arXiv:2301.10102v1 [cs.CC] 24 Jan 2023

面向 AC^0 电路的改进的伪随机生成器

Xin Lyu*

2023年1月25日

摘要

我们给出一种伪随机生成器，适用于深度为d、规模为m AC^0 的电路，其种子长度为 $O(\log^{d-1}(m)\log(m/\varepsilon)\log\log(m))$。我们的伪随机生成器在多个方面改进了先前的工作 [TX13、ST19、Kel21] 。它对 $\frac{1}{\varepsilon}$ 的依赖达到最优，并且距离下界障碍仅差一个“$\log\log(m)$”。对于 $d=2$的情况，其种子长度与已知最佳的合取范式伪随机生成器相匹配 [DETT10、Tal17]。

我们的新结果背后有两个技术要点；它们各自也可能具有独立的研究价值。首先，我们采用基于划分的方法，根据限制引理为 AC^0 构造伪随机生成器。以往的工作 [TX13、ST19、Kel21] 通常基于 Ajtai-Wigderson 框架构造伪随机生成器[AW89]。与之相比，基于划分的方法避免了 Ajtai-Wigderson 框架通常带来的额外“$\log(n)$”因子，使我们得以获得近乎最优的种子长度。基于划分的方法相当通用，我们相信它能帮助为常深度电路以外的类别设计伪随机生成器。

其次，在改进并拓展 [TX13、ST19、Kel21], 的基础上，我们证明了强大的多重切换引理可以完全去随机化 [Hås14]。我们证明，可以使用较短的随机种子抽样一个限制，使得一族析取范式在该限制下以高概率同时得到简化。这回答了 [Kel21]中的一个未决问题。以往的去随机化要么不完全（即，伪随机地选择要施加限制的变量，然后将这些变量固定为真正随机的比特），要么种子长度不是最优的。在我们的应用中，完全去随机化的切换引理至关重要，而我们去随机化方法的随机性效率使我们得以获得近乎最优的种子长度。

1 引言

设 $\mathcal{F}$ 是一个函数类。针对 $\mathcal{F}$ 的伪随机生成器（PRG）是一个算法 $G\colon \{0,1\}^s \to \{0,1\}^n$ ，它将一个较短的随机种子 x 映射为一个较长的字符串 $\mathcal{G}(x)$ ，该字符串对于 $\mathcal{F}$ 中的每个区分器来说都像是随机的。更具体地说，如果对于每个 $f \in \mathcal{F}$，都有如下性质，我们就称 $\mathcal{G}$ ε 能欺骗 $\mathcal{F}$:

$$|\mathbb{E}_{x\sim\mathcal{U}_s}[f(G(x))] - \mathbb{E}_{x\sim\mathcal{U}_n}[f(x)]| \le \varepsilon.$$

在这项工作中，我们研究有界深度布尔电路类（又称AC^0 电路），这是过去几十年来得到广泛研究的电路类。为AC^0电路构造伪随机生成器是一个核心问题，已得到广泛研究 [AW89、Has89、NW94、Bra10、Tal17、HS19、TX13、ST19、Kel21]。

定量地说，令 $AC^0_d[m(n)]$ 表示深度为-d、规模为-m 的电路类。根据概率方法，可以证明存在一个适用于 $AC^0_d[m(n)]$ 的 ε-伪随机生成器，其种子长度为 $O(\log(m/\varepsilon))$。然而，要构造具有相同种子长度的显式伪随机生成器，似乎超出了当前技术所能达到的范围。特别是，任何适用于 $AC^0_d[m(n)]$ 且种子长度为 $\log^{o(d)}(m)$ 的伪随机生成器，都将给出一个非平凡的伪随机生成器

*电气工程与计算机科学系，加州大学伯克利分校。电子邮箱：xinlyu@berkeley.edu

用于 NC^1，并意味着 $\mathrm{NP} \not\subseteq \mathrm{NC}^1$（例如，参见 [GW13, 附录 A]）。此外，[TX13, ST19, Kel21]指出，如果存在一个适用于 $\mathrm{AC}^0_d[m(n)]$ ，、种子长度为 $o(\log^d(m/\varepsilon))$ 的显式伪随机生成器，那么就存在一个显式函数，计算该函数需要规模为 $2^{\omega(n^{1/(d-1)})}$ 的 AC^0_d-电路，从而改进 Håstad 的下界 [Has89] 。这一下界三十多年来一直未能被突破！

已有大量研究 [AW89, Has89, NW94, Bra10, Tal17, HS19, TX13, ST19, Kel21]致力于为 AC^0构造越来越好的伪随机生成器。Ajtai 和 Wigderson 的开创性论文 [AW89] 为 AC^0提出了首个非平凡的伪随机生成器。对于多项式规模的 AC^0 电路，其伪随机生成器的种子长度为$n^{o(1)}$ 。之后，Nisan 将 Håstad 的相关性界 [Has89] 应用于 Nisan-Wigderson“困难性到随机性”框架 [NW94]，为 AC^0 电路构造了伪随机生成器。当 ε-伪随机化 $\mathrm{AC}^0_d[m(n)]$ 电路时，Nisan 的伪随机生成器种子长度为 $\log^{2d+O(1)}(m/\varepsilon)$。Braverman 的突破性成果 [Bra10]证明，任何 $\log^{O(d^2)}(m/\varepsilon)$-独立分布 ε都能对 $\mathrm{AC}^0_d[m(n)]$电路实现伪随机化。结合标准的 k-独立分布构造，这给出了种子长度为 $\log^{O(d^2)}(m/\varepsilon)$ 的伪随机生成器。Tal [Tal17] 以及 Harsha 和 Srinivasan [HS19], 进一步改进了 Braverman 的分析，将种子长度降至 $\log^{3d+O(1)}(m)\log(1/\varepsilon)$。Trevisan 和 Xue [TX13],重新审视了 Ajtai-Wigderson 技术，构造出种子长度为 $\log^{d+O(1)}(m/\varepsilon)$ 的伪随机生成器。近期，针对 Trevisan-Xue 的结果出现了两项互不相容的改进：一项由 Servedio 和 Tan 提出 [ST19]，种子长度为 $\log^{d+O(1)}(m)\cdot\log(1/\varepsilon)$（即对 $\frac{1}{\varepsilon}$ 的依赖达到最优）；另一项由 Kelley 提出 [Kel21],，其种子长度为 $\widetilde{O}(\log^d(m/\varepsilon)\log n)$。

1.1 我们的结果

本文的主要结果是为 $\mathrm{AC}^0_d[m]$ 构造了一种新的伪随机生成器，其种子长度有所改进 $O(\log^{d-1}(m)\cdot\log(m/\varepsilon)\cdot\log\log m)$。

定理 1。以下结论对每个 $d\in\mathbb{N}$ 都成立。对于每个 m，$n\in\mathbb{N}$ 满足 $m\geq n$，以及每个$\varepsilon>0$，都存在一个 ε-伪随机生成器，适用于 $\mathrm{AC}^0_d[m]$ 电路，其种子长度为 $O(\log^{d-1}(m)\cdot\log(m/\varepsilon)\cdot\mathrm{loglog}(m))$。

我们的伪随机生成器构造改进了 Servedio、Tan [ST19] 和 Kelley[Kel21] 提出的两个互不可比的结果。其种子长度对 $\frac{1}{\varepsilon}$ 的依赖是最优的，并且与 Håstad 下界 [Has89, Hås14] 的障碍仅相差一个“log log m”。对于 $d=2$ 的情形，种子长度变为$O(\log(m)\log(m/\varepsilon)\log\log m)$，与目前已知针对合取范式的最佳伪随机生成器 [DETT10, Tal17] 紧密匹配。此外，如果能消去合取范式伪随机生成器中的 log log(m) 项，那么我们的构造将直接推出针对深度为 d 的电路的伪随机生成器，其种子长度为 $O(\log^{d-1}(m)\log(m/\varepsilon))$，与目前针对 AC^0 电路的困难性下界紧密匹配。从“困难性到随机性”的视角来看 [NW94], ，我们的结果已将几乎所有针对 AC^0 的“困难性”转化为 AC^0 的伪随机性。

2 技术

我们的伪随机生成器关键依赖于两个新的技术要素，它们都可能具有独立的研究价值。首先，我们展示了一种基于切换引理构造伪随机生成器的模板[1]。我们的模板与开创性的 Ajtai-Wigderson 框架 [AW89] 有些相似，但种子长度更短。其次，在改进并扩展 [TX13、ST19、Kel21], 的结果基础上，我们证明了一个针对小宽度析取范式的完全去随机化多重切换引理。也就是说，我们给出一种算法，它从短随机种子中采样一个伪随机限制，使得一族析取范式

[1]更一般地，正如 Ajtai-Wigderson 框架一样，我们的模板可以应用于任何关于布尔装置的“限制下简化”引理（例如，德摩根公式的收缩引理）。

以高概率同时在该限制下简化。将我们的模板与新的去随机化方法结合应用，可为 AC^0 电路构造出具有所述种子长度的伪随机生成器。

记号 我们先定义一些有用的记号。设 $f\colon \{0,1\}^n \to \{0,1\}$ 是一个函数。设 $\Lambda \subseteq [n]$ 是一个集合，$x \in \{0,1\}^n$ 是一个字符串。集合-字符串对 (Λ, x) 给出对 f的一个限制。受限函数记作 $f|_{\Lambda[x,\star]}$，定义为 $f|_{\Lambda[x,\star]}(y) = f(\Lambda[x,y])$，其中

$$\Lambda[x,y]_i = \begin{cases} x_i & i \notin \Lambda \\ y_i & i \in \Lambda \end{cases}.$$

直观地说，这意味着输入中除 Λ 部分外的所有部分都固定为 x中对应的位，而 $f|_{\Lambda[x,\star]}$ 现在只是这些 Λ 位的函数。

设 $\mathbf{\Lambda} \subseteq [n]$ 是一个随机变量。如果对每个 $i \in [n]$ 都有 $\Pr[i \in \mathbf{\Lambda}] = p$，我们就称 $\mathbf{\Lambda}$ 的边缘概率为 p。如果每个 $i \in [n]$ 都以概率 p独立地包含在 $\mathbf{\Lambda}$ 中，且 $\mathbf{U}$ 是均匀随机的 n位字符串，我们就称 $\mathbf{\Lambda}[\mathbf{U}\ \star]$ 是一个（真正的） p-随机限制。我们始终用 $\mathbf{U}$ 表示 $\{0,1\}^n$上的均匀分布。

2.1 基于划分的伪随机生成器

我们几乎达到最优的种子长度（相对于下界障碍而言），关键在于一种新的伪随机生成器构造方法，我们称之为基于划分的方法。此前最好的 AC^0 [ST19、Kel21] 伪随机生成器基于迭代限制框架，该框架由 Ajtai 和 Wigderson 在他们的开创性工作 [AW89]中提出。利用划分策略，我们以更简单的证明验证了伪随机生成器的正确性，同时还改进了种子长度。我们认为，基于划分的方法也适用于有界深度电路之外的函数类。

伪随机生成器模板 我们简要介绍一下构造。假设我们要为电路类 $\mathcal{C}_{goal}$设计一个伪随机生成器。我们已知一个针对另一相关电路类 $\mathcal{C}_{simple}$的伪随机分布 $\mathbf{X} \in \{0,1\}^n$ 。我们进一步假设存在一个去随机化的“限制下简化”引理：存在一个整数 $k \geq 1$、一个实数 $p > 0$ 和一个伪随机分布 $\mathbf{Y} \in \{0,1\}^n$，满足以下条件：

- 令 $\mathbf{\Lambda} \subseteq [n]$ 为一个边缘概率为 p 的 k-路独立集合。那么对于每个 $\mathcal{C}_{goal}$ 电路 C，以高概率（关于 $\mathbf{\Lambda}$、$\mathbf{Y}$），受限电路

$$C|_{\mathbf{\Lambda}[\mathbf{Y},\star]}(x) := C(\mathbf{\Lambda}[\mathbf{Y},x])$$

  属于 $\mathcal{C}_{simple}$。

然后，选择 $w = \frac{1}{p}$，并令 $\mathbf{H}\colon [n] \to [w]$ 为一个 k-路独立哈希函数。对于每个 $i \in [w]$，记 $\mathbf{H}_i := \mathbf{H}^{-1}(i)$。考虑以下分布

$$\mathbf{B} = \mathbf{Y} \oplus (\mathbf{X}^{(1)} \wedge \mathbf{H}_1) \oplus (\mathbf{X}^{(2)} \wedge \mathbf{H}_2) \oplus \cdots \oplus (\mathbf{X}^{(w)} \wedge \mathbf{H}_w).$$

这里，$\oplus$、$\wedge$ 分别表示按位异或、合取。$\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(w)}$ 是 $\mathbf{X}$ 的 w 个独立副本。

其思路是，对于每个 $i \in [w]$，如果我们放大并检查集合 $\mathbf{H}^{-1}(i)$，就会发现 $\mathbf{H}^{-1}(i)$ 是一个边缘概率为 $\frac{1}{w}$ 的 k-路独立集合。设 $C \in \mathcal{C}_{goal}$ 是我们希望愚弄的电路。设想我们先对 $\mathbf{B}$ 中除 $\mathbf{X}^{(i)}$ 部分以外的所有部分进行采样。然后计算 $\mathbf{Z}_i := \mathbf{Y} \oplus \bigoplus_{j \neq i} (\mathbf{X}^{(j)} \wedge \mathbf{H}_j)$，并考虑受限函数 $C|_{\mathbf{H}_i[\mathbf{Z}_i,\ \star]}$。我们希望，对于 $\mathbf{H}$ 和 $\mathbf{Z}_i$，以高概率有，

