能否加速：不透明性验证的条件下界

Jiří Balun ！ 帕拉茨基大学奥洛穆茨分校理学院
Tomáš Masopust ！ 帕拉茨基大学奥洛穆茨分校理学院**Petr Osička** ！ 帕拉茨基大学奥洛穆茨分校理学院

arXiv:2304.09920v1 [cs.FL] 19 Apr 2023

摘要

不透明性是隐私与安全应用的一项性质，用于判断：给定一个系统模型后，一个被动入侵者通过在线观察系统的行为，能否推断出系统的某些“秘密”信息。判定不透明性是一个 PSpace 完全问题，因此，在假设 PSpace 与 PTime 不同的前提下，不存在用于验证不透明性的多项式时间算法。然而，这一假设引出了一个问题：现有的指数时间算法是否已达到可能的最佳效率，还是存在更快的次指数时间算法？我们证明，在（强）指数时间假设成立的前提下，不存在显著快于现有算法的算法。作为附带结果，我们还得到了判定非确定性有限自动机全性的一个新的条件下界（因此也适用于包含关系和等价性）的时间复杂度下界。

2012 年 ACM 学科分类 计算理论 → 形式语言与自动机理论

关键词与短语 有限自动机、不透明性、细粒度复杂性

资助 *Jiří Balun*：获 IGA PrF 2023 026 资助。

1 引言

在隐私与安全应用以及通信协议中，通常需要对系统或其行为的某些信息保密。此类要求对系统的信息流施加了额外限制，文献中对此进行了广泛讨论，并将其归纳为各种性质，包括匿名性 [14, 32, 38], 无干扰 [8, 12, 19,33], 保密性 [1, 3, 10], 安全性 [17], 完美安全性 [46], 和不透明性 [20, 30]。匿名性 是指保护身份信息不被泄露的性质；例如，网络服务器不应能够获知请求的真实来源。无干扰 关注这样一个问题：给定系统的两个输入状态，若它们在指定变量上的取值相同，那么从这两个状态出发的系统行为在观察者看来，是否无法通过对这些指定变量的观察加以区分。保密性 描述观察者能否发现系统的一条轨迹属于秘密轨迹集合；而完美安全性 要求，即使观察者知道系统的所有轨迹集合，也无法通过观察低安全级事件推断出任何有关高安全级事件是否发生的信息。</output>

本文关注（多种类型的）不透明性，从某种意义上说，它概括了前面提到的其他性质；也就是说，上述性质都可以通过归约到不透明性来验证。更具体地说，Alur 等人 [1] 证明了保密性 涵盖非干扰性和完美安全性，Lin [26] 则对所有这些性质进行了广泛讨论和比较。他尤其证明了匿名性 和保密性，以及离散事件系统的可观测性 [27], 可诊断性 [25, 37], 和可检测性 [28, 29, 39]均为不透明性的特例。Wu 等人 [44] 和 Góes 等人 [18] 讨论了不透明性在位置隐私中的应用，Wintenberg 等人 [42] 则将不透明性应用于接触者追踪。

2 能否加快速度：不透明性验证的条件下界

给定一个系统模型，不透明性要回答的是：恶意被动观察者（入侵者）若完全了解模型的结构，能否通过在线观察系统行为的不完整信息，推断出系统的某些“秘密”信息。秘密信息可以建模为系统的一组状态，也可以建模为系统的一组行为。入侵者根据不完整的观察来估计系统的状态/行为；如果对于系统的每个秘密状态/秘密行为，都存在一个对入侵者而言看起来相同的非秘密状态/行为，那么该系统就是不透明的。如果秘密信息以一组状态的形式给出，我们称之为基于状态的不透明性 [9, 11], ；如果秘密信息以一组秘密行为（语言）的形式给出，我们称之为基于语言的不透明性 [4, 15]。文献中已讨论了多种不透明性概念，适用于由自动机和佩特里网建模的系统；概述参见 Jacob 等人 [24] 。本文聚焦于有限自动机模型，以及我们在第 3 节回顾的不透明性概念。

目前用于验证所研究不透明性概念的最快算法，相对于自动机的状态数具有指数时间复杂度。事实上，不透明性验证是一个 PSpace 完全问题 [5, 6, 24], ，因此我们可以得出结论：除非 PTime =PSpace，否则不存在判定不透明性的多项式时间算法。尽管 PTime $\neq$ PSpace 的假设排除了多项式时间算法的存在，但能否找到显著更快（即亚指数时间）的算法仍然是一个悬而未决的问题。

为了获得更强的下界（尽管仍然是有条件的），我们采用指数时间假设 （ETH）及其强版本——强指数时间假设（SETH）。这两个假设由 Impagliazzo 和 Paturi [23]提出，其依据是这样一个观察：到目前为止，我们尚未找到能够在最坏情况下显著快于穷举所有可能真值赋值的算法来求解可满足性问题。特别地，ETH 表明，3-SAT 无法在时间 $2^{o(n)}$ 内求解，其中 n 是变量的数量。然而，它允许使用在时间 $O(c^n)$ 内求解 3-SAT 的算法，其中 $c < 2$。事实上，目前最快的 3-SAT 算法由 Paturi 等人 [31],提出，并由 Hertli [21],改进，其运行时间为 $O^*(1.30704^n)$。随着 k的增加，目前最快的 k-SAT 算法会变得越来越慢；例如，Hertli [21] 给出的最佳 4-SAT 算法的运行时间为 $O^*(1.46899^n)$。这一观察促成了 SETH 的提出：该假设声称，对于任意常数 $c < 2$，总存在足够大的 k ，使得 k-SAT 无法在时间 $O(c^n)$ [23]内求解。这两个假设都意味着复杂性类 PTime 和 NP 是分离的；此外，SETH 蕴含 ETH。

本文表明，在 (S)ETH 假设下，不存在能够显著更快地验证不透明性的算法。具体而言，我们证明，除非 SETH 不成立，否则不存在一种算法，能够判定给定的 n 状态自动机是否满足所考虑的不透明性概念，并在 $O^*(2^{n/c})$ 时间内运行；这一结论对任意 $c > 2$ 均成立（定理 8 和推论 15）。由于我们的构造中字母表中的符号数量没有上界，而在 SETH 假设下，对符号进行标准二进制编码的方法并不适用，因此尚不清楚这一结果是否也适用于固定大小的（二元）字母表自动机。我们在 ETH 而非 SETH 假设下对这个问题进行了部分探讨。我们证明，除非 ETH 不成立，否则不存在一种算法，能够判定给定的 n 状态自动机（其字母表为二元字母表）是否满足所考虑的不透明性概念，并在 $O^*(2^{o(n)})$ 时间内运行。我们的结果总结于表 1；关于复杂度上界，请读者参阅文献 [6, 34]。

作为副产品，我们为判定非确定性自动机（NFA）的全性（以及包含关系和等价性）获得了一个新的条件下界：除非 SETH 不成立，否则不存在这样的 $c > 2$ ，使得一个具有 n 个状态的 NFA 的全性能够在 $O^*(2^{n/c})$ 时间内判定

	下界		上界	
	Γ未固定	$\lvert\Gamma\rvert = 2$		
LBO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$O^*(2^n)$	
CSO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$O^*(2^n)$	
ISO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$O^*(2^n)$	
IFO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$\begin{cases} O^*(2^n) \\ O^*\left(2^{n^2}\right) \end{cases}$	如果 $IQ_{NS} = I_{NS} \times F_{NS}$ 否则
k-SO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$O^*(2^n)$	
INSO	$O^*\left(2^{n/(2+\varepsilon)}\right)$	$O^*(2^{o(n)})$	$O^*(2^n)$	

表 1 在投影P下判定不透明性算法复杂度的概述： $\Sigma^* \to \Gamma^*$，其中 n 是自动机的状态数，且 $\varepsilon > 0$。

（推论 18）。这一结果强化了 Fernau 和 Krebs [16] 的研究成果，该研究表明，如果 ETH 成立，则无法在 $O^*(2^{o(n)})$ 时间内判定一个具有 n 个状态的非确定有限自动机的全性。

2 预备知识

我们假设读者熟悉自动机理论 [22]。对于集合 S， S 的基数记为 $|S|$， S 的幂集记为 2^S。若 S 是单元素集合， $S = \{x\}$，我们通常简写为 x，而不写作$\{x\}$。所有非负整数的集合记为 $\mathbb{N}$。

字母表 Σ是一个有限非空符号集合。 Σ 上的字符串是由 Σ中的符号组成的有限序列。 Σ 上的所有字符串构成的集合记为 Σ^*；空字符串记为 ε。 Σ 上的语言 L 是 Σ^*的子集。对于字符串 $u \in \Sigma^*$，其长度记为 $|u|$。对于任意满足 $\Gamma \subseteq \Sigma$的字母表对（Σ， Γ），我们关联一个态射 P: $\Sigma^* \to \Gamma^*$，其定义为：对于 $a \in \Sigma - \Gamma$， $P(a) = \varepsilon$；对于 $a \in \Gamma$， $P(a) = a$；此类态射通常称为投影。直观来说，投影 P 的作用是擦除所有不属于 Γ的符号； Γ 中的符号通常称为在投影 P下 Σ 的可观测符号。我们以通常的方式将投影 P 从字符串扩展到语言。 P的逆投影是函数 P^{-1}: $\Gamma^* \to 2^{\Sigma^*}$，定义为$P^{-1}(w) = \{w' \in \Sigma^* \mid P(w') = w\}$。

