# CS 6810 Course Project: On Basing (Aux-input) OWFs on $\mathsf{NP} \not\subseteq \mathsf{BPP}$ via $\mathsf{NP}$-Hardness of Meta-complexity

Yanyi Liu

December 21, 2023

## 1 Introduction

A *one-way function* [DH76] (OWF) is a function $f$ that can be efficiently computed (in polynomial time), yet no probabilistic polynomial-time ($\mathsf{PPT}$) algorithm can invert $f$ with inverse polynomial probability for infinitely many input lengths $n$. Whether one-way functions exist is unequivocally the most important open problem in Cryptography (and arguably the most important open problem in the theory of computation, see e.g., [Lev03]): OWFs are both necessary [IL89] and sufficient for many of the most central cryptographic primitives and protocols [BM84, HILL99, GGM84, GM84, Rom90, Nao91, FS90, Blu82]. However, the existence of OWFs will immediately imply that the (worst-case) hardness of $\mathsf{NP}$ (or $\mathsf{NP} \not\subseteq \mathsf{BPP}$). (In addition, it will also imply that $\mathsf{NP}$ is average-case hard: efficient algorithms cannot decide $\mathsf{NP}$ languages even when the instances are sampled from an efficient distribution.) Thus, proving the existence of OWFs unconditionally seems far beyond reach. The question, then, is whether the existence of OWFs can be based on $\mathsf{NP} \not\subseteq \mathsf{BPP}$. Indeed, this question, originating in the seminal work of Diffie and Hellman [DH76], is often referred to as the "holy-grail" of Cryptography. Observe that to resolve the holy-grail, it is sufficient (and necessary) to prove that (1) the worst-case hardness of $\mathsf{NP}$ implies the average-case hardness of $\mathsf{NP}$ and (2) the average-case hardness of $\mathsf{NP}$ gives Cryptography (or to exclude "Heuristica" and "Pessiland" in Impagliazzo's Five Worlds [Imp95]).

Recently, there is a sequence of works trying to make progress towards showing (1) and (2). These works, starting with the work by Hirahara [Hir18], are closely related to meta-complexity problems. Meta-complexity problems are problems themselves concerning the complexity (of strings). One notable example is the time-bounded Kolmogorov complexity problem ($\mathsf{MK^tP}$) (parametrized by the running time bound $t$), in which given a string $x$, we are asked to decide whether the length of the shortest $t(|x|)$-time-bounded program that produces the string $x$ is at most, e.g., $|x|/2$. Hirahara [Hir18] presented a worst-case to average-case reduction for an approximation variant of $\mathsf{MK^tP}$, and thus roughly speaking, to deduce (1), we only need to show the $\mathsf{NP}$-hardness of (the approximation variant of) $\mathsf{MK^tP}$.

In addition, a recent result by Liu and Pass [LP20] showed that average-case hardness of $\mathsf{MK^tP}$ is equivalent to the existence of OWFs. However, the average-case hardness needed in [LP20] is slightly different from the notion in [Hir18]. (Specifically, [LP20] considers the notion of two-sided average-case hardness whereas [Hir18] considers errorless average-case hardness.) Taken together, it would seem that resolving the holy-grail of Cryptography boils down to proving $\mathsf{NP}$-hardness of $\mathsf{MK^tP}$ and filling the gap between two notions of average-case hardness for $\mathsf{MK^tP}$.

Somewhat surprisingly, a very recent result by Hirahara [Hir23] formalized this folklore intuition. Moreover, using his proof techniques, we are able to show that the gap between two notions of

average-case hardness can be filled if we consider a stronger variant of NP-hardness of $\mathsf{MK^tP}$. Thus, if we assume the (stronger variant of) NP-hardness of $\mathsf{MK^tP}$, then just $\mathsf{NP} \notin \mathsf{BPP}$ implies the existence of OWFs. The stronger variant requires that there exists a single reduction that proves NP-hardness of $\mathsf{MK^tP}$ for all sufficiently large polynomial $t$. However, there is evidence that this stronger variant of NP-hardness of $\mathsf{MK^tP}$ is *unlikely* to holds [SS22]. To get around this impossibility result, Hirahara [Hir23] instead considered a "distributional" variant of $\mathsf{MK^tP}$. [Hir23] proved that the stronger variant of NP-hardness of "distributional" $\mathsf{MK^tP}$ is a plausible assumption (implied by the existence of OWFs), and assuming it holds, $\mathsf{NP} \not\subseteq \mathsf{BPP}$ implies the existence of OWFs.

In this course project, we aim to survey the ideas behind [Hir23]. We will focus on the promise problem $\mathsf{Gap}_\infty\mathsf{MK^tP}$, formally defined in Section 2.1, which in our eyes is a good candidate problem to be used to reflect ideas in [Hir23]. The other important notions used in this report can also be found in Section 2.

We will generalize [SS22] to also show that the problem $\mathsf{Gap}_\infty\mathsf{MK^tP}$ is unlike to be NP-hard via a parametric-honest reduction (formally stated and proved in Section 3). We say that a reduction $R$ is *parametric-honest* if $|R(x)| \geq |x|^{\Omega(1)}$ for all strings $x$.

**Theorem 1.1.** *Assume that* $\mathsf{Gap}_\infty\mathsf{MK^tP}$ *is* NP*-hard (via a parametric-honest reduction) and* $\mathsf{AM} = \mathsf{NP}$*. Then,* $\mathsf{NE} = \mathsf{coNE}$*.*

We will present proofs (or proof sketches) for the statement that if $\mathsf{Gap}_\infty\mathsf{MK^tP}$ is NP-hard, then $\mathsf{NP} \not\subseteq \mathsf{BPP}$ implies the existence of the so-called "auxiliary-input" one-way functions (formally defined in Section 2.4).

