arXiv:2601.13991v1 [cs.PL] 20 Jan 2026

生成函数与占用测度：概率循环的不变式合成（扩展版）

Darion Haase(✉)[1] 、Kevin Batz[2,3], Adrian Gallus[4],、Benjamin Lucien Kaminski[5,2], 、Joost-Pieter Katoen[1],、Lutz Klinkenberg[1],和 Tobias Winkler[1]

[1] 亚琛工业大学，亚琛，德国 {darion.haase, katoen, lutz.klinkenberg, tobias.winkler}@cs.rwth-aachen.de[2] 伦敦大学学院，伦敦，英国[3] 康奈尔大学，伊萨卡，纽约州，美国 k.batz@ucl.ac.uk[4] 独立研究人员，亚琛，德国 adrian.gallus@rwth-aachen.de[5] 萨尔兰大学，萨尔布吕肯，德国 kaminski@cs.uni-saarland.de

摘要。 概率程序设计中的一项基本计算任务，是根据给定的初始（先验）分布推断程序的输出（后验）分布。这个问题颇具挑战性，尤其是对于包含循环或无界递归的表达能力较强的语言。现有文献大多聚焦于统计近似，而本文研究数学意义上的精确推断。为了实现对包含循环的程序进行精确推断，我们依赖一种相对较少被探索的概率循环不变式，该不变式与循环的所谓占用测度相关。占用测度将程序状态与给定初始分布时对这些状态的预期访问次数联系起来。在此基础上，我们提出了占用不变式的概念。这类不变式本质上与概率鞅对偶，而概率鞅是文献中形式化概率循环分析的主流技术。占用不变式的一个关键特征是，它们能够将初始分布纳入考量，并且通常可以作为副产品给出几乎必然正终止的证明。最后，我们提出了一种针对占用不变式的自动化、基于模板的不变式合成方法，将其编码为生成函数。该方法已实现，并在一组基准测试上进行了评估。

1 引言

概率程序（PPs）类似于普通程序，但增加了抛硬币，或更一般地说，从概率分布中采样值的能力。PPs 在现代计算中无处不在；例如，它们出现在随机化

/// X
当($x = 1$){
  {$c := c + 1$} [½] {$x := 0$}
}
/// $\frac{1}{2} + \frac{1}{4}C + \frac{1}{8}C^2 + \frac{1}{16}C^3 \ldots$

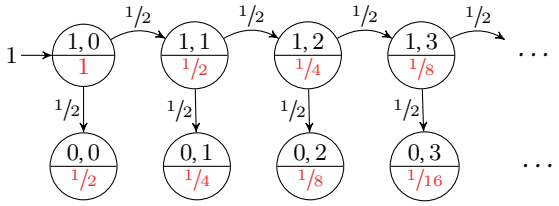

图1：**左：** 一个概率循环，其标注包含初始分布和最终分布。标注使用概率生成函数（PGF），其中未定元表示程序变量（例如，X 对应于 x），指数表示取值（取值为 0 的变量会被省略，因为 $X^0 = 1$；指数为 1 时省略指数，因为 $X^1 = X$），系数表示概率（系数为 1 时也会省略）[33]。例如，PGF X 表示狄拉克分布，其中以概率 1，x 的取值为 1，且 c 的取值为 0。

右图：程序的马尔可夫链表示（转换模拟整个循环迭代）。每个状态的上半部分是变量赋值 x、c。这些红色数字构成了占用测度（预期访问次数）。

算法 [27], 随机采样 [40], 统计推断程序 [10, 21, 8],认知科学 [22], 和自主系统 [41]。

概率程序作为测度变换器。直观而言，在给定输入上运行概率程序会产生一个概率分布[6]，其涵盖可能的输出。近半个世纪前，Kozen [36] 将这一直觉转化为形式化（指称）程序语义，为每个程序 C 关联一个函数[7]

$$\overbrace{\mathsf{K}\,[\![C]\!]}^{\text{Kozen 的程序 } C\text{测度变换器语义}}\colon \quad \underbrace{\mathsf{M}(\text{状态集})}_{\text{输入状态上的测度}} \to \overbrace{\mathsf{M}(\text{状态集})}^{\text{关于输出状态的测度}}。$$

本文仅限于离散的关于可数状态空间上的测度状态。例如，图1将状态 $[x \mapsto 1, c \mapsto 0]$ 上的初始点质量分布转换为一个分布，其中 c 服从参数为 ½ 的几何分布（并且 $x = 0$ 以概率1）。注意，图1 使用了生成函数（GF）记法来表示分布。生成函数在我们的论述中发挥着重要作用，稍后我们将在第4页进行解释。

我们的目标：循环推断——精确且自动。刻画程序的精确输出分布对许多应用都很有价值，但通常极具挑战性，尤其是在存在循环的情况下。本文旨在突破精确且自动循环分析的极限：

[6] 或者是次分布，如果程序不以概率为1终止。[7] 注意，$\mathsf{K}\,[\![\cdot]\!]$ 作用于一般的测度，其中（次）概率分布是其特殊情况。

问题陈述： 给定一个离散概率循环$\texttt{while}(\phi)\,\{C\}$，其输入状态上的分布为 μ，自动计算精确输出分布的表示$\mathcal{K}\,[\![\texttt{while}(\phi)\,\{C\}]\!](\mu)$。

关于我们的问题陈述，需要说明两点：（1）由于 while 循环的标准不可判定性结果，不存在完整的算法解决方案。因此，我们提供了涵盖许多实例的启发式方法。（2）由于最终分布可能具有无限支撑（如图1所示），我们采用闭式生成函数作为概率分布的简洁编码 [33]（详情如下）。

使用 *Kozen* 的定义进行自动推理——不可行？Kozen [36]通过函数式最小不动点方程[8]刻画循环的语义

$$\mathcal{K}\,[\![\texttt{while}\,(\phi)\,\{C\}]\!] \;=\; \mathcal{K}\,[\![\texttt{while}\,(\phi)\,\{C\}]\!] \circ \mathcal{K}\,[\![C]\!] \circ [\![\phi]\!] \;+\; [\![\neg\phi]\!]\,, \quad (\dagger)$$

其中 $[\![\phi]\!]$ 和 $[\![\neg\phi]\!]$ 分别根据 ϕ 和 $\neg\phi$对输入分布进行筛选。该方程体现了这样一种直觉：循环不受将其展开一次的影响。然而，（†）与我们的问题陈述之间存在差异：我们的目标是，对于给定的初始分布 μ，确定$\mathcal{K}\,[\![$，使得$(\phi)C\}]\!](\mu)$成立；然而，（†）并不依赖于这样的 μ。换言之，基于求解（†）的方法需要同时对所有可能的初始分布进行推理（见 [33, 13, 34, 32]），而我们只关心函数方程在一个点处的解，即初始分布 μ处的解。

占用测度来救场。 Sharir、Pnueli 和 Hart 提出了一种等价但似乎鲜为人知的循环输出分布刻画方法 [49]。他们的方法将初始分布纳入考量，并研究由此诱导的随机过程的所谓占用测度（例如，[47]）。占用测度将每个程序状态与其在循环头处的预期访问次数关联起来。要了解占用测度如何帮助确定循环后的最终分布，以下几点观察至关重要：

– 对于满足循环守卫的状态，访问次数是$\mathbb{N}\cup\{\infty\}$中的随机值。（与确定性程序不同，概率循环可以多次返回同一状态，而不一定会进入无限循环。）因此，预期访问次数位于区间 $[0,\infty]$ 内。

– 对于违反循环守卫的状态，情况则有所不同：这些状态在循环头处至多被访问一次，因为一旦进入这些状态，循环就会立即终止。因此，此类终止状态 s 的预期访问次数等于

$$\begin{aligned} &0\cdot Pr(s\text{ 永远不会到达}) && + \;1\cdot Pr(s\text{ 到达一次})\\ +\;&\underbrace{2\cdot Pr(s\text{ 到达两次}) \;+\; \cdots}_{=0} && = Pr(s\text{ 到达一次})\,。\end{aligned}$$

[8] 该方程可能有多个解；通常，我们感兴趣的是相对于某个合适序关系的最小解（参见第4.2节）。[9] 除此之外，[49],据我们所知，还没有其他关于概率程序占用测度的研究。

因此，访问 s 的预期次数恰好等于曾经到达 s 的概率，也就是循环在 s处终止的概率。

总之，若将占用测度记为OM，则有

$$\mathcal{K}[\![\texttt{while}\,(\phi)\,\{C\}]\!](\mu) \;=\; [\![\neg\phi]\!](OM)\,; \qquad \text{(as formally proved in Theorem 2)}$$

也就是说，通过将OM 限制在不满足守卫条件的状态上 ϕ，我们得到循环$\mathcal{K}$ $[\![\texttt{while}\,\phi)\,\{C\}]\!](\mu)$ 的最终分布 $\texttt{while}(\phi)\,\{C\}$。

我们运行示例程序的占用测度通过图1中的程序马尔可夫链展开图示出。底行的状态是不满足循环守卫的状态。可以看到，它们的预期访问次数确实构成了预想的几何分布。

利用占用测度推理循环。可以证明（定理3），循环$\texttt{while}(\phi)\,\{C\}$ 在初始分布 μ 下的占用测度是固定点方程的最小解OM

$$OM \quad = \quad \mu \;+\; \mathcal{K}[\![C]\!]([\![\phi]\!](OM))\,. \qquad (\ddagger)$$

请注意，与（†）不同，这个不动点方程取决于初始分布μ。方程（‡）启发了以下猜测并检验启发式方法：

1. 猜测一个候选解 M （‡）——可以将其视为一个循环不变量。2. 检查该候选 M 是否确实是一个解。如果是，那么 M 就是占用测度OM 的（逐点）上界，因此，根据前面的论证，$\mathrm{K}\,[\![\text{当}(\phi)\,\{C\}]\!](\mu) \le [\![\neg\phi]\!](M)$。

我们可以利用占用测度的另一个关键特性：其总质量等于循环的期望运行时间[10]，输入为 μ [49]。因此，任何满足（‡）的有限测度M 都能证明该循环在输入 μ时正概率几乎必然终止（PAST）。由于终止循环会保留其输入测度的概率质量，我们可以得出以下结论：如果有限测度 M 满足（‡），且 $[\![\neg\phi]\!](M)$ 是一个概率分布（即总质量为 1），那么步骤 2 中的不等式 $\mathrm{K}\,[\![\text{while}\,(\phi)\,\{C\}]\!](\mu) \le [\![\neg\phi]\!](M)$ 就变为精确等式 $\mathrm{K}\,[\![\text{while}\,(\phi)\,\{C\}]\!](\mu) = [\![\neg\phi]\!](M)$。详情参见定理 6。

利用闭式生成函数进行有效验证与综合。前述段落中的步骤（1）和（2）仍需处理定义在无限程序状态集合上的测度。这正是生成函数的优势所在：例如，图1中循环的占用测度具有无限支撑，其生成函数为

$$OM \quad = \quad 1X \;+\; \tfrac{1}{2}XC \;+\; \tfrac{1}{4}XC^2 \;+\; \ldots \quad + \quad \tfrac{1}{2} \;+\; \tfrac{1}{4}C \;+\; \tfrac{1}{8}C^2 \;+\; \ldots\,,$$

其具有如下（有理）闭式：

$$= \quad \frac{1+2X}{2-C}\,.$$

[10] 我们将期望运行时间定义为循环头处的预期访问次数。

在适当的假设下，可以针对以这种闭式表示的候选测度，以算法方式验证方程（‡），即实现猜测与检验启发式的步骤（ M 2）。基于此，我们进一步证明，步骤（1）可以通过基于模板的综合来实现，使用参数化的有理生成函数作为模板（详见第6节）。

贡献。我们重新审视占用测度在概率循环分析中鲜为人知的用途 [49]，并提出以下创新贡献：

1. 我们证明，循环的占用测度可刻画为一个测度变换器的最小不动点；该变换器通过对程序结构进行归纳组合定义（定理3）。2. 我们正式建立了我们的语义视角与 Kozen 语义 [36] （定理2）、操作马尔可夫链上的占用测度（定理3）以及期望运行时间 [30] （定理4）之间的联系。3. 我们提出将生成函数（GF）作为一种有效的编码方式，用于表示定义在非负整数变量上的程序状态占用测度（第6节）。4. 我们提出首个基于有理生成函数模板的占用测度全自动综合（或其近似）的方案。我们通过实验评估了算法的有效性（第7节）。

概述。第2 节介绍测度论和马尔可夫链的预备知识。第3节规定程序语法；第4 节阐述指称语义，并证明其与 Kozen 语义等价。第5节介绍基于占用测度的循环不变量。第6节介绍使用有理生成函数进行符号综合。第7 节提供实现细节和实验结果。第8 节讨论相关工作，第9 节作总结。

2 预备知识

$\mathrm{P}(X)$ 表示 X 的幂集。 $\mathbb{N}$ 是包含 0 的自然数集合。

2.1 测度论

我们简要回顾测度论 [9] 中所需的概念。

定义 1（σ-代数）。设 X 是一个集合。若子集 $\Sigma \subseteq \mathcal{P}(X)$ 满足以下条件，则称其为 σ-代数（在 X上）： (1) $X \in \Sigma$; (2) $A \in \Sigma$ 蕴含 $X \setminus A \in \Sigma$; (3) 对于所有可数集 I, $(A_i)_{i\in I} \in \Sigma^I$蕴含 $\bigcup_{i\in I} A_i \in \Sigma$。

构成 σ-代数的集合称为可测集合。测度为 σ-代数中的每个可测集赋予一个非负实数（或 ∞）：

定义 2（测度与分布）。令 Σ 为定义在 X上的 σ-代数。

– 若函数 $\mu\colon \Sigma \to \mathbb{R}^{\infty}_{\geq 0}$ 满足：对于 Σ 中任意可数族$(A_i)_{i\in I}$两两不交集，都有 $\mu(\bigcup_{i\in I} A_i) = \sum_{i\in I} \mu(A_i)$（这意味着$\mu(\emptyset) = 0$），则称其为测度。我们用 $|\mu| := \mu(X)$表示 μ 的总质量。X 上的测度集合记作 $\mathcal{M}(X)$ （ σ-代数总是可由上下文确定）。

– 若测度 μ 满足 $|\mu| = 1$，则称其为概率分布；若满足 $|\mu| \leq 1$，则称其为次-概率分布。X 上概率分布和次概率分布的集合分别记作 $\mathcal{D}_{=1}(X)$ 和 $\mathcal{D}(X)$。

本文中，我们通常将次概率分布简称为分布。如果 μ 和 ν 是定义在同一 σ-代数上的测度，那么它们的逐点和 $\mu + \nu$ 以及对任意 $c \geq 0$而言的逐点缩放结果 $c \cdot \mu$ 也都是测度。

函数 $f\colon X \to \mathbb{R}^{\infty}_{\geq 0}$，其中 X配备了一个 σ-代数 Σ，若 $f^{-1}((a, \infty]) \in \Sigma$对所有 $a \in \mathbb{R}_{\geq 0}$均成立，则称其为可测。$f$在 $\mu \in$下的期望值$\mathbb{M}(X)$定义为勒贝格积分 $\mathbb{E}_{\mu}[f] := \int f \mathrm{d}\mu$。元素 $x \in X$ 的狄拉克测度 δ_x 以及可测集 $A \in \Sigma$ 的指示函数 $\mathbf{1}_A$ 是

$$\delta_x\colon \Sigma \to \mathbb{R}^{\infty}_{\geq 0},\ B \mapsto \begin{cases} 1, & \text{if } x \in B, \\ 0, & \text{else;} \end{cases} \quad \text{and} \quad \mathbf{1}_A\colon X \to \mathbb{R}^{\infty}_{\geq 0},\ y \mapsto \begin{cases} 1, & \text{if } y \in A, \\ 0, & \text{else.} \end{cases}$$

我们通常用 X上的谓词 ϕ_A 来表示可测集 $A \in \Sigma$；在这种情况下，*Iverson* 括号 $[\phi_A]$ 表示指示函数 $\mathbf{1}_A$。我们用 $[\phi_A] \cdot \mu$ 表示测度 μ 限制在 A上的结果，即$([\phi_A] \cdot \mu)(B) := \mu(A \cap B)$，其中 $B \in \Sigma$。

在本文中，X 通常是一个可数集合，其所有子集的集合 $\mathrm{P}(X)$ 构成一个 σ-代数。因此，我们通过写作 $\mu(x) = \mu(\{x\})$（其中 $x \in X$），将测度 $\mu \in \mathbb{M}(X)$ 与函数 $X \to \mathbb{R}^{\infty}_{\geq 0}$ 等同起来。在这种情形下，每个函数 $f\colon X \to \mathbb{R}^{\infty}_{\geq 0}$ 都是可测的，勒贝格积分也就变成了可数求和：$\mathbb{E}_{\mu}[f] = \sum_{x \in X} f(x) \cdot \mu(x)$。测度集合 $\mathbb{M}(X)$ 按照 $\mu \leq \nu$ 进行偏序，其中 $\mu(x) \leq \nu(x)$ 对所有 $x \in X$ 均成立；该序使 $(\mathbb{M}(X) \leq)$ 成为一个完备格（也就是说，$\mathbb{M}(X)$ 的每个子集相对于 $\leq$都有下确界和上确界）。

2.2 马尔可夫链

马尔可夫链描述了这样一种随机过程：其转移概率只取决于当前状态，而不考虑任何先前的历史 [14]。

定义 3（马尔可夫链）。A 马尔可夫链是一个二元组 $C = (S_C, P_C)$，其中S_C 是一个由状态组成的可数集，而 $P_C\colon S_C \to \mathcal{D}_{=1}(S_C)$ 是转移概率函数。对于 $s, t \in S_C$，我们记 $P_C[s](t) := P_C(s(t))$。如果上下文已明确指明马尔可夫链，我们就省略下标 C。

马尔可夫链在其无限路径集合上诱导出一个测度：

定义 4（无限路径）。设 C 是一个马尔可夫链。C中的一条无限路径是一个序列 $\pi = \pi_0 \pi_1 \ldots \in S_C^{\omega}$，其中 $P_C[\pi_i](\pi_{i+1}) > 0$，对任意 $i \in \mathbb{N}$均成立。路径$_C$表示 C中所有无限路径组成的集合（通常不可数）。

对于一个有限序列 $\bar{\pi} \in S_C^*$，柱集$\mathrm{Cyl}(\bar{\pi}) := \{\pi \in$路径$_C \mid \exists \pi' \in$路径$_C\colon \pi = \bar{\pi}\pi'\}$由 C 中所有以前缀 $\bar{\pi}$开头的无限路径组成。路径$_C$ 配备由所有柱集生成的 σ-代数；详情请参见 [14]。

定义5（马尔可夫链诱导的（概率）测度）。设C 为一条马尔可夫链。给定一个初始测度 $\iota \in \mathcal{M}(S_C)$，$C$ 会在路径$^{\iota}$ $\bar{\pi}_0 \quad \prod_{i=0}^{|\bar{\pi}|-1} P_C[\bar{\pi}_i]\ \bar{\pi}_{i+1}$. 上诱导

出唯一测度$\Pr^{\iota}_C$，满足$\Pr^{\iota}_C(\mathrm{Cyl}()) := |\iota|$，且对于 $\bar{\pi} \in S_C^+$，$\Pr^{\iota}_C(\mathrm{Cyl}(\bar{\pi}))\colon() \cdot ()$

注意，Pr^{ι}_{C} 在定义5 中是概率测度，当且仅当 ι 是概率分布。不过，我们的框架要求对任意测度Pr^{ι}_{C} 进行定义， $\iota \in \mathcal{M}(S_C)$包括质量为 $|\iota| < 1$ 和 $|\iota| > 1$的测度。

现在我们可以定义占用测度，这是本文的一个关键概念：

定义 6（占用测度）。设 C 为一个初始测度为 $\iota \in \mathcal{M}(S_C)$的马尔可夫链。我们将占用测度$\mathrm{OM}^{\iota}_{C} \in \mathcal{M}(S_C)$ 定义为$\mathrm{OM}^{\iota}_{C}(s) :=\mathbb{E}_{\mathsf{Pr}^{\iota}_{C}}\left[\mathsf{occ}^{s}_{C}\right]$，对于每个状态 $s \in S_C$，它给出可测函数occ^{s}_{C}：路径$_C \to \mathbb{R}^{\infty}_{\geq 0}, \pi \mapsto \sum_{i\in\mathbb{N}} [\pi_i = s]$的期望值。