非确定有限自动机（NFA）是一个结构 $\mathrm{A} = (Q, \Sigma, \delta, I, F)$，其中 Q是有限状态集合， Σ 是输入字母表， $I \subseteq Q$ 是初始状态集合， $F \subseteq Q$ 是接受状态集合， δ: $Q \times \Sigma \to 2^Q$ 是一个转移函数，可通过归纳扩展到定义域 $2^Q \times \Sigma^*$。如果接受状态无关紧要，我们就省略它们，简写为 $\mathrm{A} = (Q, \Sigma, \delta, I)$。A 从 $Q_0 \subseteq Q$ 中的状态出发，经由 $F_0 \subseteq F$中的状态所接受的语言是集合 $L_m(\mathrm{A}\ Q_0, F_0) = \{w \in \Sigma^* \mid \delta(Q_0, w) \cap F_0 \neq \emptyset\}$，而 A 从 Q_0 中的状态出发所生成的语言是集合 $L(\mathrm{A}\ Q_0) = L_m(\mathrm{A}\ Q_0, Q)$；特别地，A 所接受的语言是 $L_m(\mathrm{A}) = L_m(\mathrm{A}\ I, F)$，A 所生成的语言是$L(\mathrm{A}) = L(\mathrm{A}\ I)$。如果对于每个状态 $q \in Q$ 和每个符号 $a \in \Sigma$，都有 $|I| = 1$ 且 $|\delta(q, a)| \leq 1$，则 NFA A 是确定性的（DFA）。

对于字母表 $\Gamma \subseteq \Sigma$，我们将通过以下方式得到的确定有限自动机的可达部分定义为$\mathcal{A}$的投影自动机，记作 $P(\mathcal{A})$：将$\mathcal{A}$中的每个转移 (q, a, r) 替换为转移 $(q, P(a)\ r)$，然后进行标准的子集构造 [22]。

我们将$\mathcal{A}$ 的配置 定义为 A 的投影自动机 $P(\mathcal{A})$ 的状态。

(布尔)公式由变量、逻辑连接词符号（合取、析取、否定）以及括号组成。文字是一个变量或其否定。子句是文字的析取。若一个公式是子句的合取，则称其为合取范式(CNF)。若每个子句最多包含 k 个文字，则该公式属于 k-CNF。若存在一种对变量赋予 1 和 0 的赋值，使公式的值为 1，则称该公式可满足。给定一个$k \geq 3$ 和一个 k-CNF 公式， k-CNF 布尔可满足性问题(k-SAT)要求判定该公式是否可满足。若 k-CNF 公式有 n 个变量，枚举所有 2^n 种可能的真值赋值，即可得到一个运行时间为 $O(2^n n^k)$ 的 k-SAT 算法；其中多项式部分 $O(n^k)$ 来自检查至多 n^k 个子句。我们使用记号 O^* 隐去多项式因子，即 $O^*(g(n)) = O(g(n) \cdot poly(n))$.

指数时间假设指出，对于 3-SAT，无法在亚指数时间 $2^{o(n)}$ 内求解，其中 n 是 3-CNF 公式 [23] 中变量的数量。

▶ **假设1** (指数时间假设 (ETH))。 存在某个 $\varepsilon > 0$ ，使得 *3-SAT* 无法在时间 $O(2^{\varepsilon n})$ 内求解，其中 n 是公式中的变量数量。

强指数时间假设指出，对于较大的 k [23]，判定 k-SAT 需要 $O^*(2^n)$ 时间。

▶ **假设2** (强指数时间假设 (SETH))。 对于每个 $\varepsilon > 0$，都存在某个 $k \geq 3$ ，使得k-*SAT* 无法在时间 $O(2^{(1-\varepsilon)n})$ 内求解。

3 不透明性定义

现在我们回顾本文讨论的不透明性概念。我们区分两种类型的不透明性：一种用字符串表示秘密，另一种用状态表示秘密。

基于语言的不透明性是一种性质，它要求对于每个秘密行为，都存在一个在所考虑的投影下与之相同的非秘密行为；在这种情况下，入侵者无法区分秘密行为与非秘密行为。

▶ **定义3**。 一个非确定有限自动机 $A = (Q, \Sigma, \delta, I)$ 相对于互不相交的语言 $L_S, L_{NS} \subseteq L(A)$（分别称为秘密语言和非秘密语言）以及投影 $P\colon \Sigma^* \to \Gamma^*$ ，对于 $\Gamma \subseteq \Sigma$，是基于语言的不透明 *(LBO)*，当且仅当 $L_S \subseteq P^{-1}P(L_{NS})$。

LBO 问题是判定 A 是否相对于 L_S、 L_{NS} 和 P 满足 *LBO*。

这个定义足够一般，可以涵盖其他概念，例如 Best 等人的强非确定性无干扰 或组合不可推断性 [8] 以及 Busi 和 Gorrieri [12],提出的概念，或 Bryans 等人 [9]提出的轨迹不透明性 。为确保包含问题可判定，秘密语言和非秘密语言通常被视为正则语言 [2]。

基于状态的不透明性通过将秘密信息隐藏在状态中来实现。本文讨论五种基于状态的不透明性概念。当前状态不透明性要求入侵者在任何时刻都无法确定系统当前是否处于秘密状态。

▶ **定义 4.** 若对于任意字符串 $w \in \Sigma^*$ ，只要 $\delta(I, w) \cap Q_S \neq \emptyset$，就存在字符串 $w' \in \Sigma^*$ ，使得 $P(w) = P(w')$ 且 $\delta(I, w') \cap Q_{NS} \neq \emptyset$，则称 *NFA* $A = (Q, \Sigma, \delta, I)$ 相对于两个互不相交的集合 Q_S, $Q_{NS} \subseteq Q$ （分别为秘密状态和非秘密状态）以及投影$P\colon \Sigma^* \to \Gamma^*$，对 $\Gamma \subseteq \Sigma$具有当前状态不透明性 *(CSO)*。

要判定的问题是：A 相对于 Q_S, Q_{NS} 和 P 是否具有CSO 。

初始状态不透明性要求入侵者永远无法确定计算是否始于秘密状态。

▶ **定义 5.** 若对于两个互不相交的集合 I_S 和 $I_{NS} \subseteq I$ （分别表示秘密和非秘密初始状态），以及投影 P： $\Sigma^* \to \Gamma^*$，对于 $\Gamma \subseteq \Sigma$，若对每个 $w \in L(A\ I_S)$，都存在 $w' \in L(A\ I_{NS})$ 使得 $P(w) = P(w')$，则称 *NFA* $A = (Q, \Sigma, \delta, I)$ 是初始状态不透明的 （*ISO*）。

问题是判断 A 是否相对于 I_S、 I_{NS} 和 P 满足 ISO 。

初始态与终态不透明性 [43] 将 CSO 和 ISO 都推广了。秘密表示为由初始状态和接受状态组成的状态对集合。因此，ISO 是初始态与终态不透明性的一种特殊情况，其中接受状态不起作用；而 CSO 则是一种特殊情况，其中初始状态不起作用。

▶ 定义 *6*。若一个 *NFA* $A = (Q、\ \Sigma、\ \delta、\ I、\ F)$ 相对于两个互不相交的状态对集合 IQ_S（分别为秘密对和非秘密对）以及投影 P 满足以下条件，则称其具有初始态与终态不透明性（*IFO*）：$\Sigma^* \to \Gamma^*$ 对于每个秘密对 $(q_0、\ q_f) \in IQ_S$以及每个字符串 $w \in L_m(A\ q_0、\ q_f)$，都存在一个非秘密对 $(q'_0、\ q'_f) \in IQ_{NS}$和一个字符串 $w' \in L_m(A\ q'_0、\ q'_f)$，使得 $P(w) = P(w')$。

IFO 问题是判断 A 关于 IQ_S、 IQ_{NS} 和 P 是否具有 *IFO* 性质。

一般而言，判定 IFO 的算法时间复杂度已知为 $O^*(2^{n^2})$；如果 $IQ_S = I_S \times F_S$ 和 $IQ_{NS} = I_{NS} \times F_{NS}$，则为 $O(2^{2n})$，其中 $I_S, I_{NS} \subseteq I$ 和 $F_S, F_{NS} \subseteq F$ [43]。表 1 中的复杂度基于以下观察。

考虑一个 NFA A $= (Q, \Sigma, \delta, I, F)$ 以及两个集合 IQ_S、 $IQ_{NS} \subseteq I \times F$。如果将左分量相同的所有对 (s, f_1)、(s, f_2)、.....、(s, f_k) 替换为一个形如 $(s, \{f_1, f_2, \ldots, f_k\})$ 的对，则 A 的 IFO 性质保持不变。这样一来，需要考虑的对数减少为 n，其中 n 是 A 的状态数。对于每一对 $(s_i, F_i) \in I \times 2^F$，我们定义语言 $L_i = L_m(\text{A}\ s_i, F_i)$ 以及以下语言

$$L_S = \bigcup_{(s_i,F_i)\in IQ_S} L_i \quad \text{and} \quad L_{NS} = \bigcup_{(s_i,F_i)\in IQ_{NS}} L_i\,.$$