受限函数属于 $\mathcal{C}_{simple}$，它可以被 $\mathbf{X}^{(i)}$愚弄。正如我们将要证明的，除了一个小小的技术问题之外，确实如此[2]。因此，我们得出结论， C 无法区分 $\mathbf{B}$与另一种分布，后者将 $\mathbf{X}^{(i)}$ 部分从 $\mathbf{B}$替换为均匀随机串 $\mathbf{U}^{(i)}$。应用混合论证可以证明， C 无法区分 $\mathbf{B}$ 和

$$\mathbf{B}' = \mathbf{Y} \oplus (\mathbf{U}^{(1)} \wedge \mathbf{H}_1) \oplus (\mathbf{U}^{(2)} \wedge \mathbf{H}_2) \oplus \cdots \oplus (\mathbf{U}^{(w)} \wedge \mathbf{H}_w) \equiv \mathbf{U},$$

这意味着 B 能欺骗 $\mathcal{C}_{goal}$ 电路。

假设用于采样 Y、H 的种子长度足够短，不会成为采样 B 的瓶颈。那么，采样 B 所需的种子长度比采样 X 所需的种子长度大 $w = O(1/p)$ 倍。

在 AC^0 电路中的应用 在下一节中，我们将展示一个用于 AC^0 电路的去随机化的“化简后再施加限制”引理（即去随机化的 Håstad 多重切换引理）。现在先假设该引理成立：它使用参数 $\frac{1}{p} = O(\log m)$，并以高概率将深度-d 电路化简为深度-$(d-1)$电路。将该引理代入我们的模板，可得到一个用于 AC^0_d 电路的伪随机生成器，其种子长度比 AC^0_{d-1}-伪随机生成器长 $O(\log m)$。目前，深度 2 电路（即合取范式/析取范式）的最佳伪随机生成器的种子长度为 $O(\log(m)\log(m/\varepsilon)\log\log m)$（[DETT10、Tal17]）。以此为起点，我们为每个 $d \geq 3$ 构造一个用于深度-d 电路的伪随机生成器，其种子长度为 $O(\log^{d-1}(m)\log(m/\varepsilon)\log(m))$，如所声称的那样。

与 Ajtai-Wigderson 框架的比较 Ajtai 和 Wigderson 首次将限制引理应用于构造伪随机生成器 [AW89]。他们提出了所谓的“迭代限制”框架，并给出了首个非平凡的 AC^0 电路伪随机生成器。与 Nisan-Wigderson 的“困难性到随机性”框架 NW94], 相比，Ajtai-Wigderson 框架能够“打开”下界证明的黑箱，使我们得以为某些精细的电路类构造短伪随机生成器（例如，读一次 AC^0 公式 [GMR+12]）。由于这些原因，近年来 Ajtai-Wigderson 框架日益受到欢迎，其应用范围也远远超出了 AC^0[GMR+12、TX13、HLV18、LV20、FK18、MRT19、DMR+21]。

将我们的方法与他们的框架进行比较会很有启发性。下面我们先简要回顾他们的框架。设 $t = \Theta(\log n/p)$ 为一个参数。令 $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(t)}$ 为 X（针对 $\mathcal{C}_{simple}$ 的伪随机分布）的 t 个独立副本。我们按如下方式抽取一个随机集合列表 $\mathbf{\Lambda}^1, \ldots, \mathbf{\Lambda}^t$。

- 首先，抽样得到 $\mathbf{\Lambda}^1 \subseteq [n]$，它是 $[n]$的一个 k-路独立、边缘概率为 p 的子集。
- 观察到 $\mathbf{\Lambda}^1$ 后，我们以 k-路独立且边缘概率为 p 的方式抽样 $\mathbf{\Lambda}^2 \subseteq [n] \setminus \mathbf{\Lambda}^1$。
- 对于每个 $i \geq 3$。我们先观察 $\mathbf{\Lambda}^1 \cup \cdots \cup \mathbf{\Lambda}^{i-1}$，然后以 k-路独立且边缘概率为 p 的方式抽样 $\mathbf{\Lambda}^i \subseteq [n] \setminus (\bigcup_{j=1}^{i-1} \mathbf{\Lambda}^j)$。

在实际实现中，我们可以先抽样得到 $\mathbf{\Lambda}^t \subseteq [n]$，再从中减去 $\bigcup_{j=1}^{i-1} \mathbf{\Lambda}^j$。给定这些基元，Ajtai-Wigderson 伪随机生成器的输出为

$$\mathbf{D} = (\mathbf{X}^{(1)} \wedge \mathbf{\Lambda}^1) \oplus (\mathbf{X}^{(2)} \wedge \mathbf{\Lambda}^2) \oplus \cdots \oplus (\mathbf{X}^{(t)} \wedge \mathbf{\Lambda}^t).$$

我们观察到，以高概率， $\mathbf{\Lambda}^1 \sqcup \cdots \sqcup \mathbf{\Lambda}^t$ 构成了 $[n]$的一个划分。正确性的证明也采用混合论证。我们观察到 Ajtai-Wigderson 框架与我们的方法之间有两个主要区别。

[2]具体来说，请注意，$\mathbf{H}$ 与 $\mathbf{Z}_i$之间可能存在相关性，因为 $\mathbf{H}$ 决定了每个 $\mathbf{X}^{(j)}$ 中哪一部分会被加到 $\mathbf{Z}_i$上。我们将在第 2.2 节说明如何处理这一问题。另请参阅第 4 节，其中给出了该构造的形式化证明。

1. 作为一项优势，Ajtai-Wigderson 框架无需对 Y 进行采样，而“限制下简化”引理的部分去随机化就足以应用 Ajtai-Wigderson。也就是说，只要求电路 $C|_{\mathbf{\Lambda}[\mathbf{U},\star]}$ 在部分伪随机限制（$\mathbf{\Lambda}$，U）下以高概率简化，其中限制集 $\mathbf{\Lambda}$ 是伪随机的，而字符串 U 是真正随机的。

为了理解这一点为何成立，我们来分析混合论证。例如，考虑比较混合分布

$$\mathbf{D}^{(0)} = (\mathbf{U}^{(1)} \wedge \mathbf{\Lambda}^1) \oplus (\mathbf{U}^{(2)} \wedge \mathbf{\Lambda}^2) \oplus \cdots \oplus (\mathbf{U}^{(t)} \wedge \mathbf{\Lambda}^t),$$

with

$$\mathbf{D}^{(1)} = (\mathbf{X}^{(1)} \wedge \mathbf{\Lambda}^1) \oplus (\mathbf{U}^{(2)} \wedge \mathbf{\Lambda}^2) \oplus \cdots \oplus (\mathbf{U}^{(t)} \wedge \mathbf{\Lambda}^t).$$

为简单起见，假设 $\mathbf{\Lambda}^1 \sqcup \cdots \sqcup \mathbf{\Lambda}^t$ 总是覆盖 $[n]$。那么我们有 $\mathbf{D}^{(0)} = \mathbf{\Lambda}^1[\mathbf{U}、\mathbf{U}^{(1)}]$ 和 $\mathbf{D}^{(1)} = \mathbf{\Lambda}^1[\mathbf{U}、\mathbf{X}^{(1)}]$，因此可以应用部分去随机化限制引理。

2. 然而，当存在完全去随机化限制引理时，采用我们的方法可以得到种子长度更短的伪随机生成器。注意，Ajtai-Wigderson 框架将 $[n]$ 个坐标划分为 $t = \Theta(\log n/p)$ 个块，并用独立的伪随机字符串填充每个块。由于集合 $\mathbf{\Lambda}^i$ 在每个时刻 $i \in [t]$ 都覆盖大约 p 比例的当前未覆盖的坐标，因此将 $t =$ 设为 $\Omega(\log n/p)$ 至关重要，这样 $\bigcup_{j=1}^{t} \mathbf{\Lambda}^j$ 就能以高概率覆盖 $[n]$。另一方面，我们的构造会采样一个 k-路独立哈希函数 $\mathbf{H}: [n] \to [w]$，它自然地诱导出一个划分 $\mathbf{H}^{-1}(1) \sqcup \cdots \sqcup \mathbf{H}^{-1}(w)$。然后，我们只需要 $w = O(1/p)$ 个独立的 X 样本即可完成构造。换句话说，我们通过利用设计中各块之间的对称性，省去了 $\log(n)$ 的额外开销。

总结 基于划分的方法相当通用：对于 Ajtai-Wigderson 框架适用的每种情形，如果我们能够证明一个完全去随机化的“限制下简化”引理，那么就有望利用这一新框架，将种子长度中的 $\log(n)$额外开销削减掉。另两个更具体的例子是稀疏 $\mathbb{F}_2$-多项式 [ST19] 以及小规模德摩根公式 [IMZ12、HHTT21]。不过，在这些应用中，Ajtai-Wigderson 框架带来的 $\log(n)$额外开销并不大。例如，针对 S-稀疏 $\mathbb{F}_2$-多项式的伪随机生成器，其种子长度为 $2^{O(\sqrt{S})}$ [ST19]。在这里改进一个 $\log(n)$ 因子，不如针对 AC^0 电路那样意义重大。

对于（任意顺序的）只读一次分支程序，Forbes 和 Kelley [FK18] 证明了伪随机生成器，其种子长度分别为 $O(\log^3 n)$ 和 $\widetilde{O}(\log^2 n)$，适用于一般 ROBP 和常宽只读一次分支程序。他们的伪随机生成器基于 Ajtai-Wigderson 框架，因此两者的种子长度都额外多出 $\log(n)$ 的开销。若能利用基于划分的方法，为这些模型构造改进的伪随机生成器，将会令人振奋。如果这一思路可行，我们或许能为所有常宽 ROBP 构造更简单的近对数种子长度伪随机生成器，这将是小空间计算去随机化的一项重大进展。目前，我们只知道一种针对宽度为3 的 ROBP 的近对数种子长度伪随机生成器 [MRT19]，其证明似乎难以推广到更大的宽度。

划分（在部分文献中也称为“分桶”）并非伪随机文献中的新技术。已有研究[LV96，GMR13]利用划分为析取范式设计近似计数算法。我们还注意到，Meka 和 Zuckerman [MZ13] 采用了类似策略，为低次数多项式阈值函数（PTF）构造伪随机生成器。他们的伪随机生成器同样利用有界独立哈希函数将 $[n]$ 坐标划分为小块，并用相互独立但伪随机的比特填充每个块。然而，他们的分析与我们的完全不同

，而且他们也不需要添加噪声字符串“Y”来绕过任何限制引理。据我们所知，我们的工作首次使用划分，基于限制引理构造伪随机生成器。

2.2 去随机化多重切换引理

我们结果背后的第二个技术要素，是针对小宽度析取范式的完全去随机化多重切换引理。

哈斯塔德切换引理 切换引理或许是分析低深度布尔电路时最强大、用途最广的工具，其应用范围涵盖证明下界[Has89, Hås14,Vio21], 构造伪随机生成器 [AW89, TX13, ST19, Kel21], 学习 AC^0 函数 [LMN93], 设计用于 AC^0 [的电路分析算法 BIS12, IMP12], ， 以及证明 AC^0 [的傅里叶分析性质 LMN93, Tal17], ， 等等。

标准切换引理最初由 Håstad 证明。该引理指出，对于宽度为 w 的析取范式[3] F，若施加一个 $\frac{1}{20w}$-随机限制（$\mathbf{\Lambda}$, $\mathbf{x}$），那么以 $1-\varepsilon$的概率， $F|_{\mathbf{\Lambda}[\mathbf{x},\star]}$ 会简化为一棵深度为 $O(\log(1/\varepsilon))$ 的决策树。我们也可以证明适用于小规模析取范式的切换引理：假设 F是一个规模为 m 的无界宽度析取范式。那么，施加一个 $\frac{1}{\log(m/\varepsilon)}$-随机限制后， F 会以 $1-\varepsilon$的概率简化为一棵深度为 $O(\log(m/\varepsilon))$ 的决策树。

在第 5 节中，我们展示了标准切换引理的一种去随机化方法。也就是说，我们证明

引理 1 （去随机化切换引理，略作简化）。设 $k, m \geq 1$ 为整数，且$\varepsilon, p > 0$ 为实数。设 $F = \bigvee_{i=1}^{m} C_i$ 是输入 $\{0,1\}^n$ 上的一个大小为 m 、宽度为 k 的析取范式。对于所有 $t \geq 1$，设 $(\mathbf{\Lambda},\mathbf{x})$ 是满足以下条件的任意联合随机变量：

- $\mathbf{\Lambda}$ 是 $[n]$ 的一个$(t+k)$路 p-边缘概率子集。
- 在给定 $\mathbf{\Lambda}$ 的条件下，$\mathbf{x}$ 是一个能以 ε 欺骗大小至多为.....的合取范式的随机串 m.

