Electronic Colloquium on Computational Complexity, Report No. 206 (2023)

Hardness of Range Avoidance and Remote Point for Restricted Circuits via Cryptography

Yilei Chen
Tsinghua University
chenyilei@mail.tsinghua.edu.cn

Jiatu Li
MIT
jiatuli@mit.edu

December 9, 2023

Abstract

A recent line of research has introduced a systematic approach to explore the complexity of explicit construction problems through the use of meta problems, namely, the *range avoidance problem* (abbrev. Avoid) and the *remote point problem* (abbrev. RPP). The upper and lower bounds for these meta problems provide a unified perspective on the complexity of specific explicit construction problems that were previously studied independently. An interesting question largely unaddressed by previous works is whether Avoid and RPP are hard for *simple* circuits such as low-depth circuits.

In this paper, we demonstrate, under plausible cryptographic assumptions, that both the range avoidance problem and the remote point problem cannot be efficiently solved by nondeterministic search algorithms, even when the input circuits are as simple as constant-depth circuits. This extends a hardness result established by Ilango, Li, and Williams (STOC'23) against deterministic algorithms employing witness encryption for NP, where the inputs to Avoid are general Boolean circuits.

Our primary technical contribution is a novel construction of *witness encryption* inspired by *public-key encryption* for certain promise language in NP that is unlikely to be NP-complete. We introduce a generic approach to transform a public-key encryption scheme with particular properties into a witness encryption scheme for a promise language related to the initial public-key encryption scheme. Based on this translation and variants of standard lattice-based or coding-based PKE schemes, we obtain, under plausible assumption, a provably secure witness encryption scheme for some promise language in $\mathsf{NP} \setminus \mathsf{coNP}_{/\mathsf{poly}}$. Additionally, we show that our constructions of witness encryption are plausibly secure against *nondeterministic adversaries* under a generalized notion of security in the spirit of Rudich's super-bits (RANDOM'97), which is crucial for demonstrating the hardness of Avoid and RPP against nondeterministic algorithms.

ISSN 1433-8092

Contents

1 Introduction

Proving explicit lower bounds against concrete computation models is a central problem in complexity theory. While exponential lower bounds have been shown for weak models such as AC^0 circuits [Ajt83; FSS84; Yao85; Hås89] or $\mathsf{AC}^0[p]$ circuits for prime p [Raz87; Smo87], it remains open to construct an explicit function, say in NP, that requires general circuits of size $10n$ [FGHK16; LY22], or to construct a function in NEXP that cannot be computed by polynomial-size general circuits. This stands in sharp contrast to the fact that a random Boolean function requires $2^{0.1n}$ size to compute with high probability [Sha49].

The study of circuit lower bound is not the only example where a random object enjoys certain properties with high probability, yet the *explicit construction* of such objects is unknown. For instance, it is not clear how to construct Ramsey graphs (i.e. graphs with neither large cliques nor large independent sets, see [Erd59]) or rigid matrices (i.e. matrices far in Hamming distance from every low-rank matrix, see [Val77]) in deterministic polynomial time. This motivates a systematic investigation of the computational complexity of explicit construction problems.

1.1 Background: Range Avoidance and Remote Point Problem

A recent line of works [KKMP21; Kor21; RSW22; GLW22; Kor22; CHLR23; ILW23; GGNS23] established a promising paradigm towards understanding the complexity of explicit construction problems via *meta-problems*, such as the *range avoidance problem*.

Definition 1.1. The *range avoidance problem*, denoted by Avoid, is the total search problem that given a circuit $C : \{0,1\}^n \to \{0,1\}^m$ for $m > n$, outputs any $y \in \{0,1\}^m \setminus \mathsf{Range}(C)$, where $\mathsf{Range}(C) := \{C(x) \mid x \in \{0,1\}^n\}$.

The range avoidance problem is a typical explicit construction problem in the sense that a random string is a correct answer with probability $1 - 2^{n-m} \geq 1/2$, whereas there is no obvious deterministic polynomial-time algorithm that solves it. Moreover, Korten [Kor21] presented simple reductions from many explicit construction problems, including circuit lower bounds, Ramsey graphs, and rigid matrices, to the range avoidance problem. This established the central position of the range avoidance problem in the study of explicit construction problems.

- On the algorithmic side, a deterministic algorithm (e.g. FP or $\mathsf{FP}^{\mathsf{NP}}$ algorithm) for the range avoidance problem implies deterministic solutions to tons of concrete explicit construction problems. For instance, Korten [Kor21] proved that $\mathsf{Avoid} \in \mathsf{FP}^{\mathsf{NP}}$ if and only if E^{NP} requires Boolean circuits of size $2^{\Omega(n)}$, which (assuming circuit lower bounds) provides a systematic approach to solve the explicit construction problems with an NP oracle.
- On the hardness side, the intractability of Avoid can be interpreted as a barrier to achieving unconditional solutions to concrete explicit construction problems. For instance, Ilango, Li, and Williams [ILW23] proved that $\mathsf{Avoid} \notin \mathsf{FP}$ assuming plausible cryptographic assumptions, which suggests that one cannot hope to design a polynomial-time explicit construction algorithm using techniques that are general enough to solve the range avoidance problem.

Range avoidance for restricted circuits. Following the paradigm in circuit lower bounds, it is natural to consider a variant of the range avoidance problem where each output bit of the circuit C is in a *restricted circuit class* $\mathscr{C}$. This problem, denoted by $\mathscr{C}$-Avoid, was formally introduced by Ren, Santhanam, and Wang [RSW22], who proved that NC^1-Avoid can be reduced to NC^0-Avoid using randomized encoding [AIK06]. Based on this result, subsequent papers [GLW22; GGNS23]

proved that several natural explicit construction problems, say finding rigid matrices and near-optimal binary linear codes, can be reduced to NC^0-Avoid. This shows that the range avoidance problem can already be very useful for weak circuit classes such as NC^0.

Another reason to study range avoidance for restricted circuits is that we can obtain *unconditional upper bounds* using the techniques for proving unconditional circuit lower bounds. Ren, Santhanam, and Wang [RSW22] proposed an approach to solve $\mathscr{C}$-Avoid in $\mathsf{FP}^{\mathsf{NP}}$ by generalizing Williams' algorithmic approach in circuit lower bounds (see, e.g., [Wil13; Wil18]). A subsequent work [CHLR23] improved the framework and proved that ACC^0-Avoid with quasi-polynomial stretch can be solved in $\mathsf{FP}^{\mathsf{NP}}$, which derives the best known almost-everywhere lower bound against ACC^0 [CLW20] as an easy corollary.

Example 1.2. Upper bounds for Avoid can be considered as a natural generalization of circuit lower bounds. By a folklore view of circuit lower bounds, an almost-everywhere lower bound for E^{NP} against $\mathscr{C}$ circuits is equivalent to an $\mathsf{FP}^{\mathsf{NP}}$ algorithm for the problem $\mathscr{C}$-Hard: Given 1^N, generate any string of length N that is not the truth-table of any small $\mathscr{C}$-circuit. For a good circuit class $\mathscr{C}$, an $\mathsf{FP}^{\mathsf{NP}}$ algorithm for $\mathscr{C}$-Avoid implies an $\mathsf{FP}^{\mathsf{NP}}$ algorithm for $\mathscr{C}$-Hard by fixing the input circuit of $\mathscr{C}$-Avoid to be the truth-table generator for $\mathscr{C}$ circuit, which takes a $\mathscr{C}$-circuit as its input and outputs its truth-table (see, e.g., [Kor21; RSW22; CHLR23]).

Remote point problem. An important variant of the range avoidance problem is the *remote point problem*, formally defined as follows.

Definition 1.3. The *remote point problem*, denoted by RPP_ε, is the search problem that given a circuit $C : \{0,1\}^n \to \{0,1\}^m$, outputs any $y \in \{0,1\}^m$ that is ε-far (in Hamming distance) from every string in $\mathsf{Range}(C)$.

By Chernoff bound, we know that when $\varepsilon \leq 1/2 - c\sqrt{n/m}$ for some constant c, most of the strings $y \in \{0,1\}^m$ are correct answers. Similar to Avoid, we can define $\mathscr{C}$-RPP as the special case where the input circuit is a $\mathscr{C}$ circuit. Moreover, by fixing the input circuit to be the truth-table generator in Example 1.2, we can show that a deterministic algorithm for $\mathscr{C}$-RPP implies an almost-everywhere average-case circuit lower bound against $\mathscr{C}$ circuits (also see [CHLR23]).

The special case XOR-Remote (i.e. the circuit is a $\mathsf{GF}(2)$-linear function) has been studied by Alon, Panigrahy, and Yekhanin [APY09] as an intermediate step towards the construction of rigid matrices. They designed a non-trivial algorithm for very weak parameters. Arvind and Srinivasan [AS10] proved that XOR-Remote $\in$ FP implies "help function lower bounds", a generalization of circuit lower bounds. The result of [CHLR23] on Avoid also generalizes to Remote. In particular, they proved that ACC^0-Remote with quasi-polynomial stretch and $\varepsilon = 1/2 - 1/\mathsf{quasi\text{-}poly}(n)$ can be solved in $\mathsf{FP}^{\mathsf{NP}}$, which implies the best known almost-everywhere average-case ACC^0 lower bound.

1.2 Complexity of Avoid and RPP

As we have discussed above, existing work [Kor21; CHLR23; ILW23] suggests, under plausible assumptions, that the complexity of Avoid is between FP and $\mathsf{FP}^{\mathsf{NP}}$. An immediate next step, as noted in [RSW22], is to obtain a finer complexity-theoretic characterization of them based on (possibly stronger) plausible assumptions. In particular, we want to know whether Avoid admits *nondeterministic search algorithms*, or SearchNP algorithm[1], which outputs a correct answer to the search problem on every accepting computation paths.

[1]Note that some authors call this class FNP, while people also use FNP to denote a similar but different class, see Section 2.4 for our definitions and clarification.

Problem 1.4 (Ren, Santhanam, and Wang [RSW22])**.** Prove $\mathsf{Avoid} \in \mathsf{SearchNP}$ or $\mathsf{Avoid} \notin \mathsf{SearchNP}$ under plausible assumptions.

An intriguing aspect of the problem is that there is no clear intuition on what the ground truth should be [RSW22; ILW23]. On the one hand, for some specific explicit construction problems such as rigid matrix [BHPT20] and ACC^0 circuit lower bounds [Wil14; Wil18; CR22], we have unconditional nondeterministic search algorithms. On the other hand, however, existing algorithmic results for range avoidance [Kor21; Kor22; RSW22; CHLR23] in $\mathsf{FP}^{\mathsf{NP}}$ rely crucially on the adaptive accesses to the NP oracle.

To go a step further, we may also consider the complexity of $\mathscr{C}$-Avoid and $\mathscr{C}$-RPP for restricted circuit classes $\mathscr{C}$.

Problem 1.5. Prove complexity upper bounds and lower bounds of $\mathscr{C}$-Avoid or $\mathscr{C}$-RPP for restricted circuit classes $\mathscr{C}$ (e.g., $\mathscr{C} =$ depth-2 ACC^0) with respect to FP and $\mathsf{SearchNP}$ under plausible assumptions.

Proof complexity generators. The hardness of range avoidance against nondeterministic algorithms is closely related to the concept of *proof complexity generators* (see [Kra19; Kra22] and the references therein) in proof complexity, as observed in [RSW22].

Let P be a propositional proof system. A proof complexity generator is a family of functions $g = \{g_n : \{0,1\}^n \to \{0,1\}^m\}_{n\in\mathbb{N}}$ computable by polynomial-sized circuits, where $m > n$, such that the statement $y \notin \mathsf{Range}(g_n)$, when encoded as a Boolean formula with n variables, has no $\mathsf{poly}(n)$-sized proof in P, for any fixed $y \in \{0,1\}^m$. Proof complexity generators provides a systematic approach to attacking strong proof complexity lower bounds as well as $\mathsf{NP} \neq \mathsf{coNP}$.

Strong proof complexity generators are known to exist unconditionally against weak proof systems such as AC^0-Frege [Kha22]. For strong proof systems such as Frege or Extended Frege, Razborov [Raz15] and Krajíček [Kra19] proposed several candidate proof complexity generators; Indeed, Krajíček [Kra22] further conjectured that there is a proof complexity generator that fools *every* proof system. The construction of proof complexity generators from standard complexity-theoretic assumptions remains an important open problem (see the discussion in [Kra22]).

Theorem 1.6 ([RSW22])**.** $\mathsf{Avoid} \notin$ *i.o.* $\mathsf{SearchNP}$ *if and only if for every propositional proof system, there is a propositional proof complexity generator fooling it.*

This reduces the task of constructing strong proof complexity generators to the hardness of Avoid. Moreover, the lower bound for $\mathscr{C}$-Avoid in Problem 1.5 will imply a proof complexity generator computable by $\mathscr{C}$ circuits.

1.3 Our Results

The main focus of the paper is to extend our understanding to Problem 1.4 and 1.5. We show the hardness of these problems against deterministic algorithms for *restricted circuit classes* under variants of standard cryptographic assumptions. Moreover, by strengthening the assumptions, we prove that the problems above cannot be solved efficiently even by *non-deterministic algorithms*.

Hardness for range avoidance. Our first result is that the range avoidance problem is hard for even very simple circuits under variants of standard lattice assumptions, *learning-with-error* (LWE) and *inhomogeneous short integer solution* (ISIS), formally stated as follows.

Problem	Algorithms	Hardness	Capability
Avoid	$\mathsf{FP}^{\mathsf{NP}}$-reducible to $\mathsf{Hard}_{\varepsilon}$ [Kor21]	Not likely in FP [ILW23] Not likely in SearchNP (*our results*)	Most explicit constructions [Kor21; GLW22]
NC^1-Avoid	FP-reducible to NC^0-Avoid [RSW22]	Not likely in SearchNP (*our results*)	Good Linear Code, Rigid Matrices, etc. [GLW22]
ACC^0-Avoid & ACC^0-RPP	$\mathsf{FP}^{\mathsf{NP}}$ (weak parameters) [RSW22; CHLR23]	Not likely in SearchNP (*our results*)	Best known lower bounds against ACC^0 [CHLR23]
NC^0-Avoid	FP (weak parameters) [GLW22; GGNS23]	Not likely in SearchNP (*our results* + [RSW22])	NC^0-Avoid with strong parameters simulate NC^1-Avoid [RSW22]
XOR-RPP	$\mathsf{FP}^{\mathsf{NP}}$ (weak parameters) [CHLR23] FP (very weak parameters) [APY09]	-	Imply "help function lower bounds" [AS10]

Table 1: Summary of previous and our results for Avoid and RPP. Capability of $\mathscr{C}$-Avoid refers to the consequences that there is a deterministic algorithm for $\mathscr{C}$-Avoid. Note that our hardness results hold even for Avoid of depth-3 circuits and RPP of depth-2 circuits (see Theorems 1.11 and 1.12 for formal statements).

Assumption 1.7. *There exists a constant $\varepsilon \in (0,1)$ and $\beta = \beta(n) \in (0,1)$ such that the following holds. For sufficiently large n and every constant $\delta \in (0, 0.1)$, there exists $q = n^{O(1)}$, $m = O(n \log q)$, $\kappa = \Theta(m^{\delta})$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = q/(10\kappa)$ such that:*

- (LWE *against adaptive adversary*). *For at least a $2/3$ fraction of matrices $A \in \mathbb{Z}_q^{n \times m}$, there exists $\beta \cdot q^n$ vectors $v \in \mathbb{Z}_q^n$ such that no non-uniform polynomial-size adversary can distinguish the following two distributions with advantage $2^{-m^{\varepsilon}}$,*

$$\mathcal{U}(\mathbb{Z}_q^m \times \mathbb{Z}_q) \quad \text{and} \quad (sA + e, \langle v, s \rangle),$$

  where $s \leftarrow \mathcal{U}(\mathbb{Z}_q^n)$, $e \in \mathbb{Z}_q^m$ is uniformly random over strings satisfying $\|e\|_{\infty} = w$.
- (GapISIS *against non-determinism*). *Let $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be a function satisfying that $\mathsf{Ext}(x)$ is κ-sparse for every $x \in \{0,1\}^k$, and $g_A(x) := A \cdot \mathsf{Ext}(x)^{\top} \bmod q$. For at least a $2/3$ fraction of $A \in \mathbb{Z}_q^{n \times m}$, there is no polynomial-size proof system P (i.e. non-deterministic circuits) that rejects every string in $\mathsf{Range}(g_A)$, and accepts at least $\beta \cdot q^n$ strings in $\mathbb{Z}_q^n$.*

The first bullet states the LWE assumption against *adaptive adversary*, in the sense that after revealing the query matrix $(A \mid v)$, the adversary can *choose* a non-uniform circuit to attack LWE given the outcome $(sA + e, \langle v, s \rangle)$ of the query. This assumption is stronger than standard LWE as the choosing phase can be computationally unbounded; nevertheless, the security is still plausible as far as we know. The second bullet states that ISIS is hard on average to approximate against nondeterministic adversaries, which is closely related to the notion of demi-bits [Rud97; TZ23] and natural proofs [RR97]. Also note that the function Ext in the second bullet can be any standard approach to encode sparse vectors; specifically, we will choose a simple encoding that can be implemented in a single-layer AND circuit. We also verify that both of the assumptions are secure under known attacks (see Section 5.4).

The circuit class we will consider is a depth-3 class called $\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext}$, where:

- DOR stands for a disjoint OR gate, that is, an unbounded fan-in boolean OR gate with a semantic guarantee that at most one of its input wires is 1.
- EMAJ stands for an *exact majority gate*, which outputs 1 if and only if exactly one half of its input wires are 1.

- Ext stands for the function Ext in the second bullet of the assumption, which could be a layer of AND gates (see, e.g., Remark 5.4).

Formally, a (single-output) $\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext}$ circuit has a DOR gate on top of several EMAJ gates, whose input wires connect to the output of the function Ext. The size of the circuit is measured by the number of wires inside of it.

Theorem 1.8 (Theorem 5.10, informal)**.** *Under Assumption 1.7,* $(\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext})$-$\mathsf{Avoid}$ *cannot be solved by polynomial-size circuits on any sufficiently large input length.*

In particular, if Ext is implemented by a layer of AND gates, the theorem implies that the range avoidance problem is hard even for depth-3 TC^0 circuits.

Hardness for Remote Point. We also obtain a hardness result for remote point problem of even simpler circuit class under variants of standard assumptions related to binary linear codes, namely *learning parity with noise* (LPN) and *nearest codeword problem* (NCP).

Assumption 1.9. *There exists a constant $\varepsilon \in (0,1)$ and $\beta = \beta(n) \in (0,1)$ such that the following holds. For sufficiently large n and every constant $\delta \in (0, 0.1)$, there exists $m = O(n)$, $\kappa = \Theta(m^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = \Theta(m/\kappa^2)$ such that:*

- (LPN *against adaptive adversary*)*. For at least a $2/3$ fraction of matrices $A \in \{0,1\}_q^{n \times m}$, there exists $\beta \cdot 2^n$ vectors $v \in \{0,1\}^n$ such that no non-uniform polynomial-size adversary can distinguish the following two distributions with advantage 2^{-m^ε},*

$$\mathcal{U}(\{0,1\}^m \times \{0,1\}) \quad \text{and} \quad (sA + e, \langle v, s \rangle),$$

  where $s \leftarrow \mathcal{U}(\{0,1\}^n)$, $e \in \{0,1\}^m$ is uniformly random over strings satisfying $|e| = w$.
- (GapNCP *against non-determinism*)*. Let $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be a function satisfying that $\mathsf{Ext}(x)$ is κ-sparse for every $x \in \{0,1\}^k$, and $g_A(x) := A \cdot \mathsf{Ext}(x)^\top$. For at least a $2/3$ fraction of matrices $A \in \{0,1\}^{n \times m}$, there is no polynomial-size proof system P (i.e. non-deterministic circuits) that rejects every string in $\mathsf{Range}(g_A)$, and accepts at least $\beta \cdot 2^n$ strings in $\{0,1\}^n$.*

Intuitively, the assumptions we need can be considered as the Boolean version (i.e. $q = 2$) of Assumption 1.7. As far as we know, these two assumptions and the hardness results based on them are incomparable.

The benefit of using coding-based conjecture is to reduce the circuit complexity in the hardness result. We will consider the *remote point problem* for $\mathsf{XOR} \circ \mathsf{Ext}$ circuits, which consists of an XOR gate on the top, whose input wires are from the output of Ext. In particular, we can implement Ext by AND gates of $O(\log n)$ fan-in (see Remark 5.4), so that each output bit is simply a degree-$O(\log n)$ polynomial over $\mathsf{GF}(2)$.

Theorem 1.10 (Theorem 5.7, informal)**.** *Under Assumption 1.9,* $(\mathsf{XOR} \circ \mathsf{Ext})$-$\mathsf{RPP}_{\Omega(1)}$ *cannot be solved by polynomial-size circuits on any sufficiently large input length.*

Note that if the decoder of an asymptotic good error-correcting code can be implemented in circuit class $\mathscr{D}$, then $\mathscr{C}$-$\mathsf{RPP}_{\Omega(1)}$ can be reduced to $(\mathscr{D} \circ \mathscr{C})$-$\mathsf{Avoid}$ (see, e.g., [CHLR23]). However, since we do not have a super-efficient decoder, it is unknown whether the conclusion of Theorem 1.8 implies that of Theorem 1.10.

Hardness against nondeterministic algorithms. An appealing feature of Theorem 1.8 and 1.10 is that they can be strengthened to show the hardness of range avoidance and remote points against *nondeterministic algorithms*.

A nondeterministic search algorithm, or SearchNP algorithm, is a non-deterministic Turing machine M that outputs a string on each of its accepting states. It is said to solve a total search problem if, for every input x, there is an accepting computation path for $M(x)$, and for every accepting computation path for $M(x)$, it outputs a correct answer to the search problem on input x. A non-uniform SearchNP algorithm is an SearchNP algorithm together with a non-uniform advice of length $\mathsf{poly}(n)$ on each input length n.