**Theorem 1.2.** *Assume that* $\mathsf{NP} \notin \mathsf{BPP}$ *and* $\mathsf{Gap}_\infty\mathsf{MK^tP}$ *is* NP*-hard for some polynomial* $t$ *(via a parametric-honest reduction). Then, there exists an auxiliary-input OWF.*

We remark that even if the assumption in the above theorem is unlikely to hold and we only get auxiliary-input one-way functions (as opposed to standard OWFs), we still find many ideas in the proof of the above theorem very interesting and therefore decide to present the proof. Theorem 1.2 is proved by combining the ideas in [Hir18] (briefly reviewed in Section 4) and [Nan21] (briefly reviewed in Section 5). Finally, we give a proof sketch for Theorem 1.2 in Section 6.

## 2 Preliminaries

Let $\mathcal{Q} = \{\mathcal{Q}_x\}_{x\in\{0,1\}^*}$ be a family of distributions. $\mathcal{Q}$ is said to be *polynomial-time samplable* if there exists a randomized polynomial-time machine $M$ such that for every $x \in \{0,1\}^*$, $M(x)$ samples the distribution $\mathcal{Q}_x$.

### 2.1 Kolmogorov complexity

Let $U$ be some fixed Universal Turing machine that can emulate any Turing machine $M$ with polynomial overhead. Given a description $\Pi \in \{0,1\}^*$ which encodes a pair $(M, w)$ where $M$ is a (single-tape) Turing machine and $w \in \{0,1\}^*$ is an input, let $U(\Pi, 1^t)$ denote the output of $M(w)$ when emulated on $U$ for $t$ steps. Note that (by assumption that $U$ only has polynomial overhead) $U(\Pi, 1^t)$ can be computed in time $\mathsf{poly}(|\Pi|, t)$.

The *$t$-time bounded Kolmogorov Complexity, $K^t(x)$, of a string $x$* [Kol68, Sip83, Tra84, Ko86] is defined as the length of the shortest description $\Pi$ such that $U(\Pi, 1^t) = x$:

$$K^t(x) = \min_{\Pi\in\{0,1\}^*} \{|\Pi| : U(\Pi, 1^{t(|x|)}) = x\}.$$

When there is no running time bound, we recover the notion of *Kolmogorov Complexity*, which is defined to be

$$K(x) = \min_{\Pi \in \{0,1\}^*} \{|\Pi| : \exists t \in \mathbb{N}, U(\Pi, 1^t) = x\}.$$

In this report, we are going to focus on the promise problem $\mathsf{Gap}_\infty\mathsf{MK^tP}$, defined as follows. For any polynomial $t(n) \geq 0$, let $\mathsf{Gap}_\infty\mathsf{MK^tP}$ denote the following promise problem where

- YES instances: $x \in \{0,1\}^*$, $K^t(x) \leq \sqrt{|x|}$.
- NO instances: $x \in \{0,1\}^*$, $K(x) \geq \frac{2}{3}|x|$.

## 2.2 Hitting Set Generators

We recall the notion of *hitting set generators (HSG)* [ACR98, ACRT99] that we consider in this report. Roughly speaking, an efficient computable function $H : \{0,1\}^{\ell(n)} \to \{0,1\}^n$ is said to be a HSG if no PPT attacker $A$ "avoids" $H$ for all $n \in \mathbb{N}$. And we say that $A$ *avoids* $H$ on strings of length $n$ if $A$ always output 0 on strings in the range of $H(\mathcal{U}_{\ell(n)})$ but outputs 1 with probability at least $1/2$ over a random string. We mention that hitting set generators are usually used to derandomize one-sided randomized algorithms and people often only require that no a-priori poly-time bounded machines (or circuits) avoid $H$. In this report, we instead look at "cryptographic" HSGs where we ask its security to hold against all PPT machines.

**Definition 2.1** (Hitting set generators (HSG))**.** *We say that an efficiently computable function $H : \{0,1\}^{\ell(n)} \to \{0,1\}^n$ is a* hitting set generator (HSG) *if $\ell(n) < n$ and there is no* PPT *attacker $A$ that avoids $H$ for all $n \in \mathbb{N}$, where $A$ avoids $H$ on strings of length $n$ if for every $x \in \{0,1\}^{\ell(n)}$, $y = H(x)$, it holds that*

$$\Pr[A(y) = 1] \leq 1/3$$

*and*

$$\Pr[A(\mathcal{U}_n) = 1] \geq 1/2$$

## 2.3 Karp Reductions and Weakly Black-box Reductions

**Karp reductions** For any languages $L, L'$ and a randomized poly-time algorithm $R$, we say that $R$ is a Karp reduction from $L$ to $L'$ if $|R(x)| = |R(x')|$ for any $x, x', |x| = |x'|$, and for any $x \in L$, $n = |x|$,

$$\Pr[R(x) \in L'] \geq 2/3$$

and for any $x \notin L$, $n = |x|$,

$$\Pr[R(x) \notin L'] \geq 2/3$$

We can define the reduction $R$ in a similar way when $L$ or $L'$ is a promise problem (as opposed to a language). We say that the reduction $R$ is *parametric-honest* if there exists a constant $\gamma > 0$ such that $|R(x)| \geq |x|^\gamma$ for every $x \in \{0,1\}^*$.

**(Weakly) black-box reductions** We turn to introducing the notion of *weakly black-box reduction* we rely on. Roughly speaking, a black-box reduction from a task $A$ to a task $B$ is a (probabilistic) poly-time oracle machine $R$ such that for every oracle $\mathcal{O}$ that solves the task $B$, $R^{\mathcal{O}}$ solves the task $A$. A weakly black-box reduction is just a black-box reduction except that it puts some restrictions on $\mathcal{O}$: It only works when $\mathcal{O}$ can be implemented by a "short" machine. In this report, we will focus on such reductions from a language $L$ to avoiding some HSG $H : \{0,1\}^{\ell(m)} \to \{0,1\}^m$.