考虑随机限制 $F|_{\mathbf{\Lambda}[\mathbf{x},\star]}$，我们有：

$$\Pr_{\mathbf{\Lambda},\mathbf{x}}[\mathsf{DT}(F|_{\mathbf{\Lambda}[\mathbf{x},\star]}) > t] \leq (10kp)^t + (4m)^{t+k}\varepsilon.$$

更强的结论和形式化证明见第 5 节。另请注意，我们允许 $\mathbf{\Lambda}$ 与 $\mathbf{x}$之间存在相关性，这对于在我们的伪随机生成器模板中应用该引理至关重要。

根据引理 1，已经可以构造一个适用于 $\mathsf{AC}^0_d[m]$、错误率为 $\frac{1}{m}$ 的伪随机生成器，其种子长度为 $O(\log^d(m)\log\log(m))$。设 $C \in \mathsf{AC}^0_d[m]$ 是一个规模为 m、深度为 d 的电路。为简单起见，我们假设 C 的每个底层门的扇入都不超过 $O(\log(m))$[4]。我们应用引理 1，其中取 $p = \frac{1}{c\log m}$、$t = c\log(m)$，并令 $\varepsilon = 2^{-c\log^2(m)}$，其中 $c > 1$ 是某个较大的常数。我们还将 $\mathbf{x}$ 取为一个能够以误差 ε 伪随机化合取范式的伪随机字符串。那么，对于伪随机限制 $(\Lambda, \mathbf{x})$，以至少 $1-\frac{1}{m^2}$ 的概率， C 的每个深度为 2 的子电路都会化简为深度为 t 的决策树；这意味着我们可以将 $C|_{\mathbf{\Lambda}[\mathbf{x},\star]}$ 表示为一个深度为 $(d-1)$、规模为 $\mathrm{poly}(m)$ 的电路。因此，假设我们能用种子长度为 $O(\log^{d-1}(m)\log\log(m))$ 的伪随机生成器伪随机化深度为 $(d-1)$ 的电路，那么应用基于划分的伪随机生成器，就能用种子长度为 $O(\log^d(m)\log\log(m))$ 的伪随机生成器伪随机化深度为 d 的电路。这里我们省略了采样 Y 和 H 所需的种子长度。事实证明，它们不会成为瓶颈：详见第 4 节。

[3]析取范式 F 的宽度定义为其中任一项所含变量数的最大值。[4]这一假设可通过先应用一个 $\frac{1}{2}$-(伪)随机限制来满足，因为在这种限制下，扇入较大的底层门以高概率都会被消去。

多重切换引理 对于 $\varepsilon < m^{-\omega(1)}$ 的情形，使用引理 1 可能会导致种子长度更长。事实上，若要以至少 $1-\varepsilon$ 的概率化简电路，必须将引理 1 中的“t”参数设为 $\Theta(\log(m/\varepsilon))$。随后应用一次引理 1，可将 C 化简为深度为 $(d-1)$ 的电路，其底层扇入不超过 $t = \Theta(\log(m/\varepsilon))$。若要进一步应用该引理，就必须将 p 设为[1] $\Omega(\log(m/\varepsilon))$，才能使引理 1 中的概率下界非平凡。因此，深度为 d 的伪随机生成器的种子长度将比深度为 $(d-1)$ 的伪随机生成器长 $\frac{1}{p} = \Omega(\log(m/\varepsilon))$，使总种子长度达到 $\Omega(\log^d(m/\varepsilon)\log\log(m))$。

如果我们坚持使用[1] $\log(m)$ 随机限制，并希望获得相同的 $(1-\varepsilon)$ 概率保证，就可以使用多重切换引理。我们先给出其陈述。

引理2。 设 $t, w, k, n, m \geq 1$ 为整数。设 $p, \delta > 0$ 为实数。设 $\mathcal{F} = \{F_1, \ldots, F_m\}$ 是一个列表，其中包含定义在输入 $\{0,1\}^n$ 上的大小为 m 、宽度为 k 的析取范式。设 $(\mathbf{\Lambda}, \mathbf{x})$ 为满足以下条件的联合随机变量：

- $\mathbf{\Lambda}$ 是 $[n]$ 的一个$(t+k)$路 p-边缘概率子集，
- 在给定 $\mathbf{\Lambda}$ 的条件下，$\mathbf{x}$ 是一个 n 位随机串，能够 δ-欺骗大小为(m^2) 的合取范式。

那么，以至少 $1 - \left(4m^{t/w}(24pk)^t + (24m)^{t+k}\cdot\delta\right)$ 的概率，在 $(\mathbf{\Lambda}, \mathbf{x})$ 上存在一个适用于 $\mathcal{F}|_{\mathbf{\Lambda}[\mathbf{x},\star]}$ 的公共 w-部分深度为 t 的决策树[5] 。也就是说，我们可以构造一个决策树列表 $T_1, \ldots, T_m$，分别计算 $F_1|_{\mathbf{\Lambda}[\mathbf{x},\star]}, \ldots, F_m|_{\mathbf{\Lambda}[\mathbf{x},\star]}$。每个 T_i 的深度至多为$(t+w)$，并且所有 T_i 在前 t 次查询中共享相同的查询策略。

设 $C \in \mathsf{AC}^0_d[m]$ 是一个底层扇入至多为 $k = \log(m)$ 的电路。设 $\mathcal{F}$ 表示 2 层子电路的族，且它们都是 C 的子电路。在 $\mathcal{F}$ 上应用引理 2，参数为 $\frac{1}{p} = O(\log(O \quad m \quad))$、$t = O(\log(m/\varepsilon))(\log())$ 和 $\delta = \frac{\varepsilon}{m^{O(t)}}$。我们知道，$\mathcal{F}|_{\mathbf{\Lambda}[\mathbf{x},\star]}$ 无法简化的概率至多为

$$\left(4m^{t/w}(24pk)^t + (24m)^{t+k}\cdot\delta\right) \leq \varepsilon.$$

当 $\mathcal{F}$ 确实能够简化时，我们可以通过混合模型计算 $C|_{\mathbf{\Lambda}[\mathbf{x},\star]}$：一个深度为 t 的决策树，其叶节点上带有 $\mathsf{AC}^0_{d-1}[m\cdot 2^w]$-电路。决策树部分根据 $\mathcal{F}$ 的公共部分决策树执行 t 次自适应查询。之后，$\mathcal{F}$ 中的函数可以表示为深度为 w 的决策树，这意味着 C 可以由一个 $\mathsf{AC}^0_{d-1}[m\cdot 2^w]$ 电路计算。

如果我们能够以错误 $\frac{\varepsilon}{2^t}$ 欺骗叶节点上的每个深度为 $(d-1)$ 的电路，那么就能以错误 ε 欺骗这个混合模型。假设存在一个用于深度为 $(d-1)$ 电路的 $O(\log^{d-2}(m)\log(m/\varepsilon)\log\log(m))$ 伪随机生成器，要欺骗这个混合模型所需的种子长度为

$$O\left(\log^{d-2}(m\cdot 2^w)\log(m2^t/\varepsilon)\log\log(m)\right) = O(\log^{d-2}(m)\log(m/\varepsilon)\log\log(m)).$$

由此，我们可以构造一个适用于深度为 d 的电路的伪随机生成器，其种子长度为 $O(\log^{d-1}(m)\cdot\log(m/\varepsilon)\cdot\log\log(m))$，如所声称的那样。

证明直觉 当限制串是真正随机时，Kelley 技术 [Kel21] 清楚地揭示了随机限制 $\mathbf{\Lambda}[\mathrm{U}\ \star]$ 何时会变得糟糕。固定一个 $\mathsf{AC}^0_d[m]$ 电路 C。给定一个限制 $\mathbf{\Lambda}[\mathrm{U}\ \star]$，我们想知道 C 是否会在 $\mathbf{\Lambda}[\mathrm{U}\ \star]$下简化。粗略地说，Kelley 表明，我们可以先观察限制串 U，然后构造一个包含 $t = \log(m)^{\log(m)}$ 个集合 $S_1, \ldots, S_t$ 的列表，每个集合的大小为 $c\log(m)$，其中 $c \geq 1$ 是一个较大的常数。接着，我们考察集合 $\mathbf{\Lambda}$：C

[5]参见第 3.2 节，了解“部分决策树”的正式定义。

只有当 $\mathbf{\Lambda}$ 至少包含列表中的一个集合 S_i 时，才不会在 $\mathbf{\Lambda}[\mathbf{U}\ \star]$ 下简化。由于 $\mathbf{\Lambda}$ 是$O(\log(m))$-wise $\frac{1}{\log(m)}$边缘概率，因此根据简单的并集界，这种情况发生的概率至多为 $2^{-c\log(m)}$。

我们应用 Kelley 技术对多重切换引理进行去随机化 [Hås14]。多重切换引理的证明涉及许多技巧和技术细节。这里我们试着给出一些（过度简化的）直观解释。从非常宏观的层面来看，多重切换引理是通过结合标准切换引理与并集界来证明的。如果 $\mathcal{F}|_\rho$ 不满足一个 w-部分深度-t 决策树，则存在一个由至多 $\frac{t}{w}$ 个公式组成的子集 $\mathcal{F}_\rho$，使得它们的决策树复杂度之和超过 t。这是因为每个决策树复杂度不超过 w 的公式都可以“免费”处理。因此，每个坏公式至少会为总和贡献 w。这样的子集至多有 $m^{t/w} = 2^{O(t)}$ 个。对于每个这样的子集，我们将其决策树复杂度之和超过 t 的概率上界为 $O(kp)^t$。这一步在思路上与标准切换引理的情形相当类似，而且 Kelley 技术也适用。

我们分析中的最后一部分是完全去随机化。通过 Kelley 技术，我们知道部分伪随机限制 $\mathbf{\Lambda}[\mathbf{U}\ \star]$ 和真正随机的限制一样好。我们进一步使用 Trevisan、Xue、Servedio 和 Tan 的技术对随机串进行去随机化[TX13，ST19]。具体来说，他们构造了有界深度电路（称为“测试器”），以限制作为输入，并判断该限制对于简化目标电路而言是好是坏。现在，考虑从一个能够欺骗有界深度电路的分布中采样一个串 $\mathbf{x}$。给定测试器，我们可以证明 $\mathbf{\Lambda}[\mathbf{x}\ \star]$ 和 $\mathbf{\Lambda}[\mathbf{U}\ \star]$一样好。由于欺骗更高深度的电路需要更长的随机比特，为了控制伪随机生成器最终的种子长度，我们需要测试器能够由 AC^0_2实现。对于标准切换引理，Trevisan-Xue 测试器 [TX13]的确具有深度 2。对于多重切换引理，情况稍微棘手一些：Servedio-Tan 测试器[ST19] 被设计为深度为3 的电路，并能忠实地完成测试。我们（隐式地）实现了他们测试器的一个“上侧近似器”。我们的单侧测试器可以表示为合取范式，在对选取坏限制的概率进行上界估计时同样有用。

关于标准版本和多重切换版本，请参见第5 节和第6节中介绍的去随机化切换引理。我们不再围绕决策树展开讨论，也不再追踪查询路径，而是力求基于“规范查询算法”给出证明。我们的证明更具操作性，而且在我们看来更容易理解。

与先前工作的比较 在为 AC^0 电路设计伪随机生成器这一任务上，在我们的结果之前，前沿领域有两项互不支配的结果：一项来自 Servedio 和 Tan [ST19]，另一项来自 Kelley [Kel21]。两者都建立在切换引理的去随机化结果之上。

Servedio-Tan伪随机生成器基于 Håstad 多重切换引理的去随机化 [Hås14]，其种子长度为 $\log^{d+O(1)}(m)\log(1/\varepsilon)$，当 ε-伪随机化 $\mathsf{AC}^0_d[m(n)]$时成立。由于使用了多重切换引理，他们的伪随机生成器对错误参数具有最优依赖性 ε。然而，他们对切换引理的去随机化在种子长度方面较弱。这两个因素共同决定了其伪随机生成器的最终种子长度。

另一方面，Kelley 的伪随机生成器基于对标准切换引理更强的去随机化 [Has89]。其种子长度为 $\widetilde{O}(\log^d(m/\varepsilon)\log(n))$。$\log(m)$ 上的指数与下界障碍相符，这得益于他们更强的去随机化，使得人们能够用短得多的种子采样一个限制。然而，Kelley 只证明了部分去随机化，这并不适用于我们的构造。此外，由于标准切换引理的分析略显粗糙，对 $\frac{1}{\varepsilon}$ 的依赖性也较差。在[Kel21] 中，这被作为一个未决问题提出：能否对多重切换引理实现同等程度的高等同性去随机化，并优化对 $\frac{1}{\varepsilon}$ 的依赖性。

我们通过证明一个完全去随机化多重切换引理，对这两项工作均有所改进，从而对这个问题给出了肯定的回答。结合基于划分的伪随机生成器框架，我们的引理为 AC^0 给出了一个种子长度几乎紧的伪随机生成器。我们希望，对切换引理的去随机化能够在其他情境中找到应用。

最后，我们指出，“决策树后接电路”这一类型的混合模型也出现在许多以往的工作中。其应用包括证明相关性界和傅里叶谱界 [Hås14、Tal17], 构造伪随机生成器 [ST19、HHTT21], 设计电路分析算法[CSS18], 等。

3 预备知识

本节中，我们将介绍必要的记号，并回顾伪随机性和复杂性理论文献中的一些知名且有用的事实。

3.1 限制、部分赋值和字符串

我们交替使用“限制”和“部分赋值”这两个术语。二者均指形如 $\rho \in \{0,1,\star\}$ 的字符串。此处，如果 $\rho_i = 0/1$，则表示 ρ 的第 i 位被固定为该值。否则， ρ 的第 i 位未固定。

对于两个部分赋值 ρ、$\sigma \in \{0,1,\star\}$，定义它们的合成 $\rho \circ \sigma$ 如下：

$$(\rho \circ \sigma)_i = \begin{cases} \rho_i & \rho_i \neq \star \\ \sigma_i & o.w. \end{cases}.$$

注意，左侧部分赋值的优先级始终高于右侧部分赋值 .