Theorem 1.11 (Theorem 5.10, informal)**.** $(\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext})$-Avoid *cannot be solved by any polynomial-time non-uniform* SearchNP *algorithm on any sufficiently large input length, under Assumption 1.7 with the first bullet replaced by the following stronger assumption:*

- (LWE *against adaptive nondeterministic adversary*). *For at least a* $2/3$ *fraction of matrices* $A \in \mathbb{Z}_q^{n\times m}$*, there exists* $\beta \cdot q^n$ *vectors* $v \in \mathbb{Z}_q^n$ *such that for every polynomial-size nondeterministic circuit* C*,*

$$\Pr_{(x,y)\leftarrow\mathcal{U}(\mathbb{Z}_q^m\times\mathbb{Z}_q)}[C(x,y)=1] - \Pr_{s,e}[C(sA+e,\langle v,s\rangle)=1] < 2^{-m^{\varepsilon}}, \tag{1}$$

where $s \leftarrow \mathcal{U}(\mathbb{Z}_q^n)$*,* $e \in \mathbb{Z}_q^m$ *is uniformly random over strings satisfying* $\|e\|_\infty = w$*.*

The assumption states that LWE is secure against adaptive adversary even if it can choose a nondeterministic circuit, in the sense that it cannot accept uniformly random input sufficiently more often than the LWE samples. This is equivalent to say that the LWE function $f_{A,v}(s,e) := (sA+e,\langle v,s\rangle)$ is a *super-bits generator* defined by Rudich [Rud97] (also see Section 2.6). Note that the absence of the absolute value over the subtraction in Equation (1), which occurs in the standard definition of indistinguishability, is necessary, as the following simple nondeterministic algorithm accepts LWE samples much more often than the uniform distribution:

- Given the input $(x,y) \in \mathbb{Z}_q^n \times \mathbb{Z}_q$, it guesses $s \in \mathbb{Z}_q^n$ and $e \in \mathbb{Z}_q^m$, and accepts if $x = sA+e$, $y = \langle v,s\rangle$, and $\|e\|_\infty = w$.

Although the assumptions could be much stronger than the first bullet of Assumption 1.7 and 1.9, the parameter regime that we are considering is still plausible as far as we can see. In particular, we show in Section 5.4 that there is a large gap between the parameter regime we need and that can be broken using seed-guessing, linear algebraic attacks, and geometric attacks (i.e. the nondeterministic algorithm for GapCVP [GG98; AR05]).

Similarly, we can strengthen Assumption 1.9 to obtain the hardness of $(\mathsf{XOR} \circ \mathsf{Ext})$-$\mathsf{RPP}_{\Omega(1)}$ against non-uniform SearchNP algorithms.

Theorem 1.12 (Theorem 5.7, informal)**.** $(\mathsf{XOR}\circ\mathsf{Ext})$-$\mathsf{RPP}_{\Omega(1)}$ *cannot be solved by polynomial-time non-uniform* SearchNP *algorithms for sufficiently large input lengths, under Assumption 1.9 with the first bullet replaced by the following stronger assumption:*

- (LPN *against adaptive nondeterministic adversary*). *For at least a* $2/3$ *fraction of matrices* $A \in \{0,1\}^{n\times m}$*, there exists* $\beta \cdot 2^n$ *strings* $v \in \{0,1\}^n$ *such that for every polynomial-size nondeterministic circuit* C*,*

$$\Pr_{(x,y)\leftarrow\mathcal{U}(\{0,1\}^m\times\{0,1\})}[C(x,y)=1] - \Pr_{s,e}[C(sA+e,\langle v,s\rangle)=1] < 2^{-m^{\varepsilon}}, \tag{2}$$

where $s \leftarrow \mathcal{U}(\{0,1\}^n)$*,* $e \in \{0,1\}^m$ *is uniformly random over strings satisfying* $|e| = w$*.*

Witness encryption for problems in $\mathsf{NP} \setminus \mathsf{coNP}$. The hardness results for Avoid of Ilango, Li, and Williams [ILW23] is based on $\mathsf{NP} \neq \mathsf{coNP}$ and *witness encryption* for NP. Although witness encryption is considered to be a plausible cryptographic primitive, all known candidates [GGSW13; CVW18; Bar+20; BIOW20; VWW22; Tsa22; JLS21; JLS22] are too complicated for us to generalize the results in [ILW23] to restricted circuit classes and hardness against nondeterminism.

One of our main technical tools is a new construction of witness encryption for a promise problem in NP that is plausibly not in $\mathsf{coNP}_{/\mathsf{poly}}$, which admits all properties required for the hardness result in [ILW23]. An appealing feature of our construction is its simplicity: we define a promise language in NP based on variants of existing *public-key encryption schemes* [AD97; Reg09; Ale11][2], and the encryption and decryption algorithms are simply that of the public-key encryption scheme.

Theorem 1.13 (Theorems 5.1 and 5.8, informal)**.** *Under Assumption 1.7 or Assumption 1.9, there is a promise language $L = (L^{\mathsf{YES}}, L^{\mathsf{NO}})$ in $\mathsf{NP}_{/\mathsf{poly}} \not\subseteq \mathsf{coNP}_{/\mathsf{poly}}$ that admits a sub-exponentially secure witness encryption satisfying the following properties.*

- *(Succinct proofs). Let n be the input length, $k = k(n)$ be the proof length, and $\varepsilon = \varepsilon(n) = 2^{-n^{\Omega(1)}}$ be the maximum advantage that can be obtained by polynomial-size adversaries to break the witness encryption. Then $k = o(\ln(\varepsilon^{-1}))$. Namely, the proof length is succinct compared to the security level of the witness encryption.*
- *(Efficient decryption). The decryption circuit $\mathsf{Dec}(\mathsf{ct}, x, \cdot)$, which decrypts a hardwired ciphertext ct given the proof for a hardwired input $x \in L^{\mathsf{YES}}_n$, can be implemented in weak circuit classes (i.e. $\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext}$ under Assumption 1.7 and $\mathsf{XOR} \circ \mathsf{Ext}$ under Assumption 1.9).*
- *(Security against nondeterminism). If, in addition, the first bullet of Assumption 1.7 or 1.9 satisfies the stronger notion of "adaptive security against nondeterministic adversary" (see Theorem 1.11 and 1.12), then the witness encryption is secure against nondeterministic adversary in the following sense:*
  - *There is a polynomial-time randomized algorithm S (called simulator) such that for every $x \in L^{\mathsf{NO}}_n$, every message $b \in \{0,1\}$, and every s-size non-deterministic adversary Adv,*

$$\Pr[\mathsf{Adv}(S(1^n, x)) = 1] - \Pr[\mathsf{Adv}(\mathsf{Enc}(b, x)) = 1] < \varepsilon,$$

  where $s = s(n) = \mathsf{poly}(n)$ and $\varepsilon = \varepsilon(n) = 2^{-n^{\Omega(1)}}$.
- *(Decryption error). The decryption of the witness encryption scheme is perfectly correct under Assumption 1.7, and $n^{-\Omega(1)}$ under Assumption 1.9.*

Our construction is completely different from the existing paradigms such as GGH-encoding [CVW18; VWW22], Affine determinant programs [Bar+20], generic group model [BIOW20], and indistinguishability obfuscation (see, e.g., [JLS21; JLS22]), and is surprisingly simple. Intuitively, our construction relies on the observation that a public-key encryption scheme, whose hardness relies on a hard problem called its PKE problem, can be viewed as the witness encryption of its PKE problem (see Section 1.5 for more details). As a drawback, we can only construct witness encryption for a *special hard language* instead of all languages in NP.

[2]It might be confusing why we can obtain hard languages plausibly not in coNP from these encryption schemes, as the hard problems underlying [AD97; Reg09] are known to be in $\mathsf{NP} \cap \mathsf{coNP}$ [GG98; AR05]. Intuitively, we avoid this attack and plausibly avoid all similar attacks by introducing an additional assumption (i.e. $\mathsf{GapISIS}$, see Assumption 1.7) so that we can set the parameters beyond the capacity of the leftover hash lemma (see, e.g., [BBD09, Chapter 5]). More details can be found in Section 1.5.

1.4 Related Works

Hardness of range avoidance. Our results are based on the general connection between witness encryption and the hardness of range avoidance developed by Ilango, Li, and Williams [ILW23]. They proved that assuming sub-exponentially secure witness encryption for NP with perfect correctness and NP $\neq$ coNP, range avoidance is hard against polynomial-time deterministic algorithms. We strengthened their results in several dimensions: We established a connection between *remote point problem* and witness encryption with imperfect correctness, generalized their results to hardness against SearchNP, and provided concrete witness encryption constructions that imply the hardness of range avoidance and remote point problems for restricted circuits.

Cryptography against nondeterminism. Rudich [Rud97] introduced the notion of demi-bits and super-bits generators as the generalization of pseudorandom generators against nondeterministic adversary, which the application of ruling out $\mathsf{NP}_{/\mathsf{poly}}$ natural proofs for circuit lower bounds. Demi-bits generators are weaker than super-bits, with the drawback of losing some properties of PRGs such as stretching from $n \to n+1$ bits generators to $n \to 2n$ bits generators (see [TZ23] for detailed discussion). Our definitions for PKE and WE against nondeterminism are inspired by Rudich's definition and the standard simulation paradigm in cryptography. Our assumptions can be interpreted as new candidates of demi-bits and super-bits generators based on lattice and coding problems (see Assumption 5.5 and 5.9 for details).

Concrete cryptographic candidates and complexity theory. Our hardness proof for Avoid and RPP utilizes specific PKE constructions from lattice [AD97; Reg09] and coding problems [Ale11]. Indeed, there are several other examples for *concrete cryptographic constructions* to have applications in the frontier of complexity theory: Hirahara's proof for the NP-hardness of PartialMCSP [Hir22] utilized secret-sharing schemes and private-key encryption; Huang, Ilango, and Ren [HIR23] proved the NP-hardness of meta-complexity problems *unconditionally* based on a construction of witness encryption in generic group model; The proof of Chen et al. [CLORS23] for pseudo-deterministic constructions of primes relies on Goldreich-Levin construction of PRGs from OWPs [GL89]. These results emphasize that cryptographic constructions can be used in complexity theory not only as assumptions in a black-box fashion but also as a tool to connect different notions of hardness.

1.5 Technical Overview

Now we briefly explain the proof of our results and highlight the main technical challenges.

A generalization of [ILW23]. Our results build on the hardness result of the range avoidance problem by Ilango, Li, and Williams [ILW23]. They proved that Avoid $\notin$ FP assuming NP $\neq$ coNP and the existence of a sub-exponentially secure, perfectly correct witness encryption for SAT. Intuitively, their proof follows the intuition that a deterministic algorithm for Avoid can not only search for a solution (i.e., a string outside of the range of the input circuit) but also *certify* the correctness of the solution, which leads to an efficient proof system for UNSAT and therefore NP = coNP.

Now we explain their technique more formally. Let $(\mathsf{Enc}, \mathsf{Dec})$ be a witness encryption scheme for SAT. The proof system for UNSAT works as follows. For a formula φ over n variables, the proof system accepts φ if and only if there is a string $y \in \{0,1\}^n$ and a random tape $r \in \{0,1\}^{\mathsf{poly}(n)}$ for

the witness encryption scheme such that

$$\mathsf{Avoid}(\mathsf{Dec}(\mathsf{Enc}(\varphi, y; r), \cdot)) = y.$$

In other words, the proof system accepts if there is a message y and a random tape r such that:

- Let ct be the encryption of y on the statement $\varphi \in \mathsf{SAT}$ using the randomness r.
- Let $\mathsf{Dec}(\mathsf{ct}, \varphi, \cdot)$ be the circuit takes a witness of $\varphi \in \mathsf{SAT}$ (i.e. a satisfying assignment $w \in \{0,1\}^n$) as input and decrypts the hard-wired cipher text ct.
- The proof system accepts the proof (y, r) if and only if the range avoidance algorithm says that y is outside of the range of the circuit $\mathsf{Dec}(\mathsf{ct}, \varphi, \cdot)$.

It is easy to see that the correctness of the decryption algorithm ensures that if φ is satisfiable, then the proof system will always reject; that is, the proof system is sound. Indeed, the completeness can also be proved using the security property of the witness encryption scheme (see Section 3.1 for more details).

With a closer inspection of their proof, it is easy to verify that if the decryption algorithm of the witness encryption scheme can be implemented in a restricted circuit class, say ACC^0, then the hardness result works for ACC^0-Avoid. Also, it is not necessary to have witness encryption schemes for all NP languages; instead, it suffices to have any particular language $L \in \mathsf{NP}_{/\mathsf{poly}} \setminus \mathsf{coNP}_{/\mathsf{poly}}$ with a secure witness encryption scheme. Moreover, we generalize their results in two dimensions.

- We prove that if, instead of the range avoidance problem, we have an algorithm for the remote point problem, we can design a similar proof system for UNSAT even if the decryption algorithm Dec has a small decryption error. This allows us to obtain the hardness of the remote point problem (or $\mathscr{C}$-RPP) from a witness encryption scheme with imperfect correctness.
- We define the security of witness encryption against nondeterministic adversaries (see Theorem 1.13 and Section 3.2), and prove that assuming such strong security of the witness encryption, we can obtain stronger consequences that the range avoidance problem or remote point problem are not in $\mathsf{SearchNP}$.

Lemma 1.14 (Lemma 3.3, informal)**.** *Range avoidance problem cannot be solved in deterministic polynomial time if there is a promise language $L \in \mathsf{NP}_{/\mathsf{poly}} \setminus \mathsf{coNP}_{/\mathsf{poly}}$ that admits a subexponentially secure and perfectly correct witness encryption scheme. Moreover:*

- *If the decryption algorithm has a constant decryption error instead of being perfectly correct, then the hardness result still holds for the remote point problem.*
- *If the decryption algorithm of the witness encryption scheme can be implemented by $\mathscr{C}$ circuits, for any typical circuit classes $\mathscr{C}$, then the hardness result holds for $\mathscr{C}$-Avoid (or $\mathscr{C}$-RPP if the decryption algorithm is not perfectly correct).*
- *If the witness encryption scheme is secure against nondeterministic adversaries, then the range avoidance problem (or remote point problem, $\mathscr{C}$-Avoid, $\mathscr{C}$-RPP as discussed above) cannot be solved by $\mathsf{SearchNP}$ algorithms.*

Witness encryption inspired by public-key encryption. With Lemma 1.14, we can reduce the hardness of the range avoidance or remote point problems to the construction of witness encryption schemes, preferably with low decryption complexity and security against nondeterminism. However, there is no known candidate witness encryption whose decryption can be implemented in bounded-depth circuits, say $\mathsf{AC}^0[2]$, and it is not clear whether existing candidates [CVW18; VWW22; Bar+20; BIOW20; JLS21; JLS22] are secure against nondeterministic adversaries. Indeed, proposing a simple candidate witness encryption for NP is a long-standing open problem, and

it is probably hard to solve as many advanced cryptographic primitives including identity-based encryption (IBE) and attribute-based encryption (ABE) can be derived from witness encryption [GGSW13].

Our result relies on the observation that many public-key encryption schemes can be translated into the witness encryption scheme of some hard problems that are probably not NP-complete. This is sufficient for our purpose as Lemma 1.14 works with the witness encryption of any promise language in $\mathsf{NP}_{/\mathsf{poly}} \setminus \mathsf{coNP}_{/\mathsf{poly}}$, and it does not necessarily imply witness encryption schemes for all NP languages.

The translation works with a class of public-key encryption schemes called *PKEs with pseudorandom public keys* ($\widetilde{\text{PKE}}$ for short). Let ℓ be the length of the public keys. A PKE scheme $(\mathsf{Enc}, \mathsf{Dec}, \mathsf{Gen})$ is said to be a $\widetilde{\text{PKE}}$ if there is a distribution $\mathcal{D}$ supported over $\{0,1\}^\ell$, which is called the "ideal distribution" for the public key, such that the following properties hold.

- (*Hardness of PKE Problem*). Let $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$ be the keys. The *PKE problem* of the scheme is the task to distinguish between the "ideal world", i.e. $u \leftarrow \mathcal{D}$, and the "real world", i.e. the actual public key pk. This property suggests that these two distributions (i.e. the ideal world and the real world) are computationally indistinguishable.
- (*Security in Ideal World*). This property shows that the encryption scheme is secure when the public key is drawn from the ideal distribution $\mathcal{D}$. That is, $\mathsf{Enc}(\mathcal{D}, 0)$ and $\mathsf{Enc}(\mathcal{D}, 1)$ are computational indistinguishable.
- (*Verifiable Public Key*). There is an NP proof system that verifies the public key with the secret key, i.e., it accepts pk if and only if the witness is a possible secret-key for pk.

The notion of $\widetilde{\text{PKE}}$ is a generalization of a model known as *meaningful/meaningless encryption* [KN08] or *dual-mode cryptosystem* [PVW08], in which the encryption is statistically secure in ideal world. Indeed, it is easy to verify that many standard public-key encryption schemes (such as Regev's lattice-based PKE [Reg09] and the Goldwasser-Micali cryptosystem from the hardness of quadratic residuosity [GM84]) are $\widetilde{\text{PKE}}$ (see Examples 4.3 and 4.4).

The intuition behind the translation is that, if we are given a $\widetilde{\text{PKE}}$ scheme $(\mathsf{Enc}, \mathsf{Dec}, \mathsf{Gen})$, then the PKE problem derives a *hard problem* that admits a *witness encryption scheme*. Ideally, we want to define the following promise problem $L = (\Pi_\ell^{\mathsf{YES}}, \Pi_\ell^{\mathsf{NO}})$ where

- Π_ℓ^{YES} consists of all valid public keys of length ℓ;
- Π_ℓ^{NO} consists of all strings y such that $\mathsf{Enc}(y, 0)$ and $\mathsf{Enc}(y, 1)$, the distributions of cipher texts over the internal randomness of $\mathsf{Enc}(y, \cdot)$, are computationally indistinguishable.

This language is clearly in NP according to the verifiability of the public keys. Moreover, the encryption and decryption of the $\widetilde{\text{PKE}}$ scheme is a correct and secure witness encryption for L, where the security condition is encoded in the definition of Π_ℓ^{NO}. The only problem is that the hardness of L, say $L \notin \mathsf{P}_{/\mathsf{poly}}$, does not follow from the standard hardness of the PKE problem.

To address this issue, we introduce the following stronger notion of hardness for the PKE problem, called *adaptive hardness*.

- (*Adaptive Hardness of PKE Problem*). Let $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$ be the keys. The PKE problem is said to be hard against adaptive adversary if we sample $y \leftarrow \mathcal{D}$, then with high probability, $\mathsf{Enc}(y, 0)$ and $\mathsf{Enc}(y, 1)$ are computationally indistinguishable against any non-uniform adversary. In other words, $\mathsf{Enc}(\mathcal{D}, 0)$ and $\mathsf{Enc}(\mathcal{D}, 1)$ are indistinguishable even if the adversary can choose the attacking algorithm after it observes the ideal public key $y \leftarrow \mathcal{D}$, and the choosing phase can be computationally unbounded.

Therefore we can conclude the following.

Theorem 1.15 (Theorem 4.5, informal)**.** *Assume that there is a $\widetilde{PKE}$ scheme with adaptive hardness of its PKE problem. Then there is a language $L \in \mathsf{NP} \setminus \mathsf{P}_{/\mathsf{poly}}$ that admits a secure witness encryption scheme. Moreover, the decryption algorithm of the witness encryption scheme is exactly the decryption algorithm of the original $\widetilde{PKE}$.*

Indeed, we can also strength the hardness of the PKE problem and the security of the $\widetilde{\mathsf{PKE}}$ to against nondeterministic adversaries to obtain a language in $\mathsf{NP} \setminus \mathsf{coNP}_{/\mathsf{poly}}$ that admits a witness encryption secure against nondeterministic adversaries (see Section 4.3 for more details).

Instantiation. To prove the hardness of the range avoidance and remote point problem from plausible cryptographic assumptions, it remains to design a $\widetilde{\mathsf{PKE}}$ scheme and compile it into a witness encryption by Theorem 1.15 that satisfies the requirement of Lemma 1.14. Since Lemma 1.14 requires security against nondeterministic adversaries, the PKE candidates based on trapdoor one-way permutations or the hardness of discrete logarithm (e.g., RSA and ElGamal) do not work as they are inherently in $\mathsf{NP} \cap \mathsf{coNP}$.

The starting point of our instantiation is the standard lattice-based PKE scheme known as *dual Regev* introduced by Gentry, Peikert, and Vaikuntanathan [GPV08], which is a variant of Regev's public-key encryption from LWE [Reg09]. Let n be the security parameter, $q = \mathsf{poly}(n)$, $m = O(n \log q)$, and Ψ and χ be error distributions supported over $\mathbb{Z}_q^m$ (e.g., Gaussian over $\mathbb{Z}^m$ or uniformly random 0-1 vectors). The PKE scheme is as follows.

- (*Key generation*). Let $A \leftarrow \mathbb{Z}_q^{n \times m}$ be a random matrix, and $v = Ax \bmod q \in \mathbb{Z}_q^n$ for $x \leftarrow \Psi$. The public key is $(A, v) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^n$, and the secret key is x.
- (*Encryption*). Let $A \in \mathbb{Z}_q^{n \times m}$ be the public key. To encrypt a bit $b \in \{0, 1\}$, sample $s \leftarrow \mathbb{Z}_q^n$ uniform and compute $p = sA + e \in \mathbb{Z}_q^m$ for some noise $e \leftarrow \chi$, and the cipher text will be $(p, \langle v, s \rangle + b \cdot \lfloor q/2 \rfloor) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$.
- (*Decryption*). Let (p, c) be the cipher text. The message is 1 if and only if $b' = c - \langle v, x \rangle \in \mathbb{Z}_q$ is closer to $\lfloor q/2 \rfloor$ than to 0 modulo q.

In standard settings (see, e.g., [PVW08]), one can choose Ψ and χ to be Gaussian distributions with certain parameters so that the distribution of the public key $(A, v = Ax)$ is *statistically close* to the uniform distribution, and therefore the security follows from the hardness of LWE. To achieve this, the entropy of the distribution Ψ should be sufficiently high so that the *leftover hash lemma* can be applied. We can then interpret this scheme as a $\widetilde{\mathsf{PKE}}$ scheme as follows.

- (*Ideal Distribution*). Let the ideal distribution $\mathcal{D}$ be the uniform distribution over $\mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^n$.
- (*Hardness of PKE Problem*). The PKE problem is to distinguish the ideal public key distribution $\mathcal{D}$ and the real public key distribution $(A, v = Ax) \in \mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^n$, where $x \leftarrow \Psi$. In standard settings, these two distributions are statistically close.
- (*Security in Ideal World*). This property means that $\mathsf{Enc}(\mathcal{D}, 0)$ is indistinguishable from $\mathsf{Enc}(\mathcal{D}, 1)$. By unwinding the construction, we can see that this is exactly the LWE assumption.
- (*Verifiable Public Key*). This is obvious as given the secret key x, it is easy to verify that (A, v) is a corresponding public key by checking $v = Ax \bmod q$.

We cannot put this scheme into Lemma 1.14 directly to obtain a hard language that admits a witness encryption scheme for several technical reasons. Firstly, the scheme is *not* adaptively secure as for most of ideal public keys $(A, v) \leftarrow \mathcal{D}$, there is some x over the typical set of the error distribution Ψ such that $v = Ax \bmod q$, i.e., most ideal public keys are real public keys! This is inevitable, as otherwise the ideal and real public key distributions cannot be statistically close. Therefore, an adaptive adversary can compute the secret key after observing the ideal public key

in the unbounded choosing phase and use it to break the scheme easily. Moreover, the encryption scheme is not secure against nondeterministic adversaries (see Section 4.3 for formal definitions) as LWE in the standard parameter setting is known to be in $\mathsf{NP} \cap \mathsf{coNP}_{/\mathsf{poly}}$ (see [GG98; AR05]). Furthermore, there is a subtle issue that the security in ideal world, which corresponds to the security of the witness encryption scheme, is not strong enough with respect to the length of the secret key to obtain non-trivial hardness results for the range avoidance or remote point problem.