**Definition 2.2** ((Weakly) black-box reduction)**.** *Let $L$ be a language and $H : \{0,1\}^{\ell(m)} \to \{0,1\}^m$ be a efficiently computable function. We say that a probabilistic poly-time oracle machine $R$ is a* black-box reduction from $L$ to avoiding the HSG $H$ *if for every $n \in \mathbb{N}$, $R$ on input length $n$ only make oracle queries on strings of length $m = m(n)$, and for every $x \in \{0,1\}^n$, every $\mathcal{O}$ such that $\mathcal{O}$ avoids the HSG $H$ on strings of length $m$, it holds that*

$$\Pr[R^{\mathcal{O}}(x) = L(x)] \geq 2/3$$

*We say that $R$ is just* weakly black-box *if the above condition only holds when $\mathcal{O}$ can be implemented by a probabilistic machine of description length $\leq 2m$.*

*In addition, we say that $R$ is* non-adaptive *if the queries $R$ makes to $\mathcal{O}$ do not depend on the answers to the previous queries that $R$ makes to $\mathcal{O}$ before them. We say that $R$ is* length-increasing *if it holds that $m \geq n$. We say that $R$ is* parametric-honest *if it holds that $m \geq n^{\Omega(1)}$.*

We remark that we can define weakly black-box reduction from a promise problem $\Pi$ to avoiding some HSG in a similar way.

### 2.4 (Auxiliary-input) One-way Functions

We start by recalling the standard definition of one-way functions (OWFs).

**Definition 2.3** (OWFs)**.** *Let $f : \{0,1\}^* \to \{0,1\}^*$ be a polynomial-time computable function. $f$ is said to be a* one-way function (OWF) *if for every* PPT *algorithm $\mathcal{A}$, there exists a negligible function $\mu$ such that for all $n \in \mathbb{N}$,*

$$\Pr[x \leftarrow \{0,1\}^n; y = f(x) : A(1^n, y) \in f^{-1}(f(x))] \leq \mu(n)$$

We move on to introducing the notion of *auxiliary-input one-way functions*. Roughly speaking, an auxiliary-input OWF is a family of functions $\{f_x\}_{x\in\{0,1\}^*}$ (indexed by the auxiliary input) such that for every attacker $A$, there are infinitely many indices $x$ such that $f_x$ will be one-way with respect to $A$.

**Definition 2.4** (Auxiliary-input OWFs)**.** *Let $f = \{f_x : \{0,1\}^{p_1(|x|)} \to \{0,1\}^{p_2(|x|)}\}_{x\in\{0,1\}^*}$ be a family of polynomial-time computable functions where $p_1, p_2$ are polynomials. $f$ is said to be an* auxiliary-input one-way function (auxiliary-input OWF) *if for every* PPT *algorithm $\mathcal{A}$, there exists a negligible function $\mu$ and an infinitely sequence of auxiliary-input $I \subseteq \{0,1\}^*$ such that for every $x \in I$, $n = |x|$,*

$$\Pr[z \leftarrow \{0,1\}^{p_1(n)}; y = f_x(z) : A(1^n, x, y) \in f_x^{-1}(f_x(z))] \leq \mu(n)$$

## 3 Impossibility Result for the NP-hardness of $\mathsf{Gap}_\infty\mathsf{MK^tP}$

In this section, we aim to show that $\mathsf{Gap}_\infty\mathsf{MK^tP}$ is unlikely to be NP-hard. Towards this, we will show that if there exists a Karp reduction proving the NP-hardness of $\mathsf{Gap}_\infty\mathsf{MK^tP}$ and additionally $\mathsf{AM} = \mathsf{NP}$, then $\mathsf{NE} = \mathsf{coNE}$. Note that $\mathsf{AM} = \mathsf{NP}$ is very likely to hold since it follows from plausible circuit lower bounds [KVM02, MV05], and on the other hand, the conclusion $\mathsf{NE} = \mathsf{coNE}$ is very unlikely to be true. Taken together, we conclude that it's quite unlikely for such a reduction to exist, and $\mathsf{Gap}_\infty\mathsf{MK^tP}$ is unlikely to be NP-hard.

**Some Additional Preliminaries** We start by recalling the entropy approximation problem [GSV99]. The entropy approximation problem is a promise problem where

- YES instances: $(C, k)$ where $C$ is a circuit mapping $\{0,1\}^n \to \{0,1\}^m$, $k$ is an integer, and $H(C(\mathcal{U}_n)) \geq k$.
- NO instances: $(C, k)$ where $C$ is a circuit mapping $\{0,1\}^n \to \{0,1\}^m$, $k$ is an integer, and $H(C(\mathcal{U}_n)) \leq k-1$.

In addition, the entropy approximation problem is in $\mathsf{AM} \cap \mathsf{coAM}$ [AH91].

We are now ready to state the main theorem of this section.

