Instance Compression, Revisited

Gal Arnon
gal.arnon@weizmann.ac.il
Weizmann Institute

Shany Ben-David
shany.ben-david@biu.ac.il
Bar-Ilan University

Eylon Yogev
eylon.yogev@biu.ac.il
Bar-Ilan University

October 14, 2024

Abstract

Collision-resistant hashing (CRH) is a cornerstone of cryptographic protocols. However, despite decades of research, no construction of a CRH based solely on one-way functions has been found. Moreover, there are black-box limitations that separate these two primitives.

Harnik and Naor [HN10] overcame this black-box barrier by introducing the notion of *instance compression*. Instance compression reduces large NP instances to a size that depends on their witness size while preserving the "correctness" of the instance relative to the language. Shortly thereafter, Fortnow and Santhanam showed that efficient instance compression algorithms are unlikely to exist (as the polynomial hierarchy would collapse). Bronfman and Rothblum defined a computational analog of instance compression, which they called *computational instance compression* (CIC), and gave a construction of CIC under standard assumptions. Unfortunately, this notion is not strong enough to replace instance compression in Harnik and Naor's CRH construction.

In this work, we revisit the notion of computation instance compression and ask what the "correct" notion for CIC is, in the sense that it is sufficiently strong to achieve useful cryptographic primitives while remaining consistent with common assumptions. First, we give a natural strengthening of the CIC definition that serves as a direct substitute for the instance compression scheme in the Harnik–Naor construction. However, we show that even this notion is unlikely to exist.

We then identify a notion of CIC that gives new hope for constructing CRH from one-way functions via instance compression. We observe that this notion is achievable under standard assumptions and, by revisiting the Harnik–Naor proof, demonstrate that it is sufficiently strong to achieve CRH. In fact, we show that our CIC notion is existentially *equivalent* to CRH.

Beyond Minicrypt, Harnik and Naor showed that a strengthening of instance compression can be used to construct OT and public-key encryption. We rule out the computational analog of this stronger notion by showing that it contradicts the existence of *incompressible public-key encryption*, which was recently constructed under standard assumptions.

Keywords: instance compression; collision-resistant hash; one-way functions; interactive proofs

Contents

1 Introduction

Collision-resistant hashing (CRH) is one of the cornerstones of modern cryptography. It has a wide range of applications, including the classical "hash-and-sign" paradigm, zero knowledge, and succinct arguments [Kil92; Kil95]. A family $\mathcal{H}$ of collision-resistant hash functions has the property that efficient adversaries cannot find two distinct x_1, x_2, such that $h(x_1) = h(x_2)$ for a randomly sampled function $h \in \mathcal{H}$. There are constructions of CRH families based on diverse assumptions, including concrete structured assumptions such as the intractability of factoring, finding discrete logarithms in finite groups, and lattice-based assumptions [Dam87; GGH11; IKO05; YZWGL19], and more generic assumptions such as homomorphic encryption, and one-round PIR protocols [IKO05].

However, despite decades of research, there is no construction of a CRH from one-way functions. This missing connection implies that CRH cannot be placed within "Minicrypt", a class based solely on one-way functions according to Impagliazzo's five worlds of relative complexity [Imp95]. Additionally, any potential construction of CRH from one-way functions must leverage non-black-box techniques, even when using advanced tools like indistinguishability obfuscation and one-way permutations[Sim98; AS16]. Thus, constructing CRH from one-way functions remains one of the major open problems of our field.

Instance compression. To make progress on this question, Harnik and Naor [HN10] introduced the notion of *instance compression*, with the goal to compresses large NP instances to a size that depends on their witness length, which may be significantly smaller, while preserving the "correctness" of the instance relative to the language. Roughly, an instance compression for a relation R consists of a pair of efficient algorithms $(\mathsf{IC}, \mathsf{WT})$, one for compressing the instance into a *smaller* instance $x' = \mathsf{IC}(x)$ and one for translating the witness $w' = \mathsf{WT}(x, w)$. The condition to be satisfied is that $(x, w) \in R$ if and only if $(\mathsf{IC}(x), \mathsf{WT}(x, w)) \in R$ (not all definitions require WT to be efficient). While the compression rate may vary, the standard regime involves an instance x' of size $\mathrm{poly}(|w|, \log |x|)$.

Harnik and Naor [HN10] showed that by using instance compression, it is possible to construct a CRH family from one-way functions in a non-black-box manner. This raised hope that constructing CRH from one-way functions might be feasible if one could first solve the (arguably) simpler problem of instance compression. Instance compression has proven useful in other works, such as in [BDFH09], where it was employed as a pre-processing technique for fixed-parameter tractable (FPT) problems.

Unfortunately, this hope was short-lived as Fortnow and Santhanam [FS11] showed that efficient instance compression algorithms are unlikely to exist. Specifically, they showed that an instance compression for SAT (which is required for the CRH construction) would imply that NP $\subseteq$ coNP/poly, ultimately leading to a collapse of the polynomial hierarchy. Their results additionally extend to relaxed notions of instance compression that allow for a small error probability and also to *quantum instance compression* [FS11; Dru15]. As a result, the initial optimism surrounding new non-black-box constructions of CRH waned.

Computational instance compression. This unfulfilling state of affairs remained until Bronfman and Rothblum [BR22] defined a computational analog of instance compression, which they called *computational instance compression* (CIC). This relaxed notion allows for errors that are computationally hard to find (in the CRS model). Specifically, they allow for instances that are not in the language to be compressed to an instance $x' = \mathsf{IC}(x)$ that is in the language, as long as

proving the validity of x' is computationally hard. In other words, it is computationally hard to find a witness w' for x'. This notion bypasses the barriers posed by Fortnow and Santhanam, and indeed, Bronfman and Rothblum constructed CICs for bounded-depth NP relations from standard assumptions (LWE). Ben-David [Ben24] extended this work to cover all NP relations while relying on a broader range of assumptions (e.g., DDH).

Despite this progress, their CIC definition could not serve as a replacement instance compression primitive in Harnik and Naor's CRH construction [HN10], leaving the problem unresolved. This leads us to ask the question:

Is there an achievable notion of computational instance compression that is sufficient for constructing a CRH family?

1.1 Our contributions

We provide an in-depth and thorough (positive) answer to the aforementioned question.

Impossiblity of CIC with strong soundness. We begin by providing a CIC definition with a strong soundness notion that serves as a direct substitute, for instance, compression in the Harnik–Naor construction. The completeness property remains the same (instances in the language are always mapped to instances in the language). The difference is in the soundness requirement. As opposed to the notions given in [BR22; Ben24],[1] here we do not require the adversary to find a witness for the compressed instance x' (additionally, we do not require an efficient witness transformation algorithm).

In more detail, for a setup algorithm Gen and a compression algorithm IC we say that the scheme has strong soundness s if for every security parameter $\lambda \in \mathbb{N}$, and any efficient adversary $\mathbf{A}$:

- *Strong soundness (informal):*

$$\Pr\left[\begin{array}{l} x \notin L(R) \\ \wedge\ \mathsf{IC}(\mathsf{crs}, x) \in L(R') \end{array} \middle| \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda) \\ x \leftarrow \mathbf{A}(\mathsf{crs}) \end{array}\right] \leq \mathsf{s}(\lambda)\,.$$

A CIC with strong soundness $\mathsf{s}(\lambda) = \mathsf{negl}(\lambda)$ suffices to recover the CRH constructions of [HN10]. Unfortunately, we show that this definition is unlikely to exist. Inspired by [FS11] and [Dru15], we prove that a *computational* instance compression for SAT with strong soundness would imply that the polynomial hierarchy collapses (the limitations of [FS11; Dru15] apply only for information-theoretic definitions). In fact, this limitation holds even for very large strong soundness error $\mathsf{s}(\lambda) = 1 - 2^{-\lambda}$.

Theorem 1.1 (Informal)**.** *If there exists a computational instance compression for* SAT *with strong soundness error* $\mathsf{s}(\lambda) \leq 1 - 2^{-\lambda}$ *(as defined above) then* $\mathsf{UNSAT} \in \mathrm{NP}/poly$.

Therefore, we need to identify a weaker notion that is both achievable and still adequate for constructing collision-resistant hash functions.

Adaptive CIC to the rescue. We give new hope for constructing CRH from one-way functions via instance compression. We identify that an adaptive version of the CIC given in [BR22] is precisely the notion we need. Informally, the adaptive CIC soundness notion we consider is the following:

[1] A discussion in [BR22] suggests a variant that does not require finding a witness (which they call "strong CIC"). Their notion is incomparable to ours and cannot serve as a substitute for instance compression in the Harnik–Naor construction.

- *Adaptive soundness (informal).* For every $\lambda \in \mathbb{N}$, and every efficient adversary $\mathbf{A}$:

$$\Pr\left[\begin{array}{l|r} x \notin L(R) & \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda) \\ \wedge\ (x', w') \in R' & (x, w') \leftarrow \mathbf{A}(\mathsf{crs}) \\ & x' \leftarrow \mathsf{IC}(\mathsf{crs}, x) \end{array}\right] = \mathsf{negl}(\lambda)\,.$$

On the one hand, we observe that this definition is achievable, and in fact, it was implicitly realized in [Ben24] under standard assumptions. On the other hand, we strengthen the proof of [HN10] to rely on this adaptive CIC to construct a CRH family. As our construction of CRH is non-black-box it provides a path bypass the CRH black-box separation, and raises the natural task of a (black-box) construction of adaptive CIC from one-way functions.

In fact, we establish a stronger result: we provide a construction of adaptive CIC from CRH. This means that any construction of CRH from one-way functions must go via adaptive CIC.

Theorem 1.2 (Informal)**.** *Assume that one-way functions exist. Then, there is a non-black-box construction of a family of collision-resistant hash functions from any adaptive computational instance compression scheme for* SAT*.*

Conversely, any family of collision-resistant hash functions implies the existence of an adaptive computational instance compression scheme for SAT*.*

We note that the construction of CIC from CRH yields a CIC with a large witness for the compressed instance. That is, the size of the witness w' for the compressed instance x' depends on the original large instance x. For some advanced applications, a CIC with a smaller witness (independent of the size of x) is desired. The precise theorem statement is given in Section 6.

Beyond collision-resistant hashing? Harnik and Naor [HN10] showed that instance compression can be used to construct primitives beyond Minicrypt. In particular, they show that if the instance compression is equipped with an efficient *witness retrieval* algorithm, it can be used to construct OT and public-key encryption from one-way functions, thus showing that constructing instance compression has the potential of collapsing Minicrypt and Cryptomania. A witness retrieval algorithm gets as input a witness w for x, and $x' = \mathsf{IC}(x)$, and outputs a witness w' for x'.

We observe that an adaptive CIC with witness retrieval can serve as a replacement for the instance compression in the [HN10] for public-key encryption. This naturally leads to the question of whether CIC with witness retrieval can exist.

We answer this question negatively by showing a connection between instance compression and incompressible encryption [Dzi06; GWZ22]. Roughly speaking, an incompressible encryption scheme produces large, incompressible ciphertexts in the sense that any adversary who "forgets" a small fraction of the ciphertext data learns nothing about the encrypted data, even given the secret key. Incompressible encryption (with various rates) has been constructed under various assumptions.

We show that the existence of adaptive CIC with an efficient witness retrieval algorithm (along with a NIZK scheme) implies that incompressible public-key encryption does not exist, which contradicts existing constructions of incompressible PKE schemes [GWZ22]. See Figure 1 for a summary of our results.

Theorem 1.3 (Informal)**.** *If there exists an incompressible PKE with an efficient key generation algorithm and NIZK for* NP*, then there is no computational instance compression (adaptive or non-adaptive) with an efficient witness retrieval algorithm.*

For our results, we require the key generation algorithm of the incompressible PKE scheme to run in time that is polynomial in the security parameter and polylogarithmic in the compression parameter. In [GWZ22], several suitable constructions meeting these requirements have been proposed, either based on CPA-secure public key encryption or indistinguishability obfuscation.

Corollary 1.4. *If there exists a CPA secure public-key encryption scheme and NIZK for* NP*, then there is no computational instance compression (adaptive or non-adaptive) with an efficient witness retrieval algorithm.*

CIC and pre-processing SNARGs. Finally, we show (somewhat surprising) connections between CIC and other cryptographic primitives. We observe that any (adaptive) pre-processing SNARG[2] implies an adaptive CIC, thus getting the first formal connection from SNARGs to CRH. In [WW24b], Waters and Wu constructed an adaptive SNARG from indistinguishability obfuscation and any rerandomizable one-way function. Later, Waters and Zhandry [WZ24] constructed an adaptive SNARG based on indistinguishability obfuscation and lossy functions. We observe that both constructions also imply an adaptive pre-processing SNARG and, in turn, a CRH. This shows that the reradomizable one-way functions or the lossy functions are the key elements that allow to escape the impossibility results of [AS16].

Waters and Wu [WW24a] later improved this result to rely solely on indistinguishability obfuscation (and one-way functions). However, in this case, their construction *does not* imply a pre-processing SNARG, and there is a possible explanation for that. Otherwise, we would get a construction of a CRH from indistinguishability obfuscation. While Theorem 1.2 is non-black-box in the sense of [Sim98], the overall construction could still be described using oracle-aided circuits and thus capture under the impossibility of [AS16]. See Figure 1 for a summary of our results.

1.2 Related work

On the non-existence of instance compression. Dubrov and Ishai [DI06] studied the (non-existence) of instance compression through the notion of non-boolean PRGs. They asked if every efficiently samplable distribution can be efficiently sampled, up to a small statistical distance, using roughly as much randomness as the length of its output. They showed that the non-existence of (strong forms of) instance compression yields such a non-boolean PRG (they give constructions under other assumptions as well). They further showed that the non-existence of a non-boolean PRG implies a construction of a *distributional* collision-resistant hash function (dCRH) from any one-way permutation. While dCRH is a weaker notion than standard CRH, the separations of [Sim98] and [AS16] apply also to dCRH.

More non-black-box constructions of CRH. In [KY18], the authors give a non-black-box construction of dCRH from a multi-collision resistance hash (MCRH) family. MCRH was introduced in [KNY17], where it is hard to find multiple elements (more than 2) that all hash to the same value. The different flavors of hashing were formulated in [KNY18] into four worlds of hashing (analogously to the five complexity worlds of [Imp95]). Recently, the techniques from [KY18] were extended to get a non-black-box construction of a CRH from MCRH for certain parameters [RV24; BT24].

[2] In a pre-processing SNARG, the verifier runs in sublinear time given the proof and the output of a more costly pre-processing phase on the input.

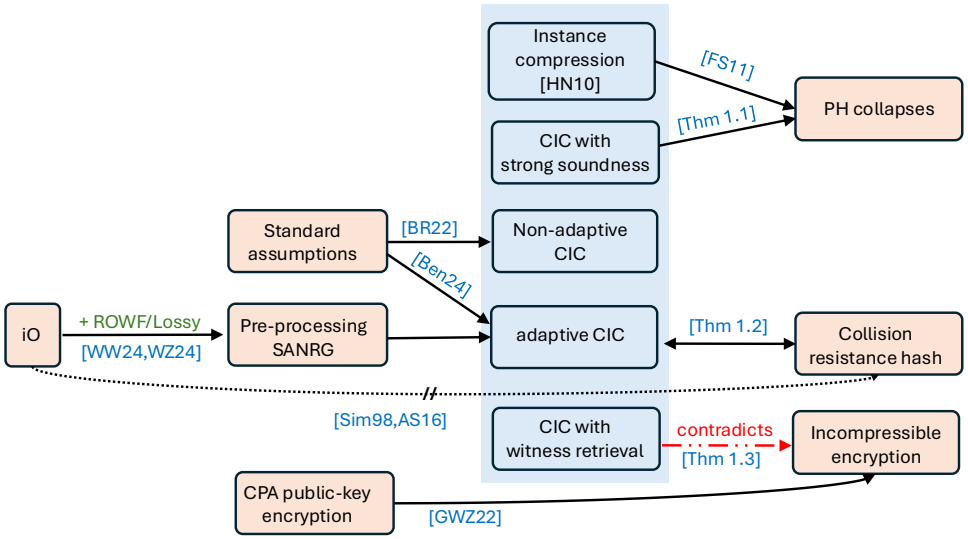

Figure 1: An illustration of the known results and our new implications. Solid lines mean positive implications, and crossed-out dashed lines mean black-box separations. ROFW is short for rerandomizable one-way functions; lossy refers to lossy functions.

2 Techniques

In this section, we give an overview of our techniques:

- In Section 2.1, we show that the existence of CIC with strong soundness implies a collapse of the polynomial hierarchy.
- In Section 2.2, we show that adaptively secure CIC exists if and only if there exists a family of collision-resistant hash functions.
- In Section 2.3, we show that CIC with efficient witness retrieval is at odds with incompressible encryption.
- In Section 2.4, we discuss connections between CIC and pre-processing SNARGs.

2.1 Impossibility of CIC with strong soundness

We give an overview of our impossibility result described in Theorem 1.1. Intuitively, a CIC for source relation R and target relation R' with strong soundness is an efficient mapping from large elements to smaller ones, specified by the choice of the reference string crs and computed using $\mathsf{IC}(\mathsf{crs}, \cdot)$. For completeness, it holds that for any crs generated in the setup phase, every instance $x \in L(R)$ is mapped to an instance $x' \in L(R')$. While this definition permits incorrect mappings where $x \notin L(R)$ is mapped to $x' \in L(R')$, strong soundness ensures that, with probability s, a bounded adversary cannot find such an instance x (the probability is over the choice of crs and the adversary's randomness).

We assume the existence of a CIC with strong soundness from the NP-complete source relation R_{ORSAT} (defined below) to an arbitrary NP target relation R', which compresses instances to a size of $k(\lambda, n, w) = \mathrm{poly}(\lambda, \log n, m)$, where λ is the security parameter, n is the instance size, and m is the witness size. Given this, we show how to construct an NP/*poly* verifier that decides UNSAT.