Fortunately, we manage to resolve all these three technical problems with a simple trick. We *reduce* the entropy of the distribution Ψ for the public key generation to the extent that the real public key distribution $(A, v = Ax)$ for $x \leftarrow \Psi$ is statistically *far* from being uniformly random. Therefore, the ideal public keys are no longer real public keys, which allows us to plausibly conjecture the security against adaptive adversaries. Correspondingly, we *increase* the entropy of the noise distribution χ for LWE so that it is plausibly not in $\mathsf{NP} \cap \mathsf{coNP}_{/\mathsf{poly}}$. These adjustments will maintain the correctness of the decryption algorithm as long as $\mathbb{E}[\langle x, e\rangle] \ll q/2$ for $x \leftarrow \Psi$ and $e \leftarrow \chi$. For security, we need to have two hardness assumptions listed as follows rather than one as the hardness of the PKE problem no longer follows from the leftover hash lemma.

- To show that the encryption is secure in ideal world, we need to assume that the LWE distribution $(sA + e, \langle v, s\rangle) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$ for $s \leftarrow \mathbb{Z}_q^n$ and $e \leftarrow \chi$ and the uniform distribution are indistinguishable against *(nondeterministic) adaptive adversary*. That is, for $A \leftarrow \mathbb{Z}_q^{n\times m}$, $x \leftarrow \Psi$, and $v = Ax$, with high probability, there is no non-uniform (nondeterministic) algorithm that accepts the uniform distribution sufficiently more often than the LWE distribution. (Hardness against nondeterministic algorithms is needed to obtain witness encryption secure against nondeterminism.) This is exactly the first bullet of Assumption 1.7 and is a variant of the standard LWE assumption.
- To show that the PKE problem is hard, we need to assume $(A, v = Ax) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n$ for $x \leftarrow \Psi$ and the uniform distribution are indistinguishable against nondeterministic algorithms, in the sense that there is no efficient nondeterministic algorithm that accepts the uniform distribution with decent probability and rejects $(A, v = Ax)$ with probability 1. This corresponds to the second bullet of Assumption 1.7 and is a variant of the standard *Short Integer Solution* assumption.

Moreover, since the security in ideal world is parameterized by the entropy of Ψ, which can be much larger compared to the length of the secret key, this trick also solves the third technical issue. This leads to the witness encryption construction in Theorem 1.13 from Assumption 1.7. Similarly, we can also perform the same trick to Alekhnovich's PKE scheme [Ale11], which is essentially a binary analog of Regev's PKE scheme [Reg09], to obtain the witness encryption in Theorem 1.13 from Assumption 1.9.

Note that although we state our results in terms of LWE with ℓ_∞-norm (see Theorem 1.13 and Assumption 1.7), the same construction still works and is still plausibly secure for other natural norms, say ℓ_1 or ℓ_2. The reason that we choose ℓ_∞ is because it reduces the circuit complexity of the decryption circuits and achieves perfect correctness for decryption, which lead to stronger hardness results for the range avoidance problem.

1.6 Open Problems

Hardness of remote point for XOR circuits. We have shown in Theorem 1.10 that the remote point problem for $\mathsf{XOR} \circ \mathsf{AND}$ circuits is hard against nondeterministic algorithms from plausible cryptographic assumptions. However, it is unclear how to strengthen our technique to show the hardness of XOR-RPP when the circuit is as simple as an XOR gate, or equivalently, a linear function

over $\mathsf{GF}(2)$, which is the original model studied in [APY09] as an intermediate task towards the construction of rigid matrices. Similarly, it is interesting to explore the hardness of range avoidance for restricted arithmetic circuits, say linear functions over finite fields or low-degree polynomials.

Assumptions independent of concrete structures. In this paper, we show the hardness of the range avoidance and remote point problem against nondeterministic algorithms from latticed-based assumptions and coding-based assumptions, respectively. These assumptions heavily rely on specific algebraic structures such as $\mathbb{Z}_q$ or $\mathsf{GF}(2)$. This seems necessarily for the hardness against extremely weak circuit models like $\mathsf{XOR} \circ \mathsf{AND}$. However, if we only want to show the hardness of the range avoidance problem for general Boolean circuits against nondeterministic algorithms, it is more desirable to have assumptions that are independent of concrete structures such as the existence of "mainstream" cryptographic primitives or separation of complexity classes. Note that it might be hard to build the hardness results without any cryptographic assumption by improving our current techniques, as it is already a longstanding open problem to have a candidate of public-key encryption whose security is independent of concrete structures (see, e.g., [Bar17]).

Cryptographic applications of our results. In this paper we provide two simple candidates of witness encryption for certain promise problems that are not likely to be NP-complete. An important future direction is to further investigate the security of our candidate witness encryption schemes. Another interesting question is: if our witness encrytion schemes are secure, would they lead to new cryptographic constructions? Recall from [GGSW13] that witness encryption for *all* NP languages implies public-key encryption, identity-based encryption, attribute-based encryption, and more advanced cryptographic primitives. Although those implications do not necessarily need the full power of NP, they certainly need sufficiently complicated languages, for which we do not know how to reduce to the promise problems that our witness encryption schemes can handle. So an interesting direction is finding applications for our witness encryption in designing advanced cryptographic functionalities.

Acknowledgement. We thank Richard Ryan Williams for insightful discussion and providing the open problem to show hardness of range avoidance from assumptions without concrete structures. Parts of this work were done when Jiatu Li was an undergraduate student in Tsinghua university, and was visiting Shanghai Qi Zhi Institute as a research intern. Yilei Chen is supported by Tsinghua University startup funding and Shanghai Qi Zhi Institute. Jiatu Li is supported by an Akamai Presidential Fellowship.

2 Preliminaries

We assume basic familiarity to complexity theory (e.g. complexity classes and circuit classes) and cryptography (see [AB09]).

Notation. Let $[m] := \{1, 2, \ldots, m\}$. We use the Iverson bracket $[\phi]$ that equals 1 if ϕ is true and equals 0 otherwise. For a finite set S, $\mathcal{U}(S)$ denotes the uniform distribution over S; in particular, $\mathcal{U}_n := \mathcal{U}(\{0,1\}^n)$.

For a string $x \in \{0,1\}^n$, the Hamming weight of x, denoted by $|x|$, is the number of 1's in x. The Hamming distance between $x, y \in \{0,1\}^n$, denoted by $|x - y|$, is defined as the Hamming weight of bitwise-XOR of x and y. A string x is said to be κ-sparse if $|x| \leq \kappa$. A string $y \in \{0,1\}^n$ is said to be δ-far from $x \in \{0,1\}^n$ if $|y - x| \geq \delta n$. We use $x \| y$ to denote the concatenation of two

strings x and y. For any function $f : X \to Y$, we use $\mathsf{Range}(f) := \{f(x) \mid x \in X\}$ to denote the range of f.

Let q be a modulus and $x \in \mathbb{Z}_q$. We define the norm $|x| = \min\{x, q-x\}$. We use $\|x\|_k$ to denote the ℓ_k-norm of a vector v for $k \in \mathbb{N} \cup \{\infty\}$. In particular, we use $\|v\|$ to denote the ℓ_∞-norm of v, i.e., $\|v\| = \max_i\{|v_i|\}$. We use $\mathbb{Z}_n^\times$ to denote the multiplicative group of $\mathbb{Z}_n$.

If there is no ambiguity, we identify $\{0,1\}$ and $\mathsf{GF}(2)$, and identify a string $x \in \{0,1\}^n$ and an n-dimensional row vector in $\mathsf{GF}(2)$. We use $\langle \cdot, \cdot \rangle$ and $+$ to denote the inner product and vector addition, respectively. (We also use $-$ to represent $+$ in $\mathsf{GF}(2)$, since $-1 = 1$.)

2.1 Circuit Classes

We define the types of gates and circuit classes mentioned in this paper.

- AND_d denotes a Boolean AND gate of fan-in at most d, and AND denotes a Boolean AND gate of unbounded fan-in.
- XOR denotes a Boolean Exclusive-OR gate of unbounded fan-in.
- DOR denotes a disjoint Boolean OR gate of unbounded fan-in, i.e., a Boolean OR gate with a semantic promise that at most one of its input wires is 1.
- EMAJ denotes an exact majority gate (see, e.g., [HP10]). An exact majority gate outputs 1 if and only if exactly one half of its input wires are 1.
- For any gate G, an m-output G circuit C is a circuit with m output gates, each of which is a G gate of the input of C.
- For any multi-output function F, an F circuit is a circuit that computes F. This is a convention for convenience of discussion.
- For two circcuit classes $\mathscr{C}$ and $\mathscr{D}$, the composition of $\mathscr{C}$ and $\mathscr{D}$, denoted by $\mathscr{C} \circ \mathscr{D}$, is the circuit class consisting of circuits of form $C(D_1(x), D_2(x), \cdots, D_k(x))$, where x is the input, $C \in \mathscr{C}$, $D_1, \ldots, D_k \in \mathscr{D}$.

As we are considering gates of unbounded fan-in, we measure the size of a circuit by the number of wires.

2.2 Promise Languages in NP

A promise language $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ is defined as a pair of disjoint sets $\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}} \subseteq \{0,1\}^*$, where Π^{YES} (called YES-instances) denotes the set of strings in L, Π^{NO} (called NO-instances) denotes the set of strings not in L, and we do not care about the strings in $\{0,1\}^* \setminus \Pi^{\mathsf{YES}} \cup \Pi^{\mathsf{NO}}$. Similarly, we can define $L_n = (\Pi_n^{\mathsf{YES}}, \Pi_n^{\mathsf{NO}}) \subseteq \{0,1\}^n \times \{0,1\}^n$ as a promise language restricted to length-n strings.

A algorithm $V(x,y)$ (resp. circuit $V(x,y)$) is said to be an verifier of $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ (resp. $L_n = (\Pi_n^{\mathsf{YES}}, \Pi_n^{\mathsf{NO}})$) if there is a polynomial $p(n)$ such that the following two conditions hold.

- **(Completeness).** For every $x \in \Pi^{\mathsf{YES}} \cap \{0,1\}^n$ (resp. $x \in \Pi_n^{\mathsf{YES}}$), there is a witness $w \in \{0,1\}^{p(n)}$ such that $V(x,y) = 1$.
- **(Soundness).** For every $x \in \Pi^{\mathsf{NO}} \cap \{0,1\}^n$ (resp. $x \in \Pi_n^{\mathsf{NO}}$) and every witness $w \in \{0,1\}^{p(n)}$, $V(x,y) = 0$.

We use $\mathsf{PromiseNP}$ to denote the set of promise languages that admit polynomial-time verifiers. Similarly, $\mathsf{PromiseNP}_{/\mathsf{poly}}$ denotes the set of promise languages that admit polynomial-sized verifiers.

2.3 Goldwasser-Sipser Protocol

We will use the standard AM protocol for the set lower bound problem.

Theorem 2.1 (Goldwasser and Sipser [GS89])**.** *There is an* AM*-protocol that works as follows. Given an* $\mathsf{NP}_{/\mathsf{poly}}$ *circuit* $C : \{0,1\}^n \to \{0,1\}$ *and a parameter* $s \leq 2^n$*.*

- (Completeness). *If* $|\{x \mid C(x) = 1\}| \geq s$*, there is a prover such that the verifier accepts with probability at least* $2/3$*.*
- (Soundness). *If* $|\{x \mid C(x) = 1\}| \leq s/2$*, for every prover, the verifier accepts with probability at most* $1/3$*.*

Moreover, since AM can be derandomized using advice (see, e.g., [AB09]), there is an $\mathsf{NP}_{/\mathsf{poly}}$ circuit that accepts (C, s) if $|\{x \mid C(x) = 1\}| \geq s$ and rejects (C, s) if $|\{x \mid C(x) = 1\}| \leq s/2$.

2.4 Nondeterministic Algorithms

Definition 2.2. Let P be a search problem and R be the binary relation defining P. We say P can be solved by a nondeterministic polynomial-time algorithm if there is a non-deterministic Turing machine M such that for every input x,

- If x has a solution, then $M(x)$ has an accepting computation path, and every accepting path will output a valid solution y, i.e., $R(x, y)$ is true.
- If x has no solution, then $M(x)$ has no accepting computation path.

The class of search problem solvable by nondeterministic polynomial-time algorithms is defined as SearchNP.

We note that some authors call this class FNP (e.g., [BHPT20]), while people also use FNP to denote the search problems defined by a polynomial-time relation, i.e., $R \in \mathsf{P}$ (see, e.g., [Ric08]). To avoid ambiguity, we use SearchNP for the former class, and FNP for the latter one. It is clear that $\mathsf{FNP} \subseteq \mathsf{SearchNP}$.

The following proposition gives a characterization of SearchNP.

Proposition 2.3. $P \in \mathsf{SearchNP}$ *if and only if there is a relation* $\hat{R} \subseteq R$ *satisfying that* $\hat{R} \in \mathsf{NP}$*, and for every* x *with a solution in* P*, there is a* y *such that* $\hat{R}(x, y)$ *is true.*

Proof. ($\Rightarrow$). If $P \in \mathsf{SearchNP}$, then there is a nondeterministic polynomial-time Turing machine that satisfies the definition. Consider the relation $\hat{R}(x, y)$ that is true if and only if there is an accepting path of $M(x)$ that outputs y. Clearly $\hat{R} \in \mathsf{NP}$. For every x with a solution in P, the machine will have an accepting path that outputs a solution y, i.e., $\hat{R}(x, y)$ is true.

($\Leftarrow$). If there is a relation $\hat{R}$ as defined above, we consider the following nondeterministic algorithm: Given input x, we nondeterministically search for a y and a witness w that certifies $\hat{R}(x, y)$; we accept and output y if we find valid y and w. It is easy to verify that the algorithm satisfies Definition 2.2. □

Separations between SearchNP, FP, and FNP. SearchNP is strictly stronger than FP if $\mathsf{P} \neq \mathsf{NP}$: The search of satisfying assignment, i.e., given a formula φ, find an assignment x such that $\varphi(x) = 1$, is in SearchNP but not in FP assuming $\mathsf{P} \neq \mathsf{NP}$. The following example shows that SearchNP should also be a strict superset of FNP.

Proposition 2.4. *If* $\mathsf{P} \neq \mathsf{NP}$*, then there is a total search problem in* $\mathsf{SearchNP} \setminus \mathsf{FNP}$*.*

Proof. Consider the following problem: Given a circuit $C : \{0,1\}^n \to \{0,1\}^{n-1}$, output an $x \in \{0,1\}^n$ such that there exists a $y \in \{0,1\}^n$ such that $y \neq x$ and $C(x) = C(y)$. Clearly this is a total search problem in $\mathsf{SearchNP}$.

Suppose that the problem is in FNP, then the relation defining the problem, i.e., $R(C, x)$ if and only if there exists a $y \neq x$ such that $C(x) = C(y)$, is polynomial-time decidable. We now show that this implies $\mathsf{SAT} \in \mathsf{P}$. Given an n variable Boolean formula φ, we define the following circuit $C_\varphi : \{0,1\}^n \to \{0,1\}^{n-1}$:

$$C_\varphi(x) := \begin{cases} 0^{n-1} & x = 0^n \text{ or } \varphi(x) = 1, \\ 1^{n-1} & \text{otherwise.} \end{cases}$$

We can then decide in polynomial-time whether $R(C_\varphi, x)$ is true. One can see that φ is satisfiable if and only if $R(C_\varphi, 0^n)$ is true or $\varphi(0^n) = 1$, which implies $\mathsf{SAT} \in \mathsf{P}$. □

$\mathsf{SearchNP}$ and $\mathsf{FP}^{\mathsf{NP}}$. The following proposition shows that $\mathsf{FP}^{\mathsf{NP}}$ (i.e., the class of polynomial-time computable function with an NP oracle) is an upper bound of $\mathsf{SearchNP}$.

Proposition 2.5. $\mathsf{SearchNP} \subseteq \mathsf{FP}^{\mathsf{NP}}$.

Proof. We use the characterization of $\mathsf{SearchNP}$ in Proposition 2.3. For a problem $P \in \mathsf{SearchNP}$ and the relation R defining P, there is a relation $\hat{R} \subseteq R$ such that $\hat{R} \in \mathsf{NP}$ and for every x that has a solution in P, there is a y such that $\hat{R}(x, y)$ is true. Given an x that has a solution, we can use an NP oracle to find a solution y such that $\hat{R}(x, y)$ bit by bit, leading to an $\mathsf{FP}^{\mathsf{NP}}$ algorithm for P. □

On the other hand, the following proposition shows that $\mathsf{SearchNP}$ should be a strict subset of $\mathsf{FP}^{\mathsf{NP}}$ unless the polynomial-time hierarchy collapses to the first level.

Proposition 2.6. $\mathsf{SearchNP} \neq \mathsf{FP}^{\mathsf{NP}}$ *unless* $\mathsf{NP} = \mathsf{coNP}$.

Proof. Towards a contradiction, we assume that $\mathsf{SearchNP} = \mathsf{FP}^{\mathsf{NP}}$, we will describe an NP algorithm for the Boolean unsatisfiability problem UNSAT that is known to be coNP-complete.

Note that if we view UNSAT as a search problem with a single-bit output, we will have $\mathsf{UNSAT} \in \mathsf{FP}^{\mathsf{NP}}$. Since $\mathsf{SearchNP} = \mathsf{FP}^{\mathsf{NP}}$, there is a nondeterministic Turing machine M that solves UNSAT, in the sense that

- If the input formula φ is satisfiable, every computation path of $M(\varphi)$ rejects.
- If the input formula φ is unsatisfiable, there is a computation path of $M(\varphi)$ that accepts and outputs the correct answer, namely 1.

Then it is clear that the following NP algorithm solves UNSAT: Given a formula φ, we simulate $M(\varphi)$ and accept if and only if there is an accepting path of M. □

2.5 Witness Encryption

Witness encryption [GGSW13] is a cryptographic primitive that allows the owners of a short proof to a hard statement (say, Riemann Hypothesis), instead of the owners of the secret key, to decrypt the message.

Let $L \in \mathsf{NP}$ and V be a fixed polynomial-time verifier of L with witness of length $q(n) = \mathsf{poly}(n)$, i.e., $x \in L$ if and only if there exists a $w \in \{0,1\}^{q(|x|)}$ such that $V(x, w) = 1$. The *witness encryption* for L with respect to the verifier V consists of two probabilistic polynomial-time (p.p.t.) functions:

- $\mathsf{Enc}(x, b)$: given $x \in \{0,1\}^n$ and $b \in \{0,1\}$, outputs a cipher-text.

- $\mathsf{Dec}(x, \mathsf{ct}, w)$: given $x \in \{0,1\}^n$, a witness $w \in \{0,1\}^{q(n)}$, and a cipher-text ct, decrypts the message.

The witness encryption scheme should satisfy two properties.

- (*Correctness*). If $x \in L$ and w is a witness for x, then for every $b \in \{0,1\}$, $\mathsf{ct} = \mathsf{Enc}(x, b)$, $\mathsf{Dec}(x, \mathsf{ct}, w) = b$ with probability at least $1 - \varepsilon_d$. The parameter ε_d is said to be the *decryption error*. In particular, it is said to have *perfect correctness* if $\varepsilon_d = 0$, i.e., there is no decryption error.
- (*Security*). For every $x \notin L$, $\mathsf{Enc}(x, 0)$ and $\mathsf{Enc}(x, 1)$ are ε-indistinguishable against any size-s adversary. Formally, for every non-uniform adversary A of size s,

$$\left| \Pr[A(1^n, \mathsf{Enc}(x, 0)) = 1] - \Pr[A(1^n, \mathsf{Enc}(x, 1))] \right| \leq \varepsilon.$$

  Here $s = s(n)$ and $\varepsilon = \varepsilon(n)$ are parameters. For instance, we can set $s = n^{\omega(1)}$ and $\varepsilon = 1/n^{\omega(1)}$ to obtain security against any polynomial-time adversary.

Note that witness encryption can also be defined for *promise languages in* NP. Let $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ be a promise language in NP and V be a verifier. A witness encryption scheme for L should satisfy the correctness property for every YES-instance $x \in \Pi^{\mathsf{YES}}$, and the security property should hold for every NO-instance $x \in \Pi^{\mathsf{NO}}$.

Witness encryption is a strong cryptographic primitive and can imply public-key encryption [DH76; GM84], identity-based encryption [Sha84], and attribute-based encryption [SW05]. It is known to construct witness encryption for every NP language from *indistinguishability obfuscation* [Gar+16], and therefore it can be based on "well-founded assumptions" [JLS21; JLS22]. Moreover, there are several candidates [CVW18; Bar+20; BIOW20; VWW22; Tsa22] from non-standard assumptions for certain lattice or coding problems.

2.6 Super-bits and Demi-bits

We need the notion of *super-bits* and *demi-bits* proposed by Rudich [Rud97] as the basic cryptographic primitives against nondeterministic adversaries.

Definition 2.7 (Super-bits)**.** Let n, m be length parameters such that $n < m$. A function $g_n : \{0,1\}^n \to \{0,1\}^m$ is said to be an (s, ε)-secure *super-bits generator* if there is no $\mathsf{NP}_{/\mathsf{poly}}$ adversary Adv of size s such that,

$$\Pr_{y \in \{0,1\}^m}[\mathsf{Adv}(y) = 1] - \Pr_{x \in \{0,1\}^n}[\mathsf{Adv}(g_n(x))] < \varepsilon. \tag{3}$$

The concept of super-bits generators is a natural generalization of standard *pseudorandom generators*, where the adversary is a polynomial-sized circuit, and the subtraction in Equation (3) is taken absolute value. The change in Equation (3) is crucial as when the adversary can make non-deterministic guesses, it can easily distinguish $g_n(\mathcal{U}_n)$ from $\mathcal{U}_m$ by guessing the seed $x \in \{0,1\}^n$ and verify whether the observed string is $g_n(x)$.

Rudich [Rud97] conjectured that for every constant $\varepsilon > 0$, there is a $\mathsf{P}_{/\mathsf{poly}}$ computable super-bits generators that is $(2^{n^\varepsilon}, 2^{n^{-\varepsilon}})$-secure. Moreover, he proposed a candidate super-bits generator based on the subset-sum problem.

Definition 2.8 (Demi-bits)**.** Let n, m be length parameters such that $n < m$. A function $g_n : \{0,1\}^n \to \{0,1\}^m$ is said to be an (s, ε)-secure *demi-bits generator* if there is no $\mathsf{NP}_{/\mathsf{poly}}$ adversary Adv of size s such that,

$$\Pr_{y \in \{0,1\}^m}[\mathsf{Adv}(y) = 1] < \varepsilon \quad \text{and} \quad \Pr_{x \in \{0,1\}^n}[\mathsf{Adv}(g_n(x))] = 0.$$

Demi-bits generators are (intuitively) weaker then super-bits generators as it only promises that there is no non-deterministic adversary that never accepts $\mathsf{Range}(g_n)$ and accepts considerably many strings of length m.