**Theorem 3.1** ([SS22, AHT23]). *Assume that there exist a polynomial $t$, and a parametric-honest randomized Karp reduction $R$ from $\mathsf{SAT}$ to $\mathsf{Gap}_\infty\mathsf{MK^tP}$, and $\mathsf{AM} = \mathsf{NP}$. Then, $\mathsf{NE} = \mathsf{coNE}$.*

**Proof:** By a standard padding argument, to show that $\mathsf{NE} = \mathsf{coNE}$, it suffices to prove that all tally languages $\in \mathsf{NP}$ are in $\mathsf{NP} \cap \mathsf{coNP}$. Consider any tally $\mathsf{NP}$ language $L$. By the Cook-Levin Theorem, there exists a (parametric-honest) deterministic poly-time Karp reduction $R'$ that reduces $L$ to $\mathsf{SAT}$. Recall that by our assumption, there exists a (parametric-honest) randomized poly-time Karp reduction $R$ that reduces $\mathsf{SAT}$ to $\mathsf{Gap}_\infty\mathsf{MK^tP}$. By combining the two reduction, we obtain a (parametric-honest) randomized poly-time Karp reduction $M$ that reduces $L$ to $\mathsf{Gap}_\infty\mathsf{MK^tP}$. We will use the reduction $M$ to construct a reduction from $L$ to the complement of the entropy approximation problem.

For any instance $1^n$ in the tally language, let $C$ be the circuit that computes the reduction $M(1^n)$: $C$ takes as input a random tape of $M(1^n)$, $r \leftarrow \mathcal{R}$ (where $\mathcal{R}$ is the distribution of the random tape), and emulates $M(1^n)$ on the random tape $r$. Let $m$ denote the output length of $M(1^n)$, and $k = \frac{7}{18}m$. (Recall that $M$ is parametric-honest, and thus $m \geq n^{\Omega(1)}$.)

We first show that if $1^n \in L$, then $H(C(\mathcal{R})) \leq k-1$ (when $n$ is sufficiently large). Since $1^n \in L$, it follows from the correctness of $M$ that $K^t(M(1^n)) \leq \sqrt{m}$ with probability $\geq 2/3$. And thus $K^t(C(\mathcal{R})) \leq \sqrt{m}$ with probability $\geq 2/3$. Intuitively, this says that with probability $\geq 2/3$ $C(\mathcal{R})$ will have very small support (of size $\leq 2^{\sqrt{m}}$) and therefore it must have small entropy. Let us proceed to a formal proof. Let $flag$ be a binary random variable jointly distributed with $\mathcal{R}$ such that $flag = \mathsf{low}$ if $K^t(C(\mathcal{R})) \leq \sqrt{m}$, and otherwise $flag = \mathsf{high}$. Note that $H(C(\mathcal{R}) \mid flag = \mathsf{low}) \leq \sqrt{m}+1$ since conditioned on $flag = \mathsf{low}$, $C(\mathcal{R})$ is a distribution over strings with $K^t$-complexity at most $\sqrt{m}$; by a standard counting argument, it follows that the support of $C(\mathcal{R})$ is at most $2^{\sqrt{m}+1}$ and thus has entropy at most $2^{\sqrt{m}+1}$. It follows that

$$\begin{aligned}
H(C(\mathcal{R})) \leq H(C(\mathcal{R}), flag) &= H(flag) + H(C(\mathcal{R}) \mid flag) \\
&= H(flag) + H(C(\mathcal{R}) \mid flag = \mathsf{low}) \Pr[flag = \mathsf{low}] \\
&\quad + H(C(\mathcal{R}) \mid flag = \mathsf{high}) \Pr[flag = \mathsf{high}] \\
&\leq 1 + H(C(\mathcal{R}) \mid flag = \mathsf{low}) + 1/3 \cdot H(C(\mathcal{R}) \mid flag = \mathsf{high}) \\
&\leq 1 + \sqrt{m} + 1 + m/3 \\
&\leq k - 1.
\end{aligned}$$

We turn to proving that if $1^n \notin L$, then $H(C(\mathcal{R})) \geq k$ (when $n$ is sufficiently large). Since $1^n \notin L$, it follows from the correctness of $M$ such that $K(M(1^n)) \geq \frac{2}{3}m$ with probability at least $2/3$. Thus, $K(C(\mathcal{R})) \geq \frac{2}{3}m$ with probability at least $2/3$. Notice that since $C$ is of very small description length, if any string $y$ is of high $K$-complexity, the probability that $C(\mathcal{R})$ outputs $y$ will be tiny. If this holds for a $2/3$ fraction of output of $C(\mathcal{R})$, then we can conclude that $C(\mathcal{R})$ has high

entropy. We next continue to the formal proof. We claim that for any $y \in \{0,1\}^m$, if $K(y) \geq \frac{2}{3}m$, then $\Pr[C(\mathcal{R}) = y] \leq 2^{-2m/3+O(\log m)}$. If this claim holds, it follows that

$$\begin{aligned} H(C(\mathcal{R})) &= \sum_{y \in C(\mathcal{R})} \Pr[C(\mathcal{R}) = y] \log \frac{1}{\Pr[C(\mathcal{R}) = y]} \\ &\geq \sum_{y \in C(\mathcal{R}), K(y) \geq 2m/3} \Pr[C(\mathcal{R}) = y] \log \frac{1}{\Pr[C(\mathcal{R}) = y]} \\ &\geq \frac{2}{3} \cdot (2m/3 - O(\log m)) \\ &\geq \frac{4}{9}m - O(\log m) \geq k. \end{aligned}$$

Note that the claim follows from the so-called "Coding Theorem". For completeness, we here sketch its proof. Assume for contradiction that there exists some $y \in \{0,1\}^m$, $k(y) \geq \frac{2}{3}m$ but $\Pr[C(\mathcal{R}) = y] \geq 2^{-2m/3+O(\log m)}$. Consider the set $V : \{z \in C(\mathcal{R}) : \Pr[C(\mathcal{R}) = z] \geq 2^{-2m/3+O(\log m)}\}$. It follows that $y \in V$ and $V$ contains at most $2^{2m/3-O(\log m)}$ strings. Therefore, $y$ can be produced by a program with the integer $n$ and the code of $M$ (from which it can compute the circuit $C$), and the index of $y$ in $V$. And we conclude that $K(y) \leq O(\log n) + 2m/3 - O(\log m) < 2m/3$ which is a contradiction.

