi_i, a)) \mid ia \in N(\varphi_0 \vee \varphi_1)\}$。现在假设 R 是某个不动点规则的一个实例

$$\sigma \frac{\Gamma, \varphi[\sigma x.\varphi/x]}{\Gamma, \sigma x.\varphi}$$

then

$$
\begin{aligned}
r_1 \coloneqq & \{(v, 0, v) \mid v \in \iota_{\mathbb{B}}(\Gamma)\} \cup \\
& \{((\sigma x.\varphi, \varepsilon), b, (\varphi[\sigma x.\varphi/x], a) \mid a \in O_x(\varphi)\} \cup \\
& \{((\sigma x.\varphi, 0a), 0, (\varphi[\sigma x.\varphi/x], a) \mid a \in N(\varphi)\} \cup \\
& \{((\sigma x.\varphi, 0a), 0, (\varphi[\sigma x.\varphi/x], a'0a) \mid a \in N(\varphi), a' \in O_X(\varphi)\}
\end{aligned}
$$

其中 $b = 0$ 当且仅当 $\sigma = \mu$。若将这个定义的所有细节一一展开，很快就会变得冗长繁琐，因此我们依赖读者的直觉。关于一个非常相似的迹条件的详细说明，可参见 [13]。」

今后，我们用$K\mu$表示循环证明系统 $\iota_{\mathbb{F}}(K\mu)$和 $\iota_{\mathbb{B}}(K\mu)$。无需区分二者，因为这两种轨迹解释会导出相同的循环证明系统：完全相同的预证明满足通过 $\iota_{\mathbb{B}}$ 指定的全局轨迹条件，以及通过 $\iota_{\mathbb{F}}$指定的全局轨迹条件。尽管如此，它们会导出两种不同的重置证明系统。

6.3.2 $\mathbb{F}$-重置模态 μ-演算

我们首先介绍与 $\iota_{\mathbb{F}}(K\mu)$相对应的重置证明系统$RK_{\mu}^{\mathbb{F}}$（读作“$\mathbb{F}$-重置模态μ-演算”）。

$\mathrm{RK}_{\mu}^{\mathbb{F}}$的序列式是形如 $\Theta \mid \Gamma$ 的表达式，其中控制 Θ 是一个由互不相同的字符组成的序列，称为控制；而 Γ 是 μ-公式 φ 与标注组成的有序对集合，其中标注是赋值集合 $x_1 \mapsto u_1$、…、$x_n \mapsto u_n$，其中$\{x_1$、……、$x_n\} = V_\nu(\varphi)$、u_1、……、u_n是 $u_i \sqsubseteq \Theta$ 的子序列，并且每个变量 $x \in V_\nu(\varphi)$都恰好属于一个赋值。为简化记号，我们通常将此类序列 σ直接视为函数，把 $V_\nu(\varphi)$映射到 Θ的子序列。$\mathrm{RK}_{\mu}^{\mathbb{F}}$ 的序列式集合记作$\mathsf{Seq}^{RK_{\mu}^{\mathbb{F}}}$。与 Safra 棋盘类似，若对于每个$(\varphi, \sigma) \in \Gamma$以及每个 $x \in V_\nu(\varphi)$，只要 $a \in \sigma(x)$，它就不处于该序列的最后位置，则称字母 $a \in \Theta$在 Γ中被覆盖。

定义 6.18 $\mathbb{F}$-重置模态 μ-演算的推导规则（$RK_{\mu}^{\mathbb{F}}$）如下。记Θ'为从相应 Γ 中移除所有未出现在任何标注中的字母后得到的控制。类似地，σ'表示移除所有满足 $x \notin V_\nu(\varphi)$的赋值 $x \mapsto u$后得到的标注

（这些赋值）出现在相应的 μ-公式 φ 中。

$$\textsc{Ax}\frac{}{\Theta \mid (p,\varepsilon),(\neg p,\varepsilon)} \qquad \textsc{Wk}\frac{\Theta' \mid \Gamma}{\Theta \mid \Gamma,(\varphi,\varepsilon)} \qquad \vee\frac{\Theta \mid \Gamma,(\varphi,\sigma'),(\psi,\sigma')}{\Theta \mid \Gamma,(\varphi\vee\psi,\sigma)}$$

$$\wedge\frac{\Theta' \mid \Gamma,(\varphi,\sigma') \qquad \Theta' \mid \Gamma,(\psi,\sigma')}{\Theta \mid \Gamma,(\varphi\wedge\psi,\sigma)} \qquad \textsc{Mod}\frac{\Theta \mid \Gamma,(\varphi,\sigma)}{\Theta \mid \Diamond\Gamma,(\Box\varphi,\sigma)}$$

$$\mu\frac{\Theta' \mid \Gamma,(\varphi[\mu x.\varphi/x],\sigma\setminus x)}{\Theta \mid \Gamma,(\mu x.\varphi,\sigma)} \qquad \nu\frac{\Theta a \mid \Gamma,(\varphi[\nu x.\varphi/x],(\sigma,x\mapsto ua)) \quad a\notin\Theta}{\Gamma,(\nu x.\varphi,(\sigma,x\mapsto u))}$$

$$\textsc{Reset}_a\frac{\Theta' \mid \Gamma\upharpoonright a \quad a \text{ covered in } \Gamma}{\Theta \mid \Gamma} \qquad \textsc{Merge}\frac{\Theta' \mid \Gamma,(\varphi,\mathrm{merge}_\Theta(\sigma,\sigma'))}{\Theta \mid \Gamma,(\varphi,\sigma),(\varphi,\sigma')}$$

其中 $\Diamond\Gamma \coloneqq \{(\Diamond\varphi,\sigma) \mid (\varphi,\sigma) \in \Gamma\}$ 和 $\Gamma\upharpoonright x \coloneqq \{(\varphi,\sigma\upharpoonright a) \mid (\varphi,\sigma) \in \Gamma\}$，以及上文使用的各种注释，定义如下

$$(\sigma\setminus x)(y) \coloneqq \begin{cases}\varepsilon & \text{if } x <_{\nu x.\varphi} y\\ \sigma(y) & \text{otherwise}\end{cases} \qquad (\sigma\upharpoonright a)(x) \coloneqq \begin{cases}u & \text{if } \sigma(x)=uav\\ \sigma(x) & \text{otherwise}\end{cases}$$

$$\mathrm{merge}_\Theta(\sigma,\sigma')(x) \coloneqq \min_\Theta(\sigma(x),\sigma'(x))$$

记 ε 为注释 $x_1\mapsto\varepsilon,\ldots,x_n\mapsto\varepsilon$。$RK_\mu^{\mathbb{F}}$-预证明是这样一种证明：每一对芽 $t\in\mathrm{dom}(\beta)$ 和伴随节点 $\beta(t)$ 都有一个不变量，即存在一个字母 a，使得 a 出现在 Θ 与 $\beta(t)$ 之间的所有控制中，a 在控制 Θ 中的前缀保持不变，并且重置 a 规则在 t 与 $\beta(t)$ 之间应用。若一个证明是 μ-序列式 Γ 的证明，则称其为 ${}^{\mathbb{F}}_{\mu}\vdash\Gamma RK$，如果其根节点标记为 $\varepsilon \mid \{(\varphi,\varepsilon) \mid \varphi\in\Gamma\}$。⌟

合并规则有些不够优雅，在某些情况下也可由 Wk 取代。它是完备性证明中模拟贪心运行的稀疏化步骤所必需的。我们猜想，构造与 $K\mu$-证明对应的 $RK_\mu^{\mathbb{F}}$-证明时可能需要的每个 Merge 实例，都可由 Wk 取代。事实上，Jungteerapanich-Stirling 系统 [25] 针对模态 μ-演算，采用了类似的 Thin 规则，它总是从两个 φ-实例中选择一个，且足以证明完备性。然而，要证明 Merge 是多余的，可能与直接针对模态μ-演算的语义证明$RK_\mu^{\mathbb{F}}$ 的完备性一样复杂。因此，我们不进行这项“易用性优化”，因为本文的目标是在无需投入太多精力的情况下推导出具体重置证明系统。

$RK_\mu^{\mathbb{F}}$相对于$K\mu$的可靠性，可通过构造证明态射$embed\colon RK_\mu^{\mathbb{F}}\to K\mu$来证明。