因此，判断$\mathcal{A}$ 相对于 IQ_S、 IQ_{NS}和 P 是否为 IFO，等价于判断包含关系 $P(L_S) \subseteq P(L_{NS})$是否成立。由于 L_S和 L_{NS}都可以表示为至多由 n 个$\mathcal{A}$副本组成的非确定有限自动机，因此它们有 $O(n^2)$个状态。两个非确定有限自动机的语言之间的包含关系$P(L_S) \subseteq P(L_{NS})$可以在 $O(n^2 2^{n^2}) = O^*(2^{n^2})$时间内检验；这是一个复杂度上界，与 Saboori 和 Hadjicostis [36],使用格状自动机得到的上界一致。

如果 $IQ_{NS} = I_{NS} \times F_{NS} \subseteq I \times F$，那么 L_{NS}的非确定有限自动机与$\mathcal{A}$相同，其中初始状态为 I_{NS}，终态为 F_{NS}。具体而言，这个自动机有 n 个状态，因此可以在 $O(n^2 2^n) = O^*(2^n)$ 时间内检验包含关系 $P(L_S) \subseteq P(L_{NS})$。

k步不透明性通过要求入侵者无法确定当前状态及其后续 k 个状态是否为秘密，对 CSO 进行了推广。根据定义，CSO 等价于 0 步不透明性。我们采用 Saboori 和 Hadjicostis 的一个定义的略微推广 [35]，该定义由 Balun 和 Masopust [6]加以表述。

▶ 定义*7.* 对于给定的 $k \in \mathbb{N} \cup \{\infty\}$，非确定有限自动机 $A = (Q, \Sigma, \delta, I)$ 关于两个互不相交的秘密状态集和非秘密状态集 $Q_S, Q_{NS} \subseteq Q$，以及投影 P: $\Sigma^* \to \Gamma^*$，若对每个满足 $|P(t)| \leq k$ 且$\delta(\delta(I, s) \cap Q_S, t) \neq \emptyset$的字符串 $st \in L(A)$，都存在一个字符串 $s't' \in L(A)$，使得 $P(s) = P(s')$，$P(t) = P(t')$，且 $\delta(\delta(I, s') \cap Q_{NS}, t') \neq \emptyset$，则称 A 是 k步不透明的（k-*SO*）。

k-SO 问题是判断 A 是否相对于 Q_S、 Q_{NS} 和 P 满足 k-*SO*。

当 k 为无穷大时， k-SO 的一个特殊情形称为无限步不透明性（INSO）。这两个概念对于有限自动机密切相关，因为一个具有 n 个状态的自动机是无限步不透明的，当且仅当它是 $(2^n - 2)$ 步不透明的 [45]。

对于基于状态的不透明性概念，我们可以不失一般性地假设投影 P 是恒等映射；事实上，标记为 $\Sigma - \Gamma$ 中符号的转移可以视为 ε-转移，因此可以通过消除 ε-转移的经典算法将其移除 [22]。该算法不会改变状态数，但可能使转移数增加到原来的平方级。下文中，如果投影是恒等映射，我们将省略它。

4 强指数时间假设下的下界

现在我们证明，在强指数时间假设成立的条件下，不存在显著快于当前已知最佳算法的当前状态不透明性判定算法。

▶**定理 8.** 除非 *SETH* 不成立，否则不存在判定给定 n-状态非确定有限自动机是否满足 CSO 的算法，其运行时间为 $O^*(2^{n/(2+\varepsilon)})$，其中 $\varepsilon > 0$为任意值。

证明。给定一个包含 n 个变量 $X = \{x_1, \ldots, x_n\}$ 和 m 个子句$C = \{c_1, \ldots, c_m\}$的 k-合取范式公式 φ，我们在多项式时间内构造一个 CSO 实例。该实例由一个具有 $N = 2n + 2$ 个状态的非确定有限自动机 A$_\varphi$ 以及分别由秘密状态和非秘密状态组成的集合 Q_S 和 Q_{NS}构成，使得 A$_\varphi$ 相对于 Q_S 和 Q_{NS} 满足 CSO，当且仅当 φ 可满足。因此，如果存在一个能在 $O^*(2^{N/(2+\varepsilon)})$时间内解决 CSO 的算法，那么就存在一个能在 $O(\text{poly}(n)) + O^*(2^{(2n+2)/(2+\varepsilon)}) = O^*(2^{(1-\delta)n})$时间内解决 k-可满足性问题的算法，其中$\delta = \varepsilon/(2 + \varepsilon) > 0$。这与 SETH 矛盾，从而证明了该定理。

直观而言，我们构造 NFA $\mathcal{A}_\varphi$，使得当 A_φ 读取特定类型的字符串（基于 Zimin 词）时，它被迫检查变量$x_1, \ldots, x_n$的所有可能赋值。如果没有任何赋值满足 φ，那么读取完整个字符串后，自动机 A_φ 最终会进入一个仅包含秘密状态的配置，因此 A_φ不是 CSO。另一方面，当遇到一个满足赋值（或该字符串不属于这种特定类型）时，一个非秘密状态会被永久加入A_φ的配置中，因此 A_φ 是 CSO。

形式化地，NFA A$_\varphi = (Q, \Sigma, \delta, I)$ 的状态集为 $Q = \{q_s, q_{ns}\} \cup \{x_i^0, x_i^1 \mid x_i \in X\}$，其中 x_i^r 表示变量 x_i 的 $r \in \{0, 1\}$ 赋值；字母表为$\Sigma = \Gamma = \{a_1, \ldots, a_{n+1}\} \times C$，也就是说，投影 P是恒等映射；秘密状态集为 $Q_S = \{q_s\}$，也就是说，状态 q_s 是唯一的秘密状态，其余状态均为非秘密状态。关于该构造的示例，读者可参见例 14 及证明的其余部分。

令 L 为 φ的文字集合。我们使用函数 $cl\colon L \to 2^C$，它将文字ℓ 映射到包含 ℓ的子句集合 $cl(\ell) = \{c \in C \mid \ell \in c\}$，并定义如下转移函数 δ。示意见图 1：

- 自环 (q_s, σ, q_s) 和 (q_{ns}, σ, q_{ns}) 对于每个 $\sigma \in \Sigma$ 都属于 δ；对于每个状态 x_i^0 和每个 $c \in C$，转移 $(x_i^0, (a_i, c)\ x_i^1) \in \delta$；自环 $(x_i^0, (a_j, c)\ x_i^0)$ 对于 $1 \le j \le i - 1 \in \delta$；转移 $(x_i^0, (a_i, c)\ x_j^0)$ 对于 $1 \le j \le i - 1 \in \delta$；转移 $(x_i^0, (a_j, c)\ q_{ns})$ 对于 $i + 1 \le j \le n + 1 \in \delta$；转移 $(x_i^0, (a_j, c)\ q_{ns})$ 对于 $1 \le j \le n + 1$ 和 $c \in cl(\neg x_i) \in \delta$；对于每个状态 x_i^1 和 $c \in C$，转移 $(x_i^1, (a_i, c)\ q_{ns}) \in \delta$；
-

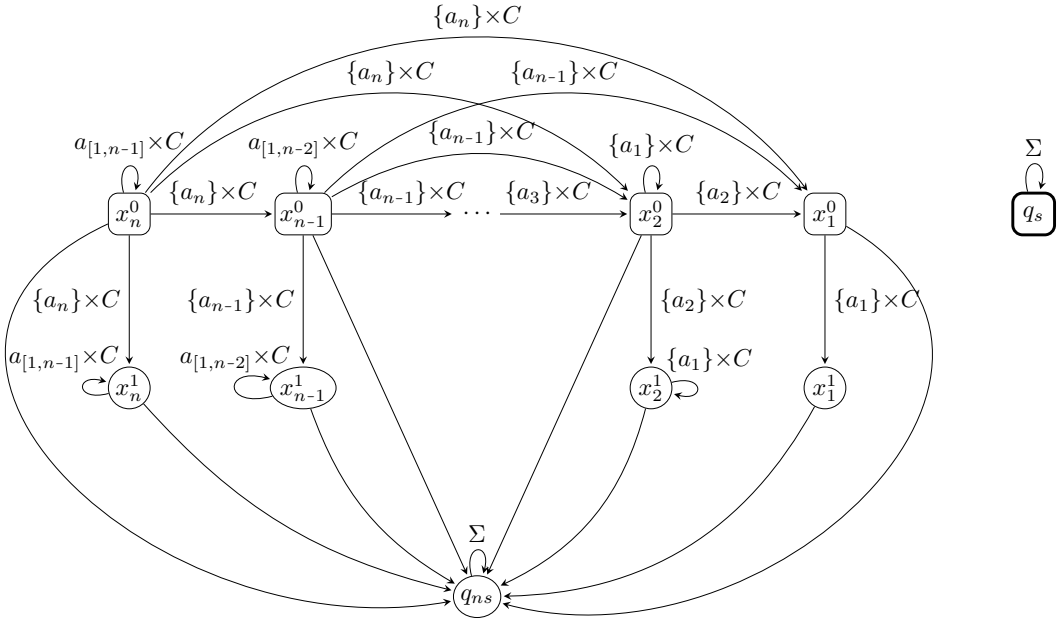


图 1 定理 8 中的非确定有限自动机 A_φ，其中初始状态用方框表示，唯一的秘密状态 q_s 用粗体表示；对于正整数 $i \le j$、$[i,j]=\{i,\ i+1,\ \ldots\ldots,\ j\}$ 以及 $a_{[i,j]}=\{a_r \mid r\in[i,j]\}$。