直观而言，occ^{s}_{C} 统计状态 s 在 C的一条路径中出现的次数。更一般的占用测度定义 [47] 会包含一个停时 T，用于规定何时停止计数；而在我们的情形中， $T = \infty$。对于给定的初始分布 $\iota \in \mathrm{D}(S_C)$， $\mathrm{OM}^{\iota}_{C}(s)$是 C到达 s的访问总次数的期望值。

3 概率程序的语法

本节介绍一种离散的 概率守卫命令语言（pGCL），其风格遵循 McIver 和 Morgan [42]。令 $\mathrm{Vars} \neq \emptyset$ 为一个有限的（程序）变量集，其元素记作 x、 y 等。程序状态 s: $\mathrm{Vars} \to \mathbb{N}$ 将自然数 $s(x) \in \mathbb{N}$赋给每个变量 $x \in$Vars。我们将满足以下条件的更新后状态记作 $s[x/n]$： $x \in$ Vars 映射到 $n \in \mathbb{N}$，而其他每个变量 $x' \neq x$ 都映射到$s(x')$。我们用 $\mathrm{States} := \mathrm{Vars} \to \mathbb{N}$ 表示所有程序状态构成的可数集。

定义 7（pGCL）。pGCL程序集遵循以下文法：

$C \to$空操作	| 发散	（无副作用的 | 无限循环）
| $x := E$	| $x :\approx \mu$	(赋值 | 概率赋值)
| $\{C\}\ [p]\{C\}$	| $C \,;\, C$	(概率选择 | 顺序组合)
| 如果 $(\phi)\ \{C\}$ 否则$\{C\}$	| 当 $(\phi)\ \{C\}$	（条件分支 | *while* 循环）

其中 $x \in$变量集， E： 状态 $\to \mathbb{N}$ 是算术表达式， μ： 状态 $\to \mathcal{D}_{=1}(\mathbb{N})$是分布表达式， $p \in [0,1] \cap \mathbb{Q}$，而 $\phi \subseteq$ 状态 是谓词。

程序 C 称为 无循环，如果其中既不包含 `while` 也不包含 发散 语句。出于实际验证的目的，定义 E中的 's、 μ中的 's，以及ϕ中的 's，其具体语法将在第 7节介绍，见第 6节。此处暂不要求这种语法，因此将其省略。

我们简要回顾定义 7 中的每种程序语句。空操作不执行任何操作。发散是循环 while (true) {skip} 的简写。赋值语句有两种类型：确定性赋值 $x := E$ 根据 $E(s)$ 更新当前程序状态 s 中 x 的值，而概率赋值 $x :\approx \mu$ 从概率分布 $\mu(s)$ 中采样一个值，并将结果赋给 x。概率选择 $\{C_1\}\ [p]\ \{C_2\}$ 以 p 的概率执行 C_1，并以 $1-p$ 的概率执行 C_2。最后， $C_1 \,;\, C_2$、if $(\phi)\ \{C_1\}$ else $\{C_2\}$ 和 while $(\phi)\ \{C\}$ 分别是标准的顺序组合、条件分支和 while 循环。

表 1：程序 C 相对于初始测度 g 的后验测度的归纳定义：基于占用测度（列 $\mathrm{pm}[\![C]\!](g)$）以及按照 Kozen 的语义 [36,Semantics 2]（列 $\mathrm{K}\ [\![C]\!](g)$）。值得注意的是，这两个定义仅在循环的处理方式上有所不同；具体而言，对于 pm，不动点变量 (ν) 的类型为 M（状态集），而对于 K，不动点变量 (T) 的类型为 M（状态集）$\to$M（状态集）。在定理 2 中，我们证明这两个定义是等价的。

C	下午$[\![C]\!]\,(g)$	$\mathrm{K}\ [\![C]\!](g)$
skip	g	
发散	0	
$x :=_E$	$\lambda s.\ \sum_{s'}\ [s'[x/E(s')] = s] \cdot g(s')$	
$x :\approx \mu$	$\lambda s.\ \sum_{s'}\ [s'[x/s(x)] = s] \cdot\ \mu(s')(s(x)) \cdot g(s')$	
$\{C_1\}\ [p]\ \{C_2\}$	$\mathrm{pm}[\![C_1]\!]\,(p \cdot g) + \mathrm{pm}[\![C_2]\!]\,((1-p) \cdot g)$	$\mathrm{K}\ [\![C_1]\!]\,(p \cdot g) + \mathrm{K}\ [\![C_2]\!]\,((1-p) \cdot g)$
$C_1 \fatsemi C_2$	下午$[\![C_2]\!]\,($下午$[\![C_1]\!]\,(g)\ )$	$\mathrm{K}\ [\![C_2]\!]\,(\mathrm{K}\ [\![C_1]\!](g))$
如果 $(\phi)\ \{C_1\}$ 否则 $\{C_2\}$	$\mathrm{pm}[\![C_1]\!]\,([\phi] \cdot g) + \mathrm{pm}[\![C_2]\!]\,([\neg\phi] \cdot g)$	$\mathrm{K}\ [\![C_1]\!]\,([\phi] \cdot g) + \mathrm{K}\ [\![C_2]\!]\,([\neg\phi] \cdot g)$
当$(\phi)\ \{C_0\}$	$[\neg\phi] \cdot$ (最小不动点 $\nu.\ g +$ 概率测度$[\![C_0]\!]\,([\phi] \cdot \nu)$)	$\Big(\mathrm{lfp}\,T.\ \lambda\nu.\,[\neg\phi] \cdot\ \nu + T(\mathrm{K}\ [\![C_0]\!]\,([\phi] \cdot\ \nu))\Big)(g)$

4 基于占用测度的概率程序语义

现在，我们介绍基于占用测度的语义，并将其与概率程序中其他几种成熟的语义概念联系起来。具体而言，我们在第 4.1 节介绍关键的语义泛函，在第 4.2 节证明它与 Kozen 语义等价，并在第 4.3 节将该语义泛函与操作马尔可夫链及期望运行时间联系起来。

4.1 基于占用测度的循环语义

为了定义我们的语义泛函，我们对测度逐点定义算术运算，即对于 $\circledcirc \in \{+, \cdot\}$，令 $f \circledcirc g = \lambda s. f(s) \circledcirc g(s)$。此外，回顾一下，$[\phi]$ 是谓词ϕ的 Iverson 括号（即指示函数）。现在考虑以下内容：

定义 8（pm）。函数$\mathrm{pm}[\![C]\!]$: M（状态）$\to$ M（状态）根据 C 的结构，按照表*1*（中间列）中的规则归纳定义。我们称$\mathrm{pm}[\![C]\!]\,(g)$ 为 C 相对于初始测度 g的后验测度。

如果初始测度 g 是概率分布，那么对后验测度$\mathrm{pm}[\![C]\!](g)$ 的直观理解简单明了，也符合预期：

> 如果 g 是概率分布，那么$\mathrm{pm}[\![C]\!](g)$ 就是执行[11]后得到的状态次概率 C 分布，初始分布为 g。

现在让我们逐一讲解表1中间列的每条规则。空操作skip语句不会修改初始测度。由于 发散 在任何状态下都不会终止，

[11] 泄漏的质量 $|g| - \big|\mathrm{pm}[\![C]\!](g)\big|$ 表示 C 在 g上发散的概率。

后验测度始终为 0。对于赋值，状态 s 中的后验测度质量，是通过累加所有状态 s' 的质量得到的；这些状态在执行赋值后会变为 s。对于确定性赋值 $x := E$，这些状态是所有满足 $E(s') = s(x)$ 且 s' 与 s 在其他所有变量上的取值都相同的状态 s。类似地，对于概率赋值 $x :\approx \mu$，输入质量 $g(s')$ 会乘以从 $\mu(s')$中采样到值 $s(x)$ 的概率。顺序组合 $C_1 \mathbin{;} C_2$ 先通过 C_1 传递测度，再通过 C_2 传递。对于条件分支 $\mathsf{if}\,(\phi)\,\{C_1\}\,\mathsf{else}\,\{C_2\}$，测度会被限制在满足 ϕ 的状态和满足 $\neg\phi$的状态上，然后分别通过 C_1 和 C_2传递。最终结果是执行两个分支所得测度之和。

循环的后验测度 $C = \text{当}\,(\phi)\,\{C_0\}$ 涉及最小不动点构造。占用测度正是在这一构造中发挥作用——我们将在第 4.3 节将这一事实形式化并加以证明。下面定义一些辅助记号。给定一个初始测度 g，我们称

$$\Phi_{g,C} \colon \mathcal{M}(\mathsf{States}) \to \mathcal{M}(\mathsf{States}), \qquad \nu \;\mapsto\; g + \mathbf{pm}[\![C_0]\!]\,([\phi]\cdot\nu)$$

C 关于 g 的特征函数。注意，根据定义，$\mathbf{pm}[\![C]\!](g) = [\neg\phi]\cdot(\mathrm{lfp}\,\nu.\,\Phi_{g,C}(\nu))$。若 g 是概率分布，则 $(\mathrm{lfp}\,\nu.\,\Phi_{g,C}(\nu))(s)$ 表示以初始测度 g 执行时，状态 s 在循环头处被遇到的期望次数（如定理 3 所形式化）——因此，最小不动点表示一个占用测度。然后，姑且这么说，将 $\mathrm{lfp}\,\nu.\,\Phi_{g,C}(\nu)$ 中所有满足循环守卫条件的状态剔除后，我们恰好得到执行 C 于 g 后到达的终态（次）分布（参见定理 2）。

示例 *1*。 程序 C 中的循环 1生成几何分布。在每次迭代中，循环体 $C_0 = \{c := c+1\}\ [1/2]\ \{x := 0\}$ 以 计数器变量 c 以概率 $1/2$, 递增，或通过将其设置 x 为零来终止循环。对于初始测度 $g = [x = 1 \wedge c = 0]$，我们有

程序 1 几何循环。

```
/// X   /* 初始时，x = 1 且 c = 0 */
当(x = 1){
  {c := c + 1} [1/2] {x := 0}
}
/// 1/2 + 1/4 C + 1/8 C^2 + ...
```

$$\mathrm{lfp}\,\Phi_{g,C} \;=\; \sum_{k=0}^{\infty} 2^{-k}\cdot[x = 1 \wedge c = k] + \sum_{k=0}^{\infty} 2^{-(k+1)}\cdot[x = 0 \wedge c = k]\ .$$

对于每个 $s \in$ 状态，值 $(\mathrm{lfp}\,\Phi_{g,\,C})(s)$ 表示在状态 g 上执行 C 时，在循环头遇到 s的预期次数。[12] 这很直观：任何满足 $s(x) \notin \{0,\ 1\}$ 的状态 s 都不会 被访问，因为我们在 $x = 1$ 上执行循环，而x只能被设为 0。所有满足 $s(x) \in \{0,\ 1\}$ 的状态最多只会被访问一次。因此，对于这类状态，$(\mathrm{lfp}\,\Phi_{g,\,C})(s)$ 与到达 s 的概率相同。

[12] 我们将展示如何使用示例 6 中的定理 3获得这个不动点。

对 $x \neq 1$ 进行筛选后，我们得到预期的几何分布：

$$\mathbf{pm}[\![C]\!]\,(g) \;=\; [x \neq 1] \cdot (\mathrm{lfp}\ \Phi_{g,C}) \;=\; \sum_{k=0}^{\infty} 2^{-(k+1)} \cdot [x = 0 \wedge c = k]\ .$$

值得注意的是，尽管 g 和 $\mathbf{pm}[\![C]\!](g)$ 都是概率分布，但不动点是一个质量为 $\left|\text{最小不动点}\ \Phi_{g,C}\right| = \sum_{k=0}^{\infty} 2^{-k} + \sum_{k=0}^{\infty} 2^{-(k+1)} = 3 > 1$的测度，这表明在 pm 的定义中需要考虑任意测度。

我们构造的最小不动点是良定义的，这一点可由 Tarski 定理 [51] 得出：测度 $\mathcal{M}$(状态) 配备逐点序 $\leq$ 后构成完备格，而 $\mathbf{pm}[\![C]\!]$ (以及 $\Phi_{g,\ C}$) 是该格上的单调函数。我们将在下一个定理中汇总后一事实及其他若干性质，这些性质可由其与 Kozen 语义的等价性（定理 2）以及 [36] 得出：

定理 1（pm的性质）。设 $C \in$pGCL。则：

1. $\mathbf{pm}[\![C]\!]$ 是单调的，即对于所有 $\mu,\ \nu \in \mathrm{M}$（状态），都有
$$\mu \leq \nu \quad \mathit{implies} \quad \mathbf{pm}[\![C]\!]\,(\mu) \leq \mathbf{pm}[\![C]\!]\,(\nu)\ .$$
2. $\mathbf{pm}[\![C]\!]$ 是线性的，即对于所有 $\mu,\ \nu \in \mathrm{M}$（状态）以及所有 $a \in \mathbb{R}^{\infty}_{\geq 0}$，都有
$$\mathbf{pm}[\![C]\!]\,(\mu+\nu) = \mathbf{pm}[\![C]\!]\,(\mu) + \mathbf{pm}[\![C]\!]\,(\nu) \quad \mathit{and} \quad \mathbf{pm}[\![C]\!]\,(a \cdot \mu) = a \cdot \mathbf{pm}[\![C]\!]\,(\mu)\ .$$
3. $\mathbf{pm}[\![C]\!]$ 是可行的，即对于所有 $\mu \in \mathrm{M}$（状态），都有 $\left|\mathbf{pm}[\![C]\!]\ (\mu)\ \right| \leq |\mu|$。
4. $\mathbf{pm}[\![C]\!]$ 保持分布不变，即对于所有 $\mu \in \mathrm{M}$（状态），都有
$$\mu \in \mathcal{D}\,(\mathsf{States}) \quad \mathit{implies} \quad \mathbf{pm}[\![C]\!]\,(\mu) \in \mathcal{D}\,(\mathsf{States})\ .$$

4.2 与 Kozen 语义等价

对于每个程序 C，Kozen 的 [36] 测度变换器语义 $\mathcal{K}\ [\![C]\!]$ 都根据 C 的结构，按照表 1 （右栏）中的规则归纳定义。 pm 和$\mathcal{K}$ 的定义完全相同，唯独循环的情形除外。然而，这一区别至关重要：而pm的循环特征泛函类型为M（状态） $\to$ M（状态），也就是说，它是一个测度变换器；Kozen 的不动点泛函则属于更高阶类型

$$\big(\mathcal{M}\,(\mathsf{States}) \to \mathcal{M}\,(\mathsf{States})\big) \;\to\; \big(\mathcal{M}\,(\mathsf{States}) \to \mathcal{M}\,(\mathsf{States})\big)\ ,$$

也就是说，一个测度变换器的变换器。这是因为 Kozen 的构造不考虑初始测度 g：$\mathrm{lfp}\,T.\,\lambda\nu.\,[\neg\phi]\cdot\nu + T(\mathrm{K}\ [\![C_0]\!]\ ([\phi]\cdot\ \nu))$ 是一个类型为 M（状态集） $\to$ M（状态集）的函数，它将每个初始测度映射到相应的后验测度。正是因为我们的函数考虑了初始测度，基于不变式的推理才大为简化，我们提出的新颖的基于模板的不变式合成方法也因此成为可能。简言之：Kozen 的方法需要寻找一个高阶对象，而我们只需一个低阶对象。

现在我们陈述并证明本节的主要结果：

定理 2（语义等价）。对于所有程序 C，$\mathbf{pm}[\![C]\!] = \mathcal{K}\,[\![C]\!]$。

因此，我们基于占用测度的 pm变换器确实能够可靠地确定所需的后验测度，尽管循环的泛函远比 Kozen 语义中涉及的泛函简单。本节余下部分将用于证明定理 2。

证明（定理*2*）。关键思路是利用*Kozen* 对偶性在测度变换器与所谓的期望变换器 [35]之间：给定一个随机变量 f：状态 $\to \mathbb{R}^{\infty}_{\geq 0}$，记

$$\mathbf{wp}[\![C]\!]\,(f) : (\mathsf{States} \to \mathbb{R}^{\infty}_{\geq 0}), \qquad s \mapsto \mathbb{E}_{\mathcal{K}\,[\![C]\!]\,(\mathbf{1}_s)}\,[f]$$

为关于 f的最弱前期望 C，也就是说，该函数将每个初始程序状态 s 映射为执行 C于 s后到达的最终状态分布 $\mathrm{K}\ [\![C]\!]\ (\mathbf{1}_s)$ 下 f的期望值。Kozen 对偶性指出，对于每个初始测度 μ，均有

$$\mathbb{E}_{\mu}\left[\mathbf{wp}[\![C]\!]\,(f)\right] = \sum_{s} \mathbf{wp}[\![C]\!]\,(f)\,(s)\cdot\mu(s) = \sum_{s} f(s)\cdot\mathcal{K}\,[\![C]\!]\,(\mu)\,(s) = \mathbb{E}_{\mathcal{K}\,[\![C]\!]\,(\mu)}\,[f]\,.$$

由 [58, 定理 4。5][13] 可立即推出，pm 同样满足

$$\mathbb{E}_{\mu}\left[\mathbf{wp}[\![C]\!]\,(f)\right] = \sum_{s} \mathbf{wp}[\![C]\!]\,(f)\,(s)\cdot\mu(s) = \sum_{s} f(s)\cdot\mathbf{pm}[\![C]\!]\,(\mu)\,(s) = \mathbb{E}_{\mathbf{pm}[\![C]\!]\,(\mu)}\,[f]\,.$$

由此可得，对于所有 $f \in$ 状态$\to \mathbb{R}^{\infty}_{\geq 0}$ 及所有初始测度 μ，均有 $\mathbb{E}_{\mathbf{pm}[\![C]\!]\,(\mu)}\,[f] = \mathbb{E}_{\mathcal{K}\,[\![C]\!]\,(\mu)}\,[f]$。对每个状态 s，定义随机变量 $f_s = \lambda s'.\ [s' = s]$。于是

$$\mathbf{pm}[\![C]\!]\,(\mu)\,(s) \;=\; \mathbb{E}_{\mathbf{pm}[\![C]\!]\,(\mu)}\,[f_s] \;=\; \mathbb{E}_{\mathcal{K}\,[\![C]\!]\,(\mu)}\,[f_s] \;=\; \mathcal{K}\,[\![C]\!]\,(\mu)\,(s)\,,$$

因此 $\mathbf{pm}[\![C]\!] = \mathcal{K}\,[\![C]\!]$，这证明了定理 2。 □

4.3 将占用测度与期望运行时间联系起来

在本节中，我们将形式化地证明，pm循环的定义中[14] 的最小不动点确实是一个占用测度（在适当定义的马尔可夫链中；见定义 6）；我们还将该测度与循环的期望运行时间联系起来。

定义 9（大步操作马尔可夫链）。设 $C = \texttt{while}\,(\phi)\,\{C_0\}$是一个带有无循环体的循环 C_0。我们定义马尔可夫链$\mathcal{L}[\![C]\!]$，其状态为$S_{\mathcal{L}[\![C]\!]}$：= 状态 $\cup\ \{\downarrow\}$ 以及转移概率函数

$$P_{\mathcal{L}[\![C]\!]}(s) \;:=\; \begin{cases} \mathbf{pm}[\![C_0]\!]\,(\delta_s), & \text{if } s \in \mathsf{States} \text{ and } s \models \phi, \\ \delta_{\downarrow}, & \text{else.} \end{cases}$$

[13] Zhang 等人的框架 [58] 需要使用半环($\mathbb{R}^{\infty}_{\geq 0}$，+，·，0，1)进行实例化。[14] 为简单起见，我们仅考虑循环体无循环的循环；若循环体中包含循环，则定义9 及后续定理需要处理内层循环可能发散的情况。

$\mathrm{L}[\![C]\!]$ 的每个状态要么是程序状态，要么是 $\downarrow$，后者表示终止。对于每个满足循环守卫 ϕ的状态 s，转移到某个状态 s' 的概率，等于从 s 经过一次循环迭代到达 s' 的概率。因此，$\mathrm{L}[\![C]\!]$ 中的每次此类转移都对应于完整的循环迭代，故称为大步。基于这一视角，我们得到：