L引理 6.19 存在一个函数 $embed\colon \mathrm{Seq}^{RK_\mu^{\mathbb{F}}}\xrightarrow{}\mathrm{R}(\mathrm{Seq}^{K_\mu})$，其定义如下

$$embed(\Theta \mid \Gamma) \coloneqq \overline{\Gamma};(\overline{\Theta},\sigma_\Gamma)$$

其中，$\overline{\Theta}$ 是集合 $\{u \in \Theta\}$，按照 $\Theta, \Gamma \coloneqq \{\varphi \mid (\varphi, \sigma) \in \Gamma\}$ 中字母的位置排序，$\overline{\ }$ 而 σ_Γ 的定义如下

$$\sigma_\Gamma((\varphi, x), 0) \coloneqq \{\{a \in \sigma(x)\} \mid (\varphi, \sigma) \in \Gamma\} \qquad \sigma_\Gamma((\varphi, x), e) \coloneqq \emptyset \quad \text{if } e \in \{1, 2\}.$$

该函数可以扩展为一个证明态射 $embed\colon RK_\mu^{\mathbb{F}} \to \mathrm{R}(K\mu)$。

证明 完成该态射的主要难点在于，对于 $\Theta \mid \Gamma$，可能存在 $(\varphi, \sigma), (\varphi, \sigma') \in \Gamma$，且 $\sigma \neq \sigma'$，也就是说，同一公式的两个实例带有不同的标注。*embed* 函数会将这两个带标注的公式“合并”为一个公式实例，其中每个 ν 变量都标注有两个栈。

一个会导致这种复杂情况的简单例子是 Wk-规则。假设从 $\Gamma, (\varphi, \sigma'), (\varphi, \sigma)$ 中删除了 Wk-实例 (φ, σ)，且 $\sigma \neq \sigma'$。那么 $\Gamma, (\varphi, \sigma'), (\varphi, \sigma) = \Gamma, \varphi$。因此，要模拟 Wk，无须应用 $\mathrm{R}(K\mu)$ 中对应的 Wk-规则；只需从 φ 中移除一些标注即可。因此，模拟过程如下所示。

$$\textsc{Wk}\frac{\Theta' \mid \Gamma, (\varphi, \sigma')}{\Theta \mid \Gamma, (\varphi, \sigma'), (\varphi, \sigma)} \quad \overset{embed}{\rightsquigarrow} \quad \textsc{Weak}\frac{\Gamma, \psi; (\overline{\Theta'}, \sigma_{\Gamma,(\varphi,\sigma')})}{\overline{\Gamma}, \overline{\psi}; (\overline{\Theta}, \sigma_{\Gamma,(\varphi,\sigma'),(\varphi,\sigma)})}$$

规则 $\vee$、$\wedge$、ν 和 μ 中也可能出现类似的“折叠”问题。举例来说，考虑如下的序列式 $\Theta \mid \Gamma_0$，其中 Γ_0 如下，且 $\sigma \neq \sigma'$。为简单起见，假设 Γ 中不再有更多 $\varphi \vee \psi$ 的“副本”。要嵌入 $\vee$-规则 $\Theta \mid \Gamma_2$ 的一个应用，必须先将 $\vee$ 的 $\mathrm{R}(\mathrm{K}\mu)$ 对应规则（其中 $\varphi \vee \psi$ 既是主公式，也是“上下文” Γ、$\varphi \vee \psi$ 的一部分）应用于 Γ_0；$(\Theta, \sigma_{\Gamma_0})$。然而，由于相同公式的“折叠”，这只能得到 Γ_1；$(\Theta, \sigma_{\Gamma_1})$，因为轨迹解释 $r\colon \iota_{\mathbb{F}}(\Gamma, \varphi \vee \psi, \varphi \vee \psi) \to \iota_{\mathbb{F}}(\Gamma, \varphi \vee \psi, \varphi, \psi)$ 中既包含 $(\varphi \vee \psi, 0, \varphi \vee \psi)$，也包含 $(\varphi \vee \psi, 0, \varphi)$ 和 $(\varphi \vee \psi, 0, \psi)$。注意到 $\Gamma_1 = \Gamma_2$，即可通过应用 $\mathrm{R}(\mathrm{K}\mu)$ 的 Weak-规则得到所需的前提 Γ_2；$(\Theta, \sigma_{\Gamma_2})$。

$$\begin{aligned}
\Gamma_0 &\coloneqq \Gamma, (\varphi \vee \psi, \sigma), (\varphi \vee \psi, \sigma') \\
\Gamma_1 &\coloneqq \Gamma, (\varphi \vee \psi, \sigma), (\varphi \vee \psi, \sigma'), (\varphi, \sigma), (\psi, \sigma), (\varphi, \sigma'), (\psi, \sigma') \\
\Gamma_2 &\coloneqq \Gamma, (\varphi \vee \psi, \sigma), (\varphi, \sigma'), (\psi, \sigma')
\end{aligned}$$

就预证明而言，这得出：

$$\vee\frac{\Theta \mid \Gamma, (\varphi \vee \psi, \sigma), (\varphi, \sigma'), (\psi, \sigma')}{\Theta \mid \Gamma, (\varphi \vee \psi, \sigma), (\varphi \vee \psi, \sigma')} \quad \overset{embed}{\rightsquigarrow} \quad \vee\frac{\textsc{Weak}\dfrac{\overline{\Gamma}, \varphi \vee \psi, \varphi, \psi; (\overline{\Theta}, \sigma_{\Gamma_2})}{\overline{\Gamma}, \varphi \vee \psi, \varphi, \psi; (\overline{\Theta}, \sigma_{\Gamma_1})}}{\overline{\Gamma}, \varphi \vee \psi; (\overline{\Theta}, \sigma_{\Gamma_0})}$$

如果有更多“副本” $(\varphi \vee \psi, \sigma'') \in \Gamma$ 带有 $\sigma \neq \sigma'' \neq \sigma'$，那么与 (φ, σ'') 和 (ψ, σ'') 对应的栈也需要通过 Weak 应用移除。如果 Γ 中只有一个 $\varphi \vee \psi$ 的“副本”，那么 $\varphi \vee \psi$ 无需成为 $\vee$ 应用中“上下文” Γ 的一部分，也不需要

应用 Weak。嵌入规则 $\wedge$、ν 和 μ 的实例时可能出现的“塌缩”相关问题是类似的，可以用类似的方式处理。

如果按照上一段所述处理与“坍缩”相关的问题，那么 $\mathrm{RK}_{\mu}^{\mathbb{F}}$ 的 ν-规则可直接翻译为 $\mathrm{R}(\mathrm{K}\mu)$ 中相应的 ν-规则实例。虽然 $\mathrm{RK}_{\mu}^{\mathbb{F}}$ 的 ν-规则所涉及的操作 $\sigma \upharpoonright a$，对应于 $\mathrm{R}(\mathrm{K}\mu)$ 中相应的 ν-规则，但 μ-规则中的操作 $\sigma \setminus x$ 并非如此。考虑序列式 $b \mid \Gamma_0$，其中 $\Gamma_0 := (\mu x.\nu y.x, y \mapsto b)$。相应的 $\mathrm{R}(\mathrm{K}\mu)$ 序列式为 Γ_0；(b, σ_{Γ_0})，其中对于 $a = 0$，$\sigma_{\Gamma_0}((\mu x.\nu y.x, y)\ a) := \{\{b\}\}$；否则，$\sigma_{\Gamma_0}((\mu x.\nu y.x, y)\ a) := \emptyset$。对该序列式应用 μ-规则，得到 $\nu y.\mu x.\nu y.x$；(b, σ')，其中对于 $a = 2$，$\sigma'((\nu y.\mu x.\nu y.x, y)\ a) := \{\{b\}\}$；否则，$\sigma'((\nu y.\mu x.\nu y.x, y)\ a) := \emptyset$。另一方面，对 $b \mid \Gamma_0$ 应用 μ-规则，得到 $\mid \Gamma_1$，其中 $\Gamma_1 := (\nu y.\mu x.\nu y.x, y \mapsto \varepsilon)$。因此，需要应用 Weak-规则，从$((\nu y.\mu x.\nu y.x, y), 2)$中移除栈 $\{b\}$，得到 σ''，其中在所有 $a \in \mathbb{F}$ 上，$\sigma'((\nu y.\mu x.\nu y.x, y)\ a) := \emptyset$。此外，还需要应用 Pop-规则，向$((\nu y.\mu x.\nu y.x, y), 0)$添加一个空栈。

$$\mu \frac{\mid (\nu y.\mu x.\nu y.x, y \mapsto \varepsilon)}{b \mid (\mu x.\nu y.x, y \mapsto b)} \quad \overset{embed}{\rightsquigarrow} \quad \begin{array}{c} \textsc{Pop}\ \dfrac{\nu y.\mu x.\nu y.x; (\emptyset, \sigma_{\Gamma_1})}{\nu y.\mu x.\nu y.x; (\emptyset, \sigma'')} \\ \mu\ \dfrac{\nu y.\mu x.\nu y.x; (\overline{b}, \sigma')}{\overline{\Gamma_0}; (\overline{b}, \sigma_{\Gamma_0})} \end{array}$$

Mor包含更多 ν变量的复杂情形可以用类似的方法处理 .