设 $\Lambda \subseteq [n]$ 是一个集合。对于两个部分赋值 ρ、$\sigma \in \{0,1,\star\}^n$，令 $\Lambda[\rho,\sigma]$ 为如下定义的赋值：

$$\Lambda[\rho,\sigma]_i = \begin{cases} \rho_i, & i \notin \Lambda \\ \sigma_i, & i \in \Lambda \end{cases}.$$

设 $f\colon \{0,1\}^n \to \{0,1\}$ 是一个函数， $\rho \in \{0,1,\star\}^n$ 是一个部分赋值。我们用 $f|_\rho\colon \{0,1\}^n \to \{0,1\}$ 表示 f 在 ρ 上的限制。也就是说， $f|_\rho(y) := f(\rho \circ y)$。

3.2 计算模型

$\mathsf{AC}^0_d[s(n)]$ 表示大小至多为 $s(n)$、深度至多为 d 的布尔电路族。这类电路可以包含合取门、或门和非门。此处，非门不计入深度，合取门和或门的扇入可以无界。我们用电路中的连线总数（包括输入连线）来度量电路的大小。对于 $d = 2$ 的情形，我们也使用 DNF 和 CNF 分别指析取 $\circ$ 合取和合取 $\circ$ 析取电路。DNF 或 CNF 的宽度定义为其底层扇入的最大值。我们也用 k-DNF（相应地，k-CNF）表示宽度至多为 k 的 DNF（相应地，CNF）。

决策树 T 是一棵二叉树。 T 的每个内部节点都标记有一个索引 $i \in [n]$，并且恰有两个子节点，分别标记为 0 和 1。 T 的每个叶节点都标记有一个布尔值 $b \in \{0,1\}$。决策树 T 按以下方式计算函数：对于输入$x \in \{0,1\}^n$，我们从 T的根节点开始。每一步，我们观察当前节点的索引 i，查询x_i，并根据收到的比特 x_i 移动到左/右子节点。一旦到达标记为 b的叶节点，我们就输出 $T(x) = b$。决策树的深度是从根节点到任意叶节点的最长路径长度。

我们还考虑一种针对函数列表的特殊决策树模型。请参见下方的定义。

定义 1. 设 $t, w, n, m \geq 1$ 为整数。设 $\mathcal{F} = \{F_1, \ldots, F_m\}$ 为一列从 $\{0,1\}^n$ 映射到 $\{0,1\}$ 的函数。对于 $\mathcal{F}$，一个 w-部分深度-t 决策树是满足以下条件的深度-t 决策树 T。对于每个 $F_i \in \mathcal{F}$ 以及 T 的每个叶节点 ℓ，设 $\alpha \in \{0,1,\star\}^n$ 为与 ℓ 对应的部分赋值（也就是说，如果 T 在到达 ℓ 之前没有查询 x_i，则 α_i 等于 $\star$；否则，α_i 等于使 T 朝 ℓ 移动的值）。那么 $\mathrm{DT}(F_i|_\alpha) \leq w$。

3.3 伪随机性

回顾有界独立性的定义。

定义 2. 设 n, m 为两个整数。设 $\mathcal{H}$ 为一个定义在哈希函数上的分布，这些哈希函数将 $\{0,1\}^n$ 映射到 $\{0,1\}^m$。如果对于任意 k 个输入-输出对 $(x_1, y_1), \ldots, (x_k, y_k) \in \{0,1\}^n \times \{0,1\}^m$（其中 $x_1, \ldots, x_t$ 互不相同），均有

$$\Pr_{h \sim \mathcal{H}}[\forall i \in [k], h(x_i) = y_i] = 2^{-km}.$$

我们有以下有界独立性哈希函数的标准构造（例如参见 [Vad12，第 3.5 章。5]）。

引理 3. 对于每个 n，m，$k \geq 1$，存在一个显式的 k 重独立哈希函数 $\mathcal{H}$，它将 $\{0,1\}^n$ 映射到 $\{0,1\}^m$ 中。可以使用 $O(k(n+m))$ 个随机比特从 $\mathcal{H}$ 中采样一个函数。

我们还考虑一种较弱的伪随机性概念，称为“k-重 p-有界性”，最早由 [Kel21] 定义并研究。

定义 3. 设 $\mathbf{\Lambda}$ 是 $[n]$ 的一个随机子集。若对于大小至多为 k 的任意集合 $B \subseteq [n]$，均有 $\Pr_{\mathbf{\Lambda}}[B \subseteq \mathbf{\Lambda}] \leq p^{|B|}$，则称 Λ 是 k 重 p 有界的。

直观地说，如果我们通过以至多为 p 的概率独立地将 $\mathbf{\Lambda}$ 中的每个 i 纳入 $\mathbf{\Lambda}$ 进行采样，那么 $\mathbf{\Lambda}$ 就是 n 重 p 有界的。

4 常深度电路的改进伪随机生成器

本节旨在证明主要定理，现将其重新陈述如下。

定理 1 的回顾： 对每个 $d \geq 2$，以下结论成立。对每个 m，$n \in \mathbb{N}$ 满足 $m \geq n$ 且对每个 $\varepsilon > 0$，存在一个 ε-伪随机生成器，适用于 $\mathsf{AC}^0_d[m]$ 电路，其种子长度为 $O(\log^{d-1}(m) \cdot \log(m/\varepsilon) \cdot \mathrm{loglog}(m))$。

我们从以下事实开始，它在我们的构造中至关重要。

定理 2 ([DETT10，Tal17])。对每个 m，$n \in \mathbb{N}$ 满足 $m \geq n$ 且对每个 $\varepsilon > 0$，存在一个ε-伪随机生成器，适用于 $\mathsf{AC}^0_2[m]$ 电路（即合取范式/析取范式公式），其种子长度为 $O(\log(m) \cdot \log(m/\varepsilon) \cdot \log\log(m))$。

我们将在下文给出去随机化多重切换引理的正式陈述。这是引言中引理 2 的完整版本。

引理 4. 设 $t, w, k, n, m \geq 1$ 为整数。设 $p, \varepsilon > 0$ 为实数。设 $\mathcal{F} = \{F_1, \ldots, F_m\}$ 为一个由作用于输入 $\{0,1\}^n$上的大小为-m k-的析取范式构成的列表。设 $(\mathbf{\Lambda}, \mathbf{x})$ 为满足以下条件的联合随机变量：

- $\mathbf{\Lambda}$ 是 $[n]$ 的一个$(t+k)$元 p-有界子集，
- 在给定 $\mathbf{\Lambda}$ 的条件下，$\mathbf{x}$ 是一个 n 位随机串，可以 ε 混淆大小为(m^2) 的合取范式。

于是我们有

$$\Pr_{x\sim\mathbf{x}}[f_{\mathbf{\Lambda}[x,\star]} \text{ does not have } w\text{-partial depth-}t\ DT] \leq 4m^{t/w}(24pk)^t + (24m)^{t+k}\,.$$

引理 4的证明推迟到第 6节。假设引理 4成立，我们证明定理 1。我们证明定理 1的一个稍强形式，陈述如下。

定理 3. 对于每个 $d \geq 2$，以下结论成立。对于每个 $m, n \in \mathbb{N}$，使得 $m \geq n$, $k \in \mathbb{N}$，以及每个 $\varepsilon > 0$，都存在一个错误为 ε、$O((\log^2(m) + k\cdot\log^{d-2}(m))\cdot\log(m/\varepsilon)\cdot\mathrm{loglog}(m))$ 种子长度的伪随机生成器，适用于底层扇入不超过 k 的 $\mathsf{AC}^0_d[m]$ 电路。

定理 3 表明，当 AC^0_d 电路的底层扇入小于 $o(\log(m))$ 时，我们可以通过构造获得更短的种子长度。注意，底层扇入无界的深度为 d AC^0 电路总可以解释为深度为 $(d+1)$ AC^0 电路，且其底层扇入为 1。于是，定理 1 可由定理 3 轻易推出。

证明。我们对深度 d进行归纳。对于 $d = 2$ 的情形，由定理 2可得。假设对于深度 $d-1 \geq 2$的情形结论成立，我们证明 d的情形。令 $w = 40k$和 $t = 80\log(m/\varepsilon)$。我们准备以下伪随机原语。

- 首先，令 $\mathrm{H} : [n] \to [w]$ 为一个 $2t$-wise 独立哈希函数，可使用 $O(\log(n)\log(m/\varepsilon))$ 个比特进行采样。下文中，我们用 H_i 表示 $\mathrm{H}^{-1}(i)$。注意，H_i 也可以等价地表示为一个 n 位字符串。也就是说，$(\mathrm{H}_i)_j = 1$ 当且仅当 $j \in \mathrm{H}_i$。
- 其次，令 $\varepsilon' = \varepsilon/(w\cdot 2^{t+1})$。采样 $\mathrm{X}1, \ldots, \mathrm{X}_w \in \{0,1\}^n$，其中每个都是一个独立字符串，能够对大小为 $2m^2$、底层宽度为 $\log m$ 的 AC^0_{d-1}-电路进行 ε'-伪随机化。当 $d-1 \geq 3$ 时，根据归纳假设，X_i 可使用 $O(\log^{d-2}(m)\cdot\log(m/\varepsilon)\cdot\log\log(m))$ 个比特进行采样。对于 $d-1=2$ 的情形，根据定理 2，X_i 可使用 $O(\log(m)\log(m/\varepsilon)\log\log(m))$ 个比特进行采样。
- 最后，令 $\mathbf{Y} \in \{0,1\}^n$ 为一个能 $\varepsilon/(24m)^{2t}$-欺骗规模为m 的合取范式的随机串，可使用$O(\log^2 m\cdot\log(m/\varepsilon)\cdot\log\log m)$ 位进行采样（根据定理 2）。

我们的生成器的种子是用于对上述所有基元进行采样的种子的串联。我们的生成器的输出计算如下：

$$\mathbf{Y} \oplus (\mathbf{X}_1 \wedge \mathbf{H}_1) \oplus (\mathbf{X}_2 \wedge \mathbf{H}_2) \oplus \cdots \oplus (\mathbf{X}_w \wedge \mathbf{H}_w). \tag{1}$$

这里，$\wedge$an d $\oplus$ 分别表示逐位与和异或。种子长度上界为

$$\begin{aligned} &O\left(\log(n)\log(m/\varepsilon) + w\cdot\log^{d-2}(m)\log(m\cdot 2^t/\varepsilon)\log\log(m) + \log^2 m\log(m/\varepsilon)\log\log m\right) \\ \leq\ & O((\log^2(m) + k\cdot\log^{d-2}(m))\cdot\log(m/\varepsilon)\cdot\log\log(m)). \end{aligned}$$

我们通过混合论证证明正确性。固定 C 为我们希望欺骗的一个 AC^0_d-电路。令 U_1、$\ldots$、U_w 表示来自 $\{0,1\}^n$的 w个相互独立的均匀随机字符串。对于每个$i \in \{0,1,\ldots,w\}$，我们将第 i个混合分布定义为

$$\mathcal{D}_i := \mathbf{Y} \oplus (\mathbf{U}_1 \wedge \mathbf{H}_1) \oplus \cdots \oplus (\mathbf{U}_i \wedge \mathbf{H}_i) \oplus (\mathbf{X}_{i+1} \wedge \mathbf{H}_{i+1}) \oplus \cdots \oplus (\mathbf{X}_w \wedge \mathbf{H}_w). \tag{2}$$

我们注意到， $\mathcal{D}_0$ 是我们的伪随机生成器的输出分布，而 $\mathcal{D}_w$ 是来自 $\{0,1\}^n$的一个均匀随机字符串。因此，只需证明

$$|\mathbb{E}_{x\sim\mathcal{D}_0}[C(x)] - \mathbb{E}_{x\sim\mathcal{D}_w}[C(x)]| \le \varepsilon. \tag{3}$$

要证明（3），只需证明对每个 $i \in \{1,\dots,w\}$ 都有

$$|\mathbb{E}_{x\sim\mathcal{D}_{i-1}}[C(x)] - \mathbb{E}_{x\sim\mathcal{D}_i}[C(x)]| \le \varepsilon/w. \tag{4}$$

下面我们证明（4）。注意到 $\mathbf{H}_i$ 是 $2t$-路 $\frac{1}{w}$-有界的。对 $\mathbf{H}$ 的一个具体取值进行条件化后，有 $\mathbf{Z}_i := \mathbf{Y} \oplus \sum_{j<i}(\mathbf{U}_j \wedge \mathbf{H}j) \oplus \sum_{j>i}(\mathbf{X}_j \wedge \mathbf{H}j)$ 是针对大小为（m^2）的合取范式的一个 $\varepsilon/(24m)^{2t}$-伪随机字符串，因为 $\mathbf{Y}$ 是。令 $\mathcal{F}$ 为 C的所有倒数第二层子电路的族。用 $\mathcal{E}$ 表示事件

$$\text{“}\mathcal{F}|_{\mathbf{H}_i[\mathbf{Z}_i,\star]} \text{ does not have } \log(m)\text{-partial depth-}t \text{ DT.”}$$

由引理 4 可知

$$\Pr_{\mathbf{H},\mathbf{Y},\mathbf{U}_1,\dots,\mathbf{U}_{i-1},\mathbf{X}_{i+1},\dots,\mathbf{X}_w}[\mathcal{E}] \le 4m^{t/\log(m)}\left(24\frac{k}{w}\right)^t + \frac{\varepsilon\cdot(24m)^{t+\log(m)}}{(24m)^{2t}} \le \frac{\varepsilon}{2w}.$$

对 $\neg\mathcal{E}$、$\mathbf{H}$、$\mathbf{Y}$、$\mathbf{U}_1$、…、$\mathbf{U}_{i-1}$、$\mathbf{X}_{i+1}$、…、$\mathbf{X}_w$进行条件化，并按上文定义计算 $\mathbf{Z}_i$后，可以将 $C|_{\mathbf{H}_i[\mathbf{Z}_i,\ \star]}$ 写成一个深度为-t 的决策树 T，其中 T的每个叶节点都标记着一个大小为 m^2、底层扇入为 $\log(m)$ 的 AC^0_{d-1}-电路。令 $\{\ell_1,\dots,\ell_{2^t}\}$ 依次表示该决策树的各个叶节点。每个ℓ_j 都对应一个大小为-m^2、深度为（$d-1$）的电路，为简便起见，也记作 $\ell_j\colon\ \{0,1\}^n \to \{0,1\}$。于是可以将 $C|_{(H_i)[Z_i,\star]}$ 写成

$$C|_{\mathbf{H}_i[Z_i,\star]}(y) = \sum_{j=1}^{2^t} \ell_j(y)\cdot \mathbb{I}\{T(y) \text{ reaches leaf } \ell_j\}.$$