The relation R_{ORSAT} is defined as follows,

$$R_{\mathsf{ORSAT}} := \{(\phi, w) = ((\varphi_1, \ldots, \varphi_t), w) \mid \exists i \in [t],\ w \text{ is a satisfying assignment for } \varphi_i\} \ .$$

We will decide instances of UNSAT of size n, using the CIC run with security parameter $\lambda = n$, and run on R_{ORSAT} instances that are comprised of t subformulas, where each subformula is of size n, where $t = \mathrm{poly}(n)$ is some sufficiently large polynomial (the exact choice of polynomial depends on the parameters of the CIC). Let $k = \mathrm{poly}(n)$ be the output size of the CIC for such formulas. We define the set T to be the set of all outputs of the CIC that are not in $L(R')$.

Structure of the advice. The NP/*poly* verifier that decides UNSAT will receive an advice string U of comprised of at most n pairs (crs, y) where crs is a common reference string for the CIC and $y \in T$. Observe that, since $|\mathsf{crs}| = \mathrm{poly}(\lambda) = \mathrm{poly}(n)$ and $|y| = k = \mathrm{poly}(n)$, the advice U can be described as a binary string of length $\mathrm{poly}(n)$. We defer discussion of the precise properties of the advice string to when we argue completeness.

The NP/*poly* verifier. Given the advice U, for an instance φ and a witness $\phi := (\varphi_1, \ldots, \varphi_t)$:
1. Check that φ is a subformula in ϕ: $\varphi \in \{\varphi_1, \ldots, \varphi_t\}$.
2. Check that there exists a pair $(\mathsf{crs}, y) \in U$ such that $\mathsf{IC}(\mathsf{crs}, \phi) = y$.
3. If both of the above checks pass, accept. Otherwise, reject.

We discuss soundness, completeness, and efficiency of the algorithm.

Soundness. We show that, by completeness of the CIC, given the advice string U, an instance $x \in \mathsf{SAT}$, and a claimed witness w, the verifier will reject (in fact, this holds for any advice string that is structured as pairs (crs, y) as described above).

Fix any $\tilde{\varphi} \in \mathsf{SAT}$, and let $\phi := (\varphi_1, \ldots, \varphi_t)$ be a potential witness. If the witness ϕ does not include φ as a subformula, then the verifier rejects during its first check. On the other hand, if it does include φ as subformula, then $\phi \in \mathsf{ORSAT}$. By the perfect completeness of the CIC, this implies that for every crs, $\mathsf{IC}(\mathsf{crs}, \phi) \in L(R')$. Since for every $(\mathsf{crs}, y) \in U$, we have $y \in T \subseteq \overline{L(R')}$, this implies that $\mathsf{IC}(\mathsf{crs}, \phi) \neq y$ for every $(\mathsf{crs}, y) \in U$, and so the verifier rejects in its second check. See Figure 2 for an illustration.

Completeness. We show that given the advice string U and an instance $x \in \mathsf{UNSAT}$, there exists a witness ϕ that will make the verifier accept.

We begin by describing how the advice is chosen (by an inefficient algorithm). We define the set S to be the set of all UNSAT formulas of size n.

1. Initialize $U := \emptyset$, $S_1 := S$, and $i := 1$.
2. While $S_i \neq \emptyset$ do the following:
   (a) Set X_i to be the set of all formulas ϕ of the form $(\varphi_1, \ldots, \varphi_t)$, where $\varphi_j \in S_i$.
   (b) Set crs_i to be a CRS such that $\Pr[\mathsf{IC}(\mathsf{crs}_i, \phi) \in T \mid \phi \leftarrow X_i] > 1 - \mathsf{s}$.
   (We prove in Claim 1 that there exists such a CRS.)
   (c) Let y_i be the element in T with the maximal number of formulas $\phi \in X_i$ with $\mathsf{IC}(\mathsf{crs}_i, \phi) = y_i$.
   (d) Add the pair (crs_i, y_i) to U.
   (e) Let S_{i+1} be the set of all elements in S_i that are not part of a formula $\phi \in X_i$ with $\mathsf{IC}(\mathsf{crs}_i, \phi) = y_i$.
   (f) Update $i := i + 1$.
3. Output U.

The connection between the sets S_i, X_i, and T is described pictorially in Figure 2.

Note that it is not immediately clear that the algorithm is well defined (i.e., Item 2b always succeeds) and that it will ever terminate. The CIC could have many "errors" in the form of formulas $\phi \in \mathsf{UNSAT}$ that are mapped to elements in $y \in \bar{T}$ ($\bar{T}$ is the complement of T). Moreover, it could be the case that for any crs, there is a subformula φ such that $\mathsf{IC}(\mathsf{crs}, \phi) \in \bar{T}$ for every ϕ that contains φ. This would mean that any single crs by itself is not sufficient for the completeness of all $\varphi \in \mathsf{UNSAT}$, and it is the reason why we need to couple a different CRS string crs_i for each element y_i.

Nevertheless, we show that the algorithm is well-defined and that it always terminates after n steps. Observe that the algorithm terminates only if for every $\varphi \in S$ there exists $(\mathsf{crs}_i, y_i) \in U$ and ϕ such that φ is a subformula of ϕ, and $\mathsf{IC}(\mathsf{crs}, \phi) = y$. This is precisely what the verifier checks, and so it will accept given advice U and witness ϕ.

Claim 1 (Informal)**.** *For every i, there exists crs_i such that*

$$\Pr[\mathsf{IC}(\mathsf{crs}_i, \phi) \in T \mid \phi \leftarrow X_i] > 1 - \mathsf{s}.$$

Proof sketch. We base the existence of such a CRS on the strong soundness guarantee of the CIC scheme. Suppose, for contradiction, that such a CRS does not exist. This would imply that for any crs, a uniformly random element $\phi \leftarrow X_i$ would be mapped to an element in the target relation, i.e., $\mathsf{IC}(\mathsf{crs}, \phi) \in L(R')$, with probability greater than s. Using an averaging argument, we can show

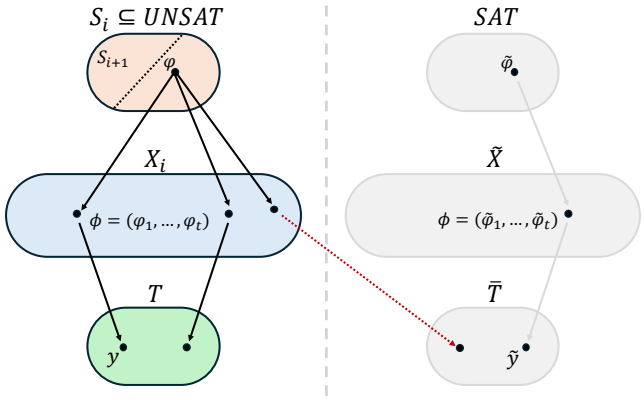

Figure 2: An illustration of the various sets and the relationship between them. The top left oval represents the set S_i of unsatisfiable formulas still not covered by the advice. The right-hand side of this set is what is covered in the current iteration, while the left-hand side, S_{i+1}, is a subset that is left to be covered by future iterations. Every $\varphi \in S_i$ has an outgoing edge to all $\phi \in X_i \subseteq \overline{\mathsf{ORSAT}}$ that contain φ. The formulas ϕ have either outgoing edges to T (in case the mapping is correct) or outgoing edges to $\bar{T}$ (these are the "errors"), which is denoted with a red dashed arrow.

that there exists a specific element $\phi \in X_i \subseteq \overline{\mathsf{ORSAT}}$ such that: $\Pr_{\mathsf{crs}}[\mathsf{IC}(\mathsf{crs}, \phi) \in L(R')] > \mathsf{s}$. Since $\phi \in X_i$ is an element that is not in the source language (not in ORSAT), an adversary that only outputs ϕ would break the strong soundness with probability greater than s. (Refer to Claim 5.5 for the formal proof). □

Next, we demonstrate that the algorithm terminates and that the advice size is small.

Claim 2 (Informal)**.** *The algorithm terminates, and outputs U comprised of n pairs (crs, y).*

Proof sketch. We show that $|S_{i+1}| \leq \frac{1}{2} \cdot |S_i|$ for every i. This proves the claim since we start with the set $S_0 = \mathsf{UNSAT}$ of size at most 2^n so it takes n steps to reduce the set all the way down (so the algorithm terminates), and in every step, we append a pair (crs, y) to U (so U has the right size).

In the i-th step, X_i consists of all formulas $\phi = (\varphi_1, \ldots, \varphi_t)$ composed of subformulas from S_i. For simplicity, we assume that the soundness error s is at most $1/2$ (in the full proof, this requirement is removed). In this case, the reference string crs_i selected by the algorithm ensures that at least half of the elements in X_i are correctly mapped (i.e., $\mathsf{IC}(\mathsf{crs}_i, \phi) \in T$). Since X_i is comprised of all t-tuples of elements in S_i, the size of X_i is $|S_i|^t$. Let y_i be the element in T with the maximal number of formulas $\phi \in X_i$ with $\mathsf{IC}(\mathsf{crs}_i, \phi) = y_i$, and let Y_i be the set of all formulas $\phi \in X_i$ that are mapped to y_i. Since there exists $(1 - \mathsf{s}) \cdot |X_i|$ elements that are mapped correctly (according to the CRS choice in Item 2b), by an averaging argument, we have $|Y_i| \geq \frac{(1-\mathsf{s}) \cdot |X_i|}{|T|}$. Since the soundness error is bounded by $1/2$, it holds that $|Y_i| \geq \frac{|X_i|}{2 \cdot |T|} = \frac{|S_i|^t}{2^{k+1}}$.

On the other hand, S_{i+1} is formed by removing the elements in S_i that are included in a tuple in Y_i. Therefore, $|S_i \setminus S_{i+1}|^t \geq |Y_i|$. By combining the two bounds on the size of Y_i, we show in the full proof that $|S_i \setminus S_{i+1}| \geq |S_i|/2$. Thus, we get that $|S_{i+1}| \leq |S_i|/2$, as required. □

2.2 Collision resistant hashing and adaptive CIC

We show that collision-resistant hash functions can be constructed from adaptive CIC in a non-black-box manner and that CRH can be used to construct adaptive CIC.

2.2.1 Collision resistant hashing from CIC

We show how to construct a CRH function family from adaptive CIC (and one-way functions). Our construction closely follows the construction presented in [HN10], using CIC in place of instance compression. In addition to an adaptive CIC scheme, our construction relies on the existence of a commitment scheme $(\mathsf{CM.Com}, \mathsf{CM.Ver})$ that is statistically binding and computationally hiding (which can be constructed from one-way functions [Nao91]).

The high-level idea of the construction is to generate a commitment σ to a random value $i \in [N]$. Then, given the commitment, for each input $x \in \{0,1\}^N$ we define a formula that is satisfiable if and only if the i-th bit of x is equal to 1. This formula is then compressed using the CIC scheme, and its output is the output of the hash function.

Roughly, if a collision x, x' is found, then either both have bit i set to the same value (which invalidates the fact that the commitment scheme hides i), or one has its i-th bit set to 0, and the other has it set to 1. In this case, one of the formulas derived from x and x' is satisfiable, and the second is not, but they both are compressed into the same value. Thus, one of the formulas is mapped to a value that does not match whether it is satisfiable, which invalidates the security of the CIC.

We describe the hash function family:

- *Key generation*: For input size N and security parameter λ, sample $i \leftarrow [N]$ and generate commitment $\sigma \leftarrow \mathsf{CM.Com}(i)$. Sample crs for the CIC scheme with security parameter λ. The hash key is then $\mathsf{hk} = (\sigma, \mathsf{crs})$.
- *Hash value*: On input $x \in \{0,1\}^N$,
  1. Let ϕ_σ be the SAT analogue of $\mathsf{CM.Ver}$ when applied to commitment σ. Let $y_1, \ldots, y_\ell$ be the variables of ϕ_σ that represent the value under the commitment (i.e., supposed to be i).
  2. Let ϕ_x be a formula over the variables $y_1, ..., y_\ell$, where for every $j \in [N]$ it holds that $\phi_x(j) = 1$ if and only if $x_j = 1$.
  3. Set $\phi_{\sigma,x} := (\phi_\sigma \wedge \phi_x)$, and output $\phi' := \mathsf{IC}(\mathsf{crs}, \phi_{\sigma,x})$.

We prove that the hash function described above is compressing and that it is collision-resistant.

Compression. The formula ϕ_σ has size $\mathrm{poly}(\lambda, \log N)$, and the number of variables in ϕ_x is $\ell = \log N$. Thus, the number of variables in the formula $\phi_{\sigma,x}$ (which is its witness size) is $\mathrm{poly}(\lambda, \log N)$, and its size is $|\phi_{\sigma,x}| = \mathrm{poly}(N)$. The CIC scheme compresses the formula into one of size $\mathrm{poly}(\lambda, \log |\phi_{\sigma,x}|, m)$ where m is the witness size. Thus, the output of the hash function has size $\mathrm{poly}(\lambda, \log N)$.

Collision resistance. We show that the scheme is collision-resistant. Let $\mathbf{A}$ be a computationally efficient adversary for the CRH, and denote by x, x' its outputs. Supposing x an x' are a collision, either the formulas $\phi_{\sigma,x}, \phi_{\sigma,x'}$ defined by x, x' have the same satisfiability (i.e., are both satisfiable or both unsatisfiable) or one is satisfiable while the other is not. We show that both cases can

happen with at most negligible probability, implying that x and x' cannot be a collision except with negligible probability.

Before analyzing the two cases, we highlight two observations regarding the formulas in the construction:

1. *$\phi_{\sigma,x}$ is satisfiable if and only if $x_i = 1$.* By the binding property of the commitment scheme (except with negligible probability), the only assignment to the variables $y_1, \ldots, y_\ell$ that satisfies ϕ_σ is i, and by correctness of the commitment scheme, this assignment will satisfy the formula (along with some assignment on the rest of the variables of the formula). Furthermore, $\phi_x(i) = 1$ if and only if $x_i = 1$. Therefore (except with negligible probability over generation of the commitment σ), $\phi_{\sigma,x}$ is satisfiable if and only if $x_i = 1$.

2. *If $\phi_{\sigma,x}$ is satisfiable, then every satisfying assignment for ϕ_σ is satisfying for $\phi_{\sigma,x}$.* Let w be a satisfying assignment to ϕ_σ. By the binding of the commitment scheme (except with negligible probability), every satisfying assignment to ϕ_σ must include in the variables $y_1, \ldots, y_\ell$ the committed value i. By Item 1, the satisfiability of $\phi_{\sigma,x}$ implies that $x_i = 1$. Since $\phi_x(i) = 1$ if and only if $x_i = 1$, we get that $\phi_x(w) = 1$. Therefore, $\phi_{\sigma,x}(w) = \phi_\sigma(w) \wedge \phi_x(w) = 1$.

We now turn to showing that each of the two cases alluded to earlier can occur with at most a negligible probability:

- *$\phi_{\sigma,x}$ and $\phi_{\sigma,x'}$ have the same satisfiability.* We base our argument on the computational hiding of the commitment scheme. Since $\phi_{\sigma,x}$ is satisfiable if and only if $x_i = 1$ (by Item 1), and since $\phi_{\sigma,x}$ and $\phi_{\sigma,x'}$ have the same satisfiability, it must hold that $x_i = x'_i$. Now, since x and x' form a collision, there must be at least one index $j \in [N]$ where x and x' differ – so this index is *not* i. If this occurs with non-negligible probability, then we can construct an adversary for the commitment scheme that, upon receiving a commitment, constructs the hash function using it, and runs $\mathbf{A}$ to get x and x' and guesses the index i out of (at most) $N - 1$ indices that x and x' agree on identical will have a noticeable advantage in guessing i. Due to the computational hiding property of the commitment scheme, this can only occur with negligible probability.

- *$\phi_{\sigma,x}$ and $\phi_{\sigma,x'}$ have different satisfiability.* In this case, we demonstrate an attack on the CIC scheme. Without loss of generality, assume that $\phi_{\sigma,x} \in \mathsf{SAT}$ and $\phi_{\sigma,x'} \notin \mathsf{SAT}$. Consider the following adversary to adaptive soundness of the CIC scheme.

  1. Given as input a crs, sample $i \leftarrow [N]$, and generate a commitment σ.
  2. Run $\mathbf{A}$ with $\mathsf{hk} = (\mathsf{crs}, \sigma)$ to obtain x, x'.
  3. Compute a satisfying assignment w for ϕ_σ. This can be done efficiently as the adversary can generate the opening to the commitment σ.
  4. Output $(\phi_{\sigma,x'}, \mathsf{WT}(\phi_{\sigma,x}, w))$.

  First, note that since w is a satisfying assignment for ϕ_σ, and since $\phi_{\sigma,x}$ is satisfiable, by Item 2, w is also a satisfying assignment (and a witness) for $\phi_{\sigma,x}$. Therefore, by completeness of the CIC scheme: $(\mathsf{IC}(\mathsf{crs}, \phi_{\sigma,x}), \mathsf{WT}(\mathsf{crs}, \phi_{\sigma,x}, w)) \in R'$. Since x, x' form a collision, it holds that $\mathsf{IC}(\mathsf{crs}, \phi_{\sigma,x}) = \mathsf{IC}(\mathsf{crs}, \phi_{\sigma,x'})$, and therefore $(\mathsf{IC}(\mathsf{crs}, \phi_{\sigma,x'}), \mathsf{WT}(\mathsf{crs}, \phi_{\sigma,x}, w)) \in R'$. Since $\phi_{\sigma,x'} \notin \mathsf{SAT}$, by adaptive security of the CIC scheme, this can happen with at most negligible probability, which implies that $\mathbf{A}$ could not have output such x and x' with non-negligible probability.

2.2.2 CIC from collision-resistant hashing

Given a collision-resistant hash (CRH) scheme, we construct an adaptively sound CIC scheme for any NP relation R as follows:

- *Setup:* Sample hash key hk, and output $\mathsf{crs} := \mathsf{hk}$.
- *Compression (IC):* Given as input $\mathsf{crs} = \mathsf{hk}$, and instance x. Output the compressed instance $x' := (\mathsf{hk}, \mathsf{Hash}(\mathsf{hk}, x))$.
- *Witness transformation (WT):* Given as input $\mathsf{crs} := \mathsf{hk}$, an instance x, and a witness w. Output the new witness $w' := (x, w)$.

