Impossibilities in Succinct Arguments: Black-box Extraction and More

Matteo Campanelli[1], Chaya Ganesh[2], Hamidreza Khoshakhlagh[3], and Janno Siim[4]

[1] Protocol Labs, matteo@protocol.ai
[2] Indian Institute of Science, India, chaya@iisc.ac.in
[3] Concordium, hk@concordium.com
[4] Simula UiB, Bergen, Norway, janno@simula.no

Abstract. The celebrated result by Gentry and Wichs established a theoretical barrier for succinct non-interactive arguments (SNARGs), showing that for (expressive enough) hard-on-average languages, we must assume non-falsifiable assumptions. We further investigate those barriers by showing new negative and positive results related to the proof size.

1. We start by formalizing a folklore lower bound for the proof size of black-box extractable arguments based on the hardness of the language. This separates knowledge-sound SNARGs (SNARKs) in the random oracle model (that can have black-box extraction) and those in the standard model.
2. We find a positive result in the non-adaptive setting. Under the existence of non-adaptively sound SNARGs (without extractability) and from standard assumptions, it is possible to build SNARKs with black-box extractability for a non-trivial subset of **NP**.
3. On the other hand, we show that (under some mild assumptions) all **NP** languages cannot have SNARKs with black-box extractability even in the non-adaptive setting.
4. The Gentry-Wichs result does not account for the preprocessing model, under which fall several efficient constructions. We show that also, in the preprocessing model, it is impossible to construct SNARGs that rely on falsifiable assumptions in a black-box way.

Along the way, we identify a class of non-trivial languages, which we dub "trapdoor languages", that can bypass these impossibility results.

1 Introduction

Proof systems have been studied extensively both in cryptography and in the theory of computation [8, 28, 34], and are a fundamental building block in various cryptographic constructions today, including delegating computation [10, 11, 23] and privacy-preserving cryptocurrencies [9] to name a few. In a *succinct* proof, it is additionally required that the communication be sublinear (ideally polylogarithmic) in the size of the non-deterministic witness used to verify the relation (*proof* succinctness). This requirement is often extended to verification complexity (*verification* succinctness).

Statistically-sound proofs are unlikely to allow for significant improvements in proof size [33, 35, 61], that is, for **NP**, statistical soundness requires the prover to communicate, roughly, as much information as the size of the witness. If we restrict ourselves to *argument systems* [14] where soundness is *computational*, then proofs can potentially be shorter than the length of the witness.

Succinct arguments. Succinct arguments were first studied by Kilian [46], who gave an interactive construction based on probabilistically checkable proofs (PCP) and collision-resistant hash functions. Kilian's construction was turned into a non-interactive argument in the random oracle model using the Fiat-Shamir heuristic [27] by Micali [53]. In the standard model (i.e., without idealized primitives), non-interactivity is achieved by a trusted party generating a Common Reference String (CRS) during a setup phase. The notion of *adaptive soundness* requires soundness to hold even when a malicious prover can choose the statement after receiving CRS. Otherwise, we call soundness non-adaptive.

In this work, we are concerned with the theoretical limitations for building efficient, succinct non-interactive arguments in the standard model[5]. One of the best-known impossibility results on SNARGs is that of Gentry and Wichs [32] (we occasionally refer to it as "GW"), which shows that in the standard model, adaptively-sound SNARGs for (hard enough) **NP** languages cannot be proven secure via a black-box (BB) reduction to a falsifiable assumption [54]. A falsifiable assumption is an assumption where the challenger can efficiently confirm that the challenge was broken[6].

A folklore way to interpret GW has been "*we cannot escape non-falsifiable assumptions to build SNARGs for* **NP**". While this is essentially true, there are several caveats to this interpretation (which we discuss later in this work in section 6 and some have already been noticed in prior work). We formally explore the boundaries of this simplifying interpretation, especially motivated by the focus on (composable) extractability [6, 48] and the popular model of "preprocessing SNARGs" in recent works, e.g. [39]. We strive to provide a *modern* view of these topics, for example, by adopting the language of indexed relations from [19] (we later argue why this is a meaningful switch).

(Black-box) knowledge soundness. A strengthening of the soundness property is *knowledge soundness*. It requires that, whenever an efficient prover convinces the verifier, not only can we conclude that $\mathsf{x} \in \mathcal{L}$, but also an **NP** witness w can be efficiently extracted from the prover. This helpful property is satisfied by many proof systems and is necessary for a lot of applications of succinct arguments. A Succinct Non-interactive ARgument of Knowledge (SNARK) is a SNARG with the knowledge soundness property.

Constructions of SNARKs for **NP** in the standard model all rely on non-falsifiable assumptions that are *knowledge-type* assumptions related to some algebraic problem, e.g., guaranteeing the existence of an extractor algorithm that can output a discrete log "from" a specific adversary. This example also hints to why these assumptions are non-falsifiable—they are *non-black-box*, that is, they require knowledge of the internal state of the adversary and an extractor aware of the concrete adversarial algorithm. This contrasts with the milder *black-box* extraction, namely the ability to extract a witness from a malicious prover only using its input/output interface.

Understanding whether we can build SNARKs with black-box extraction in the standard model is still an elusive problem. In addition to being a theoretical curiosity, if answered positively, it would allow us to construct more robust cryptographic protocols using SNARKs. Black-box extraction is required in strong notions of composition security, e.g., in universal composability (or UC-security [17]) where the "ideal-world" simulator must extract a witness without knowledge of the environment's algorithm. (See [45] for an attempt to combine composability and knowledge-type assumptions.) If answered negatively, it would confirm the seeming incompatibility of SNARKs in the standard model and UC. In this work, we then ask the question:

Is non-black-box extraction inherent to SNARKs?

Addressing this question is, we believe, even more pressing because prior works [5, 6, 18, 48] have used as motivation the fact that succinctness *must* be sacrificed for black-box extraction, implying that the question had been settled (see also section 1.2). However, to the best of our knowledge, there was no formal treatment for this question prior to our work.

Our First Contribution: We formally confirm the folklore belief that black-box extraction is impossible for adaptive knowledge soundness in the standard model if one requires proof-succinctness. As a consequence, this result separates the standard model and other idealized models in terms of what is possible for black-box extraction (for example, in the ROM and through the Fiat-Shamir transform, there exist black-box extractable proof-succinct non-interactive arguments [15]).

Our Second Contribution: We explore whether the impossibility extends to the non-adaptive case. We find out that non-adaptive black-box extractability is possible for a non-trivial subset of **NP**—which

[5] There exist efficient SNARKs (SNARGs of knowledge) in idealized models like ROM (random oracle model), GGM (generic group model), or AGM (algebraic group model), including constructions like Groth16 [39], Bulletproofs [15]. We later discuss the implications of our results for different models.

[6] For example, DLOG is a falsifiable assumption since the challenger can efficiently test if the adversary has found the correct discrete logarithm.

encompasses distributionally hard problems such as knowledge of a discrete logarithm — by assuming the existence of a non-adaptively sound SNARG and some standard assumptions (FHE). In particular, we show that a SNARG can be lifted to a SNARK with the features above for the class of languages **FewP** (roughly, **NP** statements with at most a polynomial number of valid witnesses). If the starting SNARG (non-adaptively sound) is based on falsifiable assumptions and in the standard model then so is the resulting SNARK. [7] Our transformation also preserves zero-knowledge of the initial SNARG.

Our Third Contribution: A natural question is whether the previous construction for **FewP** can be extended to **NP**. We answer this in the negative under some mild assumptions. In particular, we show that if the relation is $y = f(w)$ where f is an L-continuous leakage-resilient one-way function (CLR-OWF, a one-way function where L bits may leak multiple times given that preimage w is updated), then the proof size must be more than L bits. There exists a CLR-OWF under the discrete logarithm assumption [1] where L is linear in the size of w. Thus, the proof cannot be succinct.

Preprocessing and the Gentry-Wichs impossibility result. In many applications we want to look beyond proof-succinctness and keep the verifier as efficient as possible. Ideally, we would like verification to run sublinearly in the size/time of the computation. It may seem counterintuitive that this is even possible: naturally, in circuit-based[8] arguments for general computations the verifier should *at least* read the statement being proven. The latter includes both the description of the computation (i.e., the circuit) and its input (i.e., the deterministic input for an **NP** statement). There exists, however, a (commonly used) way around this problem: a *preprocessing* phase. In a preprocessing SNARG, one generates a CRS, usually depending on a specific circuit C, which is constructed once and for all and can later be used to prove/verify an unbounded number of proofs for the computation of C. This CRS is structured as a *pair* of CRS'es, the prover's CRS and the verifier's CRS, used by each respective party. The verifier's CRS is morally a digest of the circuit. If the verifier's CRS is "short enough", then the online verification stage can be fast, requiring to read only the SNARG proof and a *partial* input description (the deterministic input to the circuit, without its description); thus the verifier can run in time sublinear in $|C|$ (and in the witness size).

This preprocessing model encompasses a rich line of efficient SNARGs [13, 31, 38, 50, 51, 55]. The fact that it is a practically interesting model, as it achieves verifier-succinct SNARGs, further motivates a deeper theoretical understanding of it. A fundamental question is:

Can we construct preprocessing SNARGs based on falsifiable assumptions?

We argue this question has not been settled. First, none of the known preprocessing constructions rely on falsifiable assumptions. Also, known impossibility results do not inform us on the matter either. The Gentry-Wichs impossibility—which separates SNARGs and falsifiable assumptions—has long served as a justification to SNARGs for **NP** on non-falsifiable assumptions, *but it fails to shed light on the preprocessing setting.* The reason is that the GW result presumes a SNARG with a CRS with a specific pattern (we mean "prover's CRS" when we just say CRS from now on): their CRS cannot grow with the size of the instance, but should instead be bounded by a polynomial in the security parameter. In principle the question is then still open, more so because all existing preprocessing constructions, do have a CRS with the opposite pattern: it is usually as long as the instance[9].

Besides GW, other existing works also fail to provide an answer. For example, the work of [12] shows how to "bootstrap" a preprocessing SNARK into one without preprocessing to obtain a *complexity-preserving* SNARK, i.e., one without expensive preprocessing. The transformation can be applied to known SNARKs with expensive preprocessing to obtain a SNARK without the costly preprocessing. This complexity-preserving

[7] It is still an open problem how to obtain non-adaptively secure SNARGs from polynomial-time secure falsifiable assumptions. The only candidate, the recent construction in [52], was shown to have a fundamentally flawed proof of security (see [60]).

[8] There are other models of computation that have a succinct description, for instance, machine computations. However, in general, the description of a computation could be as large as the computation itself.

[9] For example, in pairing-based constructions such as [31] it consists of at least one group element per wire in the circuit to be proven.

compilation, informally, establishes that preprocessing does not give any additional power; if preprocessing SNARKs were possible from falsifiable assumptions, one could apply the bootstrapping transformation and obtain short CRS SNARKs from falsifiable assumptions. Thus, any impossibility for SNARKs holds even for SNARKs that rely on expensive preprocessing. However, this bootstrapping crucially requires the *knowledge soundness* property and, therefore, only applies to SNARKs. The question of whether allowing a preprocessing phase allows constructing *SNARGs* based on falsifiable assumptions remains.

Our Fourth Contribution: We fill the gap left by the GW result and show that even preprocessing SNARGs with a loosely-bounded CRS cannot be constructed from falsifiable assumptions in the standard model.

The landscape of impossibilities for non-interactive arguments. For our work to be as self-contained as possible, we complement the results above with an overarching view of impossibilities on non-interactive arguments (section 6). This discussion strives to give a complete picture of existing impossibility results, related key properties of positive results, and gaps between positive and negative results. Motivated by the observation that preprocessing SNARGs do not come under the GW impossibility, we articulate the assumptions behind the impossibilities, and identify settings that would bypass them. Along the way, we formalize a class of languages that does not come under the Gentry-Wichs impossibility result. We dub them *trapdoor languages* (where there exists a "trapdoor" that makes the problem feasible) and exemplify several application settings that fall under the same category. Trapdoor languages can be thought of as a generalization of witness-sampleable (algebraic) languages in the work of [24].

1.1 Technical Overview

BB extraction is impossible for any hard language (adaptive case). We show the impossibility of black-box extraction for non-interactive succinct arguments following the intuition that if an argument is too "small", it cannot contain information about a "long" witness. This makes extraction impossible since the extractor does not have any additional power, like access to the prover's randomness (as in non-black-box extractors for popular SNARKs) or the ability to rewind the prover (as in interactive arguments, such as Kilian's protocol).

Our result gives a precise characterization between the hardness of guessing the witness and the size of the proof. We show that if an efficient adversary can guess the witness at most with probability $\varepsilon(\lambda)$ and the knowledge soundness error of the argument system is $\varepsilon_{ks}(\lambda)$, then the proof size is at least $-\log(\varepsilon(\lambda)+\varepsilon_{ks}(\lambda))$ bits. For example, if we consider for simplicity that $\varepsilon_{ks}(\lambda) = 0$ and $\varepsilon(\lambda) = 1/2^{\delta|\mathsf{w}(\lambda)|}$ for some $\delta > 0$ and the witness size is $|\mathsf{w}(\lambda)|$, then the proof size will be at least $\delta|\mathsf{w}(\lambda)|$. In appendix B, we show how to obtain a similar result based on the hardness of leakage-resilient OWFs.

BB extraction is possible for FewP (non-adaptive case). We then ask if the impossibility holds if we weaken the knowledge soundness requirement to be *non-adaptive*. Indeed, the non-adaptive case escapes the GW impossibility for SNARGs as we discuss in Section 6.1, and it is natural to hope for a positive result for extraction as well. In the non-adaptive knowledge soundness definition, the adversary chooses the statement before seeing the CRS, and then outputs a proof for the chosen statement. Intuitively, an extractor for such an adversary *does have* additional power – the extractor can rewind the prover to the point after the statement is chosen, sample different CRS'es and obtain multiple proofs for the same statement. Thus, non-adaptivity makes the prover stateful allowing for rewinding to be useful for an extractor[10]. We give a positive result in the non-adaptive case by showing a SNARK with black-box non-adaptive extraction (for a subset of **NP**). In the construction, we take advantage of our observation that the extractor can obtain more information by seeing multiple proofs corresponding to cleverly crafted CRS'es. At a high level, we ask the prover to encrypt a bit of the witness as part of the proof, in addition to proving the underlying relation. Given the secret key of the encryption scheme as the CRS trapdoor, the extractor can recover this witness

[10] Contrast this with the adaptive case, where the prover is stateless and rewinding is not useful.

bit. Now, the crafted CRS'es are such that they ask for different bits of the witness to be encrypted so that with every rewinding, the extractor learns a new bit until it can completely recover the witness.

While this works for valid statements with a *unique* witness, there are some subtleties that we need to address in order to show extraction for languages that have polynomially many witnesses, that is, class **FewP**. Here, the problem is that the adversary can choose to use a different witness each time, and there is no guarantee that the extractor can collect enough bits for any one witness. We now provide an overview of our construction. Let $\mathcal{R}$ be the relation for the language. We start with an existing SNARG for $\mathcal{R}$ and lift it to a SNARK. We use a Fully Homomorphic Encryption (FHE) scheme in order to hide the index of the bit the prover is asked to encrypt. Intuitively this is to hide the index so that the prover cannot adversarially choose a different witness for different indices. We augment the relation the SNARG proves to include a hash of the witness. Now the extractor keeps track of which witness it is extracting by using the hash to fingerprint. The extractor still needs to collect all bits of one witness. Here, we rely on the semantic security of the FHE scheme to show that the prover cannot consistently use witness w_1 for index i, and witness w_2, for index j. Since there are only polynomially many witnesses, assuming universality (non-adaptive collision resistance) of the hash function, the extractor succeeds in recovering all bits of some witness.

We also show that if the hash is encrypted and the initial SNARG has computational zero-knowledge, the resulting SNARK will also have computational zero-knowledge.

BB extraction is impossible for all NP (non-adaptive case). The previous result, however, cannot be extended to all **NP** languages. We show this by relating an extractor's existence to breaking the relation's leakage resilience. A SNARK proof can be thought of as leakage on the witness. When this leakage is small, no extractor can succeed if the **NP** relation is leakage resilient. This impossibility due to leakage resilience is easy to see in the adaptive case. In non-adaptive extraction, an extractor can potentially rewind the adversary and obtain multiple proofs; akin to a leakage resilience adversary obtaining leakage multiple times. We formalize this connection using *continuous* leakage resilience. In L-leakage-resilient OWF (LR-OWF), one-wayness holds even if L bits of the preimage are leaked. In L-continuous LR-OWF (CLR-OWF), L bits can be leaked multiple times with the caveat that the preimage has to be updated before each leakage. Moreover, if for an OWF f we have $y = f(w)$ and w is updated to w', then also $y = f(w')$.

We connect this primitive to the impossibility of non-adaptive black-box knowledge soundness of SNARKs. Suppose we have a SNARK for the relation $y = f(w)$ where f, y are public, and w is the witness. We view the proof as leakage on the witness given to the adversary. If the proof is at most L bits long, then the extractor can learn at most L bits of information about the witness with each rewinding. Now if the adversary also updates its witness w between queries, L-CLR of f implies that the extractor cannot recover the witness. Thus, the SNARK proof is at least L bits long.

We can instantiate this result with $(1-\frac{2}{n})|w|$-CLR-OWF from [1] which is based on the discrete logarithm assumption. The witness size $|w| = n \log q$, where q is the size of the discrete logarithm group and n is an input size parameter. Thus, the proof size will be asymptotically linear in $|w|$.

Extending GW to preprocessing SNARGs. The central idea in the GW proof is to show that every SNARG for an **NP** language has a *simulatable* adversary. An unbounded adversarial prover that breaks soundness comes with an efficient simulator such that no efficient machine can tell whether it is interacting with the prover or the simulator. A black-box reduction is an efficient oracle-access machine that breaks some falsifiable assumption when given access to a successful adversary. Suppose the reduction given oracle access to the prover breaks the assumption. In that case, the efficient machine with oracle access to the efficient simulator also breaks it since the efficient challenger of the falsifiable assumption cannot distinguish the prover from the simulator. Thus, assuming a simulatable adversary, the theorem follows.

Our proof extending the GW impossibility to preprocessing SNARGs follows the GW template. We observe that the GW proof needs the CRS to be short in constructing a simulatable adversary: the reduction that has oracle access to either the computationally unbounded prover or the efficient simulator can query the oracle with 1^m where m is different from the security parameter n. If m is small enough compared to the actual security parameter n, then the reduction can distinguish the adversary from the simulator. Therefore,

the proof modifies the simulator to behave differently in answering queries with a sufficiently small m; this is done by hardcoding a table of responses as non-uniform advice. The table has hardcoded entries (x, π) for every m and every CRS. Therefore, the CRS size is bounded by a polynomial in the security parameter and cannot grow with the size of the instance.

When considering security-parameter preserving reductions, the reduction queries its oracle with the same security parameter. Therefore, a hardcoded table is unnecessary, and we show how the proof goes through when the size of the CRS depends on the instance, as in indexed relations. We leave the case with non-parameter-preserving reductions as an open problem.

1.2 Related Work

Succinctness vs black-box extraction. Here we discuss works that trade succinctness for black-box extraction. C∅C∅ [48] and Tiramisu [6] aim at compiling a SNARK into a UC-secure scheme. However, this transformation results in NIZK arguments whose proof size and verification time are (quasi-)linear in the witness size. This degradation in succinctness is claimed to be unavoidable if one demands black-box extraction. In [5], Baghery et al. add black-box extraction to [39] SNARK. Although the proof size is again asymptotically linear in the witness size, the authors' goal is to strive for concrete efficiency. In [18], Chase et al. construct controlled malleable proofs that crucially require the stronger black-box version of extractability. Even though their starting point is a SNARG, to obtain black-box extraction of the controlled malleable proof, they give up succinctness and achieve controlled malleable NIZKs.