Super-bits and demi-bits generators are closely related to the concept of natural proofs [RR97] (also see the discussion in [Rud97]).

3 Hardness of Explicit Constructions from Witness Encryption

In this section, we explain the technique of [ILW23] that shows the hardness of range avoidance problem from witness encryptions. We then generalize their technique in several dimensions.

- We verify that if the decryption algorithm of the witness encryption can be implemented in a restricted class $\mathscr{C}$, we can also prove the hardness of $\mathscr{C}$-Avoid.
- We show that witness encryption schemes with a decryption error still implies the hardness of *remote point problems*. This is crucial as witness encryption schemes with low decryption complexity (including one scheme of our instantiation) could have a small decryption error.
- We show that if the witness encryption scheme is hard against nondeterminism, which will be formally defined in Section 3.2, we can obtain the hardness of range avoidance problem or remote point problem against nondeterministic search algorithms.

3.1 Hardness of Avoid from Witness Encryption

In this subsection, we demonstrate the recent result of Ilango, Li, and Williams [ILW23] on the conditional hardness of range avoidance problem based on secure witness encryption.

Let $L \in \mathsf{NP}$ be a language, V be a verifier of L with proof length $q(n)$, and $(\mathsf{Enc}, \mathsf{Dec})$ be a perfectly correct witness encryption for L with respect to V. We define the *plain-text avoidance problem* of the witness encryption scheme as follows.

Problem 3.1 (Plain-Text Avoidance Problem)**.** Let n be the input length of L, $q = q(n)$ be the length of witness, and $\ell = \ell(n)$ be a parameter. The *Plain-Text Avoidance Problem*, denoted by $\mathsf{PTAvoid}[n, q, \ell]$, is defined as follows.

Given $x \in \{0,1\}^n$ and a sequence of cipher-texts $\mathsf{ct}_1, \mathsf{ct}_2, \ldots, \mathsf{ct}_\ell$ encrypted on x, i.e., $\mathsf{ct}_i \leftarrow \mathsf{Enc}(x, b_i)$ for some $b_i \in \{0,1\}$. Find a string $y \in \{0,1\}^\ell$ such that for every witness $w \in \{0,1\}^q$, $\mathsf{Dec}(x, \mathsf{ct}, w) \neq y$, where

$$\mathsf{Dec}(x, \mathsf{ct}, w) := \mathsf{Dec}(x, \mathsf{ct}_1, w) \| \mathsf{Dec}(x, \mathsf{ct}_2, w) \| \ldots \| \mathsf{Dec}(x, \mathsf{ct}_\ell, w) \in \{0,1\}^\ell.$$

Note that when $q < \ell$, the plain-text avoidance problem is a special-case of the range avoidance problem. Concretely, given x, ct as input, if we treat $\mathsf{Dec}(x, \mathsf{ct}, \cdot) : \{0,1\}^q \to \{0,1\}^\ell$ as a circuit, then y is a solution to $\mathsf{PTAvoid}$ if and only if $y \notin \mathsf{Range}(\mathsf{Dec}(x, \mathsf{ct}, \cdot))$. Therefore if we can prove the hardness of solving $\mathsf{PTAvoid}[n, q, \ell]$ for certain witness encryption scheme, where $\mathsf{Dec}(x, \mathsf{ct}, \cdot)$ can be implemented in a circuit class $\mathscr{C}$, we can also prove the hardness of $\mathscr{C}$-Avoid.

Theorem 3.2 ([ILW23])**.** *Suppose that $L \in \mathsf{NP}$, V is a verifier for L with witness length $q = q(n)$, and $(\mathsf{Enc}, \mathsf{Dec})$ be a perfectly correct witness encryption scheme for L w.r.t. V that is ε-secure against $\mathsf{poly}(n)$-sized adversary. If $\mathsf{PTAvoid}[n, q, \ell] \in \mathsf{FP}$, $\ell > q$, and $\varepsilon < 2^{-\ell}$, then $L \in \mathsf{coNP}$.*

Specifically, if we choose L such that $L \notin \mathsf{coNP}$ is plausible (e.g. let L be an NP-complete language), then the existence of a secure witness encryption scheme will imply the hardness of $\mathsf{PTAvoid}$ (and thus also the hardness of Avoid). In particular, if we assume indistinguishability

obfuscation [Gar+16; JLS21; JLS22] or any witness encryption scheme for NP [CVW18; Bar+20; BIOW20; VWW22; Tsa22], then $\mathsf{Avoid} \notin \mathsf{FP}$.

The proof of Theorem 3.2 is quite simple and can be demonstrated in a few paragraphs.

Proof of Theorem 3.2. Let A be a polynomial-time algorithm for $\mathsf{PTAvoid}[n, q, \ell]$. We will show that the following NP algorithm B solves $\overline{L}$ (and thus $L \in \mathsf{coNP}$).

1. Given any $x \in \{0,1\}^m$. Non-deterministically guess $b = b_1 \| b_2 \| \dots \| b_\ell \in \{0,1\}^\ell$ and a sequence of strings $r_1, r_2, \dots, r_\ell$ served as the randomness of Enc.
2. Let $\mathsf{ct}_i := \mathsf{Enc}(x, b_i; r_i)$. Then $I = (x, \mathsf{ct}_1, \dots, \mathsf{ct}_\ell)$ is an instance of $\mathsf{PTAvoid}[n, q, \ell]$.
3. Accept if and only if $A(I) = b$.

We now prove that B solves $\overline{L}$.

Soundness. Suppose that $x \notin \overline{L}$, i.e., $x \in L$, we need to prove that $B(x) = 0$. Let $w \in \{0,1\}^q$ be a witness of x with respect to V. For any $b \in \{0,1\}^\ell$ and $r_1, \dots, r_\ell$ as guessed in item 1 of B, let $\mathsf{ct}_i := \mathsf{Enc}(x, b_i; r_i)$ and $\mathsf{ct} = \mathsf{ct}_1 \| \dots \| \mathsf{ct}_\ell$, then by the perfect correctness of the witness encryption scheme, $\mathsf{Dec}(x, \mathsf{ct}_i, w) = b_i$. This further means that $b \notin \mathsf{Range}(\mathsf{Dec}(x, \mathsf{ct}, \cdot))$. Since A solves $\mathsf{PTAvoid}[n, q, \ell]$, we know that $A(I) \neq b$ and thus the algorithm B rejects.

Completeness. Suppose that $x \in \overline{L}$, i.e., $x \notin L$, we need to prove that $B(x) = 1$. In other words, we need to show that there exists an $b \in \{0,1\}^\ell$ and a sequence of strings $r_1, \dots, r_\ell$ as guessed in item 1 of B such that B accepts.

Let $\mathsf{ct}_i := \mathsf{Enc}(x, 0)$ be the encryption of 0 for every $i \in [\ell]$, $\mathsf{ct} := (\mathsf{ct}_1, \dots, \mathsf{ct}_\ell)$, and $I = (x, \mathsf{ct}_1)$. By an averaging argument, there exists a $b^\star \in \{0,1\}^\ell$ such that $A(I) = b^\star$ with probability at least $2^{-\ell}$. We define $\mathsf{ct}_i^\star := \mathsf{Enc}(x, b_i^\star)$, $\mathsf{ct}^\star := (\mathsf{ct}_1^\star, \dots, \mathsf{ct}_\ell^\star)$, and $I^\star = (x, \mathsf{ct}^\star)$ By the security of the witness encryption scheme, we know that I and $I^\star$ are ε-indistinguishable against $\mathsf{poly}(n)$-sized adversary. In particular, we know that

$$\Pr[A(I^\star) = b^\star] \geq \Pr[A(I) = b^\star] - \varepsilon \geq 2^\ell - \varepsilon > 0.$$

Therefore there exists a sequence $r_1^\star, \dots, r_\ell^\star$ of randomness such that for $\mathsf{ct}_i^\star := \mathsf{Enc}(x, b_i^\star; r_i^\star)$, $\mathsf{ct}^\star = \mathsf{ct}_1^\star \| \dots \| \mathsf{ct}_\ell^\star$, and $I^\star = (x, \mathsf{ct}^\star)$, $A(I^\star) = b^\star$. This implies that the algorithm B accepts x. □

3.2 Witness Encryption against Non-Deterministic Adversary

We now introduce the notion of witness encryption secure against *non-deterministic* adversaries, and propose the framework of proving the hardness of explicit constructions with witness encryption of certain hard languages.

Let $L = (L_n^{\mathsf{YES}}, L_n^{\mathsf{NO}}) \subseteq \{0,1\}^n \times \{0,1\}^n$ be a promise language such that there is an efficient proof system P satisfying that:

- **(Completeness).** For every $x \in L_n^{\mathsf{YES}}$, there is a $w \in \{0,1\}^k$ such that $P(x, w) = 1$.
- **(Soundness).** For every $x \in L_n^{\mathsf{NO}}$ and any $w \in \{0,1\}^k$, $P(x, w) = 0$.

The syntax of a *witness encryption* for L consists of an encryption algorithm $\mathsf{Enc}(b \in \{0,1\}, x \in \{0,1\}^n)$ and a decryption algorithm $\mathsf{Dec}(\mathsf{ct}, x \in \{0,1\}^n, w \in \{0,1\}^\ell)$. The correctness property is defined as in Section 2.5. Moreover, it should also satisfy the following security property against non-deterministic adversary.

- **(Security).** The encryption scheme $(\mathsf{Enc}, \mathsf{Dec})$ is said to be (s, ε)-secure (against non-deterministic adversary) if there is a polynomial-time randomized algorithm S (called simulator) such that for every $x \in L_n^{\mathsf{NO}}$, every message $b \in \{0,1\}$, and every s-size non-deterministic

adversary Adv,

$$\Pr[\mathsf{Adv}(S(1^n, x)) = 1] - \Pr[\mathsf{Adv}(\mathsf{Enc}(b, x)) = 1] < \varepsilon.$$

The definition of security is an analogue of the standard simulation paradigm in cryptography: To argue that the encryption $\mathsf{Enc}(b, x)$ does not leak the knowledge of b, for every $b \in \{0, 1\}$, we show that there is a simulator S producing a distribution $\mathcal{D}_{\mathsf{ideal}}$ that is indistinguishable to $\mathcal{D}_{\mathsf{real}} := \mathsf{Enc}(b, x)$ without knowing b.

A crucial difference between our definition and the standard simulation paradigm is that we force the adversary to accept the ideal distribution $\mathcal{D}_{\mathsf{ideal}}$ more often, which, intuitively, prevents the adversary from utilizing of the power of non-determinism by guessing the bit b and the internal randomness of $\mathsf{Enc}(b, x)$ directly. This definition is inspired by Rudich's *super-bits generators* [Rud97] (see Section 2.6 for the definitions).

3.3 Main Lemma

We are now ready to state our main lemma that generalizes Theorem 3.2.

Lemma 3.3. *There are constants $\varepsilon_1, \alpha \in (0, 1/2)$ such that the following holds. Let $n, k, s, t, \varepsilon_2$ be parameters. Suppose that n is sufficiently large. Let $L = (\Pi_n^{\mathsf{YES}}, \Pi_n^{\mathsf{NO}}) \subseteq \{0,1\}^n \times \{0,1\}^n$ be a promise problem that admits a $\mathsf{poly}(n)$-sized proof system P with proof length k. Then there is a $\mathsf{poly}(n)$-sized proof system for $\overline{L} = (\Pi_n^{\mathsf{NO}}, \Pi_n^{\mathsf{YES}})$ assuming the following two conditions:*

1. *There is a $\mathsf{poly}(n)$-sized witness encryption scheme for L with decryption error ε_1 and is $(\mathsf{poly}(t, n, \log(\varepsilon_2^{-1})), k), \varepsilon_2)$-secure against nondeterministic (resp. deterministic) adversary. In addition, for every cipher-text ct and x, $\mathsf{Dec}(\mathsf{ct}, x, \cdot) : \{0,1\}^k \to \{0,1\}$ can be implemented by a $\mathscr{C}$-circuit of size s.*
2. *There is a t-sized non-uniform $\mathsf{SearchNP}$ algorithm (resp. t-sized deterministic circuits) for $\mathscr{C}$-$\mathsf{RPP}[k, \ell, s, \alpha]$, where $\ell = \Omega_{\varepsilon_1,\alpha}\left(\ln(\varepsilon_2^{-1})\right)$.*

Moreover, if the decryption error $\varepsilon_1 = 0$, i.e., the witness encryption is perfectly correct, then item 2 can be replaced by:

2'. *There is a t-sized non-uniform $\mathsf{SearchNP}$ algorithm (resp. t-sized deterministic circuits) for $\mathscr{C}$-$\mathsf{Avoid}[k, \ell, s]$, where $\ell = \Omega\left(\ln(\varepsilon_2^{-1})\right)$.*

Proof. We only prove the general case and leave the "moreover" part and the deterministic case (i.e. the witness encryption is secure against deterministic adversary and the algorithm is deterministic) to the readers.

Suppose that $\varepsilon_1, \alpha, n, k, s, t, \varepsilon_2$ are defined as above and n is sufficiently large. Let $L = (\Pi_n^{\mathsf{YES}}, \Pi_n^{\mathsf{NO}}) \subseteq \{0,1\}^n \times \{0,1\}^n$ be a promise problem that admits a $\mathsf{poly}(n)$-sized proof system P with proof length k. Moreover, we assume that there is a $\mathsf{poly}(n)$-sized witness encryption $(\mathsf{Enc}, \mathsf{Dec})$ for L as described above, and a t-sized non-uniform $\mathsf{SearchNP}$ algorithm A for $\mathscr{C}$-$\mathsf{RPP}[k, \ell, s, \alpha]$, where the output length $\ell = \Omega_{\varepsilon_1,\alpha}\left(\ln(\varepsilon_2^{-1})\right)$ will be determined later.

Suppose that $x \in \{0,1\}^n$, and $b = b_1 \| b_2 \| \dots \| b_\ell \in \{0,1\}^\ell$ be a sequence of bits. Let $\mathsf{ct}_i = \mathsf{Enc}(x, b_i)$ be a random variable for $i \in [\ell]$, and $\mathsf{ct} = (\mathsf{ct}_1, \dots, \mathsf{ct}_\ell)$. Consider the circuit $D_i(\cdot) := \mathsf{Dec}(\mathsf{ct}_i, x, \cdot) : \{0,1\}^k \to \{0,1\}$, and let $D : \{0,1\}^k \to \{0,1\}^\ell$ be the circuit whose i-th output bit is computable by D_i. Clearly, we can regard D as an instance of $\mathscr{C}$-$\mathsf{RPP}[k, \ell, s, \alpha]$, so that $y = A'(D)$ should be α-far from the range of D.

Claim 3.4. *Suppose that $x \in \Pi_n^{\mathsf{YES}}$, then for every $b \in \{0,1\}^\ell$ and $\xi \in (0, 1)$,*

$$\Pr\left[\exists y \in A(D), |y - b| \le (\alpha - (1+\xi)\varepsilon_1)\ell\right] \le \exp\left(-\xi^2 \varepsilon_1 \ell / 3\right).$$

Claim 3.5. *Suppose that $x \in \Pi_n^{\mathsf{NO}}$, then for every $\tau \in (0, 1/2)$, there exists a $b \in \{0,1\}^\ell$ such that*

$$\Pr\left[\exists y \in A(D), |y - b| \le (1/2 - \tau)\ell\right] \ge \exp\left(-5\tau^2\ell\right) - \varepsilon_2\ell.$$

Given these two claims, we can choose $\xi := 1$, $\tau := \sqrt{\varepsilon_1/20}$, and $\alpha := 1/2 + 2\varepsilon_1 - \tau$. We choose ε_1 to be sufficiently small such that $\alpha < 1/2$. Moreover, we let $\ell := \ln\left(\varepsilon_2^{-1}\right)/(20\tau^2)$ so that $\varepsilon_2\ell \ll \exp(-5\tau^2\ell)$ and $\exp(-5\tau^2\ell) - \varepsilon_2\ell > 2\exp(-\varepsilon_1\ell/3)$. Therefore

- For every $x \in \Pi_n^{\mathsf{NO}}$, there exists a $b \in \{0,1\}^\ell$ such that

$$\Pr\left[\exists y \in A(D), |y - b| \le (1/2 - \tau)\ell\right] \ge 2\exp\left(-\varepsilon_1\ell/3\right). \tag{4}$$

- For every $x \in \Pi_n^{\mathsf{YES}}$, for every $b \in \{0,1\}^\ell$,

$$\Pr\left[\exists y \in A(D), |y - b| \le (1/2 - \tau)\ell\right] \le \exp\left(-\varepsilon_1\ell/3\right). \tag{5}$$

Then it can be verified that the following non-deterministic polynomial-time algorithm solves $\overline{L} = (\Pi_n^{\mathsf{NO}}, \Pi_n^{\mathsf{YES}})$. Given any input $x \in \{0,1\}^n$, we non-deterministically guess a $b \in \{0,1\}^\ell$, and use Goldwasser-Sipser protocol (see Theorem 2.1) to accept if Equation (4) holds and reject if Equation (5) holds.

It remains to prove the two claims above.

Proof of Claim 3.4. Suppose that $x \in \Pi_n^{\mathsf{YES}}$, then there exists a witness $w \in \{0,1\}^k$ such that $V(x, w) = 1$. By the correctness of the witness encryption scheme, we know that for every $b_i \in \{0,1\}$, $\mathsf{Dec}(\mathsf{Enc}(x, b_i), x, w) = b_i$ with probability at least $1 - \varepsilon_1$. By Chernoff bound, we know that for every $b \in \{0,1\}^\ell$,

$$\Pr\left[|\mathsf{Dec}(\mathsf{ct}, x, w) - b| \ge (1 + \xi)\varepsilon_1\ell\right] \le \exp\left(-\xi^2\varepsilon_1\ell/3\right).$$

Since for every $y \in A(D)$, y is α-far from the range of D, we know that $|y - \mathsf{Dec}(\mathsf{ct}, x, w)| \ge \alpha$, thus $|y - b| \le (\alpha - (1 + \xi)\varepsilon_1)\ell$ implies that $|\mathsf{Dec}(\mathsf{ct}, x, w) - b| \ge (1 + \xi)\varepsilon_1\ell$ by triangle inequality. Therefore we have

$$\Pr\left[\exists y \in A(D), |y - b| \le (\alpha - (1 + \xi)\varepsilon_1)\ell\right] \le \exp\left(-\xi^2\varepsilon_1\ell/3\right). \qquad \square$$

Proof of Claim 3.5. Let S be the polynomial-time simulater of the witness encryption scheme. Instead of ct, D above, we define the following random variables: For every $i \in [\ell]$, let $\hat{\mathsf{ct}}_i = S(1^n, x)$ be a random variable, and $\hat{\mathsf{ct}} = (\hat{\mathsf{ct}}_1, \ldots, \hat{\mathsf{ct}}_\ell)$. Let $\hat{D}_i(\cdot) := \mathsf{Dec}(\hat{\mathsf{ct}}_i, x, \cdot)$ and $\hat{D} : \{0,1\}^k \to \{0,1\}^\ell$ be the circuit whose i-th output bit is computed by $\hat{D}_i$.

Let $A'(D)$ be the single-valued that output the lexicographic first string $y \in A(D)$. (Note that $A'(D)$ may not be computable in $\mathsf{SearchNP}_{/\mathsf{poly}}$.) We define $\hat{y} = A'(\hat{D}) \in \{0,1\}^\ell$. By a counting argument, we know that there is a $b^\star \in \{0,1\}^\ell$ such that

$$\Pr\left[|\hat{y} - b^\star| \le (1/2 - \tau)\ell\right] \ge 2^{-\ell}\binom{\ell}{(1/2 - \tau)\ell} \ge \exp\left(-5\tau^2\ell\right),$$

where the last inequality follows from Lemma A.1. Therefore we know that

$$\Pr\left[\exists y \in A(\hat{D}), |y - b^\star| \le (1/2 - \tau)\ell\right] \ge \exp\left(-5\tau^2\ell\right). \tag{6}$$

Let $b^\star = b_1^\star \| b_2^\star \| \dots \| b_\ell^\star$. We now define $\mathsf{ct}_i = \mathsf{Enc}(x, b_i^\star)$, $\mathsf{ct} = (\mathsf{ct}_1, \dots, \mathsf{ct}_\ell)$, $D_i(\cdot) = \mathsf{Dec}(\mathsf{ct}_i, x, \cdot)$, and D be the circuit whose i-th output bit is computed by D_i. By the security of the witness encryption scheme, we know that for every non-uniform $\mathsf{SearchNP}$ adversary Adv of size $\mathsf{poly}(t, n, \ell, k)$,

$$\Pr\left[\mathsf{Adv}(\hat{\mathsf{ct}}_i) = 1\right] - \Pr\left[\mathsf{Adv}(\mathsf{ct}_i) = 1\right] < \varepsilon_2.$$

We define $\mathsf{Hyb}_i := (\hat{\mathsf{ct}}_1, \dots, \hat{\mathsf{ct}}_i, \mathsf{ct}_{i+1}, \dots, \mathsf{ct}_\ell)$. Then $\mathsf{Hyb}_0 \equiv \mathsf{ct}$ and $\mathsf{Hyb}_\ell = \hat{\mathsf{ct}}$. By a standard hybrid argument, we know that for every non-uniform $\mathsf{SearchNP}$ adversary of size $\mathsf{poly}(t, n, \ell, k)$,

$$\Pr\left[\mathsf{Adv}(\hat{\mathsf{ct}}) = 1\right] - \Pr\left[\mathsf{Adv}(\mathsf{ct}) = 1\right] < \varepsilon_2 \ell.$$

Combining this with Equation (6), we know that

$$\Pr\left[\exists y \in A(D), |y - b^\star| \le (1/2 - \tau)\ell\right] \ge \exp\left(-5\tau^2 \ell\right) - \varepsilon_2 \ell.$$

(Here, we regard the event in the probability, which is decidable by a non-uniform $\mathsf{SearchNP}$ adversary of size $\mathsf{poly}(t, n, \ell, k)$, as the adversary, and use the indistinguishability of $\hat{\mathsf{ct}}$ and ct above.) □

This completes the proof of the theorem. □

Summary of witness encryption. We can see that the following requirements are needed for proving hardness of range avoidance and remote point problems.