The compressed instance consists of both the hash key and the hashed value, while the new witness consists of the original instance and witness. The target relation R' is defined as:

$$R' := \{(x', w') = ((\mathsf{hk}, z), (x, w)) \mid (x, w) \in R \wedge \mathsf{Hash}(\mathsf{hk}, x) = z\}\,.$$

It is immediate by construction that completeness holds, i.e., if $(x, w) \in R$, then it holds that $(\mathsf{CIC}(\mathsf{crs}, x), \mathsf{WT}(\mathsf{crs}, x, w)) \in R'$. We turn to showing adaptive soundness.

We argue that any adversary that outputs (x, w') that breaks CIC soundness with noticeable probability can be used to generate a hash collision with noticeable probability. Consider such a CIC adversary. Given crs, its output (x, w') breaks soundness of the CIC if $x \notin L(R)$ with $(\mathsf{IC}(\mathsf{crs}, x), w') \in R'$. Parsing w' as a pair $(\tilde{x}, \tilde{w})$, and letting $(\mathsf{hk}, z) = \mathsf{IC}(\mathsf{crs}, x)$, by the definition of R', this implies that $(\tilde{x}, \tilde{w}) \in R$, and $\mathsf{Hash}(\mathsf{hk}, \tilde{x}) = z = \mathsf{Hash}(\mathsf{hk}, x)$. Observe that $x \notin L(R)$ and $\tilde{x} \in L(R)$, and so $x \neq \tilde{x}$. Thus, $x \neq \tilde{x}$ and $\mathsf{Hash}(\mathsf{hk}, \tilde{x}) = \mathsf{Hash}(\mathsf{hk}, x)$, forming a hash collision.

2.3 Impossibility of CIC with efficient witness retrieval

We show a surprising connection between CIC and incompressible encryption. Specifically, we show that the existence of incompressible encryption implies the inexistence of CIC with efficient witness retrieval.

Witness retrieval. Harnik and Naor posited a notion of instance compression where, given the compressed instance $x' \in R$ and a witness w for the original instance, one can efficiently derive a witness w' for x', and show that it implies public-key encryption through oblivious transfer. We observe that CIC with such an efficient witness retrieval algorithm WR also suffices for these constructions.

Incompressible encryption. Roughly, an incompressible PKE scheme is a PKE scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ where we require that any computationally bounded two-stage algorithm $(\mathbf{A}_1, \mathbf{A}_2)$ wins in the following game with at most negligible probability:

1. The challenger samples $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}$, $b \leftarrow \{0, 1\}$, and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, b)$.
2. $\mathbf{A}_1$ receives pk and ct and outputs a state st with $|\mathsf{st}| \ll |\mathsf{ct}|$.
3. $\mathbf{A}_2$ receives pk, sk, and st and outputs b'.
4. The adversary wins in the game if $b = b'$.

Witness retrieval versus incompressible encryption. We show that CIC with efficient witness retrieval is incompatible with incompressible PKE. Intuitively, they are at odds in the following way: incompressible encryption says that it is hard to compress the ciphertext ct while preserving information about its decryption. CIC for the relation of ciphertexts that decrypt to 0 negates

this by positing that ct can be compressed while retaining information about the decryption, and witness retrieval allows for this information to leak via the original secret key.

In more detail, suppose that there exists a CIC scheme $(\mathsf{CIC}, \mathsf{WT})$ with efficient witness retrieval algorithm WR for the relation

$$R := \{((\mathsf{pk}, \mathsf{ct}), \mathsf{sk}) \mid \mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = 0\} \ ,$$

and some output relation R'. We propose an adversary $(\mathbf{A}_1, \mathbf{A}_2)$ for the PKE scheme: (1) $\mathbf{A}_1(\mathsf{pk}, \mathsf{ct})$ outputs $\mathsf{st} := \mathsf{CIC}(\mathsf{pk}, \mathsf{ct})$, and (2) $\mathbf{A}_2(\mathsf{pk}, \mathsf{sk}, \mathsf{st})$ outputs $b' = 0$ if $(\mathsf{st}, \mathsf{WR}(\mathsf{st}, \mathsf{sk})) \in R'$ and otherwise outputs 1.

Intuitively, this adversary wins in the incompressibility security game since st generated by the CIC compresses the input while preserving information on whether it belongs to R, which in turn informs about the encrypted bit. Later, this bit of information is leaked by using the witness retrieval algorithm: since sk is a witness to R, the algorithm WR outputs a correct witness for st belonging to R'.

Incompressible PKE with key consistency. Unfortunately, this intuition does not quite hold, since for ciphertexts $\mathsf{ct}' \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$, it may be that there exist secret keys sk' such that $\mathsf{Dec}(\mathsf{sk}', \mathsf{ct}') = 0$. By perfect correctness of the PKE scheme, this can only be the case if $(\mathsf{pk}, \mathsf{sk}')$ is not in the image of Gen. In order to resolve this issue, we transform the PKE scheme into one that has *key consistency*, informally meaning that there are (with high probability) no alternate secret keys that can decrypt the ciphertext to the wrong bit (but decrypting with incompatible keys may output $\perp$). This suffices for our adversary for the incompressible encryption.

We show how to transform an incompressible PKE scheme into one with key consistency using NIZK proofs. Roughly, when generating key pairs, we add a NIZK proof that the key pair was generated by Gen. This NIZK proof is then appended to the secret key. When decrypting with a secret key, it is also verified that $(\mathsf{pk}, \mathsf{sk})$ is in the image of Gen using the NIZK proof (and otherwise outputs $\perp$). This ensures that the secret-key is valid, and so if there is decryption to a value other than $\perp$ then this must be the correct value. Zero-knowledge ensures that the proof does not leak any information about the randomness used to generate the key-pair (knowing it could potentially break incompresibility of the original scheme).

2.4 Pre-processing SNARGs and CIC

We observe that pre-processing SNARGs are intimately connected to CIC.

Pre-processing SNARGs. An adaptively sound SNARG for a relation R is a tuple of algorithms $(\mathsf{Gen}, \mathbf{P}, \mathbf{V})$ with the following properties. Roughly, completeness says that for every $\mathsf{crs} \leftarrow \mathsf{Gen}$, if $(x, w) \in R$ then $\mathbf{V}(\mathsf{crs}, x, \pi) = 1$ with probability 1 for $\pi \leftarrow \mathbf{P}(\mathsf{crs}, x, w)$. Adaptive soundness posits that no computationally bounded malicious prover $\tilde{\mathbf{P}}$ causes the verifier to accept with noticeable probability in the following game: $\mathsf{crs} \leftarrow \mathsf{Gen}$ is sampled honestly, then $\tilde{\mathbf{P}}(\mathsf{crs})$ outputs (x, π) with $x \notin L(R)$, and the verifier is then given crs, x, and π. The standard definition of SNARG additionally requires the size of π to be sub-linear in the size of w, and for verification time $\mathrm{poly}(|\mathsf{crs}|, |x|, |\pi|)$.

We say that $(\mathsf{Gen}, \mathbf{P}, \mathbf{V})$ is a *pre-processing* SNARG if the CRS can be described as two strings $\mathsf{crs} = (\mathsf{crs}_1, \mathsf{crs}_2)$, and verifier can be described as a two-stage machine $\mathbf{V} = (\mathbf{V}_1, \mathbf{V}_2)$ such that $\mathbf{V}(\mathsf{crs}, x, \pi) = \mathbf{V}_2(\mathsf{crs}_2, x', \pi)$ where $x' \leftarrow \mathbf{V}_1(\mathsf{crs}_1, x)$. In other words $\mathbf{V}_1$ is an offline pre-processing phase that is based only on crs_1 and x, and $\mathbf{V}_2$ is on online phase that makes the decision given

crs_2, x', and the prover message π. The standard parameter setting requires $\mathbf{V}_2$ to run in time that is sublinear in $|x|$, meaning that both crs_2 and x' are sublinear in the size of x.

Pre-processing SNARGs and CIC. We observe that an adaptively sound pre-processing SNARG for a relation R can be viewed as an adaptively sound CIC: Indeed, we let the CRS generation algorithm of the CIC be the same as that of the SNARG, set $\mathsf{IC}(\mathsf{crs}, x) := \mathbf{V}_1(\mathsf{crs}, x)$, and $\mathsf{WT}(\mathsf{crs}, x, \pi) := \mathbf{P}(\mathsf{crs}, x, \pi)$. The target relation is $R' := \{(x', \pi) \mid \mathbf{V}_2(\mathsf{crs}, x', \pi) = 1\}$.[3]

It is immediate by construction that the CIC has perfect completeness, and adaptive soundness follows from the soundness of the SNARG, since finding a proof π that causes the verifier to accept is identical to finding a witness that puts x' into R'. Compression stems from the fact that $|x'| \ll |x|$.

Note that, while pre-processing SNARGs typically require $|\pi| \ll |w|$ and that $\mathbf{V}_2$ run in sublinear time in $|x|$, the construction of CIC only requires $|\pi| = \mathrm{poly}(|w|)$, and $\mathbf{V}_2$ to run in time $\mathrm{poly}(|\mathsf{crs}|, |x|, |w|)$. Thus, it also applies to a more relaxed notion of pre-processing SNARG with these properties.

SNARGs and CRH. The above observations bring a new perspective to the recent constructions of adaptively sound SNARGs [WW24b; WZ24; WW24a] from one-way functions (OWF) and indistinguishability obfuscation (iO).

In [WW24b; WZ24] Waters and Wu, and Waters and Zhandry construct SNARGs with adaptive soundness from OWF and iO, along with re-randomizable one-way functions (ROWF) [WW24b] or lossy functions [WZ24]. In both papers, the verifier, given a CRS containing (the obfuscation of) a "challenge generation" algorithm $\mathsf{GenInst}$, instance x, and prover message π, works as follows: (1) run $\mathsf{GenInst}$ on x to receive challenge x', and (2) accept if and only if $f(x') = \pi$, where f is a one-way function. We observe that in both papers $|x'| \ll |x|$, and that $\mathsf{GenInst}$ does not use π. Therefore, the verifier can be described as a pre-processing SNARG where $\mathbf{V}_1$ runs $\mathsf{GenInst}$ on x and outputs x', and $\mathbf{V}_2$, given x' and π, outputs 1 if and only if $f(x') = \pi$.

In light of Theorem 1.2 and the above discussion, we achieve constructions of CRH that rely on OWFs and iO, combined with either ROWF or lossy functions. We observe that while Theorem 1.2 is non-black-box in the sense of [Sim98], the overall construction could still be described using oracle-aided circuits and thus capture under the impossibility of [AS16], which rules out certain constructions of CRH from OWF and iO. Thus, ROWFs and lossy functions are the crucial stepping stone towards pre-processing SNARGs and through them to CIC and CRH.

[WW24a] construct adaptively sound SNARGs from only OWFs and iO in a manner captured by the model of [AS16]. Unlike the prior works, their verifier does not use a similar $\mathsf{GenInst}$ paradigm and cannot be naturally described as a two-stage algorithm that implies a pre-processing SNARG. Our new connection to CIC explains that this fact is inherent as otherwise, the [WW24a] construction would imply a CRH from OWF and iO, contradicting [AS16].

[3] We remark that while R' depends on crs, this suffices for our construction of CRH.

3 Preliminaries

In this section we define objects and state results that we use throughout this paper.

3.1 Collision resistant hash scheme

A collision-resistant hash family is a tuple of algorithms $(\mathcal{H}.\mathsf{Gen}, \mathcal{H}.\mathsf{Hash})$ where $\mathcal{H}.\mathsf{Gen}$ is probabilistic polynomial-time algorithm and $\mathcal{H}.\mathsf{Hash}$ is deterministic polynomial-time algorithm with the following properties:

- *Collision resistance.* There exist a negligible function μ such that for every adversary $\mathbf{A}$ and for every $\lambda, N \in \mathbb{N}$:

$$\Pr\left[\begin{array}{l} x \neq x' \\ \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x) = \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x') \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^N) \\ (x, x') \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right] = \mu(\lambda)\,.$$

- *Compressibility.* For every $\lambda, N \in \mathbb{N}$, and for every $x \in [N]$, for hk in the image of $\mathcal{H}.\mathsf{Hash}(1^\lambda, 1^N)$, the size of the output of $\mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x)$ is λ.

3.2 Commitment schemes

A non-interactive commitment scheme (in the CRS model) for the message space $\{0,1\}^\ell$ is a tuple of polynomial-time algorithms $(\mathsf{Gen}, \mathsf{CM.Com}, \mathsf{CM.Ver})$ that allows a sender to commitment to a message to a receiver by sending a *commitment*. The commitment does not reveal the content of the message. At a later stage, the sender can *open* the commitment, revealing the true content of the commitment. Crucially, the sender is not able to open to a different message than the one it commitment to in the first stage.

Formally, $\mathsf{CM.Gen}$ and $\mathsf{CM.Com}$ are randomized algorithms and $\mathsf{CM.Ver}$ is a deterministic algorithm that satisfy the following properties.

- *Completeness.* For every $\lambda \in \mathbb{N}$, and any message $\mathsf{msg} \in \{0,1\}^\ell$:

$$\Pr\left[\begin{array}{l} \mathsf{CM.Ver}(\mathsf{crs}, \sigma, \mathsf{msg}, \rho) = 1 \end{array} \middle| \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{CM.Gen}(1^\lambda) \\ \rho \leftarrow \{0,1\}^{\mathsf{r}} \\ \sigma = \mathsf{CM.Com}(\mathsf{crs}, \mathsf{msg}, \rho) \end{array}\right] = 1\,.$$

- *Statistical binding.* There exist a negligible function μ such that for every $\lambda \in \mathbb{N}$:

$$\Pr\left[\begin{array}{l} \exists\, \mathsf{msg}_0, \mathsf{msg}_1, \rho_0, \rho_1, \sigma \text{ s.t.} \\ \mathsf{CM.Ver}(\mathsf{crs}, \sigma, \mathsf{msg}_0, \rho_0) = 1 \\ \wedge\, \mathsf{CM.Ver}(\mathsf{crs}, \sigma, \mathsf{msg}_1, \rho_1) = 1 \\ \wedge\, \mathsf{msg}_0 \neq \mathsf{msg}_1 \end{array} \middle| \begin{array}{l} \mathsf{crs} \leftarrow \mathsf{CM.Gen}(1^\lambda) \end{array}\right] = \mathsf{negl}(\lambda)$$

- *Computational hiding.* There exist a negligible function μ such that for every $\lambda \in \mathbb{N}$, any two

message $\mathsf{msg}_0, \mathsf{msg}_1 \in \{0,1\}^\ell$, and any polynomial-sized distinguisher $\mathbf{A}$, it holds that:

$$\left|\Pr\left[\mathbf{A}(\mathsf{crs}, \mathsf{msg}_0, \mathsf{msg}_1, \mathsf{CM.Com}) = 1 \;\middle|\; \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{CM.Gen}(1^\lambda)\\ \rho \leftarrow \{0,1\}^{\mathsf{r}}\\ \sigma = \mathsf{CM.Com}(\mathsf{crs}, \mathsf{msg}_0, \rho)\end{array}\right]\right| -$$
$$\left|\Pr\left[\mathbf{A}(\mathsf{crs}, \mathsf{msg}_0, \mathsf{msg}_1, \mathsf{CM.Com}) = 1 \;\middle|\; \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{CM.Gen}(1^\lambda)\\ \rho \leftarrow \{0,1\}^{\mathsf{r}}\\ \sigma = \mathsf{CM.Com}(\mathsf{crs}, \mathsf{msg}_1, \rho)\end{array}\right]\right| = \mathsf{negl}(\lambda)$$

Theorem 3.1 ([Nao91])**.** *If one-way functions exist, then there exists a statistically binding and computationally hiding commitment scheme.*

3.3 NIZK

A non-interactive zero knowledge (NIZK) proof system for a relation R is a tuple of algorithms $(\mathsf{NIZK.Gen}, \mathsf{NIZK.}\mathbf{P}, \mathsf{NIZK.}\mathbf{V})$ where $\mathsf{NIZK.Gen}$ and $\mathsf{NIZK.}\mathbf{P}$ are probabilistic polynomial-time algorithms and $\mathsf{NIZK.}\mathbf{V}$ is deterministic polynomial-time algorithm with the following properties:

- *Completeness.* For every $(x, w) \in R$, and for every $\lambda \in \mathbb{N}$:

$$\Pr\left[\mathsf{NIZK.}\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \;\middle|\; \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda, 1^{|x|})\\ \pi \leftarrow \mathsf{NIZK.}\mathbf{P}(\mathsf{crs}, x, w)\end{array}\right] = 1\,.$$

- *Statistical soundness.* There exist a negligible function μ such that for every $\lambda, n \in \mathbb{N}$:

$$\Pr\left[\begin{array}{l}\exists(x,\pi) \text{ s.t.}\\ x \notin L(R)\\ \wedge\ \mathsf{NIZK.}\mathbf{V}(\mathsf{crs}, x, \pi) = 1\end{array} \;\middle|\; \mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda, 1^n)\right] = \mathsf{negl}(\lambda)\,.$$

- *Adaptive computational zero knowledge.* There exists a PPT simulator $\mathbf{S} = (\mathbf{S}_1, \mathbf{S}_2)$ such that for every non-uniform polynomial-size adversary $\tilde{\mathbf{V}} = (\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2)$ and every $\lambda, n \in \mathbb{N}$,

$$\left|\Pr\left[\begin{array}{l}x \in L(R)\\ \wedge\ \tilde{\mathbf{V}}_2(\mathsf{crs}, x, \pi, \mathsf{st}) = 1\end{array} \;\middle|\; \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda, 1^n)\\ (x, w, \mathsf{st}) \leftarrow \tilde{\mathbf{V}}_1(\mathsf{crs})\\ \pi \leftarrow \mathsf{NIZK.}\mathbf{P}(\mathsf{crs}, x, w)\end{array}\right]\right.$$
$$\left. - \Pr\left[\begin{array}{l}x \in L(R)\\ \wedge\ \tilde{\mathbf{V}}_2(\mathsf{crs}, x, \pi, \mathsf{st}) = 1\end{array} \;\middle|\; \begin{array}{r}(\mathsf{crs}, \mathsf{aux}) \leftarrow \mathbf{S}_1(1^\lambda, 1^n)\\ (x, w, \mathsf{st}) \leftarrow \tilde{\mathbf{V}}_1(\mathsf{crs})\\ \pi \leftarrow \mathbf{S}_2(\mathsf{crs}, x, \mathsf{aux})\end{array}\right]\right| = \mathsf{negl}(\lambda)$$

Theorem 3.2 ([PS19])**.** *Assuming the hardness of LWE with suitable polynomial factors, for every* NP *language, there exists non-interactive zero knowledge proof.*

3.4 Incompressible PKE

We use the definition of incompressible encryption by Guan et al. [GWZ22]. The syntax of incompressible PKE is analogous to that of a standard PKE scheme, except that Gen algorithm gets an additional security parameter S, which is the space bound of the adversary. Informally, the

security guarantee is that any adversary, even given the secret key, needs to know more than S bits of information on the ciphertext in order to distinguish two encrypted messages.