What is common in all the works above as an idea is to perform verifiable encryption by encrypting the witness and then proving knowledge of the value inside the ciphertext in addition to the original relation. The black-box extractor works by decrypting. This is why the black-box extractor comes at the cost of succinctness: the proof includes a ciphertext and a proof of correct encryption.

Other works. The work in [45] proposes an alternative composability model to the UC model, which can (at least to some extent) use non-black-box extractability and knowledge-type assumptions. In this case, one can still obtain succinct UC SNARKs (under some restrictions) without needing black-box extraction. The recent work in [30] obtains witness-succinct non-interactive arguments of knowledge in UC but applying the random oracle model.

2 Preliminaries

PPT stands for probabilistic polynomial time. We use λ to denote the security parameter. We write $x \leftarrow\!\$\ X$ to denote that x is sampled from a distribution X. If X is a set, then $x \leftarrow\!\$\ X$ denotes uniform sampling. We write $f(\lambda) = \mathsf{negl}(\lambda)$ when f is negligible in λ and $f(\lambda) = \mathsf{poly}(\lambda)$ when f is polynomial in λ. For an integer $N \geq 1$, we define $[N] := \{1, \ldots, N\}$.

Indistinguishability. We say that two distributions X_1 and X_2 are $(s(\lambda), \epsilon(\lambda))$-indistinguishable if for any circuit $\mathcal{D}$ of size $s(\lambda)$, we have $|\Pr[\mathcal{D}(X_1) = 1] - \Pr[\mathcal{D}(X_2) = 1]| \leq \epsilon(\lambda)$.

Hard-on-average problems. We define a language $\mathcal{L} \in \mathbf{NP}$ to be a hard-on-average problem if

- It has an efficient instance sampler $\mathsf{Samp}_{\mathcal{L}}(1^\lambda)$ that outputs $\mathsf{x} \in \mathcal{L}$ together with an **NP** witness w.
- There is an efficient sampler $\mathsf{Samp}_{\bar{\mathcal{L}}}(1^\lambda)$ that with an overwhelming probability outputs $\mathsf{x} \notin \mathcal{L}$.
- It is computationally hard to distinguish outputs of $\mathsf{Samp}_{\mathcal{L}}(1^\lambda)$ and $\mathsf{Samp}_{\bar{\mathcal{L}}}(1^\lambda)$.

Language $\mathcal{L}$ is $(s(\lambda), \epsilon(\lambda))$-hard if distributions of x from $\mathsf{Samp}_{\mathcal{L}}(1^\lambda)$ and $\mathsf{Samp}_{\bar{\mathcal{L}}}(1^\lambda)$ are $(s(\lambda), \epsilon(\lambda))$-indistinguishable. It is sub-exponentially hard if there exits some constant $\delta > 0$ such that previous distributions are $(s(\lambda), \epsilon(\lambda))$-indistinguishable for $s(\lambda) = 2^{\Omega(\lambda^\delta)}$ and $\epsilon(\lambda) = 1/2^{\Omega(\lambda^\delta)}$. Lastly, $\mathcal{L}$ is exponentially hard if the above holds and moreover $|\mathsf{x}| + |\mathsf{w}| = O(\lambda^\delta)$ for $(\mathsf{x}, \mathsf{w}) \leftarrow\!\$\ \mathsf{Samp}_{\mathcal{L}}(1^\lambda)$.

Simple example is the DDH language where $\mathsf{Samp}_{\mathcal{L}}$ outputs group elements g^a, g^b, g^{ab}, where a, b are chosen uniformly at random and g is a group generator, and $\mathsf{Samp}_{\bar{\mathcal{L}}}$ outputs 3 random group elements g^a, g^b, g^c. More generally, hard-on-average problem is implied by the existence of one-way-functions since it is possible to construct a PRG from a one-way function [41].

Falsifiable assumptions. Below we recall the notion of falsifiable assumptions.

Definition 1 ([32]). *A falsifiable cryptographic assumption $(\mathcal{C}, c)$ consists of a PPT challenger $\mathcal{C}$ and a constant $c \in [0, 1)$. We say that $\mathcal{A}$ wins $(\mathcal{C}, c)$ if $\mathcal{A}(1^\lambda)$ and $\mathcal{C}(1^\lambda)$ interact and finally $\mathcal{C}$ outputs 1. The assumption $(\mathcal{C}, c)$ holds if for all non-uniform PPT $\mathcal{A}$, $\Pr[\mathcal{A}$ wins $(\mathcal{C}, c)] \leq c + \mathsf{negl}(\lambda)$. Otherwise we say that $(\mathcal{C}, c)$ is false.*

Definition 1 captures most cryptographic assumptions from the literature. In the case of *search* assumptions (e.g., discrete logarithm problem and shortest vector problem), we set $c = 0$. In the case of *decisional* assumptions (e.g., decisional Diffie-Hellman, decisional Learning with Errors), we set $c = 1/2$ since the adversary can win with probability $1/2$ by random guessing. Knowledge assumptions [25, 40] are seemingly non-falsifiable.

2.1 Continuous Leakage-Resilient OWFs

A leakage-resilient OWF (LR-OWF) f is a function that is one-way even when the adversary is allowed to learn arbitrary functions of $f(x)$'s preimage as long as this leakage is restricted to L bits. Continuous LR-OWF (CLR-OWF) in the floppy model [1, 3] is a generalization of this where leakages can happen multiple times. In short, it assumes a master secret key which is kept in a leakage-free server (e.g., on a floppy disk) and then can be used to securely update the preimage x. L bits of leakage on the preimage can occur after each update. Importantly however, updates have to preserve the output of the OWF, that is $f(x) = f(x')$ when x' is an update of x.

More formally, a CLR-OWF consists of the following probabilistic polynomial time (PPT) algorithms: (1) $\mathsf{KGen}(1^\lambda)$ that outputs a public parameter pp and an update key uk. (2) $\mathsf{Sample}(\mathsf{pp})$ takes as input the parameter pp and outputs a random OWF input x. (3) $\mathsf{Eval}(\mathsf{pp}, x)$ is a deterministic algorithm that produces the OWF output y. (4) $\mathsf{Update}(\mathsf{uk}, x)$ takes in the update key uk and x, and outputs an updated OWF input x'.

We assume that a CLR-OWF satisfies the following properties.

Correctness. For any $(\mathsf{pp}, \mathsf{uk}) \in \mathsf{KGen}(1^\lambda)$ and $x \in \{0,1\}^*$, we have that $\mathsf{Eval}(\mathsf{pp}, \mathsf{Update}(\mathsf{uk}, x)) = \mathsf{Eval}(\mathsf{pp}, x)$.

L-Continuous leakage-resilience. Let $L = L(\lambda)$. For any PPT $\mathcal{A}$,

$$\Pr\left[\begin{array}{l}(\mathsf{pp}, \mathsf{uk}) \leftarrow \mathsf{KGen}(1^\lambda), x \leftarrow \mathsf{Sample}(\mathsf{pp}), \\ \quad y \leftarrow \mathsf{Eval}(\mathsf{pp}, x), x' \leftarrow \mathcal{A}^{\mathsf{O}_L(\cdot)}(\mathsf{pp}, y)\end{array} : y = \mathsf{Eval}(\mathsf{pp}, x')\right] = \mathsf{negl}(\lambda),$$

where $\mathsf{O}_L(\cdot)$ is an oracle that takes as an input a leakage function $h : \{0,1\}^* \to \{0,1\}^L$, on which $\mathsf{O}_L(h)$ sets $x \leftarrow \mathsf{Update}(\mathsf{uk}, x)$ and then returns $h(x)$.

There exists CLR-OWFs [1], which can leak almost the full key. We recall this result in appendix A.

2.2 Argument System

We recall the notion of non-interactive argument systems.

Definition 2 (Indexed relation [19]). *An* indexed relation $\mathcal{R}$ *is a set of triples* $(\mathsf{i}, \mathsf{x}, \mathsf{w})$ *where* i *is the index,* x *is the instance, and* w *is the* **NP***-witness; the corresponding* indexed language $\mathcal{L}(\mathcal{R})$ *is the set of pairs* (i, x) *for which there exists a witness* w *such that* $(\mathsf{i}, \mathsf{x}, \mathsf{w}) \in \mathcal{R}$*. Indexed relation is associated with an efficient index sampling algorithm $\mathcal{I}$ that outputs an index* i *on input* 1^λ*.*

For example, i can be an arithmetic circuit and x and w public and private inputs to the circuit such that the circuit outputs 1. We say that an indexed language is a hard-on-average problem if it is defined like in section 2, but additionally $\mathsf{Samp}_{\mathcal{L}}$ and $\mathsf{Samp}_{\bar{\mathcal{L}}}$ take $\mathsf{i} \leftarrow \mathcal{I}(1^\lambda)$ as an input.

A non-interactive argument system for an indexed relation $\mathcal{R}$ is a tuple of PPT algorithms $\Pi = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$. The setup algorithm $\mathsf{Setup}(1^\lambda, \mathsf{i})$ produces a common reference string crs and a trapdoor td. The prover algorithm $\mathsf{P}(\mathsf{crs}, \mathsf{x}, \mathsf{w})$ produces a proof π for the statement $(\mathsf{i}, \mathsf{x}) \in \mathcal{L}$. The verifier algorithm $\mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi)$ decides if π is a valid proof for a statement (i, x) by outputting either 0 or 1. Notice that P and V are not directly given i as an input and instead get a crs which depends on i. This allows to potentially compress the index description by preprocessing.

We require that Π satisfies the following two properties.

Completeness. For all $(\mathsf{i}, \mathsf{x}, \mathsf{w}) \in \mathcal{R}$, $\Pr[(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}), \pi \leftarrow \mathsf{P}(\mathsf{crs}, \mathsf{x}, \mathsf{w}) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1] = 1$.
Soundness. For all non-uniform PPT adversaries $\mathcal{A}$,

$$\Pr\left[\begin{matrix}\mathsf{i} \leftarrow \mathcal{I}(1^\lambda), (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}) \\ (\mathsf{x}, \pi) \leftarrow \mathcal{A}(1^\lambda, \mathsf{i}, \mathsf{crs})\end{matrix} : \begin{matrix}\mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1 \wedge \\ (\mathsf{i}, \mathsf{x}) \notin \mathcal{L}\end{matrix}\right] = \mathsf{negl}(\lambda) \ .$$

In some parts of the paper (where it does not matter), we drop i and $\mathcal{I}(1^\lambda)$ from the definitions for simplicity. However, index plays a crucial role in section 5.

We call an argument system a SNARG (succinct non-interactive argument) if additionally the following holds.

Proof succinctness. [32] Exists a constant $c < 1$ such that the length of the proof π is bounded by $\mathsf{suc}_c(\lambda, |\mathsf{x}|, |\mathsf{w}|) := \mathsf{poly}(\lambda) \cdot (|\mathsf{x}| + |\mathsf{w}|)^c$.

Various other succinctness definitions can be found from the literature. We occasionally discuss two other forms of succinctness.

Verifier succinctness. Exists a constant $c < 1$ such that the verifier's running time is bounded by $\mathsf{poly}(|\mathsf{x}| + \mathsf{suc}_c(\lambda, |\mathsf{x}|, |\mathsf{w}|))$.
CRS succinctness. CRS size is $\mathsf{poly}(\lambda)$. Importantly, CRS size is independent of $|\mathsf{i}|$.

For example, [16, 19, 39, 55, 57] are proof and verifier succinct but not CRS succinct.

3 On Adaptively-Secure Black-Box Extraction

A folklore understanding is that if an argument has black-box knowledge soundness (i.e., there is an efficient algorithm Ext that can recover a witness from a proof by using a trapdoor and Ext is independent of adversary's code), then the proof has to be "as long as the witness". It is easy to see that such a statement is only partially accurate. Consider an argument system for some relation $\mathcal{R}_{\mathcal{L}}$ where $\mathcal{L}$ is an **NP**-language. The same argument system works for a modified relation $\mathcal{R}'_{\mathcal{L}} = \{(\mathsf{x}, \mathsf{w} \| 0^k) : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_{\mathcal{L}}\}$ where the witness is padded with k zeroes for an arbitrary number k. An extractor Ext for $\mathcal{R}'_{\mathcal{L}}$ needs to append 0^k to the witness it extracts for $\mathcal{R}_{\mathcal{L}}$. Notably, the proof length for $\mathcal{R}'_{\mathcal{L}}$ remains the same as for $\mathcal{R}_{\mathcal{L}}$ independently of witness padding length. This section correctly formalizes the folklore result about the proof size and witness length by associating the hardness of finding the witness with the size of the argument.

We begin by recalling the definition of black-box knowledge soundness.

Black-box knowledge soundness. An argument system is black-box $\varepsilon_{ks}(\lambda)$-knowledge sound for a relation $\mathcal{R}$ if there exists a PPT extractor Ext, such that for any PPT adversary $\mathcal{A}$,

$$\Pr\left[\begin{matrix}(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), (\mathsf{x}, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \\ \mathsf{w} \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, \mathsf{x}, \pi)\end{matrix} : \begin{matrix}\mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1 \wedge \\ (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}\end{matrix}\right] \leq \varepsilon_{ks}(\lambda) \ .$$

We say the argument system is black-box knowledge sound if $\varepsilon_{ks}(\lambda) = \mathsf{negl}(\lambda)$.

We prove that if the witness of the language cannot be guessed, except for probability ε, then the proof size must be at least $-\log(\varepsilon + \varepsilon_{ks})$ bits long. We start by formalizing the witness guessing probability.

Definition 3. *Let $\mathcal{L}$ be an* **NP** *language and $\mathcal{R}_\mathcal{L}$ a corresponding relation. We say that an efficiently sampleable distribution $\mathcal{D}_\mathcal{L}$ over $\mathcal{L}$ is $\varepsilon(\lambda)$-witness-hard for a relation $\mathcal{R}_\mathcal{L}$ if for any PPT guesser $\mathcal{M}$, and any security parameter $\lambda \in \mathbb{N}$,*

$$\Pr[\mathsf{x} \leftarrow \mathcal{D}(1^\lambda), \mathsf{w} \leftarrow \mathcal{M}(1^\lambda, \mathsf{x}) : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_\mathcal{L}] \leq \varepsilon(1^\lambda) \ .$$

Theorem 1. *Suppose an efficiently sampleable distribution $\mathcal{D}_\mathcal{L}$ over some* **NP** *language $\mathcal{L}$ is $\varepsilon(\lambda)$-witness-hard for a relation $\mathcal{R}_\mathcal{L}$. Let Π be an argument system that has (perfect) completeness and black-box $\varepsilon_{ks}(\lambda)$-knowledge soundness. Then the argument size of Π is at least $-\log(\varepsilon(\lambda) + \varepsilon_{ks}(\lambda))$ bits.*

Proof. Suppose that Π is an argument system with black-box extractor Ext and the argument size is bounded by $p(\lambda)$ bits. We construct a witness-guesser $\mathcal{M}^*$ (see fig. 1), which picks a crs and an extraction key td and guesses uniformly randomly a proof π of size $p(\lambda)$ bits. It then returns the output of the black-box witness extractor $\mathsf{Ext}(\mathsf{crs}, \mathsf{td}, \mathsf{x}, \pi)$.

Let us analyze the success probability $\varepsilon_{\mathcal{M}^*}$ of $\mathcal{M}^*$ in the witness-hardness game against $\mathcal{D}_\mathcal{L}$. Let $\mathcal{E}$ be the distribution $(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi)$ obtained by running $\mathsf{x} \leftarrow \mathcal{D}(1^\lambda)$ and $\mathsf{w} \leftarrow \mathcal{M}^*(1^\lambda, \mathsf{x})$ (crs and π are generated inside $\mathcal{M}^*$). Then,

$$\begin{aligned}
\varepsilon_{\mathcal{M}^*} &= \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_\mathcal{L}\right] \\
&\geq \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_\mathcal{L} \wedge \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right] \\
&= \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_\mathcal{L} \mid \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right] \\
&\quad \cdot \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right] \ .
\end{aligned}$$

Let us separately analyze

$$\varepsilon_1 := \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right]$$

and

$$\varepsilon_2 := \Pr\left[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E} : (\mathsf{x}, \mathsf{w}) \in \mathcal{R}_\mathcal{L} \mid \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right]$$

For ε_1: since $\mathsf{x} \in \mathcal{L}$, by perfect completeness there exists at least one proof of size at most $p(\lambda)$ bits that is accepted by the verifier. Thus, $\varepsilon_1 \geq 1/2^{p(\lambda)}$. In order to lower bound ε_2, we construct an adversary $\mathcal{B}$ against black-box knowledge soundness. The adversary $\mathcal{B}$, described in fig. 1, outputs $\mathsf{x} \leftarrow \mathcal{D}_\mathcal{L}$ and a randomly sampled proof $\pi \leftarrow \{0,1\}^{p(\lambda)}$. By inlining $\mathcal{B}$ into the black-box knowledge soundness game, we get $\Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1 \wedge (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}_\mathcal{L}] \leq \varepsilon_{ks}(\lambda)$. That is

$$\begin{aligned}
&\Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1 \wedge (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}_\mathcal{L}] \\
&= \Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}_\mathcal{L} \mid \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1] \\
&\cdot \Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1] \\
&\geq \Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}_\mathcal{L} \mid \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1] \cdot \tfrac{1}{2^{p(\lambda)}} \ .
\end{aligned}$$

Thus, $\Pr[(\mathsf{x}, \mathsf{w}, \mathsf{crs}, \pi) \leftarrow \mathcal{E}(1^\lambda) : (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}_\mathcal{L} \mid \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1] \leq \varepsilon_{ks}(\lambda) \cdot 2^{p(\lambda)}$, which means that $\varepsilon_2 > 1 - \varepsilon_{ks}(\lambda) \cdot 2^{p(\lambda)}$.

By combining those results, we get that $\varepsilon(\lambda) \geq \varepsilon_{\mathcal{M}^*} > \frac{1}{2^{p(\lambda)}} \cdot (1 - \varepsilon_{ks}(\lambda) \cdot 2^{p(\lambda)}) = \frac{1}{2^{p(\lambda)}} - \varepsilon_{ks}(\lambda)$. It follows that $\varepsilon(\lambda) + \varepsilon_{ks}(\lambda) > \frac{1}{2^{p(\lambda)}}$, which we can rewrite as $p(\lambda) > -\log(\varepsilon(\lambda) + \varepsilon_{ks}(\lambda))$. □

To understand this claim better, let us consider for simplicity that $\varepsilon_{ks}(\lambda) = 0$. Then if $\varepsilon = \frac{1}{2^{k(\lambda)}}$, we obtain the lower bound $p(\lambda) \geq -\log(\frac{1}{2^{k(\lambda)}} + 0) = k(\lambda)$. In one extreme case, we can imagine that the best PPT witness guesser is no better than an algorithm that guesses the witness at random, i.e., $\varepsilon(\lambda) = 1/|\mathsf{w}|$. Then we would get the folklore result that $p(\lambda) = |\pi| \geq |\mathsf{w}|$. In the other extreme, suppose that the language is in P, in which case $\varepsilon(\lambda) = 1$. Then we get that $-\log(\varepsilon) = 0$, which fits the intuition that there is no need to communicate a proof for languages in P. However, in a typical situation (where we have some hard language), the lower bound falls somewhere between those extremes.

A closely related but somewhat less precise result can be directly concluded from leakage-resilient OWFs by viewing a proof as leakage on the witness. We show the proof of this impossibility using LR-OWFs in appendix B.

$\mathcal{M}^*(1^\lambda, \mathsf{x})$

$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda); \pi \leftarrow\!\!\$\ \{0,1\}^{p(\lambda)};$

return $\mathsf{w} \leftarrow \mathsf{Ext}(\mathsf{crs}, \mathsf{td}, \mathsf{x}, \pi);$

$\mathcal{B}(\mathsf{crs})$

$\mathsf{x} \leftarrow \mathcal{D}(1^\lambda); \pi \leftarrow\!\!\$\ \{0,1\}^{p(\lambda)};$

return $(\mathsf{x}, \pi);$

Fig. 1: A witness guessing algorithm $\mathcal{M}^*$ for $\mathcal{R}_\mathcal{L}$ and a knowledge soundness adversary $\mathcal{B}$

4 Non-Adaptive Black-Box Knowledge Soundness

This section defines non-adaptive black-box knowledge soundness and shows our positive result for **FewP** and a negative result for **NP**.