暂且固定一个索引 $j\in[2^t]$。注意， $\ell_j(y)\cdot\mathbb{I}\{$在 $y\}$ 上到达 ℓ_j，其本身是一个深度为 $d-1$、规模至多为 $2m^2$ 的电路。由 X_i 的构造可知

$$\Big|\mathbb{E}_{\mathbf{X}_i}[\ell_j(\mathbf{X}_i+Z_i)\cdot\mathbb{I}\{T \text{ reaches } \ell_j \text{ on } \mathbf{X}_i+Z_i\}] - \mathbb{E}_{\mathbf{U}_i}[\ell_j(\mathbf{U}_i+Z_i)\cdot\mathbb{I}\{T \text{ reaches } \ell_j \text{ on } \mathbf{U}_i+Z_i\}]\Big| \le \frac{\varepsilon}{2^{t+1}w}.$$

对所有叶子 j 求和，可得

$$\Big|\mathbb{E}_{\mathbf{X}_i}[C|_{\mathbf{H}_i[Z_i,\star]}(\mathbf{X}_i+Z_i)] - \mathbb{E}_{\mathbf{U}_i}[C|_{\mathbf{H}_i[Z_i,\star]}(\mathbf{U}_i+Z_i)]]\Big| \le \frac{\varepsilon}{2w}.$$

最后，可得

$$|\mathbb{E}_{x\sim\mathcal{D}_{i-1}}[C(x)] - \mathbb{E}_{x\sim\mathcal{D}_i}[C(x)]| \le \Pr[\neg\mathcal{E}]\cdot\frac{\varepsilon}{2w} + \Pr[\mathcal{E}] \le \frac{\varepsilon}{w},$$

证明（4）。 □

由定理 3，通过调整参数来证明定理 1 。

定理 1 的证明。 对于每个 $d\ge 3$ 以及每个 $\mathsf{AC}^0_d[m]$ 电路（底层扇入无界），我们可以将其视为深度为 $(d+1)$ 且底层扇入为 1 的电路。在这种情况下应用定理 3 可得到一个种子长度为 $O(\log^{d-1}(m)\log(m/\varepsilon)\log\log(m))$ 的伪随机生成器，如所需。对于 $d=2$ 的情况，我们直接使用定理 2 。证明完毕。 □

最后补充一点，假设我们能够为合取范式构造种子长度为 $O(\log(m)\log(m/\varepsilon))$ 的伪随机生成器（也就是说，如果我们能在定理 2 中将 $\log\log m$ 因子削去）。那么我们的构造意味着可以为 AC^0_3 构造种子长度为 $O(\log^2(m)\log(m/\varepsilon))$ 的伪随机生成器，进而意味着可以为 AC^0_d 构造种子长度为$O(\log^{d-1}(m)\log(m/\varepsilon))$ 的伪随机生成器，与下界障碍 [Has89、Hås14] 相匹配。

5 完全去随机化切换引理

在给出引理 4 的证明之前，本节先陈述并证明经典切换引理的一个较简单版本。下面给出的陈述是引言中引理 1 的完整版本。引理 1 显然可由引理 5 推出。

引理 5。 设 $k, m \geq 1$ 为整数，$\varepsilon, p > 0$ 为实数。设 $F = \bigvee_{i=1}^{m} C_i$ 是关于输入 $\{0,1\}^n$ 的一个大小为 m k 的析取范式。对于所有 $t \geq 1$，令 $(\mathbf{\Lambda}, \mathbf{x})$ 为满足以下条件的联合随机变量：

- $\mathbf{\Lambda}$ 是 $[n]$ 的一个 $(t+k)$-路 p-有界子集。
- 在给定 $\mathbf{\Lambda}$ 的条件下，$\mathbf{x}$ 是一个能欺骗大小至多为 m 的合取范式的随机串，误差为 ε。

考虑随机限制 $F|_{\mathbf{\Lambda}[\mathbf{x},\star]}$，我们有：

$$\Pr_{\mathbf{\Lambda},\mathbf{x}}[\mathsf{DT}(F|_{\mathbf{\Lambda}[\mathbf{x},\star]}) > t] \leq (10kp)^t + (4m)^{t+k} \cdot \varepsilon.$$

理解引理 5 的证明，是阅读引理 4证明的必要条件。另一方面，一旦引理 5 得以证明，我们就可以用一种相当类似的策略证明引理 4 。本节剩余部分将用于证明引理 5。

我们首先将在第 [Has89、TX13、ST19、Tal17]5.1 节介绍“规范决策树”这一重要概念。我们在第 5.2 节证明 $\mathbf{x}$ 是真正随机的情形（而只有 $\mathbf{\Lambda}$ 是伪随机的），随后在第 5.3 节说明如何证明 $\mathbf{x}$ 为伪随机的情形。

在整个本节中，我们始终用 $F = \bigvee_{i=1}^{m} C_i$ 表示正在分析的 k-析取范式。对于每个 $i \in [m]$，令 $V_i \subseteq [n]$ 表示项 C_i中涉及的变量。

5.1 规范决策树

对于析取范式 $F = \bigvee_{i=1}^{m} C_i$，记 T_F为 规范决策树，其对应的 F，构造过程如算法 1所示。我们用 规范决策树(F) 表示 F对应的规范决策树的深度。由于规范决策树是 F的一种特定决策树，其深度必不小于 决策树深度(F)。为了证明引理 5，我们将分析规范决策树，并证明以高概率

在随机限制 ρ 下，我们有 $\mathsf{CDT}(F|_\rho) < t$。

Alg算法 1： 规范决策树

输入： 一个析取范式 $F = \bigvee_{i=1}^{m} C_i$，以及对字符串 $\alpha \in \{0,1\}^n$ 的黑盒访问。

初始化： $j^* \leftarrow 0$.$x \leftarrow (\star)^n$.**当** $j^* < m$ **时**查找第一个满足 $C_j(x) \neq 0$ 的 $j > j^*$。

如果不存在这样的 j，则退出循环。$B_j \leftarrow C_j$ 中未知变量的集合。查询 α_{B_j}。

设置 $x_{B_j} \leftarrow \alpha_{B_j}$。**如果** $C_j(x) = 1$ **则返回** 1.$j^* \leftarrow j$.**返回** 0

5.2 证明：当 x 真正随机时

定义见证 令 $\rho = \Lambda[x, \star]$ 为一个坏限制，在该限制下，$\mathsf{CDT}(F|_\rho) \geq t$。考虑模拟规范决策树 $T_{F|_\rho}$。我们知道，对于某些输入，$\alpha \in \{0,1\}^n$ 在进行 $(t-1)$ 次查询后，$T_{F|_\rho}$ 未能输出决策。我们选择这样一个 α，并模拟 $T_{F|_\rho}$，直到它至少进行 t 次查询。$T_{F|_\rho}$ 在 α 上的"运行记录"自然就是以下事实的一个见证：$\mathsf{CDT}(F_\rho) \geq t$。我们在以下定义中将这一想法形式化。

定义 4。 令 $F = \bigvee_{i=1}^{m} C_i$ 为 k-析取范式，$\rho \in \{0,1,\star\}^n$ 为一个限制。令 $t \geq 1$。考虑一个元组 $(r, \ell_i, s_i, B_i, \alpha_i)$，其中：

1. $r \in [1,t]$ 是一个整数。2. $(\ell_1, \ldots, \ell_r) \in [m]^r$ 是一个递增指标列表。3. $(s_1, \ldots, s_r)$ 是一个正整数列表，满足 $s := \sum_{i=1}^{r} s_i \in [t, t+k-1]$。4. $(B_1, \ldots, B_r)$ 是 $[k]$ 的子集列表。此外，对于每个 $i \in [r]$, $|B_i| = s_i$。5. $(\alpha_1, \ldots, \alpha_r)$ 是一个二进制字符串列表。对于每个 $i \in [r]$, $|\alpha_i| = s_i$。

如果存在 $\alpha \in \{0,1\}^n$ 使得....., 我们就称 $(r, \ell_i, s_i, B_i, \alpha_i)$ 是 ρ 的一个 t-见证。 :

- 当我们在 α 上运行 $T_{F|_\rho}$ 时，对于每个 $i \in [r]$，C_{ℓ_i} 都是 $T_{F|_\rho}$ 查询的第 i 项。
- 当 $T_{F|_\rho}$ 发出第 i 个查询集合时，C_{ℓ_i} 中恰有 s_i 个变量尚未知晓，而它们在 V_{ℓ_i} 中的"相对位置"由 B_i 指定。
- 对第 i 个查询集合的响应是 α_i。

我们将见证 $(r, \ell_i, s_i, B_i, \alpha_i)$ 的大小 定义为 $s := \sum_{i=1}^{r} s_i$。

记号约定 当 B_i 和 α_i 与一个项 C_{ℓ_i}相关联时，我们有时会稍微滥用记号，用 B_i 指代它在 V_{ℓ_i} 中对应的变量集合（回忆一下， V_{ℓ_i} 是出现在 C_{ℓ_i}中的变量集合），并用 α_i 表示一个部分赋值 $\alpha_i \in \{0,1,\star\}^n$，其中我们只对 α_i 的 B_i 部分赋值，而将其他坐标留作未赋值状态。在本节中，我们通常用"见证"指代 t-见证，因为我们始终是在固定的 t下进行分析。

容易看出，如果 决策树深度$(F|_\rho) \geq t$，那么必然存在一个见证，使得 规范决策树$(F|_\rho) \geq t$[6]。到目前为止，要界定选到坏限制的概率，一个自然的想法是：(1) 枚举所有可能的见证；(2) 计算随机 ρ 具有此类见证的概率；(3) 对它们使用并集界。不幸的是，这对我们来说代价太高：列表 $\binom{m}{t}$ 至少有 (ℓ_i)种可能的取值。要使这种方法行得通，我们必须将随机限制具有某个特定见证的概率界定为 m^{-t}，而这看起来极其困难，甚至可能根本无法做到。

事实证明，我们可以避免枚举见证中的 (ℓ_i) 部分。为了简洁地描述这个想法，我们先来定义部分见证。

定义 5. 令 $F = \bigvee_{i=1}^m C_i$为 k-DNF， $\rho \in \{0,1,\star\}^n$为一个限制。令 $t \geq 1$。考虑一个元组 (r, s_i, B_i, α_i)。如果存在 $(\ell_1, \ldots, \ell_r)$，使得 $(r, \ell_i, s_i, B_i, \alpha_i)$ 是 ρ的一个 t-见证，我们就称 (r, s_i, B_i, α_i) 是 ρ的一个部分 t-见证。

注 *1.* 这里有一个重要观察：如果 (r, s_i, B_i, α_i) 是 ρ 的部分见证，那么只有一个有效列表 (ℓ_i) 能使 $(r, \ell_i, s_i, B_i, \alpha_i)$ 成为 ρ 的见证（要看出这一点，首先注意 ℓ_1 是固定的。在查询 ℓ_1 并得到响应 α_1 后，使用归纳法）。这一观察在第 *5.3* 节和第 *6* 节中会很有用。

接下来，证明通过枚举 r, s_i, B_i, α_i 并对以下数量进行界定来完成：

$$\Pr_\rho[\mathsf{DT}(F|_\rho) \geq t] \leq \sum_{(r,s_i,B_i,\alpha_i)} \Pr_\rho[(r, s_i, B_i, \alpha_i) \text{ is a partial witness for } \rho]. \tag{5}$$

固定一个 $s \in [t, t+k-1]$，大小为 s 的部分见证 (r, s_i, B_i, α_i) 至多有 $(4k)^s$ 个。对于其中每一个，如果我们能将其概率界定为例如 $(2p)^s$，那么引理便得证。

见证搜索器 现在，给定一个限制 ρ和一个候选部分见证 (r, s_i, B_i, α_i)，我们如何判断是否存在一个列表 $(\ell_i)_i$ ，使得 $(r, \ell_i, s_i, B_i, \alpha_i)$ 构成 ρ 的一个见证？我们将通过设计并使用一个简单的过程来实现这一点，我们称之为见证搜索器。具体而言，我们的搜索器接收一个限制 ρ、一个部分见证 (r, s_i, B_i, α_i) 和一个建议串 $y \in \{0,1\}^n$ 作为输入。它输出一个完整见证 $(r, \ell_i, s_i, B_i, \alpha_i)$，或一个特殊的错误

[6]然而，我们注意到反过来未必成立，因为我们的见证只能反驳浅层规范决策树存在这一说法。

符号。其描述见算法 2。

算法 2： 见证搜索器

输入： 一个析取范式 $F = \bigvee_{i=1}^{m} C_i$、一个限制 $\rho \in \{0,1,\star\}^n$、一个部分见证 (r, s_i, B_i, α_i)、一个建议串 $y \in \{0,1\}^n$。**初始化：** $z \leftarrow \rho \circ y. j^* \leftarrow 0. c \leftarrow 1.$**当** $c \le r$ **时**查找第一个满足条件 C_j 的 $j > j^*$，其中该条件由 z 满足。如果不存在这样的 i，则返回错误。设置 $\ell_c \leftarrow j$，并将 B_c, α_c 与 C_{ℓ_c} 关联。将 z 的 B_c 部分替换为 α_c。也就是说，$z \leftarrow \rho \circ \alpha_1 \circ \cdots \circ \alpha_c \circ y. c \leftarrow c+1. j^* \leftarrow j.$**返回** $(r, \ell_i, s_i, B_i, \alpha_i)$