Formally, an incompressible public-key encryption scheme is a tuple of algorithms $(\mathsf{IE.Gen}, \mathsf{IE.Enc}, \mathsf{IE.Dec})$ where $\mathsf{IE.Gen}$ and $\mathsf{IE.Enc}$ are probabilistic polynomial-time algorithms, and $\mathsf{IE.Dec}$ is a polynomial-time algorithm with the following properties:

- *Correctness.* For every $\lambda, S \in \mathbb{N}$, messages $\mathsf{msg} \in \{0,1\}^*$,

$$\Pr\left[\mathsf{msg} = \mathsf{IE.Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{IE.Enc}(\mathsf{pk}, \mathsf{msg})) \,\middle|\, (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{IE.Gen}(1^\lambda, S) \right] = 1\,.$$

- *Incompressible encryption security.* For every adversary $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2)$, we define the experiment $\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)$ as follows:

  1. The adversary $\mathbf{A}_1$, on input 1^λ, outputs a space bound 1^S.
  2. Generate $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{IE.Gen}(1^\lambda, S)$.
  3. Sample $b \leftarrow \{0,1\}$ uniformly at random.
  4. The adversary $\mathbf{A}_1$ is then provided the public key pk and submits an auxiliary input aux, msg_0, and msg_1.
  5. The adversary $\mathbf{A}_1$ then receives $\mathsf{ct} \leftarrow \mathsf{IE.Gen}(\mathsf{pk}, \mathsf{msg}_b)$, and submits a state st of size at most S.
  6. The adversary $\mathbf{A}_2$ receives $(\mathsf{pk}, \mathsf{sk}, \mathsf{aux}, \mathsf{st})$ and outputs a guess b'.
  7. If $b = b'$ then the adversary succeeds and the experiment outputs 1. Otherwise, the experiment outputs 0.

  For every $\lambda \in \mathbb{N}$, for every probabilistic polynomial-time adversary $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2)$ there exists a negligible function s such that:

$$\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1\right] \leq \frac{1}{2} + \mathsf{s}(\lambda)\,.$$

  Note that we define $\mathsf{IE.Gen}$ to run in time that is $\mathrm{poly}(\lambda, \log S)$, which is a restriction compared to the definition in [GWZ22] that allows the function to run in time $\mathrm{poly}(\lambda, S)$. In a closer look at their incompressible PKE construction one can see that the running time of their generation algorithm is indeed $\mathrm{poly}(\lambda, S)$.

Theorem 3.3 ([GWZ22])**.** *If there exists a CPA secure public key encryption scheme, then there exists incompressible PKE.*

4 Computational instance compression

Instance compression, introduced by [HN10], is an algorithm that compresses a large instance into a small instance with size that is proportional to the witness size (rather than the original instance). In more details, an *instance compression* from a source relation $R \in \text{NP}$ to a target relation $R' \in \text{NP}$ is a polynomial time algorithm f that takes as input an instance x and output $x' = f(x)$. The compressed instance x' is of size proportional to the witness size, and $x' \in L(R')$ if and only if $x \in L(R)$.

Although instance compression has proven useful in various settings ([HN10; BDFH09]), it is strongly believed that this object does not exist for certain NP-complete languages, as shown by Fortnow and Santhanam [FS11] (under the assumption that coNP $\not\subseteq$ NP/*poly*). To overcome this limitation, Bronfman and Rothblum [BR22] introduced and constructed the new notion of *computational instance compression* (CIC), which relaxes the requirements of instance compression. While their work focused on the non-adaptive setting, the notion of adaptive soundness is critical for many cryptographic applications. In our work, we define CIC in the adaptive setting.[4]

Definition 4.1 (Computational instance compression)**.** *A computational instance compression scheme for a source relation R and target relation R' with instance compression parameter k, witness size z, adversary size $T : \mathbb{N} \to \mathbb{N}$, is a tuple of polynomial-time algorithms* $(\mathsf{Gen}, \mathsf{IC}, \mathsf{WT})$ *where* Gen *is probabilistic, and* IC *and* WT *are deterministic algorithms. We say that the scheme has perfect completeness and adaptive soundness error* $\mathsf{s}\colon \mathbb{N} \to [0,1]$ *if the following holds:*

- Perfect completeness. *For every $\lambda \in \mathbb{N}$, $x \in \{0,1\}^n, w \in \{0,1\}^m$ such that $(x,w) \in R$,*

$$\Pr\left[(\mathsf{IC}(\mathsf{crs}, x), \mathsf{WT}(\mathsf{crs}, x, w)) \in L(R') \mid \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)\right] = 1\,.$$

- Adaptive soundness. *For every $\lambda \in \mathbb{N}$, $n, m \in \mathbb{N}$, for every $T(\lambda)$-sized adversary $\mathbf{A}$:*

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, (\mathsf{IC}(\mathsf{crs}, x), w') \in R' \end{array} \;\middle|\; \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ (x, w') \leftarrow \mathbf{A}(\mathsf{crs}) \end{array}\right] \leq \mathsf{s}(\lambda)\,.$$

- Instance succinctness. *For every* crs *in the image of* $\mathsf{Gen}(1^\lambda, 1^n, 1^m)$ *and $x \in \{0,1\}^n$,* $\mathsf{IC}(\mathsf{crs}, x) \in \{0,1\}^{\leq k}$*, where $k := k(\lambda, n, m)$.*

- Witness succinctness. *For every* crs *in the image of* $\mathsf{Gen}(1^\lambda, 1^n, 1^m)$*, and for every $x \in \{0,1\}^n$, $w \in \{0,1\}^m$ such that $(x,w) \in R$, we have that* $\mathsf{WT}(\mathsf{crs}, x, w) \in \{0,1\}^{\leq z(\lambda,n,m)}$*, where $z := z(\lambda, n, m)$.*

Unless stated otherwise, we assume a CIC scheme with instance compression parameter $k = \text{poly}(\lambda, \log n, m)$ and witness size $z = \text{poly}(\lambda, n, m)$, security against adversaries of size $T = \text{poly}(\lambda)$, and soundness error $\mathsf{s} = \mathsf{negl}(\lambda)$*. We say that the CIC is also* witness-succinct *if $z = \text{poly}(\lambda, \log n, m)$.*

Remark 4.2. The CIC defined in [BR22] refers to the non-adaptive setting where the soundness requirement is replaced with the following:

[4] The notion of CIC with adaptive soundness was implicitly referred to in [Ben24].

Non-adaptive soundness. For every $\lambda \in \mathbb{N}$, $n, m \in \mathbb{N}$, for every $T(\lambda)$-sized adversary $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2)$:

$$\Pr\left[\begin{array}{l|r} x \in \{0,1\}^n & (\mathsf{aux}, x) \leftarrow \mathbf{A}_1 \\ \wedge\, x \notin L(R) & \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ \wedge\, (\mathsf{IC}(\mathsf{crs}, x), w') \in R' & w' \leftarrow \mathbf{A}_2(\mathsf{crs}, \mathsf{aux}) \end{array}\right] \leq \mathsf{s}(\lambda)\,.$$

5 Impossibility of CIC with strong soundness

In this section, we show that a variant of CIC with a stronger notion of soundness and relaxed witness transformation requirement does not exist under standard complexity assumptions.

Definition 5.1. *A computational instance compression algorithm with strong soundness for a source relation R and target relation R' is a computational instance compression (Definition 4.1), with the following modifications:*

- Inefficient witness transformation. *The witness transformation algorithm* WT *is allowed to run in unbounded time.*
- Strong adaptive soundness. *The scheme has the following stronger soundness guarantee: For every $\lambda \in \mathbb{N}$, $n, m \in \mathbb{N}$, for every $T(\lambda)$-sized adversary $\mathbf{A}$:*

$$\Pr\left[x \in \{0,1\}^n \wedge x \notin L(R) \wedge \mathsf{IC}(\mathsf{crs}, x) \in L(R') \;\middle|\; \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ x \leftarrow \mathbf{A}(\mathsf{crs}) \end{array}\right] \leq \mathsf{s}(\lambda)\,.$$

We show that there is no computational instance compression with strong soundness for NP unless coNP $\subseteq$ NP/*poly*. Specifically, use the following NP-complete relation:

Definition 5.2. *The OR-SAT relation R_{ORSAT} is defined as follows*

$$R_{\mathsf{ORSAT}} := \{(\phi, w) = ((\varphi_1, \ldots, \varphi_t), w) \mid \exists i \in [t],\ w \text{ is a satisfying assignment for } \varphi_i\}\,.$$

We state and prove the barrier for constructing CIC with strong soundness. Our results hold even for scheme with soundness error that is exponentially close 1:

Theorem 5.3. *If there exists a strong CIC from R_{ORSAT} to a target relation $R' \in$ NP with compression parameter $k(\lambda, n, m) = \mathrm{poly}(\lambda, \log n, m)$ and strong soundness error $\mathsf{s}(\lambda) \leq 1 - 2^{-\lambda}$ against any polynomial size adversary, then* $\mathsf{UNSAT} \in$ NP/*poly*.

Proof. We construct an NP/*poly* verifier for UNSAT. Let $(\mathsf{Gen}, \mathsf{IC}, \mathsf{WT})$ denote the algorithms of the CIC scheme and $A = L(R')$ denote its target language. We use the CIC scheme in order to construct an advice string for the NP/*poly* verifier. Recall that, by assumption, for any security parameter $\lambda \in \mathbb{N}$, instance size $\ell \in \mathbb{N}$, and witness size $z \in \mathbb{N}$, it holds that $k(\lambda, \ell, z) = \mathrm{poly}(\lambda, \log \ell, z)$.

We use the following notation and setting of parameters:

- $k := k(\lambda, \ell, z)$,
- c is the smallest constant such that $k(\lambda, \ell, z) \leq (\lambda + \log \ell + z)^c$,
- $\ell := n^{2c+2}$, $z := n$, $\lambda := n$, $t := \ell / z$,
- $T := \bar{A} \cap \{0,1\}^{\leq k}$, where $\bar{A}$ is the complement of A,
- $S = \mathsf{UNSAT} \cap \{0,1\}^z$ is the set of unsatisfiable formulas of size at most z,
- $X := S^t$ is the set of unsatisfiable ORSAT instances comprised of t formulas, each of size z.

We now describe our verification algorithm:

Construction 5.4. We describe an algorithm that constructs the advice string, and then describe the UNSAT verifier. Let $\mathsf{s} := \mathsf{s}(\lambda)$.

- **Constructing the advice U.** For length n:

- Initialize $S_1 := S$, and $i := 1$.
- While $S_i \neq \emptyset$ do the following:
  1. Set $X_i := S_i^t$.
  2. Set crs_i to be a CRS in the image of $\mathsf{Gen}(1^\lambda, 1^\ell, 1^z)$ such that for the set
  $$|\{\phi \in X_i \mid \mathsf{IC}(\mathsf{crs}_i, \phi) \in T\}| > (1 - \mathsf{s}) \cdot |X_i| \,.$$
  (see Claim 5.5 for why such a choice exists.)
  3. Set $y_i := \text{argmax}_{y \in T} |\{\phi \mid \phi \in X_i,\ \mathsf{IC}(\mathsf{crs}_i, \phi) = y\}|$. In other words, y_i is an element in T with a maximum number of pre-images in X_i.
  4. Set $S_{i+1} := S_i \setminus \{\varphi \mid \varphi \in S_i,\ \exists \phi \in X_i \text{ s.t. } (\varphi \in \phi) \wedge (\mathsf{IC}(\mathsf{crs}_i, \phi) = y_i)\}$. In other words, S_{i+1} is the the set of all elements in S_i that are not part of a tuple $\phi \in X_i$ such that $\mathsf{IC}(\mathsf{crs}_i, \phi) = y_i$.
  5. Set $i := i + 1$.
- Output $U := \{\mathsf{crs}_j, y_j\}_{j \in [i-1]}$

- **The verifier.** Given the advice U constructed as above, an instance $\varphi \in \{0,1\}^n$, and a witness $w \in \{0,1\}^m$, the verifier does the following:

  1. Parse $w := (\varphi_1, \ldots, \varphi_t)$ (reject if it cannot be correctly parsed).
  2. Output 1 if and only if both of the following checks pass:
     (a) $\varphi \in (\varphi_1, \ldots, \varphi_t)$, and
     (b) there exists a tuple $(\mathsf{crs}, y) \in U$ such that $\mathsf{IC}(\mathsf{crs}, (\varphi_1, \ldots, \varphi_t)) = y$.

  We now prove efficiency, completeness, and soundness of the verifier.

Efficiency. The verifier receives advice of size $|U|$ and runs in time $O(|U| \cdot \text{poly}(n))$. We show that $|U| = \text{poly}(n)$, and so the verifier runs in polynomial time and receives a polynomial-length advice string. We do this by first showing that for every i there is always a choice of CRS for the advice construction algorithm in Item 2, and that, $|S_{i+1}| \leq \frac{1}{2} \cdot |S_i|$. This suffices to prove the claim, since $|S| \leq 2^n$ (since S is comprised of formulas of binary representation size n), and so there is a constant d such that $|S_{d \cdot n}| \leq 1$ after which point $S_{d \cdot n + 1}$ is empty, and the algorithm stops.

The following claim shows that the algorithm always has a choice of CRS:

Claim 5.5. *There exists* crs_i *in the image of* $\mathsf{Gen}(1^\lambda, 1^\ell, 1^z)$ *such that,*
$$|\{\phi \in X_i \mid \mathsf{IC}(\mathsf{crs}_i, \phi) \in T\}| > (1 - \mathsf{s}) \cdot |X_i| \,.$$

Proof. Equivalently, we prove that
$$\Pr\left[\mathsf{IC}(\mathsf{crs}_i, \phi) \in T \mid \phi \leftarrow X_i\right] > 1 - \mathsf{s} \,.$$
Since $X_i \subseteq X$, for every $\phi \in X_i$ it holds that $\phi \notin \mathsf{ORSAT}$. Assume toward contradiction (to strong soundness of the CIC) that for every crs in the image of $\mathsf{Gen}(1^\lambda, 1^\ell, 1^z)$:
$$\Pr\left[\mathsf{IC}(\mathsf{crs}, \phi) \notin T \mid \phi \leftarrow X_i\right] > \mathsf{s} \,.$$
Or equivalently, by the definition of $T := \bar{A} \cap \{0,1\}^{\leq k}$, and recalling that IC never outputs elements of length larger than k,
$$\Pr\left[\mathsf{IC}(\mathsf{crs}, \phi) \in A \mid \phi \leftarrow X_i\right] > \mathsf{s} \,.$$

Therefore,

$$\Pr\left[\mathsf{IC}(\mathsf{crs},\phi)\in A \;\middle|\; \begin{array}{l}\phi\leftarrow X_i\\ \mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\end{array}\right]\geq \mathsf{s}\,.$$

By the law of total probability,

$$\Pr\left[\mathsf{IC}(\mathsf{crs},\phi)\in A \;\middle|\; \begin{array}{l}\phi\leftarrow X_i\\ \mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\end{array}\right]=\sum_{\phi\in X_i}\frac{\Pr\left[\mathsf{IC}(\mathsf{crs},\phi)\in A \mid \mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\right]}{|X_i|}>\mathsf{s}\,.$$

Therefore, by an averaging argument, there exists $\phi\in X_i$ for which,

$$\Pr\left[\mathsf{IC}(\mathsf{crs},\phi)\in A \mid \mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\right]>\mathsf{s}\,. \tag{1}$$

Let ϕ be the formula described above. We describe an adversary $\mathbf{A}$ for breaking strong soundness of the CIC. $\mathbf{A}$ has ϕ hardwired, and for every crs, $\mathbf{A}$ always outputs ϕ. By Equation 1,

$$\Pr\left[\begin{array}{l}\mathsf{IC}(\mathsf{crs},\phi)\in A\end{array}\;\middle|\; \begin{array}{l}\mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\\ \phi\leftarrow \mathbf{A}(\mathsf{crs})\end{array}\right]>\mathsf{s}\,.$$

Note that since $\phi\in X_i$, it holds that $\phi\notin\mathsf{ORSAT}$. Moreover, the size of ϕ is $\ell=n^{2c+2}=\mathrm{poly}(\lambda)$, which implies that the size of $\mathbf{A}$ is $\mathrm{poly}(\lambda)$. We get a poly-sized adversary $\mathbf{A}$ such that,

$$\Pr\left[\begin{array}{l}\phi\notin\mathsf{ORSAT}\\ \mathsf{IC}(\mathsf{crs},\phi)\in A\end{array}\;\middle|\; \begin{array}{l}\mathsf{crs}\leftarrow \mathsf{Gen}(1^\lambda,1^\ell,1^z)\\ \phi\leftarrow \mathbf{A}(\mathsf{crs})\end{array}\right]>\mathsf{s}\,,$$

as a contradiction to the strong soundness guaranty of the CIC scheme. □

We now show that the sets S_i decrease in size by half in each iteration:

Claim 5.6. $|S_{i+1}|\leq\frac{1}{2}\cdot|S_i|\,.$

Proof. Let $Y_i:=\{\phi\in X_i \mid \mathsf{IC}(\mathsf{crs}_i,\phi)=y_i\}$, and let $X_i':=\{\phi\in X_i \mid \mathsf{IC}(\mathsf{crs}_i,\phi)\in T\}$ be the set of all formulas $\phi\in X_i$ such that $\mathsf{IC}(\mathsf{crs}_i,\phi)\in T$. By the definition of y_i together with the pigeonhole principle we get that,

$$|Y_i|\geq\frac{|X_i'|}{|T|}\,.$$

Plugging in the fact that, $|X_i'|\geq(1-\mathsf{s})\cdot|X_i|$ (derived from Claim 5.5), we get that $|Y_i|\geq|X_i|\cdot\left(\frac{1-\mathsf{s}}{|T|}\right)$. Recall that the CIC always outputs elements of length at most k, and so $|T|\leq 2^{k+1}$. Thus,

$$|Y_i|\geq\frac{|X_i'|}{|T|}\geq\frac{(1-\mathsf{s})\cdot|X_i|}{2^{k+1}}=|S_i|^t\cdot\left(\frac{1-\mathsf{s}}{2^{k+1}}\right)\,,$$

where the equality is by the definition of X_i. By assumption, we have that $\mathsf{s}\leq 1-2^{-\lambda}$. Therefore, $1-\mathsf{s}\geq 2^{-\lambda}$, and we get that,

$$|Y_i|\geq|S_i|^t\cdot\left(\frac{1-\mathsf{s}}{2^{k+1}}\right)\geq|S_i|^t\cdot\frac{1}{2^{\lambda+k+1}}\,. \tag{2}$$

Next we upper bound the size of Y_i. Note that every subformula of any $\phi \in Y_i$ is in S_i since $Y_i \subseteq X'_i \subseteq X_i = S_i^t$. On the other hand, S_{i+1} is formed by removing the elements in S_i that are included in a tuple in Y_i. Therefore,

$$|S_i \setminus S_{i+1}|^t \geq |Y_i| \,. \tag{3}$$

By combining Equation 2 and Equation 3, we get that,

$$|S_i \setminus S_{i+1}|^t \geq |Y_i| \geq |S_i|^t \cdot \frac{1}{2^{\lambda+k+1}} \,.$$

Therefore,

$$|S_i \setminus S_{i+1}| \geq |S_i| \cdot \left(\frac{1}{2^{\lambda+k+1}}\right)^{1/t} \,.$$

Since $k \leq (\lambda + \log \ell + z)^c$, $\lambda = n$, $\ell = n^{2c+2}$, $z = n$, and $t = \ell/z$, we get that for large enough n,

$$\begin{aligned}(\lambda + k + 1)/t &\leq (\lambda + (\lambda + \log \ell + z)^c + 1)/t \\ &= (n + (\log n^{2c+1} + 2n)^c + 1)/n^{2c+1} \\ &\leq n \cdot (3n)^c / n^{2c+1} \\ &\leq n^{2c+1}/n^{2c+1} = 1\end{aligned}$$

Therefore, $|S_i \setminus S_{i+1}| \geq \frac{|S_i|}{2}$. Since $S_{i+1} \subseteq S_i$, it holds that $|S_i \setminus S_{i+1}| = |S_i| - |S_{i+1}|$, and so we conclude that $|S_{i+1}| \leq \frac{1}{2} \cdot |S_i|$, as required. □

Completeness. Let $\varphi \in \mathsf{UNSAT} \cap \{0,1\}^n$. To prove completeness, we show that there exists a witness of size $m = n^{2c+1}$ that will make the verifier accept, i.e., that there exist $\varphi_1, \ldots, \varphi_t \in \{0,1\}^z$ such that (a) $\varphi \in (\varphi_1, \ldots, \varphi_t)$, and (b) for some $(\mathsf{crs}, y) \in U$ it holds that $\mathsf{IC}(\mathsf{crs}, (\varphi_1, \ldots, \varphi_t)) = y$.

Following the construction of U, first note that since $S = \mathsf{UNSAT} \cap \{0,1\}^n$ it holds that $\varphi \in S$. At the beginning, the algorithm is initiated with $S_1 = S$, and therefore, $\varphi \in S_1$. When the algorithm halts, we have that $S_i = \emptyset$. Therefore, there exists j such that $\varphi \in S_j$ and $\varphi \notin S_{j+1}$.

Let $(\mathsf{crs}_j, y_j) \in U$ be the CRS and target element chosen at the j-th stage of the algorithm. In Item 4, S_{j+1} is formed by removing from S_j the formulas φ' such that there exists $\phi \in X_j$ where $\varphi' \in \phi$ and $\mathsf{IC}(\mathsf{crs}_j, \phi) = y_j$. Since we know that φ is removed from S_j, there exists $\phi \in X_j$ such that $\varphi \in \phi$ and $\mathsf{IC}(\mathsf{crs}_j, \phi) = y_j$. Fix such ϕ. By the definition of X_j, we have that $X_j = S_j^t \subseteq S^t$. Therefore, ϕ is a list of t sub-formulas $(\varphi_1, \ldots, \varphi_t)$ such that $\varphi \in (\varphi_1, \ldots, \varphi_t)$ and $\mathsf{IC}(\mathsf{crs}_j, (\varphi_1, \ldots, \varphi_t)) = y_j$, as required.

Soundness. Let $\varphi \in \mathsf{SAT} \cap \{0,1\}^n$. We prove that the algorithm will reject for any $w \in \{0,1\}^m$. Fix some $w \in \{0,1\}^m$. If it is not possible to parse w as $(\varphi_1, \ldots, \varphi_t)$ or is $\varphi \notin (\varphi_1, \ldots, \varphi_t)$, then the algorithm rejects, and so we assume this not to be the case. By assumption, $\varphi \in \mathsf{SAT}$ and so $(\varphi_1, \ldots, \varphi_t) \in \mathsf{ORSAT}$.

Let $\phi = (\varphi_1, \ldots, \varphi_t)$. By the perfect completeness of the CIC, since $\phi \in \mathsf{ORSAT}$, we get that for every crs that is in the image of $\mathsf{Gen}(1^\lambda, 1^\ell, 1^z)$, it holds that $\mathsf{IC}(\mathsf{crs}, \phi) \in A$. On the other hand, for every $(\mathsf{crs}, y) \in U$, it holds that $y \in T \subseteq \bar{A}$. Therefore, the check in Item 2b will fail, and the verifier will reject.

□

6 Collision resistant hashing and CIC

We give new hope for constructing CRH from one-way functions via instance compression. In Section 6.1, we strengthen the proof of [HN10] to rely on this adaptive CIC to construct a CRH family. As our construction of CRH is non-black-box it provides a path bypass the CRH black-box separation, and raises the natural task of a (black-box) construction of adaptive CIC from one-way functions. In Section 6.2, we provide a construction of adaptive CIC from CRH. This means that any construction of CRH from one-way functions must go via adaptive CIC.

6.1 Adaptive CIC implies collision resistance

Theorem 6.1. *If there exist one-way functions and a computational instance compression scheme for* SAT *with instance succinctness parameter* $k(\lambda, n, m) \leq n^{(1-\epsilon)} \cdot \mathrm{poly}(\lambda, m)$ *for* $\epsilon \in (0, 1]$*, then there exists a family of collision-resistance hash functions.*

Proof. Let λ be the security parameter, and let N be the input size. Let $(\mathsf{CM.Gen}, \mathsf{CM.Com}, \mathsf{CM.Ver})$ be a statistically binding and computationally hiding commitment scheme based on OWF (Theorem 3.1), and let $(\mathsf{CIC.Gen}, \mathsf{CIC.IC}, \mathsf{CIC.WT})$ be a CIC scheme for SAT.

Notation. Let crs be in the image of $\mathsf{CM.Gen}(1^\lambda)$, and let σ be a commitment to some value $i \in [N]$. We define $\mathsf{CM.Ver}_{\mathsf{crs},\sigma}$ to be the circuit of $\mathsf{CM.Ver}$ with crs and σ fixed. We denote $|\mathsf{CM.Ver}_{\mathsf{crs},\sigma}|$ to be the size of the circuit $\mathsf{CM.Ver}_{\mathsf{crs},\sigma}$.

The hash family $\mathcal{H}$ is defined as follows:

Construction 6.2. The construction works as follows.

- $\mathsf{Gen}(1^\lambda, 1^N)$. Given security parameter λ and input size N,
  1. Sample index $i \leftarrow [N]$.
  2. Generate $\mathsf{crs}_1 \leftarrow \mathsf{CM.Gen}(1^\lambda)$ and commit $\sigma \leftarrow \mathsf{CM.Com}(\mathsf{crs}_1, i)$.
  3. Let $\mathsf{CM.Ver}_{\mathsf{crs}_1,\sigma}$ be the circuit $\mathsf{CM.Ver}$ with crs_1 and σ fixed.
  4. Set $m := |\mathsf{CM.Ver}_{\mathsf{crs}_1,\sigma}|$, and $n := |\mathsf{CM.Ver}_{\mathsf{crs}_1,\sigma}| + N \cdot \log N$.
  5. Sample $\mathsf{crs}_2 \leftarrow \mathsf{CIC.Gen}(1^\lambda, 1^n, 1^m)$.
  6. Output $\mathsf{hk} = (\sigma, \mathsf{crs}_1, \mathsf{crs}_2)$.
- $\mathsf{Hash}(\mathsf{hk}, x)$. Given hash key $\mathsf{hk} := (\sigma, \mathsf{crs}_1, \mathsf{crs}_2)$, on input $x \in \{0,1\}^N$,
  1. Let ϕ_σ be the SAT analogue of $\mathsf{CM.Ver}_{\mathsf{crs}_1,\sigma}$ over variables $y_1, \ldots, y_\ell$ that represent the value under the commitment, bits of the randomness, and dummy variables.
  2. Set $\phi_x := \left(\bigwedge_{j\in[N]} C_{j,x}\right)$, where $C_{j,x} := (y_1^{1-j_1} \vee \cdots \vee y_\ell^{1-j_\ell})$ if $x_j = 0$ and $C_{j,x} := 1$ if $x_j = 1$.
  3. Set $\phi_{\sigma,x} := (\phi_\sigma \wedge \phi_x)$.
  4. Output $\phi' := \mathsf{CIC.IC}(\mathsf{crs}_2, \phi_{\sigma,x})$.

Efficiency. By efficiency of the commitment and the CIC schemes, the hash function is computable in polynomial time.

Compressibility. Fix some security parameter λ, and fix some constant c such that $k(\lambda, n, m) \leq n^{(1-\epsilon)} \cdot (\lambda + m)^c$. Let $n(\lambda, N), m(\lambda, N)$ be the instance and witness size as a function of the security parameter and input size. By the efficiency of the commitment scheme and by construction,

$$m(\lambda, N) = \text{poly}(\lambda, \log N)\,,$$
$$n(\lambda, N) = \text{poly}(\lambda, \log N) + N \cdot \log N\,.$$

By the above, for large enough constant c' and for large enough N it holds that

$$n \geq (\lambda + w)^{c'/\epsilon}\,.$$

Fix some c' that satisfies the above condition. By the above, for large enough N.

$$k(\lambda, n, m) \leq n^{(1-\epsilon)} \cdot (\lambda + m)^c = n \cdot \frac{(\lambda + m)^c}{n^\epsilon} \leq n \cdot \frac{1}{(\lambda + w)^{c'-c}}\,. \tag{4}$$

By embedding the values of n, m in the above equation, we get that,

$$k(\lambda, n, m) \leq (\text{poly}(\lambda, \log N) + N \cdot \log N) \cdot \frac{1}{(\text{poly}(\lambda, \log N))^{c'-c}}\,.$$

Therefore, for large enough constant c', and for large enough N,

$$k(\lambda, n, m) \ll N\,.$$

Collision resistant. Fix a collision-finding adversary $\mathbf{A}$, and an index $i \in [N]$. Define the following events:

- $\mathsf{E}_{\mathsf{bind}}$: the event that σ has exactly one legal opening.
- $\mathsf{E}_{\mathsf{col}}$: the event that $\mathbf{A}$ outputs $x, x' \in [N]$ with $x \neq x'$ and $\mathsf{Hash}(\mathsf{hk}, x) = \mathsf{Hash}(\mathsf{hk}, x')$.
- $\mathsf{E}_{\mathsf{corr}}$: the event that $\mathbf{A}$ outputs $x, x' \in [N]$ where one of the following is true:
  - $\phi_{\sigma,x} \in \mathsf{SAT}$ and $\phi_{\sigma,x'} \in \mathsf{SAT}$, or
  - $\phi_{\sigma,x} \notin \mathsf{SAT}$ and $\phi_{\sigma,x'} \notin \mathsf{SAT}$.

In the following, we bound $\Pr[\mathsf{E}_{\mathsf{col}}]$. It holds that,

$$\Pr[\mathsf{E}_{\mathsf{col}}] \leq \Pr[\mathsf{E}_{\mathsf{col}} \mid \mathsf{E}_{\mathsf{bind}}] + \Pr[\neg \mathsf{E}_{\mathsf{bind}}]\,.$$

By the binding property of the commitment scheme,

$$\Pr[\mathsf{E}_{\mathsf{col}}] \leq \Pr[\mathsf{E}_{\mathsf{col}} \mid \mathsf{E}_{\mathsf{bind}}] + \mathsf{negl}(\lambda)\,.$$

By the law of total probability,

$$\Pr[\mathsf{E}_{\mathsf{col}} \mid \mathsf{E}_{\mathsf{bind}}] \leq \Pr[\mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] + \Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}]\,.$$

In Claim 6.3 and Claim 6.4 we bound the two probabilities on the right-hand side of the above expression by functions negligible in λ. Overall, we get that,

$$\Pr[\mathsf{E}_{\mathsf{col}}] \leq \mathsf{negl}(\lambda)\,,$$

as required.

Claim 6.3. $\Pr[\mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] \leq \mathsf{negl}(\lambda)$.

Proof. Suppose towards contradiction that for some non-negligible function p,

$$\Pr[\mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] = p(\lambda)\,.$$

We now use $\mathbf{A}$ in order to build an adversary $\mathbf{A}'$ that breaks hiding of σ:

1. Generate hk as in Construction 6.2, but replace σ with the commitment that was given as input.
2. Run $(x, x') \leftarrow \mathbf{A}(\mathsf{hk})$.
3. If $x \neq x'$ and $\mathsf{Hash}(x) = \mathsf{Hash}(x')$,
   (a) Let $S \subseteq [N]$ be the set of all indices j such that $x[j] = x'[j]$.
   (b) Sample $y' \leftarrow S$.
4. Otherwise, sample $y' \leftarrow [N]$.
5. Output y'.

Let i be the value under the commitment σ. Consider the probability that $\mathbf{A}'$ wins (by guessing i) in the following events condition on the biding event $\mathsf{E}_{\mathsf{bind}}$.

- $\Pr[\mathbf{A}' \text{ wins } \wedge \mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}]$: By the completeness of the commitment scheme, ϕ_σ is satisfiable. By the biding event, any satisfying assignment for ϕ_σ must contain $y = i$. Observe that, by the definition of ϕ_x, it holds that $\phi_x(y) = 1$ if and only if $x[y] = 1$. Combining both statements, we get that

  1. $\phi_{\sigma,x}$ is satisfiable if and only if $x[i] = 1$.
  2. $\phi_{\sigma,x'}$ is satisfiable if and only if $x'[i] = 1$.

  By the event $\mathsf{E}_{\mathsf{corr}}$, it holds that $\phi_{\sigma,x}$ and $\phi_{\sigma,x'}$ are either both satisfiable or both unsatisfiable. By the above, we get that $i \in S$. Therefore, $\mathbf{A}'$ guesses i with probability $1/|S|$. By the event $\mathsf{E}_{\mathsf{col}}$, since $x \neq x'$, it must hold that $|S| < N$. Therefore, it holds that $\mathbf{A}'$ guesses i with probability at least $1/N - 1$.

- $\Pr[\mathbf{A}' \text{ wins } \wedge \neg\mathsf{E}_{\mathsf{col}} \mid \mathsf{E}_{\mathsf{bind}}]$: In this case, $\mathbf{A}'$ guesses i uniformly at random, and therefore succeeds with probability $1/N$.

Thus we get that for $i \leftarrow [N]$,

$$\begin{aligned}
&\Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i \mid \mathsf{E}_{\mathsf{bind}}\right] \\
&\quad \geq \Pr\left[\mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}\right] \cdot \Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i \wedge \mathsf{E}_{\mathsf{col}} \wedge \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}\right] \\
&\quad + \Pr\left[\neg\mathsf{E}_{\mathsf{col}} \vee \neg\mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}\right] \cdot \Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i \wedge \neg\mathsf{E}_{\mathsf{col}} \mid \mathsf{E}_{\mathsf{bind}}\right] \\
&\quad = p(\lambda) \cdot \frac{1}{N-1} + (1 - p(\lambda)) \cdot \frac{1}{N} \\
&\quad = \frac{1}{N} + p(\lambda) \cdot \left(\frac{1}{N-1} - \frac{1}{N}\right) \\
&\quad = \frac{1}{N} + p(\lambda) \cdot \left(\frac{1}{N \cdot (N-1)}\right)\,.
\end{aligned}$$

Since $\Pr[\mathsf{E}_{\mathsf{bind}}] \geq 1 - \mathsf{negl}(\lambda)$, we get that,

$$\begin{aligned}\Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i\right] &\geq (1 - \mathsf{negl}(\lambda)) \cdot \Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i \mid \mathsf{E}_{\mathsf{bind}}\right] \\ &\geq \Pr_{\mathsf{crs}_1}\left[\mathbf{A}'(\mathsf{CM.Com}(\mathsf{crs}_1, i)) = i \mid \mathsf{E}_{\mathsf{bind}}\right] - \mathsf{negl}(\lambda) \\ &\geq \frac{1}{N} + p(\lambda) \cdot \left(\frac{1}{N \cdot (N-1)}\right) - \mathsf{negl}(\lambda)\,,\end{aligned}$$

contradicting the computational hiding of the commitment scheme. □

Claim 6.4. $\Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg \mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] \leq \mathsf{negl}(\lambda)$.