Mod规则直接对应于 $\mathrm{R}(K\mu)$ 中的 Mod规则。由于在应用 Mod规则时，所有公式都是主公式，因此在转换过程中不会出现与“折叠”有关的复杂情况。重置a规则直接对应于 $\mathrm{R}(K\mu)$ 中的重置a规则。不难看出，Merge 总是可以由 Weak 的某个适当实例来模拟。■

推论 6.20（可靠性）若 $RK_{\mu}^{\mathbb{F}} \vdash \Gamma$ 则$K_{\mu} \vdash \Gamma$。

设 F 是 $K\mu$ 的一个有限片段。为了证明完备性，构造一个证明态射$search$：$\mathrm{S}(\mathrm{F}) \to RK_{\mu}^{\mathbb{F}}$，将 F 的证明搜索系统嵌入$RK_{\mu}^{\mathbb{F}}$。

引理 6.21 存在一个函数$search$：$\mathrm{S}(\mathrm{Seq}^{\mathcal{F}}) \to \mathrm{Seq}^{RK_{\mu}^{\mathbb{F}}}$，满足

$$search(\Gamma; (\Theta, \sigma)) := \widehat{\Theta} \mid \Gamma^{\sigma}$$

b，其中对于任意 $S \subseteq \Theta$，$S \in \Theta^*$ 是长度为 $|S|$的无重复序列，由 S 的元素构成，并按照 Θ 严格排序。记号 $\Gamma^{\sigma} := \{(\varphi,\ \sigma \upharpoonright \varphi) \mid \varphi \in \Gamma\}$ œwith $(\sigma \upharpoonright \varphi)\ (x) := \sigma(\ (\varphi,\ x),\ 0)$。

该函数可以扩展为证明态射$search$：$\mathrm{S}(\mathrm{F}) \to RK_{\mu}^{\mathbb{F}}$。

证明 为论证这一主张，首先选取某个 $\mathrm{R}(\Theta, \sigma) \in \mathrm{S}(\mathrm{F})$，其排列如下

$$\mathrm{R}(\Theta, \sigma) \frac{\Gamma_1; (\Theta_1, \sigma_1) \quad \ldots \quad \Gamma_n; (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

于是存在 $R \in \mathrm{F}$，使得 $\rho(R) = (\Gamma, \Gamma_1, \ldots, \Gamma_n)$，以及由轨迹解释给出的态射 $r_i \colon \iota_{\mathbb{F}}(\Gamma) \to \iota_{\mathbb{F}}(\Gamma_i)$。然后，对每个 $i \leq n$，都有 $(\Theta, \sigma) \overset{r_i}{\rightsquigarrow}_g (\Theta_i, \sigma_i)$，其中包含展开后的序列 展开后的序列

$$(\Theta, \sigma) \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_r^1, \sigma_r^1) \ldots \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_r^k, \sigma_r^k) \overset{P}{\rightsquigarrow} (\Theta_p, \sigma_p) \overset{r_i}{\rightsquigarrow} (\Theta_i^*, \sigma_i^*) \overset{T}{\rightsquigarrow} (\Theta_i, \sigma_i)$$

其中，初始的 R_γ 步和 P 步由所有 $i \leq n$ 共享（见引理 4.18）。与引理 5.8 类似，我们可以在 $RK_\mu^{\mathbb{F}}$ 中推导出以下内容：

$$\dfrac{\dfrac{\widehat{\Theta_1} \mid \Gamma_1^{\sigma_1}}{\vdots}\,\textsc{Merge}}{\dfrac{}{\widehat{\Theta_1^*} \mid \Gamma_1^0}} \quad \ldots \quad \dfrac{\dfrac{\widehat{\Theta_n} \mid \Gamma_n^{\sigma_n}}{\vdots}\,\textsc{Merge}}{\dfrac{}{\widehat{\Theta_n^*} \mid \Gamma_n^0}}\,\mathrm{R}$$

$$\dfrac{\dfrac{\widehat{\Theta_r^k} \mid \Gamma^{\sigma_r^k}}{\vdots}\,\textsc{Reset}_{\gamma_1}}{\widehat{\Theta_0} \mid \Gamma^{\sigma_0}}$$

也就是说，首先从 Θ_0 最大的 γ_0 开始，应用所有可能的重置γ_i规则。由于 Γ^σ 会将 σ 未分配任何栈的 ν 变量标注为空栈 ε，因此总体化步骤无需在预证明中像 $\Gamma^{\sigma_r^k} = \Gamma^{\sigma_p}$ 那样重复。接着应用与$R \in M$对应的规则。注意，尽管应用该规则后得到的控制与 Θ_i^* 匹配，但序列式会是某个序列式 Γ_i^0，其中可能包含同一公式的多个副本，且标注各不相同（类似于引理 6.19 中的“折叠”问题）。在这种情况下，存在某个 $(\varphi,\ x)$，σ_i^* 为其分配了两个或更多栈。贪心运行会使用稀疏化，以重新确保每个量词至多分配一个栈。在 $RK_\mu^{\mathbb{F}}$ 中，可以通过对所有公式 $(\varphi,\ \sigma)$，$(\varphi,\ \sigma') \in \Gamma_i^0$ 应用合并规则来复现这一过程。请注意，应用顺序以及选择哪两个“φ实例”进行合并操作都无关紧要，因为得到的序列式总是 $\mathfrak{b}\,\Theta_i \mid \Gamma_i^{\sigma_i}$。

类似于引理 5.8 中针对 *expand* 所给出的论证表明，*search* 维护了可靠性条件。关键在于，接受的 Safra 棋盘运行穿过 S(F) 中的证明时，永远不可能“进入某个状态” $((\varphi,\ x),\ 2)$。如果确实进入了，从那一点开始，该运行将永远无法再到达任何带有 $a < 2$的 $((\psi,\ x)\ a)$，这意味着从那一点开始只能发生有限次重置。因此，像函数*search*: S(F) $\to RK_\mu^{\mathbb{F}}$那样在 $((\varphi,\ x),\ 2)$ 上“丢弃”栈，并不会妨碍可靠性条件的维护。■

推论 6.22（完备性）如果 $K_\mu \vdash \Gamma$ 则 $RK_\mu^{\mathbb{F}} \vdash \Gamma$。

可以看出，在将 *embed* 和 *search* 确立为证明态射时，大多数问题都围绕着公式同一性的“合并”展开。规避这类问题的一个简单办法，是将 $K\mu$ 的序列式视为公式列表，其中同一公式可以有多个不同的出现位置（并以类似列表的序列式来定义 $RK_\mu^{\mathbb{F}}$ ）。我们选择不这样做，因为这会偏离 [3] 中对 $K\mu$ 的呈现方式。本文的一个核心目标是提供生成重置系统的“配方”，且无需对

原始循环证明系统做任何修改。因此，在这个例子中这样做会违背这一目标。尽管如此，我们仍建议在为循环证明系统生成重置系统时遇到类似问题的读者考虑进行此类修改，以减轻证明负担。

6.3.3 $\mathbb{B}$-重置模态 μ-演算

接下来，我们介绍与 $\iota_{\mathbb{B}}(K\mu)$ 对应的重置证明系统 $RK_\mu^{\mathbb{B}}$（读作"$\mathbb{B}$-重置模态μ-演算"）。

该系统使用带标注量词的 μ-公式。也就是说，每个 ν-量词 $\nu^u x.\varphi$ 都标注有一个由互不相同的字符组成的序列 u:

$$\varphi \in \textsc{Form}_{\mathbb{B}} ::= p \mid \neg p \mid x \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \Box\varphi \mid \Diamond\varphi \mid \mu x.\varphi \mid \nu^u x.\varphi \quad p \in \mathrm{Prop}, x \in \mathrm{Var}$$