Below we define non-adaptive black-box *knowledge-soundness*. To the best of our knowledge it has not appeared in prior literature.

Definition 4 (Non-adaptive Black-box Knowledge Soundness.). *An argument system is non-adaptive black-box $\varepsilon_{ks}(\lambda)$-knowledge sound for a relation $\mathcal{R}$ if there exists a non-uniform PPT extractor* Ext*, such that for any non-uniform PPT adversary $\mathcal{A} = (\mathcal{A}_{inp}, \mathcal{A}_{prf})$,*

$$\Pr\left[\begin{matrix}(\mathsf{x}, \mathsf{st}) \leftarrow \mathcal{A}_{inp}(1^\lambda), (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda) \\ \pi \leftarrow \mathcal{A}_{prf}(\mathsf{st}, \mathsf{crs}), \mathsf{w} \leftarrow \mathsf{Ext}^{\mathcal{A}_{prf}(\mathsf{st},\cdot)}(\mathsf{crs}, \mathsf{td}, \mathsf{x}, \pi)\end{matrix} : \begin{matrix}\mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1 \\ \wedge (\mathsf{x}, \mathsf{w}) \notin \mathcal{R}\end{matrix}\right] \leq \varepsilon_{ks}(\lambda) \ .$$

We say that the argument system is (non-adaptively) black-box knowledge sound if $\varepsilon_{ks}(\lambda) = \mathsf{negl}(\lambda)$.

Remark 1. The adversary in definition 4 is stateful only between the input-challenge stage and the proof-challenge stage (through st), but not otherwise. We also assume that on each query $\mathcal{A}_{prf}(\mathsf{st}, \cdot)$ gets fresh random coins.

4.1 A Construction for FewP

In this section, we show that, under the existence of fully homomorphic encryption, universal hash functions and SNARGs (not necessarily of knowledge) for a specific complexity class K, there exists a non-adaptively secure SNARK with black-box extraction for K[11]. We can obtain non-adaptive black-box knowledge soundness for a non-trivial subset of **NP** called **FewP**. The class **FewP** can be described as the class of languages admitting at most a polynomial number of witnesses. We remark that if one-way permutations exist, then $\mathbf{P} \neq \mathbf{FewP}$[12]. One example of a natural application of a SNARK for **FewP** is proving knowledge of w such that $\mathcal{R}(\mathsf{w})$ is satisfied (for arbitrary relation $\mathcal{R}$) and w opens a perfectly binding commitment.

Further preliminaries for this section can be found in appendix A, where we define the non-adaptive soundness of SNARG (which simply adapts definition 4 to the non-extractable case) and the standard definitions of fully homomorphic encryption (FHE) and universal hash functions (UHF) which will be tools in our construction.

We present our extractable construction in fig. 2[13]. As discussed in the introduction, its main intuition is that the prover provides a (ciphertext containing a) bit of the witness together with the proof. The index for which it is providing such a bit must be somehow hidden. This intuitively prevents the adversary from acting differently for different bits (e.g., using different valid witnesses). This allows us to extract by repeatedly asking the prover for a proof referring to a different index. To achieve the latter, we use an FHE scheme (see also remark 2). When extracting, we will need to keep track of what witness we are extracting (since

[11] This class should include FHE encryption and UHF and should be closed under conjunction. In our theorem statement, we simply require a SNARG for **NP**.

[12] More generally, if poly-to-one one-way functions exist then $\mathbf{P} \neq \mathbf{FewP}$ [2].

[13] A slightly simpler construction for the case of **UP** (**NP** statements with a unique witness) is in appendix D.

$\mathsf{Setup}(1^\lambda)$

$(\hat{\mathsf{crs}}, \hat{\mathsf{td}}) \leftarrow \Pi_\exists.\mathsf{Setup}(1^\lambda)$

$i^* \leftarrow\$ [N_w]$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KG}(1^\lambda)$

$(\mathsf{pk}_{\mathrm{FHE}}, \mathsf{sk}_{\mathrm{FHE}}) \leftarrow \mathsf{FHE.KG}(1^\lambda)$

$\mathsf{ct}_{i^*} \leftarrow \mathsf{FHE.Enc}(\mathsf{pk}, i^*)$

$\mathsf{hk} \leftarrow\$ \mathcal{K}_{\mathrm{UHF}}$

return $(\mathsf{crs} := (\hat{\mathsf{crs}}, \mathsf{ct}_{i^*}, \mathsf{hk}, \mathsf{pk}, \mathsf{pk}_{\mathrm{FHE}}), \mathsf{td} := (\mathsf{sk}, \mathsf{sk}_{\mathrm{FHE}}, \hat{\mathsf{td}}))$

$\mathsf{P}(\mathsf{crs}, \mathcal{R}, \mathsf{x}, \mathsf{w})$

$\mathsf{ct}_{\mathrm{bit}} \leftarrow \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$

where $f_{\mathrm{proj}}(i, w) := w_i$

$\pi \leftarrow \Pi_\exists.\mathsf{P}(\hat{\mathsf{crs}}, \mathcal{R}', (\mathsf{x}, \mathsf{hk}, \mathsf{pk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_h, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}}), \mathsf{w})$

where $\mathcal{R}'(\mathsf{x}, \mathsf{hk}, \mathsf{pk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_h, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}}; \mathsf{w}, r) \iff$

$\mathcal{R}(\mathsf{x}, \mathsf{w}) \wedge \mathsf{ct}_h = \mathsf{PKE.Enc}(\mathsf{pk}, \mathsf{H}_{\mathsf{hk}}(\mathsf{w}), r) \wedge \mathsf{ct}_{\mathrm{bit}} = \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$

return $\pi^* := (\pi, \mathsf{ct}_h, \mathsf{ct}_{\mathrm{bit}})$

$\mathsf{V}(\mathsf{crs}, \mathcal{R}, \mathsf{x}, \pi^*)$

Parse π^* as $(\pi, \mathsf{ct}_h, \mathsf{ct}_{\mathrm{bit}})$

return $\Pi_\exists.\mathsf{V}\left(\hat{\mathsf{crs}}, \mathcal{R}', (\mathsf{x}, \mathsf{hk}, \mathsf{pk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_h, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}})\right)$

where $\mathcal{R}'$ is defined like above

Fig. 2: Non-adaptively secure black-box extractable construction for **FewP**. N_w is a bound on the witness size. $\Pi_\exists$ is the SNARG scheme.

there could be several). We do this by using a fingerprint through a universal hash function. We encrypt the hash with some (not necessarily homomorphic) cryptosystem to obtain zero-knowledge (ZK). In fact, our soundness proof requires that the prover encrypts the hash with a different public key than the witness bit.

We denote FHE encryptions of a message x (with an implicit public-key that should be clear from the context) through double brackets $[\![x]\!]$.

The extractor for **FewP**. The extractor is presented in fig. 3. It works by collecting different bits of the witness by decrypting ct_b (the ciphertext returned by the prover) and storing it in some table indexed by the corresponding hash. The crucial point is that there is only a polynomial number of witnesses and thus the extractor can (in the worst-case) "fingerprint" them all. Hashing the witness (through a universal hash function) keeps the proof succinct. We finally encrypt the latter hash to achieve zero-knowledge.

Theorem 2. *If $\Pi_\exists$ is a non-adaptively sound SNARG scheme for* **NP**, FHE *is a semantically secure FHE scheme,* PKE *is a sematically secure cryptosystem with perfect correctness and* H *is a family of UHFs, then the construction in fig. 2 is a non-adaptive SNARK for* **FewP** *satisfying definition 4. If $\Pi_\exists$ has additionally computational ZK, then so does the resulting SNARK.*

The proof of the theorem above is in appendix C.1.

Remark 2 (On replacing FHE with PIR). While our construction is described with FHE, we observe that the assumption of FHE can easily be replaced by the milder existence of Private Information Retrieval (or PIR) [20].

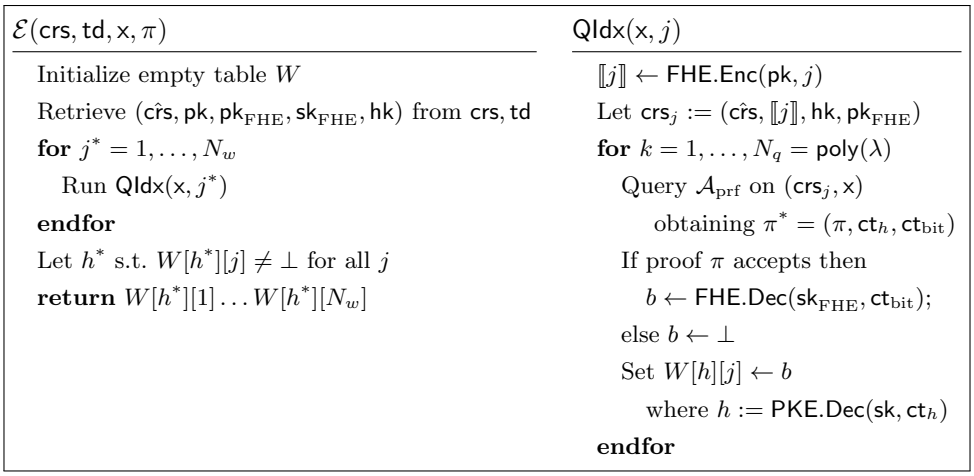

Fig. 3: Extractor for the case for **FewP**

4.2 Impossibility for All NP

We now show that the previous constructive result cannot be extended from **FewP** to **NP**. We mentioned at the end of section 3 that we could view the proof as a leakage of the witness and use leakage resilient (LR) cryptography to prove the impossibility of succinct black-box adaptive knowledge soundness. (appendix B) For the non-adaptive case, we can no longer view the SNARK proof as a *one-time* leakage since the extractor (LR adversary) has the ability to rewind the prover and obtain multiple proofs (leakages). Using *continuous leakage resilience*, we extend the impossibility to non-adaptive extraction.

Consider a L-CLR-OWF $\Sigma = (\mathsf{KGen}, \mathsf{Sample}, \mathsf{Eval}, \mathsf{Update})$. We define a relation $\mathcal{R}_\Sigma = \{((\mathsf{pp}, y), w) : \mathsf{pp} \in \mathsf{KGen}(1^\lambda), w \in \mathsf{Sample}(\mathsf{pp}), \mathsf{Eval}(\mathsf{pp}, w) = y\}$. Suppose there is a non-adaptive black-box extractable SNARK for $\mathcal{R}_\Sigma$. Let us further assume that the proof size of this SNARK is less than L bits.

We construct the following adversary $\mathcal{A}$. First, $\mathcal{A}$ samples $(\mathsf{pp}, \mathsf{uk}) \leftarrow \mathsf{KGen}(1^\lambda)$, a random w, and outputs $((\mathsf{pp}, y = \mathsf{Eval}(\mathsf{pp}, w)), \mathsf{st} = (\mathsf{pp}, \mathsf{uk}, w))$. Next, the extractor can query $\mathcal{A}(\mathsf{st}, \cdot)$ with different CRS'es and get proofs for the statement $(\mathsf{pp}, y) \in \mathcal{L}_{\mathcal{R}_\Sigma}$. Here we define $\mathcal{A}$'s behavior as follows: on each query, $\mathcal{A}$ updates w, that is it computes $w' \leftarrow \mathsf{Update}(\mathsf{uk}, w)$. Then it creates a proof with w', $\pi \leftarrow \mathsf{P}(\mathsf{crs}, (\mathsf{pp}, y), w')$, and returns π. This proof is at most size L, thus at most L bits of information about w' gets leaked. By L-CLR property it is not possible to recover a witness for (pp, y) from this amount of information. Hence, the extractor cannot extract the witness and such a SNARK cannot exist. We show this formally.

Theorem 3. *Let $\Sigma = (\mathsf{KGen}, \mathsf{Sample}, \mathsf{Eval}, \mathsf{Update})$ be an L-CLR-OWF and let Π be a non-adaptive black-box $\varepsilon_{ks}(\lambda)$-knowledge sound argument for $\mathcal{R}_\Sigma$ as defined above. If the proof size is less than $L(\lambda)$ bits, then L-CLR-OWF can be broken with probability $1 - \varepsilon_{ks}(\lambda)$.*

The proof of this theorem is in appendix C.2. By combining theorem 3 and theorem 7, we obtain the following result.

Theorem 4. *If the discrete logarithm assumption holds in some group, then there exists an* **NP***-language $\mathcal{L}$ such that any non-adaptive BB knowledge sound argument system for $\mathcal{R}_\mathcal{L}$ has a proof size $\Omega(|w|)$ where $|w|$ is the witness size.*

5 GW Impossibility for Preprocessing SNARGs

Careful study of [32] reveals that the CRS generation algorithm of a SNARG in their definition depends only on the security parameter. In other words, the proof separating SNARGs from falsifiable assumptions

assumes that the SNARG is CRS succinct and does not allow preprocessing. Many modern SNARGs have a relatively large CRS, which depends on the size of the index i (e.g., a circuit description) in some way [16, 19, 29, 31, 39, 57]. This makes it questionable if the impossibility result of Gentry and Wichs extends to such SNARGs. We reprove the impossibility theorem for SNARGs that are not necessarily CRS-succinct.

Let us recall the leakage lemma from [32]. We say that a distribution A over tuples (x, π) is an augmented distribution of X if x is distributed according to X and π is some arbitrary information, possibly correlated to x. More formally, we may write A is the distribution over (x, π) such that $x \leftarrow\!\!\$\ X$ and $\pi \leftarrow f(x)$ where f is some (randomized and possibly inefficiently computable) function.

Lemma 1 (Leakage lemma [32]). *There exists a polynomial p for which the following holds. Let X_λ and $\bar{X}_\lambda$ be two distributions that are $(s(\lambda), \varepsilon(\lambda))$-indistinguishable. Let A_λ over (x, π) be an augmented distribution of X_λ, where $|\pi| = \ell(\lambda)$. Then there exist an augmented distribution $\bar{A}_\lambda$ of $\bar{X}_\lambda$ such that A_λ and $\bar{A}_\lambda$ are $(s^*(\lambda), \varepsilon^*(\lambda))$-indistinguishable where $s^*(\lambda) = s(\lambda)p(\varepsilon(\lambda)/2^{\ell(\lambda)})$ and $\varepsilon^*(\lambda) = 2\varepsilon(\lambda)$.*

We also present some definitions which help to prove the main result.

Definition 5 (Breaking Adaptive Soundness [56]). *We say that an algorithm $\mathcal{A}$ breaks adaptive soundness of a SNARG Π for a language $\mathcal{L}$ with probability $\varepsilon(\cdot)$ if there exists an index $\mathsf{i} \in \mathcal{I}$ such that for every $\lambda \in \mathbb{N}$,*

$$\Pr[\mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}), (\mathsf{x}, \pi) \leftarrow \mathcal{A}(1^\lambda, \mathsf{crs}) : (\mathsf{i}, \mathsf{x}) \notin \mathcal{L} \wedge \mathsf{Verify}(\mathsf{crs}, \mathsf{x}, \pi) = 1] \geq \varepsilon(\lambda).$$

Note that if $\varepsilon(\lambda)$ is non-negligible, then adaptive soundness cannot be satisfied.

Definition 6 (Soundness Reduction [56]). *We say that a PPT machine R is a black-box reduction for adaptive soundness of an argument Π based on a falsifiable assumption $(\mathcal{C}, c)$ if there exists a polynomial $p(\cdot, \cdot)$ such that for every $\mathcal{A}$ that breaks adaptive soundness with probability $\varepsilon(\cdot)$, for every $\lambda \in \mathbb{N}$, $R^{\mathcal{A}}(1^\lambda)$ wins $(\mathcal{C}, c)$ with a probability at least $p(\varepsilon(\lambda), 1/\lambda)$.*

We say that $R^{\mathcal{A}}$ is *security-parameter preserving* if additionally there exist a polynomial q such that $R^{\mathcal{A}}(1^\lambda)$ queries $\mathcal{A}$ with inputs of the form $(1^\lambda, x \in \{0,1\}^*)$ and at most $q(\lambda)$ times.

We start by stating two technical lemmas, which we prove in appendix C.

Lemma 2. *If an indexed languages $\mathcal{L} \in \mathbf{NP}$ has a sub-exponentially hard-on-average problem, then for any $d > 0$, $\mathcal{L}$ also has a hard-on-average problem with $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishability.*

Lemma 3. *Let X_λ and $\bar{X}_\lambda$ be $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishable distributions for some integer $d \geq 2$. Let A_λ over (x, π) be an augmented distribution of X_λ, where $|\pi| = \ell(\lambda) = o(\lambda^d)$. Then there exists an augmented distribution $\bar{A}_\lambda$ of $\bar{X}_\lambda$ such that A_λ and $\bar{A}_\lambda$ are $(\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$-indistinguishable.*

Remark 3. Note that $(s(\lambda), \varepsilon(\lambda)) = (\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$-indistinguishability is not enough in the previous lemma because. Suppose $|\pi| = \ell(\lambda) = \lambda^{d-1}$. Then,

$$\begin{aligned} s^*(\lambda) &= s(\lambda)p(\varepsilon(\lambda)/2^{\ell(\lambda)}) = \mathsf{poly}(\lambda)p(\mathsf{negl}(\lambda)/2^{\lambda^{d-1}}) \\ &= \mathsf{poly}(\lambda)p(2^{-\omega(\lambda^{d-1})}) = 2^{-\omega(\mathsf{poly}(\lambda))}\ , \end{aligned}$$

given that p is not a constant polynomial. Thus, A_λ and $\bar{A}_\lambda$ would be (provably) indistinguishable only for very small circuits.

Now we are ready to restate the Gentry-Wichs impossibility result with respect to preprocessing SNARGs from section 2.2.

Theorem 5. *Assume that,*
- *$\mathcal{L}$ is an indexed language with a sub-exponentially hard-on-average problem (see section 2).*
- *Π is a SNARG for $\mathcal{L}$, i.e., it is complete, sound, and proof-succinct (but not necessarily verifier-succinct or CRS-succinct).*

$\mathcal{A}^*(1^\lambda, \mathsf{crs}, \mathsf{i})$	$\mathsf{Emul}(1^\lambda, \mathsf{crs}, \mathsf{i})$	$\mathsf{O}_i(1^\lambda, \mathsf{crs}, \mathsf{i})$//Initially $j = 1$
if $\mathsf{i} \notin \mathcal{I}(1^\lambda)$	**if** $\mathsf{i} \notin \mathcal{I}(1^\lambda)$	**if** $j \leq i$
return $\bot$;	**return** $\bot$;	$(\mathsf{x}, \pi) \leftarrow \mathsf{Emul}(1^\lambda, \mathsf{crs}, \mathsf{i})$;
$(\bar{\mathsf{x}}, \bar{\pi}) \leftarrow \bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$;	$(\mathsf{x}, \mathsf{w}) \leftarrow \mathsf{Samp}_{\mathcal{L}}(1^\lambda, \mathsf{i})$;	**else**
return $(\bar{\mathsf{x}}, \bar{\pi})$;	$\pi \leftarrow \mathbf{P}(\mathsf{crs}, \mathsf{x}, \mathsf{w})$;	$(\mathsf{x}, \pi) \leftarrow \mathcal{A}^*(1^\lambda, \mathsf{crs}, \mathsf{i})$;
	return (x, π);	$j \leftarrow j + 1$;
		return (x, π);

Fig. 4: Soundness adversary $\mathcal{A}^*$, its efficient emulator Emul, and hybrid adversaries O_i

Then, for any falsifiable assumption $(\mathcal{C}, c)$ either:
- *(C, c) is false or,*
- *there is no security-parameter preserving black-box reduction for adaptive soundness of Π based on $(\mathcal{C}, c)$.*

Proof. Suppose there exists a security-parameter preserving black-box reduction R for adaptive soundness of Π based on a falsifiable assumption $(\mathcal{C}, c)$ and that R makes at most $q(\lambda)$ queries to its oracle, where q is some polynomial. The proof idea is to construct a computationally unbounded adversary $\mathcal{A}^*$ that is able to break adaptive soundness. Then we show using lemma 1 that there is an efficient emulator Emul that gives outputs which are indistinguishable from outputs of $\mathcal{A}^*$. Thus, if $R^{\mathcal{A}^*}(1^\lambda)$ is able to break the assumption $(\mathcal{C}, c)$, then so is $R^{\mathsf{Emul}}(1^\lambda)$ and it follows that $(\mathcal{C}, c)$ must be false.