Proof. Let $\mathbf{A}'$ be an adversary to the CIC scheme:

1. Given input crs_2.
2. Sample $i \leftarrow [n]$, and randomness $\rho \leftarrow \{0,1\}^{\mathrm{poly}(\lambda)}$ for $\mathsf{CM.Com}$.
3. Generate $\mathsf{crs}_1 \leftarrow \mathsf{CM.Gen}(1^\lambda)$ and compute $\sigma := \mathsf{CM.Com}(\mathsf{crs}_1, i; \rho)$.
4. Set $\mathsf{hk} = (\sigma, \mathsf{crs}_1, \mathsf{crs}_2)$.
5. Run $(x, x') \leftarrow \mathbf{A}(\mathsf{hk})$.
6. If $\mathsf{Hash}(\mathsf{hk}, x) \neq \mathsf{Hash}(\mathsf{hk}, x')$ or $x = x'$, then abort.
7. Set w to be the satisfying assignment to ϕ_σ. (This can be efficiently computed given $(\mathsf{crs}_1, i, \rho)$).
8. If $\phi_{\sigma,x}(w) = 1$, then set $\phi_1 := \phi_{\sigma,x}$, and $\phi_0 := \phi_{\sigma,x'}$. Otherwise, set $\phi_0 := \phi_{\sigma,x}$, and $\phi_1 := \phi_{\sigma,x'}$.
9. Compute $w_0 := \mathsf{CIC.WT}(\mathsf{crs}_2, \phi_1, w)$.
10. Output (ϕ_0, w_0).

By the law of total probability, and by the binding property of the commitment scheme,

$$\begin{aligned}\Pr[\mathbf{A}' \text{ succeeds}] &= \Pr[\mathsf{E}_{\mathsf{bind}}] \cdot \Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{bind}}] + \Pr[\neg\mathsf{E}_{\mathsf{bind}}] \cdot \Pr[\mathbf{A}' \text{ succeeds} \mid \neg\mathsf{E}_{\mathsf{bind}}] \\ &\geq (1 - \mathsf{negl}(\lambda)) \cdot \Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{bind}}] - \mathsf{negl}(\lambda) \\ &= \Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{bind}}] - \mathsf{negl}(\lambda)\,.\end{aligned}$$

By the law of total probability,

$$\Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{bind}}] \geq \Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] \cdot \Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}}]\,.$$

In Claim 6.5 we prove that $\Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}}] = 1$. Therefore,

$$\Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{bind}}] \geq \Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}]\,.$$

Overall, we get that,

$$\Pr[\mathbf{A}' \text{ succeeds}] \geq \Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] - \mathsf{negl}(\lambda)\,.$$

Therefore, by the soundness of the CIC scheme we conclude that,

$$\Pr[\mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \mid \mathsf{E}_{\mathsf{bind}}] \leq \mathsf{negl}(\lambda)\,.$$

We left to prove Claim 6.5.

Claim 6.5. $\Pr[\mathbf{A}' \textit{ succeeds} \mid \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}}] = 1$.

Proof. By the CIC definition,

$$\Pr[\mathbf{A}' \text{ succeeds} \mid \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}}] = \Pr\left[\begin{array}{l|r} \phi_0 \notin \mathsf{SAT} & \mathsf{crs}_2 \leftarrow \mathsf{CIC.Gen}(1^\lambda, 1^n, 1^m) \\ \wedge\ (\mathsf{CIC.IC}(\mathsf{crs}_2, \phi_0), w_0) \in R' & (\phi_0, w_0) \leftarrow \mathbf{A}'(\mathsf{crs}_2) \\ & \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}} \end{array}\right].$$

By the event $\neg\mathsf{E}_{\mathsf{corr}}$ we get that either $\phi_{\sigma,x}$ is satisfiable and $\phi_{\sigma,x'}$ is unsatisfiable or vise versa. Without loss of generality, we assume that $\phi_{\sigma,x}$ is satisfiable and $\phi_{\sigma,x'}$ is unsatisfiable. In the following, we prove that $\phi_{\sigma,x}(w) = 1$. In that case, $\phi_1 = \phi_{\sigma,x}$ and $\phi_0 = \phi_{\sigma,x'}$. By completeness of the CIC scheme,

$$(\mathsf{CIC.IC}(\mathsf{crs}_2, \phi_1), \mathsf{CIC.WT}(\mathsf{crs}_2, \phi_1, w)) \in R'\,.$$

By the event $\mathsf{E}_{\mathsf{col}}$, we get that $\mathsf{CIC.IC}(\mathsf{crs}_2, \phi_0) = \mathsf{CIC.IC}(\mathsf{crs}_2, \phi_1)$. Therefore,

$$(\mathsf{CIC.IC}(\mathsf{crs}_2, \phi_0), \mathsf{CIC.WT}(\mathsf{crs}_2, \phi_1, w)) \in R'\,.$$

Since ϕ_0 is unsatisfiable, the above implies that,

$$\Pr\left[\begin{array}{l|r} \phi_0 \notin \mathsf{SAT} & \mathsf{crs}_2 \leftarrow \mathsf{CIC.Gen}(1^\lambda, 1^n, 1^m) \\ \wedge\ (\mathsf{CIC.IC}(\mathsf{crs}_2, \phi_0), w_0) \in R' & (\phi_0, w_0) \leftarrow \mathbf{A}'(\mathsf{crs}_2) \\ & \mathsf{E}_{\mathsf{col}} \wedge \neg\mathsf{E}_{\mathsf{corr}} \wedge \mathsf{E}_{\mathsf{bind}} \end{array}\right] = 1\,.$$

We now left to prove that if $\phi_\sigma(w) = 1$, then $\phi_{\sigma,x}(w) = 1$. Let ℓ be the number of variables in the formula ϕ_σ, and let $y_1, \ldots y_{\log n} \in \{w_1, \ldots, w_m\}$ be the variables that represent the committed value in σ. Recall that $\phi_{\sigma,x} = \phi_\sigma \wedge \phi_x$, and that ϕ_x is over the variables $y_1, \ldots y_{\log n}$. By the binding event $\mathsf{E}_{\mathsf{bind}}$, we get that i is the only assignment to $y_1, \ldots y_{\log n}$ that satisfies ϕ_σ. Since $\phi_{\sigma,x}$ is satisfiable, it holds that ϕ_x is satisfiable, and that $\phi_x(i) = 1$. Overall we get that if $\phi_\sigma(w) = 1$, then $\phi_{\sigma,x}(w) = 1$, as required. □

□

□

6.2 From collision resistance hashing to adaptive CIC

In this subsection we construct adaptive CIC from collision resistance hash family.

Theorem 6.6. *If there exists a collision resistance hash with hash key of size* $\mathrm{poly}(\lambda)$*, then there exists CIC for* NP *with instance succinctness* $k(\lambda, n, m) = \mathrm{poly}(\lambda)$ *and witness succinctness* $z(\lambda, n, m) = n + m$*.*

Proof. Let R be an NP relation, and let R' be the following relation:

$$R' := \{(\mathsf{hk}, x'), (x, w) \mid (x, w) \in R \wedge \mathsf{Hash}(\mathsf{hk}, x) = x'\}\,.$$

Let λ be the security parameter, let n be an instance size, and let m be the size of the witness. We construct the following CIC scheme from R to R'.

Construction 6.7. The construction is as follows,

- $\mathsf{Gen}(1^\lambda, 1^n, 1^m)$. Given as input security parameter λ, instance size n, and witness size m. Sample $\mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n)$, and output $\mathsf{crs} := \mathsf{hk}$.
- $\mathsf{IC}(\mathsf{crs}, x)$. Given as input $\mathsf{crs} = \mathsf{hk}$, and instance x. Output $x' := (\mathsf{hk}, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x))$.
- $\mathsf{WT}(\mathsf{crs}, x, w)$. Given as input $\mathsf{crs} = \mathsf{hk}$, instance x, and a witness w. Output $w' := (x, w)$.

Completeness follows directly from the construction. In the following, we prove instance and witness succinctness, and adaptive soundness.

Instance and witness succinctness. For the instance size, the IC algorithm outputs hash key of size $\mathrm{poly}(\lambda)$ and hashed value of size λ bits. Therefore, $k(\lambda, n, m) = \mathrm{poly}(\lambda)$. For the witness size, the WT algorithm outputs the original instance and witness, which is of size $z(\lambda, n, m) = n + m$.

Adaptive soundness. Let λ be a security parameter, let n be the instance size, and let m be the witness size. By construction we get that,

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, (\mathsf{IC}(\mathsf{crs}, x), w') \in R' \end{array} \middle| \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ (x, w') \leftarrow \mathbf{A}(\mathsf{crs}) \end{array}\right]$$
$$= \Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, ((\mathsf{hk}, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x)), w') \in R' \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w') \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right]$$

By the definition of R' we get that,

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, ((\mathsf{hk}, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x)), w') \in R' \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w') \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right]$$
$$= \Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, (\tilde{x}, \tilde{w}) \in R \\ \wedge\, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x) = \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, \tilde{x}) \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w' := (\tilde{x}, \tilde{w})) \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right]$$

Note that $x \notin L(R)$ and $(\tilde{x}, \tilde{w}) \in R$ implies that $x \neq \tilde{x}$. Therefore,

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, (\tilde{x}, \tilde{w}) \in R \\ \wedge\, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x) = \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, \tilde{x}) \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w' := (\tilde{x}, \tilde{w})) \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right]$$
$$\leq \Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \neq \tilde{x} \\ \wedge\, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x) = \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, \tilde{x}) \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w' := (\tilde{x}, \tilde{w})) \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right]$$

By the security of the collision resistance hash function, we get that,

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \neq \tilde{x} \\ \wedge\, \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, x) = \mathcal{H}.\mathsf{Hash}(\mathsf{hk}, \tilde{x}) \end{array} \middle| \begin{array}{r} \mathsf{hk} \leftarrow \mathcal{H}.\mathsf{Gen}(1^\lambda, 1^n) \\ (x, w' := (\tilde{x}, \tilde{w})) \leftarrow \mathbf{A}(\mathsf{hk}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Combining all of the above equations concludes the proof by getting the following bound,

$$\Pr\left[\begin{array}{l} x \in \{0,1\}^n \\ \wedge\, x \notin L(R) \\ \wedge\, (\mathsf{IC}(\mathsf{crs}, x), w') \in R' \end{array} \middle| \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ (x, w') \leftarrow \mathbf{A}(\mathsf{crs}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

□

7 Impossibility of CIC with witness retrieval

In [HN10] Harnik and Naor define a notion of instance compression with "witness retrieval" and use such an instance compression scheme to build an oblivious transfer protocol. In this section, we show that under standard cryptographic assumptions the computational equivalent of their definition does not exist.

Definition 7.1. *We say that a computational instance compression algorithm for a source relation R and target relation R' has (efficient)* ***witness retrieval*** *if it has a polynomial time algorithm* WR *with the following guaranty:*

- Witness retrieval. *For every $\lambda \in \mathbb{N}$, $x \in \{0,1\}^n, w \in \{0,1\}^m$ such that $(x,w) \in R$,*

$$\Pr\left[(x', \mathsf{WR}(\mathsf{crs}, x', w)) \in R' \;\middle|\; \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m) \\ x' \leftarrow \mathsf{IC}(\mathsf{crs}, x) \end{array}\right] = 1\,.$$

Note that witness retrieval implies witness transformation using $\mathsf{WT}(\mathsf{crs}, x, w) := \mathsf{WR}(\mathsf{crs}, \mathsf{IC}(\mathsf{crs}, x), w)$*, and so for a CIC with witness retrieval we sometimes omit the algorithm* WT*.*

We show the following:

Theorem 7.2. *If there exists incompressible PKE whose key-generation algorithm runs in time* $\mathrm{poly}(\lambda, \log S)$ *and NIZK for* NP*, then there is no computational instance compression for* NP *with witness retrieval and compression parameter $k(\lambda, n, m) \leq \mathrm{poly}(\lambda, \log n, m)$.*

Proof. Letting $\mathsf{t} := \mathrm{poly}(\lambda, \log S)$ be the running time of the key-generation algorithm, and applying Lemma 7.5, we use the NIZK to transform the incompressible PKE into a stronger incompressible PKE that has *key consistency* (which we define in Section 7.1). In addition, the new incompressible PKE has secret key of size $\mathrm{poly}(\lambda, \mathsf{t}) = \mathrm{poly}(\lambda, \log S)$. Combined with Lemma 7.11, we conclude that there is no computational instance compression with witness retrieval and compression parameter $k(\lambda, n, m) \leq \mathrm{poly}(\lambda, \log n, m)$.

□

We use the incompressible PKE constructed from CPA secure PKE in Theorem 3.3 together with Theorem 7.2 and obtain the following corollary.

Corollary 7.3. *If there exists a CPA secure public key encryption scheme and NIZK for* NP*, then there is no computational instance compression for* NP *with witness retrieval and compression parameter $k(\lambda, n, m) \leq \mathrm{poly}(\lambda, \log n, m)$.*

In Section 7.1 we show how to transform incompressible PKE schemes into ones with key consistency, and in Section 7.2 we show that such schemes imply the nonexistence of CIC schemes with witness retrieval.

7.1 Incompressible PKE with key consistency

In this section we define "key consistency" for an incompressible PKE scheme which will be useful for the impossibility proof for CIC. Roughly, an incompressible PKE scheme has key consistency if the decryption algorithm cannot be made to decrypt incorrectly a correctly encrypted message by altering the secret key. We then show that any incompressible PKE can be adapted to have this property.

Definition 7.4. *We say that incompressible PKE has (statistical)* ***key consistency*** *if there exist a negligible function μ such that for every $\lambda \in \mathbb{N}$, $S = \text{poly}(\lambda)$, and for every (unbounded) adversary* $\mathbf{A}$*,*

$$\Pr\left[\mathsf{IE.Dec}(\mathsf{pk}, \tilde{\mathsf{sk}}, \mathsf{ct}) \notin \{\mathsf{msg}, \perp\} \;\middle|\; \begin{array}{r} (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, S) \\ (\mathsf{msg}, \rho, \tilde{\mathsf{sk}}) \leftarrow \mathbf{A}(\mathsf{pk}) \\ \mathsf{ct} := \mathsf{IE.Enc}(\mathsf{pk}, \mathsf{msg}; \rho) \end{array} \right] = \mu(\lambda)\,.$$

We remark that a similar but incomparable notions have been defined in prior work in very different contexts (e.g., "decryption consistency" [SG02; GLFFLMS10]).

We show that, assuming NIZKs, any incompressible PKE scheme can be transformed into one that has key consistency:

Lemma 7.5. *If there exists incompressible PKE whose key-generation algorithm runs in time* t*, and a NIZK for* NP*, then there exists an incompressible PKE with key consistency and secret key size* $\text{poly}(\lambda, \mathsf{t})$*.*

Proof. Let $(\mathsf{IE.Gen}', \mathsf{IE.Enc}', \mathsf{IE.Dec}')$ be an incompressible PKE scheme. Let $\mathsf{r} := \mathsf{r}(\lambda, S)$ be the number of random bits that the algorithm uses. Let R be the following NP relation,

$$R := \left\{ ((\lambda, S, \mathsf{pk}, \mathsf{sk}), \rho) \mid (\mathsf{pk}, \mathsf{sk}) = \mathsf{IE.Gen}'(1^\lambda, S; \rho) \right\}.$$

Let $(\mathsf{NIZK.Gen}, \mathsf{NIZK.P}, \mathsf{NIZK.V})$ be a NIZK proof system for R with simulator $\mathbf{S} = (\mathbf{S}_1, \mathbf{S}_2)$. We construct an incompressible PKE scheme with statistical key consistency.

Construction 7.6. The construction is as follows,

- $\mathsf{IE.Gen}(1^\lambda, S)$*:*
  1. Sample $\rho \leftarrow \{0,1\}^{\mathsf{r}}$.
  2. Compute $(\mathsf{pk}', \mathsf{sk}') := \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho)$, and set $x := (\lambda, S, \mathsf{pk}', \mathsf{sk}')$ and $w := \rho$.
  3. Generate $\mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda, 1^n)$ and $\pi \leftarrow \mathsf{NIZK.P}(\mathsf{crs}, x, w)$.
  4. Output $\mathsf{pk} := (\lambda, S, \mathsf{pk}', \mathsf{crs})$ and $\mathsf{sk} := (\mathsf{sk}', \pi)$.
- $\mathsf{IE.Enc}(\mathsf{pk}, \mathsf{msg})$*:*
  1. Parse $\mathsf{pk} := (\lambda, S, \mathsf{pk}', \mathsf{crs})$.
  2. Output $\mathsf{IE}'.\mathsf{Enc}(\mathsf{pk}', \mathsf{msg})$.
- $\mathsf{IE.Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct})$*:*
  1. Parse $\mathsf{pk} := (\lambda, S, \mathsf{pk}', \mathsf{crs})$ and $\mathsf{sk} := (\mathsf{sk}', \pi)$, and set $x := (\lambda, S, \mathsf{pk}', \mathsf{sk}')$.
  2. If $\mathsf{NIZK.V}(\mathsf{crs}, x, \pi) = 0$, then output $\perp$, and otherwise output $\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct})$.

Completeness follows directly from the completeness of the underlying incompressible PKE and NIZK schemes. The secret key size is $|\mathsf{sk}'| + |\pi|$. We have (a) $|\mathsf{sk}'| \leq \ell_{\mathsf{sk}} \leq \mathsf{t}$, and (b) $|\pi| = \text{poly}(\lambda, \mathsf{t})$ since the NIZK prover has time that is polynomial in its witness verification algorithm, which is $\mathsf{IE.Gen}'$. Thus, the secret-key size is $\text{poly}(\mathsf{t})$.

In Claim 7.7 we prove that the scheme has incompressible encryption security, and in Claim 7.10, we show that it has key consistency.