定理 3（pm 与OM的关系）。对于 $C = \text{循环}(\phi)\,\{C_0\}$， C_0 无循环：

$$\forall\, \textit{initial measures}\ g \in \mathcal{M}(\mathsf{States}) : \forall s \in \mathsf{States}: \quad \left(\mathrm{lfp}\,\Phi_{g,C}\right)(s) = \mathrm{OM}^{g}_{\mathcal{L}[\![C]\!]}(s)\ .$$

定理3 将第4.1节中的论断形式化：若 g 是概率测度，则 $\left(\mathrm{lfp}\,\Phi_{g,\,C}\right)(s)$ 是执行 C 于 g时，状态 s在循环头被访问的期望次数。

现在我们来联系循环的占用测度与其期望运行时间。对于每个循环 $C = \text{循环}(\phi)\,\{C_0\}$ 其循环体无循环 C_0，我们定义

$$\mathbf{ert}[\![C]\!] \colon \mathsf{States} \to \mathbb{R}^{\infty}_{\geq 0}, \quad s \quad \mapsto \quad \begin{array}{c} \textit{expected number of loop guard } \phi \\ \textit{evaluations when executing } C \textit{ on } s \end{array}\ .$$

函数 **期望运行时间**$[\![C]\!]$ 可以按照最弱前置条件的形式，通过对 C的结构进行归纳来定义； [30] 详情见此处。我们得到如下对应关系：

定理 4（pm 与期望运行时间的关系）。对于 $C = \text{循环}(\phi)\,\{C_0\}$：

$$\forall\, \textit{initial measures}\ g \in \mathcal{M}(\mathsf{States}) : \qquad \left|\mathrm{lfp}\,\Phi_{g,C}\right| = \sum_{s \in \mathsf{States}} \mathbf{ert}[\![C]\!](s) \cdot g(s)\ .$$

特别是，如果 $g = \delta_s$是一个狄拉克测度，则$\mathrm{lfp}\Phi_{g,C}$的质量等于执行 C 时在 s上守卫条件求值次数的期望。这一观察揭示了它与正概率几乎必然终止(PAST) 的联系——根据定义，对于循环 C 和初始状态 s，若**期望运行时间**$[\![C]\!](s) < \infty$ [30]，则该性质成立。

推论 1. 设 $C = \texttt{while}(\phi)\,\{C_0\}$ 是一个循环，其循环体无循环 C_0 且令$s \in$状态。那么，对于 s， C是 *PAST*，当且仅当对于 $g = \delta_s$，有 $\left|\mathrm{lfp}\,\Phi_{g,C}\right| < \infty$。

5 基于占用测度的不变式

本节介绍占用不变式的概念，它可用于确定或给出循环后验测度的上界，并构成我们不变式合成方法的基础。如果使用（可数）离散时间马尔可夫链作为程序模型，我们的不变式对应于 [49, p. 306], 。

定义 10（占用不变式）。设 $C = \texttt{while}\,(\phi)\,\{C_0\}$，并设 $g \in$M（状态）为初始测度。若测度 $I \in$ M（状态）是 C的（占用）超不变式，关于 g，则称其为超不变式；若 $\Phi_{g,\,C}\,(I) \leq I$。若 $\Phi_{g,\,C}\,(I) = I$，则称 I为（占用）不变式。

请注意，正如基于不变式的推理所预期的那样，确定$\Phi_{g,\,C}(I)$只需对单次带守卫的循环迭代 进行推理。超不变式给出了循环后验测度的上界，如下所示：

定理 5。 若 I 是 $C = \texttt{while}(\phi)\,\{C_0\}$ 关于 g的超不变式，则

$$\mathbf{pm}[\![C]\!]\,(g) \;\;\leq\;\; [\neg\phi]\cdot I\;.$$

证明。 根据 Park 归纳 [46], $\Phi_{g,\ C}(I) \;\leq\; I$蕴含 $\mathrm{lfp}\,v.\ \Phi_{g,\ C}(v) \;\leq\; I$。因此，

$$\begin{array}{lll} & \Phi_{g,C}(I) \;\leq\; I & \\ \text{implies} & \mathrm{lfp}\,v.\ \Phi_{g,C}(v) \;\leq\; I & \text{(Park induction)} \\ \text{implies} & [\neg\phi]\cdot \mathrm{lfp}\,v.\ \Phi_{g,C}(v) \;\leq\; [\neg\phi]\cdot I & \\ \text{implies} & \mathbf{pm}[\![C]\!]\,(g) \;\leq\; [\neg\phi]\cdot I\;. & \text{(definition)} \quad \square \end{array}$$

例 2。 回顾一下几何循环 C，其初始概率分布为 $g = [x = 1 \wedge c = 0]$，来自程序 1。现在我们利用超不变式为 $\mathbf{pm}[\![C]\!]\,(g)$ 求上界。为此，考虑测度

$$I(x,c) \quad = \quad 2^{-(c+1)}\cdot[x=0] \;+\; 2^{-c}\cdot[x=1]\;.$$

我们有 $\Phi_{g,C}(I) \leq I$，因此 $\mathrm{lfp}\,\Phi_{g,C} \leq I$。因此， $I(s)$ 为执行 C 于 g 时在循环头遇到状态 $s \in \mathrm{States}$ 的预期次数提供了上界。这与我们的直觉一致：如果 $s(x) \notin \{0,1\}$，那么就永远不会访问 s，因为我们在 $x = 1$ 上执行循环，而 x 只能被设为 0。另一方面，满足 $s(x) \in \{0,1\}$ 的状态至多被访问一次：在对 c 递增 $s(c)$ 次迭代后，如果 $s(x) = 1$；如果 $s(x) = 0$，则再迭代一次。因此， $I(s)$ 为到达 s 的概率提供了上界。根据定理 5，我们现在如预期得到，

$$\mathbf{pm}[\![C]\!]\,(g) \quad \leq \quad [x \neq 1]\cdot I \quad = \quad 2^{-(c+1)}\cdot[x=0]\;;$$

也就是说，在满足 $s(x) \neq 0$ 的状态 s 中终止的概率为 0；对于所有满足 $s(x) = 0$ 的状态，在 s 中终止的概率至多为 $2^{-(s_{(c)}+1)}$。

如果 I 是质量有限的超不变式，并且其质量与输入测度 g的质量相同，那么我们可以得出一个更强的结论：

定理 6（精确后验测度证明规则）。考虑一个循环 $C = \text{当}(\phi)\,\{C_0\}$，其循环体无循环， C_0 且初始测度为 $g \in \mathcal{M}$ (状态)。设 $I \in \mathcal{M}$(状态) 是满足以下所有条件的测度：

1. I是占用超不变式，即 $\Phi_{g,\ C}(I) \;\leq\; I$。*2.* I 质量有限，即 $|I| < \infty$。*3.* $[\neg\phi]\cdot I$ 的质量等于 g的质量，即 $|[\neg\phi]\cdot I| = |g|$。

那么 $[\neg\phi]\cdot I$ 是精确的后验测度，即 $\mathbf{pm}[\![C]\!]\,(g) = [\neg\phi]\cdot I$。

证明（概要）。根据定理4， $|\mathrm{lfp}\,\Phi_{g,\ C}|$是循环在输入分布 g下的期望运行时间ERT。由于 I 是一个占用超不变式（条件（1 $\Phi_{g\,C} \leq I$ $|\Phi_{g\,C}| \leq$）），由 $|I|$ 2 $= |\Phi_{g\,C}| \leq |I| < \infty$ Park 归纳可得 lfp，因此 lfp，也成立。结合条件（），可得ERTlfp，因此循环的期望运行时间有限。期望运行时间有限意味着几乎必然

终止 [30]。几乎必然终止意味着输入测度 g 的质量等于输出测度的质量，因此 $\left|\mathbf{pm}[\![C]\!](g)\right| = |g|$。现在，由条件（3）可得

$$\left|\mathbf{pm}[\![C]\!](g)\right| = |g| = |[\neg\phi]\cdot I| \ . \tag{††}$$

根据条件（1）和定理 5，我们还可以得到

$$\mathbf{pm}[\![C]\!](g) \leq [\neg\phi]\cdot I \ . \tag{‡‡}$$

反设，存在一个状态 s' 使得 $\mathbf{pm}[\![C]\!](g)(s') < ([\neg\phi]\cdot I)(s')$。那么，

$$\begin{aligned}
|[\neg\phi]\cdot I| &= \left|\mathbf{pm}[\![C]\!](g)\right| && \text{(by (††))}\\
&= \mathbf{pm}[\![C]\!](g)(s') + \sum_{s\neq s'} \mathbf{pm}[\![C]\!](g)(s)\\
&< ([\neg\phi]\cdot I)(s') + \sum_{s\neq s'} \mathbf{pm}[\![C]\!](g)(s) && \text{(by assumption above)}\\
&\leq ([\neg\phi]\cdot I)(s') + \sum_{s\neq s'} ([\neg\phi]\cdot I)(s) && \text{(by (‡‡))}\\
&= |[\neg\phi]\cdot I| \ . && \text{Contradiction: } |[\neg\phi]\cdot I| < |[\neg\phi]\cdot I| \quad \square
\end{aligned}$$

请注意，对于初始分布 $g = \delta_s$，满足条件（1）和（2）的占用超不变式足以通过推论 1证明状态 s 上的 PAST。

例 *3*。我们继续分析程序 C 中的几何循环 1。在例 2中，我们建立了占用超不变式 $I = 2^{-(c+1)}\cdot[x=0] + 2^{-c}\cdot[x=1]$，从而得出 $\mathbf{pm}[\![C]\!](g) \leq [x\neq 1]\cdot I$。这里，我们证明这个不等式实际上是一个等式。我们有 $|I| = \sum_{k\in\mathbb{N}} 2^{-(k+1)} + \sum_{k\in\mathbb{N}} 2^{-k} = 3 < \infty$，并且 $|[x\neq 1]\cdot I| = \sum_{k\in\mathbb{N}} 2^{-(k+1)} = 1 = |g|$。根据定理 6，现在可得

$$\mathbf{pm}[\![C]\!](g) = [x\neq 1]\cdot I = 2^{-(c+1)}\cdot[x=0] \ ,$$

并得出结论：对于所有状态 $s \in \mathsf{States}$，若 $s(x) = 0$，则终止于 s 的概率恰好为 $2^{-(s(c)+1)}$，否则为 0。

6 不变式合成

本节将介绍一种自动化的、基于模板的占用不变量综合技术（见定义 10）。基于模板的概率循环不变式综合在文献中十分常见（例如，[31, 4]）；本文的新颖之处在于使用生成函数模板来表示测度集合。给定这样的模板，我们可以自动检查是否存在某种参数实例化，使其产生有效的占用不变式。我们首先介绍生成函数，并调整我们的pm语义，使其能够在这一域上运行。

6.1 生成函数语义

设 $\mathbf{X} = (X_v)_{v\in\mathsf{Vars}}$ 是一族形式变量或不定元，每个程序变量 $v \in$变量集 对应一个（在示例中，我们只写 X， Y而不写 X_x， X_y等）。形式上，生成函数是值的（无限）序列，通常表示为形式幂级数。详细介绍请参见 [55, 37]。

定义 *11*（形式幂级数）。设 R 为一个半环。以不定元 X 为变量的 R 值形式幂级数（*FPS*）是一个函数 $\mathfrak{g}$: 状态 $\to R$，记作 $\mathfrak{g} = \sum_{s\in}$状态 $\mathfrak{g}(s)\cdot X^s$，其中 $X^s = \prod_{v\in}$变量集 $X^s(v)_v$。多项式是一个形式幂级数 $\mathfrak{g}$，其中 $\mathfrak{g}(s) \neq 0$ 仅对有限个 $s \in$ 状态成立。形式幂级数的集合记作 $R[[X]]$。多项式的集合记作 $R[X]$。

$\mathrm{R}[[\mathbf{X}]]$ 和 $\mathrm{R}[\mathbf{X}]$ 在（逐点）加法和（柯西）乘法下构成半环:

$$\mathfrak{g}+\mathfrak{h} \coloneqq \sum_{s\in\mathsf{States}} (\mathfrak{g}(s)+\mathfrak{h}(s))\cdot\mathbf{X}^s, \quad \mathfrak{g}\cdot\mathfrak{h} \coloneqq \sum_{s\in\mathsf{States}} \Big(\sum_{s_1+s_2=s} \mathfrak{g}(s_1)\cdot\mathfrak{h}(s_2)\Big)\cdot\mathbf{X}^s .$$

若R是一个环，那么$\mathrm{R}[[\mathbf{X}]]$ 和 $\mathrm{R}[\mathbf{X}]$也都是环。

我们使用形式幂级数来表示程序状态集合上的测度，方法如下:

定义 12（生成函数）。测度 $\mu \in \mathcal{M}$(状态)的生成函数GF_μ 是形式幂级数GF_μ: $= \sum_{s\in}$状态 $\mu(s)\cdot\mathbf{X}^s \in \mathbb{R}^\infty_{\geq 0}[[\mathbf{X}]]$.反过来，每个形式幂级数 $\mathfrak{g} \in \mathbb{R}^\infty_{\geq 0}[[\mathbf{X}]]$都定义了一个测度 $\mu_\mathfrak{g} \in \mathcal{M}$(状态)，满足 $\mu_\mathfrak{g}(s) =$$\mathfrak{g}(s)$，其中 $s \in$状态。

定义 12 中的生成函数目前尚无太大用处；它们只是表示测度的无穷表达式的另一种记法。为了能够综合以生成函数表示的占用不变量，我们的关键思路是利用一大类测度都允许将其生成函数表示为闭式；本文仅考虑将其生成函数表示为有理函数（即多项式之比）这种闭式。例如，几何循环示例程序（程序1）的最终分布为

$$\mathrm{GF}_{\mathrm{geom}(1/2)} = \frac{1}{2}+\frac{1}{4}C+\frac{1}{8}C^2+\ldots, \qquad \text{with rational closed form } \frac{1}{2-C}.$$

在解释细节之前，我们先定义一个能够保持测度具有有理闭式的 pGCL 片段。

该ReDiP 片段pGCL。我们通过 [13]考察 pGCL 的矩形离散概率程序（ReDiP）这一语法片段。该片段能够在概率测度的生成函数上定义程序语义，并保持生成函数的某些闭式表示。我们将这一结果推广到定义12中的生成函数。

ReDiP 的语法见表2。它包含多种用于修改变量的语句：常量赋值、递减操作和递增操作。后者从定义在 $\mathbb{N}$ 上的概率分布中抽取多个独立同分布样本，并将这些样本之和加到变量的值上

表 2：ReDiP 程序的语法（左列）和 pm-语义（右列），其中 v、$y \in$ 变量集、$n \in \mathbb{N}$ 和 $D \in \mathcal{D}_{=1}(\mathbb{N})$ 表示 上的概率分布，其中 $\mathbb{N}$ 带有一个新的形式变量 $[\![D]\!] \in \mathbb{R}_{\geq 0}[[T]]$ $T \notin \mathbf{X}$。

P	$\mathbf{pm}[\![P]\!](\mathfrak{g})$
$v := n$	$\mathfrak{g}[X_v/1] \cdot X_v^n$
$v{--}$	$(\mathfrak{g} - \mathfrak{g}\restriction_{v<1}) \cdot X_v^{-1} + \mathfrak{g}\restriction_{v<1}$
$v \mathrel{+}= $ 独立同分布$(D,\ y)$	$\mathfrak{g}[X_y/X_y \cdot [\![D]\!][T/X_v]]$
若（$v < n$）$\{P_1\}$否则$\{P_2\}$	$\mathbf{正负}[\![P_1]\!](\mathfrak{g}\restriction_{v<n}) + \mathbf{正负}[\![P_2]\!](\mathfrak{g} - \mathfrak{g}\restriction_{v<n})$
$P_1 \fatsemi P_2$	$\mathbf{正负}[\![P_2]\!](\mathbf{正负}[\![P_1]\!](\mathfrak{g}))$
当（$v < n$）时$\{P_1\}$	$\mathfrak{h} - \mathfrak{h}\restriction_{v<n}$，　其中 $\mathfrak{h} =$ 最小不动点 $\mathfrak{i}.\, \mathfrak{g} + \mathbf{pm}[\![P_1]\!](\mathfrak{i}\restriction_{v<n})$

（在纯 pGCL 中，此操作可以通过循环来表示；但在 ReDiP 中，它被视为原语操作，而非循环）。样本数量由当前状态下另一个程序变量的值决定。该语言支持分支，以及带有如下形式矩形守卫的循环：$v < n$ 其中 $v \in$ 为变量集 且 $n \in \mathbb{N}$。我们可以用 ReDiP 的无循环片段模拟递增操作右侧的线性表达式（例如 $x \mathrel{+}= x + 2y + 1$），也可以模拟变量与常数比较所得布尔组合（例如 $x = 5 \,||\, y > 3$）。我们还将递增赋值 $x \mathrel{+}= E$ 写作 $x := x + E$。语言的详细介绍见 [13] 。

定义 13（ReDiP）。基于生成函数的 pm-语义定义如下：矩形离散概率程序（*ReDiP*）在表 *2* 中定义，其中在定义 $\mathbf{pm}[\![P]\!] : \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]] \to \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 时，我们使用以下运算：

– 代换：对于 $\mathfrak{g}, \mathfrak{h} \in \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 和 $v \in$ 变量集：

$$\mathfrak{g}[X_v/\mathfrak{h}] \;:=\; \sum_{s \in \mathsf{States}} \Big(\sum_{\substack{s_1 + s_2 = s \\ s_2(v) = s(v)}} \sum_{k \in \mathbb{N}} \mathfrak{g}(s_1[v \mapsto k]) \cdot \mathfrak{h}^k(s_2) \Big) \cdot \mathbf{X}^s \ .$$

– （向下）移位：对于 $\mathfrak{g} \in \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 和 $v \in$ 变量集，满足 $\mathfrak{g}(s) = 0$ 对所有满足 $s(v) = 0$的 $s \in$ 状态 均成立：

$$\mathfrak{g} \cdot X_v^{-1} \;:=\; \sum_{s \in \mathsf{States}} \mathfrak{g}(s[v \mapsto s(v) + 1]) \cdot \mathbf{X}^s \ .$$

– 形式导数：对于 $\mathfrak{g} \in \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 和 $v \in$ 变量集：

$$\partial_{X_v} \mathfrak{g} \;:=\; \sum_{s \in \mathsf{States}} (s(v) + 1) \cdot \mathfrak{g}(s[v \mapsto s(v) + 1]) \cdot \mathbf{X}^s \ .$$

– 限制：对于 $\mathfrak{g} \in \mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$、 $v \in$ 变量集、 $n \in \mathbb{N}$:

$$\mathfrak{g}\restriction_{v<n} \;\coloneqq\; \sum_{i=0}^{n-1} \tfrac{1}{i!}(\partial^{i}_{X_v}\mathfrak{g})[X_v/0]\cdot X_v^{i}$$

$$\textit{and} \qquad \mathfrak{g} - \mathfrak{g}\restriction_{v<n} \;\coloneqq\; \sum_{s\in \mathsf{States}} [\mathfrak{g}(s) \neq \mathfrak{g}\restriction_{v<n}(s)]\, \mathfrak{g}(s)\cdot \mathbf{X}^{s}\ .$$

这些操作都是单调的，其中 $\mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 上的偏序由 $\mathcal{M}$(状态)继承。因此，表 *2* 中的最小不动点是良定义的。

直观而言，更新通过代入操作提取生成函数中受更新影响的部分来执行。例如， $v \coloneqq n$ 对 X_v 进行边缘化（即暂时将 v 设为 0），然后在 X_v 中按 n 进行（向上）移位。

用于表示附加操作的符号暗示了它们预期的效果；例如，尽管在 $\mathbb{R}^{\infty}_{\geq 0}[[\mathbf{X}]]$ 上没有减法，但它实际上相当于对 $\mathfrak{g}$ 和 $\mathfrak{g}\restriction_{v<n}$ 做减法。代换操作对应于直观的代换 $\mathfrak{g}[X_v/\mathfrak{h}] = \sum_{s\in \text{状态}} \mathfrak{g}(s)\mathfrak{h}^{s(v)}\cdot \mathbf{Y}^{s}$，其中 $\mathbf{Y}$ 包含 $\mathbf{X}$ 的所有形式变量，但不包括 X_v。代换总是良定义的，因为在我们的非负扩展实数域中，每个单调序列都会收敛。