1. *A hard language* $L \in \mathsf{NP}_{/\mathsf{poly}} \setminus \mathsf{coNP}_{/\mathsf{poly}}$. The main lemma shows that if range avoidance (or remote point problem) is easy and the witness encryption is secure, then $L \in \mathsf{coNP}_{/\mathsf{poly}}$. Therefore, we need to design witness encryption for a language that is plausibly not in coNP. In particular, it is sufficient (but not necessary) to have a witness encryption scheme for an NP-complete problem as in [ILW23].
2. *The proof length is succinct.* The input circuit of the range avoidance (or remote point) problem in the main lemma has input length k and output length $\ell = \Omega\left(\ln\left(\varepsilon_2^{-1}\right)\right)$, where k is the proof length and ε_2 is the maximum advantage that can be obtained by s_2-size adversaries to attack the witness encryption scheme. Therefore, we should have $k = o\left(\ln\left(\varepsilon_2^{-1}\right)\right)$ to make it non-trivial. In particular, it is sufficient (but not necessary) to have a witness encryption with flexible security parameters, e.g., to have a $(\mathsf{poly}(\lambda), 2^{-\lambda^{\Omega(1)}})$-secure witness encryption for any security parameter 1^λ given as input.
3. *The decryption algorithm has low circuit complexity.* The circuit classes for the range avoidance problem or remote point problem in the main lemma need to support the decryption of the witness encryption. More precisely, for hardness of $\mathscr{C}$-Avoid or $\mathscr{C}$-RPP, we will need $\mathsf{Dec}(\mathsf{ct}, \cdot)$ (i.e. the decryption algorithm with hardcoded cipher text) to be implemented in $\mathscr{C}$.
4. *Security against nondeterminism is needed for hardness result against* $\mathsf{SearchNP}$. To the best of our knowledge, there is no known generic witness encryption candidate with provable security against nondeterminism from well-founded assumption.
5. *Perfect correctness is needed for hardness of range avoidance problem.* If there is a decryption error, we can only prove the hardness of remote point problem.

4 Witness Encryption Inspired by Public-key Encryption

In this section, we will explain the critical observation that (a type of) public-key encryption is closely related to witness encryption for certain language. Indeed, we will derive our constructions

of witness encryption in the next section by compiling known PKE schemes [Ale11; AD97; Reg09] with this observation.

This observation works for PKE and WE with both standard security and security against nondeterminism. For clarity, we first describe the realm of standard security, and then generalize the results to security against nondeterminism.

4.1 PKE with Pseudorandom Public Key

Recall that a public-key encryption consists of the following functions:

- A *key-generation algorithm* $\mathsf{Gen}(1^\lambda; r) \to (\mathsf{pk}, \mathsf{sk})$ that given the security parameter λ and a random seed r, outputs a pair $(\mathsf{pk}, \mathsf{sk})$ of public-key and secret key.
- An *encryption algorithm* $\mathsf{Enc}(1^\lambda, b, \mathsf{pk}; r) \to \mathsf{ct}$ that given the security parameter λ, a bit $b \in \{0,1\}$, the public-key pk, and a random seed r, outputs the cipher-text ct.
- An *decryption algorithm* $\mathsf{Dec}(1^\lambda, \mathsf{ct}, \mathsf{sk}; r) \to b$ that given the security parameter λ, a cipher-text ct, the secret-key sk, and a random seed r, outputs the plain-text b.

For simplicity, we may omit the security parameter λ and the random seed. The public-key encryption scheme should satisfy the correctness and security properties.

- **(Correctness).** For every $b \in \{0,1\}$, $\mathsf{Dec}(\mathsf{Enc}(b, \mathsf{pk}), \mathsf{sk}) = b$ with probability at least $1 - \varepsilon_d$, where ε_d is called the decryption error.
- **(Security).** For some functions $s = s(\lambda)$ and $\varepsilon = \varepsilon(\lambda)$, $\mathsf{Enc}(0, \mathsf{pk})$ and $\mathsf{Enc}(1, \mathsf{pk})$ are ε-indistinguishable against s-size adversaries.

Besides the standard definition, we introduce the notion of PKE *with pseudorandom public key* ($\widetilde{\text{PKE}}$ for short) that summarizes a standard approach to construct PKE schemes from hardness assumptions, which serves as the key to the connection to witness encryption.

Definition 4.1 ($\widetilde{\text{PKE}}$)**.** A $\widetilde{\text{PKE}}$ scheme, or PKE with pseudorandom public key, is a PKE scheme $(\mathsf{Enc}, \mathsf{Dec}, \mathsf{Gen})$ satisfying the following properties.

- (*Ideal World*). Let $\ell := |\mathsf{pk}|$ be the length of the public key. There is an efficiently samplable distribution $\mathcal{D}$ supported over $\{0,1\}^\ell$ called the distribution of the public key *in ideal world*.
- (*Hardness of PKE Problem*). Let $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$ and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, b)$. The hard problem supporting the PKE scheme, called the *PKE problem* of the scheme, is the task to distinguish between the "ideal world", i.e. $u \leftarrow \mathcal{D}$, and the "real world", i.e. the actual public key pk. This property suggests that these two distributions (i.e. the ideal world and the real world) are ε_1-indistinguishable against s_1-size adversaries, where $\varepsilon_1 = \varepsilon_1(\lambda)$ and $s_1 = s_1(\lambda)$.
- (*Security in Ideal World*). This property shows that the encryption scheme is secure when the public key is sampled from the ideal world distribution $\mathcal{D}$. That is, $\mathsf{Enc}(\mathcal{D}, 0)$ and $\mathsf{Enc}(\mathcal{D}, 1)$ are ε_2-indistinguishable against s_2-size adversaries, where $\varepsilon_2 = \varepsilon_2(\lambda)$ and $s_2 = s_2(\lambda)$.
- (*Verifiable Public Key*). There is an NP proof system that verifies the public key with the secret key. Formally, it accepts pk if and only if the witness is a possible secret-key corresponding to pk.

The following proposition shows that a $\widetilde{\text{PKE}}$ is a secure PKE.

Proposition 4.2. *A $\widetilde{\text{PKE}}$ scheme with parameters $(\varepsilon_1, s_1, \varepsilon_2, s_2)$ is $(2\varepsilon_1 + \varepsilon_2)$-secure against $\min(s_1, s_2) - \mathsf{poly}(\lambda)$ size adversaries.*

Proof. We use a hybrid argument. Let $\ell := |\mathsf{pk}|$ be the length of the public key.

- $\mathsf{Hyb}_0 := \mathsf{Enc}(\mathsf{pk}, 0)$.
- $\mathsf{Hyb}_1 := \mathsf{Enc}(\mathcal{D}, 0)$. By "hardness of PKE problem", we know that the distribution $\mathsf{pk} \leftarrow \mathsf{Gen}$ is ε_1-indistinguishable to $\mathcal{D}$ against s_1 size adversaries. Therefore, Hyb_0 is ε_1-indistinguishable to Hyb_1 against $s_1 - \mathsf{poly}(\lambda)$ size adversaries.
- $\mathsf{Hyb}_2 := \mathsf{Enc}(\mathcal{D}, 1)$. By "security in ideal world", Hyb_1 is s_2-indistinguishable to Hyb_2 against s_2-size adversaries.
- $\mathsf{Hyb}_3 := \mathsf{Enc}(\mathsf{pk}, 1)$. Similar to the indistinguishability of Hyb_0 and Hyb_1, we know that Hyb_2 is ε_1-indistinguishable to Hyb_3 against $s_1 - \mathsf{poly}(\lambda)$ size adversaries.

Therefore, $\mathsf{Hyb}_0 = \mathsf{Enc}(\mathsf{pk}, 0)$ is $(2\varepsilon_1+\varepsilon_2)$-indistinguishable to $\mathsf{Hyb}_3 := \mathsf{Enc}(\mathsf{pk}, 1)$ against $\min(s_1, s_2) - \mathsf{poly}(\lambda)$ size adversaries. This satisfies the security property of the PKE scheme. □

The defining feature of $\widetilde{\text{PKE}}$ is that there are two computationally indistinguishable distributions for the public keys, the *real* distribution and the *ideal* distribution, such that the encryption using ideal public keys are secure though it may not be decryptable. This is similar to the notion of *meaningful/meaningless encryption* [KN08] and *dual-mode cryptosystem* [PVW08] where the encryption is statistically secure in ideal world. The following examples show that several standard PKE schemes are indeed $\widetilde{\text{PKE}}$.

Example 4.3 (Goldwasser-Micali PKE [GM84])**.** The Goldwasser-Micali public-key encryption scheme [GM84] is based on the hardness of Quadratic Residuosity modulo RSA primes.

- (*Key generation*). Randomly pick two distinct large primes p, q and an x that is a quadratic non-residue modulo both p and q. Let $N := pq$. Then the public key is (N, x), and the secret key is (p, q).
- (*Encryption*). To encrypt a bit $b \in \{0, 1\}$, we random pick a number $y \in \mathbb{Z}_N^\times$, and output $\mathsf{ct} := y^2 \cdot x^m \bmod N$.
- (*Decryption*). To decrypt a cipher-text ct, we check whether ct is a quadratic residue modulo p using the Euler's criterion. The plain-text is 1 if and only if ct is a quadratic non-residue modulo p.

We now verify that this PKE scheme is plausibly a $\widetilde{\text{PKE}}$ scheme.

- (*Ideal World*). The ideal world distribution $\mathcal{D}$ for the public key is as follows: We first generate an $N = pq$ as in the key generation of Goldwasser-Micali, and then pick x from the uniform distribution over the quadratic residues modulo N, i.e., $x = y^2 \bmod N$ for a uniformly random $y \in \mathbb{Z}_N^\times$.
- (*Hardness of PKE Problem*). This property requires the ideal world (i.e. the distribution $\mathcal{D}$) and the real world (i.e. a random quadratic non-residue modulo both p and q) to be computationally indistinguishable, which can be based on the worst-case hardness of quadratic residuosity since it is random self-reducible.
- (*Security in Ideal World*). Let $\mathsf{pk} := (x, N)$ be the public key in ideal world, that is, x is a random quadratic residue modulo N. Then it is easy to see that $\mathsf{Enc}(\mathsf{pk}, 0)$ and $\mathsf{Enc}(\mathsf{pk}, 1)$ are identical distributions, as both of them are the uniform distribution over quadratic residues modulo N.
- (*Verifiable Public Key*). It is clear that we can check whether (x, N) is a valid public-key with the factorization of N using the Euler's criterion for quadratic residuosity.

Example 4.4 (Regev's lattice-based PKE [Reg09])**.** Let n be the security parameter, $q = \mathsf{poly}(n)$ be a modulus, $m = O(n \log q)$, and χ be the error distribution in [Reg09].

- (*Key Generation*). Let $s \leftarrow \mathcal{U}(\mathbb{Z}_q^n)$, $A \in \mathbb{Z}_q^{n \times m}$, and $b = sA + e \bmod q$, where $e \leftarrow \chi$. The private key is s, and the public key is (A, b).
- (*Encryption*). To encrypt a bit $m \in \{0,1\}$, we randomly select a 01-vector $x \in \{0,1\}^m$ and compute the cipher-text $\mathsf{ct} = (Ax, \langle b, x \rangle + m \cdot (q/2))$. Note that the matrix-vector multiplication and the inner product refers to the operations in $\mathbb{Z}_q$.
- (*Decryption*). To decrypt a cipher-text $\mathsf{ct} = (u, c)$, we compute $t := \langle s, u \rangle + c \bmod q$. The encrypted bit is 1 if and only if t is closer to $q/2$ than to 0.

We now verify that this PKE scheme is plausibly a $\widetilde{\text{PKE}}$ scheme.

- (*Ideal World*). The ideal world distribution $\mathcal{D}$ for the public key is simply the uniform distribution $\mathcal{U}(\mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m)$.
- (*Hardness of PKE Problem*). The indistinguishability of the ideal world (i.e. the uniform distribution $\mathcal{U}(\mathbb{Z}_q^{n \times m} \times \mathbb{Z}_q^m)$) and the real world (i.e. the distribution $(A, sA + e)$ where $A \leftarrow \mathcal{U}(\mathbb{Z}_q^{n \times m}), e \leftarrow \chi, s \leftarrow \mathcal{U}(\mathbb{Z}_q^n)$) is indeed the Learning-with-Error Problem (aka. the LWE Problem), which is one of the central hard problems in lattice-based cryptography.
- (*Security in the Ideal World*). Let $\mathsf{pk} := (A, b)$ be the public key in ideal world, that is, $A \leftarrow \mathcal{U}(\mathbb{Z}_q^{n \times m})$ and $b \leftarrow \mathcal{U}(\mathbb{Z}_q^{n \times m})$. Then one can prove that $(u_0, c_0) \leftarrow \mathsf{Enc}(\mathsf{pk}, 0)$ and $(u_1, c_1) \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$ are statistically close by the leftover hash lemma (see, e.g., [BBD09; PVW08]).
- (*Verifiable Public Key*). It is easy to verify that we can check whether (A, b) is a valid public key with the secret key s.

4.2 Compiling $\widetilde{\text{PKE}}$ to WE

Now we explain the construction of witness encryption for a special hard problem from public-key encryption with pseudorandom public key.

The intuition of the translation is quite straightforward. We will construct a witness encryption for the *PKE problem*, i.e., the average-case problem of distinguishing the "ideal world" (uniformly random) and the "real world" (public key in the scheme). Note that the PKE problem admits a proof system since a $\widetilde{\text{PKE}}$ scheme has a verifiable public key. Then the decryption of the witness encryption is simply the decryption of the PKE scheme, and the security is implied by the "security in ideal world" of the PKE scheme.

Let $\Gamma = (\mathsf{Enc}, \mathsf{Gen}, \mathsf{Dec})$ be a $\widetilde{\text{PKE}}$ scheme. To obtain a secure witness encryption scheme, we need the following stronger property to hold.

- (*Adaptive Security in Ideal World*). Let $\ell := |\mathsf{pk}|$ be the length of the public key. Then for $u \leftarrow \mathcal{D}$, with probability at least $\gamma = \gamma(\ell)$, $\mathsf{Enc}(u, 0)$ will be ε_2-indistinguishable to $\mathsf{Enc}(u, 1)$ against s_2-size adversaries, where $\varepsilon_2 = \varepsilon_2(\lambda)$ and $s_2 = s_2(\lambda)$.

This property states that the adversary cannot break the PKE scheme even if it is adaptive, in the sense that it can choose the attacking algorithm *after observing the public key*. The choosing phase of the adversary can be *computationally unbounded*.

At first glance, the adaptive security seems to be quite strong. In particular, adaptive security in real world is *impossible*: if the public key pk are generated following the scheme instead of from the ideal world distribution, the adversary can guess the secret key sk in the choosing phase and hard-wire it in the attacking algorithm, so it can decrypt the message and break the security property. Moreover, the encryption algorithm must be probabilistic since otherwise the adversary can hard-wire the cipher-text. Nevertheless, we will later show that several known PKE schemes are plausibly secure in this strong sense.

Theorem 4.5. *Let $\Gamma = (\mathsf{Enc}, \mathsf{Gen}, \mathsf{Dec})$ be a $\widetilde{PKE}$ scheme with adaptive security in ideal world, where $\varepsilon_1, \varepsilon_2, s_1, s_2, \gamma$ are parameters. Then there is a promise language $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}}) \in \mathsf{NP}$ satisfying the following properties:*

- (YES-instances of L contains valid public keys). $\Pi^{\mathsf{YES}} := \{\mathsf{pk} \mid \mathsf{pk} \leftarrow \mathsf{Gen}\}$.
- (Hardness of L). $L \notin \mathsf{SIZE}[s_1(\lambda)]$ *if* $\gamma(\ell) > \varepsilon_1(\lambda)$.
- (Witness Encryption). *Let V be the verifier of L using the verifiable public key property of Γ. Then L admits an (s_2, ε_2)-secure witness encryption scheme with respsect to V.*

Proof. Let ℓ be the length of the public key and $\mathcal{D}$ be the ideal world distribution for the public key. We define $\Pi^{\mathsf{YES}} := \{\mathsf{pk} \mid \mathsf{pk} \leftarrow \mathsf{Gen}\}$ to be the set of valid public keys, and Π^{NO} to be the set of strings u on the support of $\mathcal{D}$ such that $\mathsf{Enc}(u, 0)$ is ε_2-indistinguishable to $\mathsf{Enc}(u, 1)$ against s_2-size adversaries. Note that $\Pi^{\mathsf{NO}} \cap \Pi^{\mathsf{YES}} = \varnothing$, since for every string $u \in \Pi^{\mathsf{YES}}$, we can always distinguish $\mathsf{Enc}(u, 0)$ and $\mathsf{Enc}(u, 1)$ by decrypting the message with the secret key corresponding to u. By the adaptive security in ideal world, we know that $\Pr\left[u \in \Pi^{\mathsf{NO}}\right] \geq \gamma$ for $u \leftarrow \mathcal{D}$. Let $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}}) \in \mathsf{NP}$.

Hardness of L. Suppose that $\gamma(\ell) > \varepsilon_1(\lambda)$ and $L \in \mathsf{SIZE}[s_1]$. Let $C : \{0,1\}^\ell \to \{0,1\}$ be the $s_1(\lambda)$-size circuit that solves L. It is easy to verify that C solves the PKE problem of Γ with advantage γ, which is impossible since the PKE problem is ε_1-hard against s_1-size adversaries.

- Given a public key $u \leftarrow \mathsf{Gen}$, $C(x) = 1$ with probability 1.
- Given a random string $u \leftarrow \mathcal{D}$ from the ideal world distribution, $C(u) = 0$ as long as $u \in \Pi^{\mathsf{NO}}$. It implies that $\Pr[C(u) = 1] \leq 1 - \gamma$.

Witness Encryption for L. Let V be the verifier of L using the verifiable public key property of Γ. We will construct a witness encryption scheme for L with respect to V.

- (*Encryption*). The encryption algorithm simulates the encryption algorithm of the original $\widetilde{\text{PKE}}$ scheme. That is, given any string $u \in \{0,1\}^\ell$ to be encrypted on and any message $b \in \{0,1\}$, the cipher-text is $\mathsf{Enc}(u, b)$.
- (*Decryption*). The decryption algorithm simulates the decryption algorithm of the original $\widetilde{\text{PKE}}$ scheme. That is, given any cipher-text ct, a string $u \in L$, and a witness w of u, it outputs $\mathsf{Dec}(\mathsf{ct}, w)$.

The correctness of the witness encryption scheme follows from the correctness of Γ. If $u \in L$ and w is the witness of u, by the definition of L, we know that (u, w) is a pair of valid public key and secret key. Therefore $\mathsf{Dec}(\mathsf{Enc}(u, b), w) = b$.

The security of the witness encryption scheme has been encoded in the definition of L. For every $u \in \Pi^{\mathsf{NO}}$, we know that $\mathsf{Enc}(u, 0)$ and $\mathsf{Enc}(u, 1)$ are ε_2-indistinguishable against s_2-size adversaries, which is exactly the (s_2, ε_2)-security of the witness encryption scheme. □

Remark 4.6. We can see that the encryption and decryption algorithms are exactly the algorithms of Γ. Therefore, if the decryption of the original $\widetilde{\text{PKE}}$ scheme can be implemented by $\mathscr{C}$ circuits, then the decryption of the new witness encryption scheme can also be implemented by $\mathscr{C}$ circuits.

Moreover, one can verify that the construction also works when the $\widetilde{\text{PKE}}$ scheme has a decryption error. Formally speaking, the witness encryption has ε_d decryption error as long as the $\widetilde{\text{PKE}}$ scheme is decryptable for every public key with decryption error ε_d, in the sense that for *every* valid key $(\mathsf{pk}, \mathsf{sk})$ (instead of a random $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$) and *every* $b \in \{0,1\}$, $\Pr[\mathsf{Dec}(\mathsf{Enc}(\mathsf{pk}, b), \mathsf{sk}) = b] \geq 1 - \varepsilon_d$.

4.3 Generalization to Security against Nondeterminism

In this subsection, we generalize the results to construct witness encryption for hard languages against $\mathsf{coNP}_{/\mathsf{poly}}$ that is secure against nondeterministic adversaries.

4.3.1 Witness Encryption for Harder Languages

Recall that to prove the hardness of range avoidance problem and remote point problem, we need to construct witness encryption for languages not in $\mathsf{coNP}_{/\mathsf{poly}}$ (see Section 3.3). To achive this, we will need a stronger hardness property for the PKE problem.

- (*Demi-Hardness of PKE Problem*). We define the function $g(r) := pk$ for $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(r)$, where $g : \{0,1\}^k \to \{0,1\}^\ell$. The property suggests that for every $\mathsf{NP}_{/\mathsf{poly}}$ adversary Adv of size s_1,
$$\Pr_{y \leftarrow \mathcal{D}}[\mathsf{Adv}(y) = 1] > \varepsilon \text{ and } \Pr_{x \leftarrow \{0,1\}^k}[\mathsf{Adv}(g(x))] = 0.$$

In other words, there is no s_1-size proof system that rejects every real public key and accepts an ideal public key with probability at least ε. In particular, if $\mathcal{D}$ is the uniform distribution, this property requires g to be a *demi-bits generator*.

Theorem 4.7. *Suppose that the $\widetilde{PKE}$ scheme in Theorem 4.5 has demi-hardness with parameters (s_1, ε_1) in addition to standard hardness, then the language L, which satisfies the properties in Theorem 4.5, will also satisfy the following property.*

- (Hardness of L against nondeterminism). *$L \notin \mathsf{coNSIZE}[s_1(\lambda)]$ if $\gamma(\ell) > \varepsilon_1(\lambda)$.*

Proof. Let ℓ be the length of the public key and $\mathcal{D}$ be the ideal world distribution for the public key. Recall that $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ is defined as follows:

- $\Pi^{\mathsf{YES}} := \{\mathsf{pk} \mid \mathsf{pk} \leftarrow \mathsf{Gen}\}$ is the set of valid public keys;
- Π^{NO} is the set of strings u on the support of $\mathcal{D}$ such that $\mathsf{Enc}(u, 0)$ is ε_2-indistinguishable to $\mathsf{Enc}(u, 1)$ against s_2-size adversaries.

We suppose, towards a contradiction, that $L \in \mathsf{coNSIZE}[s_1(\lambda)]$ and $\gamma(\ell) > \varepsilon_1(\lambda)$. Then there is a $s_1(\lambda)$-size proof system that rejects every string in Π^{YES} and accepts every string in Π^{NO}. By the adaptive security of the $\widetilde{\mathrm{PKE}}$ scheme in ideal world, we know that $\Pr\left[u \in \Pi^{\mathsf{NO}}\right] \geq \gamma(\lambda)$ for $u \leftarrow \mathcal{D}$. This means that the proof system rejects every public key and accepts $u \leftarrow \mathcal{D}$ with probability at least $\gamma(\ell) > \varepsilon_1(\lambda)$, which leads to a contradiction to the demi-hardness of PKE problem. □

4.3.2 Witness Encryption against Nondeterminism

To prove the hardness of range avoidance and remote point problem against nondeterministic algorithms, we will need a witness encryption that is secure against nondeterminism. We will need the following stronger security property of the $\widetilde{\mathrm{PKE}}$ scheme.