给定 $\varphi \in \mathrm{Form}_{\mathbb{B}}$，记 $\overline{\varphi} \in \mathrm{Form}$ 为移除φ中的标注后得到的 μ-公式。给定一个由互不相同的字母组成的序列 Θ，定义偏函数 $\mathrm{Merge}_\Theta : \mathrm{Form}_{\mathbb{B}} \times \mathrm{Form}_{\mathbb{B}} \to \underline{\mathrm{Form}}_{\mathbb{B}}$，使得 $\mathrm{Merge}_\Theta(\varphi, \psi)$ 在 $\varphi, \psi \in \mathrm{Form}_{\mathbb{B}}$ 上有定义，当且仅当 $\overline{\varphi} = \psi$，并且 φ, ψ 中的标注是 Θ 的子序列。定义如下:

$$\mathrm{merge}_\Theta(a, a) := a \qquad \mathrm{merge}_\Theta(\varphi_0 \bullet \varphi_1, \psi_0 \bullet \psi_1) := \mathrm{merge}_\Theta(\varphi_0, \psi_0) \bullet \mathrm{merge}_\Theta(\varphi_1, \psi_1)$$

$$\mathrm{merge}(\bigcirc\varphi, \bigcirc\psi) := \bigcirc\mathrm{merge}(\varphi, \psi) \qquad \mathrm{merge}(\nu^u x.\varphi, \nu^v x.\psi) := \nu^{\min_\Theta(u,v)} x.\mathrm{merge}(\varphi, \psi)$$

其中 $a \in \mathrm{Prop} \cup \mathrm{Var}$，$\bullet \in \{\wedge, \vee\}$ 且 $\bigcirc \in \{\neg, 2, 3\} \cup \{\mu x \mid x \in \mathrm{Var}\}$，$\min_\Theta(u, v)$ 是根据定义 4.1 中给出的序关系 $<_\Theta$ 所确定的最短序列。

如果在每个标注 u 中，只要 a 出现在 u中，a就不处于 u的最后位置，则称字母 a 在 φ 中被覆盖。这个概念可扩展到带标注公式的集合 Γ。重置操作 $\varphi \upharpoonright a$ 定义如下:

$$p \upharpoonright a := p \qquad \varphi \bullet \psi \upharpoonright a := (\varphi \upharpoonright a) \bullet (\psi \upharpoonright a) \qquad \bigcirc\varphi \upharpoonright a := \bigcirc(\varphi \upharpoonright a)$$

$$\nu^u x.\varphi \upharpoonright a := \nu^{u'} x.\varphi \upharpoonright a \qquad \text{where } u' := \begin{cases} u & \text{if } \sigma(x) = uav \\ \sigma(x) & \text{otherwise} \end{cases}$$

其中 $p \in$ 表示命题，$\cup$ 表示变量，$\bullet \in \{\wedge, \vee\}$以及 $\bigcirc \in \{\neg, 2, 3\} \cup \{\mu x \mid x \in$ 变量$\}$。

$RK_\mu^{\mathbb{B}}$的序列式是形如 $\Theta \mid \Gamma$ 的表达式，其中控制 Θ是由互不相同的字符组成的序列，Γ 是 $\mathrm{Form}_{\mathbb{B}}$ 中公式构成的有限集，并且 Γ 中出现的每个量词 $\sigma^u x.\varphi$ 都标注有 Θ 的一个子序列。对于一个公式 φ，记 φ^ϵ 为其中所有量词都标注为空序列的公式 ε；此记法也扩展到公式集 Γ。$RK_\mu^{\mathbb{B}}$-序列式的集合记作 $\mathrm{Seq}^{RK_\mu^{\mathbb{B}}}$。

定义 6.23 $RK_\mu^{\mathbb{B}}$的推导规则如下。记 Θ' 为从控制中删除 Γ 中任何标注都未出现的所有字母后所得的控制。

$$\textsc{Ax}\frac{}{\Theta \mid p, \neg p} \qquad \textsc{Wk}\frac{\Theta' \mid \Gamma}{\Theta \mid \Gamma, \varphi} \qquad \vee\frac{\Theta \mid \Gamma, \varphi, \psi}{\Theta \mid \Gamma, \varphi \vee \psi} \qquad \wedge\frac{\Theta' \mid \Gamma, \varphi \qquad \Theta' \mid \Gamma, \psi}{\Theta \mid \Gamma, \varphi \wedge \psi}$$

$$\textsc{Mod}\frac{\Theta \mid \Gamma, \varphi}{\Theta \mid \Diamond\Gamma, \Box\varphi} \qquad \mu\frac{\Theta \mid \Gamma, \varphi[\mu x.\varphi/x]}{\Theta \mid \Gamma, \mu x.\varphi} \qquad \nu\frac{\Theta a \mid \Gamma, \varphi[\nu^{u a} x.\varphi/x] \quad a \notin \Theta}{\Theta \mid \Gamma, \nu^{u} x.\varphi}$$

$$\textsc{Reset}_a\frac{\Theta' \mid \Gamma \upharpoonright a \quad a \text{ covered in } \Gamma}{\Theta \mid \Gamma} \qquad \textsc{Merge}\frac{\Theta' \mid \Gamma, \xi \quad \mathrm{merge}_\Theta(\varphi, \psi) = \xi}{\Theta \mid \Gamma, \varphi, \psi}$$

若每一对芽 $t \in$ 定义域(β)与伴随节点 $\beta(t)$都有一个不变量，也就是说，存在一个字母 a，使得 a 出现在 t与 $\beta(t)$之间的所有控制 Θ中， a 在控制 Θ 中的前缀保持不变，并且重置$_a$ 规则在 t 与 $\beta(t)$之间应用，那么 $RK_\mu^{\mathbb{B}}$-预证明就是一个证明 。若某个证明 的根节点标记为 $\varepsilon \mid \Gamma^{\epsilon}$，则该$RK_\mu^{\mathbb{B}}$-证明是 Γ的一个证明。若在$RK_\mu^{\mathbb{B}}$中存在 Γ的证明，则记作$RK_\mu^{\mathbb{B}} \vdash \Gamma$。⌟

通过构造证明态射$embed\colon RK_\mu^{\mathbb{B}} \to K\mu$，可以证明$RK_\mu^{\mathbb{B}}$ 相对于$K\mu$ 的可靠性。

L引理 6.24 存在一个函数$embed\colon \mathsf{Seq}^{RK_\mu^{\mathbb{B}}} \to \mathrm{R}(\mathsf{Seq}^{K\mu})$，其定义如下：

$$embed(\Theta \mid \Gamma) := \overline{\Gamma}; (\overline{\Theta}, \sigma_\Gamma)$$

其中，$\overline{\Theta}$ 是集合 $\{u$ 在 Θ中出现$\}$，按字母在 Θ中的位置排序； Γ 是移除了所有注释后的 Γ，而 σ_Γ 定义如下：

$$(\overline{\varphi}, i) \mapsto \{\{a \text{ occurs in } u\} \mid \psi \in \Gamma \text{ and } \overline{\psi} = \overline{\varphi} \text{ and } \psi @ i = \nu^u x.\xi\} \text{ for each } \varphi \in \Gamma \text{ and } i \in N(\varphi)$$

该函数可以扩展为一个证明态射 嵌入 : $RK_\mu^{\mathbb{B}} \to K\mu$。

证明 与 $RK_\mu^{\mathbb{B}}$ 一样，构造该态射的主要难点在于，对于 $\Theta \mid \Gamma \in \mathsf{Seq}'$，可能存在两个公式 $\varphi, \psi \in \Gamma$ 且满足 $\overline{\varphi} = \psi$，也就是说，同一公式的两个实例具有不同的标注。*embed* 函数会将这两个带标注的公式“合并”为一个公式实例，其中每个 ν-实例都带有两个栈。可能出现的两类难点及其通过 $K\mu$-规则 Wk 和 R($K\mu$)-规则 Weak 对其进行的处理，与 $RK_\mu^{\mathbb{F}}$ 中的情况完全类似。因此，关于这些问题的更详尽讨论，请读者参阅引理 6.19。

每条 $RK_\mu^{\mathbb{B}}$ 规则都被转换为对应的 R($K\mu$)-规则。规则 Merge 总是通过 Weak 的一个适当实例来模拟。■

推论6.25（可靠性） 若 $RK_\mu^{\mathbb{B}} \vdash \Gamma$ 则$K_\mu \vdash \Gamma$。

设 F 是$K\mu$的一个有限片段。为证明完备性，需要构造一个证明态射$search\colon \mathsf{S}(\mathrm{F}) \to RK^{\mathbb{B}}_{\mu}$，将 F 的证明搜索系统嵌入$RK^{\mathbb{B}}_{\mu}$。