By the above two arguments, we have that the tally language $L$ reduces to the complement of the entropy approximation problem, and thus $L \in \mathsf{AM} \cap \mathsf{coAM} = \mathsf{NP} \cap \mathsf{coNP}$ (by our assumption that $\mathsf{AM} = \mathsf{NP}$). ■

## 4 Weakly Black-box Reduction from $\mathsf{Gap}_\infty\mathsf{MK^tP}$ to HSG

We turn to briefly reviewing the ideas behind [Hir18] in which he showed that there exists a (weakly black-box) worst-case to average-case reduction for the problem of approximating $K^t$-complexity of strings. We will show that the same ideas enable us to also show that there exists a weakly black-box reduction from $\mathsf{Gap}_\infty\mathsf{MK^tP}$ to avoiding HSGs.

**Theorem 4.1** ([Hir18])**.** *For any polynomial $t(n) \geq n$, there exist a HSG $H : \{0,1\}^{m/2} \to \{0,1\}^m$ and a parametric-honest non-adaptive weakly black-box reduction $R$ from $\mathsf{Gap}_\infty\mathsf{MK^tP}$ to avoiding $H$.*

**Proof Sketch.** We will consider a HSG $H$ defined as follows. For any $m \in \mathbb{N}$, $H$ picks a random program of length $\leq m/2$, runs the program for $\mathsf{poly}(t(m))$ steps, and finally $H$ outputs whatever the program outputs. $H$ simply aborts (or outputs an arbitrary $m$-bit string) if the program does not halt within $\mathsf{poly}(t(m))$ steps or it outputs a string that is not of length $m$. $H$ is also referred to as the universal hitting set generator. We will show that there exists a weakly black-box reduction from $\mathsf{Gap}_\infty\mathsf{MK^tP}$ to avoiding $H$.

Our reduction, $R$, on input a string $x \in \{0,1\}^n$, needs to decide whether $K^t(x) \leq \sqrt{n}$ or $K(x) \geq 2n/3$ given an oracle $\mathcal{O}$ that avoids $H$. The construction of $R$ relies on the Nisan-Wigderson generator and a good error correcting code, and $R$ proceeds as follows. $R$ first computes $x' \in \{0,1\}^{\mathsf{poly}(n)}$, the error correcting code of $x$, and then uses $x'$ as the truth table of a hard function to instantiate the Nisan-Wigderson generator $\mathsf{NW}^{x'} : \{0,1\}^{O(\log n)} \to \{0,1\}^{O(\sqrt{n})}$. Finally, $R$ checks whether the oracle $\mathcal{O}$ distinguishes between the output of $\mathsf{NW}^{x'}$ and $\mathcal{U}_{O(\sqrt{n})}$ with advantage at least $1/6$. Let $m = O(\sqrt{n})$ be the output length of $\mathsf{NW}^{x'}$. Notice that $R$ is parametric-honest since it only makes queries to $\mathcal{O}$ on strings of length $m$, $R$ is also non-adaptive since the queries $R$ makes to $\mathcal{O}$ only depend on the input $x$, and also notice that $R$ runs in polynomial time.

To show that $R$ is indeed a weakly black-box reduction, we need to show that $R$ decides $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ if the oracle $\mathcal{O}$ avoids $H$ and $\mathcal{O}$ can be implemented by a machine of length $\leq 2m$. We first show that if $K^t(x) \leq \sqrt{n}$, then $\mathcal{O}$ will be guaranteed to distinguish between $\mathsf{NW}^{x'}$ and the uniform distribution. Since $K^t(x) \leq \sqrt{n}$, there exists a program of length $\leq \sqrt{n}$ that generates $x$ within time $t(n)$. Notice that the Nisan-Wigderson generator and the error correcting code can be implemented using a program with running time $\mathsf{poly}(n)$ and description length $O(\log n)$, and the seed of the NW generator is also of length $O(\log n)$. Thus, each output string of $\mathsf{NW}^{x'}$ can be produced by a program of length $\leq \sqrt{n} + O(\log n) \leq m/2$ within time $t(n) + \mathsf{poly}(n) \in \mathsf{poly}(t(m))$. It follows that the output of $\mathsf{NW}^{x'}$ will be contained in the range of $H(\mathcal{U}_{m/2})$. Since $\mathcal{O}$ avoids $H$, $\mathcal{O}$ will output 1 with probability at least $1/2$ on input a uniform random string, and it will always output 0 on the output of $H$ (and thus always output 0 on the output of $\mathsf{NW}^{x'}$). Therefore, we conclude that $\mathcal{O}$ will distinguish between $\mathsf{NW}^{x'}$ and the uniform distribution.

It remains to show that if $K(x) \geq 2n/3$, $\mathcal{O}$ can never distinguish between $\mathsf{NW}^{x'}$ and the uniform distribution. Roughly speaking, this claim follows from the security of NW generator (and also the correctness of the error correcting code), and its proof heavily relies on the fact that $\mathcal{O}$ has a relatively small description length. Assume for contradiction that $\mathcal{O}$ distinguishes between $\mathsf{NW}^{x'}$ and the uniform distribution. It follows from the reconstruction property of the NW generator that we can approximately compute the string $x'$ (with accuracy $\geq 1/2 + O(1/m)$) given the oracle $\mathcal{O}$ and an advice string of length, e.g., $\leq m\sqrt{m}$. Then, we can use the decoding algorithm for the error correcting code to compute the string $x$ exactly with another advice string of length $\leq O(\log n)$ (since we will need the error correcting code to be list-decodable, and the advice string in this step is to specify which string in the list the string $x$ will be). Taken together, we can reconstruct $x$ with the oracle $\mathcal{O}$ (which can be implemented using $2m$ bits) and $m\sqrt{m} + O(\log m)$ advice bits, which contradicts to the assumption that $K(x) \geq 2n/3$.