在下文中，我们用 $\mathcal{S}$ 表示算法 2。

引理 6。 若 (r, s_i, b_i, α_i) 是 ρ的一个部分见证，则存在一个建议串 y，使得 $\mathcal{S}$找到 $(\ell_i)_{i=1}^{r}$。更重要的是，对于均匀随机的 $\mathbf{y} \sim \mathcal{U}_n$，$\mathcal{S}$ 以恰好 2^{-s}的概率找到 $(\ell_1, \ldots, \ell_r)$。

证明。 若 $(r, s_i, \ell_i, b_i, \alpha_i)$ 是 ρ的一个见证，它记录了 $T_{F|_\rho}$ 在某个输入$\alpha \in \{0,1\}^n$上的运行记录。这意味着 C_{ℓ_1} 是 $T_{F|_\rho}$查询的第一项，因此：

1. C_{ℓ_1} 之前的每一项都被 ρ 判定为假（因为 $T_{F|_\rho}$ 跳过了它们）。
2. C_{ℓ_1} 与 ρ 一致（因为$T_{F|_\rho}$ 不确定 C_{ℓ_i}的值（α ）.

现在，考虑在（r、s_i、b_i、α_i）上运行 $\mathcal{S}$，$\mathcal{S}$也会跳过 C_{ℓ_1}之前的所有项。在 C_{ℓ_1}处，由于 ρ与 C_{ℓ_1}一致，随机串 $\rho \circ \mathbf{y}$以 $2^{-|B_1|}$的概率满足 C_{ℓ_1}。以此为条件，在之后的执行中，我们修改 z，并将与 B_1 对应的部分替换为α_1。

此时，我们回过头来检查$T_{F|_\rho}$在 α上的执行。由于（r、s_i、ℓ_i、b_i、α_i）是$T_{F|_\rho}$在 α上的运行记录，我们知道，α_1是$T_{F|_\rho}$从 C_{ℓ_1}中查询活跃变量时收到的响应。由于$T_{F|_\rho}$向 C_{ℓ_2}中的活跃变量发出了第二批查询，这意味着从 C_{ℓ_1+1} 到 C_{ℓ_2-1} 的项都被 $\rho \circ \alpha_1$否定，且C_{ℓ_2}与 $\rho \circ \alpha_1$一致。于是，$\rho \circ \alpha_1 \circ \mathbf{y}$以 $2^{-|B_2|}$的概率满足C_{ℓ_2}。以此为条件，搜索器将使用字符串 $\rho \circ \alpha_1 \circ \alpha_2 \circ y$继续执行，我们可以再次考虑 $T_{F|_\rho}$在查询 C_{ℓ_2}之后的执行。如此反复，直到我们确定了 r 的所有指标ℓ_1、……、ℓ_r并退出该过程。总之，我们已经证明存在 y，使得 $\mathcal{S}$找到列表 $(\ell_i)_{i=1}^{r}$，而采样这样一个 y 的概率为

$$2^{-\sum_{i=1}^{c} |B_i|} = 2^{-s}$$

正如所希望的。 □

解耦如果我们仔细检查 $\mathcal{S}$ 的执行过程，就会发现它只需要知道字符串 $z = \rho \circ y$ 就能运行。具体来说，它不需要知道 z 的哪一部分在限制 ρ中是固定的。因此，我们可以修改 $\mathcal{S}$，得到一个搜索器 $\mathcal{S}'$：现在，$\mathcal{S}'$的输入是一个字符串 z

和$(r,\ s_i,\ B_i,\ \alpha_i)$。否则，它的运行方式与 $\mathcal{S}$完全相同。我们将 $\mathcal{S}'$ 的输出记为$\mathcal{S}'(z,\ (r,\ s_i,\ B_i,\ \alpha_i))$。根据引理6，我们有

$$\begin{aligned}&\Pr_{\rho\sim\mathcal{R}}[(r,s_i,B_i,\alpha_i) \text{ is a partial witness for } \rho]\\ &\leq 2^s\cdot\Pr_{\rho\sim\mathcal{R},y\sim\mathcal{U}_n}[\mathcal{S}'(\rho\circ y,(r,s_i,B_i,\alpha_i)) \text{ is a witness for } \rho].\end{aligned}$$

记 $\rho=\Lambda[x,\star]$。我们观察到

$$\mathbb{I}\{\ (r,\ell_i,s_i,B_i,\alpha_i) \text{ is a witness for } \rho\}\leq \mathbb{I}\left\{\left(\bigcup_{j=1}^{r}B_j\right)\subseteq\Lambda\right\}. \tag{6}$$

对于一个 $(t+k)$-路 p-有界集合 $\mathbf{\Lambda}$，右侧事件发生的概率为 $p^{\sum_j|B_j|}$。于是，我们有

$$\begin{aligned}&\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},0],y\sim\mathcal{U}_n}[\ \mathcal{S}'(\rho\circ y,(r,s_i,B_i,\alpha_i)) \text{ is a witness for } \rho]\\ &=\mathbb{E}_{\mathbf{\Lambda}}\mathbb{E}_{x,y\sim\mathcal{U}_n}\left[\mathbb{I}\{\mathcal{S}'(\mathbf{\Lambda}[x,y],(r,s_i,B_i,\alpha_i)) \text{ is a witness for } \rho\}\right]\\ &=\mathbb{E}_{z\sim\mathcal{U}_n}\left[\Pr_{\mathbf{\Lambda}}[\mathcal{S}'(z,(r,s_i,B_i,\alpha_i)) \text{ is a witness for } \rho]\right]\\ &\leq p^s,\end{aligned}$$

其中第三个等号是因为 $\Lambda[x,y]$ 在 x时服从 $\mathcal{U}_n$ ， $y\sim\mathcal{U}_n$，最后一个不等式由（6）以及 $(t+k)$-路 p-有界性质可得，其中 $\mathbf{\Lambda}$。

总结 我们通过枚举所有的 (r,s_i,B_i,α_i) 并求和来完成证明。

$$\begin{aligned}\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[\mathsf{DT}(F|_\rho)\geq t]&\leq\sum_{(r,s_i,B_i,\alpha_i)}\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[(r,s_i,B_i,\alpha_i) \text{ is a partial witness for } \rho]\\ &\leq\sum_{(r,s_i,B_i,\alpha_i)}2^s\cdot\mathbb{E}_{z\in\mathcal{U}_n}\left[\Pr_{\mathbf{\Lambda}}[\mathcal{S}'(z,(r,s_i,B_i,\alpha_i)) \text{ is a witness for } \mathbf{\Lambda}[\mathbf{x},\star]]\right]\\ &\leq\sum_{s=t}^{t+k}2^s(4k)^sp^s\\ &\leq(10kp)^t.\end{aligned} \tag{7}$$

5.3 证明当 x 是伪随机时

现在我们考虑 $\mathbf{x}$ 并非真正随机的情况。甚至 $\mathbf{x}$ 与 $\mathbf{\Lambda}$之间也可能存在相关性。对于 $\mathbf{x}$ ，我们唯一的要求是，对每个固定的 $\mathbf{\Lambda}$，$\mathbf{x}$ 都是一个伪随机字符串，能够ε-欺骗大小至多为 m的合取范式。首先，根据备注 1，可得以下等式。

$$\mathbb{I}\{(r,s_i,B_i,\alpha_i) \text{ is a partial witness for } \rho\}=\sum_{\ell_i}\mathbb{I}\{(r,\ell_i,s_i,B_i,\alpha_i) \text{ is a witness for } \rho\}. \tag{8}$$

如果我们检查（7）中的推导，实际上可以得出以下结论：

$$\sum_{(r,s_i,B_i,\alpha_i)}\sum_{\ell_i}\Pr_{\rho\sim\mathbf{\Lambda}[\mathcal{U}_n,\star]}[(r,\ell_i,s_i,B_i,\alpha_i) \text{ is a witness for } \rho]\leq(10kp)^t. \tag{9}$$

接下来，固定集合 Λ 和元组 $(r, \ell_i, s_i, B_i, \alpha_i)$，我们考虑谓词

$$h_{\Lambda}^{(r,\ell_i,s_i,B_i,\alpha_i)}(x) := \mathbb{I}\{(r, \ell_i, s_i, B_i, \alpha_i) \text{ is a witness for } \Lambda[x, \star]\}.$$

当上下文已足以明确时，我们省略 h 的上标和下标。我们声称，h 可以由大小至多为 m 的合取范式实现。为此，注意到 $h(x) = 1$ 当且仅当以下条件全部成立。

- 对于每个 $j < \ell_1$，C_j 都被 $\Lambda[x, \star]$ 否定。要使这一点成立，C_j 中至少有一个变量未被 Λ 选中，且其赋值与它在 C_j 中出现时的取值相反。注意，我们可以对相关变量或其否定取析取，以检验 C_j 是否被否定。
- 对于 $j = \ell_1$，C_{ℓ_1} 不可能被 $\Lambda[x, \star]$ 否定，这意味着 C_{ℓ_1} 中所有未被 Λ 选中的变量，其赋值都不会与 C_{ℓ_1} 矛盾。我们可以对这些变量取合取来检验这一点。
- 对于每个 $\ell_1 < j < \ell_2$，C_j 都被 $\Lambda[x, \alpha_1]$ 否定，这意味着 C_j 要么被 α_1 否定，要么 C_j 中至少有一个未被 Λ 选中的变量被赋予相反的取值。如果 C_j 被 α_1 否定，则无需检验。否则，我们可以对相关变量取析取来检验。
- 同样的推理也适用于 C_j 中的每个 $j \geq \ell_2$。最后，在完成对C_{ℓ_r}的验证后，证明即告完成。

总之，我们可以用一个合取范式（析取的合取）来实现 h。显然，该合取范式的规模不超过 F的规模，也就是 m。因此，我们得出结论：$\mathbf{x}$ ε-愚弄 h。也就是说，

$$\left| \Pr_{x \sim \mathcal{U}_n}[h_{\Lambda}^{(r,\ell_i,s_i,B_i,\alpha_i)}(x)] - \Pr_{x \sim \mathbf{x}}[h_{\Lambda}^{(r,\ell_i,s_i,B_i,\alpha_i)}(x)] \right| \leq \varepsilon. \tag{10}$$

最后，我们有

$$\begin{aligned}
\Pr_{\rho \sim \mathbf{\Lambda}[\mathbf{x},\star]}[\mathsf{DT}(F|_\rho) \geq t] &\leq \sum_{(r,s_i,B_i,\alpha_i)} \Pr_{\rho \sim \mathbf{\Lambda}[\mathbf{x},\star]}[(r, s_i, B_i, \alpha_i) \text{ is a partial witness for } \rho] \\
&\leq \sum_{(r,s_i,B_i,\alpha_i)} \sum_{\ell_i} \Pr_{\rho \sim \mathbf{\Lambda}[\mathbf{x},\star]}[(r, \ell_i, s_i, B_i, \alpha_i) \text{ is a witness for } \rho] \\
&\leq \sum_{(r,s_i,B_i,\alpha_i)} \sum_{\ell_i} \varepsilon + \Pr_{\rho \sim \mathbf{\Lambda}[\mathcal{U}_n,\star]}[(r, \ell_i, s_i, B_i, \alpha_i) \text{ is a witness for } \rho] \\
&\leq (10kp)^t + (4m)^{t+k} \cdot \varepsilon.
\end{aligned}$$

这里，第一行根据第 5.2 节中的论证得出。第二行由（8）成立。第三行由（10）得出。最后一行成立，因为（9），且至多有 $(4m)^{k+t}$ 个可能的元组$(r, \ell_i, s_i, B_i, \alpha_i)$。

6 对多重切换引理进行去随机化

本节中，我们证明引理 4。回顾其陈述。

引理回顾 4。 设 t，w，k，n，$m \geq 1$均为整数。设 p，$\varepsilon > 0$均为实数。设 $\mathcal{F} = \{F_1, \ldots, F_m\}$是一个由大小为 m k 的析取范式组成的列表，其输入为 $\{0,1\}^n$。设 $(\mathbf{\Lambda}, \mathbf{x})$ 是一个满足以下条件的联合随机变量：

- $\mathbf{\Lambda}$ 是 $[n]$ 的一个$(t+k)$-*wise* p-有界子集，
- 在给定 $\mathbf{\Lambda}$ 的条件下，$\mathbf{x}$ 是一个能对大小为(m^2) 的合取范式实现 ε-伪随机的随机串。

于是我们有

$$\Pr_{T\sim\mathbf{\Lambda},x\sim\mathbf{x}}[\mathcal{F}|_{\Lambda[x,\star]} \textit{ does not have } w\textit{-partial depth-}t\textit{ DT}] \leq 4m^{t/w}(24pk)^t + (24m)^{t+k}\cdot\varepsilon.$$

本节余下部分将用于证明引理 4。第 6.1 节介绍一些预备工具，并概述证明思路。第 6.2 节讨论 $\mathbf{x}$ 是真正随机的情形。该节扩展了第 5.2 节中的思路，结构也相当相似。最后，第 6.3节讨论 $\mathbf{x}$ 是伪随机的情形，并完成证明。

6.1 预备知识

规范部分决策树 设 ρ 是一个随机限制，在该限制下 $\mathcal{F}|_\rho$ 没有一个 w-部分深度为t 的决策树。设 $\beta\in\{0,1\}^n$ 是一个未知串，我们的决策树将对其进行查询。我们考虑以下尝试（见算法 3），为 $\mathcal{F}|_\rho$构造一棵部分决策树。这里 $z\in\{0,1\}^n$ 是一个待选定的辅助串。每一种 $z\in\{0,1\}^n$ 的取值都会为 $\mathcal{F}$生成一棵不同的部分决策树。在 [ST19], 中，该构造被称为“规范部分决策树”。