引理 6.26 存在一个函数 $search\colon \mathsf{S}(\mathsf{Seq}_{\mathcal{F}}) \to \mathsf{Seq}^{RK^{\mathbb{B}}_{\mu}}$，满足

$$search(\Gamma; (\Theta, \sigma)) \coloneqq \widehat{\Theta} \mid \Gamma^{\sigma}$$

b，其中对于任意 $S \subseteq \Theta, S \in \Theta^{*}$，它是长度为 $|S|$ 的无重复序列，由 S 的元素组成，并按照 Θ 严格排序。记号 $\Gamma^{\sigma} \coloneqq \{\varphi_{\varepsilon}^{\sigma\restriction\varphi} \mid \varphi \in \Gamma\}$ with$^{\frown}(\sigma \restriction \varphi)\,(a) \coloneqq \sigma(\varphi, a)$ 和 φ_a^{σ} 递归定义如下

$$c_a^{\widehat{\sigma}} \coloneqq c \qquad (\varphi \bullet \psi)_a^{\widehat{\sigma}} \coloneqq \varphi_{a0}^{\widehat{\sigma}} \bullet \psi_{a1}^{\widehat{\sigma}} \qquad (\bigcirc\varphi)_a^{\widehat{\sigma}} \coloneqq \bigcirc\varphi_{a0}^{\widehat{\sigma}} \qquad (\nu x.\varphi)_a^{\widehat{\sigma}} \coloneqq \begin{cases} \nu^{\widehat{S}} x.\varphi_{a0}^{\widehat{\sigma}} & \text{if } \widehat{\sigma}(i) \coloneqq \{S\} \\ \nu^{\epsilon} x.\varphi_{a0}^{\widehat{\sigma}} & \text{otherwise} \end{cases}$$

其中 $c \in \mathsf{Prop} \cup \mathsf{Var}$，$\bullet \in \{\wedge, \vee\}$ 和 $\bigcirc \in \{\neg,\ 2,\ 3\} \cup \{\mu x \mid x \in \mathsf{Var}\}$。

该函数可以扩展为证明态射 $search\colon \mathsf{S}(\mathrm{F}) \to RK^{\mathbb{B}}_{\mu}$。

证明 为证明这一主张，先取一个 $\mathrm{R}(\Theta, \sigma) \in \mathsf{S}(\mathrm{F})$，其排列如下

$$\mathrm{R}(\Theta,\sigma)\frac{\Gamma_1; (\Theta_1, \sigma_1) \quad \dots \quad \Gamma_n; (\Theta_n, \sigma_n)}{\Gamma; (\Theta, \sigma)}$$

那么存在一个 $R \in \mathrm{F}$，使得 $\rho(R) = (\Gamma,\ \Gamma_1,\ \dots,\ \Gamma_n)$，并且存在态射 $r_i\colon \iota_{\mathbb{B}}(\Gamma) \to \iota_{\mathbb{B}}(\Gamma_i)$，由轨迹解释给出。于是，对每个 $i \leq n$，都有 $(\Theta,\ \sigma) \overset{r_i}{\rightsquigarrow}_g (\Theta_i,\ \sigma_i)$，其展开序列为该展开序列

$$(\Theta, \sigma) \overset{R_{\gamma_1}}{\rightsquigarrow} (\Theta_r^1, \sigma_r^1) \dots \overset{R_{\gamma_k}}{\rightsquigarrow} (\Theta_r^k, \sigma_r^k) \overset{P}{\rightsquigarrow} (\Theta_p, \sigma_p) \overset{r_i}{\rightsquigarrow} (\Theta_i^*, \sigma_i^*) \overset{T}{\rightsquigarrow} (\Theta_i, \sigma_i)$$

其中，初始的 R_{γ}- 和 P-步骤由所有 $i \leq n$ 共享（参见引理 4.18）。与引理 5.8 类似，我们可以在 $RK^{\mathbb{B}}_{\mu}$ 中推导出以下结论：

$$\textsc{Reset}_{\gamma_1}\dfrac{\mathrm{R}\dfrac{\textsc{Merge}\dfrac{\widehat{\Theta_1} \mid \Gamma_1^{\sigma_1}}{\widehat{\Theta_1^*} \mid \Gamma_1^0} \quad \dots \quad \textsc{Merge}\dfrac{\widehat{\Theta_n} \mid \Gamma_n^{\sigma_n}}{\widehat{\Theta_n^*} \mid \Gamma_n^0}}{\widehat{\Theta_r^k} \mid \Gamma^{\sigma_r^k}}}{\widehat{\Theta_0} \mid \Gamma^{\sigma_0}}$$

也就是说，首先从最大的 γ_0 开始，应用所有可能的重置γ_i规则。由于 Γ^{σ} 不会将栈标注给 σ“分配”了 ν^{ε} 的不动点，因此无需在预证明中将总体化步骤复制为 $\Gamma^{\sigma_r^k} = \Gamma^{\sigma_P}$。接着应用与 $R \in M$ 对应的规则。请注意，尽管此次应用产生的控制与 Θ_i^* 相符，但序列式将是某个序列式 Γ_i^0，其中可能包含多个“相同”公式的副本，而它们的

标注不同（类似于 σ_i^*将多个栈分配给其某些参数）。一个简单的例子是，若 $\Gamma = \varphi, \psi \vee \xi$ 满足 $\overline{\varphi} = \psi$，但 $\varphi \neq \psi$。若对这个序列式应用 $\vee$ 规则，得到的序列式 $\Gamma_1^0 = \varphi, \psi, \xi$ 将包含两个 φ 的“副本”（对应于 σ_1^* 将多个栈分配给 φ中的某些量词）。在这种情况下，贪心运行会使用稀疏化，重新确立每个量词至多被分配一个栈这一性质。在 $RK_\mu^{\mathbb{B}}$ 中，可以通过对所有满足 $\overline{\varphi} = \psi$ 的公式 $\varphi, \psi \in \Gamma_i^0$应用合并规则来复现这一做法。请注意，应用顺序以及选择哪两个“φ 实例”进行合并操作都无关紧要，因为得到的序列式总是 $\mathrm{b}\,\Theta_i \mid \Gamma_i^{\sigma_i}$。

与引理 5.8 中给出的 expand 论证类似的论证表明，search 保持可靠性条件。■

推论 6.27（完备性）如果 $K_\mu \vdash \Gamma$ 那么 $RK_\mu^{\mathbb{B}} \vdash \Gamma$。

6.3.4 Jungteerapanich-Stirling 系统

第一个重置证明系统由 Jungteerapanich [12] 为模态 μ-演算提出。该系统是一个表列系统，它为 μ-序列式的可满足性导出了一个判定算法。后来，Stirling [25]将该系统转换为一个正则有效性证明系统。后一个系统在文献中通常称为 Jungteerapanich-Stirling（$\mathcal{JS}$）。该系统也受到 Safra 构造的启发（详情见 [12, 第 4.3 节5] ），但正如我们稍后指出的，最终的系统相当独特。其可靠性和完备性是直接依据模态 μ-演算的语义证明的，而非像本文采用的论证那样依赖自动机理论。由于$\mathcal{JS}$ 在循环证明理论领域广为人知，因此值得将我们的系统 $RK_\mu^{\mathbb{F}}$ 和 $RK_\mu^{\mathbb{B}}$ 与之比较。为更好地符合本文的记号风格，我们对$\mathcal{JS}$ 的介绍与 [25] 中的介绍略有不同。

系统 $\mathcal{JS}$假设变量按照某个固定的线性序 $<$排列，这些变量用大写字母表示 X， Y， Z。在本节余下部分，我们只考虑满足良命名条件且包含序 $<_\varphi$ 与变量序关系 $<$一致的公式 φ。有限个良命名公式总能通过 α重命名，使其满足这一性质。对于每个变量 X，都有无限多个与之关联的名称 x_1， x_2，..... X。不同变量的名称互不相同。

序列式 $\mathcal{JS}$的形式为 $\Theta \mid \Gamma$，其中 Γ 是由带标注的 μ-公式构成的有限集 φ^u，而控制 Θ 是一个有限且不重复的序列，其中包含公式中出现的变量名称$\varphi^u \in \Gamma$。u的标注 u 是 Θ的子序列 $u \sqsubseteq \Theta$。此外，它们必须按照变量的序关系 $<$排列；也就是说，若名称 x属于 X，名称 y属于 Y，且它们分别出现在 u和 $X < Y$中，则 x必须在 u中先于 y出现。