- 自环 $(x_i^1,\ (a_j,\ c)\ x_i^1)\in\delta$，对于 $1\le j\le i-1$；转移 $(x_i^1,$
- $(a_j,\ c)\ q_{ns})\in\delta$ 对于 $1\le j\le n+1$ 和 $c\in cl(x_i)$。

最后，初始状态集合为 $I=\{q_s\}\cup\{x_1^0,\ \ldots\ldots,\ x_n^0\}$，对应于将 φ 的所有变量都赋值为 0。

现在我们定义语言 $W_\varphi = Z_n\cdot(\{a_{n+1}\}\times C)$，其中 Z_n 是一个基于字母表 $\{a_1、\ a_2、\ \ldots\ldots、\ a_n\}\times C$ 的语言，其递归定义如下：

$$Z_1=\{a_1\}\times C \qquad \text{and} \qquad Z_i = Z_{i-1}\cdot(\{a_i\}\times C)\cdot Z_{i-1},\quad \text{for } 1<i\le n\,.$$

这类字符串称为*Zimin* 词，众所周知，Z_n 中的任意字符串长度为 2^n-1 [40]，并且

任意字符串中第 ℓ 个位置上的符号，Z_n 具有以下形式：$\{a_j\}\times C$，其中 $j-1$ 是 ℓ [41] 的二进制表示中末尾零的个数。 (1)

我们通过一系列断言完成证明。第一个断言表明，沿着 Z_n 中的任意字符串，$\{x_i^0,\ x_i^1 \mid x_i\in X\}$ 中 $\mathcal{A}_\varphi$ 的状态编码了变量的所有可能赋值。

▷ 断言 9. 令 $\mathcal{A}_\varphi^X$ 表示去掉状态 q_s 和 q_{ns} 及其对应转移后的 $\mathcal{A}_\varphi$。对于每个 $w\in Z_n$，读取 w 长度为 $\ell\le 2^n-1$ 的前缀后，$\mathcal{A}_\varphi^X$ 的配置为 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}},\ldots,x_1^{r_1}\}$，其中 $r_nr_{n-1}\cdots r_1$ 表示 ℓ 的二进制表示。

▷ 断言 10. $\mathcal{A}_\varphi$ 的每个配置都包含秘密状态 q_s。

根据断言 9 和 10，并且由于从 q_s 只能到达状态 q_s 本身，而从 q_{ns} 只能到达状态 q_{ns} 本身，我们得到以下观察，该观察描述了 $\mathcal{A}_\varphi$ 沿着 Z_n 中字符串的计算。

▷ 断言 11. 读入 $w\in Z_n$ 的长度为 $\ell\le 2^n-1$ 的前缀后，$\mathcal{A}_\varphi$ 的配置要么是 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}},\ldots, x_1^{r_1}\}\cup\{q_s\}$，要么是 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}},\ldots,x_1^{r_1}\}\cup\{q_s,q_{ns}\}$，其中 $\cdots r_1$ 以二进制表示 ℓ。◁

如果存在满足赋值，那么 $\mathcal{A}_\varphi$ 会到达非秘密状态 q_{ns}。▷ 断言 12. 对于 Z_n 中某个字符串的任意前缀 w，如果 $\mathcal{A}_\varphi$ 读入 w 后的配置满足 φ，那么在读入 Σ 中任何后续符号后，$\mathcal{A}_\varphi$ 的配置都包含 q_{ns}。

我们现在证明，若 φ 可满足，则$\mathcal{A}_\varphi$ 属于 CSO。为此，我们考虑任意一个 w 字符串，其字符来自 Σ，并将 u 记为 w 的最长前缀，且该前缀也是 Z_n中某个字符串的前缀。令 ℓ 为满足以下条件的最小数：其二进制表示 $r_n r_{n-1} \cdots r_1$ 是变量 φ的一个满足赋值。

若 $\ell \leq |u|$，则根据断言 11，$\mathcal{A}_\varphi$ 在读取长度为 $\ell' \leq \ell$ 的 u 任意前缀后，其配置包含非秘密状态 $x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_1^{r_1}$，其中 $r_n r_{n-1} \cdots r_1$ 以二进制表示 ℓ'；并且根据断言 12，w 的第 $(\ell+1)$ 个符号会使$\mathcal{A}_\varphi$ 转移到一个包含 q_{ns}的配置；也就是说，$\mathcal{A}_\varphi$ 属于 CSO。

如果 $\ell > |u|$，则有 $w = u(a_s, c)v$ 对于 $(a_s, c) \in \Sigma$ 和 $v \in \Sigma^*$。由于 φ是可满足的，我们有 $|u| < 2^n - 1$。根据断言 11，Aφ 在读取 u 后的配置包含 $x_n^{r_n}$，…，$x_1^{r_1}$，其中 $r_n r_{n-1} \cdots r_1$ 表示 $|u|$ 的二进制形式。令 r_t 为 $r_n r_{n-1} \cdots r_1$最右侧的零，即 $r_n r_{n-1} \cdots r_1 = r_n r_{n-1} \cdots r_{t+1} 01 \cdots 1$。于是，$|u|+1$ 的二进制形式为 $r_n r_{n-1} \cdots r_{t+1} 10 \cdots 0$，并且根据 (1)，符号 $(a_s, c) \notin \{a_t\} \times C$。然而，如果 $s < t$，则 x_s^1 在 $\{a_s\} \times C$下转移到状态 q_{ns}；而如果 $s > t$，则 x_t^0在 $\{a_s\} \times C$下转移到状态 q_{ns}。在这两种情况下，非秘密状态 q_{ns} 都位于 Aφ的下一个配置中，因此 Aφ 是 CSO。

为了证明如果 φ 不可满足，那么$\mathcal{A}_\varphi$ 不是 CSO，我们使用以下结论。

▷ 结论 13. 如果 φ 不可满足，则存在一个字符串 $w_\varphi \in Z_n$，使得$\mathcal{A}_\varphi$ 读取w_φ 后的配置为$\{x_n^1, x_{n-1}^1, \ldots, x_1^1\} \cup \{q_s\}$。

现在我们证明，如果 φ 不可满足，那么 Aφ 不是 CSO。为此，我们考虑结论 13 中构造的字符串w_φ，并添加一个形如$\{a_{n+1}\} \times C$的符号，将其扩展为来自 $W_\varphi = Z_n \cdot (\{a_{n+1}\} \times C)$ 的字符串。由于 φ 不可满足，因此存在一个子句$c \in C$ 未被将 1 赋给这些变量的赋值满足；也就是说，存在 $c \notin \bigcup_{i=1}^n cl(x_i)$。然后，字符串$w_\varphi(a_{n+1}, c)$将自动机 A$\varphi$ 从配置$\{x_n^1, x_{n-1}^1, \ldots, x_1^1\} \cup \{q_s\}$转移到配置$\{q_s\}$，因此 A$\varphi$ 不是 CSO。◀

现在我们来说明这一构造。

▶ **例 14.** 为简单起见，我们考虑一个 2-CNF 公式

$$\varphi = (x_2 \vee x_2) \wedge (x_1 \vee x_2) \wedge (\neg x_1 \vee x_3) \wedge (\neg x_2 \vee \neg x_3) \wedge (x_3)$$

它包含三个变量 x_1, x_2, x_3 和五个子句 $c_1 = \{x_2, x_2\}$，$c_2 = \{x_1, x_2\}$，$c_3 = \{\neg x_1, x_3\}$,$c_4 = \{\neg x_2, \neg x_3\}$, 和 $c_5 = \{x_3\}$。自动机 Aφ $= (Q, \Sigma, \delta, \{q_s, x_1^0, x_2^0, x_3^0\})$ 如图 2 所示，其中 $Q = \{q_s, q_{ns}\} \cup \{x_1^0, x_1^1, x_2^0, x_2^1, x_3^0, x_3^1\}$，$\Sigma = \Gamma = \{a_1, a_2, a_3, a_4\} \times \{c_1, c_2, c_3, c_4, c_5\}$，而 q_s 是唯一的秘密状态。由于 φ 不可满足，Aφ 不是 CSO；事实上，字符串 $w = (a_1, c_1)(a_2, c_2)(a_1, c_5)(a_3, c_3)(a_1, c_1)(a_2, c_1)(a_1, c_4)(a_4, c_4)$ 会将 Aφ 移至仅由秘密状态构成的配置 $\{q_s\}$。参见图 3，其中展示了 Aφ 的可达配置。

另一方面，如果我们考虑公式 $\varphi' = c_1 \wedge c_2 \wedge c_3 \wedge c_4$，那么 φ' 是可满足的，因此，从 $\mathcal{A}_\varphi$ 中移除所有包含 c_5 的符号下的转移后得到的 NFA $\mathcal{A}_{\varphi'}$ 是 CSO；从图 4 所示的 $\mathcal{A}_{\varphi'}$ 的可达配置中可以看出这一点。

所考虑的问题都是 PSpace 完全的，因此可以在多项式时间内相互归约。然而，这一事实并没有为我们提供关于这些归约的太多信息。尽管 Wu 和 Lafortune [43] 以及 Balun 和 Masopust [6, 7]，曾在文献中讨论过一些特定的归约，但在大多数情况下，它们并不适合用于证明下界。