Since Π is proof-succinct there exists some n such that the proof size ℓ is bounded by $\lambda^n \cdot (|\mathsf{x}| + |\mathsf{w}|)^{\mathrm{o}(1)}$. Moreover by lemma 2, since we assume that some sub-exponentially hard-on-average problem exists for $\mathcal{L}$, there also exists a sub-exponentially hard-on-average problem with $(2^{\lambda^{n+2}}, 1/2^{\lambda^{n+2}})$-indistinguishability. Let it be defined by an index sampler $\mathcal{I}$ and instance samplers $\mathsf{Samp}_{\mathcal{L}}$ and $\mathsf{Samp}_{\bar{\mathcal{L}}}$. It is more convenient to start from describing the emulator Emul before we describe $\mathcal{A}^*$. The emulator (see also fig. 4) on input $(1^\lambda, \mathsf{crs}, \mathsf{i})$ checks that i is well-formed, samples $(\mathsf{x}, \mathsf{w}) \leftarrow \mathsf{Samp}_{\mathcal{L}}(1^\lambda, \mathsf{i})$, creates a proof $\pi \leftarrow \mathsf{P}(\mathsf{crs}, \mathsf{x}, \mathsf{w})$ and returns (x, π).

Notice that since $\mathsf{Samp}_{\mathcal{L}}$ runs in polynomial time in λ, then $|\mathsf{x}| = \mathsf{poly}(\lambda)$ and $|\mathsf{w}| = \mathsf{poly}(\lambda)$. Therefore, the proof size is $\ell(\lambda) = \lambda^{\mathrm{o}(n^{d+2})}$.

Fix an arbitrary oracle input $(1^\lambda, \mathsf{crs}, \mathsf{i})$. Let $X_{\lambda,\mathsf{i}}$ be the distribution of x from sampling $(\mathsf{x}, \mathsf{w}) \leftarrow \mathsf{Samp}_{\mathcal{L}}(1^\lambda, \mathsf{i})$ and $\bar{X}_{\lambda,\mathsf{i}}$ the distribution of $\bar{\mathsf{x}}$ we get by sampling $\bar{\mathsf{x}} \leftarrow \mathsf{Samp}_{\bar{\mathcal{L}}}(1^\lambda, \mathsf{i})$. As we established, these distributions are $(2^{\lambda^{n+2}}, 1/2^{\lambda^{n+2}})$-indistinguishable. Let $A_{\lambda,\mathsf{i},\mathsf{crs}}$ be the augmented distribution of $X_{\lambda,\mathsf{i}}$ defined as $(\mathsf{x}, \pi) \leftarrow \mathsf{Emul}(1^\lambda, \mathsf{crs}, \mathsf{i})$. By lemma 3, there exists an augmented distribution $\bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$ of $\bar{X}_{\lambda,\mathsf{i}}$ such that $A_{\lambda,\mathsf{i},\mathsf{crs}}$ and $\bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$ are $(\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$-indistinguishable.

Now we can describe the adversary $\mathcal{A}^*$. On the query input $(1^\lambda, \mathsf{crs}, \mathsf{i})$ it simply returns $(\bar{x}, \bar{\pi}) \leftarrow\!\!\$\ \bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$. Since $\bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$ is not necessarily efficiently sampleable, $\mathcal{A}^*$ may be inefficient.

Our goal is to show that the assumption $(\mathcal{C}, c)$ is false if R exists, i.e.,

$$\Pr[R^{\mathsf{Emul}}(1^\lambda) \text{ wins } (\mathcal{C}, c)] > c + \mathsf{negl}(\lambda) \ .$$

We show this in two parts.

1) $R^{\mathcal{A}^*}$ wins $(\mathcal{C}, c)$: First, let $\varepsilon_{\mathcal{A}^*}(\lambda)$ be the probability that $\mathcal{A}^*$ breaks adaptive soundness of Π,

$$\varepsilon_{\mathcal{A}^*}(\lambda) := \Pr\left[\begin{matrix}\mathsf{i} \leftarrow \mathcal{I}(1^\lambda), \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}) \\ (\mathsf{x}, \pi) \leftarrow \mathcal{A}^*(1^\lambda, \mathsf{crs}, \mathsf{i})\end{matrix} \ : \ (\mathsf{i}, \mathsf{x}) \notin \mathcal{L} \wedge \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right].$$

Let us first only consider the probability of the verifier accepting a proof,

$$\varepsilon_{\mathsf{Vf}=1}(\lambda) := \Pr\left[\begin{matrix}\mathsf{i} \leftarrow \mathcal{I}(1^\lambda), \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}), \\ (\mathsf{x}, \pi) \leftarrow \mathcal{A}^*(1^\lambda, \mathsf{crs}, \mathsf{i})\end{matrix} \ : \ \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right].$$

Due to completeness, we know that $\varepsilon_{\mathsf{Emul}}(\lambda) = 1$, where

$$\varepsilon_{\mathsf{Emul}}(\lambda) := \Pr\left[\begin{matrix}\mathsf{i} \leftarrow \mathcal{I}(1^\lambda), \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}), \\ (\mathsf{x}, \pi) \leftarrow \mathsf{Emul}(1^\lambda, \mathsf{crs}, \mathsf{i})\end{matrix} \; : \; \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi)\right] = 1].$$

Since V can be seen as a polynomial-size distinguisher for $A_{\lambda,\mathsf{i},\mathsf{crs}}$ and $\bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$, we get from before that $|\varepsilon_{\mathsf{Emul}}(\lambda) - \varepsilon_{\mathsf{Vf}=1}(\lambda)| \leq \mathsf{negl}(\lambda)$. Therefore, $1 - \mathsf{negl}(\lambda) \leq \varepsilon_{\mathsf{Vf}=1}$. Since $\Pr[\mathsf{i} \leftarrow \mathcal{I}(1^\lambda), \mathsf{crs} \leftarrow \mathsf{Setup}(1^\lambda, \mathsf{i}), (\mathsf{x}, \pi) \leftarrow \mathcal{A}^*(\mathsf{crs}, \mathsf{i}) : (\mathsf{i}, \mathsf{x}) \notin \mathcal{L}] = 1$, $\varepsilon_{\mathcal{A}^*} = \varepsilon_{\mathsf{Vf}=1} \geq 1 - \mathsf{negl}(\lambda)$. Thus, $\mathcal{A}^*$ breaks adaptive soundness with an overwhelming probability. Since we assumed a black-box reduction R, there must exist a polynomial $p(\cdot, \cdot)$ such that $R^{\mathcal{A}^*}(1^\lambda)$ breaks $(\mathcal{C}, c)$ with probability at least $p(1 - \mathsf{negl}(\lambda), 1/\lambda)$.

2) $R^{\mathcal{A}^*}$ is indistinguishable from R^{Emul}: Suppose R makes $q(\lambda)$ oracle queries. Let O_i for $i \in \{0, \ldots, q(\lambda)\}$ denote a stateful algorithm that we describe in the following. The machine O_i for the first i queries responds as Emul and for the rest of the queries $(1^\lambda, \mathsf{crs}, \mathsf{i})$ responds as $\mathcal{A}^*$ (see fig. 4). In particular $\mathsf{O}_0 = \mathcal{A}^*$ and $\mathsf{O}_{q(\lambda)} = \mathsf{Emul}$. We denote $\varepsilon_i := \Pr[R^{\mathsf{O}_i}(1^\lambda) \text{ wins } (\mathcal{C}, c)]$. We can again use indistinguishability of $A_{\lambda,\mathsf{i},\mathsf{crs}}$ and $\bar{A}_{\lambda,\mathsf{i},\mathsf{crs}}$ to show that $|\varepsilon_i - \varepsilon_{i+1}| \leq \mathsf{negl}(\lambda)$. Therefore, by triangle inequality $|\varepsilon_0 - \varepsilon_{q(\lambda)}| \leq q(\lambda)\mathsf{negl}(\lambda) = \mathsf{negl}(\lambda)$.

Since $\varepsilon_0 = \varepsilon_{\mathcal{A}}$, we get that $\Pr[R^{\mathsf{Emul}}(1^\lambda) \text{ wins } (\mathcal{C}, c)] = \varepsilon_{q(\lambda)} \geq \epsilon_{\mathcal{A}} - \mathsf{negl}(\lambda) = p(1 - \mathsf{negl}(\lambda), 1/\lambda) - \mathsf{negl}(\lambda)$. Thus, $R^{\mathsf{Emul}}(1^\lambda)$ can break the assumption $(\mathcal{C}, c)$ with an overwhelming probability. □

6 Understanding SNARG Impossibilities

In this section, we attempt to provide a complete overview of known impossibilities for non-interactive arguments. This illustrates the precise assumptions behind these impossibilities in order to identify avenues for further research. The following are some of the major impossibility results.

1. Gentry-Wichs [32]: Adaptive soundness of a SNARG cannot be proven via a black-box reduction to a falsifiable assumption.
2. Pass [56]: Adaptive soundness of a statistical NIZK argument cannot be proven via a black-box reduction to a falsifiable assumption.
3. Groth [39]: Any pairing-based SNARK obtained from a NILP (a non-interactive linear proof) must contain at least 2 group elements, one in each of the pairing source groups.

Since [39] is relevant only in a very specific setting, and [56] is about general NIZKs, we will not focus on it in the rest of the paper. The proof idea of [32] is quite similar to our extension of it. We recall the proof idea of [56] in appendix F and the proof idea of [32] in appendix E. In the following, we discuss the impossibility result of [32] and then outline the landscape of positive and negative results in Table 1.

6.1 Impossibility of Gentry-Wichs

We recall the main result of [32].

Theorem 6. *Let $\mathcal{L}$ be a sub-exponentially hard* **NP** *language and let Π be a SNARG for $\mathcal{L}$, satisfying completeness and proof succinctness. Then, for any falsifiable assumption $(\mathcal{C}, c)$, either $(\mathcal{C}, c)$ is false, or there is no black-box reduction showing the (adaptive) soundness of Π based on $(\mathcal{C}, c)$.*

We take a closer look at GW impossibility and enumerate the scenarios to which it *does not* apply. While some of these are known results, they are all scattered in the literature.

– **Non-adaptive soundness**: The impossibility holds only for *adaptive soundness*. The proof technique used in GW to rule out a black-box reduction uses a stateless adversary that outputs an instance proof pair (x, π) on input a CRS. In particular, this does not rule out our reductions that can rewind the prover and obtain different proofs for the same x, which is possible in the case of non-adaptive soundness. Recent work attempted to show tightness from new albeit falsifiable assumptions, but the construction was shown to be faulty [60].

– **Low-space non-deterministic computation**: The high-level idea of the GW impossibility result is a "leakage lemma" that says the following: assuming the underlying NP language is 2^ℓ-hard, a reduction that breaks the assumption, cannot distinguish between pairs (x, π) generated by a (possibly inefficient) cheating prover, where $\mathsf{x} \notin L$ and π is a proof of length ℓ, and a pair $(\tilde{\mathsf{x}}, \tilde{\pi})$ where $\tilde{\mathsf{x}} \in L$ and $\tilde{\pi}$ is an efficiently generated proof. Therefore, for computations recognizable in $\mathsf{poly}(\lambda)$ time and $S(\lambda)$ space with a non-deterministic Turing machine (the class $NTISP(\mathsf{poly}(\lambda), S(\lambda))$), the GW result does not rule out the possibility of a SNARG with proofs of length $\mathsf{poly}(\lambda)(S(\lambda))$, since a computation in $NTISP(\mathsf{poly}(\lambda), S(\lambda))$ is in $DTIME(\mathsf{poly}(\lambda) \cdot 2^{S(\lambda)})$ which is not $\mathsf{poly}(\lambda) \cdot 2^{O(S(\lambda))}$-hard. The work of [4] constructs a delegation scheme for non-deterministic computations with a proof length that grows only with the space of the computation.
– **Preprocessing SNARGs**: The GW separation result holds for SNARGs that have a "short" CRS. More precisely, the impossibility proof requires that the size of the CRS depends only on the security parameter, and does not grow with the size of the instance. This gap is now closed with the current work.
– **Trapdoor languages**: For some languages there are efficient proofs [42, 49] in the quasi-adaptive setting (QA-NIZK). These proofs have a constant number group elements – regardless of the instance size. The construction of [47] for languages consisting of linear subspaces of a vector space, have constant-sized proofs, achieve adaptive soundness (based on a falsifiable assumption) and perfect ZK. This seemingly contradicts the GW impossibility result (as well as the impossibility on perfect ZK [56]).[14] The results in the quasi-adaptive setting do not contradict the GW impossibility because the CRS hides a trapdoor for deciding membership in the language. The proof of GW rules out reductions that *cannot* efficiently detect when the soundness property is broken. We formalize this notion of *trapdoor languages* below.

Bypassing GW: Trapdoor Languages. Intuitively, a trapdoor language allows verifying membership in the language if one knows a trapdoor. Towards formalizing such languages, we illustrate it by taking the linear subspace language as an example. Recall the language of linear subspaces from [47]. We have a distribution $\mathcal{D}$ that outputs a language parameter $\mathsf{lpar} = [M]_1 \in \mathbb{G}_1^{n \times m}$[15] for some matrix M and the respective linear subspace language is defined as

$$\mathcal{L}_{[M]_1} = \{[\vec{x}]_1 \in \mathbb{G}_1^n \mid \exists \vec{\mathsf{w}} \in \mathbb{Z}_p^m : \vec{x} = M \cdot \vec{\mathsf{w}}\}.$$

In the proof of [32], the reduction algorithm R that picks the CRS, (and in the case of the linear subspace language, also picks the language parameter lpar), should not efficiently distinguish between elements $\mathsf{x} \in \mathcal{L}$ from $\overline{\mathsf{x}} \in \overline{\mathcal{L}}$. The latter condition does not hold for linear subspace languages.

In particular, we now argue that it is possible to efficiently decide if $[\vec{x}]_1 \in \mathcal{L}_{[M]_1}$ by knowing M.

Observe that given both M and $\vec{x}$ as integers, by Kronecker–Capelli theorem there exists $\vec{w} \in \mathbb{Z}_p^m$ such that $\vec{x} = M\vec{w}$ (i.e., $[\vec{x}]_1 \in \mathcal{L}_{[M]_1}$) if and only if $\mathsf{rank}(M) = \mathsf{rank}(M \mid \vec{x})$. Turns out a similar test can be used even when given only $[x]_1$ and M, but some extra care needs to be taken to compute $\mathsf{rank}(M \mid \vec{x})$. Firstly, consider a submatrix $A = (M' \mid x') \in \mathbb{Z}_p^{d \times d}$ of $(M \mid \vec{x})$ which includes the last column $\vec{x}$. By using Laplace expansion, we are able to compute $[\det(A)]_1 = \sum_{i=1}^{d} (-1)^{i+d} [x'_i]_1 D_{i,d}$ where $D_{i,d}$ is a determinant of the submatrix that we get by removing i-th row and d-th column from A. We still do not know $\det(A)$, but by comparing $[\det(A)]_1$ to $[0]_1$, we can tell if A is a singular or a non-singular matrix. Considering that rank of a matrix is the largest order of any of its non-zero minors, we obtain the algorithm in fig. 5 for deciding elements of $\mathcal{L}_{[M]_1}$.

In more detail, we first compute rank r of M, which can be done efficiently. The rank of $(M \mid \vec{x})$ can be at most $r + 1$ since it includes only one extra column. To test this, we iterate over all the $(r + 1) \times (r + 1)$ submatrices A of $(M \mid \vec{x})$ that contain the $\vec{x}$ column and compute $[\det(A)]_1$. If one of the determinants is non-zero, then $\mathsf{rank}(M \mid \vec{x}) = r + 1$ and it follows that $[\vec{x}]_1 \notin \mathcal{L}_{[M]_1}$. Otherwise, $\mathsf{rank}(M \mid \vec{x}) = r = \mathsf{rank}(M)$ and $[\vec{x}]_1 \in \mathcal{L}_{[M]_1}$. In order for $\mathcal{D}_{\mathcal{L}_{[M]_1}}$ to be efficient, we assume that n and m are small constants.

[14] The proof of [47] contains 1 group element and bypasses the [39] impossibility as well. This is not contradictory because the [39] impossibility only applies to pairing-based NIZKs that are compiled from NILPs

[15] Here, $\mathbb{G}_1$ is an additive pairing group and $[x]_1$ denotes a group element with a discrete logarithm x.

$\mathcal{D}_{\mathcal{L}_{[M]_1}}(M, [\vec{x}]_1)$

$r \leftarrow \mathsf{rank}(M)$;

for $A = (M' \mid \vec{x}') \in \mathbb{Z}_p^{(r+1)\times(r+1)}$ submatrix of $(M \mid \vec{x})$

$[\det(A)]_1 \leftarrow \sum_{i=1}^{d} (-1)^{i+d} [x'_i]_1 D_{i,d}$;

if $[\det(A)]_1 \neq [0]_1$: **return false**;

return true;

Fig. 5: Efficient decision algorithm for $\mathcal{L}_{[M]_1}$, given access to trapdoor M

As we saw above, the linear subspace language has a trapdoor M which allows to efficiently recognize language elements and this sufficient to avoid the [32] impossibility.

We now generalize this observation by defining a trapdoor language.

Definition 7. *Let $\mathcal{D}(1^\lambda)$ be an efficiently sampleable distribution that outputs $(\mathsf{lpar}, \mathsf{td})$ and each lpar is associated with a language $\mathcal{L}_{\mathsf{lpar}}$. A family of languages $\{\mathcal{L}_{\mathsf{lpar}}\}_{(\mathsf{lpar},\mathsf{td})\in\mathcal{D}(1^\lambda),\lambda\in\mathbb{N}}$ are trapdoor languages if there exists a PPT decider $\mathcal{M}$ such that for all $\lambda \in \mathbb{N}$ and all $(\mathsf{lpar}, \mathsf{td}) \in \mathcal{D}(1^\lambda)$,*

$$\mathsf{x} \in \mathcal{L} \Leftrightarrow \mathcal{M}(1^\lambda, \mathsf{lpar}, \mathsf{td}, \mathsf{x}) = 1.$$

The security definitions from section 2.2 for non-interactive arguments slightly change in that the Setup takes the language parameter instead of index as input and outputs a CRS. The soundness definition in general is not efficiently falsifiable because checking $\mathsf{x} \notin \mathcal{L}$ is usually not efficient. However, with trapdoor languages it is falsifiable since $\mathcal{M}$ is efficient. In particular, a tautological assumption "Π is sound" becomes a falsifiable assumption.

Examples of useful trapdoor languages. We are interested in "hard" trapdoor languages, i.e., trapdoor languages that are hard to decide without knowledge of td. We illustrate a few examples below.