对于 $u \sqsubseteq \Theta$，用 $u \restriction X$ 表示从 u 中移除所有与变量 $Y > X$ 对应的名称后得到的子序列。对于两个名称 x， $y \in \Theta$，若 x 是 X 的名称，且 y 是 Y 的名称，其中 $X < Y$；或者若 x 和 y 是同一个变量 X 的名称，且 x 出现在 y 之前，则记作 $x \sqsubset_\Theta y$

在 Θ 中。这一概念可推广到序列 u，$v \sqsubseteq \Theta$，记作 $u \sqsubset_\Theta v$，若 u 包含仅出现在两个序列之一中的、按 $\sqsubset_\Theta$ 排序最靠前的名称。

定义 6.28 *JS* 的推导规则如下。用 Θ' 表示从相应的 Γ 中移除所有未出现在注释中的名称后得到的控制 Θ。

$$\text{Ax}\frac{}{\Theta \mid \Gamma, p^u, \neg p^v} \qquad \vee\frac{\Theta \mid \Gamma, \varphi^u, \psi^u}{\Theta \mid \Gamma, \varphi \vee \psi^u} \qquad \wedge\frac{\Theta \mid \Gamma, \varphi^u \quad \Theta \mid \Gamma, \psi^u}{\Theta \mid \Gamma, \varphi \wedge \psi^u} \qquad \Box\frac{\Theta \mid \Gamma, \varphi^u}{\Theta \mid \Diamond\Gamma, \Box\varphi^u}$$

$$\mu\frac{\Theta' \mid \Gamma, \varphi[\mu X.\varphi/X]^{u\restriction X}}{\Theta \mid \Gamma, \mu X.\varphi^u} \qquad \nu\frac{\Theta' x \mid \Gamma, \varphi[\nu X.\varphi/X]^{(u\restriction X)x} \quad x \text{ fresh } X \text{ name}}{\Theta \mid \Gamma, \nu X.\varphi^u}$$

$$\textsc{Thin}\frac{\Theta' \mid \Gamma, \varphi^u \quad u \sqsubset_\Theta v}{\Theta \mid \Gamma, \varphi^u, \varphi^v} \qquad \textsc{Reset}_x\frac{\Theta' \mid \Gamma, \varphi_1^{ux}, \ldots, \varphi_n^{ux} \quad x \text{ not in } \Gamma}{\Theta \mid \Gamma, \varphi_1^{uxx_1u_1}, \ldots, \varphi_n^{uxx_nu_n}}$$

如果每一对芽 $t \in \mathrm{dom}(\beta)$和伴随节点 $\beta(t)$都具有一个不变量，那么$\mathcal{JS}$-预证明就是一个证明 *</style}*，也就是说，存在一个名称，使得出现在与*()*之间的所有控制中，并且在与*()*之间应用了重置规则。若一个*JS*-证明的根节点标记为 $\varepsilon \mid \Gamma^\varepsilon$，则该证明就是 Γ 的证明。 ⌟

如果在绑定子 νX 或 μX 与其每个约束变量 X之间的路径上都出现 2，那么 μ-公式就是受限的。以下内容在 [25, 定理 4] 中得到证明。

若 γ是闭合且受限的，则**命题 6.29** $\mathcal{JS} \vdash \gamma$当且仅当 γ有效。

乍一看，系统$RK_\mu^{\mathbb{B}}$ 和$\mathcal{JS}$ 截然不同：$RK_\mu^{\mathbb{B}}$ 为量词实例添加注释，而$\mathcal{JS}$的注释涉及 Γ 中的公式及其 ν-变量（很容易看出，这些注释绝不可能扩展到包含 μ-变量）。两者的共同点仅在于，它们都是模态 μ-演算的重置证明系统。对比$RK_\mu^{\mathbb{F}}$和$\mathcal{JS}$ 将会揭示更多有意义的内容。

从某种意义上说，$RK_\mu^{\mathbb{F}}$和$\mathcal{JS}$ 的注释都“追踪”相同的轨迹值：Γ 中每个公式各自的 ν 变量。在$RK_\mu^{\mathbb{F}}$中，这一点显而易见：Γ 中的每个公式 φ 都带有一个赋值 σ，它会为 φ 中由 νX 绑定的每个变量 X 赋值，这些变量构成控制Θ的一个子序列。在$\mathcal{JS}$ 中，这一点则更为微妙：乍看之下，注释只针对每个公式φ。然而，这些注释仅由 φ中出现的变量[2]构成。 μ规则和 ν规则确保注释中的名称按照变量上的全局序 $<$ 排列。从这个意义上说， φ^u的注释 u可以拆分为名称 $x_1 \ldots x_n$的子序列；对于 φ中的每个变量 X，这些子序列都会追踪各个轨迹值（φ， X）的进展（就 GTC $\iota_{\mathbb{F}}$的含义而言，适用于

[2]这并不完全准确。更确切地说，注释中的名称对应的是在公式中「遗传式」出现过的变量。例如，公式 $\nu X.\nu Y.Y$ 在应用两次 ν-规则后，将展开为由序列 xy 标注的 $\nu Y.Y$，并具有显而易见的变量对应关系，即使 X 已「不再」出现在 $\nu Y.Y$ 中。然而，这些「异常」只会作为标注的前缀出现，而这些标注「最终」将不再影响证明搜索。

K^{μ}）。由此可见，两个系统的 μ规则本质上相同：它通过移除相关注释，“取消”被展开的 μ公式所涵盖的 ν变量取得的所有进展。

序关系 $\sqsubset_{\Theta}$ 是字典序：首先按照变量上的全局序 $<$比较，其次按照 Θ中位置所给出的“年龄序”比较。这体现了$RK_{\mu}^{\mathbb{F}}$与$\mathcal{JS}$之间最重要的区别：$\mathcal{JS}$ 的设计考虑了对模态 μ-演算语义的一些更深刻的洞见，特别是包含序 $<_{\varphi}$ 在有效性条件中的作用。这些洞见也体现在 ν-规则中：在$\mathcal{JS}$中展开公式 $\nu X.\varphi^{u}$ 的 ν-量词时，所得标注为 $(u \upharpoonright X)x$，即“清除”被 X包含的变量的注释。这是必要的，可确保标注中的名称仍按全局变量序 $<$排列（否则，名称 x 很可能会被追加在被包含变量的名称之后，从而破坏这一序关系）。该规则的完备性依赖于一项语义洞见：如果公式中的一个 ν-变量存在成功轨迹，那么所有包含它的“更高” ν-变量也必须存在成功轨迹。因此，在“更高”ν-变量取得进展时遗忘“更低” ν-变量的进展，并不会危及完备性。Jungteerapanich 和 Stirling 在一定程度上能够“这样做而不受影响”，因为他们直接依据模态 μ-演算的语义证明可靠性和完备性，省去了本文所作的自动机理论考量。要在更通用的设置中，即在轨迹对象上带有包含序的轨迹范畴中证明类似结果，则极其复杂。因此，系统$RK_{\mu}^{\mathbb{F}}$并未采用这种“优化”，因为这会偏离我们最初的目标：以简单、“毫不费力”的方式生成重置证明系统。

JS 与 $\mathrm{RK}_{\mu}^{\mathbb{F}}$ 之间另一个类似的差异是 JS 的 Thin 规则。需要指出的是，JS 之所以只采用这种严格的稀疏化规则，而不采用更一般的弱化规则，唯一原因在于它最初被设计为一种用于模态 μ-演算证明搜索的表列系统。Thin 规则是使该系统具备完备性的最弱规则（其作用类似于 Safra 棋盘中的稀疏化步骤）。然而，若加入一条允许从 Γ 中任意舍弃公式的一般弱化规则，JS 的可靠性仍不会改变。就此而言，JS 与 $\mathrm{RK}_{\mu}^{\mathbb{F}}$ 之间值得关注的比较点，是下文将提及的 JS 中的 Thin 和 $\mathrm{RK}_{\mu}^{\mathbb{F}}$ 中的 Merge。

$$\textsc{Thin}\ \frac{\Theta' \mid \Gamma, \varphi^{u} \quad u \sqsubset_{\Theta} v}{\Theta \mid \Gamma, \varphi^{u}, \varphi^{v}} \qquad\qquad \textsc{Merge}\ \frac{\Theta' \mid \Gamma, (\varphi, \mathrm{merge}_{\Theta}(\sigma, \sigma'))}{\Theta \mid \Gamma, (\varphi, \sigma), (\varphi, \sigma')}$$