Claim 7.7. *The scheme has incompressible encryption security.*

Proof. Let $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2)$ be a PPT adversary for the above scheme. Let $n := |(\lambda, S, \mathsf{pk}', \mathsf{sk}')|$ for $(\mathsf{pk}', \mathsf{sk}') \in \mathsf{Img}(\mathsf{IE.Gen}'(1^\lambda, S))$. Let $\mathsf{IE}^{\mathbf{s}}$ be the same scheme as IE, except that $\mathsf{IE}^{\mathbf{s}}.\mathsf{Gen}$ is implemented using the simulator of the NIZK scheme. Specifically, $\mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda)$ is replaced with $(\mathsf{crs}, \mathsf{aux}) \leftarrow \mathbf{S}_1(1^\lambda, 1^n)$, and $\pi \leftarrow \mathsf{NIZK}.\mathbf{P}(\mathsf{crs}, x, w)$ is replaced with $\pi \leftarrow \mathbf{S}_2(\mathsf{crs}, x, \mathsf{aux})$.

We prove that $(\mathsf{IE.Gen}, \mathsf{IE.Enc}, \mathsf{IE.Dec})$ is incompressible using Claim 7.8 and Claim 7.9. In Claim 7.8 we prove that,

$$\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1\right] - \mathsf{negl}(\lambda) \leq \Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}(\lambda) = 1\right] .$$

In Claim 7.9 we prove that there exists a PPT adversary $\mathbf{A}'$ such that,

$$\Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}} = 1\right] = \Pr\left[\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'} = 1\right] .$$

By incompressibility of IE' we have that $\Pr[\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'}] \leq \frac{1}{2} + \mathsf{negl}(\lambda)$. Therefore,

$$\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1\right] - \mathsf{negl}(\lambda) \leq \Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}(\lambda) = 1\right] = \Pr\left[\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'} = 1\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda) ,$$

and so $\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1\right] \leq \frac{1}{2} + \mathsf{negl}(\lambda)$ as required. All that remains is to prove Claim 7.8 and Claim 7.9.

Claim 7.8. $\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1\right] - \mathsf{negl}(\lambda) \leq \Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}(\lambda) = 1\right] .$

Proof. Let $\tilde{\mathbf{V}} = (\tilde{\mathbf{V}}_1, \tilde{\mathbf{V}}_2)$ be the following cheating verifier for the NIZK scheme:

- $\tilde{\mathbf{V}}_1$ is defined as follows:
  1. Given input crs.
  2. Sample $S \leftarrow \mathbf{A}_1(1^\lambda)$ and $\rho \leftarrow \{0,1\}^{\mathsf{r}(\lambda)}$.
  3. Compute $(\mathsf{pk}', \mathsf{sk}') := \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho)$.
  4. Output (x, w, st) where $x := (\lambda, S, \mathsf{pk}', \mathsf{sk}')$, $w := \rho$, and $\mathsf{st} := \bot$
- $\tilde{\mathbf{V}}_2$ is defined as follows:
  1. Given inputs $\mathsf{crs}, x, \pi, \mathsf{st}$.
  2. Parse $x := (\lambda, S, \mathsf{pk}', \mathsf{sk}')$ and set $\mathsf{pk} := (\lambda, S, \mathsf{pk}', \mathsf{crs})$ and $\mathsf{sk} := (\mathsf{sk}', \pi)$.
  3. Sample $b \leftarrow \{0,1\}$.
  4. Send pk to $\mathbf{A}_1$, and receive $\mathsf{aux}, \mathsf{msg}_0, \mathsf{msg}_1$.
  5. Compute $\mathsf{ct} \leftarrow \mathsf{IE}'.\mathsf{Enc}(\mathsf{pk}', \mathsf{msg}_b)$.
  6. Send ct to $\mathbf{A}_1$, and receive a state st'.
  7. Send $(\mathsf{pk}, \mathsf{sk}, \mathsf{aux}, \mathsf{st}')$ to $\mathbf{A}_2$, and receive b'.
  8. Output b'.

By the security of the scheme, and since $\tilde{\mathbf{V}}$ can be described as a non-uniform poly-size circuit (by fixing the best randomness),

$$\left| \Pr\left[\begin{array}{l} x \in L(R) \\ \wedge\ \tilde{\mathbf{V}}_2(\mathsf{crs}, x, \pi, \mathsf{st}) = 1 \end{array} \middle| \begin{array}{r} \mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda, 1^n) \\ (x, w, \mathsf{st}) \leftarrow \tilde{\mathbf{V}}_1(\mathsf{crs}) \\ \pi \leftarrow \mathsf{NIZK}.\mathbf{P}(\mathsf{crs}, x, w) \end{array} \right] \right.$$
$$\left. - \Pr\left[\begin{array}{l} x \in L(R) \\ \wedge\ \tilde{\mathbf{V}}_2(\mathsf{crs}, x, \pi, \mathsf{st}) = 1 \end{array} \middle| \begin{array}{r} (\mathsf{crs}, \mathsf{aux}) \leftarrow \mathbf{S}_1(1^\lambda, 1^n) \\ (x, w, \mathsf{st}) \leftarrow \tilde{\mathbf{V}}_1(\mathsf{crs}) \\ \pi \leftarrow \mathbf{S}_2(\mathsf{crs}, x, \mathsf{aux}) \end{array} \right] \right| = \mathsf{negl}(\lambda)$$

Note that the left probability equals to $\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1\right]$, and the right probability equals to $\Pr[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}=1]$. Therefore,

$$|\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1\right]-\Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}=1\right]|=\mathsf{negl}(\lambda)\,.$$

And so $\Pr\left[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1\right]-\mathsf{negl}(\lambda)\leq\Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}=1\right]$. □

Claim 7.9. *There is a PPT adversary* $\mathbf{A}'=(\mathbf{A}'_1,\mathbf{A}'_2)$ *such that* $\Pr[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}=1]=\Pr[\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'}=1]$.

Proof. Let $\mathbf{A}'=(\mathbf{A}'_1,\mathbf{A}'_2)$ be the following adversary for the IE' scheme:

- $\mathbf{A}'_1$ is defined as follows:
  1. On input 1^λ:
     (a) Run $\mathbf{A}_1(1^\lambda)$ to receive 1^S.
     (b) Output 1^S.
  2. After receiving pk:
     (a) Let $n:=|(\lambda,S,\mathsf{pk},\mathsf{sk})|$ for $(\mathsf{pk},\mathsf{sk})\in\mathsf{Img}(\mathsf{IE.Gen}(1^\lambda,S))$.
     (b) Compute $(\mathsf{crs},\mathsf{aux}_1)\leftarrow\mathbf{S}_1(1^\lambda,1^n)$.
     (c) Send $(\mathsf{pk},\mathsf{crs})$ to $\mathbf{A}_1$ as the public key.
     (d) Receive $(\mathsf{aux}_2,\mathsf{msg}_0,\mathsf{msg}_1)$ from $\mathbf{A}_1$.
     (e) Submit $\mathsf{aux}:=(\mathsf{crs},\mathsf{aux}_1,\mathsf{aux}_2),\mathsf{msg}_0,\mathsf{msg}_1$.
  3. After receiving ct:
     (a) Send ct to $\mathbf{A}_1$ and receive st of size at most S.
     (b) Output st.
- $\mathbf{A}'_2$, on receiving $(\mathsf{pk},\mathsf{sk},\mathsf{aux},\mathsf{st})$, is defined as follows:
  1. Parse $(\mathsf{crs},\mathsf{aux}_1,\mathsf{aux}_2)$ and set $x:=(\lambda,S,\mathsf{pk},\mathsf{sk})$.
  2. Compute $\pi\leftarrow\mathbf{S}_2(\mathsf{crs},x,\mathsf{aux}_1)$.
  3. Set $\mathsf{pk}':=(\mathsf{pk},\mathsf{crs})$ and $\mathsf{sk}':=(\mathsf{sk},\pi)$.
  4. Run $\mathbf{A}_2$ on $(\lambda,S,\mathsf{pk}',\mathsf{sk}')$ to receive b'.
  5. Output b'.

Observe that the view of the adversary $\mathbf{A}$ in the experiment $\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}$ is identical to the view of the adversary $\mathbf{A}$ in the experiment $\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'}$. Therefore,

$$\Pr\left[\mathsf{Dist}^{\mathsf{IE}^{\mathsf{s}}}_{\mathbf{A}}=1\right]=\Pr\left[\mathsf{Dist}^{\mathsf{IE}'}_{\mathbf{A}'}=1\right]\,,$$

as required. □

□

We now show that our new incompressible PKE has key consistency, finalizing the proof of Lemma 7.5.

Claim 7.10. *The scheme has key consistency.*

Proof. Fix some $\lambda \in \mathbb{N}$, $S = \mathrm{poly}(\lambda)$ and a key consistency adversary $\mathbf{A}$. By construction of the IE scheme:[5]

$$\Pr\left[\begin{array}{l|r} \mathsf{IE.Dec}(\mathsf{pk}, \tilde{\mathsf{sk}}, \mathsf{ct}) \notin \{\mathsf{msg}, \perp\} & \begin{array}{r}(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{IE.Gen}(1^\lambda, S) \\ (\mathsf{msg}, \rho, \tilde{\mathsf{sk}}) \leftarrow \mathbf{A}(\mathsf{pk}) \\ \mathsf{ct} := \mathsf{IE.Enc}(\mathsf{pk}, \mathsf{msg}; \rho)\end{array}\end{array}\right]$$

$$= \Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) = \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1\end{array} & \begin{array}{r}(\mathsf{pk}', \cdot) \leftarrow \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S) \\ \mathsf{crs} \leftarrow \mathsf{NIZK.Gen}(1^\lambda) \\ \mathsf{pk} := (\lambda, S, \mathsf{pk}', \mathsf{crs}) \\ (\mathsf{msg}, \rho, \tilde{\mathsf{sk}}) \leftarrow \mathbf{A}(\mathsf{pk}) \\ \tilde{\mathsf{sk}} := (\mathsf{sk}', \pi) \\ x := (\lambda, S, \mathsf{pk}', \mathsf{sk}') \\ \mathsf{ct} := \mathsf{IE}'.\mathsf{Enc}(\mathsf{pk}', \mathsf{msg}; \rho)\end{array}\end{array}\right].$$

Let Exp be the experiment described in the above second probability expression. By the law of total probability,

$$\Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right] \tag{5}$$

$$= \Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\ \exists \rho',\ (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right]$$

$$+ \Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\ \nexists \rho',\ (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right].$$

We show that the probability in Equation 5 is negligible by proving the following two equations,

$$\Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\ \exists \rho',\ (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right] = 0\,, \tag{6}$$

$$\Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\ \nexists \rho',\ (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right] = \mathsf{negl}(\lambda)\,. \tag{7}$$

- Equation 6: Note that,

$$\Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\ \exists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right]$$

$$\leq \Pr\left[\begin{array}{l|r} \begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\ \exists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} & (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp}\end{array}\right].$$

  By the perfect completeness of the underlying incompressible PKE scheme,

$$\Pr\left[\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) = \mathsf{msg} \,\middle|\, \begin{array}{r}(\mathsf{pk}', \mathsf{sk}') \leftarrow \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S) \\ \mathsf{ct} \leftarrow \mathsf{IE}'.\mathsf{Enc}(\mathsf{pk}', \mathsf{msg})\end{array}\right] = 1\,.$$

[5]Note that here we assume without loss of generality that $\mathsf{IE}'.\mathsf{Dec}$ never outputs $\perp$, which can be ensured by having $\perp$ be a symbol outside of the alphabet of $\mathsf{IE}'.\mathsf{Dec}$.

Therefore, if there exist ρ' and ρ such that $(\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}.\mathsf{Gen}(1^\lambda, S; \rho')$ and $\mathsf{ct} = \mathsf{IE}'.\mathsf{Enc}(\mathsf{pk}', \mathsf{msg}; \rho)$, then $\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) = \mathsf{msg}$. Observe that in the above probability, $\mathsf{pk}', \mathsf{sk}'$, and ct are generated as in Exp. Consequently,

$$\Pr\left[\begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\, \exists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] = 0\,.$$

As a result,

$$\Pr\left[\begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\, \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, \exists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] = 0\,,$$

as required.

- Equation 7: Note that,

$$\begin{aligned}&\Pr\left[\begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\, \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, \nexists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] \\ &\leq \Pr\left[\begin{array}{l}\mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, \nexists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right].\end{aligned}$$

By the definition of R we get that,

$$\begin{aligned}&\Pr\left[\begin{array}{l}\mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, \nexists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] \\ &= \Pr\left[\begin{array}{l}\mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, x \notin L(R)\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right].\end{aligned}$$

By the definition of Exp, we get that,

$$\begin{aligned}&\Pr\left[\begin{array}{l}\mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, x \notin L(R)\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] \\ &\leq \Pr\left[\begin{array}{l}\exists (x, \pi) \text{ s.t.} \\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, x \notin L(R)\end{array} \middle| \; \mathsf{crs} \leftarrow \mathsf{NIZK}.\mathsf{Gen}(1^\lambda, 1^n) \right].\end{aligned}$$

By the statistical soundness of the NIZK scheme,

$$\Pr\left[\begin{array}{l}\exists (x, \pi) \text{ s.t.} \\ \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, x \notin L(R)\end{array} \middle| \; \mathsf{crs} \leftarrow \mathsf{NIZK}.\mathsf{Gen}(1^\lambda, 1^n) \right] = \mathsf{negl}(\lambda)\,.$$

Overall, we get that,

$$\Pr\left[\begin{array}{l}\mathsf{IE}'.\mathsf{Dec}(\mathsf{pk}', \mathsf{sk}', \mathsf{ct}) \neq \mathsf{msg} \\ \wedge\, \mathsf{NIZK}.\mathbf{V}(\mathsf{crs}, x, \pi) = 1 \\ \wedge\, \nexists \rho' \text{ s.t } (\mathsf{pk}', \mathsf{sk}') = \mathsf{IE}'.\mathsf{Gen}(1^\lambda, S; \rho')\end{array} \middle| \; (\mathsf{pk}', \mathsf{sk}', \mathsf{ct}, \mathsf{crs}, x, \pi) \leftarrow \mathsf{Exp} \right] = \mathsf{negl}(\lambda)\,,$$

as required.

□

□

7.2 Incompressible encryption and witness retrieval are incompatible

In this section we show that the existence incompressible encryption with statistical key consistency implies the inexistence of CIC with witness retrieval:

Lemma 7.11. *If there exists incompressible PKE with key consistency and secret key of size* $\mathrm{poly}(\lambda, \log S)$*, then there is no computational instance compression for* NP *with witness retrieval and compression parameter* $k(\lambda, n, m) \leq \mathrm{poly}(\lambda, \log n, m)$*.*

Note that the impossibility works even for the weaker notion of CIC with non-adaptive soundness.

Proof. Let $(\mathsf{IE.Gen}, \mathsf{IE.Enc}, \mathsf{IE.Dec})$ be the incompressible PKE scheme with key consistency, and let $\ell_{\mathsf{sk}}(\lambda, S) = \mathrm{poly}(\lambda, \log S)$. We prove the theorem by defining a specific NP language, and use the CIC for that language to attack the incompressible PKE. Let R be the following NP relation,

$$R := \left\{ ((\lambda, S, \mathsf{pk}, \mathsf{ct}), \mathsf{sk}) \;\middle|\; \mathsf{IE.Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct}) = 0 \right\} .$$

Assume towards contradiction (to the incompressible encryption security of the PKE scheme) that there exists a CIC scheme with witness retrieval $(\mathsf{Gen}, \mathsf{IC}, \mathsf{WR})$ for the source NP relation R and target relation $R' \in \mathrm{NP}$ with compression parameter $k(\lambda, n, m) \leq (\lambda + \log n + m)^c$ for a constant c.

We construct an adversary to the incompressible encryption security of the PKE scheme. Let $S(\lambda) = \mathrm{poly}(\lambda)$ be a large enough polynomial such that for every large enough λ it holds that $S(\lambda) \geq (2\lambda + \ell_{\mathsf{sk}}(\lambda, S(\lambda)))$. Note that there exists such a polynomial since $\ell_{\mathsf{sk}}(\lambda, S) = \mathrm{poly}(\lambda, \log S)$. Let $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2)$ be the following adversary,

- $\mathbf{A}_1$ is defined is follows:
  1. On input 1^λ, outputs a space bound 1^S where $S := S(\lambda)$.
  2. On receiving pk,
     (a) Sample $\mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)$, where $m := \ell_{\mathsf{sk}}(\lambda)$ and $n := |(\lambda, S, \mathsf{pk}, \mathsf{ct})|$ for ct that is a one bit encryption.
     (b) Output $(\mathsf{aux} = \mathsf{crs}, \mathsf{msg}_0 = 0, \mathsf{msg}_1 = 1)$.
  3. On receiving ct,
     (a) Set $x := (\lambda, S, \mathsf{pk}, \mathsf{ct})$.
     (b) Compute $x' := \mathsf{IC}(\mathsf{crs}, x)$.
     (c) If $|x'| \leq S$, then $\mathsf{st} := x'$. Otherwise, $\mathsf{st} := \bot$.
     (d) Output st.
- $\mathbf{A}_2$ on receiving $(\mathsf{pk}, \mathsf{sk}, \mathsf{aux}, \mathsf{st})$,
  1. Parse $\mathsf{aux} := \mathsf{crs}$, $\mathsf{st} := x'$, and $\mathsf{sk} := w$.
  2. Compute $w' \leftarrow \mathsf{WR}(\mathsf{crs}, x', w)$.
  3. If $(x', w') \in R'$, then output $b' = 0$.
  4. Otherwise, output $b' \leftarrow \{0, 1\}$.

We show that for large enough λ,

$$\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1] \geq \frac{1}{2} + \frac{1}{4} \cdot (1 - \mathsf{negl}(\lambda)) ,$$

which breaks incompressible encryption security of the PKE scheme.

We start by proving that for large enough λ, the adversary $\mathbf{A}_1$ always outputs $\mathsf{st} = x'$ (as opposed to $\perp$), and then we show that, given $\mathsf{st} = x'$, $\mathbf{A}_2$ guesses the correct $b' = b$ with probability $\frac{1}{2} + \frac{1}{4} \cdot (1 - \mathsf{negl}(\lambda))$.