- *Linear subspace language.* Firstly, let us observe that the linear subspace languages fits into the trapdoor language definition. We let $\mathcal{D}(1^\lambda)$ pick a pairing description bp and sample a matrix M according to some distribution. $\mathcal{D}(1^\lambda)$ outputs $\mathsf{lpar} = (\mathsf{bp}, [M]_1)$ and $\mathsf{td} = M$. Deciding if $\mathsf{x} \in \mathcal{L}_{[M]_1}$ can be decided efficiently given td as we argued before. For many distributions of M, $\mathcal{L}_{[M]_1}$ is considered to be a hard language on average. For example, if $M = (1, x)^\top$ and $x, \mathsf{w} \leftarrow\!\!\$\ \mathbb{Z}_p$, then $[M\mathsf{w}]_1 = (\mathsf{w}, \mathsf{w}x)$, which is indistinguishable from a random tuple $[u, v]_1^\top \leftarrow\!\!\$\ \mathbb{G}_1^2$ under the decisional Diffie-Hellman assumption. More generally, hardness of such distributions is characterized by the matrix decisional Diffie-Hellman (MDDH) assumption [26].
- *Statements about encrypted values.* Many statements about ciphertexts can be naturally formalized as a trapdoor language by using the public key as lpar and the secret key as td. Consider the following example.
  Let $(\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ be a public key cryptosystem for encrypting ℓ-bit messages and let $C : \{0,1\}^\ell \to \{0,1\}$ be an efficiently computable boolean circuit. We set $\mathcal{D}(1^\lambda) = \mathsf{KGen}(1^\lambda)$, that is $\mathsf{lpar} = \mathsf{pk}$ is the public key and $\mathsf{td} = \mathsf{sk}$ is the corresponding secret. We define the language as $\mathcal{L}^C_{\mathsf{pk}} = \{c \mid C(\mathsf{Dec}(\mathsf{sk}, c)) = 1\}$. In other words, $\mathcal{L}^C_{\mathsf{pk}}$ contains ciphertexts that encrypt a message m which satisfy some property characterized by the circuit C. For example, in range proofs we have C which checks that $k_1 \leq m \leq k_2$ for some constants k_1 and k_2. Clearly, $\mathcal{L}^C_{\mathsf{pk}}$ is a trapdoor language since given sk it is possible to decrypt c and efficiently check that the plaintext satisfies C.
- *Shuffle.* Popular ciphertext-based language that fits into the trapdoor language mould is the ciphertext shuffle. We set $\mathcal{D} = \mathsf{KGen}$. Let Σ_n be the set of permutations on n elements. The shuffle language for n

ciphertexts is

$$\mathcal{L}_{\mathsf{pk}}^{n\text{-}shuf} = \{\big((c_1,\ldots,c_n),(c'_1,\ldots,c'_n)\big) \mid \exists\sigma \in \Sigma_n\ \forall i \in \{1,\ldots,n\} : \mathsf{Dec}(\mathsf{sk},c_i) = \mathsf{Dec}(\mathsf{sk},c'_{\sigma(i)})\}$$

With the secret key as the trapdoor, statements are easy to verify since one can decrypt both ciphertext vectors, sort the resulting plaintext vectors, and then check their equality.
Shuffle proofs are often used to prove correct behaviour of mix-networks, which have for instance found application in e-voting systems to anonymize ciphertexts of voters [59].

– *Set membership.* Let us consider a public set S and a public key cryptosystem. A set membership language for S is defined as $\mathcal{L}_{\mathsf{pk}}^{S} = \{c : \mathsf{Dec}(\mathsf{sk},c) \in S\}$. This is clearly a trapdoor language where again sk plays the role of a trapdoor. González and Ráfols [36] show that an argument for this language (and its aggregated version for multiple ciphertexts) can be used to obtain, for example, shuffle arguments and range arguments. Of course, there are also more direct application like showing that c encrypts a valid candidate in an e-voting system, where S is the set of all candidates.

We note that trapdoor languages are interesting, arise frequently in practice and GW impossibility does not apply. Can we construct SNARGs from falsifiable assumptions for trapdoor languages? We leave resolving this as an interesting open question, and believe that our formalization is a first step in identifying the middle ground where GW does not apply but the language remains interesting.

6.2 A Complete Picture

In table 1, we give an overview of the impossibility results for non-interactive arguments, and positive results known under various relaxations. As already highlighted above, there are two major impossibility results. Firstly, there is no adaptively sound succinct argument for all non-deterministic computations with a black-box reduction to a falsifiable assumption [32]. This holds even with a designated verifier (a verifier that holds a private verification key). Secondly, there is no statistical zero-knowledge argument (succinct or not) for all non-deterministic computations with a black-box reduction to a falsifiable assumption [56]. Although not mentioned in the original paper, this impossibility result also extends to the designated verifier. [16]

On the other hand, by relaxing some of the requirements, it is possible to achieve succinct arguments and also statistical zero-knowledge arguments. Delegation schemes are adaptively sound succinct arguments for deterministic computation and they are achievable under falsifiable pairing-based and lattice-based assumptions as was shown by [22, 37, 43]. Recently Lipmaa and Pavlyk [52] showed that non-adaptivity is another possible relaxation. They construct a non-adaptively sound SNARG for non-deterministic computation that has perfect zero-knowledge based on a new, but falsifiable assumption. A non-adaptively sound SNARG under a falsifiable assumption was known even prior to [52]. Namely, Sahai and Waters [58] constructed a succinct perfect NIZK argument with non-adaptive soundness from iO. Subsequent to their work, constructions of iO have been proposed which are secure under falsifiable assumption, e.g. [62]. We give a brief overview of this SNARG construction in appendix G.

Acknowledgements. We thank the reviewers of CRYPTO 2022 for constructive feedback, in particular, for pointing out the connection between blackbox extractability and leakage-resilient cryptography. We thank Helger Lipmaa for comments on the paper.

References

1. Agrawal, S., Dodis, Y., Vaikuntanathan, V., Wichs, D.: On continual leakage of discrete log representations. In: ASIACRYPT 2013, Part II. LNCS, vol. 8270, pp. 401–420

[16] As can be seen in appendix F, neither the inefficient soundness adversary $\mathcal{A}_{slow}$ nor its emulator $\mathcal{A}_{fast}$ need to run the verifier internally and thus the same impossibility proof applies for the designated verifier setting.

Table 1: (Im)possibility results for non-interactive arguments under falsifiable assumptions. BB stands for black-box.

adaptive soundness	public verifier	succinct argument	language class	statistical ZK	notes/citation
+	+/−	+	**NP**	+/−	No BB reduction [32]
+	+/−	+/−	**NP**	+	No BB reduction [56]
−	+	+	**NP**	+	[52] and [58]
+	+	+	**P**	trivial	[22, 37, 43]
+ (quasi-adaptive)	+	+	linear subspace	+	[47]
−	+	+	batch **NP**	−	[21]
+	+	+	non-deterministic bounded space	−	[44]

2. Allender, E.W.: The complexity of sparse sets in p. In: Structure in Complexity Theory, pp. 1–11
3. Alwen, J., Dodis, Y., Wichs, D.: Leakage-resilient public-key cryptography in the bounded-retrieval model. In: CRYPTO 2009. LNCS, vol. 5677, pp. 36–54
4. Badrinarayanan, S., Kalai, Y.T., Khurana, D., Sahai, A., Wichs, D.: Succinct delegation for low-space non-deterministic computation. In: 50th ACM STOC, pp. 709–721
5. Baghery, K., Kohlweiss, M., Siim, J., Volkhov, M.: Another look at extraction and randomization of groth's zk-snark. In: Financial Cryptography and Data Security, pp. 457–475
6. Baghery, K., Sedaghat, M.: Tiramisu: black-box simulation extractable nizks in the updatable crs model. In: International Conference on Cryptology and Network Security, pp. 531–551
7. Barta, O., Ishai, Y., Ostrovsky, R., Wu, D.J.: On succinct arguments and witness encryption from groups. In: CRYPTO 2020, Part I. LNCS, vol. 12170, pp. 776–806
8. Ben-Or, M., Goldreich, O., Goldwasser, S., Håstad, J., Kilian, J., Micali, S., Rogaway, P.: Everything provable is provable in zero-knowledge. In: CRYPTO'88. LNCS, vol. 403, pp. 37–56
9. Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., Virza, M.: Zerocash: Decentralized anonymous payments from bitcoin. In: 2014 IEEE Symposium on Security and Privacy, pp. 459–474
10. Ben-Sasson, E., Chiesa, A., Genkin, D., Tromer, E., Virza, M.: SNARKs for C: Verifying program executions succinctly and in zero knowledge. In: CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 90–108
11. Ben-Sasson, E., Chiesa, A., Tromer, E., Virza, M.: Succinct non-interactive zero knowledge for a von neumann architecture. In: USENIX Security 2014, pp. 781–796
12. Bitansky, N., Canetti, R., Chiesa, A., Tromer, E.: Recursive composition and bootstrapping for SNARKS and proof-carrying data. In: 45th ACM STOC, pp. 111–120
13. Bitansky, N., Chiesa, A., Ishai, Y., Ostrovsky, R., Paneth, O.: Succinct non-interactive arguments via linear interactive proofs. In: TCC 2013. LNCS, vol. 7785, pp. 315–333
14. Brassard, G., Chaum, D., Crépeau, C.: Minimum disclosure proofs of knowledge. Journal of computer and system sciences **37**(2) (1988) pp. 156–189
15. Bünz, B., Bootle, J., Boneh, D., Poelstra, A., Wuille, P., Maxwell, G.: Bulletproofs: Short proofs for confidential transactions and more. In: 2018 IEEE Symposium on Security and Privacy, pp. 315–334
16. Campanelli, M., Faonio, A., Fiore, D., Querol, A., Rodríguez, H.: Lunar: a toolbox for more efficient universal and updatable zkSNARKs and commit-and-prove extensions. Cryptology ePrint Archive, Report 2020/1069 (2020) `https://eprint.iacr.org/2020/1069`.
17. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: 42nd FOCS, pp. 136–145
18. Chase, M., Kohlweiss, M., Lysyanskaya, A., Meiklejohn, S.: Succinct malleable NIZKs and an application to compact shuffles. In: TCC 2013. LNCS, vol. 7785, pp. 100–119
19. Chiesa, A., Hu, Y., Maller, M., Mishra, P., Vesely, N., Ward, N.P.: Marlin: Preprocessing zkSNARKs with universal and updatable SRS. In: EUROCRYPT 2020, Part I. LNCS, vol. 12105, pp. 738–768
20. Chor, B., Goldreich, O., Kushilevitz, E., Sudan, M.: Private information retrieval. In: 36th FOCS, pp. 41–50
21. Choudhuri, A.R., Jain, A., Jin, Z.: Non-interactive batch arguments for NP from standard assumptions. In: CRYPTO 2021, Part IV. LNCS, vol. 12828, pp. 394–423

22. Choudhuri, A.R., Jain, A., Jin, Z.: Snargs for $\mathcal{P}$ from lwe. Cryptology ePrint Archive, Report 2021/808 (2021) https://ia.cr/2021/808.
23. Costello, C., Fournet, C., Howell, J., Kohlweiss, M., Kreuter, B., Naehrig, M., Parno, B., Zahur, S.: Geppetto: Versatile verifiable computation. In: 2015 IEEE Symposium on Security and Privacy, pp. 253–270
24. Couteau, G., Hartmann, D.: Shorter non-interactive zero-knowledge arguments and ZAPs for algebraic languages. In: CRYPTO 2020, Part III. LNCS, vol. 12172, pp. 768–798
25. Damgård, I.: Towards practical public key systems secure against chosen ciphertext attacks. In: CRYPTO'91. LNCS, vol. 576, pp. 445–456
26. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.: An algebraic framework for Diffie-Hellman assumptions. In: CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 129–147
27. Fiat, A., Shamir, A.: How to prove yourself: Practical solutions to identification and signature problems. In: CRYPTO'86. LNCS, vol. 263, pp. 186–194
28. Fortnow, L.: The complexity of perfect zero-knowledge (extended abstract). In: 19th ACM STOC, pp. 204–209
29. Gabizon, A., Williamson, Z.J., Ciobotaru, O.: PLONK: Permutations over lagrange-bases for oecumenical non-interactive arguments of knowledge. Cryptology ePrint Archive, Report 2019/953 (2019) https://eprint.iacr.org/2019/953.
30. Ganesh, C., Kondi, Y., Orlandi, C., Pancholi, M., Takahashi, A., Tschudi, D.: Witness-succinct universally-composable snarks. Cryptology ePrint Archive (2022)
31. Gennaro, R., Gentry, C., Parno, B., Raykova, M.: Quadratic span programs and succinct NIZKs without PCPs. In: EUROCRYPT 2013. LNCS, vol. 7881, pp. 626–645
32. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: 43rd ACM STOC, pp. 99–108
33. Goldreich, O., Håstad, J.: On the complexity of interactive proofs with bounded communication. Inf. Process. Lett. **67**(4) (1998) pp. 205–214
34. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity and a methodology of cryptographic protocol design (extended abstract). In: 27th FOCS, pp. 174–187
35. Goldreich, O., Vadhan, S., Wigderson, A.: On interactive proofs with a laconic prover. Computational Complexity **11**(1) (2002) pp. 1–53
36. González, A., Ràfols, C.: New techniques for non-interactive shuffle and range arguments. In: ACNS 16. LNCS, vol. 9696, pp. 427–444
37. González, A., Zacharakis, A.: Fully-succinct publicly verifiable delegation from constant-size assumptions. Cryptology ePrint Archive, Report 2021/353 (2021) https://eprint.iacr.org/2021/353.
38. Groth, J.: Short pairing-based non-interactive zero-knowledge arguments. In: ASIACRYPT 2010. LNCS, vol. 6477, pp. 321–340
39. Groth, J.: On the size of pairing-based non-interactive arguments. In: EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 305–326
40. Hada, S., Tanaka, T.: On the existence of 3-round zero-knowledge protocols. In: CRYPTO'98. LNCS, vol. 1462, pp. 408–423
41. HÅstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM J. Comput. **28**(4) (1999) p. 1364–1396
42. Jutla, C.S., Roy, A.: Shorter quasi-adaptive NIZK proofs for linear subspaces. In: ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 1–20
43. Kalai, Y.T., Paneth, O., Yang, L.: How to delegate computations publicly. In: 51st ACM STOC, pp. 1115–1124
44. Kalai, Y.T., Vaikuntanathan, V., Zhang, R.Y.: Somewhere statistical soundness, post-quantum security, and SNARGs. Cryptology ePrint Archive, Report 2021/788 (2021) https://eprint.iacr.org/2021/788.
45. Kerber, T., Kiayias, A., Kohlweiss, M.: Composition with knowledge assumptions. In: CRYPTO 2021, Part IV. LNCS, vol. 12828, pp. 364–393
46. Kilian, J.: A note on efficient zero-knowledge proofs and arguments. In: Proceedings of the twenty-fourth annual ACM symposium on Theory of computing, pp. 723–732
47. Kiltz, E., Wee, H.: Quasi-adaptive NIZK for linear subspaces revisited. In: EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 101–128
48. Kosba, A., Zhao, Z., Miller, A., Qian, Y., Chan, H., Papamanthou, C., Pass, R., abhi shelat, Shi, E.: CØCØ: A framework for building composable zero-knowledge proofs. Cryptology ePrint Archive, Report 2015/1093 (2015) https://ia.cr/2015/1093.
49. Libert, B., Peters, T., Joye, M., Yung, M.: Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In: EUROCRYPT 2014. LNCS, vol. 8441, pp. 514–532

50. Lipmaa, H.: Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In: TCC 2012. LNCS, vol. 7194, pp. 169–189
51. Lipmaa, H.: Succinct non-interactive zero knowledge arguments from span programs and linear error-correcting codes. In: ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 41–60
52. Lipmaa, H., Pavlyk, K.: Gentry-wichs is tight: a falsifiable non-adaptively sound snarg. In: Advances in Cryptology – ASIACRYPT 2021, pp. 34–64
53. Micali, S.: CS proofs. In: Proceedings 35th Annual Symposium on Foundations of Computer Science, pp. 436–453
54. Naor, M.: On cryptographic assumptions and challenges (invited talk). In: CRYPTO 2003. LNCS, vol. 2729, pp. 96–109
55. Parno, B., Howell, J., Gentry, C., Raykova, M.: Pinocchio: Nearly practical verifiable computation. In: 2013 IEEE Symposium on Security and Privacy, pp. 238–252
56. Pass, R.: Unprovable security of perfect NIZK and non-interactive non-malleable commitments. In: TCC 2013. LNCS, vol. 7785, pp. 334–354
57. Ràfols, C., Zapico, A.: An algebraic framework for universal and updatable SNARKs. In: CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 774–804
58. Sahai, A., Waters, B.: How to use indistinguishability obfuscation: deniable encryption, and more. In: 46th ACM STOC, pp. 475–484
59. Sako, K., Kilian, J.: Receipt-free mix-type voting scheme - a practical solution to the implementation of a voting booth. In: EUROCRYPT'95. LNCS, vol. 921, pp. 393–403
60. Waters, B., Wu, D.J.: Batch arguments for np and more from standard bilinear group assumptions. Cryptology ePrint Archive, Paper 2022/336 (2022) `https://eprint.iacr.org/2022/336`.
61. Wee, H.: On round-efficient argument systems. In: ICALP 2005. LNCS, vol. 3580, pp. 140–152
62. Wee, H., Wichs, D.: Candidate obfuscation via oblivious LWE sampling. In: EUROCRYPT 2021, Part III. LNCS, vol. 12698, pp. 127–156

A Further Preliminaries

A.1 Argument's Security

We say an argument system satisfies *non-adaptive* soundness for a relation $\mathcal{R}$ if for any PPT adversary $\mathcal{A} = (\mathcal{A}_{\text{inp}}, \mathcal{A}_{\text{prf}})$,

$$\Pr\left[\begin{matrix}(\mathsf{x},\mathsf{st}) \leftarrow \mathcal{A}_{\text{inp}}(1^\lambda) \\ (\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda), \pi \leftarrow \mathcal{A}_{\text{prf}}(\mathsf{st},\mathsf{crs})\end{matrix} : \begin{matrix}\mathsf{V}(\mathsf{crs},\mathsf{x},\pi)=1 \\ \wedge\ \forall \mathsf{w}(\mathsf{x},\mathsf{w}) \notin \mathcal{R}\end{matrix}\right] = \mathsf{negl}(\lambda).$$

We say that an argument system for a relation $\mathcal{R}$ satisfies computational zero-knowledge if there exists a PPT simulator Sim, such that for any PPT $\mathcal{A}$, $|\varepsilon_0 - \varepsilon_1| = \mathsf{negl}(\lambda)$, where

$$\varepsilon_b := \Pr\left[(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{Setup}(1^\lambda) \ : \ \mathcal{A}^{\mathsf{O}_b(\mathsf{crs},\mathsf{td},\cdot)}(\mathsf{crs}) = 1\right] \ .$$

The oracle O_b takes an input $(\mathsf{crs},\mathsf{td},\mathsf{x},\mathsf{w})$. It returns $\perp$, when $(\mathsf{x},\mathsf{w}) \notin \mathcal{R}$. Otherwise, if $b=0$ it returns $\pi \leftarrow \mathsf{P}(\mathsf{crs},\mathsf{x},\mathsf{w})$ and if $b=1$ it returns $\pi \leftarrow \mathsf{Sim}(\mathsf{crs},\mathsf{td},\mathsf{x})$.

A.2 Fully Homomorphic Encryption (FHE)

A FHE scheme consists of a tuple of algorithms $(\mathsf{KG},\mathsf{Enc},\mathsf{Dec},\mathsf{Eval})$ with the following syntax:

$\mathsf{KG}(1^\lambda) \to (\mathsf{pk},\mathsf{sk})$**:** generates a key pair (the algorithm is randomized).

$\mathsf{Enc}(\mathsf{pk},m) \to \mathsf{ct}$**:** produces a ciphertext corresponding to a message m through the public key (the algorithm is randomized).

$\mathsf{Dec}(\mathsf{sk},\mathsf{ct}) \to m$**:** decrypts a ciphertext through the secret key (the algorithm is deterministic).