这两条规则的引入，实质上都是出于同一个原因：为了对应完备性所要求的贪心 Safra 棋盘运行中的稀疏化步骤。Thin 规则再次体现了一种语义洞见：只需“保留”在最高变量上“进展最佳”的公式。为说明这一点，假设 Θ 是 $x_1 y_1 x_2$，对应于 φ中的变量 X和 Y，且序列式中包含两份 φ：$\varphi^{x_1 x_2}$和 $\varphi^{x_1 y_1}$。在这种情况下，应用 Thin 会舍弃 $\varphi^{x_1 y_1}$，本质上是因为 $x_1 x_2$“在 X上进展更佳”。这便足以保证“完备性”，因为正如上一段所阐述的，在“较低”变量上的“成功”总会意味着在“较高”变量上的“成功”。另一方面，$RK_{\mu}^{\mathbb{F}}$ 中的 Merge 规则则再次没有体现这一洞见。

考虑类似的情况：一个形式为 abc 的控制 Θ，以及 φ 的两份带标注副本，分别为 $(\varphi,(X \mapsto ac, Y \mapsto \varepsilon))$ 和 $(\varphi,(X \mapsto a, Y \mapsto b))$。在这种情况下，Merge 规则会在 φ的合并标注中“保留”标注 $X \mapsto ac$和 标注 $Y \mapsto b$，分别作为每个变量的“最佳”标注。这是为了对应贪心 Safra 棋盘运行的稀疏化步骤，因为它们也会采取相同做法。虽然在 $RK_{\mu}^{\mathbb{F}}$ 中，Merge 改为始终“保留”对“最高”变量具有“最佳”标注的标注 σ，很可能同样是可靠且完备的，但要证明这一点仍会极其复杂。

7 结论

我们证明了，对于每个可靠性条件以激活代数 A 表述的循环证明系统 R，都存在一个相应的循环证明系统 R(R)，其可靠性条件基于 Reset。R(R) 的构造完全独立于 R 的底层逻辑，只依赖于以 A 表述的全局轨迹条件。我们通过循环证明系统同态以及一个专为简化证明搜索而设计的证明系统 S(R)，证明了 R 与 R(R) 等价。循环证明系统同态的方法使得 R 与 R(R) 之间的等价性可以扩展到定制的基于 Reset 的证明系统 S，如第 6 节所示。我们运用这一策略，为循环算术、G¨odel 的 T 和模态 μ-演算给出了等价的重置系统。

讨论 我们的方法也有一些不足。首先，尽管适用范围广泛，我们的结果并非普遍适用。我们仅展示了如何为具有可由 $\mathsf{T}_{\mathcal{A}}$ 表述的全局轨迹条件的循环证明系统给出相应的重置系统。有些全局轨迹条件可能无法用这种方式表述，例如 Hazard 为超限表达式 [11] 给出的条件。此外，我们的方法要求原系统 R 的可靠性条件是全局轨迹条件。我们没有考虑其他类型的可靠性条件，例如归纳序 [24], 弹跳线程 [7] 或轨迹流形 [8]。另一个不足是，由适当的循环证明系统 R 生成的朴素循环系统 R(R) 可能会“难以驾驭”。对于我们在第6节给出的每个具体重置系统，都需要进行一些修改，才能使最终系统便于人类使用，尤其是为具体重置系统的序列式找到合适的“语法”。更一般地说，我们认为，要将 R(R) 变成一个“易用”的系统，仍需一定程度的人为创造。我们所述方法最大的优势之一——它不依赖于关于逻辑的语义考量——同时也是其最大的缺点之一：该方法生成的系统无法利用相关逻辑的语义洞见；要证明我们的系统与文献中的其他重置系统等价，可能需要付出相当大的努力。Jungteerapanich 为模态 μ-演算 [12] 设计的重置系统就是一个例子：它以深刻的语义洞见为基础，而 R($K\mu$) 对 $K\mu$ 的两种轨迹解释都没有采用这种做法（详见第6.3.4节）。

需要指出的是，R(R) 仅描述了设计重置证明系统的一种方式。

在 [1] 中给出的重置证明系统展示了其可能存在的变化。该系统中使用的重置规则 RS(κ) 对应于 Safra 棋盘上的以下转移：固定一个棋盘 $(\Theta, \sigma) \in \mathrm{Sb}(\mathrm{A}\ X)$，并选取一个被覆盖的 $\kappa \in \Theta$。将 κ 的子节点集合 $C(\kappa)$ 定义为

$$C(\kappa) \coloneqq \left\{ \min_{\Theta}\{\gamma \in S \mid \kappa < \gamma\} \;\middle|\; x \in X, a \in \mathcal{A}, S \in \sigma(a, x) \text{ and } \kappa \in S \right\}.$$

如果每个 $\gamma \in C(\kappa)$ 也都被覆盖，就可以执行一次“重置操作”，得到棋盘 $(\Theta \setminus C(\kappa)\ \sigma \setminus C(\kappa))$，其中 $(\sigma \setminus C(\kappa))\ (x,\ a) \coloneqq \{S \setminus C(\kappa) \mid S \in \sigma(x,\ a)\}$。将定义 4.6 中的重置转移替换为这一变体的重置条件，即可得到一个抽象重置证明系统，其可靠性和完备性与我们选用的 R(R) 形式相同，不过需要对证明稍作修改。我们提出的“重置机制”很可能还有多种可行的替代方案。我们只介绍了其中一种，即最接近 Rabin 自动机传统 Safra 构造的方案。

相关工作 我们知道有三篇文章为循环证明系统设计了重置系统：Jungteerapanich-Stirling [12, 25], Afshari 等人 [1] 以及 Afshari 等人 [2]。

Jungteerapanich [12] 和 Stirling [25] 分别为模态 μ-演算提出了重置证明系统，分别用于可满足性和有效性。这是循环证明理论文献中首次出现重置系统。关于他们的有效性系统与本文推导出的用于 μ-演算有效性的系统之间的比较，见第6.3.4节。总体而言，尽管他们的系统同样受到 Safra 构造的启发，但也融入了对 μ-演算语义的多项见解，而我们的系统忽略了这些见解。

Afshari 等人 [1] 为一阶 μ-演算提出了一个重置证明系统。该系统基于 Sprenger 和 Dam [23, 24]提出的一阶 μ-演算循环证明系统，该系统采用了序数近似。其设计背后的关键洞见是，Sprenger-Dam 系统中对序数变量进行排序的机制，已经与 Jungteerapanich-Stirling 系统的控制机制非常相似。因此，Sprenger-Dam 系统得以以一种非常自然的方式扩展为重置系统。

Afshari 等人 [2] 为完全计算树逻辑（CTL*）提出了一种重置证明系统。他们的系统是一种超序列演算，因此需要更复杂的迹条件。他们的重置条件中使用的注释要么为空，要么是一个字母。在这方面，他们的系统介于“完整”的重置证明系统（例如 $\mathfrak{Js}$、针对一阶 μ-演算的系统或我们的系统）与后续段落讨论的“纯粹”路径条件系统之间。

需要指出的是，所有上述重置系统都是通过结合自动机理论考量（具体而言，即 Safra 构造）以及关于其所针对逻辑的语义洞见而设计的。在这两种情况下，由此得到的系统都比本文生成的系统更加简洁优雅。不过，设计出如此优雅的系统需要付出相当大的努力。此外，我们认为，对于循环证明理论中的许多技术用途而言，本文推导出的朴素重置证明系统应当已经足够。

文献中还探讨了带路径条件的循环证明系统，即其轨迹条件允许分别考察预证明中的每个简单环；这类系统并非严格意义上的重置证明系统，因为它们以更简单的方式实现可靠性条件。具体而言，Marti 和 Venema [15] 证明，对于 μ-演算的无交替片段，Jungteerapanich-Stirling 系统可以简化为一种系统，其中公式只需标注“一位信息”（分别称为处于焦点或不处于焦点），且无需重置规则。由此得到的系统仍具有路径条件。Rooduijn [20] 为带主模态的模态逻辑循环证明系统给出了一个非常相似的路径条件。需要指出的是，本文提及的重置证明系统的所有优良性质，例如适用于证明论研究和证明搜索，也适用于所有带路径条件的循环证明系统。