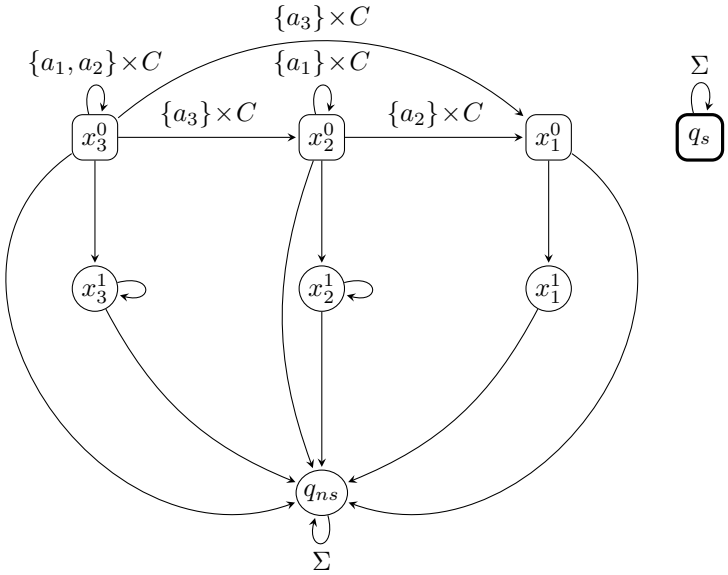


图 2 展示定理 8 的 NFA A_φ；初始状态用方框表示。

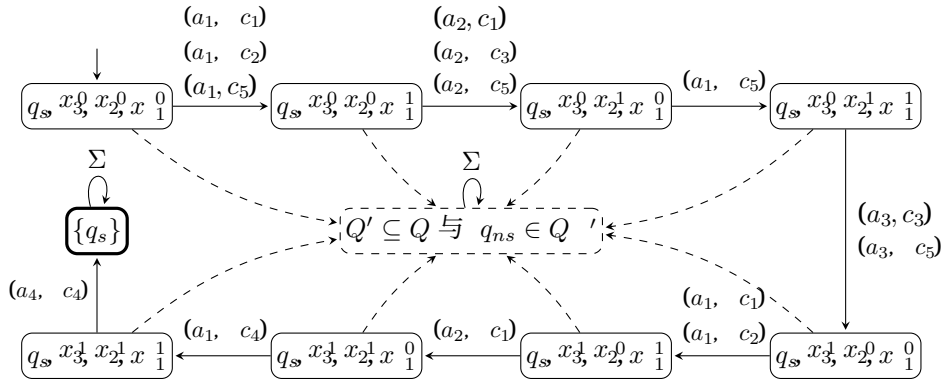


图 3 A_φ 的配置——所有未定义转移均指向虚线标示的中间状态。

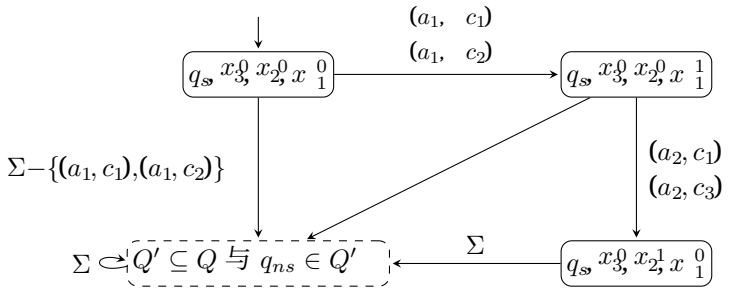


图 4 $A_{\varphi'}$ 的配置。

10 能否加快速度：不透明性验证的条件下界

现在我们讨论其他类型的不透明性。

▶**推论 15.** 除非 *SETH* 不成立，否则不存在一个算法能够判定给定的 n 状态非确定有限自动机是否为 *LBO/ISO/IFO/k-SO/INSO*，且运行时间为 $O^*(2^{n/(2+\varepsilon)})$，对任意 $\varepsilon > 0$。

证明。考虑在定理 8 的证明中构造的、由 NFA A 以及集合 Q_S 和 Q_{NS} 给出的 CSO 实例。那么，A 关于 Q_S ={q_s} 和 $Q_{NS} = Q - Q_S$ 是 CSO，当且仅当 A 关于 $L_S = L_m$(A q_s, q_s) $= \Sigma^*$ 和 $L_{NS} = L_m$(A, {$x_1^0, \ldots, x_n^0$} $Q - \{q_s\}$) 是 LBO。由于 A 中对应于语言 L_S 和 L_{NS} 的部分互不相交，因此可以通过定义接受语言 L_S 和 L_{NS} 的相应状态，将 LBO 实例直接编码到 A 中。因此，LBO 实例与 CSO 实例的规模相同。所以，如果我们能在时间 $O^*(2^{(1-\delta)n})$ 内求解 LBO 实例，也就能在时间 $O^*(2^{(1-\delta)n})$ 内求解 CSO 实例。

由于$\mathcal{A}$ 中唯一的秘密状态 q_s 没有转移到其他状态，因此对于任意 $k \in \mathbb{N} \cup \{\infty\}$，NFA$\mathcal{A}$ 是 CSO，当且仅当$\mathcal{A}$ 是 k-SO。因此，该结果对 k-SO 和 INSO 同样成立。

此外，NFA $\mathcal{A}$ 关于集合 Q_S 和 Q_{NS} 是 CSO，当且仅当$\mathcal{A}$ 关于秘密初始状态 I_S = {q_s} 和非秘密初始状态$I_{NS} = \{x_1^0, \ldots, x_n^0\}$是 ISO。事实上，由于 $L(\mathcal{A}\ I_S) = \Sigma^*$，NFA $\mathcal{A}$ 不是 CSO，当且仅当存在一个字符串 $w \in \Sigma^*$，使$\mathcal{A}$ 从初始配置 I_{NS} 转移到配置 $\emptyset$；这又当且仅当$\mathcal{A}$ 不是 ISO。因此，在 $O^*(2^{(1-\delta)n})$ 时间内求解 ISO，就能在 $O^*(2^{(1-\delta)n})$ 时间内求解 CSO。

最后，如果 $\mathcal{A}$ 的所有状态都是接受状态，那么当且仅当 $\mathcal{A}$ 相对于 I_S ={q_s} 和$I_{NS} = \{x_1^0, \ldots, x_n^0\}$ 是 ISO 时，$\mathcal{A}$相对于 $IQ_S = \{(q_s, q_s)\}$ 和 IQ_{NS} =$I_{NS} \times Q$ 是 IFO；因此，如果能在 $O^*(2^{(1-\delta)n})$ 时间内求解 IFO，就能在 $O^*(2^{(1-\delta)n})$ 时间内求解 ISO。◀

5 指数时间假设下的下界

定理 8 中构造的非确定有限自动机所含符号的数量取决于 SAT 实例中的子句数量。由于标准的符号二进制编码在 SETH 下不起作用，因此定理 8 和推论 15 的结果是否也适用于固定大小的字母表，仍是一个开放问题。

尽管我们没有回答这个问题，但我们给出了 ETH 下二元字母表上的非确定有限自动机的下界。具体而言，我们证明，对于这类具有 n 个状态的非确定有限自动机，不存在能在 $O^*(2^{o(n)})$ 时间内解决所考虑的不透明性问题的算法。我们通过调整 Fernau 和 Krebs [16], 的构造得到了这一结果；他们证明了除非 ETH 不成立，否则不存在能在 $O^*(2^{o(n)})$ 时间内解决二元字母表上具有 n 个状态的非确定有限自动机的普遍性问题的算法。此外，我们还利用了 Cassez 等人 [13]的观察：普遍性可以归约为不透明性。

▶**定理16**。 除非 *ETH* 不成立，否则不存在一种算法，能在时间 $O^*(2^{o(n)})$内判定给定的 n 状态 *NFA*（基于二元字母表）是否为 *CSO*。

证明。 图 $G = (V, E)$ 的 3-着色问题是一个函数 μ: $V \to \{a, b, c\}$。若在 $\mu(u) \neq \mu(v)$ 的情况下 $uv \in E$，则该着色是正常的。 3-着色问题 是指给定一个图 G，判断 G是否存在正常3-着色。

对于一个具有 n 个顶点 $V = \{v_1, v_2, \ldots, v_n\}$ 和 m 条边的图 G，我们构造一个 NFA A =(Q, Σ, δ, I)，其中状态为 $Q = \{s, f\} \cup \{q_1, \ldots, q_n\} \cup \{x_1, \ldots, x_{n-1} \mid x \in \{a, b, c\}\}$，字母表为 $\Sigma = \Gamma = \{a, b, c\}$，初始状态为 $I = \{q_1\}$，秘密状态为 $Q_S = \{s\}$，非秘密状态为 $Q_{NS} = Q - Q_S$。我们按照如下方式定义转移函数 δ。

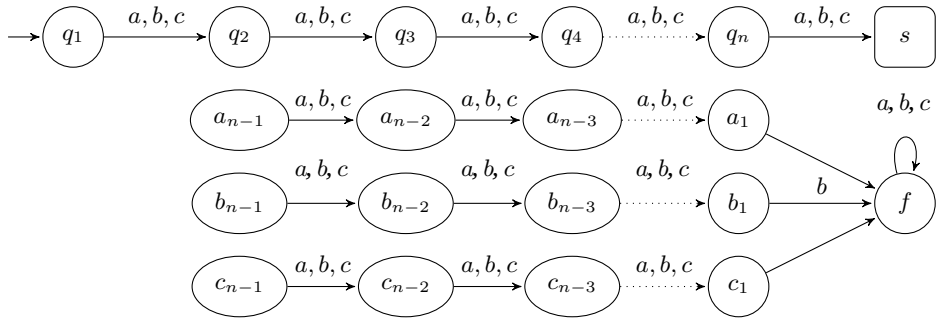


图5 定理16的归约所得的非确定有限自动机 $\mathcal{A}$ 的主体部分，其中去除了与 G 的边相对应的转移。秘密状态为状态 s。

在图5中，并通过为每条满足 $i < j$ 的边 $v_iv_j \in E$ 添加三个转移 (q_i, a, a_{j-i})、(q_i, b, b_{j-i}) 和 (q_i, c, c_{j-i}) 对其进行了进一步扩展。