算法 3： 规范部分决策树

输入： 析取范式列表 $\mathcal{F}=\{F_1,\ldots,F_m\}$、对字符串 $\beta\in\{0,1\}^n$ 的黑盒访问，以及一个 n
辅助串 $z\in\{0,1\}^n$。初始化：$x\leftarrow(\star)^n$。$j\leftarrow 1$。计数器 $\leftarrow 0$。当计数器 $<t$ 时，
执行：找到最小的 $i\geq j$，使得 DT($F_i|_x$) $>w$。若不存在这样的 i，则退出循环。
$y\leftarrow(\star)^n$。$I\leftarrow\emptyset$。当 $F_i|_{x\circ y}(\star)$ 不是常量且计数器 $<t$ 时，执行：C_i，$_q\leftarrow$ 将要查询
的项$T_{F_i|_{x\circ y}}$。B_i，$_q\leftarrow$ 中未知变量的集合C_i，$_q|_{x\circ y}$。y_{B_i}，$_q\leftarrow z_{B_i}$，$_q$。$I\leftarrow I\cup B_i$，$_q$。
计数器 $\leftarrow$ 计数器 $+|B_i$，$_q|$。查询 β_I，并设置 $x_I\leftarrow\beta_I$。$j\leftarrow i$。

返回 x

注意，在算法 3 中，一个公式 F_i 可能会被外层“while”循环选中多次。

证明概述 在深入形式化证明之前，我们先在此尝试给出一些（过度简化的）直觉。如果 $\mathcal{F}$ 不具有 w-部分决策树，且其深度为 t，那么算法 3 就无法针对每个辅助串 t构造深度为 z的部分决策树。不过，我们只会考虑一些

“由对手选择的” z。也就是说，我们只考虑那些 z，它们会诱使算法 3对每个选定的公式至少进行 w 次查询。这样一来，我们就能保证算法 3 总共只会选择 $\frac{t}{w}$ 个公式。

然后，我们可以付出 $m^{\frac{t}{w}}$ 的因子，来枚举 $\frac{t}{w}$ 个 $\mathcal{F}$公式的子集，并计算算法 3 在这些公式上“陷入停滞”的概率。固定这些 $\frac{t}{w}$ 个公式后，我们可以使用与第 5.2 节类似的思路，将概率上界限制为 $O(pk)^t$。总而言之，我们得到

$$\Pr[\text{bad restriction}] \leq m^{\frac{t}{w}} \cdot O(pk)^t.$$

这里我们只考虑限制串 $\mathbf{x}$ 完全随机的情形。对于 $\mathbf{\Lambda}$ 和 $\mathbf{x}$ 都是伪随机的情形，我们采用与第 5.3 节类似的技巧，并额外付出一个加法因子。

6.2 证明：当x 是真正随机时

我们从考虑 $\mathbf{x}$ 是真正随机串的情形开始证明。固定 ρ 为一个坏限制，在该限制下 $\mathcal{F}|_\rho$ 无法具有 w-部分深度为 t 的决策树。这意味着，对于每个 $z \in \{0,1\}^n$，算法 3 都无法为 $\mathcal{F}$ 构造部分决策树。不过，为了便于分析，我们只考虑一类特殊的字符串 z。下面给出定义。

定义 6。 设 ρ是 $\mathcal{F}$的一个坏限制。我们称 $z \in \{0,1\}^n$为一个 强力的 $(w,\ t)$-反驳，ρ如果存在 $\beta \in \{0,1\}^n$满足以下条件：算法 3在输入 $(\mathcal{F}|_\rho,\ \beta,\ z)$上至少进行 t 次查询。此外，每次查询 β_I时，都保证 $|I| \geq w$。

坏限制总是存在强力反驳，如下所示。

引理 7。 如果 $\mathcal{F}|_\rho$ 没有深度为 t的 w-部分决策树，那么存在一个强力的$(w,\ t)$-反驳，使其对应于 ρ。

证明。 如果 $\mathcal{F}_\rho$ 没有 w-部分决策树，其深度为 t，这意味着算法 3 无法为每个 $z \in \{0,1\}^n$构造一个深度为 t 的 w-部分决策树。接着，我们按如下方式构造一个强力反驳z及其对应的对抗性输入 β。

- 最初，我们设置 $z = \beta = (\star)^n$。然后，我们监控算法 3 在 $(\mathcal{F}, \beta, z)$上的执行，并在算法 z, β 3 访问 时逐步填入这些值。
- 每当算法 3 在外层 while 循环中选择一个公式 $F_i|_{\rho \circ x}$ 时，我们就执行以下操作：– 在内层 while 循环中运行时，算法 3 需要查阅辅助串 z。我们可以用一种方式填入 z 中的相关变量，使得算法 3 在循环中至少查阅 w 位 z。由于决策树深度 DT($F_i|_{\rho \circ x}$) $\geq w$，这总是可行的。– 完成内层循环后，算法 3 会查询 β_I。此时，我们以一种方式设置β_I，使得 $\mathcal{F}|_{\rho \circ x \circ \beta}$不具有深度为（$t$ – 计数器）的 w-部分决策树深度。这总是可行的，只要 $\mathcal{F}_{\rho \circ x}$ 不具有深度为（t – 计数器 $+ |I|$）的 w-部分决策树深度；根据归纳法，这一条件成立。
- 在 Al 之后 g算法 3返回 ，我们将剩余的 g 位填 β, z 为零。 □

在证明了引理 7之后，我们自然地给出如下“全局见证”的定义。

定义 7。 设 t, w 是两个整数。考虑一个由 k 个析取范式组成的列表 $\mathcal{F} = \{F_1, \ldots, F_m\}$。假设 $\rho \in \{0,1,\star\}^n$ 是一个限制。令 $(R, L_i, S_i, W_i, \beta_i)$ 为一个元组，其中

1. $1 \le R \le \frac{t}{w}$ 是一个整数；2. $1 \le L_1 \le L_2 \le \cdots \le L_R \le m$ 是由以下元素组成的列表：R 非递减指标；3. $S_1 \ldots,$ S_R 是由以下元素组成的列表：R 满足 $\sum_{i=1}^{R} S_i \in [t, t+k]$ 的整数；4. W_1、……、W_R 是见证列表（根据定义 4）。对于每个 $i \in [R]$，W_i 的大小为 S_i；5. β_1、……、β_R 是 R 字符串，其中 $|\beta_i| = S_i$ 对每个 $i \in [R]$ 均成立。

如果该元组满足以下条件，我们称其为 ρ 的 (w,t)-全局见证。

1. 令 $\rho_1 = \rho$。W_1 是 $F_{L_1}|_{\rho_1}$ 的 S_1-见证。

2. 对每个 $i \ge 2$，令 $I_{i-1} \subseteq [n]$ 为涉及 W_{i-1} 的变量集合。注意，由于 W_{i-1} 的大小为 S_{i-1}，因此 $|I_{i-1}| = S_{i-1}$。将 β_{i-1} 视为 $\{0,1,\star\}^n$ 中的部分赋值，其中仅设置部分 $\beta_{i-1,\ I_{i-1}}$，其他坐标则填入 $\star$。构造 $\rho_i = \rho_{i-1} \circ \beta_{i-1}$。那么 W_i 是 $F_{L_i}|_{\rho_i}$ 的 S_i-见证。

全局见证的大小定义为 $\sum_{i=1}^{R} S_i$。

根据引理 7，我们有：

推论 1。 考虑一个由 k-析取范式 $\mathcal{F} = \{F_1, \ldots, F_m\}$ 组成的列表。假设 $\rho \in \{0,1,\star\}^n$ 是一个限制，使得 $\mathcal{F}|_\rho$ 不具有 w-部分深度为 t 的决策树。那么存在一个针对 ρ 的(w,t)-全局见证。

证明。 根据引理 7，存在一个强大的 (w,t)-反驳，针对 $\mathcal{F}|_\rho$。取一个这样的反驳 z及其对抗性输入 β。检查算法 3 在 $(\mathcal{F}|_\rho, \beta, z)$ 上的执行并记录执行记录。执行记录包含所需的元组。 □

类似于我们在第 5.2 节中的做法，我们定义全局部分见证。

定义 8。 设 t、w 是两个整数。考虑一个由 k-析取范式 $\mathcal{F} = \{F_1, \ldots, F_m\}$ 组成的列表。假设 $\rho \in \{0,1,\star\}^n$ 是一个限制。令 $(R, L_i, S_i, P_i, \beta_i)$ 为一个元组，其中

1. $1 \le R \le \frac{t}{w}$ 是一个整数；2. $1 \le L_1 \le L_2 \le \cdots \le L_R \le m$ 是一个由 R 个非递减指标组成的列表；3. $S_1 \ldots, S_R$ 是一个由 R 个整数构成的列表，满足$\sum_{i=1}^{R} S_i \in [t, t+k]$；4. $P_1, \ldots, P_R$ 是一个部分见证列表。对于每个 $i \in [R]$，P_i 的大小为 S_i。5. $\beta_1, \ldots, \beta_R$ 是 R 个字符串，其中对于每个 $i \in [R]$，都有 $|\beta_i| = S_i$。

如果我们可以补全 P_i，从而得到一个见证W_i，使得对于每个 $i \in [R]$，$(R, L_i, S_i, W_i, \beta_i)$都是 ρ的全局见证，那么我们称$(R, L_i, S_i, P_i, \beta_i)$是 ρ的(w,t)-全局部分见证。

备注 2. 同样，通过归纳法很容易看出，对于每个全局部分见证，恰好只有一种方式可以将其补全并得到一个全局见证。

通过简单计数可知，对于每个 $S \in [t, t+k]$，大小为 S 的可能全局部分见证至多有 $2m^{t/w}(12k)^S$ 个：

1. 首先，在 $\mathcal{F}$ 中至多有 $2m^{t/w}$ 种方式选择 $\frac{t}{w}$ 个公式。2. 然后，将“查询预算”的 s 个单位划分为 R 个部分见证，方式有 3^S 种，例如 $S = \sum_{i=1}^{R} S_i$；再将每个部分见证的预算进一步划分，方式有 $S_i = \sum_{j=1}^{r_i} s_{i,j}$ 种。3. 接着，为每个部分见证构造集合 $B_{i,j}$，至多有 k^S 种方式。4. 最后，选择 β_i 的方式有 4^S 种，同时还可以为每个部分见证选择 $\alpha_{i,j}$。

固定一个全局部分见证 $(R, L_i, S_i, P_i, \beta_i)$ 后，我们尝试界定一个随机ρ 具有该见证的概率，其上界为 $(2p)^k$。我们将见证搜索器（算法 2）作为子程序，首先设计一个全局见证搜索器。同样，全局见证搜索器需要访问一个建议串$y \in \{0,1\}^n$。参见算法 4。

算法 4： 全局见证搜索器

输入：析取范式列表 $\mathcal{F} = \{F_1, \ldots, F_m\}$、限制 $\rho \in \{0,1,\star\}^n$、全局部分见证 $(R, L_i, S_i, P_i, \beta_i)$ 和建议串 $y \in \{0,1\}^n$。初始化：$z \leftarrow \rho \circ y$。$c \leftarrow 1$。$\rho^{(1)} \leftarrow \rho$。当 $c \le R$ 时，针对 $(F_{L_c}, \rho^{(c)}, P_c, y)$ 运行算法 2。如果报告错误，则报告错误并终止该过程。否则，令 W_c 为返回的见证。$I_c \leftarrow$ 涉及的变量集合。将 β_c 识别为部分赋值，其中仅固定 β_{I_c}。$\rho^{(c+1)} \leftarrow \rho^{(c)} \circ \beta_c$。$c \leftarrow c+1$。

返回 x

接下来的 内容中，我们用 $\mathcal{S}$ 表示算法 4 以简化表述。根据引理 6，我们证明：

引理 8. 如果$(R, L_i, S_i, P_i, \beta_i)$ 确实是 ρ 的全局部分见证，那么存在一个建议串y，使得 $\mathcal{S}$ 找到$(R, L_i, S_i, P_i, \beta_i)$。更重要的是，对于均匀随机的$\mathbf{y} \sim \mathcal{U}_n$，$\mathcal{S}$ 以恰好 2^{-S} 的概率找到$(\ell_1, \ldots, \ell_r)$，其中 $S := \sum_{i=1}^{R} S_i$ 是全局见证的大小。

证明。 我们考察 $\mathcal{S}$的执行过程。它首先将算法 2 作为子程序运行，尝试为 P_1找到 W_1。在 $\mathbf{y}$上以 2^{-S_1} 的概率，算法 2成功找到 W_1。假设这一情况已经发生，那么我们只确定了 $\mathbf{y}_{I_1}$的赋值，而 $\mathbf{y}$ 的其余部分仍然是均匀随机的。还要注意，我们将 $\rho^{(2)}_{I_1}$ 设为 β_1，这会在后续执行中隐藏 $\mathbf{y}_{I_1}$。这样我们就可以进行归纳法并完成证明。 □

解耦 现在，我们可以观察到，$\mathcal{S}$ 只需要知道 $z = \rho \circ y$ 就能运行。具体来说，它不需要知道 z 的哪一部分在 ρ 中是固定的。因此，我们可以修改 $\mathcal{S}$，得到一个搜索器 $\mathcal{S}'$：现在，$\mathcal{S}'$ 的输入是一个字符串 z 和一个全局部分见证 $(R, L_i, S_i, P_i, \beta_i)$。除此之外，它的运行方式与 $\mathcal{S}$完全相同。我们将 $\mathcal{S}'$的输出记为 $\mathcal{S}'(z, (R, L_i, S_i, P_i, \beta_i))$。根据引理 8，