- (*Adaptive Security in Ideal World against Nondeterminism*). Let $\ell := |\mathsf{pk}|$ be the length of the public key, and $\gamma = \gamma(\ell), s_2 = s_2(\lambda), \varepsilon_2 = \varepsilon_2(\lambda)$ be security parameters. This property means that there is a polynomial-time randomzied algorithm S (called simulator) such that for $u \leftarrow \mathcal{D}$, the following holds with probability at least γ:
  - For every message $b \in \{0,1\}$, and every s_2-size nondeterministic adversary Adv,
$$\Pr\left[\mathsf{Adv}(S(1^\lambda, u))\right] - \Pr[\mathsf{Adv}(\mathsf{Enc}(u, b))] < \varepsilon_2.$$

Intuitively, the property shows that there is no adaptive nondeterministic adversary that accepts the simulated distribution more often than the cipher texts. By adaptive nondeterministic adversary, we mean that the security game is as follows:

- We sample an ideal public key $u \leftarrow \mathcal{D}$.
- A *computationally unbounded* adversary chooses a $b \in \{0,1\}$ as well as a nondeterministic circuit Adv that aims to distinguish the simulated distribution and the cipher text (encrypted using an ideal public key).
- The adversary wins if the nondeterministic circuit Adv accepts the simulated distribution $S(1^\lambda, u)$ more often than the cipher texts $\mathsf{Enc}(u, b)$.

In particular, if the simulator simply outputs a uniformly random string, this property suggests that $\mathsf{Enc}(u, b; \cdot)$, whose input and output are the random seed and the cipher text, respectively, is a *super-bits generator*.

Theorem 4.8. *Suppose that the $\widetilde{PKE}$ scheme in Theorem 4.5 satisfies (s_2, ε_2) adaptive security in ideal world against nondeterminism, then we can define a language L that admits an (s_2, ε_2)-secure witness encryption against nondeterminism and satisfies the properties in Theorem 4.5.*

Proof. Let ℓ be the length of the public key and $\mathcal{D}$ be the ideal world distribution for the public key. Similar to Theorem 4.5, we define $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ as follows.

- $\Pi^{\mathsf{YES}} := \{\mathsf{pk} \mid \mathsf{pk} \leftarrow \mathsf{Gen}\}$ is the set of valid public keys;
- Π^{NO} is the set of strings u on the support of $\mathcal{D}$ such that for every $b \in \{0,1\}$ and every s_2-size nondeterministic adversary Adv,

$$\Pr\Big[\mathsf{Adv}(S(1^\lambda, u))\Big] - \Pr[\mathsf{Adv}(\mathsf{Enc}(u, b))] < \varepsilon_2,$$

  where S is the simulator.

We will only show that L admits witness encryption secure against nondeterminism, and it is straightforward to verify that the other properties in Theorem 4.5 hold.

Let V be the verifier of L using the verifiable public key property of Γ. Consider the following witness encryption scheme for L with respect to V.

- (*Encryption*). The encryption algorithm simulates the encryption algorithm of the original $\widetilde{\text{PKE}}$ scheme. That is, given any string $u \in \{0,1\}^\ell$ to be encrypted on and any message $b \in \{0,1\}$, the cipher-text is $\mathsf{Enc}(u, b)$.
- (*Decryption*). The decryption algorithm simulates the decryption algorithm of the original $\widetilde{\text{PKE}}$ scheme. That is, given any cipher-text ct, a string $u \in L$, and a witness w of u, it outputs $\mathsf{Dec}(\mathsf{ct}, w)$.

The correctness of the witness encryption follows directly from the correctness of the original $\widetilde{\text{PKE}}$ scheme. We can also see that the security is encoded in the definition of the NO-instances, where the simulator S serves as the simulator in the security definition of witness encryption against nondeterminism (see Section 3.2). □

5 Construction of Witness Encryption

In this section, we provide constructions of witness encryption for some special language in NP but plausibly not in coNP. By plugging these constructions into the framework in Section 3, we can show the hardness of range avoidance problem and remote point problem.

5.1 Public-key Encryption

Based on the discussion in Section 3 and Section 4, to show hardness results for the range avoidance and remote point problems, it suffices to construct a $\widetilde{\mathsf{PKE}}$ scheme with *adpative security in ideal world against nondeterministic adversaries*, *demi-hardness of its PKE problem*, and satisfying the following technical properties:

- The decryption algorithm should be of low circuit complexity, and should be perfectly correct for hardness of the range avoidance problem.
- The secret key should be succinct compared to the security level of the $\widetilde{\mathsf{PKE}}$ scheme *in ideal world*. More formally, let n be the length of the secret key, no $\mathsf{poly}(n)$-size (nondeterministic) adversary can break the encryption scheme in ideal world with advantage ε, where $\varepsilon \ll 2^{-n}$.

In this subsection, we outline two candidates under Assumption 1.7 and Assumption 1.9 that lead to Theorem 1.11 and Theorem 1.12, respectively. The exact parameters and proofs will be presented in remaining parts of the section.

$\widetilde{\mathsf{PKE}}$ from random linear codes. We consider the following $\widetilde{\mathsf{PKE}}$ scheme based on Alekhnovich's PKE [Ale11]. Let n be the security parameter, $m = O(n)$, $\kappa = m^{0.1}$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, and $w = \Theta(m/\kappa^2)$. Let $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be any standard map such that $\mathsf{Ext}(x)$ is κ-sparse for every $x \in \{0,1\}^k$. We identify $\{0,1\}$ as $\mathsf{GF}(2)$ and all additions and multiplications in encryption and decryption algorithms are done in $\mathsf{GF}(2)$.

- (*Key generation*). The key generations algorithm uniformly samples $A \leftarrow \mathcal{U}(\{0,1\}^{n\times m})$ and $x \leftarrow \{0,1\}^k$. It outputs $(A, A \cdot \mathsf{Ext}(x)) \in \{0,1\}^{n\times m} \times \{0,1\}^n$ as the public key and x as the secret key.
- (*"Ideal" public keys*). The ideal public key distribution $\mathcal{D}$ is $\mathcal{U}(\{0,1\}^{n\times m} \times \{0,1\}^n)$.
- (*Encryption*). Given a bit $b \in \{0,1\}$ and a public key $(A, v) \in \{0,1\}^{n\times m} \times \{0,1\}^n$, the encryption algorithm randomly samples an $s \in \{0,1\}^n$ and a uniformly random error vector $e \in \{0,1\}^m$ such that $|e| = w$. The cipher text will be $(sA + e, \langle s, v\rangle + b) \in \{0,1\}^m \times \{0,1\}$.
- (*Decryption*). Given a cipher text $(u, c) \in \{0,1\}^m \times \{0,1\}$ corresponding to the secret key $x \in \{0,1\}^k$, the decryption algorithm outputs $\langle u, \mathsf{Ext}(k)\rangle + c$.

Intuitively, the difference between our construction and Alekhnovich's original PKE scheme is that we use a noise distribution for encryption with higher entropy to satisfy the technical property for the succinctness of the secret key. The drawback is that we will need two separate assumptions instead of only one: an LPN assumption (see the first bullet of Assumption 1.9) for the security in ideal world and a GapNCP assumption (see the second bullet of Assumption 1.9) for the hardness of the PKE problem. Details will be given in Section 5.2.

$\widetilde{\mathsf{PKE}}$ from lattices. Similarly, we consider the following variant of the dual Regev scheme [Reg09; GPV08] that has slightly higher decryption circuit complexity but achieves perfect decryption correctness. Let n be the security parameter, $q = n^{O(1)}$ be a modulus, and $m = O(n \log q)$, $\kappa = m^{0.1}$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, and $w = m/10\kappa$. Let $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be any standard map such that $\mathsf{Ext}(x)$ is κ-sparse for every $x \in \{0,1\}^k$. Recall that the ℓ_∞-norm of a vector $v \in \mathbb{Z}_q^d$, denoted by $\|v\|_\infty$ (or simply $\|v\|$ if there is no ambiguity), is defined as $\max_{i\in[d]}\{\min(v_i, q - v_i)\}$. All additions and multiplications are done in $\mathbb{Z}_q$.

- (*Key generation*). The key generations algorithm uniformly samples $A \leftarrow \mathcal{U}(\{0,1\}^{n\times m})$ and $x \leftarrow \{0,1\}^k$. It outputs $(A, A \cdot \mathsf{Ext}(x)) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n$ as the public key and x as the secret key.
- (*"Ideal" public keys*). The ideal public key distribution $\mathcal{D}$ is $\mathcal{U}(\mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^n)$.

- (*Encryption*). Given a bit $b \in \{0,1\}$ and a public key $(A, v) \in \{0,1\}^{n\times m} \times \{0,1\}^n$, the encryption algorithm randomly samples an $s \in \mathbb{Z}_q^n$ and a uniformly random error vector $e \in \mathbb{Z}_q^m$ such that $\|e\| = w$. The cipher text will be $(sA + e, \langle s, v\rangle + b \cdot \lfloor q/2 \rfloor) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$.
- (*Decryption*). Given a cipher text $(u, c) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$ corresponding to the secret key $x \in \{0,1\}^k$, the decryption algorithm outputs 1 if and only if $\langle u, \mathsf{Ext}(k)\rangle + c \in [q/3, 2q/3]$.

Similar to the $\widetilde{\mathrm{PKE}}$ scheme from random linear codes, the error distribution for encryption has higher entropy compared to the original dual Regev scheme. This resolves the technical issue about the succinctness of the secret key, and also makes it (plausibly) secure against known $\mathsf{NP} \cap \mathsf{coNP}_{/\mathsf{poly}}$ approximation algorithms for lattice problems [GG98; AR05] (see Section 5.4 for more details).

5.2 Candidate from Random Linear Codes

We are now ready to propose our candidate of witness encryption secure against nondeterministic adversaries. The first candidate is powered by hard problems related to linear codes, which is inspired by *public-key encryptions* based on LPN [Ale11] (also see the survey [Bar17]).

Theorem 5.1. *Let $k < n < m$ and $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be a function such that $\mathsf{Ext}(x)$ is κ-sparse. For every matrix $A \in \{0,1\}^{n\times m}$, suppose that we define:*

- $\Pi^{\mathsf{YES}} := \{A \cdot \mathsf{Ext}(x) \in \{0,1\}^n \mid x \in \{0,1\}^k\}$.
- *Let $f_v(s, e) := (sA + e, \langle v, s\rangle)$ be a function, where $s \in \{0,1\}^n$ and $e \in \{0,1\}^m$ is a string with Hamming weight w. We define Π^{NO} be the set of $v \notin \Pi^{\mathsf{YES}}$ such that $f_v(s, e)$ is an (s_2, ε)-secure pseudorandom generator.*
- *Let $V(y \in \{0,1\}^n, x \in \{0,1\}^k)$ be the standard verifier of $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ that accepts if and only if $y = A \cdot \mathsf{Ext}(x)$.*

Then there is a witness encryption scheme for $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ w.r.t. V that is (s_2, ε)-secure with decryption error $\kappa w/(m - w)$.

Moreover, if we define Π^{NO} be the set of $v \notin \Pi^{\mathsf{YES}}$ such that $f_v(s, e)$ is an (s_2, ε)-secure super-bits generator, then there is a witness encryption scheme for $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ w.r.t. V that is (s_2, ε)-secure against any nondeterministic adversary *with decryption error $\kappa w/(m - w)$.*

We note that the witness encryption scheme as it is defined is *unconditionally secure* because the assumption is encoded in the definition of the NO instance. It could be a trivial scheme as we cannot rule out the case that $\Pi^{\mathsf{NO}} = \varnothing$. Indeed, we will set the parameters such that it is plausible that Π^{NO} is sufficiently large, and state this as an assumption of our hardness result for explicit constructions.

Construction. Let $g(x) := A \cdot \mathsf{Ext}(x)^\top$. The construction is as follows. Suppose that $v \in \{0,1\}^n$ and we want to encrypt on the statement $v \in \mathsf{Range}(g)$.

- **(Encryption).** Given any message bit $b \in \{0,1\}$, the encryption algorithm samples a random $s \in \{0,1\}^n$ and a random $e \in \{0,1\}^m$ with Hamming weight w, and outputs $\mathsf{Enc}(b, v; s, e) = (sA + e, \langle v, s\rangle + b)$.
- **(Decryption).** Given a witness $x \in \{0,1\}^k$ such that $\mathsf{Ext}(x) = v$, and a cipher text $\mathsf{ct} = (u, c) \in \{0,1\}^m \times \{0,1\}$, the decryption algorithm $\mathsf{Dec}(\mathsf{ct}, v, x)$ outputs $\langle u, \mathsf{Ext}(x)\rangle + c$.

The correctness and security of the construction is given in the following two lemmas, respectively.

Lemma 5.2. *If $v \in \mathsf{Range}(g)$, the decryption algorithm outputs correctly with probability at least $1 - \kappa w/(m - w)$ given any witness x such that $v = g(x)$.*

Proof. Given any witness x such that $v = g(x) = A \cdot \mathsf{Ext}(x)^\top$, we can see that for every $b \in \{0,1\}$,

$$\begin{aligned}
&\Pr_{s,e}\left[\mathsf{Dec}(\mathsf{Enc}(b, v; s, e), v, x) = b\right] \\
=\ &\Pr_{s,e}\left[\langle sA + e, \mathsf{Ext}(x)\rangle + \langle v, s\rangle + b = b\right] \\
=\ &\Pr_{s,e}\left[\langle sA, \mathsf{Ext}(x)\rangle + \langle e, \mathsf{Ext}(x)\rangle + sA \cdot \mathsf{Ext}(x)^\top + b = b\right] \\
=\ &\Pr_{s,e}\left[\langle e, \mathsf{Ext}(x)\rangle = 0\right].
\end{aligned}$$

Since $\mathsf{Ext}(x)$ is κ-sparse and $e \in \{0,1\}^m$ is a random string with Hamming weight w, we know that $\Pr[\langle e, \mathsf{Ext}(x)\rangle = 0]$ is at least

$$\binom{m-\kappa}{w} \Big/ \binom{m}{w} \geq \left(1 - \frac{\kappa}{m-w}\right)^w \geq 1 - \frac{\kappa w}{m-w}. \qquad \square$$

Lemma 5.3. *Under the assumptions in Theorem 5.1, the encryption algorithm is (s_2, ε)-secure. If the "moreover" part holds, then the encryption algorithm is (s_2, ε)-secure against nondeterministic adversaries.*

Proof. We only prove the security against nondeterministic adversaries and leave the deterministic case to the readers. To prove the security, we need to show that for every $v \in \Pi^{\mathsf{NO}}$, there is an efficient simulator S such that for every $b \in \{0,1\}$ and any non-deterministic adversary Adv of size s_2,

$$\Pr[\mathsf{Adv}(S(1^n)) = 1] - \Pr[\mathsf{Adv}(sA + e, \langle v, s\rangle + b) = 1] < \varepsilon. \tag{7}$$

Here we define $S(1^n)$ to output a uniformly random $(u, c) \in \{0,1\}^m \times \{0,1\}$. It is easy to see that eq. (7) follows directly from the definition of Π^{NO} in Theorem 5.1. (One can break the super-bits generator by plugging in the adversary Adv and b that breaks eq. (7).) $\square$

Remark 5.4. We can see that the decryption circuit $\mathsf{Dec}(\mathsf{ct}, v, \cdot)$ of the witness encryption scheme is quite simple: given a witness $x \in \{0,1\}^k$, the message is $\langle v, \mathsf{Ext}(x)\rangle + c$.

For instance, we can implement $\mathsf{Ext}(x)$ as follows: The m output bits are divided into κ continuous segments of m/κ bits; for every segment, there is $\log(m/\kappa)$ bits in the input that indicate a 1-entry in the segment, and we put 0's to all other entries. In such case, each output bit of $\mathsf{Ext}(x)$ is an AND of at most $\log(m/\kappa)$ input bits or their negations, thus the output of the decryption circuit is also a degree-$\log(m/\kappa)$ multi-variate polynomial over $\mathsf{GF}(2)$.

We now introduce our main assumption that can imply the hardness of $\mathscr{C}$-RPP for weak circuit classes $\mathscr{C}$. The item 1 and 2' of the conjecture is sufficient to rule out polynomial-sized nondeterministic algorithm for $(\mathsf{XOR} \circ \mathsf{AND}_{O(\log n)})$-$\mathsf{RPP}$.

Assumption 5.5. *There is a constant $\varepsilon \in (0,1)$ and a $\beta = \beta(n) \in (0,1)$ such that the following holds. For every sufficiently large n and every constant $\delta \in (0, 0.1)$, there are $m = O(n)$, $\kappa = \Theta(m^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = \Theta(m/\kappa^2)$, and a matrix $A \in \{0,1\}^{n \times m}$, such that:*

1. *Let $g : \{0,1\}^k \to \{0,1\}^n$ defined as $g(x) := A \cdot \mathsf{Ext}(x)^\top$, where Ext is defined as in Remark 5.4. Then $g(x)$ is an $(n^{\omega(1)}, \beta)$-secure demi-bits generator.*
2. *Let $f_v(s, e) := (sA + e, \langle v, s\rangle)$, where $s \in \{0,1\}^n$ and e is a string with Hamming weight w. Then for at least $\beta 2^n$ strings $v \notin \mathsf{Range}(g)$, $f_v(s, e)$ is an $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure pseudorandom generator.*

For hardness against nondeterministic algorithms, we replace property 2 by 2':

2'. *Let* $f_v(s,e) := (sA+e, \langle v, s\rangle)$*, where* $s \in \{0,1\}^n$ *and* e *is a string with Hamming weight* w*. Then for at least* $\beta 2^n$ *strings* $v \notin \mathsf{Range}(g)$*,* $f_v(s,e)$ *is an* $(n^{\omega(1)}, 2^{-w^\varepsilon})$*-secure super-bits generator.*

Indeed, we further conjecture that given the parameters δ, m, κ, k, w, a random matrices $A \in \{0,1\}^{n\times m}$ satisfies each of item 1 and 2' with high probability. Thus not only for one, but for most of the matrices A, this conjecture is likely to be true.

Remark 5.6. Note that one may hope to assume in Item 2 (or Item 2') that $f'(s,e) := sA + e$ is a PRG (or super-bits generator), and derive Item 2 (or Item 2') by the standard Goldreich-Levin theorem (see, e.g., [AB09]), since v is (somewhat) a random string over $\{0,1\}^n$. Here, we explain why this *does not work*.

Let $v \in \{0,1\}^n$ be good if $f_v(s,e)$ is a PRG (or super-bits generator) that is $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure. On can check that the Goldreich-Levin theorem can only be used to show that a 2^{-w^ε} fraction of $v \in \{0,1\}^n$ are good, therefore we need to set $\beta \approx 2^{-w^\varepsilon}$. However, this is impossible, as the demi-bits generator g in Item 1 can be broken by a simple algorithm that guesses the input of g with advantage 2^{-k}, which can be larger than 2^{-w^ε} if δ is sufficiently small.

Now we prove our main theorem.

Theorem 5.7. *Let* Ext *be the funtion in Remark 5.4, and* $\mathscr{C} := \mathsf{XOR} \circ \mathsf{AND}_{O(\log n)}$*. Assuming Item 1 and 2 in Assumption 5.5, for universal constants* $\alpha, \varepsilon \in (0, 1/2)$ *and every constant* $\delta \in (0, 0.1)$*, there is no polynomial-sized deterministic circuit for* $\mathscr{C}$-$\mathsf{RPP}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^2, \alpha]$ *for sufficiently large* n*.*

Moreover, assuming Item 1 and 2' in Assumption 5.5, for universal constants $\alpha, \varepsilon \in (0, 1/2)$ *and every constant* $\delta \in (0, 1)$*, there is no polynomial-sized non-uniform* $\mathsf{SearchNP}$ *algorithm for* $\mathscr{C}$-$\mathsf{RPP}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^2, \alpha]$ *for sufficiently large* n*.*

Proof. We only prove the "moreover" part; the other case can be proved similarly. Let α, ε be some constants to be chosen later. Suppose, towards a contradiction, that for some $\delta \in (0,1)$ and infinitely many n, there is a polynomial-sized non-uniform $\mathsf{SearchNP}$ algorithm for $\mathsf{Poly}[\log n]$-$\mathsf{RPP}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^2, \alpha]$.

Fix any such n that is sufficiently large. Let $\beta = \beta(n)$, $m = O(n), \kappa = \Theta(m^\delta) = \Theta(n^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = \Theta(m/\kappa^2) = \Theta(n^{1-2\delta})$, matrix $A \in \{0,1\}^{n\times m}$, and function $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be defined as in Assumption 5.5. Then by Assumption 5.5, we know that

1. $g(x) := A \cdot \mathsf{Ext}(x)^\top$ is an $(n^{\omega(1)}, \beta)$-secure demi-bits generator.
2. For at least $\beta 2^n$ fraction of strings $v \in \{0,1\}^n \setminus \mathsf{Range}(g)$, $f_v(s,e) := (sA+e, \langle v, s\rangle)$ (where $s \in \{0,1\}^n$ is random and e is a random string satisfying $|e| = w$) is an $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure super-bits generator. Such string v is said to be *good*.

Let $\Pi^{\mathsf{YES}} := \mathsf{Range}(g)$, $\Pi^{\mathsf{NO}} := \{v \in \{0,1\}^n \mid v \text{ is good}\} \setminus \mathsf{Range}(g)$, and $V(y \in \{0,1\}^n, x \in \{0,1\}^k)$ be the standard verifier of $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ that accepts if and only if $y = A \cdot \mathsf{Ext}(x)$. By Theorem 5.1, there is a witness encryption scheme for $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ w.r.t. V that is $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure against any non-deterministic adversary with decryption error $\kappa w/(m-w) = o(1)$. Then, by Lemma 3.3, there is a $\mathsf{poly}(n,m)$-sized proof system for $\overline{L}$. In other words, there is a polynomial-sized non-deterministic algorithm A such that

- A rejects every $x \in \Pi^{\mathsf{YES}} = \mathsf{Range}(g)$.
- A accepts every $x \in \Pi^{\mathsf{NO}}$, where $|\Pi^{\mathsf{NO}}| \geq \beta 2^n$.

This breaks the demi-bits generator g, which leads to a contradiction. □

5.3 Candidate from Random Lattices

Similar to the candidate witness encryption from LPN, we propose a candidate based on LWE (where the error is bounded by its ℓ_∞-norm) for certain hard language in NP. This candidate is inspired by the standard construction of public-key encryption from LWE. Compared to the candidate in previous sub-section, this candidate has slightly larger decryption complexity but enjoys perfect correctness, and therefore it can be used to show the hardness of the range avoidance problem.

Let $q = n^{O(1)}$ be a modulus, $m = O(n \log q)$. Arithmetic operations such as addition, multiplication, and inner product of vectors denotes the operations in $\mathbb{Z}_q$.