我们的语义是对 [13]的推广，从概率生成函数推广到一般生成函数。

命题 1。 给定一个分布 $\mu \in \mathcal{D}$(状态) 以及一个 ReDiP-程序 P。则 $\mathrm{GF}_{\mu} \in \mathbb{R}_{\geq 0}[[\mathbf{X}]]$ 和 $\mathbf{pm}[\![P]\!]\,(\mathrm{GF}_{\mu})$ 与 ReDiP 语义 $[\![P]\!]\,(\mathrm{GF}_{\mu})$ 中定义的 *[13,* 表 *2]*中定义的语义一致。

证明（概要）。 我们的定义在结构上与 [13,表 2], 中的定义完全相同，唯一区别在于循环的情形。对于（次）概率生成函数， [13] 使用 $\mathbb{R}[[\mathbf{X}]]$上的运算。（向下）移位和形式导数的定义完全相同。限制操作也类似，其中 $\mathfrak{g} - \mathfrak{g}\restriction_{v<n}$ 仅使用环 $\mathbb{R}[[\mathbf{X}]]$的减法。虽然代换操作$\mathfrak{g}[X_v/\mathfrak{h}]$ 对于所有 $\mathfrak{g},\mathfrak{h} \in \mathbb{R}[[\mathbf{X}]]$, 都未必有定义，但对于（次）概率生成函数 $\mathfrak{g},\mathfrak{h}$，它的定义相同且有定义，这足以定义程序语义。注意， $\mathfrak{g}\restriction_{v<1} = \mathfrak{g}[X_v/0]$。

对于循环， [13] 采用 Kozen 的高阶不动点构造（参见第 4.2 节），而我们采用占用测度不动点构造。等价性证明与第 4 节中相同。

有理闭式。 我们的目标是利用形式幂级数的代数结构，将测度有限地表示为形式分式，从而在保持有限表示的同时应用 ReDiP 程序的语义。对于具有 $\mu(s) \neq \infty$ 的测度，我们通过将其生成函数嵌入 $\mathbb{R}[[\mathbf{X}]]$ ，并利用其中存在的逆元集，实现了对所有 $s \in$ 状态的这一目标。

定义 14（有理闭式）。 对于 $\mathfrak{g} \in \mathbb{R}_{\geq 0}[[\mathbf{X}]]$, , 若一对多项式 $\mathfrak{n}_\mathfrak{g}$、 $\mathfrak{d}_\mathfrak{g} \in \mathbb{R}[\mathbf{X}]$ 满足： $\mathfrak{d}_\mathfrak{g}$ 在 $\mathbb{R}[[\mathbf{X}]]$ 中可逆，且 $\mathfrak{g} = \mathfrak{n}_\mathfrak{g} \cdot \mathfrak{d}_\mathfrak{g}^{-1}$ 属于 $\mathbb{R}[[\mathbf{X}]]$, 则称其为 $\mathfrak{g}$ 的 <*stype id='42'*> （有理）闭式，记作 $\mathfrak{g} = \frac{\mathfrak{n}_\mathfrak{g}}{\mathfrak{d}_\mathfrak{g}}$。
例如， $1 \cdot X^0 + 2 \cdot X^1 + 4 \cdot X^2 + \cdots = \sum_{i \in \mathbb{N}} 2^i \cdot X^i$具有有理闭式 $\frac{1}{1-2X}$。 排除 ∞ 作为系数，可在环 $\mathbb{R}[[\mathbf{X}]]$中得到闭式。如 [13], 所述，这使得我们可以在闭式上定义定义 13 中的向下移位、形式导数和限制运算。唯一的例外是代换，例如 $(\sum_{i \in \mathbb{N}} 2^i \cdot X^i)\,[X/1] = \infty \neq -1 = \frac{1}{1-2X}\,[X/1]$。只有当代入项没有常数项时，代换在闭式上才是良定义的：若 $\mathfrak{g} = \frac{\mathfrak{n}_\mathfrak{g}}{\mathfrak{d}_\mathfrak{g}}$, $\mathfrak{h} = \frac{\mathfrak{n}_\mathfrak{h}}{\mathfrak{d}_\mathfrak{h}}$满足 $\mathfrak{h}(\mathbf{0}) = 0$, 则 $\frac{\mathfrak{n}_\mathfrak{g}[X_v \;\frac{\mathfrak{d}_\mathfrak{h}}{1}]}{\mathfrak{d}_\mathfrak{g}[X_v \;\frac{\mathfrak{d}_\mathfrak{h}}{1}]} // \frac{\mathfrak{n}_\mathfrak{h}}{\mathfrak{d}_\mathfrak{h}}]$是 $\mathfrak{g}[X_v/\mathfrak{h}]$的闭式。

在对闭式上的运算施加这一限制后，pm 对于闭式上的无循环 ReDiP 程序可以计算，但有两个限制：对于常量赋值 $v := n$ ，它一般未必有良好定义，例如，对 $x := 0$应用 $\frac{1}{1-X}$ 会得到 ∞ ，而它不是有理闭式。其次，独立同分布采样增量 $v \mathrel{+}= \mathtt{iid}(D,\ y)$必须限制为具有有理闭式的分布 D。我们将这一片段称为闭式ReDiP（clReDiP）：

定义 15（闭式 ReDiP）。一个clReDiP 程序是一个无循环ReDiP程序，不含常量赋值 $v := n$ ，且其中每个使用的概率分布 D 都具有闭式 $[\![D]\!] = \frac{\mathfrak{n}_{[\![D]\!]}}{\mathfrak{d}_{[\![D]\!]}}$。

推论 2（[13]）。设 $\frac{\mathfrak{n}_\mathfrak{g}}{\mathfrak{d}_\mathfrak{g}}$ 是生成函数 $\mathfrak{g} \in \mathbb{R}_{\geq 0}[[\mathbf{X}]]$,的一个闭式，且 P 是一个 clReDiP 程序。那么 pm$[\![P]\!](\mathfrak{g})$ 具有闭式，可通过对该闭式应用表*2* 中的运算得到。

备注 *1*（我们语言的表达能力）。虽然常量赋值$v := n$ 在 clReDiP 中不被允许，但可以通过以下循环模拟：当 $(v > 0)\,\{v{-}{-}\}\,;x \mathrel{+}= n$。然而，最终得到的循环不一定总是存在有理不变量。在无循环 ReDiP 中，无法模拟将一个变量与另一个变量进行比较 [13], , 因此 clReDiP 中也无法做到。我们猜想，守卫条件和赋值中的多项式表达式一般也无法模拟。据我们所知，无循环 ReDiP 和 clReDiP 能表达的分布变换器的确切类别仍然未知。我们还注意到，尽管 ReDiP 在语法上受到限制，带循环的 ReDiP 仍是图灵完备的，因为它可以模拟双计数器机。

6.2 基于模板的不变式合成

考虑一个 ReDiP 循环 $P = \text{当}(v < n)\,\{P_0\}$，其 clReDiP 循环体为 P_0 ，初始测度为 $\mathfrak{g}$ 闭式。我们使用生成函数 pm-语义，在有理闭式上为该循环综合占用不变量 P ，起始于$\mathfrak{g}$。我们将依赖（用户）提供的模板，将不变式综合问题归约为参数综合问题。

令 T 表示一个有限的符号模板参数集合，它们与形式变量 $\mathbf{X} = (X_v)_{v \in \mathsf{Vars}}$ 不同。我们的模板是有理闭式

其中包含多项式表达式 $\mathbb{R}[\mathrm{T}]$，而不仅仅是作为系数的值 $\mathbb{R}$。对于 $\tau\colon \mathrm{T} \to \mathbb{R}$，$I[\tau]$ 表示对 I 的实例化，其中所有出现的 $t \in \mathrm{T}$ 都已替换为 $\tau(t)$，前提是结果定义良好。注意，并非每个参数赋值都会产生有效的实例化。例如，将模板 $I = p \cdot V + (1-p) \cdot V^2$ 用 $\tau = [p \mapsto -1]$ 实例化，会得到一个有效的 $\mathrm{FPS} I[\tau] = -1V + 2V^2$，其形式变量为 $\mathbb{R}[[\mathbf{X}]],$，但它并不表示一个非负测度。

使用模板合成不变式 I 分为两个步骤。首先，我们将循环的特征泛函 $\Phi_{\mathfrak{g},\ P}$ 作用于 I 一次，并得到 $\Phi_{\mathfrak{g}\text{的闭式}P}(I) = \mathfrak{g} + \mathbf{pm}[\![P_0]\!]\,(I\restriction_{v<n})$。通过将参数 T 解释为新的程序变量，我们可以直接使用上一节中的定义。为了确保 I 表示一个有效的不变式，我们要求它满足不动点方程 $\Phi_{\mathfrak{g},\ P}(I) = I$。通过对分母进行交叉相乘，可以得到两个多项式之间的方程。比较系数后得到一个方程组，它对符号参数 T 施加约束；该方程组可以使用现成的计算机代数工具求解（例如，有理函数运算、格罗布纳基）。其次，我们验证满足这些约束的一个实例化所得到的生成函数是否对应于非负测度。

定理 7。给定 $\mu \in M$（状态集），其生成函数 μ 具有有理闭式，I是一个有理模板，且 $\tau\colon T \to \mathbb{R}$，满足 $\Phi_{\mathrm{GF}_\mu,P}(I[\tau]) = I[\tau]$。如果 $I[\tau] \in \mathbb{R}^{\infty}_{\geq 0}[[X]],$，那么 $\mu_{I[\tau]}$ 是从 μ 开始执行 *while* $(v < n)\ \{P_0\}$ 的占用不变式。

例 4。我们继续讨论程序 1 中的 1几何循环示例。进入循环的初始概率分布 $g = [x = 1 \wedge c = 0]$ 的生成函数为 $\mathrm{GF}_g = 1 \cdot X^1 C^0 = X$，它也是一个闭式。考虑以下有理模板 I，其中 $\mathcal{T} = \{a_0,\ a_1,\ a_2,\ a_3,\ b_0,\ b_1,\ b_2\}$：

$$I \quad = \quad \frac{a_0 \ + \ a_1 \cdot X \ + \ a_2 \cdot C \ + \ a_3 \cdot X^2}{b_0 \ + \ b_1 \cdot X \ + \ b_2 \cdot C}\ .$$

应用 $\Phi_{\mathrm{GF}_g,\ P}$ 循环 $P = \texttt{while}\,(x = 1)\,\{\{x :\!= 0\}\ [\tfrac{1}{2}]\,\{c :\!-= c + 1\}\}$ 到 I可得：

$$\Phi_{\mathrm{GF}_g,P}(I) \ = \ X + \frac{1}{2} \cdot (1 + XC) \cdot \frac{(a_1 b_0 - a_0 b_1) + (a_1 b_2 - a_2 b_1) \cdot C}{(b_0 + b_2 \cdot C)^2}\ .$$

通过对有理函数模板进行符号求解方程 $\Phi_{\mathrm{GF}_g,\ P}(I) = I$，我们得到赋值 $\tau = [a_0 \mapsto 1, a_1 \mapsto 2, a_2 \mapsto 0, a_3 \mapsto 0, b_0 \mapsto 2,\ \ b_1 \mapsto 0,\ \ b_2 \mapsto -1]$，满足 $\Phi_{\mathrm{GF}_g,\ P}(I[\tau]) = I[\tau]$。作为

$$I[\tau] \ = \ \frac{1 \ + \ 2 \cdot X}{2 \ - \ C} \ = \ (1 + 2 \cdot X) \sum_{k \in \mathbb{N}} \frac{1}{2^{k+1}} \cdot C^k \ = \ \sum_{k \in \mathbb{N}} \frac{1}{2^{k+1}} \cdot C^k + \sum_{k \in \mathbb{N}} \frac{1}{2^k} \cdot X C^k$$

仅含非负系数，因此根据定理 $I[\tau]$ 7 表示一个占用不变式。事实上，这是我们在示例 3中得到的循环占用测度：

$$\mu_{I[\tau]} \quad = \quad [x = 0] \cdot 2^{-(c+1)} \ + \ [x = 1] \cdot 2^{-c}\ .$$

根据定理 6 对于 $\mu_{I[\tau]}$，我们得出结论

$$\mathbf{pm}[\![C]\!]\,(g) \ = \ [x \neq 1] \cdot \mu_{I[\tau]} \ = \ [x = 0] \cdot 2^{-(c+1)}\ .$$

当程序结构形成规律的递推模式，并可将其转化为生成函数上的代数等式时，这种方法尤其有效。将此类结构嵌入模板后，便可从抽象且直观的层面分析循环，并借助自动化求解器完成精确系数的繁琐计算。对于更复杂的情况，可能需要采用额外的近似技术或特定领域的模板，以下示例对此进行了说明。

例5. 考虑 快速骰子滚轮 [40]算法 FDR_N (程序 2)，其中某个固定整数常数为 $N > 0$。令 P_N 表示循环体。循环最初由以下状态到达：$\mathfrak{g}_0 := 1 \cdot V^1 C^0 F^0$。对于 $N > 4$，迭代循环体可得到

程序 2 快速骰子滚轮 [40]（pGCL 版本，来自 [5]；$N{>}0$ 常量）。

```
/// V   /* 初始时， v = 1, c = f = 0 */
当(f = 0) {
  v := 2v ;
  {c := 2c} [1/2] {c := 2c + 1} ;
  如果(N ≤ v) {
    如果 (c < N) {
      f := f + 1
    } 否则{
      v := v − N ;
      c := c − N} } }
```

$$\mathfrak{g}_1 := \mathbf{pm}[\![P_N]\!]\,(\mathfrak{g}_0 \restriction_{f=0}) = \tfrac{1}{2} V^2 (C^0 + C^1),$$
$$\mathfrak{g}_2 := \mathbf{pm}[\![P_N]\!]\,(\mathfrak{g}_1 \restriction_{f=0}) = \tfrac{1}{4} V^4 (C^0 + C^1 + C^2 + C^3).$$

c 只要 0 和 $v-1$之间，则始终服从均匀分布 $v < N$ [40]。因此，我们期望不变量包含如下形式的项$a_{N,\,i} \cdot V^i \sum_{k<i} C^k = a_{N,\,i} \cdot V^i \frac{1-C^i}{1-C}$对于某个 $a_{N,\,i} \in \mathbb{R}_{\geq 0}$, $i < N$。在一次迭代中，若 v增加至至少 N，分布会分成两部分：一部分是终止部分，其中 $c < N$和 f设为 1；另一部分是继续部分，其中 $c \geq N$和 c, v已减小 N。例如，对于 $N = 6$，我们有

$$\mathfrak{g}_3 := \mathbf{pm}[\![P_N]\!]\,(\mathfrak{g}_2 \restriction_{f=0}) = \frac{1}{8} \cdot FV^8 \frac{1-C^6}{1-C} + \frac{1}{8} \cdot V^2 \frac{1-C^2}{1-C}.$$

因此，我们预计不变式会包含项 $a_{N,i} \cdot F\,V^i \frac{1-C^N}{1-C}$，其中某些 $a_{N,i} \in \mathbb{R}_{\geq 0}, i \geq N$。我们还观察到：由于 v只会翻倍，或者在 $v \geq N$时减去 N， v必须与 2 的某个幂模 N同余，并且$v < 2N$ 在循环迭代之间始终成立。因此，我们提出如下不变式模板：

$$I_N = \sum_{1 \leq i < N \,\wedge\, \exists k:\, i \equiv_N 2^k} a_{N,i} \cdot V^i \frac{1-C^i}{1-C} + \sum_{N \leq i < 2N \,\wedge\, \exists k:\, i \equiv_N 2^k} a_{N,i} \cdot F\,V^i \frac{1-C^N}{1-C}.$$

正如第 B 节中的详细计算所示， $I_N = \mathfrak{g}_0 + \mathbf{pm}[\![P_N]\!]\,(I_N \restriction_{f=0})$有一个解 τ_N 该解给出了一个非负有限实例化，且满足

$$\sum_{N \leq i < 2N \,\wedge\, \exists k:\, i \equiv_N 2^k} \tau_N(a_{N,i}) = \frac{1}{N}.$$

根据定理 7， $\mu_{I_N[\tau_N]}$ 是 FDR_N的一个占用不变式，我们利用它来界定程序的语义。对 c 在 0 与 $N-1$ 之间的期望均匀分布进行边缘化，消去 v和 f后，我们得到其上界：

$$\begin{aligned}\mathbf{pm}[\![\mathrm{FDR}_N]\!]\,(\mathfrak{g}_0)\,[V,F/1] \;&\le\; (I_N[\tau_N]\restriction_{f>0})[V,F/1]\\ &= \sum_{N\le i<2N\,\wedge\,\exists k:\; i\equiv_N 2^k} \tau_N(a_{N,i})\cdot\frac{1-C^N}{1-C} \;=\; \frac{1}{N}\cdot\frac{1-C^N}{1-C}\,.\end{aligned}$$

由于 $\left|\mu_{I_N[\tau_N]}\right| = 1$是有限的，我们得出该循环是 $\mu_{\mathfrak{g}_0}$上的 PAST，因此 $\left|\mathbf{pm}[\![\mathrm{FDR}_N]\!]\,(\mathfrak{g}_0)\right| = |\mathfrak{g}_0| = 1$。因此，根据定理6，均匀分布上界是紧的，我们得出$\mathbf{pm}[\![\mathrm{FDR}_N]\!]\,(\mathfrak{g}_0)[V,F/1] = \frac{1}{N}\cdot\frac{1-C^N}{1-C}$。

嵌套循环。要处理嵌套循环，必须同时为每个嵌套循环找到一个不变式，因为进入内层循环的测度直接受到进入外层循环的测度影响。验证从最内层循环开始，并使用得到的测度证明下一个外层循环的不变式。

或者，作为预处理步骤，我们可以将任意程序（自动）重写为一个循环，其循环体无循环（例如， [48]），前提是能够在 ReDiP 中完成这种转换。

7 实现与评估

为了验证我们针对概率循环的不变式合成方法在实践中的可行性，我们开发了一个概念验证实现。该工具实现了第6 节介绍的方法，并支持验证以及在许多情况下自动综合有理闭式的归纳不变式。

7.1 合成概述

我们综合有理函数不变式的方法基于对概率循环进行符号不动点推理。给定一个概率`while`循环和一个有理闭式的初始分布，我们通过有理函数模板启发式生成候选项。每个候选项都被解释为一个符号分布，并接受一个前向语义步骤：我们假设候选项具有归纳性，并计算一次循环迭代的影响。由此得到候选项与一次循环迭代语义结果之间的有理方程，我们使用符号方法，根据配置，借助 SMT 求解器（例如 Z3）或计算机代数系统SymPy对其求解。

所得方程组的解对应于有理候选式中符号系数的实例化。我们会滤除平凡或无效的解，例如使分母恒为零的解。随后，使用启发式正性测试 检查剩余候选式，以确定所得生成函数是否表示有效（即非负）测度。如果找到这样的候选式，我们便可得出结论：它定义了该

概率循环的真正归纳不变式。如果没有候选式通过验证，该过程便会失败，这表明启发式搜索空间不足，或者不存在所考虑形式的有理不变量。

7.2 实现细节

该工具使用 Python 实现 [53]，支持两种符号计算后端：SymPy [44] 库，以及一组自定义的 Python 绑定，用于调用 C++库 GiNaC 。后者在有理函数运算方面性能更佳。后端可在运行时配置。

初始分布以生成函数的形式指定，通常为有理闭式。这些生成函数在内部表示为符号表达式，并根据归纳不变式的不动点刻画所需的运算进行处理。化简和代数推理任务交由所选后端完成。

除第6 节介绍的语言外，我们的实现还有限度地支持在单变量生成函数上使用取模表达式的守卫条件。与 [20], 一样，我们支持如下形式的表达式：$v = c$ 模 d，其中$v \in$ 变量集 和 $c,\ d \in \mathbb{N}$。该功能通过一种基于阿达玛积的标准方法实现；例如参见 [38,第 2 节。4]。

不变量模板生成。 对于我们的基准测试，我们通过系统地生成结构复杂度逐渐增加的多变量有理表达式，来探索候选不变量。具体而言，该启发式方法首先固定分母的总次数上界，然后枚举所有次数不超过分母次数的分子多项式。对于每一种这样的次数配置，它都会为分子和分母构造符号模板，并为每个可接受的单项式引入新的系数参数。随后，这些有理表达式会被解释为生成函数，并封装成符号分布对象。这种广度优先枚举策略确保先考察较简单的候选项，再考察较复杂的候选项，从而能够高效、系统地搜索有理归纳不变量空间。至关重要的是，从理论上讲，这种枚举是完备的：如果存在有理函数不变量，那么在某个有限次数上界下，该启发式方法就会生成一个可实例化为该不变量的相应模板。