我们有

$$\begin{aligned}&\Pr_{\rho\sim\mathcal{R}}[(R,L_i,S_i,P_i,\beta_i)\text{ is a global partial witness for }\rho]\\&\le 2^S\cdot\Pr_{\rho\sim\mathcal{R},y\sim\mathcal{U}_n}[\mathcal{S}'(\rho\circ y,(R,L_i,S_i,P_i,\beta_i))\text{ is a global witness for }\rho]\end{aligned}$$

我们观察到

$$\mathbb{I}\{\ (R,L_i,S_i,W_i,\beta_i)\text{ is a global witness for }\rho\}\le\mathbb{I}\left\{\left(\bigcup_{j=1}^{R}I_j\right)\subseteq\Lambda\right\}.\tag{11}$$

对于一个 $(t+k)$-路 p-有界 $\mathbf{\Lambda}$，右侧事件以概率 $p^{\sum_j|I_j|}$ 成立。因此，我们有

$$\begin{aligned}&\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},0],y\sim\mathcal{U}_n}[\ \mathcal{S}'(\rho\circ y,(R,L_i,S_i,P_i,\beta_i))\text{ is a global witness for }\rho]\\&=\mathbb{E}_{\mathbf{\Lambda}}\mathbb{E}_{x,y\sim\mathcal{U}_n}\left[\mathbb{I}\{\mathcal{S}'(\mathbf{\Lambda}[x,y],(R,L_i,S_i,P_i,\beta_i))\text{ is a global witness for }\rho\}\right]\\&=\mathbb{E}_{z\sim\mathcal{U}_n}\left[\Pr_{\mathbf{\Lambda}}[\mathcal{S}'(z,(R,L_i,S_i,P_i,\beta_i))\text{ is a global witness for }\rho]\right]\\&\le p^S,\end{aligned}$$

其中第三行是因为 $\Lambda[x,y]$ 在 x时服从 $\mathcal{U}_n$， $y\sim\mathcal{U}_n$，而最后一个不等式由（11）以及 $(t+k)$-路 p-有界性质得出，该性质属于 Λ。

总结 我们通过枚举所有 (R,L_i,S_i,P_i,β_i) 并求和来完成证明。

$$\begin{aligned}&\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[\mathcal{F}|_\rho\text{ does not have }w\text{-partial depth-}t\text{ decision tree}]\\&\le\sum_{(R,L_i,S_i,P_i,\beta_i)}\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[(R,L_i,S_i,P_i,\beta_i)\text{ is a global partial witness for }\rho]\\&\le\sum_{(R,L_i,S_i,P_i,\beta_i)}2^S\cdot\mathbb{E}_{z\in\mathcal{U}_n}\left[\Pr_{\mathbf{\Lambda}}[\mathcal{S}'(z,(R,L_i,S_i,P_i,\beta_i))\text{ is a global witness for }\mathbf{\Lambda}[\mathbf{x},\star]]\right]\\&\le\sum_{S=t}^{t+k}2m^{t/w}2^S(12k)^Sp^S\\&\le 4m^{t/w}(24kp)^t.\end{aligned}\tag{12}$$

6.3 证明：当x 是伪随机时

现在考虑$\mathbf{x}$ 不是真正随机的情形。我们对$\mathbf{x}$ 的唯一要求是，对于每个固定的 $\mathbf{\Lambda}$，$\mathbf{x}$ 都是一个 ε-伪随机字符串，能够骗过大小至多为 m的合取范式。首先，根据注2，可得以下等式。

$$\begin{aligned}&\mathbb{I}\{(R,L_i,S_i,P_i,\beta_i)\text{ is a global partial witness for }\rho\}\\&=\sum_{W_i:\text{completion of }P_i}\mathbb{I}\{(R,L_i,S_i,W_i,\beta_i)\text{ is a global witness for }\rho\}.\end{aligned}\tag{13}$$

接着，由（12）中的推导可知

$$\sum_{(R,L_i,S_i,W_i,\beta_i)} \Pr_{\rho\sim\mathbf{\Lambda}[\mathcal{U}_n,\star]}[(R,L_i,S_i,W_i,\beta_i) \text{ is a global witness for } \rho] \le 4m^{t/w}(24kp)^t. \tag{14}$$

接下来，固定一个元组 $(R, L_i, S_i, W_i, \beta_i)$ 和 Λ 的一个实例化，我们考虑谓词

$$h_\Lambda^{(R,L_i,S_i,W_i,\beta_i)}(x) := \mathbb{I}\{(R,L_i,S_i,W_i,\beta_i) \text{ is a global witness for } \Lambda[x,\star]\}.$$

当上下文清楚时，我们省略 h的上标和下标。我们声称， h可以由一个大小至多为 $\sum_{i=1}^m$大小$(F_i) \le m^2$的合取范式实现。为此，注意到 $h(x) = 1$ 成立，当且仅当对每个 $i \in [R]$， W_i 都是 $\rho \circ \beta_1 \circ \cdots \circ \beta_{i-1}$的一个见证。根据第6.3节中的论证，可以用大小为大小(F_{L_i})的合取范式来验证这一点。因此，我们可以通过对 h 个合取范式取合取来计算 R ，所得结果本身也是一个更大的合取范式。因此，我们得出$\mathbf{x}$ ε-伪随机化 h。也就是说，

$$\left| \Pr_{x\sim\mathcal{U}_n}[h_\Lambda^{(R,L_i,S_i,W_i,\beta_i)}(x)] - \Pr_{x\sim\mathbf{x}}[h_\Lambda^{(R,L_i,S_i,W_i,\beta_i)}(x)] \right| \le \varepsilon. \tag{15}$$

最后，我们有

$$\begin{aligned}
&\Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[\mathcal{F}|_\rho \text{ does not have } w\text{-partial depth-}t \text{ decision tree}] \\
\le &\sum_{(R,L_i,S_i,W_i,\beta_i)} \Pr_{\rho\sim\mathbf{\Lambda}[\mathbf{x},\star]}[(R,L_i,S_i,W_i,\beta_i) \text{ is a global witness for } \rho] \\
\le &\sum_{(R,L_i,S_i,W_i,\beta_i)} \varepsilon + \Pr_{\rho\sim\mathbf{\Lambda}[\mathcal{U}_n,\star]}[(R,L_i,S_i,W_i,\beta_i) \text{ is a global witness for } \rho] \\
\le &\, 4m^{t/w}(24kp)^t + (24m)^{t+k}\cdot\varepsilon.
\end{aligned}$$

这里，第一行依据第6.2 节中的论证和观察（13）。第二行依据（15）。最后一行成立，是因为（14），并且可能的元组最多有$(24m)^{t+k}$ 个$(R, L_i, S_i, W_i, \beta_i)$。

致谢

我谨向导师 Avishay Tal 致谢，感谢他在项目期间与我进行了许多富有启发性的讨论。我也感谢 Lijie Chen 和 Avishay Tal 对早期稿件提出的宝贵意见，这些意见显著改善了论文的表达。

参考文献

[AW89] Miklós Ajtai 和 Avi Wigderson。概率常深度电路的确定性模拟。计算高级研究，5：199–222，1989年。[BIS12] Paul Beame、Russell Impagliazzo 和 Srikanth Srinivasan。用小高度决策树近似 acˆ0，以及一种用于#acˆ0sat 的确定性算法。载于计算复杂性会议，第117–125页。IEEE计算机学会，2012年。[Bra10] Mark Braverman。多对数独立性可以欺骗AC^0 电路。《*ACM*期刊》，57(5)：28:1–28:10，2010年。

[CSS18] Ruiwen Chen、Rahul Santhanam 和 Srikanth Srinivasan。小型阈值电路的平均情形下界与可满足性算法。《理论计算》，14(1)：1–55，2018年。[DETT10] Anindya De、Omid Etesami、Luca Trevisan 和 Madhur Tulsiani。深度 2 电路的改进伪随机生成器。载于 Maria J. Serna、Ronen Shaltiel、Klaus Jansen 和 José D. P. Rolim 编，《近似、随机化与组合优化：算法与技术》，第13届国际研讨会 APPROX 2010 和第14届国际研讨会 RANDOM 2010，西班牙巴塞罗那，2010年9月1日至3日。论文集，计算机科学讲义第6302卷，第504–517页。施普林格，2010年。[DMR+21] Dean Doron、Raghu Meka、Omer Reingold、Avishay Tal 和 Salil P. Vadhan。单调分支程序：伪随机性与电路复杂度。《电子计算复杂性讨论》，第18页，2021年。[FK18] Michael A. Forbes 和 Zander Kelley。适用于任意顺序只读一次分支程序的伪随机生成器。载于 Mikkel Thorup 编，第59届 IEEE计算机科学基础年会，FOCS 2018，法国巴黎，2018年10月7日至9日，第946–955页。IEEE计算机学会，2018年。[GMR+12] Parikshit Gopalan、Raghu Meka、Omer Reingold、Luca Trevisan 和 Salil P. Vadhan。利用较温和的伪随机限制构造更好的伪随机生成器。载于第53届 IEEE计算机科学基础年会，FOCS 2012，美国新泽西州新不伦瑞克，2012年10月20日至23日，第120–129页。IEEE计算机学会，2012年。[GMR13] Parikshit Gopalan、Raghu Meka 和 Omer Reingold。析取范式稀疏化与更快的确定性计数算法。《计算复杂性》，22(2)：275–310，2013年。[GW13] Oded Goldreich 和 Avi Wigderson。计算多线性函数的深度三布尔电路的规模。《电子计算复杂性讨论》，第43页，2013年。[Has89] John Hastad。小深度电路的近乎最优下界。《计算高级研究》，5：143–170，1989年。[Hås14] Johan Håstad。奇偶函数与小深度电路的相关性。《美国工业与应用数学学会计算机学报》，43(5)：1699–1708，2014年。[HHTT21] Pooya Hatami、William Hoza、Avishay Tal 和 Roei Tell。欺骗常深度阈值电路。《电子计算复杂性讨论》，第2页，2021年。[HLV18] Elad Haramaty、Chin Ho Lee 和 Emanuele Viola。有界独立性加噪声可以欺骗乘积。《美国工业与应用数学学会计算机学报》，47(2)：493–523，2018年。[HS19] Prahladh Harsha 和 Srikanth Srinivasan。关于 AC 的多项式近似。《随机结构与算法》，54(2)：289–303，2019年。[IMP12] Russell Impagliazzo、William Matthews 和 Ramamohan Paturi。一种用于 ac^0的可满足性算法。载于 SODA，第961–972页。美国工业与应用数学学会，2012年。

[IMZ12] Russell Impagliazzo、Raghu Meka 和 David Zuckerman。来自收缩的伪随机性。载于第53届IEEE计算机科学基础研讨会年会（FOCS 2012），美国新泽西州新不伦瑞克，2012年10月20日至23日，第111–119页。IEEE计算机学会，2012年。[Kel21] Zander Kelley。切换引理的改进去随机化。载于 Samir Khuller 和 Virginia Vassilevska Williams（编），STOC '21：第53届 ACM SIGACT计算理论研讨会，线上活动，意大利，2021年6月21日至25日，第272–282页。ACM，2021年。[LMN93] Nathan Linial、Yishay Mansour 和 Noam Nisan。常深度电路、傅里叶变换与可学习性。《ACM期刊》，40(3)：607–620，1993年。[LV96] Michael Luby 和 Boban Velickovic。关于析取范式的确定性近似。《Algorithmica》，16(4/5)：415–433，1996年。[LV20] Chin Ho Lee 和 Emanuele Viola。关于有界独立性加噪声的进一步研究：只读一次多项式的伪随机生成器。《计算理论》，16：1–50，2020年。[MRT19] Raghu Meka、Omer Reingold 和 Avishay Tal。宽度为3的分支程序的伪随机生成器。载于 Moses Charikar 和 Edith Cohen （编），第 51 届 ACM SIGACT 计算理论研讨会论文集，STOC 2019，美国亚利桑那州菲尼克斯，2019年6月23日至26日，第626–637页。ACM，2019年。[MZ13] Raghu Meka 和 David Zuckerman。多项式阈值函数的伪随机生成器。《美国工业与应用数学学会计算杂志》，42(3)：1275–1301，2013年。[NW94] Noam Nisan 和 Avi Wigderson。困难性与随机性。《计算机系统科学杂志》，49(2)：149–167，1994年。[ST19] Rocco A. Servedio 和 Li-Yang Tan。由伪随机多重切换引理得到的改进伪随机生成器。载于 Dimitris Achlioptas 和 László A. Végh （编），《近似、随机化与组合优化：算法与技术》，APPROX/RANDOM 2019，2019年9月20日至22日，美国麻省理工学院，马萨诸塞州剑桥，美国，第145卷 LIPIcs，第45:1–45:23页。达格施图尔城堡—莱布尼茨信息学中心，2019年。[Tal17] Avishay Tal。AC^0傅里叶谱的紧界。载于 Ryan O'Donnell （编），第 32 届计算复杂性会议，CCC 2017，2017 年 7 月 6 日至 9 日，拉脱维亚里加，第79卷 LIPIcs，第15:1–15:31页。达格施图尔城堡—莱布尼茨信息学中心，2017年。[TX13] Luca Trevisan 和 Tongke Xue。一个去随机化切换引理以及 AC^0 的改进去随机化。载于第28届计算复杂性会议，CCC 2013，美国加利福尼亚州克洛阿尔托，2013年6月5日至7日，第242–247页。IEEE计算机学会，2013年。[Vad12] Salil P. Vadhan。伪随机性。《理论计算机科学基础与趋势》，7(1-3)：1–336，2012年。[Vio21] Emanuele Viola。AC^0不可预测性。《ACM计算理论汇刊》，13(1)：5:1–5:8，2021年。