Theorem 5.8. *Let $k < n < m$ and $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ be a function such that $\mathsf{Ext}(x)$ is κ-sparse. For every matrix $A \in \mathbb{Z}_q^{n \times m}$, suppose we define:*

- $\Pi^{\mathsf{YES}} := \{A \cdot \mathsf{Ext}(x) \in \mathbb{Z}_q \mid x \in \{0,1\}^k\}$.
- *Let $f_v(s,e) := (sA + e, \langle v, s\rangle)$ be a function, where $s \in \mathbb{Z}_q^n$ and $e \in \mathbb{Z}_q^m$ be a random vector such that $\|e\| = w$, $w = q/(10\kappa)$. We define Π^{NO} be the set consisting of $v \in \mathbb{Z}_q^n \setminus \Pi^{\mathsf{YES}}$ such that $f_v(s,e)$ is an (s_2, ε)-secure pseudorandom generator.*
- *Let $V(y \in \mathbb{Z}_q^n, x \in \{0,1\}^k)$ be the standard verifier of $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ that accepts if and only if $y = A \cdot \mathsf{Ext}(x)$.*

Then there is a witness encryption scheme for $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ w.r.t. V that is (s_2, ε)-secure with perfect correctness.

Moreover, if we define Π^{NO} be the set of $v \notin \Pi^{\mathsf{YES}}$ such that $f_v(s,e)$ is an (s_2, ε)-secure super-bits generator, then there is a witness encryption scheme for $(\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ w.r.t. V that is (s_2, ε)-secure against any nondeterministic adversary *with perfect correctness.*

Since the proof is essentially the same as that of Theorem 5.1, we only provide the construction of the scheme and omit the proof.

Construction. Let $g : \{0,1\}^k \to \mathbb{Z}_q^n$ be the function $g(x) := A \cdot \mathsf{Ext}(x)^\top$. Suppose that $v \in \mathbb{Z}_q^n$ and we want to encrypt on the statement $v \in \mathsf{Range}(g)$.

- **(Encryption).** Given any message bit $b \in \{0,1\}$, the encryption algorithm samples a random $s \in \mathbb{Z}_q^n$ and a random $e \in \mathbb{Z}_q^m$ with norm w, and outputs $\mathsf{Enc}(b, v; s, e) = (sA + e, \langle v, s\rangle + b \cdot \lfloor q/2 \rfloor))$.
- **(Decryption).** Given a witness $x \in \{0,1\}^k$ such that $\mathsf{Ext}(x) = v$, and a cipher text $\mathsf{ct} = (u, c) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$, the decryption algorithm $\mathsf{Dec}(\mathsf{ct}, v, x)$ computes $z = \langle u, \mathsf{Ext}(x)\rangle - c$ and outputs 1 if and only if $z \in [q/3, 2q/3]$.

To see that the decryption algorithm is perfectly correct, one can calculate that for $u = sA + e$ and $c = \langle v, s\rangle + b \cdot \lfloor q/2 \rfloor$, where $v = A \cdot \mathsf{Ext}(x)$,

$$\begin{aligned} \langle u, \mathsf{Ext}(x)\rangle - c &= sA \cdot \mathsf{Ext}(x) + \langle e, \mathsf{Ext}(x)\rangle - \langle v, s\rangle - b \cdot \lfloor q/2 \rfloor \\ &= \langle e, \mathsf{Ext}(x)\rangle - b \cdot \lfloor q/2 \rfloor \qquad (\star). \end{aligned}$$

Since $\|e\| \le q/(10\kappa)$, $\|\mathsf{Ext}(x)\| \le \kappa$, we know that $|\langle e, \mathsf{Ext}(x)\rangle| \le q/10$, and therefore $b = 1$ if and only if $(\star) \in [q/3, 2q/3]$.

Complexity of the Decryption Circuit. We now consider the circuit complexity of the decryption circuit $\mathsf{Dec}(\mathsf{ct}, v, \cdot) : \{0,1\}^k \to \{0,1\}$. Recall that:

- A DOR gate is an unbounded fan-in Boolean OR gate with a semantic promise that at most one of its input wires is 1.

- An EMAJ gate is an unbounded fan-in gate that outputs 1 if and only if exactly one half of its input wires are 1. By adding multiple wires and fixing constants, an EMAJ gate can compute a function $f(x) := [\langle w, x\rangle = \beta]$, where $w \in \mathbb{N}^n$, $\beta \in \mathbb{Z}$, in size $O(|\beta| + \sum_i |w_i|)$.

We will show that $\mathsf{Dec}(\mathsf{ct}, v, x)$ can be computed by the following $\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{Ext}$ circuit of size $O((mq)^2)$.

- Given any input $x = x_1 \| x_2 \| \dots \| x_k \in \{0,1\}^k$, we first compute $\hat{v} = \mathsf{Ext}(x)$.
- Let $\mathsf{ct} = (u, c) \in \mathbb{Z}_q^m \times \mathbb{Z}_q$ be hardcoded in the circuit. By the definition of the decryption algorithm, we will output 1 if and only if $z = \langle u, \hat{v}\rangle + c \in [q/3, 2q/3]$. If we consider the operations in $\mathbb{Z}$, $\langle u, \hat{v}\rangle + c \in mq + q \le (m+1)q$. Therefore, for every $r \le (m+1)q$ such that $r \bmod q \in [q/3, 2q/3]$, we construct an EMAJ gate G_r that computes:

$$G_r(x) = 1 \iff \sum_{i=1}^{m} u_i \hat{v}_i = r - c,$$

  in size $O(|r - c| + \sum_i |u_i|) = O(mq)$.
- It is easy to see that $\mathsf{Dec}(\mathsf{ct}, v, x) = 1$ if and only if there is an $r \le (m+1)q$ as above such that $G_r(x) = 1$. Moreover, for every input x, there is at most one $r \le (m+1)q$ such that $G_r(x) = 1$. Therefore, we construct a DOR gate whose input wires connect to all G_r gates as defined above.

In particular, if $\mathsf{Ext} : \{0,1\}^k \to \{0,1\}^m$ is the function defined in Remark 5.4, the decryption can be implemented by a $\mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{AND}_{O(\log n)}$ circuit of size $O((mq)^2)$.

We now state the assumption and prove our main result for the hardness of range avoidance.

Assumption 5.9. *There is a constant $\varepsilon \in (0,1)$ and a $\beta = \beta(n) \in (0,1)$ such that the following holds. For every sufficiently large n and every constant $\delta \in (0, 0.1)$, there are $q = n^{O(1)}$, $m = O(n \log q)$, $\kappa = \Theta(m^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = q/10\kappa$, and a matrix $A \in \{0,1\}^{n \times m}$, such that:*

1. *Let $g : \{0,1\}^k \to \mathbb{Z}_q^n$ defined as $g(x) := A \cdot \mathsf{Ext}(x)^\top$, where Ext is defined as in Remark 5.4. Then $g(x)$ is an $(n^{\omega(1)}, \beta)$-secure demi-bits generator.*
2. *Let $f_v(s, e) := (sA + e, \langle v, s\rangle)$, where $s \in \mathbb{Z}_q^n$ and $e \in \mathbb{Z}_q^m$ is a vector with ℓ_∞-norm w. Then for at least $\beta 2^n$ vectors $v \in \mathbb{Z}_q^n \setminus \mathsf{Range}(g)$ such that $f_v(s, e)$ is an $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure pseudorandom generator.*

For hardness against nondeterministic algorithms, we replace property 2 by 2′:

2′. *Let $f_v(s, e) := (sA + e, \langle v, s\rangle)$, where $s \in \mathbb{Z}_q^n$ and $e \in \mathbb{Z}_q^m$ is a vector with ℓ_∞-norm w. Then for at least $\beta 2^n$ vectors $v \in \mathbb{Z}_q^n \setminus \mathsf{Range}(g)$ such that $f_v(s, e)$ is an $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure super-bits generator.*

Theorem 5.10. *Let Ext be the funtion in Remark 5.4, and $\mathscr{C} := \mathsf{DOR} \circ \mathsf{EMAJ} \circ \mathsf{AND}_{O(\log n)}$. Assuming Item 1 and 2 in Assumption 5.9, there is a universal constant $\varepsilon \in (0, 1/2)$ such that for every constant $\delta \in (0,1)$, there is no polynomial-sized circuit that solves $\mathscr{C}$-$\mathsf{Avoid}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^{O(1)}]$ for sufficiently large n.*

Moreover, assuming Item 1 and 2′ in Assumption 5.5, there is a universal constant $\varepsilon \in (0, 1/2)$ such that for every constant $\delta \in (0,1)$, there is no polynomial-sized non-uniform $\mathsf{SearchNP}$ algorithm for $\mathscr{C}$-$\mathsf{Avoid}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^2, \alpha]$ for sufficiently large n.

Proof. We only prove the "moreover" part. Let ε be the constant in Assumption 5.9. Towards a contradiction, we assume that there is a constant $\delta \in (0,1)$ such that for infinitely many n, $\mathscr{C}$-$\mathsf{Avoid}[\Theta(n^\delta), \Theta(n^{\varepsilon(1-2\delta)}), n^{O(1)}]$ admits a polynomial-sized non-uniform $\mathsf{SearchNP}$ algorithm.

Let n be a sufficiently large input length satisfying the condition above. Let $\beta = \beta(n)$, $q = n^{O(1)}$, $m = O(n \log q)$, $\kappa = \Theta(m^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = q/(10\kappa)$, and matrix A be defined in Assumption 5.9. It follows that

- $g(x) := A \cdot \mathsf{Ext}(x)^\top$ is an $(n^{\omega(1)}, \beta)$-secure demi-bits generator;
- for at least $\beta \cdot q^n$ vectors $v \in \mathbb{Z}_q^n \setminus \mathsf{Range}(g)$, called *good vectors*, $f_v(s, e) := (sA + e, \langle v, s \rangle)$ is an $(n^{\omega(1)}, 2^{-m^\varepsilon})$-secure super-bits generator.

Let $\Pi^{\mathsf{YES}} := \mathsf{Range}(g)$, Π^{NO} be the set of good vectors, $V(y \in \mathbb{Z}_q^n, x \in \{0,1\}^k)$ is the standard verifier of $L = (\Pi^{\mathsf{YES}}, \Pi^{\mathsf{NO}})$ that accepts if and only if $y = A \cdot \mathsf{Ext}(x)^\top$. By Theorem 5.8, there is an $(n^{\omega(1)}, 2^{-m^\varepsilon})$-secure perfectly correct witness encryption scheme for L with respect to V. By Lemma 3.3, the complement $\overline{L}$ of L has a $\mathsf{poly}(n, m)$-sized proof system; that is, there is a polynomial-sized non-uniform nondeterministic algorithm A such that

- for every $x \in \Pi^{\mathsf{YES}} = \mathsf{Range}(g)$, $A(x)$ rejects;
- for at least $|\Pi^{\mathsf{NO}}| \geq \beta \cdot q^n$ vectors $x \in \mathbb{Z}_q^n$, $A(x)$ accepts.

This is impossible as g is a secure demi-bits generator. □

5.4 Attacks to the Assumptions

We do not know how to base Assumption 5.5 and 5.9 on well-established cryptographic assumptions (such as LPN or LWE). Instead, we demonstrate the attacks we have tried to Assumption 5.5 and 5.9 utilizing known methods, including brute-force, linear algebra, and geometric approaches.

5.4.1 Seed-Guessing Attack

Both Assumption 5.5 and Assumption 5.9 assume that certain concrete functions are secure super-bits or demi-bits generators. The most obvious way to break a super-bits or demi-bits generators is to guess the seed (i.e., the input) of the generators. Therefore, we need to verify that the seed-guessing attack cannot achieve the required success probability given the parameters in Assumption 5.5 and 5.9.

Since the seed-guessing attack are quite similar for Assumption 5.5 and Assumption 5.9, we only demonstrate the former scenario. Let $\varepsilon \in (0,1)$, $\beta = \beta(n) \in (0,1)$, $\delta \in (0,1)$, $m = O(n)$, $\kappa = \Theta(m^\delta)$, $k = \Theta(\kappa \cdot \log(n/\kappa))$, $w = \Theta(m/\kappa^2)$, and matrix $A \in \{0,1\}^{n \times m}$ as described in Assumption 5.5. The assumption (for the hardness against nondeterminism) states that:

- $g(x) := A \cdot \mathsf{Ext}(x)^\top$ is an $(n^{\omega(1)}, \beta)$-secure demi-bits generator.
- For at least $\beta \cdot 2^n$ strings $v \in \{0,1\}^n \setminus \mathsf{Range}(g)$, $f_v(s, e) := (sA + e, \langle s, v \rangle)$ is an $(n^{\omega(1)}, 2^{-w^\varepsilon})$-secure super-bits generator, where e is a random string with Hamming weight w.

Let $\beta = 0.1$ for concreteness.

- (*Attacking demi-bits*). Recall that we need to reject every string $y \in \mathsf{Range}(g)$ and accept a β fraction of strings in $\{0,1\}^n$ to break the demi-bits generator. However, the seed-guessing attack cannot guarantee the rejection of strings $y \in \mathsf{Range}(g)$.
- (*Attacking super-bits*). Given $y = f_v(s, e) = (sA + e, \langle s, v \rangle)$, we can randomly guess the seed (s, e), where $s \in \{0,1\}^n$ and $e \in \{0,1\}^m$ satisfying $|e| = w$, and reject if $y = f_v(s, e)$. Moreover, since s can be computed from y and e by Gaussian elimination, we only need to guess the string e, leading to an advantage of (roughly) $\binom{m}{w}/2^m \leq 2^{-w}$. This is much smaller than the required advantage 2^{-w^ε} to break the super-bits generator.

5.4.2 Linear Algebraic Attack to Demi-bits

We now demonstrate a standard linear algebraic attack that breaks the demi-bits generator $g(x) := A \cdot \mathsf{Ext}(x)^\top$ when the matrix $A \in \{0,1\}^{n \times m}$ is close to being a square matrix. This suggests that we should set m to be appropriately larger than n.

Formally, we will show that when the rows of $A \in \mathbb{Z}_q^{n \times m}$ are linearly independent and $4\kappa m q^{1-n/m} < q/2$, then $g(x) = A \cdot \mathsf{Ext}(x)^\top$ is not an $(n^{\omega(1)}, 1/2)$-secure demi-bits generator. Let $B \in \mathbb{Z}_q^{m \times (m-n)}$ be a matrix that spans the kernel of A, i.e., $AB = 0$ and the columns of B are linearly independent.

Proposition 5.11. *There is a vector $e \in \mathbb{Z}_q^m$, $1 \le \|e\| \le 4q^{1-n/m}$ such that $eB = 0$.*

Proof. It is easy to see that there are β^m vectors $e \in \mathbb{Z}_q^m$ such that $\|e\| \le \beta$, and $eB \in \mathbb{Z}_q^{m-n}$ has q^{m-n} possible values. Let $\beta := 2q^{1-n/m}$. By the pigeonhole principle, there are $e_1, e_2 \in \mathbb{Z}_q^m$ such that $\|e_1\|, \|e_2\| \le \beta$ and $e_1 B = e_2 B$. Therefore, $e := e_1 - e_2$ satisfy $eB = 0$ and $\|e\| \le 4q^{1-n/m}$. □

Fix this $e \in \mathbb{Z}_q^m$. Given an input $v \in \mathbb{Z}_q^n$, we first compute $z \in \mathbb{Z}_q^m$ such that $v = Az$ using Gaussian elimination. We know that if there is an $x \in \{0,1\}^k$ such that $g(x) = v$, i.e., $A \cdot \mathsf{Ext}(x)^\top = v$, then there exists a $t \in \mathbb{Z}_q^{m-n}$ such that $z = \mathsf{Ext}(x)^\top + Bt$. Since $\|\mathsf{Ext}(x)\| \le \kappa$, we know that

$$\langle z, e \rangle = \langle e, \mathsf{Ext}(x) \rangle \le m \cdot \|e\| \cdot \|\mathsf{Ext}(x)\| \le 4\kappa m q^{1-n/m}.$$

On the other hand, for most matrix A and vector v, let z be a random solution of $Az = v$, then $\langle z, e \rangle$ should not be always much smaller than q. This is formally shown in the following proposition.

Proposition 5.12. *For a random vector v and a random solution z of $Az = v$, the probability that $\langle e, z \rangle > q/2$ is at least $1/2$.*

Proof. Since $B \in \mathbb{Z}_q^{m \times (m-n)}$ is of full rank in columns, for every $v \in \mathbb{Z}_q^n$, there are exactly q^{m-n} solutions to q^{m-n}. This means that a random solution z to $Az = v$ of a random vector v is uniformly distributed over $\mathbb{Z}_q^m$. Since $e \neq 0$, we know that $\rangle e, z \rangle$ is uniformly distributed over q, which further means that $\Pr[\langle e, z \rangle \ge q/2] \ge 1/2$. □

This attack works when $4\kappa m q^{1-n/m} < q/2$, that is, $4\kappa m \le 2^{n \log q/m}$. For instance, we can set $m = 10n \log q$ in Assumption 5.9 to avoid this attack. Note that since similar attack also works for Assumption 5.5, we should set m to be appropriately large, say $m = 10n$, in Assumption 5.5.

5.4.3 Linear Algebraic Attack to Super-bits

We know from the discussion above that the width m of the matrix A should not be too small. In fact, Assumption 5.5 and 5.9 are also insecure when m is much larger than n.

For instance, we consider the super-bits generator $f_v(s, e) = (sA + e, \langle s, v \rangle)$ in Assumption 5.5, where $v, s \in \{0,1\}^n$, and $e \in \{0,1\}^m$ is of Hamming weight w, for $w = \Theta(m/\kappa^2)$, $\kappa = \Theta(m^\delta)$. For concreteness, we assume that $m = n^2$.

Proposition 5.13. *For every $A \in \{0,1\}^{n \times m}$, there exists a string $z \in \{0,1\}^n$ such that $Az = 0$ and $|z| \le n + 1$.*

Proof. Fix any subset $I \subseteq [m]$ such that $|I| = n + 1$. Consider the set Z consisting of strings $z \in \{0,1\}^n$ whose 1-indices are in I. Since $|Z| = 2^{n+1}$ and $Az \in \{0,1\}^n$ only has 2^n possible values, by the pigeonhole principle, there are $z_1, z_2 \in Z$ such that $z_1 \neq z_2$ and $Az_1 = Az_2$. Therefore, $z = z_1 - z_2$ is a string satisfying that $Az = 0$ and $|z| \le n + 1$. □

Fix one of the string z in the proposition above. It is easy to verify that

- For the output of the super-bits generator $f_v(s,e) = (sA+e, \langle s, v\rangle)$, $(sA+e)z = \langle e, z\rangle$. Since $|e| \cdot |z| \leq \kappa(n+1) \ll m$, we know that $(sA+e)z = 0$ with high probability.
- For uniformly random strings $(u,c) \in \{0,1\}^n \times \{0,1\}$, $\Pr[\langle u, z\rangle = 1] = 1/2$.

Therefore, we can construct a non-uniform deterministic algorithm that accepts with significantly higher probability given uniformly random string than given the distribution given by the generator $f_v(s,e)$, which means that $f_v(s,e)$ is not even a pseudorandom generator.

5.4.4 Geometric Attack

Since Assumption 5.9 is based on hardness of lattice problems, we need to verify that it survives known attacks using non-trivial algorithms for lattice problems.

Recall that an m-dimensional lattice L is a discrete subset of $\mathbb{R}^m$ defined as the set of points that are integer linear combinations of a basis $B \in \mathbb{R}^{m\times m}$, i.e., $L(B) := \{Bz \in \mathbb{R}^m \mid z \in \mathbb{Z}^n\}$. The distance of a point p to a lattice L, denoted by $\mathrm{dist}(L,p)$, is defined as the minimum distance between p and a point in L. The approximate version of the *closest vector problem* with approximation factor δ (abbrev. GapCVP_δ) is the following promise problem: Given a lattice $L = L(B)$, a point $p \in \mathbb{R}^m$, and a distance parameter d,

- (L,p,d) is a YES-instance if $\mathrm{dist}(L,p) \leq d$;
- (L,p,d) is a NO-instance if $\mathrm{dist}(L,p) \geq \delta \cdot d$.

It is easy to see that $\mathsf{GapCVP}_\delta \in \mathsf{NP}$. Moreover, it is known that $\mathsf{GapCVP}_\delta \in \mathsf{coAM}$ (and therefore $\mathsf{GapCVP}_\delta \in \mathsf{coNP}_{/\mathsf{poly}}$ since $\mathsf{AM} \subseteq \mathsf{NP}_{/\mathsf{poly}}$) for ℓ_∞-norm and $\delta = O(m/\log m)$ [GG98; AR05]. In other words, there is an $\mathsf{NP}_{/\mathsf{poly}}$ algorithm A such that

- $A(L,p,d)$ accepts if $\mathrm{dist}(L,p) \geq O(dm/\log m)$;
- $A(L,p,d)$ rejects if $\mathrm{dist}(L,p) \leq d$.

This can be used to attack both the demi-bits and the super-bits generator in Assumption 5.9.

Attacks to demi-bits. Let $g : \{0,1\}^k \to \mathbb{Z}_q^n$, $g(x) := A \cdot \mathsf{Ext}(x)^\top$ be the demi-bits generator in Assumption 5.9. Consider the lattice $L_q^\perp(A) := \{w \in \mathbb{Z}^m \mid Aw \equiv 0 \pmod q\}$ (usually called the SIS-lattice of A).

For every $v \in \mathbb{Z}_q^n$, let z be an integer solution of $Az = v \pmod q$, then:

- If $v \in \mathsf{Range}(g)$, i.e., $v = A \cdot \mathsf{Ext}(x)$ for some x, we know that $A(z - \mathsf{Ext}(x)) = 0$. This implies that $z - \mathsf{Ext}(x) \in L_q^\perp(A)$, and therefore $\mathrm{dist}(z, L_q^\perp(A)) \leq \|\mathsf{Ext}(x)\| = 1$.
- Note that $\mathrm{dist}(z, L_q^\perp(A)) = \min\{\|t\| \mid At = v\}$. By a simple counting argument, we can see that, as long as there is an $n \times n$ block of A that is invertible mod q, then

$$\Pr\left[\mathrm{dist}(z, L_q^\perp(A)) \leq q^{(n-1)/m}\right] \leq 1/q$$

  for a uniformly random $v \in \mathbb{Z}_q^n$. Note that this bound is near-optimal as we can also prove $\mathrm{dist}(z, L_q^\perp(A)) = O(q^{n/m})$ using the pigeonhole principle.

If $\mathsf{GapCVP}_\delta \in \mathsf{coNP}_{/\mathsf{poly}}$ for $\delta > q^{(n-1)/m}$, we can use this algorithm to reject every $v \in \mathsf{Range}(g)$ and accepts at least a $1/q$ fraction of strings $v \in \mathbb{Z}_q^n$. However, as we can choose $m = O(n \log q)$ so that $q^{(n-1)/m} = O(1) \ll m/\log m$, we cannot perform this attack using the $\mathsf{GapCVP}_{O(m/\log m)}$ algorithm as described above.