直观而言，G 的着色被编码为一个长度为 n的字符串 $w = c_1 \cdots c_n$，其中 c_i是顶点 i 的颜色，当且仅当非秘密状态 f 在 w下可达时，$\mathcal{A}$ 相对于$\{s\}$ 和 $Q - \{s\}$ 满足CSO；事实上，秘密状态 s 在任意长度为 n的字符串下均可达。

Fernau 和 Krebs [16] 证明，在 w 下，$\mathcal{A}$ 最终进入状态 f，当且仅当 w 编码了一个非正常着色。因此，如果 w 是 G的一个正常3-着色，那么 w 不会将 $\mathcal{A}$ 转移到状态 f；也就是说，在 w下只能到达秘密状态 s，因此，相对于 $\{s\}$ 和 $Q - \{s\}$，$\mathcal{A}$ 不是 CSO。另一方面，如果 G 不存在正常3-着色，那么每个长度为 n 的字符串都会将 $\mathcal{A}$ 转移到秘密状态 s 和非秘密状态f；也就是说，相对于 $\{s\}$ 和 $Q - \{s\}$，$\mathcal{A}$ 是 CSO。

如果 G 有 n 个顶点和 m 条边，那么 $\mathcal{A}$ 有 $N = 4n - 1$ 个状态和 $M = 12n + 3m - 12$ 次转移。如果存在一个运行时间为 $O^*(2^{o(N)})$ 的算法来判定 CSO，我们就可以在 $O(N + M)$ 时间内将 3-着色问题的实例归约为 CSO 的实例，并在 $O^*(2^{o(N)})$ 时间内求解 CSO。总之，我们可以在 $O(N + M) + O^*(2^{o(N)}) = O^*(2^{o(n)})$ 时间内求解 3-着色问题，这与 ETH 矛盾。◀

与上一节中的讨论类似，为证明其他不透明性概念的下界，自然的做法似乎是将前一个证明中的构造与这些概念之间已有的归约 [6, 7, 43] 结合起来。然而，大多数归约会产生过大的实例，尽管这些实例仍是多项式规模，因此不适用于我们的目的。因此，需要新的归约。

▶**推论17**。 除非 *ETH* 不成立，否则不存在能判定给定的 n 状态非确定有限自动机（基于二元字母表）是否为 *LBO/ISO/IFO/k-SO/INSO* 的算法，其运行时间为 $O^*(2^{o(n)})$。

证明。对于定理16中的非确定有限自动机 A，A 关于 $\{s\}$ 和 $Q - \{s\}$ 是 CSO，当且仅当 A 关于 $L_S = L_m$(A q_1, s) 和 $L_{NS} = L_m$(A $q_1, Q - \{s\}$) 是 LBO [43]。如果我们能在 $O^*(2^{o(n)})$ 时间内求解 LBO，就能在 $O^*(2^{o(n)})$ 时间内求解 3-着色问题。

此外，由于唯一的秘密状态 s 没有出边转移，自动机 $\mathcal{A}$关于 $\{s\}$ 和 $Q - \{s\}$ 是 CSO，当且仅当 $\mathcal{A}$ 关于 $\{s\}$ 和 $Q - \{s\}$ 是 k-SO；这对任意 $k \in \mathbb{N} \cup \{\infty\}$ 均成立。因此，该结果对 k-SO 以及 INSO 均成立。

现在，我们取 NFA $\mathcal{A}$，并添加一份状态 $q_1, q_2, \ldots, q_n$ 的副本，记作 $q'_1, q'_2, \ldots, q'_n$，同时添加所有指向不同于 s 的状态的转移；也就是说，对于每个满足 $p \neq s$ 的转移 (q_i, x, p)，我们都添加 (q'_i, x, p)。我们将状态 q_1 和 q'_1 设为初始状态，并将结果记为$\mathcal{A}'$。那么，NFA $\mathcal{A}$ 关于 $\{s\}$ 和 $Q - \{s\}$ 是 CSO，当且仅当 $\mathcal{A}'$ 是 ISO，且

相对于 $I_S = \{q_1\}$ 和 $I_{NS} = \{q_1'\}$。事实上，如果 A 关于 $\{s\}$ 和 $Q - \{s\}$ 是 CSO，那么对于每个将 A 移动到状态 s 的 w，都存在一个将 A 移动到状态 f 的 w'；A 中的字符串w 和 w' 也是如此′, ，这表明 A′ 关于 $I_S = \{q_1\}$ 和 $I_{NS} = \{q_1'\}$ 是 ISO。另一方面，如果 A 关于 $\{s\}$ 和 $Q - \{s\}$ 不是 CSO，那么存在一个只将 A 移动到状态 s 的 w，因此 A′ 无法从状态 q_1' 读取 w，这表明 A′ 关于 $I_S = \{q_1\}$ 和 $I_{NS} = \{q_1'\}$ 不是 ISO。如果我们能在 $O^*(2^{o(n)})$ 时间内解决 ISO 问题，那么通过将 3-着色问题归约为 ISO 并在时间$O^*(2^{o(N+n)}) = O^*(2^{o(n)})$ 内解决 ISO，我们就能在 $O^*(2^{o(n)})$ 时间内解决 3- 着色问题，其中 $N = 4n - 1$。

此外，如果我们将状态 s 和 f 设为接受状态，那么，关于 $I_S =$，$\mathcal{A}'$ 是 ISO，当且仅当 $\mathcal{A}'$ 关于 $IQ_S = \{(q_1, s)\}$ 和 $IQ_{NS} = \{(q_1', f)\}$ 是 IFO。因此，如果我们能在 $O^*(2^{o(n)})$ 时间内求解 IFO，就能在 $O^*(2^{o(n)})$ 时间内求解 3-着色问题。◀

6 讨论与结论

我们证明了，如果强指数时间假设成立，那么对于任意 $c > 2$，都不存在能在 $O^*(2^{n/c})$ 时间内判定各种类型不透明性的算法。因此，现有算法无法得到显著改进。

更准确地说，结果表明，不存在能在 $O^*(\sqrt{2}^n) = O^*(1.414213562^n)$ 时间内判定各种类型不透明性的算法。不过，这些结果并未排除存在能在 $O^*(1.5^n)$ 时间内判定不透明性的算法。此类算法是否存在，或者现有下界能否进一步加强，仍是一个悬而未决的问题。

定理 8 证明中使用的构造可用于改进判定非确定有限自动机全性的条件下界。非确定有限自动机的普遍性问题是：给定一个非确定有限自动机，判断该自动机是否接受其字母表上的所有字符串。若将定理 8 中非确定有限自动机 $\mathcal{A}$ 的唯一秘密状态 q_s 设为非接受状态，并将其余所有状态设为接受状态，则得到的非确定有限自动机是全性的，当且仅当自动机 $\mathcal{A}$ 相对于 $\{q_s\}$ 和 $Q - \{q_s\}$是 CSO。因此我们得到以下推论，改进了 Fernau 和 Krebs [16] 的结果。

▶**推论 18.** 除非 *SETH* 不成立，否则不存在能判定给定的 n-状态非确定有限自动机是否具有全性的算法，其运行时间为 $O^*(2^{n/(2+\varepsilon)})$，对任意 $\varepsilon > 0$ 均如此。◀

因此，我们立即得到以下结果。

▶ **推论 19.** 给定两个非确定有限自动机 A_1 和 A_2，它们分别有 n_1 和 n_2 个状态，令 $n = \max(n_1, n_2)$。除非 *SETH* 不成立，否则不存在能在 $O^*(2^{n/(2+\varepsilon)})$ 时间内判定 $L_m(A_1) \subseteq L_m(A_2)$ 是否成立的算法；也不存在能在$O^*(2^{n/(2+\varepsilon)})$时间内判定 $L_m(A_1) = L_m(A_2)$ 是否成立的算法，对任意 $\varepsilon > 0$ 均如此。◀

我们将定理 8 是否也适用于字母表大小固定的非确定有限自动机这一问题留待解决。尽管我们没有回答这个问题，但我们证明了，ETH 意味着不存在能够在二元字母表上判定各种类型不透明性的次指数时间算法。

查看表 1，读者可能会注意到，在不对非秘密对的形式施加任何限制时，IFO 验证的下界与上界之间存在相当大的差距。如何改进上界，或者（有条件地）证明这种改进不可能实现，是一个颇具挑战性的开放问题。

值得注意的是，推论 15 的证明中的构造生成了一个特殊情形的实例，其中非秘密对的形式为 $IQ_{NS} = I_{NS} \times F_{NS}$；因此，根据强指数时间假设，该特殊情形的界是紧的。