正性测试启发式。 判定任意有理函数是否具有非负形式幂级数展开是一个开放问题，且被认为是可判定的 [45]。然而，为确保候选有理函数确实表示一个有效的生成函数——即具有非负系数的幂级数——我们采用充分的启发式方法来确立其正性。对于基准测试，我们使用基于有理表达式结构性质的启发式方法，将候选表达式分解为分子和分母。对于分子多项式，我们验证其仅包含非负系数。对于分母，则检查一个简单的符号条件：常数项必须

表3：使用针对给定基准测试最快的后端（GiNaC 或 SymPy）对 ProDiGy 进行验证和综合的基准测试结果。

基准测试	时间（秒）	自动/用户	其他评论
次级分布 -进入	0.568013	A	
cond_and	0.017394	A	
故障_递减	0.120409	A	
几何	0.026373	A	
几何_计数器	0.124910	A	
模_几何	0.035103	U	$a = 4f, b = 2f, c = f, d = 8f,$ $e = 6f$
三分之几何	6.681842	A	使用 Z3 求解
随机 _walk	0.118271	A	
随机 _walk_计数器	0.106519	U	$a = 1, b = 1, d = -1$
fast_dice_滚轮	0.155679	U	$w_0 = \frac{1}{3}, w_1 = 0, w_2 = \frac{2}{3}, w_3 = 1,$ $w_4 = \frac{1}{6}, w_5 = 0, w_6 = 0, w_m = 0$
非终止	失败	A	实际不变量为 $\infty \cdot X + {}^{1\cdot} X_2^2$
顺序执行_循环	失败	A	无法确定 $\frac{2C}{C^2-3C+2} + 1$ 的位置——

为正，且所有其他系数必须为非正。如果函数本身或其相反数符合这一模式，该启发式方法便判定其为非负。此测试是可靠的，因为它涵盖了（广义）几何分布以及其他能够保证展开为非负系数的类似有理形式幂级数。

7.3 基准测试配置

评估基于一组规模从小型到中型的基准测试循环，这些循环包含概率更新、分支以及复杂程度各异的控制流结构。我们的实验旨在回答以下研究问题：

– **可行性**：实践中能否综合出非平凡的有理不变量？ – **性能**：针对基准测试程序，找到不变量的速度有多快？ – **覆盖范围**：对于哪些循环模式，该启发式方法能够成功或失败？

所有基准测试均在一台配备 Apple M3 Pro 处理器和 18GB 内存、运行 macOS Sequoia 15.4.1 的 Apple MacBook Pro 上执行。我们仅报告核心综合阶段的执行时间，不包括输入解析或输出格式化等预处理步骤。

7.4 基准测试结果

表3 总结了对每个基准测试运行综合工具所得的结果。每一项都记录了完成综合所用的最快时间、是否自动计算出有效不变式（A）或通过用户指定模板计算出有效不变式（U），以及在后一种情况下推断出的参数值。

7.5 讨论

结果表明，我们的符号综合框架能够处理各种概率循环，并且在大多数情况下能够高效地综合有理归纳不变式。综合时间通常不到一秒。标记为 U 的基准测试表示：给定用户指定的模板后，工具针对特定参数值综合出不变式，并从符号方程组中自动恢复了这些参数值。我们还观察到，后端（GiNaC 或 SymPy）的选择会影响性能，但不会影响正确性；对于有理表达式的化简，GiNaC 通常更快。

失败情况很少见，通常归因于启发式搜索空间或正性测试的局限性。未来的工作将探索扩展模板空间，并开发更精细的正性检查。

8 相关工作

本文的主要理论技术——利用占用不变量对概率循环进行推理——受到 [49]的启发。然而， [49] 从语义角度将概率程序视为（可数无限）马尔可夫链。因此，他们的定义和案例研究都基于显式的无限随机矩阵。相比之下，我们直接在程序文本上定义占用不变量并据此进行推理。这对于我们基于生成函数进行自动验证与综合至关重要。

本节余下部分将概述其他相关工作，涉及：（1）生成函数；（2）后验分布推断；（3）定量不变量综合；（4）概率模型检验。

生成函数在概率程序分析与语义中的应用。将生成函数（GF）应用于概率程序是一个相对较新的趋势，其理论基础源自 [33]。后续工作实现并应用了这一框架，用于程序等价 [13] 和条件化 [34]；全面综述请参见 [32] 。独立地， [57] 为该方法增添了对连续分布的基本支持，并探索了用于精确后验推断的自动微分技术。最近， [39]引入了一种函数式语言（不支持无界循环/递归），可将程序编译为生成函数表达式。另有近期工作将生成函数与巴拿赫不动点定理结合，以推导精确的运行时间分布 [15], ，并通过加权自动机对生成函数进行编码 [20], ；这似乎也与我们的方法兼容。然而，上述论文均未提出一种能够自动将循环的后验分布推导为闭式生成函数的方法。

后验推断：近似、保证性与精确性。概率程序的算法后验推断已在多个领域得到广泛研究，并催生了 STAN [10], 、WebPPL [21], 和 Pyro [8] 等成熟工具。这些系统通常依赖基于采样的近似推断，而不提供

形式化保证。如 [7], 所示，某些方法在存在无界递归或循环时可能产生极不准确的结果。下文将讨论范围限定为能够提供保证性推断的面向验证的方法，即经过数学认证的结果。

保证性推断方法要么建立可靠的界限，要么计算真正精确的结果。在前一类方法中， [7, 54]在后验域的有限区间划分上证明上界和下界。这些方法支持无界循环和递归、连续分布以及一般的软条件化（评分）。对于离散程序， [56]提出了两种界限技术：一种基于循环展开和残余质量，另一种利用所谓的收缩不变式（占用不变量的一种特殊情形），并最终得到几何分布。精确方法包括 PSI [19], ，它以符号方式表示后验密度；以及 Dice [28], ，它采用加权模型计数，在有限状态程序中实现可扩展推断。两者都仅适用于静态有界循环。后来，Dice 扩展了对无界迭代的支持 [52]。值得探究占用测度是否也能应用于这一情形。

概率程序不变量综合。据我们所知，本文首次为概率循环综合一般的占用不变量。Zaiser、Murawski 和 Ong [56]近期提出并综合了收缩不变式，可视为占用不变量的一种特殊情形。重要的是，与后者不同，收缩不变式并不完备：它们无法证明后验测度的所有有效界限。具体反例如第 D 节所示。

其他类型定量不变量的综合已得到广泛探索。我们简要回顾一些有代表性的论文。早期工作包括 [31],„ 其通过约束求解，在最弱预期望演算中综合线性期望不变量 [42] 。这类不变量是定义在程序状态上的随机变量，其期望在循环迭代过程中不会降低；后来它们被称为次不变量 [29]。多项式次不变量可以通过结合模板、Positivstellensatz 推理和半正定规划来综合 [17]。其对偶概念超不变量 则通过 CEGIS 风格方法进行了综合 [4]。期望不变量与概率鞅 密切相关（例如参见 [24]）。自 [11],开创以来，显式鞅及其变体的综合也得到了广泛研究，通常采用模板和约束求解；综述见 [50]。一种广泛使用的变体是排名超鞅 [11],„ 它是排名函数在概率终止情形下的推广，并已有多种自动综合方法 [18, 1]。此外，在终止性证明的背景下， [12]引入了随机不变量，即以至少达到指定概率为条件、能够无限期成立的谓词。[3]开创了通过递推求解自动综合基于矩的不变量的研究；这类不变量是关于程序变量的期望和高阶矩的谓词。另一种与之正交的概念是分布不变量 [25, 2], ，不过据我们所知，目前尚无关于其自动综合的研究。

概率模型检验。 模型检验器 Storm [26]可以计算 有限 马尔可夫链 [43], 的期望访问时间，原则上可据此得到有限状态空间循环的占用不变量。然而，底层数值技术并不能轻易扩展到无限状态空间。

9 结论

我们根据占用测度定义了概率循环的测度变换器语义。占用测度是由给定初始分布所诱导的、在循环头处访问各状态的预期次数。在此基础上，我们引入了占用超不变量的概念，它们能够界定循环的输出分布并证明 PAST。此外，我们还展示了如何通过对有理闭式生成函数进行基于模板的枚举，自动综合占用不变量。

未来的工作包括加入更多模板枚举方法和启发式方法，以确立闭式生成函数的正性（例如，对于 [56] 的最终几何分布），并将此方法扩展到条件化，如 [34] 所示。另一个有前景的方向是将闭式方法扩展到处理 ∞ 系数，从而扩大所支持程序的类别。然而，由于扩展域 $\mathbb{R}^{\infty}[[\mathbf{X}]]$ 不是一个环，它缺少定义和运算闭式所需的重要代数性质。目前尚不清楚如何将我们的不变式合成推广到不同的数据类型，因为该方法使用离散非负整数域上的生成函数。我们推测，大部分理论也可以迁移到连续情形，可能需要将（概率）生成函数改为矩生成函数或特征函数。最后，值得研究占用测度能否用于计算非终止概率循环的平稳分布，类似于 [43] 针对有限离散时间马尔可夫链所采用的方法。

致谢

本研究部分经费由 ERC POC Grant *VERIPROB*（资助编号 101158076）、DFG RTG 2236 *UnRAVeL*以及欧盟地平线2020研究与创新程序的 Marie Skłodowska-Curie 资助协议 *MISSION*（资助编号 101008233）提供。

数据可用性声明

包含第7 节所述实现和基准测试程序的软件工件已在 Zenodo 上公开发布 [23]。

参考文献

1. Agrawal, S., Chatterjee, K., Novotný, P.: 字典序排序超鞅：一种高效的概率程序终止方法。《ACM 程序设计语言论文集》2 3158122 2 34 1 34 32 2018 10 1145（POPL），：–：（）. https://doi.org/ . / . Barthe, G., Espitau, T., Gaboardi, M., Grégoire, B., Hsu, J., Strub, P.: 概率程序的基于断言的程序逻辑。载于：ESOP。LNCS，第10801卷，第117–144页。施普林格，海德堡（2018）。https://doi.org/10.1007/978- 3-319-89884-1_53. Bartocci, E., Kovács, L., Stankovic, M.: 概率可解循环的基于矩的不变量自动生成。载于：ATVA。LNCS，第11781卷，第255–276页。施普林格，海德堡（2019 10 1007 978 3 030 31784 3 _15）。4 https://doi.org/ . / - - - - \ . Batz, K., Chen, M., Junges, S., Kaminski, B.L., Katoen, J., Matheja, C.: 通过归纳综合归纳不变式验证概率程序。载于：TACAS。LNCS，第13994卷，第410–429页。施普林格，海德堡（2023）。https://doi.org/10 1007 978 3 031 30820 8 _25 . / - - - - \ . Batz, K., Katoen, J.,5 Winkler, T., Zilken, D.: 通过分布不变量验证采样算法。CoRR abs/2509.06410（2025）。https://doi.org/10. 48550/ARXIV.2509.06410。arXiv：2509.06410 6. Bauer, C.W., Frink, A., Kreckel, R.B.: C++ 编程语言中的符号计算框架 GiNaC 简介。《符号计算杂志》33(1)，1–12（2002）。https://doi.org/10.1006/jsco.2001.0494。arXiv：cs/0004015 [cs.SC]7. Beutner, R., Ong, C.L., Zaiser, F.: 通用概率编程中的后验推断保证界。载于：PLDI，第536–551页。ACM（2022）8. Bingham, E., Chen, J.P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P.A., Horsfall, P., Goodman, N.D.: Pyro：深度通用概率编程。《机器学习研究杂志》20，28:1–28:6（2019）9. Bogachev, V.I.: 测度论。施普林格，柏林、海德堡（2007）10. Carpenter, B., Gelman, A., Hoffman, M.D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., Riddell, A.: Stan：一种概率编程语言。《统计软件杂志》76，1–32（2017）11. Chakarov, A., Sankaranarayanan, S.: 使用鞅进行概率程序分析。载于：CAV。LNCS，第8044卷，第511–526页。施普林格，海德堡（2013）。https://doi.org/10 1007 978 3 642 39799 8 _34 . / - - - - \ .12 Chatterjee, K., Novotný, P., Zikelic, D.: 概率终止的随机不变量。载于：POPL，第145–160页。ACM（2017）。https : / / doi . org / 10 . 1145 / 3009837.3009873 13. Chen, M., Katoen, J., Klinkenberg, L., Winkler, T.: 程序能否产生正确的分布？——通过生成函数验证概率程序。载于：CAV（1）。LNCS，第13371卷，第79–101页。施普林格，海德堡（2022）。https://doi.org/10 1007 978 3 031 13185 1 _5 . / - - - - \ . Chung, K.L.: 马尔可夫链。施普林格，柏林、海德堡（2012） 15. Collodi, L., Boreale, M., Gregorio, A.P.D.: 概率程序的代数分析初探。载于：ICTCS。CEUR研讨会论文集，第147–153页。CEUR-WS.org（2025）。https://ceur-ws.org/Vol-4039/paper12.pdf 16. Davey, B.A., Priestley, H.A.:《格与序导论》，第二版。剑桥大学出版社（2002）14

17. Feng, Y., Zhang, L., Jansen, D.N., Zhan, N., Xia, B.: 为概率程序寻找多项式循环不变量。载于：ATVA。LNCS，第 10482 卷，第 400–416 页。施普林格，海德堡（ 2017 10 1007 978 3 319 68167 2 _26 ）。
18
https://doi.org/ . / - - - - \ . Fioriti, L.M.F., Hermanns, H.: 概率终止：可靠性、完备性与组合性。载于：POPL，第 489–501 页。ACM (2015)。https://doi. org/10 2677001 19 1145 2676726 . / . . Gehr, T., Misailovic, S., Vechev, M.T.: PSI：概率程序的精确符号推断。载于：CAV（1）。LNCS，第 9779 卷，第 62–83 页。施普林格，海德堡（ 2016 10 1007 978 3 319 41528 4 _4）。https://doi.org/ . / - - - -
20
\ . Geißler, D., Winkler, T.: 离散概率程序精确推断的加权自动机。载于：ICTAC。LNCS，第 16237 卷，第 261–278 页。施普林格，海德堡（ 2025 10 1007 978 3 032 11176 0 _16）。https://doi.org/ . / - - -
21
- \ . Goodman, N.D., Stuhlmüller, A.: 概率编程语言的设计与实现，http://dippl.org (2014 年)。访问日期：2025-9-30。22. Goodman, N.D., Tenenbaum, J.B., Contributors, T.P.: 认知的概率模型，http://probmods.org/ (2016)。访问日期：03-05-2025。23. Haase, D., Batz, K., Gallus, A., Kaminski, B.L., Katoen, J.-P., Klinkenberg, L., Winkler, T.: 生成函数与占用测度：概率循环的不变式合成：工件，（2026年）。https : / / doi . org / 10 . 5281 / zenodo . 18184810. 24. Hark, M., Kaminski, B.L., Giesl, J., Katoen, J.: 目标定得低反而更难：概率程序验证中下界的归纳。ACM 程序语言学报 4 3371105 25 37 1 37 28 2020 10 1145 (POPL), : – : ( )。https://doi.org/ . / . den Hartog, J., de Vink, E.P.: 使用类霍尔逻辑验证概率程序。《国际计算机科学基础杂志》13(3), 315–340 (2002)。https://doi.org/10. 1142/S012905410200114X 26. Hensel, C., Junges, S., Katoen, J., Quatmann, T., Volk, M.: 概率模型检查器 Storm。《国际软件工具与技术转移杂志》24(4), 589–610 (2022)。https://doi.org/10.1007/S10009-021-00633-Z 27. Hoare, C.A.R.: 快速排序。《计算机杂志》5(1), 10–15 (1962)。https://doi.org/10. 1093/COMJNL/5.1.10 28. Holtzen, S., den Broeck, G.V., Millstein, T.D.: 扩展离散概率程序的精确推断。ACM 程序语言学报 4(OOPSLA), 140:1–140:31 (2020 3428208 29 10 1145)。https://doi.org/ . / . Kaminski, B.L.: 概率程序的高级最弱前置条件演算。博士论文，亚琛工业大学，德国 (2019)。http://publications. rwth-aachen.de/record/755408. 30. Kaminski, B.L., Katoen, J., Matheja, C., Olmedo, F.: 概率程序期望运行时间的最弱前置条件推理。载于：ESOP。LNCS，第 963

2,

第364–389页。施普林格，海德堡（2016年）。
https://doi.org/10.1007/978-3-662-49498-1_1531. Katoen, J., McIver, A., Meinicke, L., Morgan, C.C.：概率程序的线性不变量生成：基于证明的方法的自动化支持。载于：SAS。LNCS，第6337卷，第390–406页。施普林格，海德堡（2010年）。
https://doi.org/ 10 1007 978 3 642 15769 1 _24 ./----\. Klinkenberg,
32
L.：使用生成函数分析概率程序。学位论文，亚琛工业大学，亚琛（2025年）。
https://doi.org/10.18154/RWTH-2025-07571.https://publications.rwth-aachen.de/record/1017915。

33. Klinkenberg, L., Batz, K., Kaminski, B.L., Katoen, J., Moerman, J., Winkler, T.：概率程序的生成函数。载于：LOPSTR。LNCS，第12561卷，第231–248页。施普林格，海德堡（2020年）。https://doi.org/10.1007/978-3-030- 68446-4_1234. Klinkenberg, L., Blumenthal, C., Chen, M., Haase, D., Katoen, J.：使用生成函数对含环概率程序进行精确贝叶斯推断。《ACM程序设计语言会议论文集》8(OOPSLA1)，923–953（2024年）。https : / / doi . org / 10 . 1145/3649844 35. Kozen, D.：概率PDL。《计算机与系统科学杂志》30(2)，162–178（1985年）。https://doi.org/10 1 36 1016 0022 0000 85 90012 . / - ( ) - . Kozen, D.：概率程序语义。《计算机与系统科学杂志》22(3)，328– 350 2 37 1981 10 1016 0022 0000 81 90036 ( )。https://doi.org/ . / - ( ) - . Kuich, W.：半环与形式幂级数：它们对形式语言和自动机的意义。载于：《形式语言手册》，第1卷：词、语言、文法。G. Rozenberg和A. Salomaa编，第609–677页。施普林格（ 1997 10 1007 978 3 642 59136 5 _9 ）。https://doi.org/ . / - - - - \ . Lando, S.K.：生成函数讲义。美国数学学会（2003年）39. Li, J., Zhang, Y.：使用生成函数进行编译。《ACM程序设计语言会议论文集》9 3747534 40 2025 10 1145 (ICFP)（）。https://doi.org/ . / . Lumbroso, J.O.：通过抛硬币实现最优离散均匀生成及其应用。CoRR abs/1304.1916（2013年）。arXiv: 1304.1916. http://arxiv.org/ abs/1304.1916 41. Mahdi Shamsi, S., Pietro Farina, G., Gaboardi, M., Napp：用于建模自主系统的概率编程语言。载于：2020年IEEE多传感器融合与智能系统集成国际会议（MFI），第32 9235230 42 39 2020 10 1109 49285 2020 –（）。https://doi.org/ . /MFI . . . McIver, A., Morgan, C.：概率系统的抽象、精化与证明。施普林格（2005年）43. Mertens, H., Katoen, J., Quatmann, T., Winkler, T.：马尔可夫链中期望访问时间和平稳分布的计算：快速且准确。《自动推理杂志》69 7 44 3 23 2025 10 1007 10817 025 09736（），（）。https://doi.org/ . /S - - - . Meurer, A., Smith, C.P., Paprocki, M., Čertík, O., Kirpichev, S.B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J.K., Singh, S., Rathnayake, T., Vig, S., Granger, B.E., Muller, R.P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., Curry, M.J., Terrel, A.R., Roučka, Š., Saboo, A., Fernando, I., Kulal, S., Cimrman, R., Scopatz, A.：SymPy：Python符号计算。《PeerJ计算机科学》3，103 45 103 2017 10 7717 e（）。https://doi.org/ . /peerj-cs. . Ouaknine, J., Worrell, J.：低阶线性递推数列的正性问题。载于：SODA，第366–379页。SIAM（2014年）。https://doi.org/10.1137/1. 9781611973402.27 46. Park, D.：不动点归纳与程序性质证明。《机器智能》5（1969年）47. Pitman, J.W.：马尔可夫链的占用测度。《应用概率进展》9 1425817 48 1 69 86 1977 10 2307（），–（）。https://doi.org/ . / . Rabehaja, T.M., Sanders, J.W.：具有显式概率性的精化代数。载于：TASE，第63–70页。IEEE计算机学会（2009年）。https://doi.org/10.1109/ TASE.2009.53 49. Sharir, M., Pnueli, A., Hart, S.：概率程序验证。《SIAM计算杂志》13(2)，292–314（1984年）。https://doi.org/10.1137/0213021