带路径条件的循环证明系统（例如重置证明系统）已被证明非常适用于证明论研究。迄今为止，运用这些系统的成果集中在插值 [4, 5, 15] 以及将循环证明系统中的证明转换为具有适当归纳公理的非循环证明系统中的证明 [3]这两个领域。然而，似乎没有理由认为带路径条件的循环证明系统不能用于证明其他性质，例如割消去，或研究循环证明的计算内容。

未来工作 本文的结果为未来研究开辟了许多方向。如前所述，重置证明系统已被证明是循环证明理论工具箱中的宝贵工具。如今，许多其他循环证明系统也已有现成可用的重置系统，我们希望未来能看到更多使用重置证明系统开展的证明论研究。这对于具有特别适合循环证明系统的特征（例如不动点和归纳定义）的逻辑的证明理论而言，可能尤其有价值。

本文利用 [6] 提出的轨迹抽象概念，详细阐述了全局轨迹条件与重置路径条件之间的关系。循环证明还存在其他可靠性条件，例如归纳序、 [24], 弹跳线程 [7] 和轨迹流形 [8]。我们希望未来能通过在这一抽象框架下研究这些其他可靠性条件及其相互关系，来“完整呈现全貌”。值得沿着这一方向探索的一个有趣之处是，在某些重置证明系统中，归纳序实质上可以从系统的证明中“读出”。第 6.2.2 节给出的哥德尔 T 系统就是一个例子。然而，并非所有重置证明系统都具备这一性质。例如，第 6.1.2 节中的皮亚诺算术系统就不具备这一性质。

参考文献

[1] Bahareh Afshari、SebastianEnqvist 和 Graham E Leigh。“一阶 μ-演算的循环证明”。发表于：*IGPL* 逻辑期刊 （2022年8月3日），第32页。

[2] Bahareh Afshari、Graham E Leigh 和 Guillermo Men´endez Turata。“完全计算树逻辑的循环证明系统”。发表于：第*31*届*EACSL*计算机科学逻辑年会（*CSL 2023*）。第252卷。莱布尼茨国际信息学会议论文集（LIPIcs）。即将出版。德国达格施图尔：Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik，2023年。[3] Bahareh Afshari 和 Graham E. Leigh。“模态 μ-演算的无 Cut 完备性”。发表于：*2017*年第*32*届*ACM/IEEE*计算机科学逻辑年会（*LICS*）。2017年第32届ACM/IEEE计算机科学逻辑年会（LICS）。2017年6月，第1–12页。[4] Bahareh Afshari 和 Graham E. Leigh。“模态 μ演算的林登插值”。发表于：¨语言、逻辑与计算。Ayb¨uke Ozg¨un 和 Yulia Zinova 编。计算机科学讲义。Cham：Springer 国际出版公司，2022年，第197–213页。[5] Bahareh Afshari、Graham E. Leigh 和 Guillermo Men´endez Turata。“循环证明中的一致插值：模态 μ-演算的情形”。发表于：基于解析式语义树及相关方法的自动推理。第12842卷。丛书名称：计算机科学讲义。Cham：Springer 国际出版公司，2021年，第335–353页。[6] Bahareh Afshari 和 Dominik Wehr。“抽象循环证明”。发表于：逻辑、语言、信息与计算。Agata Ciabattoni、Elaine Pimentel 和 Ruy J. G. B. de Queiroz 编。计算机科学讲义。Cham：Springer 国际出版公司，2022年，第309–325页。[7] David Baelde、Amina Doumane、Denis Kuperberg 和 Alexis Saurin。“用于循环与非良基证明的弹跳线程：迈向循环证明的组合性”。发表于：第*37*届*ACM/IEEE*计算机科学逻辑年会论文集。LICS '22。美国纽约：美国计算机协会，2022年8月4日，第1–13页。[8] James Brotherston。“归纳定义的相继式演算证明系统”。接收日期：2006-10-31T15:56:10Z。博士论文。爱丁堡大学，2006年11月。[9] Anupam Das。“G¨odel 的 T 的循环版本及其抽象复杂度”。发表于：*arXiv:2012.14421 [*计算机科学、数学*]*（2021年1月16日）。[10] Kurt G¨odel。“¨关于有限观点的一种迄今尚未使用过的扩展”。发表于：*Dialectica* 12.3（1958）。出版商：Wiley-Blackwell，第280–287页。[11] Emile Hazard 和 Denis Kuperberg。“超限表达式的循环证明”。发表于：第*30*届*EACSL*计算机科学逻辑年会（*CSL 2022*）。Florin Manea 和 Alex Simpson 编。第216卷。莱布尼茨国际信息学会议论文集（LIPIcs）。ISSN：1868-8969。德国达格施图尔：Schloss Dagstuhl – Leibniz-Zentrum f¨ur Informatik，2022年，23:1–23:18。[12] Natthapong Jungteerapanich。“模态 μ-演算的语义树系统”。博士论文。爱丁堡大学，2010年。[13] Mayuko Kori、Takeshi Tsukada 和 Naoki Kobayashi。“HFLN 的循环证明系统”。发表于：第*29*届*EACSL*计算机科学逻辑年会（*CSL 2021*）。Christel Baier 和 Jean Goubault-Larrecq 编。第183卷。莱布尼茨国际信息学会议论文集（LIPIcs）。ISSN：1868-8969。德国达格施图尔：Schloss Dagstuhl–Leibniz-Zentrum f¨ur Informatik，2021年，29:1–29:22。

[14] “Safra 构造”。载于：计算理论。Dexter C. Kozen 编。计算机科学文本丛书。伦敦：Springer，2006 年，第 167–170 页。[15] Johannes Marti 和 Yde Venema。《用于无交替 μ-演算的 Focus 系统》。载于：基于解析式语义表及相关方法的自动推理。Anupam Das 和 Sara Negri 编。计算机科学讲义。Cham：Springer 国际出版公司，2021 年，第 371–388 页。[16] Damian Niwi´nski 和 Igor Walukiewicz。“ μ-演算博弈”。载于：理论计算机科学163.1（1996 年 8 月 30 日），第 99–116 页。[17] N. Piterman 和 A. Pnueli。“更快地求解 Rabin 博弈和 Streett 博弈”。载于：第 *21* 届 *IEEE* 计算机科学逻辑年会（*LICS'06*）。第 21 届 IEEE 计算机科学逻辑年会（LICS'06）。美国华盛顿州西雅图：IEEE，2006 年，第 275–284 页。[18] Nir Piterman。“从非确定性 B¨uchi 自动机和 Streett 自动机到确定性奇偶自动机”。载于：计算机科学中的逻辑方法第 3 卷，第 3 期（2007 年 8 月 14 日）。出版方：Episciences.org。[19] John C. Reynolds。《程序设计语言理论》。剑桥：剑桥大学出版社，1998 年。[20] Jan Rooduijn。“带主模态的若干模态逻辑的循环超序列演算”。载于：基于解析式语义表及相关方法的自动推理。Anupam Das 和 Sara Negri 编。计算机科学讲义。Cham：Springer 国际出版公司，2021 年，第 354–370 页。[21] S. Safra。“关于 omega-自动机的复杂性”。载于：第 *29* 届计算机科学基础年会论文集。SFCS '88。美国：IEEE 计算机学会，1988 年 10 月 24 日，第 319–327 页。[22] Alex Simpson。“循环算术等价于皮亚诺算术”。载于：软件科学与计算结构基础。Javier Esparza 和 Andrzej S. Murawski 编。计算机科学讲义。柏林、海德堡：Springer，2017 年，第 283–300 页。[23] Christoph Sprenger 和 Mads Dam。“带显式近似的 μ-演算中的全局归纳机制”。载于：*RAIRO*——理论信息学与应用 37.4（2003 年 10 月）。出版方：EDP Sciences，第 365–391 页。[24] Christoph Sprenger 和 Mads Dam。“归纳推理的结构： μ演算中的循环证明与树形证明”。载于：软件科学与计算结构基础。Andrew D. Gordon 编。计算机科学讲义。柏林、海德堡：Springer，2003 年，第 425–440 页。[25] Colin Stirling。“带名称的模态 μ 演算证明系统”。载于：理论计算机科学电子论文集129（2013 年 9 月 19 日），第 18–29 页。[26] Dominik Wehr。《循环推导分析的抽象框架》。理学硕士论文。阿姆斯特丹大学，2021 年 8 月 26 日。