## 5 Auxiliary-input OWFs from Reductions to HSGs

We will show that if there exists a (length-increasing) weakly black-box reduction from some promise problem $\Pi$ to avoiding a HSG $H$, then we can obtain an auxiliary-input OWF assuming that the problem $\Pi$ is worst-case hard. This proof builds on the ideas in [GT00, Nan21, Hir23].

**Theorem 5.1.** *For any promise problem $\Pi$, any HSG $H : \{0,1\}^{m/2} \to \{0,1\}^m$, assume that there exists a length-increasing non-adaptive weakly black-box reduction from $\Pi$ to avoiding $H$ and $\Pi \notin \mathsf{prBPP}$. Then, there exists an auxiliary-input OWF.*

We highlight here that one may wonder if we can obtain auxiliary-input OWF from worst-case hardness of $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ by combining the above theorem together with the weakly black-box reduction from $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ to HSG we constructed in Section 4. Unfortunately, the answer is no. The reason for this is that the reduction we obtained in Section 4 (which on input of length $n$ makes queries on strings of length roughly $O(\sqrt{n})$) is not length-increasing, whereas to apply the above theorem, we need a weakly black-box reduction that is length-increasing.

It is instructive to recall that Impagliazzo and Levin [IL90] showed that approximate counting is possible assuming that there is no OWF. In more detail, they showed that assuming no OWF, for randomized machine $M$, for any string $x$, we can approximately count how many random tapes $r$ there are that will lead $M(1^n)$ to output the string $x$. We will need the auxiliary-input variant of [IL90].

**Lemma 5.1** (Auxiliary-input approximate counter [IL90])**.** *Assume that there is no auxiliary-input OWF. For every efficiently samplable family of distributions $\{\mathcal{Q}_x\}_{x\in\{0,1\}^*}$, any polynomial $p(n)$, there*

*exists a* PPT *machine* $A$ *such that for all* $x \in \{0,1\}^*$, $n = |x|$,

$$\Pr[q \leftarrow \mathcal{Q}_x : (1 - 1/p(n))\mathcal{Q}_x(q) \leq A(x,q) \leq (1 + 1/p(n))\mathcal{Q}_x(q)] \geq 1 - 1/p(n)$$

*where* $\mathcal{Q}_x(q)$ *denotes* $\Pr[\mathcal{Q}_x = q]$.

We are now ready to present a proof for Theorem 5.1.

**Proof:** (of Theorem 5.1) Let $t$ be the running time of the reduction $R$.

We consider the efficiently samplable family of distributions $\{Q_x\}_{x\in\{0,1\}^*}$ defined as the follows. Given $x \in \{0,1\}^*$, we sample a random tape $r$ for the reduction $R(x)$, and we let $q_1, \ldots, q_{t(|x|)}$ be the strings such that the $i$-the query that $R(x)$ (with the random tape $r$) makes to the oracle is on $q_i$. (Note that we are able to extract $q_1, \ldots, q_{t(|x|)}$ since $R$ is a non-adaptive reduction.) Let $q$ be a random element in $\{q_1, \ldots, q_{t(|x|)}\}$, and we simply let $\mathcal{Q}_x$ be the distribution of $q$ (given a uniformly distributed random tape $r$).

For any $x \in \{0,1\}^*$, let $n = |x|$. Let $m$ be the length of the oracle queries that $R(x)$ makes. Since $R$ is length-increasing, it holds that $m \geq n$. We start by constructing an oracle $\mathcal{O}_1$ such that $\mathcal{O}_1$ avoids the HSG $H$ and $\mathcal{O}_1$ can be implemented by an (inefficient) program of length at most $2n \leq 2m$. If so, we can conclude that $R^{\mathcal{O}_1}(x)$ will output $\Pi(x)$ with probability at least $2/3$ due to the correctness of $R$. Although it is assumed that $\Pi \notin$ prBPP, there is no contradiction yet since the implementation of $\mathcal{O}_1$ might be inefficient. Notice that we are going to allow some of entries in $\mathcal{O}_1$ to be set to $*$, meaning that $\mathcal{O}_1$ is undefined on those locations. We say that such an oracle avoids the HSG $H$ if no matter how we assign each $*$ to 0 or 1, it will still avoid $H$.

Let $\theta = 2t(n)^3$. We say that a string $q \in \{0,1\}^m$ is "light" (with respect to $\mathcal{Q}_x$) if

$$\mathcal{Q}_x(q) \leq \theta \cdot 2^{-m}$$

And we say that $q$ is "heavy" if

$$\mathcal{Q}_x(q) \geq 4\theta \cdot 2^{-m}$$

where $\mathcal{Q}_x(q)$ is defined to be $\Pr[\mathcal{Q}_x = q]$. We turn to defining the oracle $\mathcal{O}_1$. For every $q \in \{0,1\}^m$, we define

$$\mathcal{O}_1(q) = \begin{cases} 1 & \text{if } q \text{ is light and } q \notin Im(H) \\ 0 & \text{if } q \text{ is heavy or } q \in Im(H) \\ * & \text{otherwise} \end{cases}$$

where $Im(H)$ denote the range (or the set of image) of the HSG $H$. Note that $\mathcal{O}_1$ can be implemented by an inefficient machine that has the instance $x$, the code of $R$ and $H$ hardcoded in its code. Thus, the description length of $\mathcal{O}_1$ is at most $n + O(\log n) \leq 2n$.