50. Takisaka, T., Oyabu, Y., Urabe, N., Hasuo, I.: 随机化程序中用于可达性的排序与排斥超鞅。ACM Trans. Program. Lang. Syst. **43**3450967 51251546 2021101145(), :–:(). https://doi.org/./. Tarski, A.: 格论不动点定理及其应用。Pacific Jour- nal of Mathematics **5**(2), 285–309 (1955). https://doi.org/10.2140/pjm.1955.5.28552. Torres-Ruiz, M., Piedeleu, R., Silva, A., Zanasi, F.: 离散概率编程中的迭代。In: FSCD. LIPIcs, 20:1–20:21. Schloss Dagstuhl - Leibniz- Zentrum für Informatik (202420 5310423020243). https://doi.org/./LIPICS.FSCD.. Van Rossum, G., Drake, F.L.: Python 参考手册。CreateSpace, 斯科茨谷, CA (2009) 54. Wang, P., Yang, T., Fu, H., Li, G., Ong, C.L.: 通过多项式求解对贝叶斯概率编程进行静态后验推断。Proc. ACM Program. Lang. **8**3656432 5513611386 2024101145(PLDI), –(). https://doi.org/./. Wilf, H.S.: 生成函数学。A K Peters/CRC Press (2005) 56. Zaiser, F., Murawski, A.S., Ong, C.L.: 带循环的离散概率程序后验分布的保证界。Proc. ACM Program. Lang. **9**3704874 5711041135 2025101145(POPL), –(). https://doi.org/./. Zaiser, F., Murawski, A.S., Ong, C.L.: 基于概率生成函数的离散模型精确贝叶斯推断：一种概率编程方法。In: NeurIPS (2023). http://papers.nips.cc/paper%5C_files/paper/2023/hash/0747af6f877c0cb555fea595f01b0e83-Abstract-Conference.html58. Zhang, L., Zilberstein, N., Kaminski, B.L., Silva, A.: 定量弱超前置：通过谓词变换器统一正确性与错误性超性质。Proc. ACM Program. Lang. **8**(OOPSLA2), 817–845(2024). https://doi.org/10.1145/3689740

A 证明

为了证明 $\mathrm{lfp}\Phi_{g,C}$ 与大步马尔可夫链的占用测度之间的对应关系，我们将使用 Kleene 不动点定理。这要求我们的语义是序连续的。

命题 2. 如果一个有向集是非空集合 $D \subseteq \mathcal{P}(\mathcal{M}\,(\text{状态}))$，若对所有$\mu_1,\ \mu_2 \in D$，存在 $\mu_3 \in D$使得 $\mu_1 \sqcup \mu_2 \leq \mu_3$。对于 $C \in \mathrm{pGCL}$：$\mathbf{pm}[\![C]\!]$ 是序连续的，即对于每个有向集 $D \subseteq \mathcal{P}(\mathcal{M}(\text{状态}))$：$\sup \mathbf{pm}[\![C]\!]\,(D) = \mathbf{pm}[\![C]\!]\,(\sup D)$。

证明。 由 $\mathbf{pm}$与序连续的 Kozen 语义$\mathcal{K}$ [36]等价可得。

证明（定理 3的证明）。 设 $C = \text{当}(\phi)\,\{C_0\}$ 满足 C_0 无循环。我们对 $k \in \mathbb{N}$进行归纳：对于所有 $g \in \mathcal{M}(\text{状态})$，$s \in$ 状态：

$$(\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot -))^k(g)(s) = \mathbb{E}_{\mathsf{Pr}^g}\left[\lambda\ \pi.\ [\pi_k = s]\right]. \tag{1}$$

对于 $k = 0$:

$$(\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot -))^0(g)(s) = g(s) = \mathsf{Pr}^g(\mathrm{Cyl}(s)) = \mathbb{E}_{\mathsf{Pr}^g}\left[\mathbf{1}_{\mathrm{Cyl}(s)}\right] = \mathbb{E}_{\mathsf{Pr}^g}\left[\lambda\ \pi.\ [\pi_0 = s]\right].$$

对于 $k \to k+1$: 首先，对于 $s, s' \in$ 状态：

$$\begin{aligned}
([\phi]\cdot g)(s')\cdot \mathbf{pm}[\![C_0]\!]\,(\delta_{s'})\,(s) &= \begin{cases} g(s')\cdot \mathbf{pm}[\![C_0]\!]\,(\delta_{s'})\,(s), & \text{if } s' \models \phi, \\ 0, \text{else} \end{cases} \\
&= \begin{cases} g(s')\cdot \mathbf{pm}[\![C_0]\!]\,(\delta_{s'})\,(s), & \text{if } s' \models \phi, \\ g(s')\cdot \delta_{\downarrow}(s), \text{else} \end{cases} && (s \in s) \\
&= g(s')\cdot P[s'](s). && (\text{Def. } P)
\end{aligned}$$

因此，

$$\begin{aligned}
\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot g)\,(s) &= \sum_{s' \in \mathsf{States}} ([\phi]\cdot g)(s')\cdot \mathbf{pm}[\![C_0]\!]\,(\delta_{s'})\,(s) \\
&&& (\mathbf{pm} \text{ is linear and continuous}) \\
&= \sum_{s' \in \mathsf{States}} g(s')\cdot P[s'](s). && (\text{eq. above})
\end{aligned}$$

现在，为了证明归纳步骤，对于 $s \in$ 状态：

$$\begin{aligned}
&(\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot -))^{k+1}(g)(s) \\
&= (\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot -))^k(\mathbf{pm}[\![C_0]\!]\,([\phi]\cdot g))(s) && (\text{unfold}) \\
&= \mathbb{E}_{\mathsf{Pr}^{\mathbf{pm}[\![C_0]\!]([\phi]\cdot g)}}\left[\lambda\ \pi.\ [\pi_k = s]\right] && (\text{IH}) \\
&= \mathbb{E}_{\mathsf{Pr}^{\mathbf{pm}[\![C_0]\!]([\phi]\cdot g)}}\left[\lambda\ \pi.\ \sum_{\bar{\pi} \in \mathsf{States}^k} [\pi_0 = \bar{\pi}_0 \wedge \cdots \wedge \pi_{k-1} = \bar{\pi}_{k-1} \wedge \pi_k = s]\right] \\
&&& (\text{split case on 前缀})
\end{aligned}$$

$$= \sum_{\bar\pi\in\mathsf{States}^k} \mathbb{E}_{\mathsf{Pr}^{\mathbf{pm}[\![C_0]\!]([\phi]\cdot g)}}\left[\lambda\ \pi.\ [\pi_0=\bar\pi_0\wedge\cdots\wedge\pi_{k-1}=\bar\pi_{k-1}\wedge\pi_k=s]\right]$$

($\mathbb{E}[\cdot]$ is linear)

$$= \sum_{\bar\pi\in\mathsf{States}^k} \mathbb{E}_{\mathsf{Pr}^{\mathbf{pm}[\![C_0]\!]([\phi]\cdot g)}}\left[\mathbf{1}_{\mathrm{Cyl}(\bar\pi s)}\right] \qquad \text{(def. cylinder set)}$$

$$= \sum_{\bar\pi\in\mathsf{States}^k} \mathbf{pm}[\![C_0]\!]\left([\phi]\cdot g\right)(\bar\pi_0)\cdot P[\bar\pi_0](\bar\pi_1)\cdot\ldots\cdot P[\bar\pi_{k-2}](\bar\pi_{k-1})\cdot P[\bar\pi_{k-1}](s)$$

(def. induced measure Pr)

$$= \sum_{\bar\pi\in\mathsf{States}^k}\sum_{s'\in\mathsf{States}} g(s')\cdot P[s'](\bar\pi_0)\cdot P[\bar\pi_0](\bar\pi_1)\cdot\ldots\cdot P[\bar\pi_{k-2}](\bar\pi_{k-1})\cdot P[\bar\pi_{k-1}](s)$$

(eq. above)

$$= \sum_{\tilde\pi\in\mathsf{States}^{k+1}} g(\tilde\pi_0)\cdot P[\tilde\pi_0](\tilde\pi_1)\cdot\ldots\cdot P[\tilde\pi_{k-1}](\tilde\pi_k)\cdot P[\tilde\pi_k](s) \qquad \text{(combine sums)}$$

$$= \sum_{\tilde\pi\in\mathsf{States}^{k+1}} \mathsf{Pr}^g(\mathrm{Cyl}(\tilde\pi s)) \qquad \text{(def. } \mathsf{Pr}^g\text{)}$$

$$= \sum_{\tilde\pi\in\mathsf{States}^{k+1}} \mathbb{E}_{\mathsf{Pr}^g}\left[\mathbf{1}_{\mathrm{Cyl}(\tilde\pi s)}\right]$$

$$= \mathbb{E}_{\mathsf{Pr}^g}\left[\lambda\ \pi.\ \sum_{\tilde\pi\in\mathsf{States}^{k+1}} [\pi_0=\tilde\pi_0\wedge\cdots\wedge\pi_k=\tilde\pi_k\wedge\pi_{k+1}=s]\right]$$

($\mathbb{E}[\cdot]$ is linear, def. cylinder set)

$$= \mathbb{E}_{\mathsf{Pr}^g}\left[\lambda\ \pi.\ [\pi_{k+1}=s]\right]. \qquad \text{(merge based on prefix)}$$

归纳至此结束。

现在，令 $g\in\mathcal{M}$(状态)， $s\in$ 状态。由于 $\Phi_{g,\ C}$ 根据命题 2 是序连续的，我们可以应用 Kleene 不动点定理 [16]并得到

$$(\mathrm{lfp}\,\Phi_{g,C})(s) = (\sup_{k\in\mathbb{N}}\Phi^k_{g,C}(0))(s) = \sum_{k\in\mathbb{N}}(\mathbf{pm}[\![C_0]\!]([\phi]\cdot -))^k(g)(s).$$

类似地，根据 [49, 47],

$$\mathrm{OM}^g_{\mathcal{L}[\![C]\!]}(s) = \sum_{k\in\mathbb{N}}\mathbb{E}_{\mathsf{Pr}^g}\left[\lambda\ \pi.\ [\pi_k=s]\right].$$

根据等式 (1)，两个和必定相同，因此我们得出

$$(\mathrm{lfp}\,\Phi_{g,C})(s) = \mathrm{OM}^g_{\mathcal{L}[\![C]\!]}(s).$$

为了证明$\Phi_{g,\ C}$ 与循环的期望运行时间之间的关系，有必要按照程序的结构，对最弱前期望 **wp** 和期望运行时间**ert**进行归纳定义，如表4所示。 **wp** 的定义源自 [42, 29], ， 而 **ert** 则改编自 [30] ， 仅在计算最外层循环守卫时增加运行时间。

表4：定义最弱前期望的规则 $\mathbf{wp}[\![C]\!](f)$ 以及期望运行时间 $\mathbf{ert}[\![C]\!](t)$ 程序 C 相对于后期期望 f：状态 $\to \mathbb{R}^{\infty}_{\geq 0}$ 或延续时间 t：状态 $\to \mathbb{R}^{\infty}_{\geq 0}$。

程序 C	$\mathbf{wp}[\![C]\!](f)$	期望运行时间 $[\![C]\!](t)$
skip	f	t
发散	0	∞
$\{C_1\}\ [p]\ \{C_2\}$	$p \cdot \mathbf{wp}[\![C_1]\!](f) + (1-p) \cdot \mathbf{wp}[\![C_2]\!](f)$	$p \cdot$ 期望运行时间$[\![C_1]\!](t) + (1-p) \cdot$ 期望运行时间$[\![C_2]\!](t)$
$x := E$	$f[x/E]$	$t[x/E]$
$x :\approx \mu$	$\lambda s.\ \sum_v f(s[x/v]) \cdot\ \mu[s](v)$	$\lambda s.\ \sum_v t(s[x/v]) \cdot\ \mu(s)(v)$
$C_1 \fatsemi C_2$	$\mathbf{wp}[\![C_1]\!](\mathbf{wp}[\![C_2]\!](f))$	期望运行时间$[\![C_1]\!]$(期望运行时间$[\![C_2]\!](t)$)
如果 $(\phi)\ \{C_1\}$ 否则$\{C_2\}$	$[\phi] \cdot \mathbf{wp}[\![C_1]\!](f) + [\neg\phi] \cdot \mathbf{wp}[\![C_2]\!](f)$	$[\phi] \cdot$ 期望运行时间$[\![C_1]\!](t) + [\neg\phi] \cdot$ 期望运行时间$[\![C_2]\!](t)$
当$(\phi)\ \{C_0\}$	$\mathrm{lfp}\, X.\ [\neg\phi] \cdot f + [\phi] \cdot \mathbf{wp}[\![C_0]\!](X)$	$\mathrm{lfp}\, X.\ 1 + [\neg\phi] \cdot t + [\phi] \cdot \mathbf{wp}[\![C_0]\!](X)$

证明（定理4的证明）。根据表4中对ert的归纳定义，需要证明的命题变为：

For $C = \texttt{while}\,(\phi)\,\{C_0\}$:

$$\forall \text{ initial measures } g \in \mathcal{M}(\text{States}): \qquad \left|\mathrm{lfp}\,\Phi_{g,C}\right| = \sum_{s \in \text{States}} \mathbf{ert}[\![C]\!](0)(s) \cdot g(s)\,.$$

令 $C = $ 当$(\phi)\ \{C_0\}$ 和 $g \in \mathcal{M}$(状态)。
利用 pm-wp Kozen 对偶性，可以通过对 $k \in \mathbb{N}$进行归纳证明：

$$\mathbb{E}_{(\mathrm{pm}[\![C_0]\!]([\phi]\cdot -))^k(g)}[1] = \mathbb{E}_g\left[([\phi] \cdot \mathbf{wp}[\![C_0]\!](\cdot))^k(1)\right]. \tag{2}$$

对于 t：状态 $\to \mathbb{R}^{\infty}_{\geq 0}$，循环特征函数 $\Psi_{t,\ C}(X) = 1 + [\neg\phi] \cdot t + [\phi] \cdot \mathbf{wp}[\![C_0]\!](X)$ 的 **期望运行时间** 是序连续的 [30]。我们应用 Kleene 不动点定理 [16] 并得到

$$\mathbf{ert}[\![C]\!](t) = \sup_{k \in \mathbb{N}} \Psi^k_{t,C}(0) = \sum_{k \in \mathbb{N}} ([\phi] \cdot \mathbf{wp}[\![C_0]\!](\cdot))^k (1 + [\neg\phi] \cdot t) \tag{3}$$

同理，$\Phi_{g,\ C}$ 是序连续的，由命题 2，再次利用 Kleene 不动点定理可得

$$\begin{aligned}
\left|(\mathrm{lfp}\,\Phi_{g,C})\right| &= \mathbb{E}_{\mathrm{lfp}\,\Phi_{g,C}}[1] = \mathbb{E}_{\sup_{k\in\mathbb{N}} \Phi^k_{g,C}(0)}[1] \\
&= \mathbb{E}_{\sum_{k\in\mathbb{N}} (\mathrm{pm}[\![C_0]\!]([\phi]\cdot -))^k(g)}[1] \\
&= \sum_{k\in\mathbb{N}} \mathbb{E}_{(\mathrm{pm}[\![C_0]\!]([\phi]\cdot -))^k(g)}[1] && (\mathbb{E}[\cdot] \text{ is linear}) \\
&= \sum_{k\in\mathbb{N}} \mathbb{E}_g\left[([\phi] \cdot \mathbf{wp}[\![C_0]\!](\cdot))^k(1)\right] && (\text{eq. } (2)) \\
&= \mathbb{E}_g\left[\sum_{k\in\mathbb{N}} ([\phi] \cdot \mathbf{wp}[\![C_0]\!](\cdot))^k(1)\right] && (\mathbb{E}[\cdot] \text{ is linear})
\end{aligned}$$

$$= \mathbb{E}_g\left[\textbf{期望运行时间}[\![C]\!]\right] \qquad \text{（等式（3）中含有 } t = 0\text{）}$$
$$(0)$$
$$= \sum_{s \in \text{状态}} \textbf{期望运行时间}[\![C]\!](0)(s) \cdot g(s)。 \qquad \text{（定义。} \mathbb{E}[\cdot]\text{）}$$

程序 3 的计算 $\mathbf{pm}[\![P_N]\!]\,(I_N \restriction_{f=0})$。

```
/* ∑_{1≤i<N, ∃k: i≡_N 2^k} a_{N,i} · V^i (1−C^i)/(1−C) */
v := 2v;
/* ∑_{1≤i<N, ∃k: i≡_N 2^k} a_{N,i} · V^{2i} (1−C^i)/(1−C) */
{c := 2c} [1/2] {c := 2c + 1};
/* 1/2 · ∑_{1≤i<N, ∃k: i≡_N 2^k} a_{N,i} · V^{2i} (1−C^{2i})/(1−C^2) + 1/2 · C ∑_{1≤i<N, ∃k: i≡_N 2^k} a_{N,i} · V^{2i} (1−C^{2i})/(1−C^2) */
/* = ∑_{1≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i} (1−C^{2i})/(1−C) */
如果(N ≤ v){
    /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i} (1−C^{2i})/(1−C) */
    如果 (c < N){
        /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i} (1−C^N)/(1−C) */
        f := f + 1
        /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · F V^{2i} (1−C^N)/(1−C) */
    } 否则{
        /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i} (C^N − C^{2i})/(1−C) */
        v := v − N;
        /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i−N} (C^N − C^{2i})/(1−C) */
        c := c − N
        /* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i−N} (1−C^{2i−N})/(1−C) */
    }
}
/* ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · F V^{2i} (1−C^N)/(1−C) + ∑_{N/2≤i<N, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i−N} (1−C^{2i−N})/(1−C) + ∑_{1≤i<N/2, ∃k: i≡_N 2^k} 1/2 a_{N,i} · V^{2i} (1−C^{2i})/(1−C) */
```

B 分析快速骰子滚轮

回想示例中构造的模板5，它用于快速骰子滚轮 $N \in \mathbb{N},\ N > 1$，其循环体为 P_N：

$$I_N = \sum_{\substack{1 \le i < N \\ \exists k\colon i \equiv_N 2^k}} a_{N,i} \cdot V^i \frac{1 - C^i}{1 - C} + \sum_{\substack{N \le i < 2N \\ \exists k\colon i \equiv_N 2^k}} a_{N,i} \cdot F\, V^i \frac{1 - C^N}{1 - C}.$$

我们希望证明，这可以通过实例化得到一个占用测度不变量。我们首先计算将循环体应用于

$$I_N \restriction_{f=0} = \sum_{\substack{1 \le i < N \\ \exists k\colon i \equiv_N 2^k}} a_{N,i} \cdot V^i \frac{1 - C^i}{1 - C}.$$

按照程序 3 中的计算，我们得到

$$\begin{aligned}
\mathfrak{g}_0 + \mathbf{pm}[\![P_N]\!]\left(I_N \restriction_{f=0}\right) &= 1 \cdot V^1 C^0 F^0 && (4)\\
&\quad + \sum_{\substack{1 \le i < \frac{N}{2} \\ \exists k\colon i \equiv_N 2^k}} \frac{1}{2} a_{N,i} \cdot V^{2i} \frac{1 - C^{2i}}{1 - C} && (5)\\
&\quad + \sum_{\substack{\frac{N}{2} \le i < N \\ \exists k\colon i \equiv_N 2^k}} \frac{1}{2} a_{N,\ i} \cdot V^{2i-N} \frac{1 - C^{2i-N}}{1 - C} && (6)\\
&\quad + \sum_{\substack{\frac{N}{2} \le i < N \\ \exists k\colon i \equiv_N 2^k}} \frac{1}{2} a_{N,i} \cdot F\, V^{2i} \frac{1 - C^N}{1 - C}. && (7)
\end{aligned}$$