参考文献

1 Rajeev Alur、Pavol Černý 和 Steve Zdancewic。《在细化过程中保持保密性》。载于 Michele Bugliesi、Bart Preneel、Vladimiro Sassone 和 Ingo Wegener（编），《自动机、语言与程序设计国际研讨会》，ICALP 2006，意大利威尼斯，2006年7月10日至14日，计算机科学讲义第4052卷，第107–118页。Springer，2006年。 doi:10.1007/11787006_10。2 Peter R. J. Asveld 和 Anton Nijholt。《若干上下文无关语言子类的包含问题》。理论计算机科学，230(1-2):247–256，2000年。 doi:10.1016/S0304-3975(99)00113-9。3 Éric Badouel、Marek A. Bednarczyk、Andrzej M. Borzyszkowski、Benoît Caillaud 和 Philippe Darondeau。《并发秘密》。离散事件动态系统，17(4):425–446，2007年。doi:10.1007/s10626-007-0020-5。4 Éric Badouel、Marek A. Bednarczyk、Andrzej M. Borzyszkowski、Benoît Caillaud 和 Philippe Darondeau。《并发秘密》。离散事件动态系统，17(4):425–446，2007年。doi:10.1007/s10626-007-0020-5。5 Jiří Balun 和 Tomáš Masopust。《离散事件系统的不透明性验证》。载于 IFAC 世界大会，德国柏林，2020年7月11日至17日，第2105–2110页，2020年。 doi:10.1016/j.ifacol.2020.12.2524。6 Jiří Balun 和 Tomáš Masopust。《比较离散事件系统的不透明性概念》。离散事件动态系统，31(4):553–582，2021年。 doi:10.1007/s10626-021-00344-2。7 Jiří Balun 和 Tomáš Masopust。《不透明性概念之间的转换》。载于 IEEE 系统、人类与控制论国际会议，SMC 2022，捷克共和国布拉格，2022年10月9日至12日，第3012–3017页。IEEE，2022年。 doi:10.1109/SMC53654.2022.9945608。8 Eike Best、Philippe Darondeau 和 Roberto Gorrieri。《无界佩特里网上的非干扰性是否可判定》。载于 Konstantinos Chatzikokolakis 和 Véronique Cortier（编），《并发中的安全问题国际研讨会》，SecCo 2010，法国巴黎，2010年8月30日，EPTCS第51卷，第16–33页，2010年。 doi:10.4204/EPTCS.51.2。9 Jeremy W. Bryans、Maciej Koutny、Laurent Mazaré 和 Peter Y. A. Ryan。《将不透明性推广至转移系统》。国际信息安全期刊，7(6):421–435，2008年。doi:10.1007/s10207-008-0058-x。10 Jeremy W. Bryans、Maciej Koutny 和 Peter Y. A. Ryan。《使用带静默动作的佩特里网对动态不透明性建模》。载于 Theodosis Dimitrakos 和 Fabio Martinelli（编），《安全与信任的形式化方面：第二届 IFIP TC1 WG1.7 安全与信任形式化方面研讨会（FAST）》，第18届 IFIP 世界计算机大会活动之一，法国图卢兹，2004年8月22日至27日，IFIP第173卷，第159–172页。Springer，2004年。 doi:10.1007/0-387-24098-5_12。11 Jeremy W. Bryans、Maciej Koutny 和 Peter Y.A. Ryan。《使用佩特里网对不透明性建模》。理论计算机科学电子札记，121:101–115，2005年。第二届佩特里网及其他计算模型安全问题国际研讨会（WISP 2004）论文集。 doi:10.1016/j.entcs.2004.10.010。12 Nadia Busi 和 Roberto Gorrieri。《基本网与迹网中的结构性非干扰》。计算机科学数学结构，19(6):1065–1090，2009年。 doi:10.1017/S0960129509990120。13 Franck Cassez、Jérémy Dubreil 和 Hervé Marchand。《使用静态与动态掩码综合不透明系统》。系统设计形式化方法，40(1):88–115，2012年。 doi:10.1007/s10703-012-0141-9。14 Roger Dingledine、Nick Mathewson 和 Paul Syverson。《P2P 匿名系统中的声誉》。载于点对点系统经济学研讨会，P2P Econ 2003，美国加利福尼亚州伯克利，2003年6月5日至6日，2003年。

15 Jeremy Dubreil、Philippe Darondeau 和 Herve Marchand。不透明性强制控制综合。载于离散事件系统国际研讨会（WODES 2008），瑞典哥德堡，2008年5月28—30日，第28–35页。IEEE，2008年。 doi:10.1109/WODES.2008.4605918。16 Henning Fernau 和 Andreas Krebs。有限自动机与指数时间假设相关问题。Algorithms，10(1):24，2017年。 doi:10.3390/a10010024。17 R. Focardi 和 R. Gorrieri。CCS中基于轨迹的安全属性分类。载于计算机安全基础研讨会第七届，美国新罕布什尔州弗兰科尼亚，1994年6月14—16日，第126–136页，1994年。 doi:10.1109/CSFW.1994.315941。18 Rômulo Meira Góes、Blake C. Rawlings、Nicholas Recker、Gregory Willett 和 Stéphane Lafortune。通过混淆演示室内位置隐私强制实施。IFAC-PapersOnLine，51(7):145–151，2018年。 doi:10.1016/j.ifacol.2018.06.293。19 N. B. Hadj-Alouane、S. Lafrance、Feng Lin、J. Mullins 和 M. M. Yeddes。多级安全中的非传递非干扰性验证。IEEE Transactions on Systems, Man, and Cybernetics, Part B，35(5):948–958，2005年。 doi:10.1109/TSMCB.2005.847749。20 Christoforos N. Hadjicostis。《离散事件系统中的估计与推断：一种基于有限自动机的模型方法》。通信与控制工程。Springer Cham，2020年。 doi:10.1007/978-3-030-30821-6。21 Timon Hertli。更快、更简单的3-SAT——PPSZ的一般唯一可满足性界限。SIAM Journal on Computing，43(2):718–729，2014年。 doi:10.1137/120868177。22 John E. Hopcroft、Rajeev Motwani 和 Jeffrey D. Ullman。《自动机理论、语言与计算导论》。培生国际版。Addison-Wesley，第三版，2007年。23 Russell Impagliazzo 和 Ramamohan Paturi。论k-SAT的复杂性。Journal of Computer and System Sciences，62(2):367–375，2001年。 doi:10.1006/jcss.2000.1727。24 Romain Jacob、Jean-Jacques Lesage 和 Jean-Marc Faure。离散事件系统不透明性综述：模型、验证与量化。Annual Reviews in Control，41:135–146，2016年。doi:10.1016/j.arcontrol.2016.04.015。25 Feng Lin。离散事件系统的可诊断性及其应用。Discrete Event Dynamic Systems，4(2):197–212，1994年。 doi:10.1007/BF01441211。26 Feng Lin。离散事件系统的不透明性及其应用。Automatica，47(3):496–503，2011年。 doi:10.1016/j.automatica.2011.01.002。27 Feng Lin 和 Walter Murray Wonham。论离散事件系统的可观测性。Information Sciences，44(3):173–198，1988年。 doi:10.1016/0020-0255(88)90001-1。28 Tomáš Masopust。离散事件系统中可检测性判定的复杂性。Automatica，93:257–261，2018年。 doi:10.1016/j.automatica.2018.03.077。29 Tomáš Masopust 和 Xiang Yin。模块化离散事件系统中可检测性、不透明性和A-可诊断性的复杂性。Automatica，101:290–295，2019年。 doi:10.1016/j.automatica.2018.12.019。30 Laurent Mazaré。非原子密钥下不透明性的可判定性。载于Theodosis Dimitrakos和Fabio Martinelli编，《安全与信任的形式化方面：第二届IFIP TC1 WG1.7安全与信任形式化方面研讨会（FAST），第18届IFIP世界计算机大会专题活动》，法国图卢兹，2004年8月22—27日，IFIP第173卷，第71–84页。Springer，2004年。 doi:10.1007/0-387-24098-5_6。31 Ramamohan Paturi、Pavel Pudlák、Michael E. Saks 和 Francis Zane。一种改进的k-SAT指数时间算法。Journal of the ACM，52(3):337–364，2005年。 doi:10.1145/1066100.1066101。32 Michael K. Reiter 和 Aviel D. Rubin。Crowds：Web事务匿名性。ACM Transactions on Information and System Security，1(1):66–92，1998年。 doi:10.1145/290163.290168。33 Andrei Sabelfeld 和 Andrew C. Myers。基于语言的信息流安全。IEEE Journal on Selected Areas in Communications，21(1):5–19，2003年。 doi:10.1109/JSAC.2002.806121。