It remains to show that $\mathcal{O}_1$ avoids the HSG $H$. It follows from the definition of $\mathcal{O}_1$ that $\mathcal{O}_1$ will output 0 on every string $\in Im(H)$. To show that $\mathcal{O}_1$ will output 1 with probability at least $1/2$ on input $\mathcal{U}_m$, we notice that $\mathcal{O}_1(q)$ won't output 1 if $q \in Im(H)$ or $q$ is not light. The probability that $q \in Im(H)$ is at most $\frac{2^{m/2}}{2^m} \leq 2^{-m/2}$ since the HSG $H$ has seed length only $m/2$. The probability that a random $q \in \{0,1\}^m$ is not light is at most $\frac{1}{\theta}$ since there are at most $\frac{1}{\theta 2^{-m}}$ strings that are not light. Combing the above two arguments, and by a Union Bound, we conclude that $\mathcal{O}_1$ does not output 1 with probability at most

$$2^{-m/2} + \frac{1}{\theta} \leq \frac{1}{2}$$

which concludes that $\mathcal{O}_1$ avoids $H$.

Next, we are going to construct another oracle $\mathcal{O}_2$ such that $R(x)$ can barely distinguish between $\mathcal{O}_1$ and $\mathcal{O}_2$ (with probability $\leq \frac{1}{t(n)}$) and $\mathcal{O}_2$ can be efficiently implemented given $x$. If this is the case, since recall that $R^{\mathcal{O}_1}(x)$ will output $\Pi(x)$ with probability at least $2/3$, it follows that $R^{\mathcal{O}_2}(x)$

will output $\Pi(x)$ with probability at least $2/3 - \frac{1}{t(n)}$, which (together with the fact that $R^{\mathcal{O}_2}(x)$ can be efficiently computed) concludes that $\Pi \in \mathsf{prBPP}$, a contradiction.

We proceed to the construction of $\mathcal{O}_2$. Since there is no auxiliary-input OWF, by Lemma 5.1, there exists a $\mathsf{PPT}$ machine $A$ such that for every $x \in \{0,1\}^*$, $n = |x|$, with probability at least $1 - \frac{1}{\theta(n)}$ over $q \leftarrow \mathcal{Q}_x$, we have that

$$(1 - \frac{1}{\theta(n)})\mathcal{Q}_x(q) \leq A(x,q) \leq (1 + \frac{1}{\theta(n)})\mathcal{Q}_x(q)$$

We will use the machine $A$ to define $\mathcal{O}_2$. For every $q \in \{0,1\}^m$, we define

$$\mathcal{O}_2(q) = \begin{cases} 1 & \text{if } A(x,q) \leq 2\theta \cdot 2^{-m} \\ 0 & \text{if } A(x,q) > 2\theta \cdot 2^{-m} \end{cases}$$

It follows from the definition of $\mathcal{O}_2$ that it can be implemented by an efficient machine given the instance $x$. We turn to proving that $R(x)$ can distinguish between $\mathcal{O}_1$ and $\mathcal{O}_2$ with probability at most $\frac{1}{t(n)}$. For each query $q$, $R(x)$ will distinguish between $\mathcal{O}_1$ and $\mathcal{O}_2$ if the approximate counter $A$ fails on $q$ (which happens with probability $\leq t(n) \cdot \frac{1}{\theta}$), or $q$ is light but $q \in Im(H)$, which happens with probability at most

$$(t(n) \cdot \theta \frac{1}{2^m}) \cdot 2^{m/2} \leq \frac{1}{\theta}$$

By a union bound, the probability that there exists some query that makes $R(x)$ distinguishes between $\mathcal{O}_1$ and $\mathcal{O}_2$ is at most

$$t(n) \cdot (t(n) \cdot \frac{1}{\theta} + \frac{1}{\theta}) \leq \frac{1}{t(n)}$$

which concludes the proof. ■

# 6 Auxiliary-input OWF from NP-hardness of $\mathsf{Gap}_\infty\mathsf{MK^tP}$

Finally, we put everything all together and present a proof sketch for the main theorem of this report. We will show that assuming the worst-case hardness of $\mathsf{NP}$ and the $\mathsf{NP}$-hardness of $\mathsf{Gap}_\infty\mathsf{MK^tP}$, there exists an auxiliary-input OWF.

**Theorem 6.1.** *Assume that* $\mathsf{NP} \not\subseteq \mathsf{BPP}$ *and* $\mathsf{Gap}_\infty\mathsf{MK^tP}$ *is* $\mathsf{NP}$*-hard for some polynomial* $t$ *(via a parametric-honest reduction). Then, there exists an auxiliary-input OWF.*

**Proof Sketch** We first give a potential approach of proving this theorem that does not really work. Since $\mathsf{NP} \not\subseteq \mathsf{BPP}$ and $\mathsf{Gap}_\infty\mathsf{MK^tP}$ is $\mathsf{NP}$-hard, it follows that $\mathsf{Gap}_\infty\mathsf{MK^tP} \not\subseteq \mathsf{prBPP}$. By Theorem 4.1, there exists a weakly black-box reduction from $\mathsf{Gap}_\infty\mathsf{MK^tP}$ to a HSG $H$. Given this weakly black-box reduction, together with the fact that $\mathsf{Gap}_\infty\mathsf{MK^tP} \not\subseteq \mathsf{prBPP}$, by Theorem 5.1, it seems that we would get an auxiliary-input OWF. As discussed in Section 5, this approach doesn't work since the weakly black-box reduction we obtained from Theorem 4.1 is not length-increasing whereas Theorem 5.1 requires a length-increasing weakly black-box reduction.