Claim 7.12. *$\mathbf{A}_1$ outputs $\mathsf{st} = x'$ for large enough λ.*

Proof. By the construction, if $|x'| \leq S$, then $\mathbf{A}_1$ outputs x', and so we show that this event happens for large enough λ. By the CIC compression parameter,

$$\begin{aligned} |x'| &= |\mathsf{IC}(\mathsf{crs}, x)| \\ &\leq k(\lambda, n, m) \\ &\leq (\lambda + \log n + m)^c \\ &= (\lambda + \log n + \ell_{\mathsf{sk}}(\lambda, S))^c \,. \end{aligned} \tag{8}$$

where the last equality holds since $m = \ell_{\mathsf{sk}}(\lambda, S)$. By the efficiency parameters of the PKE scheme, $|\mathsf{pk}|, |\mathsf{ct}| \leq \mathrm{poly}(\lambda, S)$. Since $S = \mathrm{poly}(\lambda)$ (so $|S| = O(\log \lambda)$), it holds that

$$n = |(\lambda, S, \mathsf{pk}, \mathsf{ct})| \leq \mathrm{poly}(\lambda, S) = \mathrm{poly}(\lambda) \,.$$

Therefore, there exists some constant c' such that,

$$(\lambda + \log n + \ell_{\mathsf{sk}}(\lambda, S))^c \leq (\lambda + c' \cdot \log \lambda + \ell_{\mathsf{sk}}(\lambda, S))^c \leq (2\lambda + \ell_{\mathsf{sk}}(\lambda, S))^c \leq S \,,$$

where by the definition of S, the rightmost inequality holds for large enough λ. By combining the above with Equation 8, we get that for large enough λ it holds that $|x'| \leq S$.

□

We henceforth assume that λ is large enough so that Claim 7.12 holds, so that we can always assume that some x' is output. Let E be the event that $(x', w') \in R'$. By the law of total probability, and since b is sampled uniformly at random,

$$\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1] \geq \frac{1}{2} \cdot \Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1 \mid b = 0] + \frac{1}{2} \cdot \Pr[\bar{\mathsf{E}} \mid b = 1] \cdot \Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1 \mid b = 1 \wedge \bar{\mathsf{E}}] \,.$$

We bound each of the above expressions:

Claim 7.13. *The following hold:*

1. $\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1 \mid b = 0] = 1$.
2. $\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1 \mid b = 1 \wedge \bar{\mathsf{E}}] = \frac{1}{2}$.
3. $\Pr[\bar{\mathsf{E}} \mid b = 1] \geq 1 - \mathsf{negl}(\lambda)$.

Proof. We prove each one individually:

1. Fix $(\mathsf{pk}, \mathsf{sk})$ and ct that was generated by $\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)$, conditioned on $b = 0$. Note that $b = 0$ implies that ct is an honestly generated encryption of 0. Therefore, by the perfect correctness of the PKE scheme, the decryption algorithm decrypts to 0: $\mathsf{IE.Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct}) = 0$. Consequently, $((\lambda, S, \mathsf{pk}, \mathsf{ct}), \mathsf{sk}) \in R$. Moreover, since $x' = \mathsf{IC}(\mathsf{crs}, (\lambda, S, \mathsf{pk}, \mathsf{ct}))$, we have $(x', \mathsf{WR}(x', \mathsf{sk})) \in R'$, and so $\mathbf{A}_2$ outputs $b' = 0 = b$. Therefore, the $\mathbf{A}_2$ outputs $b' = b$ with probability 1, as required.

2. By the definition of the event $\bar{\mathsf{E}}$, for every (x', w') generated by $\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)$, it holds that $(x', w') \notin R'$. In this case, the adversary simply outputs uniformly random bit b'. Therefore,

$$\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda) = 1 \mid b = 1 \wedge \bar{\mathsf{E}}] = \frac{1}{2}\,.$$

3. Let $\mathbf{A}'$ be the following adversary to the CIC scheme,

   (a) Given as input a crs.
   (b) Sample $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{IE.Gen}(1^\lambda, S)$ and $\mathsf{ct} \leftarrow \mathsf{IE.Enc}(\mathsf{pk}, 1)$, and set $x := (\lambda, S, \mathsf{pk}, \mathsf{ct})$ and $w := \mathsf{sk}$.
   (c) Output (x, w') where $w' := \mathsf{WR}(\mathsf{crs}, x', w)$ for $x' := \mathsf{IC}(\mathsf{crs}, x)$.

   Note that the output of the adversary $\mathbf{A}'$ is distributed identically to (x, w') generated in $\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)$ when conditioned on $b = 1$. Therefore,

$$\Pr\left[\ (\mathsf{IC}(\mathsf{crs}, x), w') \in R' \ \middle|\ \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)\\ (x, w') \leftarrow \mathbf{A}'(\mathsf{crs})\end{array}\right] = \Pr\left[\mathsf{E} \,\middle|\, b = 1\ \right]\,. \tag{9}$$

   By the definition of $\mathbf{A}'$, it is always the case that $x \in \{0,1\}^n$. By rephrasing key consistency of the PKE scheme for $\mathsf{msg} = 1$, for every λ and S:

$$\Pr\left[\ \begin{array}{l}\forall \rho, \tilde{\mathsf{sk}},\\ \mathsf{IE.Dec}(\mathsf{pk}, \tilde{\mathsf{sk}}, \mathsf{IE.Enc}(\mathsf{pk}, 1; \rho)) \in \{1, \bot\}\end{array} \ \middle|\ (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, S)\ \right] = 1 - \mathsf{negl}(\lambda)\,.$$

   Recalling that

$$R := \left\{((\lambda, S, \mathsf{pk}, \mathsf{ct}), \mathsf{sk}) \,\middle|\, \mathsf{IE.Dec}(\mathsf{pk}, \mathsf{sk}, \mathsf{ct}) = 0\ \right\}\,,$$

   we conclude that for any λ and S,

$$\Pr\left[\ \begin{array}{l}\forall \rho,\\ (\lambda, S, \mathsf{pk}, , \mathsf{IE.Enc}(\mathsf{pk}, 1; \rho)) \notin L(R)\end{array} \ \middle|\ (\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda, S)\ \right] = 1 - \mathsf{negl}(\lambda)\,.$$

   Observe that $\mathbf{A}'$ generates $x := (\lambda, S, \mathsf{pk}, \mathsf{ct})$ as in the above probability albeit with uniform ρ, and so we have:

$$\Pr\left[\ x \notin L(R)\ \middle|\ \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)\\ (x, w') \leftarrow \mathbf{A}'(\mathsf{crs})\end{array}\right] \geq 1 - \mathsf{negl}(\lambda)\,. \tag{10}$$

   Combining Equations 9 and 10, we have

$$\Pr\left[\ \begin{array}{l}x \in \{0,1\}^n\\ \wedge\ x \notin L(R)\\ \wedge\ (\mathsf{IC}(\mathsf{crs}, x), w') \in R'\end{array} \ \middle|\ \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)\\ (x, w') \leftarrow \mathbf{A}'(\mathsf{crs})\end{array}\right] \geq (1 - \mathsf{negl}(\lambda)) \cdot \Pr\left[\mathsf{E} \,\middle|\, b = 1\ \right] \tag{11}$$

   By adaptive soundness of the CIC scheme,

$$\Pr\left[\ \begin{array}{l}x \in \{0,1\}^n\\ \wedge\ x \notin L(R)\\ \wedge\ (\mathsf{IC}(\mathsf{crs}, x), w') \in R'\end{array} \ \middle|\ \begin{array}{r}\mathsf{crs} \leftarrow \mathsf{Gen}(1^\lambda, 1^n, 1^m)\\ (x, w') \leftarrow \mathbf{A}'(\mathsf{crs})\end{array}\right] \leq \mathsf{s}(\lambda) = \mathsf{negl}(\lambda)\,. \tag{12}$$

   Overall, By Equations 11 and 12 we get that,

$$(1 - \mathsf{negl}(\lambda)) \cdot \Pr\left[\mathsf{E} \,\middle|\, b = 1\ \right] \leq \mathsf{negl}(\lambda)\,.$$

   Therefore, $\Pr[\mathsf{E} \mid b = 1] \leq \mathsf{negl}(\lambda)$ or, equivalently, $\Pr\left[\bar{\mathsf{E}} \mid b = 1\right] \geq 1 - \mathsf{negl}(\lambda)$.

□

Given Claim 7.13, we finalize the proof:

$$\begin{aligned}\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1] &\geq \frac{1}{2}\cdot\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1 \mid b=0]+\frac{1}{2}\cdot\Pr[\bar{\mathsf{E}} \mid b=1]\cdot\Pr[\mathsf{Dist}^{\mathsf{IE}}_{\mathbf{A}}(\lambda)=1 \mid b=1\wedge\bar{\mathsf{E}}]\\ &\geq \frac{1}{2}\cdot 1+\frac{1}{2}\cdot\frac{1}{2}\cdot(1-\mathsf{negl}(\lambda))\\ &= \frac{1}{2}+\frac{1}{4}\cdot(1-\mathsf{negl}(\lambda))\,,\end{aligned}$$

as required. □

Acknowledgments

We thank Nico Döttling for suggesting the connection to incompressible cryptography.

Gal Arnon is supported in part by a grant from the Israel Science Foundation (Grant No. 2686/20), by the Simons Foundation Collaboration on the Theory of Algorithmic Fairness, and by the Israeli Council for Higher Education (CHE) via the Weizmann Data Science Research Center. Shany Ben-David is supported by the Israel Science Foundation (Grant no. 2302/22). Eylon Yogev is supported by the Israel Science Foundation (Grant No. 2302/22), European Research Union (ERC, CRYPTOPROOF, 101164375), and by an Alon Young Faculty Fellowship. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them.

References

[AS16] Gilad Asharov and Gil Segev. "Limits on the Power of Indistinguishability Obfuscation and Functional Encryption". In: *SIAM J. Comput.* 45.6 (2016), pp. 2117–2176.

[BDFH09] Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin. "On problems without polynomial kernels". In: *Journal of Computer and System Sciences* 75.8 (2009), pp. 423–434.

[BR22] Liron Bronfman and Ron D. Rothblum. "PCPs and Instance Compression from a Cryptographic Lens". In: *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January 31 - February 3, 2022, Berkeley, CA, USA*. Ed. by Mark Braverman. Vol. 215. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022, 30:1–30:19.

[BT24] Jan Buzek and Stefano Tessaro. "Collision Resistance from Multi-collision Resistance for All Constant Parameters". In: *Advances in Cryptology - CRYPTO 2024 - 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2024, Proceedings, Part V*. Ed. by Leonid Reyzin and Douglas Stebila. Vol. 14924. Lecture Notes in Computer Science. Springer, 2024, pp. 429–458.

[Ben24] Shany Ben-David. "Probabilistically Checkable Arguments for All NP". In: *Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part III*. Ed. by Marc Joye and Gregor Leander. Vol. 14653. Lecture Notes in Computer Science. Springer, 2024, pp. 345–374.

[DI06] Bella Dubrov and Yuval Ishai. "On the randomness complexity of efficient sampling". In: *Proceedings of the 38th Annual ACM Symposium on Theory of Computing*. ACM, 2006, pp. 711–720.

[Dam87] Ivan Damgård. "Collision Free Hash Functions and Public Key Signature Schemes". In: *Advances in Cryptology - EUROCRYPT '87, Workshop on the Theory and Application of of Cryptographic Techniques, Amsterdam, The Netherlands, April 13-15, 1987, Proceedings*. Ed. by David Chaum and Wyn L. Price. Vol. 304. Lecture Notes in Computer Science. Springer, 1987, pp. 203–216.

[Dru15] Andrew Drucker. "New Limits to Classical and Quantum Instance Compression". In: *SIAM J. Comput.* 44.5 (2015), pp. 1443–1479.

[Dzi06] Stefan Dziembowski. "On Forward-Secure Storage". In: *Advances in Cryptology - CRYPTO 2006, 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006, Proceedings*. Ed. by Cynthia Dwork. Vol. 4117. Lecture Notes in Computer Science. Springer, 2006, pp. 251–270.

[FS11] Lance Fortnow and Rahul Santhanam. "Infeasibility of instance compression and succinct PCPs for NP". In: *J. Comput. Syst. Sci.* 77.1 (2011), pp. 91–106.

[GGH11] Oded Goldreich, Shafi Goldwasser, and Shai Halevi. "Collision-Free Hashing from Lattice Problems". In: *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation - In Collaboration with Lidor Avigad, Mihir Bellare, Zvika Brakerski, Shafi Goldwasser, Shai Halevi, Tali Kaufman, Leonid Levin, Noam Nisan, Dana Ron, Madhu Sudan, Luca Trevisan, Salil Vadhan, Avi Wigderson, David Zuckerman*. Ed. by Oded Goldreich. Vol. 6650. Lecture Notes in Computer Science. Springer, 2011, pp. 30–39.

[GLFFLMS10] David Galindo, Benoît Libert, Marc Fischlin, Georg Fuchsbauer, Anja Lehmann, Mark Manulis, and Dominique Schröder. "Public-Key Encryption with Non-Interactive Opening: New Constructions and Stronger Definitions". In: *Progress in Cryptology - AFRICACRYPT 2010, Third International Conference on Cryptology in Africa, Stellenbosch, South Africa, May 3-6, 2010. Proceedings*. Ed. by Daniel J. Bernstein and Tanja Lange. Vol. 6055. Lecture Notes in Computer Science. Springer, 2010, pp. 333–350.

[GWZ22] Jiaxin Guan, Daniel Wichs, and Mark Zhandry. "Incompressible Cryptography". In: *Advances in Cryptology - EUROCRYPT 2022 - 41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, May 30 - June 3, 2022, Proceedings, Part I*. EUROCRYPT '22. 2022, pp. 700–730.

[HN10] Danny Harnik and Moni Naor. "On the Compressibility of *NP* Instances and Cryptographic Applications". In: *SIAM J. Comput.* 39.5 (2010), pp. 1667–1713.

[IKO05] Yuval Ishai, Eyal Kushilevitz, and Rafail Ostrovsky. "Sufficient Conditions for Collision-Resistant Hashing". In: *Proceedings of the 2nd Theory of Cryptography Conference*. TCC '05. 2005, pp. 445–456.

[Imp95] Russell Impagliazzo. "A Personal View of Average-Case Complexity". In: *Proceedings of the Tenth Annual Structure in Complexity Theory Conference, Minneapolis, Minnesota, USA, June 19-22, 1995*. IEEE Computer Society, 1995, pp. 134–147.

[KNY17] Ilan Komargodski, Moni Naor, and Eylon Yogev. "White-Box vs. Black-Box Complexity of Search Problems: Ramsey and Graph Property Testing". In: *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS*. 2017, pp. 622–632.

[KNY18] Ilan Komargodski, Moni Naor, and Eylon Yogev. "Collision Resistant Hashing for Paranoids: Dealing with Multiple Collisions". In: *Advances in Cryptology - EUROCRYPT 2018*. 2018, pp. 162–194.

[KY18] Ilan Komargodski and Eylon Yogev. "On Distributional Collision Resistant Hashing". In: *Advances in Cryptology - CRYPTO 2018 - 38th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2018, Proceedings, Part II*. Ed. by Hovav Shacham and Alexandra Boldyreva. Vol. 10992. Lecture Notes in Computer Science. Springer, 2018, pp. 303–327.

[Kil92] Joe Kilian. "A note on efficient zero-knowledge proofs and arguments". In: *Proceedings of the 24th Annual ACM Symposium on Theory of Computing*. STOC '92. 1992, pp. 723–732.

[Kil95] Joe Kilian. "Improved Efficient Arguments". In: *Proceedings of the 15th Annual International Cryptology Conference*. CRYPTO '95. 1995, pp. 311–324.

[Nao91] Moni Naor. "Bit Commitment Using Pseudorandomness". In: *Journal of Cryptology* 4.2 (1991). Preliminary version appeared in CRYPTO '89., pp. 151–158.

[PS19] Chris Peikert and Sina Shiehian. "Noninteractive Zero Knowledge for NP from (Plain) Learning with Errors". In: *Advances in Cryptology - CRYPTO 2019 - 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2019, Proceedings, Part I*. CRYPTO '19. 2019, pp. 89–114.

[RV24] Ron D. Rothblum and Prashant Nalini Vasudevan. "Collision Resistance from Multi-collision Resistance". In: *J. Cryptol.* 37.2 (2024), p. 14.

[SG02] Victor Shoup and Rosario Gennaro. "Securing Threshold Cryptosystems against Chosen Ciphertext Attack". In: *J. Cryptol.* 15.2 (2002), pp. 75–96.

[Sim98] Daniel R. Simon. "Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions?" In: *EUROCRYPT*. Vol. 1403. 1998, pp. 334–345.

[WW24a] Brent Waters and David J. Wu. "A Pure Indistinguishability Obfuscation Approach to Adaptively-Sound SNARGs for NP". In: *IACR Cryptol. ePrint Arch.* (2024), p. 933. URL: https://eprint.iacr.org/2024/933.

[WW24b] Brent Waters and David J. Wu. "Adaptively-Sound Succinct Arguments for NP from Indistinguishability Obfuscation". In: *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*. Ed. by Bojan Mohar, Igor Shinkar, and Ryan O'Donnell. ACM, 2024, pp. 387–398.

[WZ24] Brent Waters and Mark Zhandry. "Adaptive Security in SNARGs via iO and Lossy Functions". In: *Advances in Cryptology - CRYPTO 2024 - 44th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2024, Proceedings, Part X*. Ed. by Leonid Reyzin and Douglas Stebila. Vol. 14929. Lecture Notes in Computer Science. 2024, pp. 72–104.

[YZWGL19] Yu Yu, Jiang Zhang, Jian Weng, Chun Guo, and Xiangxue Li. "Collision Resistant Hashing from Sub-exponential Learning Parity with Noise". In: *Advances in Cryptology - ASIACRYPT 2019 - 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8-12, 2019, Proceedings, Part II*. Ed. by Steven D. Galbraith and Shiho Moriai. Vol. 11922. Lecture Notes in Computer Science. Springer, 2019, pp. 3–24.