$\mathsf{Eval}(\mathsf{pk},\mathsf{ct}_m,F) \to \mathsf{ct}_F$**:** produces an encryption of $F(m)$ from an encryption of m through the public key. Occasionally we will overload this notation for functions with arity higher than 1 or that take as input plaintexts, which can be seen as dummy ciphertexts (the algorithm is deterministic).

Occasionally, we need to explicitly write $\mathsf{Enc}(\mathsf{pk}, m; r)$, where $r \leftarrow\!\!\$\ \mathsf{Rnd}$ are the random coins sampled from the randomness space Rnd.

An FHE scheme should satisfy correctness and semantic security.

Correctness. For any λ, plaintext m and function F

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) = m\right] = 1$$

and

$$\Pr\left[\mathsf{Dec}(\mathsf{sk}, \mathsf{Eval}(\mathsf{pk}, \mathsf{ct}, F)) = F(m)\right] = 1$$

where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda)$ and $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, m)$.

Semantic security. For all λ, for any PPT adversary $\mathcal{A} = (\mathcal{A}^1, \mathcal{A}^2)$

$$\Pr\left[\begin{matrix}(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}(1^\lambda), (\mathsf{st}, m_0, m_1) \leftarrow \mathcal{A}^1(\mathsf{pk}) \\ b \leftarrow\!\!\$\ \{0,1\}, \mathsf{ct} \leftarrow \mathsf{Enc}(m_b), b' \leftarrow \mathcal{A}^2(\mathsf{st}, \mathsf{ct})\end{matrix} \ :\ b = b'\right] = \mathsf{negl}(\lambda)$$

A.3 Universal Hash Functions

We say that a family of functions $\mathcal{H} = \{\mathsf{H}_{\mathsf{hk}} : M \to Y\}_{\mathsf{hk} \in K}$ for $|M| > |Y|$ is a universal hash function (UHF) family if for all distinct $m_1, m_2 \in M$,

$$\Pr\left[\mathsf{H}_{\mathsf{hk}}(m_1) = \mathsf{H}_{\mathsf{hk}}(m_2) \ :\ \mathsf{hk} \leftarrow\!\!\$\ K\right] = 1/|Y| \ .$$

Therefore, UHF must only satisfy non-adaptive collision resistance compared to a standard cryptographic hash function. More precisely, in a cryptographic hash function, the adversary first gets a key hk and then has to find distinct m_1, m_2 that produce a collision. In the case of UHF, m_1, m_2 are fixed beforehand, and then hk is picked independently of m_1 and m_2. Although satisfying a weaker property, UHFs are very efficient to compute and information-theoretically secure.

Consider, for example, the following construction. Let $\mathbb{F}$ be a prime order finite field. For some $n \geq 0$, we define $M = K = \mathbb{F}^n$ and $Y = \mathbb{F}$. For $\mathsf{hk} = (k_1, \ldots, k_n) \leftarrow\!\!\$\ \mathbb{F}^n$ the hash function is defined as $\mathsf{H}_{\mathsf{hk}}(m_1, \ldots, m_n) = \sum_{i=1}^n k_i m_i$. Let us confirm that this is indeed a UHF. Suppose for distinct $\vec{a}, \vec{b} \in \mathbb{F}^n$, $\mathsf{H}_{\mathsf{hk}}(\vec{a}) = \mathsf{H}_{\mathsf{hk}}(\vec{b})$. Thus, $\mathsf{H}_{\mathsf{hk}}(\vec{a}) - \mathsf{H}_{\mathsf{hk}}(\vec{b}) = \sum_{i=1}^n k_i(a_i - b_i) = 0$. Since at least for some j, $a_j - b_j \neq 0$, there are $|\mathbb{F}|^{n-1}$ keys that satisfy this equation. Thus, the probability of collision is $|\mathbb{F}|^{n-1}/|\mathbb{F}|^n = 1/|\mathbb{F}|$. This probability will be negligible by picking a suitably large field $\mathbb{F}$.

One can also compress the key size in the previous construction by defining the key space to be $K = \mathbb{F}$ and $\mathsf{H}_{\mathsf{hk}}(\vec{m}) = \sum_{i=1}^n m_i k^{i-1}$, where $k \leftarrow\!\!\$\ K = \mathbb{F}$. Since, $\mathsf{H}_{\mathsf{hk}}(\vec{a}) - \mathsf{H}_{\mathsf{hk}}(\vec{b}) = \sum_{i=1}^n (a_i - b_i)k^{i-1} = 0$ if k is a root of a non-zero at most degree $n-1$ polynomial, it follows that the collision probability is bounded by $(n-1)/|\mathbb{F}|$. This is not formally a UHF anymore (collision probability is not $1/|Y|$), but still serves our purposes.

A.4 Construction of CLR-OWF

Agrawal et al. [1] propose and prove the security of the following CLR-OWF. $\mathsf{KGen}(1^\lambda)$ picks a discrete logarithm secure group $\mathbb{G}$ of order p with a generator g. It samples $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n) \leftarrow\!\!\$\ \mathbb{Z}_p^n$ and sets $g_i \leftarrow g^{\alpha_i}$ for $i = 1, \ldots, n$. The public parameter is $\mathsf{pp} = (\mathbb{G}, g, g_1, \ldots, g_n)$ and the update key is $\mathsf{uk} = \vec{\alpha}$. The sampling algorithm $\mathsf{Sample}(\mathsf{pp})$ outputs $\vec{x} \leftarrow\!\!\$\ \mathbb{Z}_p^n$. $\mathsf{Eval}(\mathsf{pp}, \vec{x})$ returns $y \leftarrow \prod_{i=1}^n g_i^{x_i}$. $\mathsf{Update}(\mathsf{uk}, \vec{x})$ chooses a random vector $\vec{\beta}$ that is orthogonal to $\vec{\alpha}$ and returns $\vec{x}' \leftarrow \vec{x} + \vec{\beta}$.

The correctness holds since $\prod_{i=1}^n g_i^{x_i'} = g^{\sum_{i=1}^n \alpha_i x_i + \sum_{i=1}^n \alpha_i \beta_i} = g^{\sum_{i=1}^n \alpha_i x_i} = \prod_{i=1}^n g_i^{x_i}$.

Theorem 7 ([1]). *If the discrete logarithm assumption holds in group $\mathbb{G}$, then there exists a L-CLR-OWF in the floppy model, with $L(\lambda) < (n-2)\log p - \omega(\log \lambda)$.*

B Impossibility of Adaptive BB Extraction Using LR-OWF

The impossibility shown in theorem 1 can be interpreted as a consequence of leakage-resilience; a SNARK proof is leakage on the witness. For an **NP**-relation that is leakage-resilient, recovering the entire witness is impossible for an extractor even given the leakage, if this leakage is small.

Definition 8 ((ℓ,ε)-LR-OWF). *A function family $\mathcal{F} = \{f_i : D_i \to R_i\}$ is (ℓ,ε)-LR one-way if:*

- *There exists efficient algorithms (i) $\mathsf{KGen}(1^\lambda)$ to sample an index i (ii) $\mathsf{Sample}(i)$ for sampling an input $x \leftarrow\!\!\$\ D_i$ (iii) $\mathsf{Eval}(i,x)$ for computing $y = f_i(x)$.*
- *For any PPT $\mathcal{A}$,*

$$\Pr\left[\begin{array}{l} i \leftarrow \mathsf{KGen}(1^\lambda), x \leftarrow \mathsf{Sample}(i), \\ y \leftarrow \mathsf{Eval}(i,x), x' \leftarrow \mathcal{A}^{\mathsf{O}_\ell(\cdot)}(i,y) \end{array} : y = \mathsf{Eval}(i,x')\right] \leq \varepsilon$$

where $\mathsf{O}_\ell(\cdot)$ is an oracle that takes as an input a leakage function $h : \{0,1\}^* \to \{0,1\}^\ell$, on which $\mathsf{O}_\ell(h)$ returns $h(x)$. Adversary can query $\mathsf{O}_\ell(\cdot)$ only once.

Let $\mathcal{F}$ be a family of (ℓ,ε)-LR OWFs. For $f \in \mathcal{F}$, consider the relation $\mathcal{R}_\mathcal{F} := \{((\mathsf{x},i),\mathsf{w}) \mid i \in \mathsf{KGen}(1^\lambda), \mathsf{w} \in \mathsf{Sample}(i), \mathsf{x} = \mathsf{Eval}(i,\mathsf{w})\}$.

Theorem 8. *A non-interactive argument system Π for $\mathcal{R}_\mathcal{F}$ with argument size at most ℓ bits, has black-box knowledge soundness error $\varepsilon_{ks} \geq 1-\varepsilon$.*

Proof. By LR one-wayness of f, we have

$$\Pr[i \leftarrow \mathsf{KGen}(1^\lambda), \mathsf{x} \leftarrow\!\!\$\ D_i, \mathsf{w} \leftarrow \mathcal{A}^{\mathsf{O}_\ell(\cdot)}(1^\lambda, \mathsf{x}) : ((\mathsf{x},i),\mathsf{w}) \in \mathcal{R}_\mathcal{F}] \leq \varepsilon.$$

Consider an argument system Π for $\mathcal{R}_\mathcal{F}$ with argument size bounded by ℓ bits. Let Ext be the black-box extractor guaranteed by Π. We construct an adversary $\mathcal{A}^{\mathsf{O}_\ell(\cdot)}$ that breaks the ℓ-leakage resilience of f. $\mathcal{A}$ receives as challenge x, picks a crs together with an extraction key td, sets $h(X) := \mathsf{P}(\mathsf{crs},\mathsf{x},X)$, and receives $\pi \leftarrow \mathsf{O}_\ell(h) = \mathsf{P}(\mathsf{crs},\mathsf{x},\mathsf{w})$. It then invokes the black-box witness extractor $\mathsf{Ext}(\mathsf{crs},\mathsf{td},\mathsf{x},\pi)$ to receive w, and returns w as preimage. Assuming perfect correctness, we have that $\mathcal{A}$ succeeds in breaking one-wayness of f with the probability that the extractor succeeds. $\Pr[\mathcal{A}\text{ succeeds }] \geq 1-\varepsilon_{ks}(\lambda)$. Thus, $\varepsilon_{ks} \geq 1-\varepsilon$. □

Since an adversary can always guess the correct leakage with probability $1/2^\ell$, all OWFs are LR with $\ell = O(\log|\mathsf{w}|)$. We therefore obtain as a corollary, that an argument for $\mathcal{R}_\mathcal{F}$ must be at least of logarithmic size.

Corollary 1. *Assuming the existence of OWFs, a SNARK with negligible black-box knowledge soundness error must have argument size at least $\Omega(\log|\mathsf{w}|)$.*

With concrete LR-OWFs even better lower bounds can be achieved. Consider for example the discrete logarithm based (C)LR-OWF from theorem 7. We obtain the following result.

Corollary 2. *If the discrete logarithm assumption holds, then there exist a LR-OWF family $\mathcal{F}$ such that any non-interactive black-box knowledge sound argument for the relation $\mathcal{R}_\mathcal{F}$ must have size $\Omega(|\mathsf{w}|)$.*

The latter also shows that black-box extractable SNARKs for all **NP** do not exist.

C Additional Proofs

C.1 Proof of Theorem 2

Theorem 9 (Restatement of theorem 2). *If $\Pi_\exists$ is a non-adaptively sound SNARG scheme for* **NP**, FHE *is a semantically secure FHE scheme,* PKE *is a sematically secure cryptosystem and* H *is a family of UHFs, then the construction in fig. 2 is a SNARK for* **FewP** *satisfying definition 4. If $\Pi_\exists$ has additionally computational zero-knowledge, then so does the resulting SNARK.*

Proof. We prove zero-knowledge in lemma 10. In the rest, we will focus on proving knowledge soundness. We use the extractor in fig. 3. The extractor can have embedded N_w, a bound on the length of witnesses, since it is non-uniform. In the remainder, we define $S(j)$, for index j, as the set of strings h for which $W[h][j] \neq \bot$ after running $\mathsf{QIdx}(\mathsf{x}, j)$. That is,

$$S(j) := \left\{h : W[h][j] \neq \bot \text{ after running } \mathsf{QIdx}(\mathsf{x}, j)\right\} \ .$$

Later in lemma 4 we show that for $h \in S(j)$ there exists w such that $\mathsf{R}(\mathsf{x}, \mathsf{w}) \wedge \mathsf{H}(\mathsf{w}) = h$ (with high probability). Therefore, $h \in S(j)$ intuitively means "the extractor holds the j-th bit of a witness w such that $\mathsf{H}(\mathsf{w}) = h$".

In order to argue black-box knowledge soundness, we should be able to successfully extract from an adversary that returns an accepting proof with noticeable probability, i.e. with probability λ^{-c} for some positive constant c[17]. We show that for this type of adversary it holds with noticeable probability that $\exists h \in \bigcap_{j=1}^{N_w} S(j)$ (this is key for extraction; see last line in extractor definition). We argue this is the case by combining two facts:

- $S(j) = S(j')$ with overwhelming probability for all j, j' (lemma 6);
- If $\Pr$ [adversary returns an accepting proof] is non-negligible then $\Pr\left[S(j) \neq \emptyset\right]$ is non-negligible (lemma 7).

If $\exists h \in \bigcap_{j=1}^{N_w} S(j)$, then the string returned by the extractor is a witness with overwhelming probability because $W[h][j]$ is a bit of a witness for the relation with overwhelming probability (by lemma 4) and because, except with negligible probability, there exists a unique w such that $\mathsf{H}_{\mathsf{hk}}(\mathsf{w}) = h$ (by lemma 8). This concludes the proof. □

The following auxiliary lemma shows that an element in the table constructed by the extractor actually captures a bit of the witness with high probability.

Lemma 4. *For any PPT adversary $\mathcal{A}$, for each $j \in \{1, \ldots, N_w\}$, for each $h \in S(j)$ (where S is defined in the proof of theorem 9) the following probability p is overwhelming:*

$$p := \Pr\left[\exists \mathsf{w} \ : \ \mathcal{R}(\mathsf{x}, \mathsf{w}) \ \wedge \mathsf{H}_{\mathsf{hk}}(\mathsf{w}) = h \ \wedge W[h][j] = \mathsf{w}_j\right] \ .$$

Proof. Consider $h \in S(j)$. Notice that the event above is implied by the event $\mathsf{R}(\mathsf{x}, \mathsf{w}) \wedge \mathsf{ct}_h = \mathsf{PKE.Enc}(\mathsf{pk}, \mathsf{H}_{\mathsf{hk}}(\mathsf{w}); r)$ $\wedge\ \mathsf{ct}_{\mathrm{bit}} = \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$. This is because the extractor in fig. 3 decrypts ct_h to recover h and sets $W[h][j]$ to the decryption of $\mathsf{ct}_{\mathrm{bit}}$. We can argue that the probability of such event is overwhelming because by definition of the extractor if $h \in S(j)$ then the adversary provided a corresponding SNARG proof for the relation $\mathcal{R}'$. Invoking correctness of the encryption schemes and soundness of the SNARG concludes the proof. □

The following auxiliary lemma observes that the probability of an adversary returning a valid proof with good probability for a CRS containing a randomly sampled index i^* should also hold when we provide them with a CRS "referring to" an arbitrary index i^*. This is useful to ensure that we can apply our extraction strategy. Otherwise we could for example conceive an adversary returning a valid proof for all indices except a few. Such an adversary would return a valid proof with high probability for a honestly generated CRS but we would not be able to extract from it.

Lemma 5. *For any PPT adversary $\mathcal{A}$, if $\Pr\left[\mathcal{A} \text{ returns an accepting proof}\right]$ in the black-box knowledge-soundness experiment (definition 4) is non-negligible then for any $i^* \in [N_w]$ the following probability is non-negligible:*

$$p_{acc}^{(i^*)} := \Pr\left[(\mathsf{crs}, \mathsf{td}) \leftarrow \overline{\mathsf{Setup}}_{i^*}(1^\lambda), (\mathsf{x}, \pi) \leftarrow \mathcal{A}(\mathsf{crs}) \ : \ \mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi) = 1\right]$$

where $\overline{\mathsf{Setup}}_{i^*}$ is defined in fig. 6.

[17] This simplifies the proof, but we can argue with minor modifications the case for an adversary returning an accepting proof with only non-negligible probability.

$\overline{\mathsf{Setup}}_{i^*}(1^\lambda)$

$(\hat{\mathsf{crs}}, \hat{\mathsf{td}}) \leftarrow \Pi_\exists.\mathsf{Setup}(1^\lambda)$

$(\mathsf{pk}_{\mathrm{FHE}}, \mathsf{sk}_{\mathrm{FHE}}) \leftarrow \mathsf{FHE.KG}(1^\lambda)$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KG}(1^\lambda)$

$\mathsf{ct}_{i^*} \leftarrow \mathsf{FHE.Enc}(\mathsf{pk}, i^*)$

$\mathsf{hk} \leftarrow\!\!\$\ \mathcal{K}_{\mathrm{UHF}}$

return $(\mathsf{crs} := (\hat{\mathsf{crs}}, \mathsf{ct}_{i^*}, \mathsf{hk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{pk}), \mathsf{td} := (\mathsf{sk}_{\mathrm{FHE}}, \mathsf{sk}, \hat{\mathsf{td}}))$

Fig. 6: Modified setup with fixed index in lemma 5.

Proof. First observe that for any adversary $\mathcal{A}$, for any $j, j' \in [N_w]$. The probabilities $p_{\mathrm{acc}}^{(j)}$ and $p_{\mathrm{acc}}^{(j')}$ must be negligibly close. If they were not then we could build an adversary breaking IND-CPA of the FHE since intuitively we could distinguish ciphertexts of j from those of j' (a formal description of this adversary would be a simpler variant of the one we build in the proof of lemma 6).

Next, we observe that we can write $p_{\mathrm{acc}}^{(\mathrm{avg})} \Pr[\mathcal{A} \text{ returns an accepting proof}]$ in the black-box knowledge-soundness experiment (definition 4) as a function of $p_{\mathrm{acc}}^{(i^*)}$ for $i^* = 1, \ldots, N_w$ through a simple marginalization and bound it as follows

$$p_{\mathrm{acc}}^{(\mathrm{avg})} = \frac{1}{|N_w|} \sum_{i^* \in |N_w|} p_{\mathrm{acc}}^{(i^*)} \leq \min_{i^* \in |N_w|} p_{\mathrm{acc}}^{(i^*)} + \epsilon \ ,$$

where ϵ is a negligible. We can argue the bound by simple algebra and by applying our previous observation. As a consequence of the above, it is easy to see that, if $p_{\mathrm{acc}}^{(\mathrm{avg})}$ is non-negligible, so must be each $p_{\mathrm{acc}}^{(i^*)}$. □

Lemma 6. *For any PPT adversary $\mathcal{A}_{ksnd} = (\mathcal{A}_{inp}, \mathcal{A}_{prf})$, for all $j \neq j'$ the sets $S(j), S(j')$ are equal except with negligible probability (where S is defined in the proof of theorem 9).*

Proof. Assume by contradiction that it is not the case. We show we can break semantic security of FHE (appendix A.2) with the adversary $\mathcal{A}_{\mathrm{CPA}}$ in fig. 7.

Intuitively the adversary $\mathcal{A}_{\mathrm{CPA}}$ does the following. After receiving a public key $\mathsf{pk}_{\mathrm{FHE}}$ from the FHE challenger, it uses it to "emulate" the extractor invoking a variant of QIdx in fig. 3 ($\overline{\mathsf{QIdx}}$ in fig. 7). That is, $\mathcal{A}_{\mathrm{CPA}}$ constructs set $S(j)$ exactly as the extractor (implicitly) does, but without storing the decrypted bits in $W[h][j]$ (which it cannot due to not having the secret key). More precisely, after receiving a valid proof for index j which includes hash h^* (after decrypting ct_h with sk), $\mathcal{A}_{\mathrm{CPA}}$ will simply set $W[h^*][j]$ to a dummy "check" value ($\checkmark$ in fig. 7). This is enough to define the sets $S(j)$ which are our concern below. By hypothesis there exist with non-negligible probability indices j_0, j_1 such that $S(j_0) \neq S(j_1)$. Adversary $\mathcal{A}_{\mathrm{CPA}}^1$ finds such indices and returns j_0 and j_1 to the FHE challenger as challenge plaintexts. Once received a ciphertext $\mathsf{ct}_?$ $\mathcal{A}_{\mathrm{CPA}}^2$ will query polynomially many times $\mathcal{A}_{\mathrm{prf}}$ with a CRS that uses $\mathsf{ct}_?$ as encrypted index. Call the set of response hash ciphertexts from these queries $S_?$.