对于 $I_N = \mathfrak{g}_0 + \mathbf{pm}[\![P_N]\!]\left(I_N \restriction_{f=0}\right)$，我们比较各个加项的系数，从而得到以下约束，其中我们将 $l \in \mathfrak{P}_{2,N}$ 记作 $l \in \mathbb{N}$，并将$\exists k$ 记作 $l \equiv_N 2^k$：

$$a_{N,1} = \underbrace{1}_{(4)} + \underbrace{\left[\frac{1+N}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{1+N}{2}}}_{(6)}, \tag{A}$$

$$a_{N,i} = \underbrace{\left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{i}{2}}}_{(5)} + \underbrace{\left[\frac{i+N}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{i+N}{2}}}_{(6)} \qquad \text{if } 1 < i < N, i \in \mathfrak{P}_{2,N}, \tag{B}$$

$$a_{N,i} = \underbrace{\left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{i}{2}}}_{(7)} \qquad \text{if } N \le i < 2N, i \in \mathfrak{P}_{2,N}. \tag{C}$$

(8)

这是一个线性方程组。通过研究 v 在每次循环迭代后的取值序列，可以找到参数的一个实例化，使其成为这些方程的解，进而得到一个有效的不变式。关键在于， v 始终是模 N意义下的二的幂。从 $v = 1$开始，它的值翻倍，并通过减去 N 使其保持在 $0 < v < N$ 中；否则循环将以 $f = 1$终止。

定义 16。 将模 N 的二的幂序列记为 $\pi_N := (2^k \bmod N)_{k\in\mathbb{N}}$。

由于模 N 的剩余类只有有限个，这个序列必然会出现重复元素。事实上，模 N 乘以二是确定性的，因此序列中的每个元素都唯一决定其后继元素：

命题 3。对于 $k \in \mathbb{N}$： $\pi_{N,\,k+1} = (2 \cdot \pi_{N,\,k}) \bmod N$。

因此， π_N 必定呈套索形式，即在初始前缀之后出现重复模式：$\pi_N = \pi_{N,\,\mathrm{init}}(\pi_{N,\,\mathrm{loop}})^{\omega}$ ，其中$\pi_{N,\,\mathrm{init}}$，$\pi_{N,\,\mathrm{loop}}$ $\in \{0, N$为某些有限值$)^*$。这种分解并非唯一。

直观地说，只要满足 $f = 0$，这个序列就对应于 v的取值。在每次循环迭代中，为 c 的新取值抛硬币，会使 v 下一个取值的概率减半。由于循环从分布 $\mathfrak{g}_0 = 1 \cdot V^1C^0F^0$ 开始，其中 v的唯一取值为 1，因此 v 的取值 i 的概率取决于：1. 首次到达 i 所需的步数；2. 如果 i 重复出现（即出现无限多次），则取决于 i 连续两次出现之间的步数。

我们引入一些记号来描述这些概念。

定义 17。对于 $i \in \mathbb{N}$ 令

– $F_N(i) := inf\{k \in \mathbb{N} \mid i \equiv_N 2^k\} \in \mathbb{N} \cup \{\infty\}$ 表示满足以下条件的最小指数 k： i 与 2^k 模 N 同余；若这样的 k 存在，否则为 ∞；– $L_N(i) := inf\{l \in \mathbb{N}_+ \mid i \equiv_N i \cdot 2^l\} \in \mathbb{N}_+ \cup \{\infty\}$ 表示满足以下条件的最小正指数 l： $i \cdot 2^l$ 与 i 模 N 同余；若这样的 l 存在，否则为∞。

直观而言，对于 $i \in [0, N)$，$\mathrm{L}_N(i)$ 是在 π_N中， i每次出现之间的周期，起始偏移量为 $\mathrm{F}_N(i)$。

这使我们能够正式给出该线性方程组的解。

定义 18。 对于 $N \in \mathbb{N}, N > 1$，其中 $\mathcal{T}_N = \{a_{N,\,i} \mid i \in \mathfrak{P}_{2,\,N}, 1 \le i < 2N\}$，令$\tau_N$： $\mathcal{T}_N \to \mathbb{R}$ 满足

$$\tau_N(a_{N,i}) := \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)} \cdot \frac{1}{1 - \left(\frac{1}{2}\right)^{\mathrm{L}_N(i)}} \qquad \text{if } 1 \le i < N, i \in \mathfrak{P}_{2,N},$$

$$\tau_N(a_{N,i}) := \left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2}\tau_N(a_{N,\frac{i}{2}}) \qquad \text{if } N \le i < 2N, i \in \mathfrak{P}_{2,N}.$$

其中我们使用 $\left(\frac{1}{2}\right)^{\infty} = 0$。

为了证明该实例化是一个解，我们将证明与模 N 的二次幂序列 π_N 相关的一些性质。首先，如果某些数字在 π_N 中重复出现，它们具有相同的周期。

命题 4. 对于 $i, j \in [0, N)$：若 $\mathrm{F}_N(i), \mathrm{F}_N(j) \in \mathbb{N}$ 且 $\mathrm{L}_N(i), \mathrm{L}_N(j) \in \mathbb{N}_+$，则 $\mathrm{L}_N(i) = \mathrm{L}_N(j)$。

证明。若 $\mathrm{F}_N(i) = \mathrm{F}_N(j)$，则 $i \equiv_N 2^{\mathrm{F}_N(i)} \equiv_N j$，因此必有 $i = j$，因为$i, j \in [0, N)$。所以该陈述显然成立。

否则，不失一般性，假设 $\mathrm{F}_N(i) < \mathrm{F}_N(j)$。由于 $\mathrm{L}_N(i) \in \mathbb{N}_+$，利用 π_N (命题 3) 的确定性构造，我们有

$$\pi_N = (\pi_{N,0} \cdots \pi_{N,\mathrm{F}_N(i)-1})(\pi_{N,\mathrm{F}_N(i)} \cdots \pi_{N,\mathrm{F}_N(i)+\mathrm{L}_N(i)-1})^{\omega}.$$

因此，$\mathrm{F}_N(j) < \mathrm{F}_N(i) + \mathrm{L}_N(i)$，也就是说，$j$出现在$\pi_N$的重复部分中。设 $k := \mathrm{F}_N(j) - \mathrm{F}_N(i) > 0$表示$_N$ j进入重复部分的偏移量，也就是说，F $\equiv_N \pi_{N\,\mathrm{F}_N\,j\,+\,N\,i} = \pi_{N\,\mathrm{F}_N\,i\,+k+\,N\,i} = \pi_{N\,\mathrm{F}_N\,i\,+k} = \pi_{N\,\mathrm{F}_N\,j} \equiv_N 2^{\mathrm{F}_N\,j} \equiv_N j$ (j) F (i)+k。但此时同样有 $2^{\mathrm{F}_N}(j)+\mathrm{L}_N(i)$, ( ) L ( ) , ( ) L ( ) , ( ) , ( ) ( )，从而得出 j的周期上界：$\mathrm{L}_N(j) \le \mathrm{L}_N(i)$。

注意到我们可以将 $\mathrm{L}_N(i) = k + (\mathrm{L}_N(i) - k)$ 写成，其中 k 和 $\mathrm{L}_N(i) - k$ 均为非负数。于是

$$\begin{aligned}
i \cdot 2^{\mathrm{L}_N(j)} &\equiv_N 2^{\mathrm{F}_N(i)+\mathrm{L}_N(j)} \\
&\equiv_N 2^{\mathrm{F}_N(i)+\mathrm{L}_N(i)+\mathrm{L}_N(j)} && \text{(def. of } \mathrm{L}_N(i)) \\
&\equiv_N 2^{\mathrm{F}_N(i)+k+(\mathrm{L}_N(i)-k)+\mathrm{L}_N(j)} \equiv_N 2^{\mathrm{F}_N(i)+k+\mathrm{L}_N(j)+(\mathrm{L}_N(i)-k)} && \text{(rearrange)} \\
&\equiv_N 2^{\mathrm{F}_N(j)+\mathrm{L}_N(j)+(\mathrm{L}_N(i)-k)} \equiv_N 2^{\mathrm{F}_N(j)+(\mathrm{L}_N(i)-k)} && \text{(def. of } k, \mathrm{L}_N(j)) \\
&\equiv_N 2^{\mathrm{F}_N(j)-k+\mathrm{L}_N(i)} \equiv_N 2^{\mathrm{F}_N(i)+\mathrm{L}_N(i)} \equiv_N i. && \text{(def. of } k, \mathrm{L}_N(i))
\end{aligned}$$

因此，根据定义，由于 $\mathrm{L}_N(i)$ 是 i 的最小周期，$\mathrm{L}_N(j) \ge \mathrm{L}_N(i)$。

结合这两个不等式，我们得出 $\mathrm{L}_N(i) = \mathrm{L}_N(j)$。

正如序列 π_N 中各元素的后继是唯一确定的一样，它们的前驱也几乎是唯一的。执行一次循环时，新值 v 上的测度会从这些前驱值收集质量。我们使用一个人工元素 $\bullet$ 来表示 1 的一个前驱。对我们的目的而言，这将用于表示从初始分布$\mathfrak{g}_0 = V^1$ 进入循环的概率质量，该初始分布是集中在 $v = 1$ 上的点质量。

定义 19。对于 $i \in [0, N)$ 令 $\mathrm{pre}_N(i) := \{x \in [0, N) \mid \exists k,\ \pi_{N,\,k} = x \wedge \pi_{N,\,k+1} = i\} \cup \{\bullet \mid i = 1\}$。

刻画该前驱集合是证明 τ_N 为线性方程组解的关键步骤。下面的引理总结了从 π_N的最小套索分解中容易观察到的重要性质。

定义20. π_N 的最小套索分解为以下分解

$$\pi_N = (\pi_{N,0}\cdots\pi_{N,S_N-1})(\pi_{N,S_N}\cdots\pi_{N,S_N+L_N-1})^\omega$$

使得 $S_N \in \mathbb{N}$ 是最小的，并且对于该 S_N， $L_N \in \mathbb{N}_+$ 是最小的。

引理1。对于 $i \in [0,N)$：

1. $\mathrm{F}_N(i) < S_N + L_N$ 或 $\mathrm{F}_N(i) = \infty$， 2. $\mathrm{F}_N(i) < S_N \Longrightarrow \mathrm{L}_N(i) = \infty$，
3. $S_N \le \mathrm{F}_N(i) < S_N + L_N \Longrightarrow \mathrm{L}_N(i) = L_N$， 4. $\mathrm{pre}_N(1) \subseteq \{\bullet, \frac{1+N}{2}\}$，
并且当 $i \neq 1$ 时，$\mathrm{pre}_N(i) \subseteq \{\frac{i}{2}, \frac{i+N}{2}\}$， 5. $|\mathrm{pre}_N(i)| = \begin{cases} 1 & \text{；如果} \\ & \end{cases}$

$\mathrm{F}_N(i) < S_N$，2 $_N\, i = S_N$ ；如果 F ( )，1 $S_N <$ $_N\, i$ ；
如果 F ( ) $< S_N + L_N$，0 $i = \infty.$ ；如果 F_N( )

证明。1. 假设 F$\pi_{N\,\mathrm{F}_N\,i-L_N} \equiv_N 2^{\mathrm{F}_N\,i-L_N}$ $\equiv_N \pi_{N\,\mathrm{F}_N\,i} =$ $_N\, i \le$ $_N\, i - L_N$ $L_N > 0$

$(i) \in \mathbb{N}$，F $(i) \ge S_N + L_N$。那么 $i \equiv_N 2^{\mathrm{F}_N(i)}$，( )，()()。因此，F ( ) F ( )。这是一个矛盾。

2. 假设 $\mathrm{F}_N(i) < S_N$ 且 $\mathrm{L}_N(i) \in \mathbb{N}_+$。那么套索分解

$$\pi_N = (\pi_{N,0}\cdots\pi_{N,\mathrm{F}_N(i)-1})(\pi_{N,\mathrm{F}_N(i)}\cdots\pi_{N,\mathrm{F}_N(i)+\mathrm{L}_N(i)-1})^\omega$$

与 S_N的最小性相矛盾。

3. 我们有 $i \equiv_N 2^{\mathrm{F}_N}(i) \equiv_N \pi_{N,\mathrm{F}_N(i)} = \pi_{N,\mathrm{F}_N(i)+L_N} \equiv_N 2^{\mathrm{F}_N}(i)+L_N$。因此，$\mathrm{L}_N(i)$ $\le L_N$。设 $j \in [0,N)$，且 $\mathrm{F}_N(j) = S_N$。那么，根据命题 4，我们有 $\mathrm{L}_N(i) = \mathrm{L}_N$ (j)。现在，$\mathrm{L}_N(j) = L_N$，因为如果 $\mathrm{L}_N(j) < L_N$，那么套索分解$\pi_N = (\pi_N,0\cdots$ $\pi_{N,S_N-1})(\pi_{N,S_N}\cdots\pi_{N,S_N+\mathrm{L}_N(j)-1})^\omega$ 就与 L_N 的最小性相矛盾。

4. 设 $x \in \mathrm{pre}_N(i)$，其中 $x \in [0,N)$。根据命题 3 $i = (2\cdot x) \bmod N$。由于$x \in [0,N)$、$2x \in [0,2N)$，因此要么 $i = 2x$，要么 $i = 2x - N$，即要么 $x = \frac{i}{2}$，要么$x = \frac{i+N}{2}$。 –

因此，$\mathrm{pre}_N(i) \subseteq \{\frac{i}{2}, \frac{i+N}{2}, \bullet\}$。最后，如果 $i = 1$，那么 $\frac{1}{2} \notin \mathbb{N}$，因此 $\frac{1}{2} \notin \mathrm{pre}_N$ (1)；而如果 $i \neq 1$，那么 $\bullet \notin \mathrm{pre}_N(i)$。

5. 首先，注意到根据（1），分类讨论涵盖了所有可能的情况。
如果 $\mathrm{F}_N(i) = \infty$， $i \neq 1$ 由于 $\pi_{N,\,0} = 2^0 = 1$。因此，如果 $\mathrm{pre}_N(i) \neq \emptyset$，则必定存在 $x \in [0,N)$ $k \in \mathbb{N}$，使得 $x \in \mathrm{pre}_N(i)$，$\pi_{N,\,k} = x$， $\pi_{N,\,k+1} = i$。但这样一来，$\mathrm{F}_N(i) \le k+1$，这就产生了矛盾。因此，确实有 $\mathrm{pre}_N(i) = \emptyset$。
现在，考虑 $\mathrm{F}_N(i) < S_N + L_N$ 的情况。如果 $\mathrm{F}_N(i) = 0$，那么 $i = 1$ 和 $\bullet \in \mathrm{pre}_N(i)$。否则，F $_N\, i > 0$ $2\,\pi_{N\,\mathrm{F}_N\,i-1}$ $N = 2$ N $N = ( )$，那么 (·, ()) $2^{\mathrm{F}_N\,i}$ $N = \pi_{N\,\mathrm{F}_N\,i} = i$ $\pi_{N\,\mathrm{F}_N\,i-1} \in$ $_N\, i$ mod (·($2^{\mathrm{F}_N(i)-1}$ mod)) mod () mod , (), 得到 , ()pre ( )。在这两种情况下，$|\mathrm{pre}_N(i)| \ge 1$。

情形 $\mathrm{F}_N(i) < S_N$：反证，假设 $|\mathrm{pre}_N(i)| > 1$。那么存在 $k > \mathrm{F}_N(i)$，使得 π_N, $_k = i$。因此，$\mathrm{L}_N(i) \le k - \mathrm{F}_N(i)$，这与 (2) 矛盾。

情形 $\mathrm{F}_N(i) = S_N$**：** 反证，假设 $|\mathrm{pre}_N(i)| = 1$，令 x 表示 $\mathrm{pre}_N(i)$ 的唯一元素。如果 $i = 1$，那么 $\mathrm{F}_N(i) = 0$ 且 $x = \bullet$。但是$1 = \pi_{N,\,0} = \pi_{N,\,0+L_N} = (2 \cdot \pi_{N,\,L_N-1}) \bmod N$，即 $\pi_{N,\,L_N-1} \in \mathrm{pre}_N(1)$。但是

- $\bullet \neq \pi_{N,\,L_N-1} \in [0, N)$，一个矛盾。

如果 $i \neq 1$，那么 $\mathrm{F}_N(i) > 0$，并且我们有 $x = \pi_{N,\mathrm{F}_N(i)-1} = \pi_{N,\mathrm{F}_N(i)+L_N-1}$。但此时 $\mathrm{F}_N(x) < \mathrm{F}_N(i) = S_N$，且 $\mathrm{L}_N(x) \leq L_N$ 与 (2) 矛盾。

情形 $S_N < \mathrm{F}_N(i) < S_N + L_N$**：** 由于 $\mathrm{F}_N(i) > S_N \geq 0$，我们必有 $i \neq 1$。根据 π_N 的套索分解，我们有

$$\pi_N = (\pi_{N,0} \cdots \pi_{N,S_N-1})(\pi_{N,S_N} \cdots \pi_{N,\mathrm{F}_N(i)} \cdots \pi_{N,S_N+L_N-1})^\omega .$$

由于该分解具有最小性，对于所有满足 π_N 的 $k \in \mathbb{N}$，我们有 $k = \mathrm{F}_N(i) + l_k \cdot L_N - 1$，其中某个 $l_k \in \mathbb{N}$。根据 (3)，由于 $\mathrm{L}_N(i) = L_N$，我们有 $\pi_{N,k} \equiv_N 2^k \equiv_N 2^{\mathrm{F}_N(i)+l_k \cdot L_N-1} \equiv_N 2^{\mathrm{F}_N(i)-1}$，即 $\pi_{N,k} = (2^{\mathrm{F}_N(i)-1}) \bmod N$。由于对于所有 $x \in \mathrm{pre}_N(i)$，都存在一个 $k \in \mathbb{N}$，使得 $\pi_{N,k} = x, \pi_{N,k+1} = i$，因此我们有 $x = (2^{\mathrm{F}_N(i)-1}) \bmod N$，即 $\mathrm{pre}_N(i) = \{x\}$。

推论 3。对于 $N \in \mathbb{N}$，$N > 1$，τ_N 是方程组的解，方程组由等式 (A)至 (C)。

证明。方程 (A)：我们根据 $\binom{1+N}{2} \in \mathfrak{P}_2$、$N$ 进行分类讨论。如果~~$\binom{1+N}{2} \in \mathfrak{P}_2$，$N$，那么~~ $\binom{1+N}{2} \in \mathrm{pre}_N$~~(1)~~，因为 $\binom{1+N}{2} < N$。~~由于~~ $\bullet \in \mathrm{pre}_N(1)$，我们也有$S_N = \mathrm{F}_N(1) = 0$，根据引理1 (5)。因此，根据最小套索分解，我们有 $\mathrm{F}_N(\binom{1+N}{2}) = L_N - 1$，并且 $\mathrm{L}_N(\binom{1+N}{2}) = \mathrm{L}_N\overline{(1)} = L_N$，根据引理1 (3)。我们计算

$$\begin{aligned}
1 + \left[\frac{1+N}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2}\tau_N(a_{N,\frac{1+N}{2}}) &= 1 + \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(\frac{1+N}{2})} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(\frac{1+N}{2})}} \\
&= 1 + \frac{1}{2}\left(\frac{1}{2}\right)^{L_N-1} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} \\
&= 1 + \left(\frac{1}{2}\right)^{L_N} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} \\
&= \frac{1-\left(\frac{1}{2}\right)^{L_N} + \left(\frac{1}{2}\right)^{L_N}}{1-\left(\frac{1}{2}\right)^{L_N}} \\
&= \frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} \\
&= \left(\frac{1}{2}\right)^{\mathrm{F}_N(1)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(1)}} \\
&= \tau_N(a_{N,1}).
\end{aligned}$$