However, the good news is that we can get around this issue by leveraging from the fact that most $\mathsf{NP}$-complete problems are paddable. For example, given a SAT formula $\phi$, it is easy to construct another SAT formula $\phi'$ that is much "longer" than $\phi$ while preserving its satisfiability. This enables us to obtain a length-increasing reduction.

Consider the reduction from SAT to the HSG $H$ defined as follows. Given a SAT formula $\phi$ of length $n$, we pad $\phi$ to get a sufficiently long formula $\phi'$ of length $\mathsf{poly}(n)$. And then we run

the parametric-honest reduction from SAT to $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ to get a $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ instance $x$ of length $O(n^2)$. Finally, we run the weakly black-box reduction in Theorem 4.1 from $\mathsf{Gap}_\infty\mathsf{MK}^\mathsf{t}\mathsf{P}$ to the HSG $H$ on the instance $x$. It follows that the reduction will query the oracle for avoiding $H$ on strings of length $\Omega(\sqrt{|x|}) \geq n$ and the reduction is length-increasing with respect to the SAT formula $\phi$. Thus, the reduction

$$\phi \to \phi' \to x \to H$$

is a length-increasing weakly black-box reduction from SAT to avoiding $H$. And we are now able to apply Theorem 5.1 to obtain an auxiliary-input OWF. This completes our proof.

## References


[ACR98] Alexander E Andreev, Andrea EF Clementi, and Jose DP Rolim. A new general derandomization method. *Journal of the ACM (JACM)*, 45(1):179–213, 1998.

[ACRT99] Alexander E Andreev, Andrea EF Clementi, José DP Rolim, and Luca Trevisan. Weak random sources, hitting sets, and bpp simulations. *SIAM Journal on Computing*, 28(6):2103–2116, 1999.

[AH91] William Aiello and Johan Håstad. Statistical zero-knowledge languages can be recognized in two rounds. *J. Comput. Syst. Sci.*, 42(3):327–345, 1991.

[AHT23] Eric Allender, Shuichi Hirahara, and Harsha Tirumala. Kolmogorov complexity characterizes statistical zero knowledge. In *Innovations in Theoretical Computer Science Conference*, 2023.

[Blu82] Manuel Blum. Coin flipping by telephone - A protocol for solving impossible problems. In *COMPCON'82, Digest of Papers, Twenty-Fourth IEEE Computer Society International Conference, San Francisco, California, USA, February 22-25, 1982*, pages 133–137. IEEE Computer Society, 1982.

[BM84] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudo-random bits. *SIAM Journal on Computing*, 13(4):850–864, 1984.

[DH76] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *STOC '90*, pages 416–426, 1990.

[GGM84] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. In *FOCS*, 1984.

[GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*, 28(2):270–299, 1984.

[GSV99] Oded Goldreich, Amit Sahai, and Salil Vadhan. Can statistical zero knowledge be made non-interactive? or on the relationship of szk and niszk. In *Advances in Cryptology—CRYPTO'99: 19th Annual International Cryptology Conference Santa Barbara, California, USA, August 15–19, 1999 Proceedings 19*, pages 467–484. Springer, 1999.

[GT00] Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*, pages 305–313. IEEE Computer Society, 2000.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[Hir18] Shuichi Hirahara. Non-black-box worst-case to average-case reductions within NP. In *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018*, pages 247–258, 2018.

[Hir23] Shuichi Hirahara. Capturing one-way functions via np-hardness of meta-complexity. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 1027–1038, 2023.

[IL89] Russell Impagliazzo and Michael Luby. One-way functions are essential for complexity based cryptography (extended abstract). In *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*, pages 230–235, 1989.

[IL90] Russell Impagliazzo and Levin LA. No better ways to generate hard np instances than picking uniformly at random. In *Proceedings [1990] 31st Annual Symposium on Foundations of Computer Science*, pages 812–821. IEEE, 1990.

[Imp95] Russell Impagliazzo. A personal view of average-case complexity. In *Structure in Complexity Theory '95*, pages 134–147, 1995.

[Ko86] Ker-I Ko. On the notion of infinite pseudorandom sequences. *Theor. Comput. Sci.*, 48(3):9–33, 1986.

[Kol68] A. N. Kolmogorov. Three approaches to the quantitative definition of information. *International Journal of Computer Mathematics*, 2(1-4):157–168, 1968.

[KVM02] Adam R KLIVANS and Dieter VAN MELKEBEEK. Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses. *SIAM journal on computing (Print)*, 31(5):1501–1526, 2002.

[Lev03] L. A. Levin. The tale of one-way functions. *Problems of Information Transmission*, 39(1):92–103, 2003.

[LP20] Yanyi Liu and Rafael Pass. On one-way functions and Kolmogorov complexity. In *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*, pages 1243–1254. IEEE, 2020.

[MV05] Peter Bro Miltersen and N. V. Vinodchandran. Derandomizing arthur-merlin games using hitting sets. *Computational Complexity*, 14(3):256–279, 2005.

[Nan21] Mikito Nanashima. On basing auxiliary-input cryptography on np-hardness via non-adaptive black-box reductions. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2021.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *J. Cryptology*, 4(2):151–158, 1991.

[Rom90] John Rompel. One-way functions are necessary and sufficient for secure signatures. In *STOC*, pages 387–394, 1990.

[Sip83] Michael Sipser. A complexity theoretic approach to randomness. In *Proceedings of the 15th Annual ACM Symposium on Theory of Computing, 25-27 April, 1983, Boston, Massachusetts, USA*, pages 330–335. ACM, 1983.

[SS22] Michael Saks and Rahul Santhanam. On randomized reductions to the random strings. In *37th Computational Complexity Conference (CCC 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

[Tra84] Boris A Trakhtenbrot. A survey of Russian approaches to perebor (brute-force searches) algorithms. *Annals of the History of Computing*, 6(4):384–400, 1984.