By setting N_q—the number of queries—to an appropriately high value in $\overline{\mathsf{QIdx}}$ and $\overline{\mathsf{QIdx}}'$ we can claim the following fact. By invoking lemma 9[18], except with non negligible probability, the set $S_?$ is equal to either $S(j_0)$ or $S(j_1)$. $\mathcal{A}_{\mathrm{CPA}}$ compares $S_?$ to them and outputs the bit corresponding to which one it is equal to. From this we can conclude that the advantage of $\mathcal{A}_{\mathrm{CPA}}$ in breaking semantic security is negligibly close to $\Pr[\exists j, j' : S(j) \neq S(j')]$. If the latter is non-negligible so is the advantage of $\mathcal{A}_{\mathrm{CPA}}$. Absurd. □

Lemma 7. *For any PPT adversary $\mathcal{A}$, if $\Pr[\mathcal{A} \text{ returns an accepting proof}]$ in the black-box knowledge-soundness experiment (definition 4) is non-negligible then for any $j \in [N_w]$ the probability $\Pr[S(j) \neq \emptyset]$ is non-negligible.*

[18] Which guarantees that $N_q = \mathsf{poly}(\lambda)$ is sufficient.

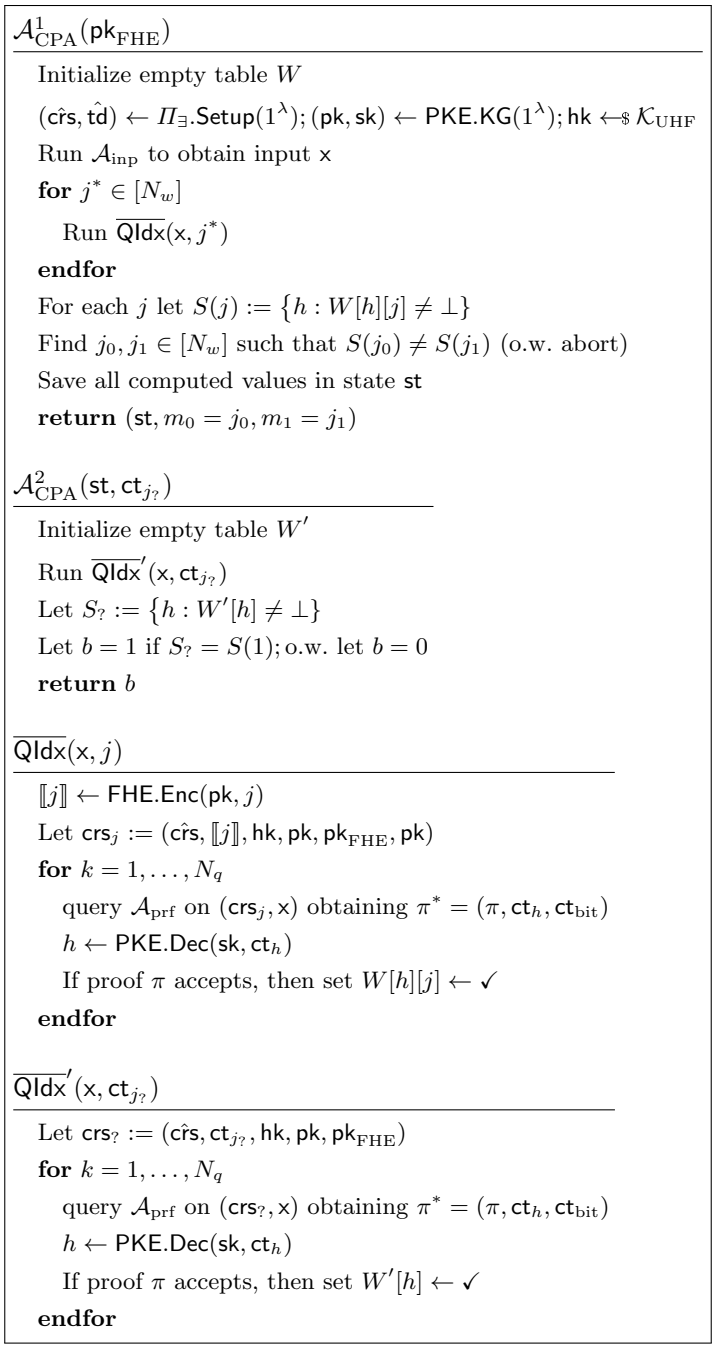

Fig. 7: IND-CPA adversary and auxiliary algorithms for proof in lemma 6.

Proof. This follows easily by the definition of set S and how QIdx works (fig. 3). In fact, by inspecting the last two lines in the loop of QIdx and the definition of S (see proof of theorem 9) we can see that $S(j)$ becomes a non-empty set as long as the adversary returns at least one accepting proof referring to index j. This can be bounded by the probability $p^{(j)}_{\mathrm{acc}}$ as defined in lemma 5. Applying that same lemma concludes the proof. □

The following fact is useful in the proofs of the lemmas above. It states that we should not expect hash collisions among witnesses.

Lemma 8. *For any PPT adversary, for all witnesses* w, w' *such that* $\mathsf{R}(\mathsf{x}, \mathsf{w})$ *and* $\mathsf{R}(\mathsf{x}, \mathsf{w}')$ *it holds that* $\Pr[\mathsf{H}_{\mathsf{hk}}(\mathsf{w}) = \mathsf{H}_{\mathsf{hk}}(\mathsf{w}')]$ *is negligible, where the probability is over the randomness of the adversary and the sampling of* hk.

Proof. The statement follows directly from the likely absence of collisions in the hash function (appendix A.3) and from the fact that the instance x is selected independently of the hash key (this is implied by non-adaptive security, i.e. definition 4).

The following lemma essentially states that the set $S(j)$ "converges" after sufficiently many queries $N_q = \mathsf{poly}(\lambda)$.

Lemma 9. *For any PPT adversary, for each index* j*, there exists constant* c *such that for all constants* $c' > c$ *the sets* $S^{(\lambda^c)}(j) = S^{(\lambda^{c'})}(j)$ *except with negligible probability, where* $S^{(t)}(j)$ *denotes the set* $S(j)$ *after* t *queries to* $\mathsf{QIdx}(\mathsf{x}, j)$.

Proof. For an adversary returning a valid proof with negligible probability, the result follows immediately. Let us then consider the case of an adversary returning a valid proof with non-negligible probability. We proceed by contradiction: assume that for any polynomial number of invocations t to $\mathsf{QIdx}(\mathsf{x}, j)$, the size of the set $S(j)$ increases with non-negligible probability after a polynomial number of steps. Call N the number of witnesses of x and recall that $N = \mathsf{poly}(|\mathsf{x}|) = \mathsf{poly}(\lambda)$ since the language is in **FewP**. Then observe that there is exists polynomial number of steps t^* after which the set S became larger than N, i.e., $|S^{(t^*)}(j)| > N$ with non-negligible probability. By our hypothesis, after a polynomial number of steps, at least one hash image will be added to $S(j)$ with non-negligible probability. By soundness of the underlying SNARG, for each of these hashes, there is a preimage that is also a witness.with non-negligible probability, implying that the number of witnesses is greater than N. Absurd. □

Lemma 10. *If* $\Pi_\exists$ *has computational zero-knowledge and* FHE *and* PKE *are semantically secure cryptosystems, then the construction in fig. 2 has computational zero-knowldege.*

Proof. Recall the computational zero-knowledge definition in appendix A.1. Let $\mathcal{A}$ be a PPT adversary, and suppose the number of queries that $\mathcal{A}$ can make is bounded by $q(\lambda)$. Since $\mathcal{A}$ runs in polynomial time, the function $q(X)$ is bounded by some polynomial. We define the simulator Sim in fig. 8. In short, Sim encrypts arbitrary values (e.g., zeros) inside ct_h and ct_{bit} and then simulates $\Pi_\exists$ proof π.

Let Game_0 be the game where the zero-knowledge oracle O_0 responds with honestly constructed proofs. In Game_1 we change the oracle to O_1, which simulates the proofs π of $\Pi_\exists$.

Next, we consider a sequence of games $\mathsf{Game}_{2:\mathsf{i}}$ for $i = 1, \ldots, q(\lambda)$. On the first i queries, in $\mathsf{Game}_{2:\mathsf{i}}$, we run O_2 which works just as O_1, but encrypts 0 in ct_h instead of the hash of the witness. Subsequent queries will still use O_1. Similarly, we will have games $\mathsf{Game}_{3:\mathsf{i}}$, which on the first i queries will run O_3 that additionally encrypts 0 in ct_{bit} and on the subsequent queries will run O_2. Oracles $\mathsf{O}_1, \mathsf{O}_2, \mathsf{O}_3$ are described in full detail in fig. 8. Observer that $\mathsf{Game}_{3:\mathsf{q}(\lambda)}$ is equivalent to the zero-knowledge game, where $\mathcal{A}$ receives a simulated proof (that is, proofs are generated by Sim) for all queries.

Let us now analyze the success probability of $\mathcal{A}$ in the previous games. Let ε_i denote $\mathcal{A}$'s success probability in Game_i. We need to show that $|\varepsilon_0 - \varepsilon_{3:q(\lambda)}| = \mathsf{negl}(\lambda)$.

$\underline{\mathsf{Game}_0 \to \mathsf{Game}_1}$. We construct a reduction $\mathcal{B}_{zk}$ for computational zero-knowledge of $\Pi_\exists$. The reduction is straightforward. $\mathcal{B}_{zk}$ has access to an oracle $\mathsf{O}_{b,\Pi_\exists}$ which returns an honest $\Pi_\exists$ proof when $b = 0$ and a simulated proof when $b = 1$. $\mathcal{B}_{zk}$ runs $\mathcal{A}$ internally and creates proofs for $\mathcal{A}$'s queries. However, when it comes to creating π, $\mathcal{B}_{zk}$ uses the oracle $\mathsf{O}_{b,\Pi_\exists}$ to create it. $\mathcal{B}_{zk}$ outputs the same value as $\mathcal{A}$. Detailed construction of $\mathcal{B}_{zk}$ can be seen in fig. 9. It follows that $|\varepsilon_0 - \varepsilon_1| = \mathsf{negl}(\lambda)$ since $\mathcal{B}_{zk}$'s advantages is negligible in the zero-knowledge game for $\Pi_\exists$.

$\underline{\mathsf{Game}_{2:\mathsf{i}-1} \to \mathsf{Game}_{2:\mathsf{i}}}$. We construct a reduction $\mathcal{B}_{sem}$ to the semantic security of PKE. Full details are described in fig. 10. Essentially, $\mathcal{B}^1_{sem}$ runs $\mathcal{A}$ for i queries and outputs $m_0 = \mathsf{H}_{\mathsf{hk}}(\mathsf{w}^{(i)})$ and $m_1 = 0$, where

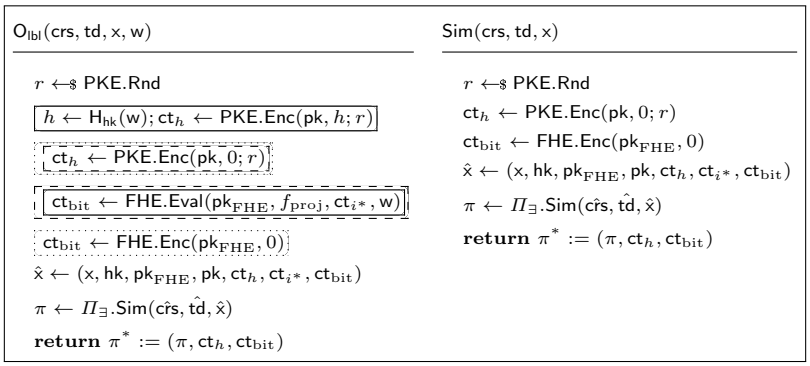

Fig. 8: Oracles $\mathsf{O}_1, \mathsf{O}_2, \mathsf{O}_3$ and the simulator Sim for lemma 10. When $\mathsf{lbl} = 1, 2, 3$, then we respectively consider only lines with $\boxed{AAA}$, $\boxed{\overline{AAA}}$, and $\boxed{\overline{\overline{AAA}}}$. Recall $f_{\mathrm{proj}}(i, \mathsf{w}) := \mathsf{w}_i$

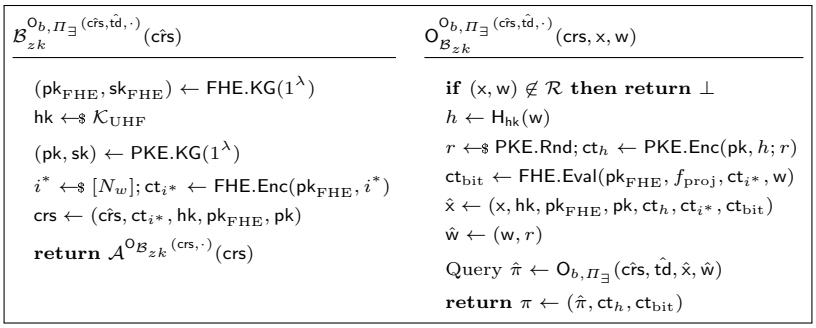

Fig. 9: $\mathsf{Game}_0 \to \mathsf{Game}_1$ reduction in lemma 10

$\mathsf{w}^{(i)}$ is the witness $\mathcal{A}$ submits on the i-th query. Then, $\mathcal{B}_{sem}^2$ gets an encryption ct, which encrypts either m_0 or m_1. $\mathcal{B}_{sem}^2$ runs internally $\mathcal{A}$. For queries $j = 1, \ldots, i-1$ adversary $\mathcal{B}_{sem}^2$ returns the answers of O_1 and for queries $j > i$ it returns the answer of O_2 to $\mathcal{A}$. However, on the i-th query it behaves as O_1 but $\mathsf{ct}_h = \mathsf{ct}$. So if ct_h encrypts $\mathsf{H}_{\mathsf{hk}}(\mathsf{w}^{(i)})$ it will be oracle O_1 and if ct_h encrypts 0 it will be the oracle O_2. $\mathcal{B}_{sem}^2$ returns the same answer as $\mathcal{A}$.

It follows from the semantic security that $|\varepsilon_{2:i-1} - \varepsilon_{2:i}| = \mathsf{negl}(\lambda)$.

$\mathsf{Game}_{3:i-1} \to \mathsf{Game}_{3:i}$. This is identical to the previous case, with the only difference being that we rely on the semantic security of FHE.

Since $q(\lambda)$ is bounded by a polynomial and at every game transition $\mathcal{A}$'s advantage increases at most by a negligible amount, then it follows that our construction has computational zero-knowledge. □

C.2 Proof of Theorem 3

Theorem 10 (Restatement of theorem 3). *Let $\Sigma = (\mathsf{KGen}, \mathsf{Sample}, \mathsf{Eval}, \mathsf{Update})$ be an L-CLR-OWF and let Π be a non-adaptive black-box $\varepsilon_{ks}(\lambda)$-knowledge sound argument for $\mathcal{R}_\Sigma$ as defined above. If the proof size is less than $L(\lambda)$ bits, then L-CLR-OWF can be broken with probability $1 - \varepsilon_{ks}(\lambda)$.*

Proof. Assume that the proof size of Π is upper bounded by $L(\lambda)$. We show that in this case it is possible to break L-continuous leakage-resilient one-wayness of Σ. Firstly, we describe an adversary $\mathcal{A} = (\mathcal{A}_{inp}, \mathcal{A}_{prf})$ for non-adaptive black-box knowledge soundness in fig. 11. This exactly follows the intuition we discussed above.

$\mathcal{B}^1_{sem}(\mathsf{pk})$

$(\mathsf{pk}_{\mathrm{FHE}}, \mathsf{sk}_{\mathrm{FHE}}) \leftarrow \mathsf{FHE.KG}(1^\lambda)$
$\mathsf{hk} \leftarrow\!\!\$\ \mathcal{K}_{\mathrm{UHF}}$
$(\hat{\mathsf{crs}}, \hat{\mathsf{td}}) \leftarrow \Pi_\exists.\mathsf{Setup}(1^\lambda)$
$i^* \leftarrow\!\!\$\ [N_w]; \mathsf{ct}_{i^*} \leftarrow \mathsf{FHE.Enc}(\mathsf{pk}_{\mathrm{FHE}}, i^*)$
$\mathsf{crs} \leftarrow (\hat{\mathsf{crs}}, \mathsf{ct}_{i^*}, \mathsf{hk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{pk})$
$\mathsf{td} \leftarrow \left(\mathsf{sk}_{\mathrm{FHE}}, \hat{\mathsf{td}}\right)$
Sample random coins r for $\mathcal{A}$
$b \leftarrow \mathcal{A}^{\mathsf{O}_2}(\mathsf{crs}; r)$
Recover $\mathcal{A}$'s i-th query $(\mathsf{x}^{(i)}, \mathsf{w}^{(i)})$
return $\left(\mathsf{st} \leftarrow (\mathsf{crs}, \mathsf{td}, r), m_0 \leftarrow \mathsf{H}_{\mathsf{hk}}(\mathsf{w}^{(i)}), m_1 \leftarrow 0\right)$

$\mathcal{B}^2_{sem}(\mathsf{st} = (\mathsf{crs}, \mathsf{td}, r), \mathsf{ct})$

return $\mathcal{A}^{\mathsf{O}_{sem}(\mathsf{st},\mathsf{ct},\cdot)}$, where
For queries $j = 1, \ldots, i-1$: $\mathsf{O}_{sem} = \mathsf{O}_2$
On i-th query $\mathsf{O}_{sem}(\mathsf{st}, \mathsf{ct}, \mathsf{x}, \mathsf{w})$:
  $\mathsf{ct}_h \leftarrow \mathsf{ct}$
  $\mathsf{ct}_{\mathrm{bit}} \leftarrow \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$
  $\hat{\mathsf{x}} \leftarrow (\mathsf{x}, \mathsf{hk}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{pk}, \mathsf{ct}_h, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}})$
  $\pi \leftarrow \Pi_\exists.\mathsf{Sim}(\hat{\mathsf{crs}}, \hat{\mathsf{td}}, \hat{\mathsf{x}})$
  return $\pi^* := (\pi, \mathsf{ct}_h, \mathsf{ct}_{\mathrm{bit}})$
For queries $j > i$: $\mathsf{O}_{sem} = \mathsf{O}_1$

Fig. 10: $\mathsf{Game}_{2:\mathsf{i}-1} \to \mathsf{Game}_{2:\mathsf{i}}$ reduction in lemma 10

$\mathcal{A}_{inp}(1^\lambda)$

$(\mathsf{pp}, \mathsf{uk}) \leftarrow \mathsf{KGen}(1^\lambda)$;
$w \leftarrow\!\!\$\ \mathsf{Sample}(\mathsf{pp})$;
$y \leftarrow\!\!\$\ \mathsf{Eval}(\mathsf{pp}, w)$;
return $(\mathsf{x} = (\mathsf{pp}, y), \mathsf{st} = (\mathsf{pp}, \mathsf{uk}, y, w))$;

$\mathcal{A}_{prf}(\mathsf{st}, \mathsf{crs})$

Parse $\mathsf{st} = (\mathsf{pp}, \mathsf{uk}, y, w)$;
$w' \leftarrow \mathsf{Update}(\mathsf{uk}, w)$;
$\pi \leftarrow \mathsf{P}(\mathsf{crs}, (\mathsf{pp}, y), w')$;
return π;

Fig. 11: Non-adaptive black-box knowledge soundness adversary for theorem 3

$\mathcal{B}^{\mathsf{O}_L(\cdot)}(\mathsf{pp}, y)$

$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{KGen}(1^\lambda)$
$\pi \leftarrow \mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\mathsf{crs})$
Run $\mathsf{Ext}^{\mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\cdot)}(\mathsf{crs}, \mathsf{td}, (\mathsf{pp}, y), \pi)$

$\mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\mathsf{crs})$

Define $h_{\mathsf{crs},y,\mathsf{pp}}(X) := \mathsf{P}(\mathsf{crs}, (\mathsf{pp}, y), X)$
Query $\pi \leftarrow \mathsf{O}_L(h_{\mathsf{crs},y,\mathsf{pp}})$
return π

Fig. 12: Continuous leakage-resilience adversary $\mathcal{B}$

Next, we construct an adversary $\mathcal{B}$ against CLR in fig. 12. Idea is that we want to use the extractor Ext of Π to recover the OWF preimage. To do so, $\mathcal{B}$ must provide Ext with proofs which are created by crs chosen by Ext. Since proofs depend on the OWF preimage w, $\mathcal{B}$ can use the leakage query oracle O_L with a function $h_{\mathsf{crs},y,\mathsf{pp}}(X) := \mathsf{P}(\mathsf{crs}, (\mathsf{pp}, y), X)$. This is possible only because the proof size is $\leq L(\lambda)$ bits. In fig. 12, $\mathcal{B}$ runs a subroutine $\mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\mathsf{crs})$ for creating proofs.