否则 $\binom{1+N}{2} \notin \mathfrak{P}_{2,\,N}$，则 $\binom{1+N}{2} \notin$ 前置$_N(1)$。根据引理 1 (5)，因此 $0 = \mathrm{F}_N(1) < S_N$。由引理 1 (2) 可得 $\mathrm{L}_N(1) = \infty$，并计算

$$1 + \left[\frac{1+N}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2}\tau_N(a_{N,\frac{1+N}{2}}) = 1$$

$$\begin{aligned} &= \left(\frac{1}{2}\right)^{\mathrm{F}_N(1)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\infty}} \qquad (\text{recall } (1/2)^{\infty}=0) \\ &= \left(\frac{1}{2}\right)^{\mathrm{F}_N(1)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(1)}} \\ &= \tau_N(a_{N,1}). \end{aligned}$$

方程（B）： 与前一种情况类似，我们利用以下事实：对于$x \in \{\frac{i}{2},\ \frac{i+N}{2}\}$有$x \in \mathfrak{P}_{2,\ N} \Longleftrightarrow x \in$前置$_N(i)$，因为 $x < N$。这使我们能够将公式（B）改写为：

$$a_{N,i} = \left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{i}{2}} + \left[\frac{i+N}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} a_{N,\frac{i+N}{2}} = \sum_{x \in \mathrm{pre}_N(i)} \frac{1}{2} a_{N,x}.$$

情形 $\mathrm{F}_N(i) < S_N$和情形 $S_N < \mathrm{F}_N(i) < S_N + L_N$：令 $x \in \mathrm{pre}_N(i)$ 表示由引理1（5）得到的 i 的唯一前驱。由最小套索分解可得 $\mathrm{F}_N(x) = \mathrm{F}_N(i) - 1$，并且

– 根据引理1（2），若 $\mathrm{F}_N(i) < S_N$，则 $\mathrm{L}_N(x) = \mathrm{L}_N(i) = \infty$；或 – 根据引理1（3），若 $S_N < \mathrm{F}_N(i) < S_N + L_N$，则 $\mathrm{L}_N(x) = \mathrm{L}_N(i) = L_N$。

在这两种情形下，我们计算

$$\begin{aligned} \sum_{x' \in \mathrm{pre}_N(i)} \frac{1}{2} \tau_N(a_{N,x'}) &= \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(x)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(x)}} \\ &= \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(i)-1} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(i)}} \\ &= \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(i)}} \\ &= \tau_N(a_{N,i}). \end{aligned}$$

情形 $\mathrm{F}_N(i) = S_N$：令 x_1、$x_2 \in \mathrm{pre}_N(i)$ $x_1 \neq x_2$ 表示根据引理1（5）得到的 i 的两个不同前驱。不失一般性地（通过交换 x_1、x_2），由最小套索分解可得 $\mathrm{F}_N(x_1) = \mathrm{F}_N(i) - 1$ 且 $\mathrm{F}_N(x_2) = \mathrm{F}_N(i) + L_N - 1$。此外，

$-\mathrm{L}_N(x_1) = \infty$ 由引理 1 (2) 得到，以及$-\mathrm{L}_N(x_2) = \mathrm{L}_N(i) = L_N$由引理 1 (3) 得到。我们计算

$$\begin{aligned} &\sum_{x' \in \mathrm{pre}_N(i)} \frac{1}{2} \tau_N(a_{N,x'}) \\ &= \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(x_1)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(x_1)}} + \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(x_2)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(x_2)}} \\ &= \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(i)-1} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\infty}} + \frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(i)+L_N-1} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} \end{aligned}$$

$$
\begin{aligned}
&= \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)} + \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)+L_N} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} && ((1/2)^\infty = 0)\\
&= \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)} \cdot \frac{1-\left(\frac{1}{2}\right)^{L_N}+\left(\frac{1}{2}\right)^{L_N}}{1-\left(\frac{1}{2}\right)^{L_N}}\\
&= \left(\frac{1}{2}\right)^{\mathrm{F}_N(i)} \cdot \frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(i)}}\\
&= \tau_N(a_{N,i}).
\end{aligned}
$$

等式 (C)： 这直接由 τ_N的定义得出。

引理 2. 对于 $N \in \mathbb{N}$，FDR_N 是 *PAST*。

证明。 对于 $N = 1$，很容易验证 $I_1 := 2 \cdot \mathfrak{g}_0 + FV^2 = 2V + FV^2$ 是一个占用不变式。由于 $|I_1| = 3 < \infty$，循环因此 FDR_1 根据推论 1，是 PAST。

对于 $N > 1$，将模板 I_N 与 τ_N 结合，可根据推论 3得到一个有效的不变式。由于

$$
\begin{aligned}
|I_N[\tau_N]| &= \sum_{\substack{1\le i<N\\ \exists k:\, i\equiv_N 2^k}} \left|\tau_N(a_{N,i})\right| \cdot \underbrace{\left|V^i \frac{1-C^i}{1-C}\right|}_{=i<N} + \sum_{\substack{N\le i<2N\\ \exists k:\, i\equiv_N 2^k}} \left|\tau_N(a_{N,i})\right| \cdot \underbrace{\left|FV^i \frac{1-C^N}{1-C}\right|}_{=N}\\
&\le \sum_{\substack{1\le i<N\\ \exists k:\, i\equiv_N 2^k}} \underbrace{\left(\frac{1}{2}\right)^{\mathrm{F}_N(i)}}_{\le 1} \cdot \underbrace{\frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(i)}}}_{\le 2} \cdot N + \sum_{\substack{N\le i<2N\\ \exists k:\, i\equiv_N 2^k}} \underbrace{\left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2}\tau_N(a_{N,\frac{i}{2}})}_{\le 1} \cdot N\\
&\le N \cdot 2N + N \cdot 1N = 3N^2 < \infty,
\end{aligned}
$$

我们可以得出结论，循环因此 FDR_N 是 PAST。

占用不变式 I_N 提供了一个上界

$$
\mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V,F/1] \le (I_N[\tau_N]\restriction_{f>0})[V,F/1] = \sum_{\substack{N\le i<2N\\ \exists k:\, i\equiv_N 2^k}} \tau_N(a_{N,i}) \cdot \frac{1-C^N}{1-C}.
$$

虽然这是 c 在 0 与 $N-1$ 之间的均匀测度，但未知的总质量带来了太多歧义，无法保证 $\mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V,F/1]$ 服从均匀分布。为弥合这一差距，我们证明 $\sum_{\substack{N\le i<2N\\ \exists k:\, i\equiv_N 2^k}} \tau_N(a_{N,i}) = \frac{1}{N}$。我们通过以下方式实现这一点证明该总和对应于 $\overline{\frac{1}{N}}$ 的二进制表示的一种闭合表示。

其中 $\{\cdot\}: \mathbb{R}_{\ge 0} \to [0,1)_{\mathbb{R}}$ 表示非负实数的小数部分，$\overline{\frac{1}{N}}$ 的二进制表示为

$$
\frac{1}{N} = \sum_{k\in\mathbb{N}} \left[\left\{\frac{1}{N}\cdot 2^k\right\} \ge \frac{1}{2}\right] \cdot \frac{1}{2^{k+1}}.
$$

我们可以将各加数直接与余数序列 π_N 联系起来。每当序列在加倍步骤中溢出，即 $2\pi_{N,k} \geq N$，二进制表示中就会包含 $\frac{1}{2^{k+1}}$。

命题5。 对于 $k \in \mathbb{N}$、$\left\{\frac{1}{N} \cdot 2^k\right\} = \pi_{N,\,N^k}$。

证明。对 $k \in \mathbb{N}$ 进行归纳：对于 $k = 0$、$\left\{\frac{1}{N} \cdot 2^0\right\} = \frac{1}{N} = \frac{2^0 \text{ 模 } N}{N} = \pi_{N,\,N^0}$。现在，令 $k \in \mathbb{N}$，使得该陈述成立。于是对于 $k+1$:

$$
\begin{aligned}
&\left\{\frac{1}{N} \cdot 2^{k+1}\right\} \\
&= 2 \cdot \left\{\frac{1}{N} \cdot 2^k\right\} - \left[2 \cdot \left\{\frac{1}{N} \cdot 2^k\right\} \geq 1\right] && \text{(inductive structure of } \{\cdot\}) \\
&= 2 \cdot \frac{\pi_{N,k}}{N} - \left[2 \cdot \frac{\pi_{N,k}}{N} \geq 1\right] && \text{(IH)} \\
&= \frac{2 \cdot \pi_{N,k} - \left[2 \cdot \pi_{N,k} \geq N\right] \cdot N}{N} \\
&= \frac{\pi_{N,k+1}}{N}. && \text{（的归纳结构，命题 3）}
\end{aligned}
$$

引理 3。 $\sum_{\substack{N \leq i < 2N \\ \exists k:\ i \equiv_N 2^k}} \tau_N\left(a_{N,\,i}\right) = \frac{1}{N}$。

证明。展开 τ_N 可得

$$\sum_{\substack{N \leq i < 2N \\ \exists k:\ i \equiv_N 2^k}} \tau_N(a_{N,i}) = \sum_{\substack{N \leq i < 2N \\ \exists k:\ i \equiv_N 2^k}} \left[\frac{i}{2} \in \mathfrak{P}_{2,N}\right] \cdot \frac{1}{2} \tau_N(a_{N,\frac{i}{2}}).$$

注意到，我们可以改变求和指标集

$$
\begin{aligned}
&\{i \in \mathbb{N} \mid N \leq i < 2N \wedge \exists k:\ i \equiv_N 2^k \wedge \frac{i}{2} \in \mathfrak{P}_{2,N}\} \\
&= \{i \in \mathbb{N} \mid N \leq i < 2N \wedge i \in \mathfrak{P}_{2,N} \wedge \frac{i}{2} \in \mathfrak{P}_{2,N}\} \\
&= \{i \in \mathbb{N} \mid N \leq i < 2N \wedge \frac{i}{2} \in \mathfrak{P}_{2,N}\}
\end{aligned}
$$

与.....一一对应

$$\{i \in \mathbb{N} \mid N \leq i < 2N \wedge \frac{i}{2} \in \mathfrak{P}_{2,N}\} \overset{\substack{i \mapsto \frac{i}{2} \\ 2j \hookleftarrow j}}{\cong} \{j \in \mathbb{N} \mid 0 \leq j < N \wedge j \in \mathfrak{P}_{2,N} \wedge 2j \geq N\}.$$

这又是 π_N 中条目的一个子集，因为 $j < N$，因此可以通过其首次出现的位置来识别，即 $\mathrm{F}_N(j)$。利用 π_N 的最小套索分解，

这些索引介于 0 和 $S_N + L_N - 1$ 之间。因此，

$$
\begin{aligned}
&\sum_{\substack{N\le i<2N\\ \exists k\colon i\equiv_N 2^k}} \left[\frac{i}{2}\in\mathfrak{P}_{2,N}\right]\cdot\frac{1}{2}\tau_N(a_{N,\frac{i}{2}})\\
&= \sum_{\substack{0\le j<N\\ \exists k\colon j\equiv_N 2^k}} [2j\ge N]\cdot\frac{1}{2}\tau_N(a_{N,j})\\
&= \sum_{\substack{0\le j<N\\ \mathrm{F}_N(j)\in\mathbb{N}}} [2j\ge N]\cdot\frac{1}{2}\tau_N(a_{N,j})\\
&= \sum_{0\le k<S_N+L_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2}\tau_N(a_{N,\pi_{N,k}})\\
&= \sum_{0\le k<S_N+L_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2}\left(\frac{1}{2}\right)^{\mathrm{F}_N(\pi_{N,k})}\cdot\frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(\pi_{N,k})}} && \text{(Def. of } \tau_N \text{ with } \pi_{N,k}<N)\\
&= \sum_{0\le k<S_N+L_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2}\left(\frac{1}{2}\right)^{k}\cdot\frac{1}{1-\left(\frac{1}{2}\right)^{\mathrm{L}_N(\pi_{N,k})}}\\
&= \sum_{0\le k<S_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2}\left(\frac{1}{2}\right)^{k}\cdot\frac{1}{1-\left(\frac{1}{2}\right)^{\infty}}\\
&\quad+\sum_{S_N\le k<S_N+L_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2}\left(\frac{1}{2}\right)^{k}\cdot\frac{1}{1-\left(\frac{1}{2}\right)^{L_N}} && \text{(Lemma 1)}\\
&= \sum_{0\le k<S_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2^{k+1}}\\
&\quad+\sum_{S_N\le k<S_N+L_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2^{k+1}}\cdot\sum_{i\in\mathbb{N}}\frac{1}{2^{L_N\cdot i}} && \text{(Geometric series)}\\
&= \sum_{0\le k<S_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2^{k+1}}\\
&\quad+\sum_{S_N\le k<S_N+L_N}\sum_{i\in\mathbb{N}} \left[2\pi_{N,k+L_N\cdot i}\ge N\right]\frac{1}{2^{k+L_N\cdot i+1}} && \text{(Lasso: } \pi_{N,k}=\pi_{N,k+L_N\cdot i} \text{ if } k\ge S_N)\\
&= \sum_{0\le k<S_N} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2^{k+1}}+\sum_{S_N\le j} \left[2\pi_{N,j}\ge N\right]\frac{1}{2^{j+1}} && \text{(Simplify nested sums)}\\
&= \sum_{k\in\mathbb{N}} \left[2\pi_{N,k}\ge N\right]\cdot\frac{1}{2^{k+1}}\\
&= \sum_{k\in\mathbb{N}} \left[2N\cdot\left\{\frac{1}{N}\cdot 2^k\right\}\ge N\right]\cdot\frac{1}{2^{k+1}} && \text{(Proposition 5)}
\end{aligned}
$$

$$= \sum_{k\in\mathbb{N}} \left[\left\{ \frac{1}{N} \cdot 2^k \right\} \geq \frac{1}{2} \right] \cdot \frac{1}{2^{k+1}}$$
$$= \frac{1}{N}. \qquad \text{（二进制表示）}$$

定理 8. 对于 $N \in \mathbb{N}$，$\mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V, F/1] = \frac{1}{N} \cdot \frac{1-C^N}{1-C}$，其中 $\mathfrak{g}_0 = V$。

证明。对于 $N = 1$，我们沿用引理 2 中的占用不变式 $I_1 = 2 \cdot \mathfrak{g}_0 + F V^2 = 2V + F V^2$。因此，$\mathrm{pm}[\![\mathrm{FDR}_1]\!](\mathfrak{g}_0)[V, F/1] \leq I_1 \restriction_{f>0} [V, F/1] = 1 \cdot C^0$。由于 FDR_1 是 PAST，$\left|\mathrm{pm}[\![\mathrm{FDR}_1]\!](\mathfrak{g}_0)[V, F/1]\right| = |1| = 1$。但由于 $\left|1 \cdot C^0\right| = 1$，这意味着 $\mathrm{pm}[\![\mathrm{FDR}_1]\!](\mathfrak{g}_0)[V, F/1] = I_1 \restriction_{f>0} [V, F/1] = 1 \cdot C^0$。

对于 $N > 1$，模板 I_N 与 τ_N 实例化后，根据推论 3可得其为占用不变式；边缘化掉 v 和 f 并使用引理3后，可得到一个过近似：

$$\begin{aligned} \mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V, F/1] &\leq (I_N[\tau_N] \restriction_{f>0})[V, F/1] \\ &= \sum_{\substack{N \leq i < 2N \\ \exists k:\, i \equiv_N 2^k}} \tau_N(a_{N,i}) \cdot \frac{1-C^N}{1-C} \\ &= \frac{1}{N} \cdot \frac{1-C^N}{1-C}, \end{aligned}$$

现在，我们利用FDR_N根据引理2是 PAST，并且有 $\left|\mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V, F/1]\right| = |1| = 1$。由于同样有 $\left|\frac{1}{N} \cdot \frac{1-C^N}{1-C}\right| = \frac{1}{N} \cdot N = 1$，因此该上界是紧的。于是，我们得出$\mathbf{pm}[\![\mathrm{FDR}_N]\!](\mathfrak{g}_0)[V, F/1] = \frac{1}{N} \cdot \frac{1-C^N}{1-C}$。 □

程序 4 倒计数，直到看到 x 成功（负二项分布）。（错误的_递减）/* 初始几何分布 x，参数为 1/2 */ 2 − X/* 概率式倒计数 */当(x > 0){{空操作} [1/2]{x := x − 1}}

程序 5 一个部分终止程序。（非终止）/* 初始均匀分布，含 $x \in \{1, 2\}$ */ 1/2 X + 1/2 X^2 /* 一个关于 x 的永不终止循环=1 */当(x = 1){空操作}

程序 *6* 带计数器的对称随机游走。（随机_游走_计数器） /* 在 *(x=1,c=0)* 上的初始狄拉克分布 */ X/* 随机游走 */ *while*(x > 1){ {x := x − 1} [1/2] {x := x + 1}; c := c + 1}

Prog程序 7 快速骰子滚轮算　　g法，适用于 N=6。(快速_骰子_滚轮)

```
/* 在 (v=1,c=0,f=0) 上的初始狄拉克分布 */// V/
* 快速掷骰滚轮算法 */ while (f = 0){v := 2v;{c:
= 2c} [1/2] {c:= 2c + 1};if (6 ≤ v){ if (c < 6){f:= f + 1}
else{v:= v − 6;c:= c − 6} } }
```

–

程序 8 随机走向某个坐标轴。(条件　　_and)

```
/* 在 (x 上的初始狄拉克分布=0, y=0) */
// 1/* 随机走向某个坐标轴 */当(
x > 0 ∧ y > 0){{x : = x − 1} [1/2] {y : = y − 1}}
```

–

程序 9 过滤几何循环。(三分之一_几何) /* 在 (x 上的初始狄拉克分布=1) */// X/* 每隔 4 项就缺失一项的过滤几何循环 */当(

```
x ≡ 1 模 3){{x : = x + 1} [1/3] {{x : = x + 2} [1/2] {x : = x + 3}}}
```

–　　–

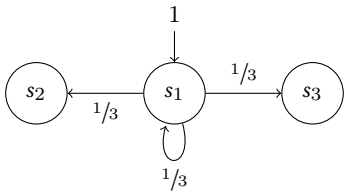

图 2：一个马尔可夫链，其中收缩不变式 [56] 无法证明关于终止概率的任何非平凡界限，适用于 s_2 和 s_3 （参见第 D节）。

D 收缩不变式的详细信息 [56]

我们回顾一下 [56] 中的以下概念：对于一个循环 while (ϕ) $\{C\}$，其初始概率分布为 μ，且常数为 $c \in [0,1)$， c-收缩不变式是指一个测度 ν，满足 $\mu \le \nu$ 且 $\mathrm{pm}[\![C]\!]([\phi]\cdot\nu) \le c\cdot\nu$。这样的 ν可作为证据，证明 $\mathrm{pm}[\![C]\!](\mu) \le \frac{[\neg\phi]\cdot\nu}{1-c}$ [56]。相比之下，回顾定义 10，占用（超）不变式是满足 $\mu+\mathrm{pm}[\![C]\!]([\phi]\cdot\nu') \le \nu'$ 的测度 ν'。它可为上界 $\mathrm{pm}[\![C]\!](\mu) \le [\neg\phi]\cdot\nu'$ 提供证据。这些定义意味着， c-收缩不变式 ν是初始测度 $(1-c)\cdot\nu$ 的一个占用不变式。换言之，收缩不变式构成占用不变式的一个特殊子类。现在我们将展示，后者的能力更强，也就是说，占用不变式能够证明收缩不变式无法达到的界。我们将通过一个具体示例来说明这一点。

考虑图 2 中的马尔可夫链，它对应于一个守卫条件为$s = 1$ 的循环；在每次迭代中，它以相同的概率执行 $s := 2$、 $s := 3$，或什么也不做。为便于阅读，我们将三个相关状态上的测度 μ 写成行向量（$\mu(s_1)$ $\mu(s_2)$ $\mu(s_3)$）。假设初始分布为$\mu = (1, 0, 0)$。由此得到的最终分布为 $(0, \frac{1}{2}, \frac{1}{2})$，其证据为精确的占用测度 $(\frac{3}{2}, \frac{1}{2}, \frac{1}{2})$。对于每个$c \in (0, 1)$，最优（逐点最小的） c-收缩不变式为 $(1, \frac{1}{3c}, \frac{1}{3c})$。如上所述，该不变式给出的上界

$$\left(0, \frac{1}{3c(1-c)}, \frac{1}{3c(1-c)}\right)$$

关于最终分布。这些表达式在 $c = \frac{1}{2}$ 处达到最小值，但这仅意味着平凡的上界 $(0, \frac{4}{3}, \frac{4}{3})$。