Attacks to super-bits. Let $f_v(s,e) = (sA + e, \langle v, s\rangle)$ be the super-bits generator in Assumption 5.9. Consider the lattice $L_q(A) := \{sA \in \mathbb{Z}^m \mid s \in \mathbb{Z}_q^n\} + q\mathbb{Z}^m$ (usually called the LWE-lattice of A).

- For every $z = f_v(s,e) = sA + e$, we know that $\mathrm{dist}(z, L_q(A)) \le \|e\| \le w = \Theta(q/m^\delta)$.
- For every $z \in \mathbb{Z}_q^m$, $\mathrm{dist}(z, L_q(A)) = \min\{\|e\| \mid \exists s \in \mathbb{Z}_q^n,\ z + e = sA\}$. By a simple counting argument, we can see that $\Pr\left[\mathrm{dist}(z, L_q(A)) \le q^{\frac{m-n-1}{m}}\right] \le 1/q$ for a uniformly random $z \in \mathbb{Z}_q^m$, i.e., most vectors are $q^{\frac{m-n-1}{m}}$-far from $L_q(A)$. Note that the bound is near-optimal as $\mathrm{dist}(z, L_q(A)) \le q$.

If $\mathsf{GapCVP}_\delta \in \mathsf{coNP}_{/\mathsf{poly}}$ for $\delta > m^\delta \cdot q^{-\frac{n+1}{m}}$, we can use the algorithm to distinguish $f_v(s,e)$ and the uniform distribution. However, since $m^\delta \cdot q^{-\frac{n+1}{m}} = O(m^\delta) \ll m/\log m$, we cannot perform this attack using the $\mathsf{GapCVP}_{O(m/\log m)}$ algorithm as described above.

References

[AR05] Dorit Aharonov and Oded Regev. "Lattice problems in NP cap coNP". In: *J. ACM* 52.5 (2005), pp. 749–765. DOI: 10.1145/1089023.1089025. URL: https://doi.org/10.1145/1089023.1089025 (cit. on pp. 6, 7, 12, 30, 37).

[Ajt83] Miklós Ajtai. "Σ_1^1-Formulae on finite structures". In: *Ann. Pure Appl. Log.* 24.1 (1983), pp. 1–48. DOI: 10.1016/0168-0072(83)90038-6 (cit. on p. 1).

[AD97] Miklós Ajtai and Cynthia Dwork. "A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence". In: *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*. Ed. by Frank Thomson Leighton and Peter W. Shor. ACM, 1997, pp. 284–293. DOI: 10.1145/258533.258604. URL: https://doi.org/10.1145/258533.258604 (cit. on pp. 7, 8, 23).

[Ale11] Michael Alekhnovich. "More on Average Case vs Approximation Complexity". In: *Comput. Complex.* 20.4 (2011), pp. 755–786. DOI: 10.1007/s00037-011-0029-x. URL: https://doi.org/10.1007/s00037-011-0029-x (cit. on pp. 7, 8, 12, 23, 29, 30).

[APY09] Noga Alon, Rina Panigrahy, and Sergey Yekhanin. "Deterministic Approximation Algorithms for the Nearest Codeword Problem". In: *APPROX-RANDOM*. Vol. 5687. Lecture Notes in Computer Science. Springer, 2009, pp. 339–351. DOI: 10.1007/978-3-642-03685-9_26 (cit. on pp. 2, 4, 13).

[AIK06] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. "Cryptography in NC^0". In: *SIAM J. Comput.* 36.4 (2006), pp. 845–888 (cit. on p. 1).

[AB09] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009. URL: http://www.cambridge.org/catalogue/catalogue.asp?isbn=9780521424264 (cit. on pp. 13, 15, 32).

[AS10] Vikraman Arvind and Srikanth Srinivasan. "Circuit Lower Bounds, Help Functions, and the Remote Point Problem". In: *ICS*. Tsinghua University Press, 2010, pp. 383–396. URL: http://conference.iiis.tsinghua.edu.cn/ICS2010/content/papers/30.html (cit. on pp. 2, 4).

[Bar17] Boaz Barak. "The Complexity of Public-Key Cryptography". In: *Tutorials on the Foundations of Cryptography*. Ed. by Yehuda Lindell. Springer International Publishing, 2017, pp. 45–77. DOI: 10.1007/978-3-319-57048-8_2. URL: https://doi.org/10.1007/978-3-319-57048-8_2 (cit. on pp. 13, 30).

[BIOW20] Ohad Barta, Yuval Ishai, Rafail Ostrovsky, and David J. Wu. "On Succinct Arguments and Witness Encryption from Groups". In: *CRYPTO (1)*. Vol. 12170. Lecture Notes in Computer Science. Springer, 2020, pp. 776–806. DOI: 10.1007/978-3-030-56784-2_26. URL: https://doi.org/10.1007/978-3-030-56784-2_26 (cit. on pp. 7, 9, 17, 19).

[Bar+20] James Bartusek et al. "Affine Determinant Programs: A Framework for Obfuscation and Witness Encryption". In: *ITCS*. Vol. 151. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020, 82:1–82:39 (cit. on pp. 7, 9, 17, 19).

[BBD09] Daniel Bernstein, Johannes Buchmann, and Erik Dahmen. *Post-quantum cryptography. First international workshop PQCrypto 2006, Leuven, The Netherland, May 2326, 2006. Selected papers*. Jan. 2009, pp. 147–191. ISBN: 978-3-540-88701-0. DOI: 10.1007/978-3-540-88702-7 (cit. on pp. 7, 25).

[BHPT20] Amey Bhangale, Prahladh Harsha, Orr Paradise, and Avishay Tal. "Rigid Matrices From Rectangular PCPs or: Hard Claims Have Complex Proofs". In: *FOCS*. IEEE, 2020, pp. 858–869. DOI: 10.1109/FOCS46700.2020.00084 (cit. on pp. 3, 15).

[CLORS23] Lijie Chen, Zhenjian Lu, Igor C. Oliveira, Hanlin Ren, and Rahul Santhanam. "Polynomial-Time Pseudodeterministic Construction of Primes". In: *FOCS (to appear)*. 2023 (cit. on p. 8).

[CLW20] Lijie Chen, Xin Lyu, and R. Ryan Williams. "Almost-Everywhere Circuit Lower Bounds from Non-Trivial Derandomization". In: *FOCS*. IEEE, 2020, pp. 1–12. DOI: 10.1109/FOCS46700.2020.00009 (cit. on p. 2).

[CR22] Lijie Chen and Hanlin Ren. "Strong Average-Case Circuit Lower Bounds from Non-trivial Derandomization". In: *SIAM J. Comput.* 51.3 (2022), STOC20-115-STOC20–173. DOI: 10.1137/20M1364886 (cit. on p. 3).

[CHLR23] Yeyuan Chen, Yizhi Huang, Jiatu Li, and Hanlin Ren. "Range Avoidance, Remote Point, and Hard Partial Truth Table via Satisfying-Pairs Algorithms". In: *STOC*. ACM, 2023, pp. 1058–1066. DOI: 10.1145/3564246.3585147 (cit. on pp. 1, 2, 3, 4, 5).

[CVW18] Yilei Chen, Vinod Vaikuntanathan, and Hoeteck Wee. "GGH15 Beyond Permutation Branching Programs: Proofs, Attacks, and Candidates". In: *CRYPTO (2)*. Vol. 10992. Lecture Notes in Computer Science. Springer, 2018, pp. 577–607. DOI: 10.1007/978-3-319-96881-0_20. URL: https://doi.org/10.1007/978-3-319-96881-0_20 (cit. on pp. 7, 9, 17, 19).

[DH76] Whitfield Diffie and Martin E. Hellman. "New directions in cryptography". In: *IEEE Trans. Inf. Theory* 22.6 (1976), pp. 644–654 (cit. on p. 17).

[Erd59] Paul Erds. "Graph theory and probability". In: *Canadian Journal of Mathematics* 11 (1959), pp. 34–38. DOI: 10.4153/CJM-1959-003-9 (cit. on p. 1).

[FGHK16] Magnus Gausdal Find, Alexander Golovnev, Edward A. Hirsch, and Alexander S. Kulikov. "A Better-Than-3n Lower Bound for the Circuit Complexity of an Explicit Function". In: *FOCS*. IEEE Computer Society, 2016, pp. 89–98. DOI: 10.1109/FOCS.2016.19 (cit. on p. 1).

[FSS84] Merrick L. Furst, James B. Saxe, and Michael Sipser. "Parity, Circuits, and the Polynomial-Time Hierarchy". In: *Math. Syst. Theory* 17.1 (1984), pp. 13–27. DOI: 10.1007/BF01744431 (cit. on p. 1).

[GGNS23] Karthik Gajulapalli, Alexander Golovnev, Satyajeet Nagargoje, and Sidhant Saraogi. "Range Avoidance for Constant-Depth Circuits: Hardness and Algorithms". In: *CoRR* abs/2303.05044 (2023). DOI: 10.48550/arXiv.2303.05044. arXiv: 2303.05044. URL: https://doi.org/10.48550/arXiv.2303.05044 (cit. on pp. 1, 4).

[GGSW13] Sanjam Garg, Craig Gentry, Amit Sahai, and Brent Waters. "Witness encryption and its applications". In: *STOC*. ACM, 2013, pp. 467–476. DOI: 10.1145/2488608.2488667. URL: https://doi.org/10.1145/2488608.2488667 (cit. on pp. 7, 10, 13, 16).

[Gar+16] Sanjam Garg et al. "Candidate Indistinguishability Obfuscation and Functional Encryption for All Circuits". In: *SIAM J. Comput.* 45.3 (2016), pp. 882–929. DOI: 10.1137/14095772X. URL: https://doi.org/10.1137/14095772X (cit. on pp. 17, 19).

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. "Trapdoors for hard lattices and new cryptographic constructions". In: *STOC*. ACM, 2008, pp. 197–206. DOI: 10.1145/1374376.1374407 (cit. on pp. 11, 29).

[GG98] Oded Goldreich and Shafi Goldwasser. "On the Limits of Non-Approximability of Lattice Problems". In: *Proceedings of the Thirtieth Annual ACM Symposium on the Theory of Computing, Dallas, Texas, USA, May 23-26, 1998*. Ed. by Jeffrey Scott Vitter. ACM, 1998, pp. 1–9. DOI: 10.1145/276698.276704. URL: https://doi.org/10.1145/276698.276704 (cit. on pp. 6, 7, 12, 30, 37).

[GL89] Oded Goldreich and Leonid A. Levin. "A Hard-Core Predicate for all One-Way Functions". In: *STOC*. ACM, 1989, pp. 25–32. DOI: 10.1145/73007.73010 (cit. on p. 8).

[GM84] Shafi Goldwasser and Silvio Micali. "Probabilistic Encryption". In: *J. Comput. Syst. Sci.* 28.2 (1984), pp. 270–299 (cit. on pp. 10, 17, 24).

[GS89] Shafi Goldwasser and Michael Sipser. "Private Coins versus Public Coins in Interactive Proof Systems". In: *Adv. Comput. Res.* 5 (1989), pp. 73–90 (cit. on p. 15).

[GLW22] Venkatesan Guruswami, Xin Lyu, and Xiuhan Wang. "Range Avoidance for Low-Depth Circuits and Connections to Pseudorandomness". In: *APPROX/RANDOM*. Vol. 245. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 20:1–20:21. DOI: 10.4230/LIPIcs.APPROX/RANDOM.2022.20 (cit. on pp. 1, 4).

[HP10] Kristoffer Arnsfelt Hansen and Vladimir V. Podolskii. "Exact Threshold Circuits". In: *CCC*. IEEE Computer Society, 2010, pp. 270–279. DOI: 10.1109/CCC.2010.33 (cit. on p. 14).

[Hås89] Johan Håstad. "Almost Optimal Lower Bounds for Small Depth Circuits". In: *Adv. Comput. Res.* 5 (1989), pp. 143–170 (cit. on p. 1).

[Hir22] Shuichi Hirahara. "NP-Hardness of Learning Programs and Partial MCSP". In: *FOCS*. IEEE, 2022, pp. 968–979. DOI: 10.1109/FOCS54457.2022.00095 (cit. on p. 8).

[HIR23] Yizhi Huang, Rahul Ilango, and Hanlin Ren. "NP-Hardness of Approximating Meta-Complexity: A Cryptographic Approach". In: *STOC*. ACM, 2023, pp. 1067–1075. DOI: 10.1145/3564246.3585154 (cit. on p. 8).

[ILW23] Rahul Ilango, Jiatu Li, and R. Ryan Williams. "Indistinguishability Obfuscation, Range Avoidance, and Bounded Arithmetic". In: *STOC*. ACM, 2023, pp. 1076–1089. DOI: 10.1145/3564246.3585187 (cit. on pp. 1, 2, 3, 4, 7, 8, 18, 22).

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability obfuscation from well-founded assumptions". In: *STOC*. ACM, 2021, pp. 60–73. DOI: 10.1145/3406325.3451093. URL: https://doi.org/10.1145/3406325.3451093 (cit. on pp. 7, 9, 17, 19).

[JLS22] Aayush Jain, Huijia Lin, and Amit Sahai. "Indistinguishability Obfuscation from LPN over $\mathbb{F}_p$, DLIN, and PRGs in NC^0". In: *EUROCRYPT (1)*. Vol. 13275. Lecture Notes in Computer Science. Springer, 2022, pp. 670–699. DOI: 10.1007/978-3-031-06944-4_23. URL: https://doi.org/10.1007/978-3-031-06944-4_23 (cit. on pp. 7, 9, 17, 19).

[Kha22] Erfan Khaniki. "Nisan-Wigderson Generators in Proof Complexity: New Lower Bounds". In: *CCC*. Vol. 234. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 17:1–17:15. DOI: 10.4230/LIPIcs.CCC.2022.17 (cit. on p. 3).

[KKMP21] Robert Kleinberg, Oliver Korten, Daniel Mitropolsky, and Christos H. Papadimitriou. "Total Functions in the Polynomial Hierarchy". In: *ITCS*. Vol. 185. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021, 44:1–44:18. DOI: 10.4230/LIPIcs.ITCS.2021.44 (cit. on p. 1).

[KN08] Gillat Kol and Moni Naor. "Cryptography and Game Theory: Designing Protocols for Exchanging Information". In: *TCC*. Vol. 4948. Lecture Notes in Computer Science. Springer, 2008, pp. 320–339. DOI: 10.1007/978-3-540-78524-8_18 (cit. on pp. 10, 24).

[Kor21] Oliver Korten. "The Hardest Explicit Construction". In: *FOCS*. IEEE, 2021, pp. 433–444. DOI: 10.1109/FOCS52979.2021.00051 (cit. on pp. 1, 2, 3, 4).

[Kor22] Oliver Korten. "Derandomization from Time-Space Tradeoffs". In: *CCC*. Vol. 234. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 37:1–37:26 (cit. on pp. 1, 3).

[Kra19] Jan Krajíek. *Proof Complexity*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2019. DOI: 10.1017/9781108242066 (cit. on p. 3).

[Kra22] Jan Krajíek. "On the existence of strong proof complexity generators". In: *CoRR* abs/2208.11642 (2022). DOI: 10.48550/arXiv.2208.11642. arXiv: 2208.11642. URL: https://doi.org/10.48550/arXiv.2208.11642 (cit. on p. 3).

[LY22] Jiatu Li and Tianqi Yang. "$3.1n - o(n)$ circuit lower bounds for explicit functions". In: *STOC*. ACM, 2022, pp. 1180–1193. DOI: 10.1145/3519935.3519976 (cit. on p. 1).

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. "A Framework for Efficient and Composable Oblivious Transfer". In: *CRYPTO*. Vol. 5157. Lecture Notes in Computer Science. Springer, 2008, pp. 554–571. DOI: 10.1007/978-3-540-85174-5_31 (cit. on pp. 10, 11, 24, 25).

[Raz87] Alexander A Razborov. "Lower bounds on the size of bounded depth circuits over a complete basis with logical addition". In: *Mathematical Notes of the Academy of Sciences of the USSR* 41.4 (1987), pp. 333–338. DOI: 10.1007/BF01137685 (cit. on p. 1).

[Raz15] Alexander A Razborov. "Pseudorandom generators hard for k-DNF resolution and polynomial calculus resolution". In: *Annals of Mathematics* (2015), pp. 415–472 (cit. on p. 3).

[RR97] Alexander A. Razborov and Steven Rudich. "Natural Proofs". In: *J. Comput. Syst. Sci.* 55.1 (1997), pp. 24–35. DOI: 10.1006/jcss.1997.1494. URL: https://doi.org/10.1006/jcss.1997.1494 (cit. on pp. 4, 18).

[Reg09] Oded Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *J. ACM* 56.6 (2009), 34:1–34:40. DOI: 10.1145/1568318.1568324. URL: https://doi.org/10.1145/1568318.1568324 (cit. on pp. 7, 8, 10, 11, 12, 23, 24, 29).

[RSW22] Hanlin Ren, Rahul Santhanam, and Zhikun Wang. "On the Range Avoidance Problem for Circuits". In: *FOCS*. 2022. URL: https://eccc.weizmann.ac.il/report/2022/048 (cit. on pp. 1, 2, 3, 4).

[Ric08] Elaine Rich. *Automata, computability and complexity: theory and applications*. Pearson Prentice Hall Upper Saddle River, 2008 (cit. on p. 15).

[Rud97] Steven Rudich. "Super-bits, Demi-bits, and NP/qpoly-natural Proofs". In: *Randomization and Approximation Techniques in Computer Science, International Workshop, RANDOM'97, Bolognna, Italy, July 11-12. 1997, Proceedings*. Ed. by José D. P. Rolim. Vol. 1269. Lecture Notes in Computer Science. Springer, 1997, pp. 85–93. DOI: 10.1007/3-540-63248-4_8. URL: https://doi.org/10.1007/3-540-63248-4_8 (cit. on pp. 4, 6, 8, 17, 18, 20).

[SW05] Amit Sahai and Brent Waters. "Fuzzy Identity-Based Encryption". In: *EUROCRYPT*. Vol. 3494. Lecture Notes in Computer Science. Springer, 2005, pp. 457–473 (cit. on p. 17).

[Sha84] Adi Shamir. "Identity-Based Cryptosystems and Signature Schemes". In: *CRYPTO*. Vol. 196. Lecture Notes in Computer Science. Springer, 1984, pp. 47–53 (cit. on p. 17).

[Sha49] Claude E. Shannon. "The synthesis of two-terminal switching circuits". In: *Bell System technical journal* 28.1 (1949), pp. 59–98. DOI: 10.1002/j.1538-7305.1949.tb03624.x (cit. on p. 1).

[Smo87] Roman Smolensky. "Algebraic Methods in the Theory of Lower Bounds for Boolean Circuit Complexity". In: *STOC*. ACM, 1987, pp. 77–82. DOI: 10.1145/28395.28404 (cit. on p. 1).

[Tsa22] Rotem Tsabary. "Candidate Witness Encryption from Lattice Techniques". In: *CRYPTO (1)*. Vol. 13507. Lecture Notes in Computer Science. Springer, 2022, pp. 535–559. DOI: 10.1007/978-3-031-15802-5_19. URL: https://doi.org/10.1007/978-3-031-15802-5_19 (cit. on pp. 7, 17, 19).

[TZ23] Iddo Tzameret and Luming Zhang. "Stretching Demi-Bits and Nondeterministic-Secure Pseudorandomness". In: *CoRR* abs/2304.14700 (2023). DOI: 10.48550/arXiv.2304.14700. arXiv: 2304.14700. URL: https://doi.org/10.48550/arXiv.2304.14700 (cit. on pp. 4, 8).

[VWW22] Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. "Witness Encryption and Null-IO from Evasive LWE". In: *ASIACRYPT (1)*. Vol. 13791. Lecture Notes in Computer Science. Springer, 2022, pp. 195–221. DOI: 10.1007/978-3-031-22963-3_7. URL: https://doi.org/10.1007/978-3-031-22963-3_7 (cit. on pp. 7, 9, 17, 19).

[Val77] Leslie G. Valiant. "Graph-Theoretic Arguments in Low-Level Complexity". In: *MFCS*. Vol. 53. Lecture Notes in Computer Science. Springer, 1977, pp. 162–176. DOI: 10.1007/3-540-08353-7_135 (cit. on p. 1).

[Wil13] R. Ryan Williams. "Improving Exhaustive Search Implies Superpolynomial Lower Bounds". In: *SIAM J. Comput.* 42.3 (2013), pp. 1218–1244. DOI: 10.1137/10080703X (cit. on p. 2).

[Wil14] R. Ryan Williams. "Nonuniform ACC Circuit Lower Bounds". In: *J. ACM* 61.1 (2014), 2:1–2:32. DOI: 10.1145/2559903 (cit. on p. 3).

[Wil18] R. Ryan Williams. "New Algorithms and Lower Bounds for Circuits With Linear Threshold Gates". In: *Theory Comput.* 14.1 (2018), pp. 1–25. DOI: 10.4086/toc.2018.v014a017 (cit. on pp. 2, 3).

[Yao85] Andrew Chi-Chih Yao. "Separating the Polynomial-Time Hierarchy by Oracles (Preliminary Version)". In: *FOCS*. IEEE Computer Society, 1985, pp. 1–10. DOI: 10.1109/SFCS.1985.49 (cit. on p. 1).

A Lemmas for Probability

Lemma A.1. *For every $\varepsilon \in (0,1)$ and every sufficiently large m,*

$$\binom{m}{(1/2-\varepsilon)m} \geq 2^m \exp\left(-5\varepsilon^2 m\right).$$

Proof. We will use the Stirling approximation formula $\ln(n!) = n\ln n - n + o(n)$. By a straightforward calculation, we know that

$$\begin{aligned}
&\ln \binom{m}{(1/2-\varepsilon)m} \\
=\ & -\left(\frac{1}{2}-\varepsilon\right) m \ln\left(\frac{1}{2}-\varepsilon\right) - \left(\frac{1}{2}+\varepsilon\right) m \ln\left(\frac{1}{2}+\varepsilon\right) + o(m) \\
=\ & \left(\frac{1}{2}-\varepsilon\right) m \left(\ln 2 - \ln(1-2\varepsilon)\right) - \left(\frac{1}{2}+\varepsilon\right) m \left(\ln 2 + \ln(1+2\varepsilon)\right) + o(m) \\
=\ & m\ln 2 - \left(\frac{1}{2}-\varepsilon\right) m \ln(1-2\varepsilon) - \left(\frac{1}{2}+\varepsilon\right) m \ln(1+2\varepsilon) + o(m) \\
\geq\ & m\ln 2 - \left(\frac{1}{2}-\varepsilon\right) m \cdot (2\varepsilon) - \left(\frac{1}{2}+\varepsilon\right) m \cdot (-2\varepsilon) + o(m) \\
\geq\ & m\ln 2 - 4\varepsilon^2 m - o(m) \\
\geq\ & m\ln 2 - 5\varepsilon^2 m.
\end{aligned}$$

Therefore $\binom{m}{(1/2-\varepsilon)m} \geq 2^m \exp(-5\varepsilon^2 m)$. □

ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il