34 Anooshiravan Saboori。离散事件系统中基于状态的不透明性概念的验证与强制实施。博士论文，伊利诺伊大学厄巴纳-香槟分校，2011年。35 Anooshiravan Saboori 和 Christoforos N. Hadjicostis。无限步不透明性的验证及复杂性考量。IEEE Transactions on Automatic Control，57(5):1265–1269，2012年。 doi:10.1109/TAC.2011.2173774。36 Anooshiravan Saboori 和 Christoforos N. Hadjicostis。离散事件系统安全应用中初始状态不透明性的验证。Information Sciences，246:115–132，2013年。doi:10.1016/j.ins.2013.05.033。37 Meera Sampath、Raja Sengupta、Stéphane Lafortune、Kasim Sinnamohideen 和 Demosthenis Teneketzis。离散事件系统的可诊断性。IEEE Transactions on Automatic Control，40(9):1555–1575，1995年。 doi:10.1109/9.412626。38 Steve A. Schneider 和 Abraham Sidiropoulos。CSP与匿名性。载于 Elisa Bertino、Helmut Kurth、Giancarlo Martella和Emilio Montolivo编，欧洲计算机安全研究研讨会（ESORICS 1996），意大利罗马，1996年9月25—27日，计算机科学讲义第1146卷，第198–218页。Springer，1996年。 doi:10.1007/3-540-61770-1_38。39 Shaolong Shu、Feng Lin 和 Hao Ying。离散事件系统的可检测性。IEEE Transactions on Automatic Control，52(12):2356–2359，2007年。 doi:10.1109/TAC.2007.910713。40 N. J. A. Sloan。在线整数数列百科全书（OEIS），2023年。A123121。URL:https://oeis.org/A123121。41 N. J. A. Sloan。在线整数数列百科全书（OEIS），2023年。A001511。URL:https://oeis.org/A001511。42 Andrew Wintenberg、Matthew Blischke、Stéphane Lafortune 和 Necmiye Ozay。面向安全性与效用的动态混淆框架。载于ACM/IEEE网络物理系统国际会议（ICCPS 2022），意大利米兰，2022年5月4—6日，第236–246页。IEEE，2022年。 doi:10.1109/ICCPS54341.2022.00028。43 Yi-Chin Wu 和 Stéphane Lafortune。集中式与协调式架构中相关不透明性概念的比较分析。Discrete Event Dynamic Systems，23(3):307–339，2013年。 doi:10.1007/s10626-012-0145-z。44 Yi-Chin Wu、Vasumathi Raman、Blake C. Rawlings、Stéphane Lafortune 和 Sanjit A. Seshia。保障隐私与效用的混淆策略综合。Journal of Automated Reasoning，60(1):107–131，2018年。 doi:10.1007/s10817-017-9420-x。45 Xiang Yin 和 Stéphane Lafortune。使用双向观察器验证无限步和K步不透明性的新方法。Automatica，80:162–171，2017年。 doi:10.1016/j.automatica.2017.02.037。46 Aris Zakinthinos 和 E. Stewart Lee。安全属性的一般理论。载于IEEE安全与隐私研讨会，美国加利福尼亚州奥克兰，1997年5月4—7日，第94–102页。IEEE计算机学会，1997年。 doi:10.1109/SECPRI.1997.601322。

证明的陈述

▷ 断言 9。令$\mathcal{A}_\varphi^X$ 表示去掉状态 q_s 和 q_{ns} 以及对应转移后的$\mathcal{A}\varphi$ 。对于每个 $w \in Z_n$，读取长度为 $\ell \leq 2^n - 1$的 w 前缀后，$\mathcal{A}_\varphi^X$的配置为$\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_1^{r_1}\}$，其中 $r_n r_{n-1} \cdots r_1$ 以二进制表示 ℓ。

证明。对 ℓ 进行归纳。A_φ^X 的初始配置为 $\{x_n^0, x_{n-1}^0, \ldots, x_1^0\}$，其中 $00\cdots0$ 表示二进制形式的 $\ell = 0$。假设 A_φ^X 在读取长度为 $\ell < 2^n - 1$ 的 $w \in Z_n$ 前缀后，其配置为 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_1^{r_1}\}$，其中 $r_n r_{n-1}\cdots r_1$ 表示二进制形式的 ℓ。令 r_t 为 $r_n r_{n-1}\cdots r_1$ 中最右边的零；即 $r_n r_{n-1}\cdots r_1 = r_n r_{n-1}\cdots r_{t+1}01\cdots1$。于是，$\ell+1$ 的二进制表示为 $r_n r_{n-1}\cdots r_{t+1}10\cdots0$，并且由于它有 $t-1$ 个末尾零，根据 (1)， w 的第 $(\ell+1)$ 个符号的形式为 $\{a_t\}\times C$。接下来只需证明，每个 $(a_t, c) \in \{a_t\} \times C$ 都会使 A_φ^X 从配置 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_{t+1}^{r_{t+1}}, x_t^0, x_{t-1}^1, \ldots, x_1^1\}$ 转移到】【。

配置 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_{t+1}^{r_{t+1}}, x_t^1, x_{t-1}^0, \ldots, x_1^0\}$。根据 A_φ^X 的定义，在状态 $x_{t-1}^1, \ldots, x_1^1$ 中，(a_t, c) 下的转移未定义；在状态 $x_n^{r_n}, \ldots, x_{t+1}^{r_{t+1}}$ 中，该转移是一个自环；而它会使 A_φ^X 从状态 x_t^0 转移到状态 x_t^1 和 $x_{t-1}^0, \ldots, x_1^0$。因此，自动机 A_φ^X 转移到配置 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_{t+1}^{r_{t+1}}, x_t^1, x_{t-1}^0, \ldots, x_1^0\}$，其中 $r_n r_{n-1} \cdots r_{t+1} 10 \cdots 0$ 表示 $\ell + 1$ 的二进制形式。◁

▷ 主张 10. A 的每个配置都包含秘密状态 q_s。

证明。 秘密状态 q_s 是以下自动机的初始状态$\mathcal{A}_\varphi$ 并且由于它在 Σ的所有符号下都有自环，因此它会出现在 $\mathcal{A}_\varphi$的每个配置中。◁

我们现在证明，如果存在一个满足赋值，那么非秘密状态 q_{ns} 就可以通过 $\mathcal{A}_\varphi$ 到达。为此，对于一个变量 x 和 $r \in \{0,1\}$，我们定义函数

$$\mathrm{LIT}(x^r) = \begin{cases} \neg x & \text{if } r = 0 \\ x & \text{if } r = 1 \end{cases}$$

▷ 声明 12. 对于 Z_n 中字符串的每个前缀 w，如果 $\mathcal{A}_\varphi$ 在读取 w 后的配置满足 φ，那么在读取 Σ 中的任意后续符号后，$\mathcal{A}_\varphi$ 的配置包含 q_{ns} .

证明。根据论断 11，读取 w 后，A_φ 的配置为 $\{x_n^{r_n}, x_{n-1}^{r_{n-1}}, \ldots, x_1^{r_1}\} \cup Y$，其中 $Y \in \{\{q_s\}, \{q_s, q_{ns}\}\}$。由于赋值 $r_n r_{n-1} \cdots r_1$ 满足 φ，对于每个 $c \in C$，都存在 i 使得 $\mathrm{lit}(x_i^{r_i})$ 满足 c，因此对于所有 $j = 1, \ldots, n+1$，在 (a_j, c) 下都存在从 $x_i^{r_i}$ 到 q_{ns} 的转移。因此，读取来自 Σ 的任意符号后，A_φ 的配置都包含 q_{ns}。◁

▷ 声明 13：如果 φ 不可满足，则存在一个字符串 $w_\varphi \in Z_n$，使得$\mathcal{A}_\varphi$ 读取 w_φ 后，其配置为 $\{x_n^1, x_{n-1}^1, \ldots, x_1^1\} \cup \{q_s\}$。

证明。 我们构造所需字符串的一列前缀 $\varepsilon = w_0, w_1, \ldots, w_{2^n-1} = w_\varphi$，使得对于 $\ell = 0, \ldots, 2^n - 1$，$\mathcal{A}_\varphi$ 读取 w_ℓ 后，其配置为 $\{x_n^{r_n}, \ldots, x_1^{r_1}\} \cup \{q_s\}$，并且 $r_n \cdots r_1$ 是 ℓ 的二进制表示。

我们对 ℓ 进行归纳。由于 A_φ 的初始配置为 $\{x_n^0, x_{n-1}^0, \ldots, x_1^0\} \cup \{q_s\}$，该断言对 $\ell = 0$ 成立。现在，假设该断言对 w_ℓ 成立，其中 $\ell < 2^n - 1$，并将 A_φ 读取 w_ℓ 后的配置记为 $\{x_n^{r_n}, \ldots, x_1^{r_1}\} \cup \{q_s\}$。我们证明存在 $(a, c) \in \Sigma$，使得 $w_{\ell+1} = w_\ell(a, c)$。令 r_t 为 $r_n \cdots r_1$ 中最右侧的 0，并取 $a = a_t$。那么，对于 c 的任意选择，字符串 $w_{\ell+1}$ 根据 (1) 是 Z_n 中某个字符串的前缀；并且根据声明 9， A_φ 读取 $w_{\ell+1}$ 后的配置为 $\{x_n^{r'_n}, \ldots, x_1^{r'_1}\} \cup Y$，其中 $Y \in \{\{q_s\}, \{q_s, q_{ns}\}\}$，且 $r'_n \cdots r'_1$ 表示 $\ell + 1$ 的二进制形式。由于 φ 不可满足，存在一个子句 c'，对任意 $i = 1, \ldots, n$，它都不被 $\mathrm{lit}(x_i^{r_i})$ 满足。取 $c = c'$ 即可得到所需符号。事实上，在 (a_t, c') 下，从 $x_i^{r_i}$ 出发的转移对于 $i = 1, \ldots, t-1$ 未定义，对于 $i = t+1, \ldots, n$ 是自环，并将 x_t^0 转移到 $\{x_t^1\} \cup \{x_j^0 \mid 1 \le j \le t-1\}$，见图 1；因此，$Y = \{q_s\}$。◁