In the following, we will analyze the success probability of $\mathcal{B}$. Essentially, we show that if Ext succeeds in extracting with high probability, then also $\mathcal{B}$ will succeed in breaking L-continuous leakage-resilience of OWF with high probability.

$\underline{\mathsf{Game}_0}$. This is the original L-CLR game with the adversary $\mathcal{B}$ from section 2.1.

$\underline{\mathsf{Game}_1}$. This is the same game with $\mathcal{B}$ being in-lined. See fig. 13. Obviously the probability that $y = \mathsf{Eval}(\mathsf{pp}, x')$ is the same in both games.

$\underline{\mathsf{Game}_2}$. This game is again just a slight rewrite of the previous Game_1. Note that the first three lines of Game_1 in fig. 13 are equivalent to $\mathcal{A}_{inp}(1^\lambda)$. So we instead write $(\mathsf{x} = (\mathsf{pp}, y), \mathsf{st} = (\mathsf{pp}, \mathsf{uk}, y, w)) \leftarrow \mathcal{A}_{inp}(1^\lambda)$ in Game_2. Moreover, $\mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\mathsf{crs})$ and $\mathcal{A}_{prf}(\mathsf{st}, \mathsf{crs})$ produce the exact same proof π. We change $\mathsf{Sim}^{\mathsf{O}_L(\cdot)}_{\mathsf{pp},y}(\mathsf{crs})$ to $\mathcal{A}_{prf}(\mathsf{st}, \mathsf{crs})$ in Game_2. Clearly again the probability of $y = \mathsf{Eval}(\mathsf{pp}, x')$ is the same as before.

$\mathsf{Game}_1(1^\lambda)$

$(\mathsf{pp}, \mathsf{uk}) \leftarrow \mathsf{KGen}(1^\lambda)$

$w \leftarrow \mathsf{Sample}(\mathsf{pp})$

$y \leftarrow \mathsf{Eval}(\mathsf{pp}, w)$

$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{KGen}(1^\lambda)$

$\pi \leftarrow \mathsf{Sim}_{\mathsf{pp},y}^{\mathsf{O}_L(\cdot)}(\mathsf{crs})$

$x' \leftarrow \mathsf{Ext}^{\mathsf{Sim}_{\mathsf{pp},y}^{\mathsf{O}_L(\cdot)}(\cdot)}(\mathsf{crs}, \mathsf{td}, (\mathsf{pp}, y), \pi)$

return $y = \mathsf{Eval}(\mathsf{pp}, x')$

$\mathsf{Game}_2(1^\lambda)$

$(\mathsf{x} = (\mathsf{pp}, y), \mathsf{st} = (\mathsf{pp}, \mathsf{uk}, y, w)) \leftarrow \mathcal{A}_{inp}(1^\lambda)$

$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{KGen}(1^\lambda)$

$\pi \leftarrow \mathcal{A}_{prf}(\mathsf{st}, \mathsf{crs})$

$x' \leftarrow \mathsf{Ext}^{\mathcal{A}_{prf}(\mathsf{st},\mathsf{crs})}(\mathsf{crs}, \mathsf{td}, (\mathsf{pp}, y), \pi)$

return $y = \mathsf{Eval}(\mathsf{pp}, x')$

Fig. 13: Security games for theorem 3

Note that Game_2 with the winning condition $\mathsf{V}(\mathsf{crs}, (\mathsf{pp}, y), \pi) = 1 \wedge y \neq \mathsf{Eval}(\mathsf{pp}, x')$ is the non-adaptive black-box knowledge soundness game. We know that this probability is bounded by $\varepsilon_{ks}(\lambda)$. Thus, $\mathsf{V}(\mathsf{crs}, (\mathsf{pp}, y), \pi) \neq 1 \vee y = \mathsf{Eval}(\mathsf{pp}, x')$ happens with a probability $> 1 - \varepsilon_{ks}(\lambda)$. However, $\mathsf{V}(\mathsf{crs}, (\mathsf{pp}, y), \pi) \neq 1$ is not possible given the construction of $\mathcal{A}_{prf}$. Thus, $\mathsf{V}(\mathsf{crs}, (\mathsf{pp}, y), \pi) \neq 1 \vee y = \mathsf{Eval}(\mathsf{pp}, x')$ is equivalent to $y = \mathsf{Eval}(\mathsf{pp}, x')$, which is the Game_2 winning condition. It follows that $\mathcal{B}$ can break L-CLR with the probability $1 - \varepsilon_{ks}(\lambda)$. □

C.3 Proofs of Lemmas 2 and 3

Lemma 11 (Restatement of lemma 2). *If an indexed languages $\mathcal{L} \in \mathbf{NP}$ has a sub-exponentially hard-on-average problem, then for any $d > 0$, $\mathcal{L}$ also has a hard-on-average problem with $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishability.*

Proof. Let us recall that X_λ and $\bar{X}_\lambda$ are said to be sub-exponentially indistinguishable if there exists a $\delta > 0$ such that X_λ and $\bar{X}_\lambda$ are $(2^{\Omega(\lambda^\delta)}, 1/2^{\Omega(\lambda^\delta)})$-indistinguishable. Let us define $n(\lambda) = \lceil \lambda^{d/\delta} \rceil$. Then $Y_\lambda := X_{n(\lambda)}$ and $\bar{Y}_\lambda := \bar{X}_{n(\lambda)}$ are $(s(\lambda), \varepsilon(\lambda))$-indistinguishable, where $s(\lambda) = 2^{\Omega(n^\delta)} = 2^{\Omega(\lceil \lambda^{d/\delta} \rceil^\delta)}$ and $\varepsilon(\lambda) = 1/2^{\Omega(n^\delta)} = 1/2^{\Omega(\lceil \lambda^{d/\delta} \rceil^\delta)}$. Firstly, since the circuit of size $s(\lambda) = 2^{\Omega(\lceil \lambda^{d/\delta} \rceil^\delta)}$ grows faster than the circuit of size 2^{λ^d}, then, for a sufficiently large λ, Y_λ and $\bar{Y}_\lambda$ are also $(2^{\lambda^d}, \varepsilon(\lambda))$-indistinguishable. Conversely, Y_λ and $\bar{Y}_\lambda$ are$(2^{\lambda^d}, \varepsilon'(\lambda))$-indistinguishable if $\varepsilon'(\lambda) \geq \varepsilon(\lambda)$. This is the case for $\varepsilon'(\lambda) = 1/2^{\lambda^d}$ if λ is again sufficiently large. It follows that for a large enough λ, exists Y_λ and $\bar{Y}_\lambda$ that are $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishable.

Let i be the index sampler and $(\mathsf{Samp}_{\mathcal{L}}, \mathsf{Samp}_{\bar{\mathcal{L}}})$ instance samplers for sub-exponentially hard-on-average problem. Then we can always define $\mathcal{I}'(1^\lambda)$ as $\mathcal{I}(1^{n(\lambda)})$, $\mathsf{Samp}'_{\mathcal{L}}(1^\lambda, \mathsf{i})$ as $\mathsf{Samp}'_{\mathcal{L}}(1^{n(\lambda)}, \mathsf{i})$, and $\mathsf{Samp}_{\bar{\mathcal{L}}}(1^\lambda, \mathsf{i})$ as $\mathsf{Samp}_{\bar{\mathcal{L}}}(1^{n(\lambda)}, \mathsf{i})$, which gives the desired hard-on-average problem with $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishability. □

Lemma 12 (Restatement of lemma 3). *Let X_λ and $\bar{X}_\lambda$ be $(2^{\lambda^d}, 1/2^{\lambda^d})$-indistinguishable distributions for some integer $d \geq 2$. Let A_λ over (x, π) be an augmented distribution of X_λ, where $|\pi| = \ell(\lambda) = o(\lambda^d)$. Then there exists an augmented distribution $\bar{A}_\lambda$ of $\bar{X}_\lambda$ such that A_λ and $\bar{A}_\lambda$ are $(\mathsf{poly}(\lambda), \mathsf{negl}(\lambda))$-indistinguishable.*

Proof. Let X_λ and $\bar{X}_\lambda$ be $(s(\lambda), \varepsilon(\lambda))$-indistinguishable, where $s(\lambda) = 2^{\lambda^d}$ and $\varepsilon(\lambda) = 1/2^{\lambda^d}$. Then X_λ and $\bar{X}_\lambda$ are $(s(\lambda), \varepsilon'(\lambda))$-indistinguishable for any $\varepsilon'(\lambda) \geq 1/2^{\lambda^d}$. Let us take $\varepsilon'(\lambda) = 1/2^{\lambda^{d-1}}$. According to the leakage lemma, there exists a polynomial p and an augmented distribution $\bar{A}_\lambda$ of $\bar{X}_\lambda$ such that A_λ and $\bar{A}_\lambda$ are $(s^*(\lambda), \varepsilon^*(\lambda))$-indistinguishable where $s^*(\lambda) = s(\lambda)p(\varepsilon'(\lambda)/2^{\ell(\lambda)})$ and $\varepsilon^*(\lambda) = 2\varepsilon'(\lambda)$.

Clearly $\varepsilon^*(\lambda)$ is negligible since $\varepsilon^*(\lambda) = 2\varepsilon'(\lambda) = 2/2^{\lambda^{d-1}} = 1/2^{\lambda^{d-1}-1} = \mathsf{negl}(\lambda)$. The distinguisher circuit size $s^*(\lambda)$ is $s^*(\lambda) = s(\lambda)p(\varepsilon'(\lambda)/2^{\ell(\lambda)}) = 2^{\lambda^d} p(2^{-\lambda^{d-1} - o(\lambda^d)})$. Here, $p(2^{-\lambda^{d-1} - o(\lambda^d)}) = 2^{-o(\lambda^{d-1})}$ and therefore $s^*(\lambda) = 2^{\lambda^d - o(\lambda^{d-1})} = 2^{\Omega(\lambda^d)}$. This means that indistinguishability holds also for all polynomial size circuits, as $s^*(\lambda)$ grows faster than any polynomial. □

D Non-Adaptive BB Extractable SNARK for UP

In fig. 14 we show a slightly simpler version of the construction in fig. 2. The following constructs a SNARK for a language in **UP** (restriction of **NP** where a statement in the language has exactly one accepting witness). In this variant we do not need hashing to fingerprint the witnesses. Our SNARK works as follows.

$\mathsf{Setup}(1^\lambda)$

$(\hat{\mathsf{crs}}, \hat{\mathsf{td}}) \leftarrow \Pi_\exists.\mathsf{Setup}(1^\lambda)$

$i^* \leftarrow\!\!\$\ [N_w]$

$(\mathsf{pk}_{\mathrm{FHE}}, \mathsf{sk}_{\mathrm{FHE}}) \leftarrow \mathsf{FHE.KG}(1^\lambda)$

$\mathsf{ct}_{i^*} \leftarrow \mathsf{FHE.Enc}(\mathsf{pk}, i^*)$

return $(\mathsf{crs} := (\hat{\mathsf{crs}}, \mathsf{ct}_{i^*}, \mathsf{pk}_{\mathrm{FHE}}), \mathsf{td} := (\mathsf{sk}_{\mathrm{FHE}}, \hat{\mathsf{td}}))$

$\mathsf{P}(\mathsf{crs}, \mathsf{R}, \mathsf{x}, \mathsf{w})$

$\mathsf{ct}_{\mathrm{bit}} \leftarrow \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$
where $f_{\mathrm{proj}}(i, w) := w_i$

$\pi \leftarrow \Pi_\exists.\mathsf{P}(\hat{\mathsf{crs}}, \mathsf{R}', (\mathsf{x}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}}), \mathsf{w})$
where $\mathsf{R}'(\mathsf{x}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}}; \mathsf{w}) \iff$
$\mathsf{R}(\mathsf{x}, \mathsf{w}) \ \wedge\ \mathsf{ct}_{\mathrm{bit}} = \mathsf{FHE.Eval}(\mathsf{pk}_{\mathrm{FHE}}, f_{\mathrm{proj}}, \mathsf{ct}_{i^*}, \mathsf{w})$

return $\pi^* := (\pi, \mathsf{ct}_{\mathrm{bit}})$

$\mathsf{V}(\mathsf{crs}, \mathsf{R}, \mathsf{x}, \pi^*)$

Parse π^* as $(\pi, \mathsf{ct}_{\mathrm{bit}})$

return $\Pi_\exists.\mathsf{V}\left(\hat{\mathsf{crs}}, \mathsf{R}', (\mathsf{x}, \mathsf{pk}_{\mathrm{FHE}}, \mathsf{ct}_{i^*}, \mathsf{ct}_{\mathrm{bit}})\right)$
where R' is defined like above

Fig. 14: Non-adaptively secure black-box extractable construction for **UP**. N_w is a bound on the witness size. $\Pi_\exists$ is the SNARG scheme.

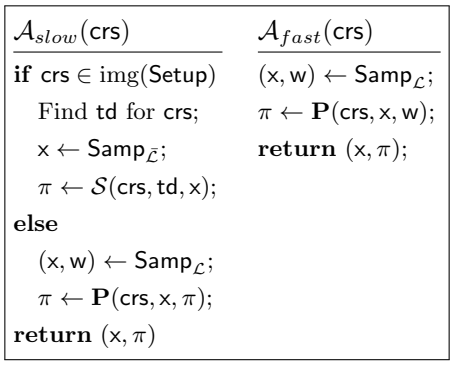

Fig. 15: Inefficient adversary $\mathcal{A}_{slow}$ against adaptive soundness and its efficient emulator $\mathcal{A}_{fast}$

E A Summary of the GW Impossibility Proof [32]

The main technique used in the proof is showing the existence of a *simulatable adversary* $\overline{\mathbf{P}}$ for any SNARG for an NP complete language. A simulatable adversary is an *inefficient* adversary that, given a CRS, outputs a false statement $\mathsf{x} \notin \mathcal{L}$ with a valid proof π for it. While the existence of such adversary is trivial for any SNARG, a simulatable adversary also comes with an *efficient* simulator $\mathcal{S}$ such that no efficient distinguisher can distinguish them. To show the existence of a simulatable adversary, a lemma in [32] shows that for any two computationally indistinguishable distributions respectively over a set $\mathcal{L}$ and its complement $\overline{\mathcal{L}} = \{0,1\}^* \backslash \mathcal{L}$, and for any leakage information π on $\mathsf{x} \in \mathcal{L}$, there exists some leakage information $\overline{\pi}$ on $\overline{\mathsf{x}} \in \overline{\mathcal{L}}$ such that (x, π) and $(\overline{\mathsf{x}}, \overline{\pi})$ are also computationally indistinguishable. What is important in this lemma is that the security degrades exponentially with the size of the leakage π and this is the reason why the underlying SNARG Π should have succinctness property.

Now, given this simulatable adversary, the result can be concluded as follows. Assume there exists a black-box reduction R that shows the soundness of π based on $(\mathcal{C}, c)$. This means that the efficient reduction $R^{\overline{\mathbf{P}}}$, given black-box access to successful adversary $\overline{\mathbf{P}}$ can break $(\mathcal{C}, c)$. But if (inefficient) $R^{\overline{\mathbf{P}}}$ can break $(\mathcal{C}, c)$, then (efficient) $R^{\mathcal{S}}$ can also break it since no efficient distinguisher (including the challenger of $(\mathcal{C}, c)$) can distinguish $\overline{\mathbf{P}}$ from $\mathcal{S}$. Thus, if this black-box reduction exists, then the assumption $(\mathcal{C}, c)$ should be false.

F A Summary of the Pass Impossibility Proof [56]

We briefly summarize the impossibility result of [56]. Let $\mathcal{L}$ be a hard-on-average language as defined in section 2. Now we can give an informal statement for the result in [56].

Theorem 11. *Let Π be an adaptively sound and perfectly zero-knowledge non-interactive argument for an hard-on-average problem $\mathcal{L}$. Suppose that there exists an efficient black-box reduction R that can reduce adaptive soundness of Π to some falsifiable assumption C. Then C can be broken in polynomial time.*

The intuition behind the result is as follows. First, we construct an *inefficient* adversary $\mathcal{A}_{slow}$ (the first algorithm in fig. 15) that can break adaptive soundness. If $\mathcal{A}_{slow}$ gets a valid CRS as an input (crs is in the image of Setup), then it brute-force computes a trapdoor td, samples a false statement x, and runs the simulator with td to produce a proof π. If the CRS is invalid (outside of the image of Setup), then it just tries to compute a proof for an honest statement x. Note that in the adaptive soundness game the CRS will always be valid and thus only the first branch will matter. Since $\mathcal{L}$ is a hard-on-average language, then false and true statements are indistinguishable, and therefore $\mathcal{S}$ will produce a proof which is accepted by a verifier with an overwhelming probability. So indeed $\mathcal{A}_{slow}$ does break adaptive soundness.

Let us suppose that there exists an efficient reduction R that given black-box access to any adaptive soundness adversary $\mathcal{A}$, can break some falsifiable assumption C. The problem is that although $\mathcal{A}_{slow}$ does

P'	V'
Constants: PRF key K	**Constants:** PRF key K
Input: (x, w)	**Input:** (x, π)
if $(\mathsf{x}, \mathsf{w}) \in \mathcal{R}$	**if** $f(\pi) = f(\mathsf{PRF}_K(\mathsf{x}))$
return $\mathsf{PRF}_K(\mathsf{x})$	**return** 1
else	**else**
return $\perp$	**return** 0

Fig. 16: Programs P' and V'

break adaptive soundness, it is not efficient. Therefore also the reduction $R^{\mathcal{A}_{slow}}$ will be inefficient. We solve this issue be constructing an efficient emulator $\mathcal{A}_{fast}$ (the second algorithm in fig. 15) for $\mathcal{A}_{slow}$.

The emulator $\mathcal{A}_{fast}$ simply samples an honest (x, w) and generates an honest proof π. Now let us compare $\mathcal{A}_{fast}$ and $\mathcal{A}_{slow}$. If CRS is invalid, then $\mathcal{A}_{slow}$ and $\mathcal{A}_{fast}$ are identical. However, if the CRS is valid, then it needs a bit more work to show that outputs are indistinguishable. Intuitively, x is indistinguishable due to the hard-on-average property and π is indistinguishable due to zero-knowledge. However, here it is important that zero-knowledge property holds even with respect to a fixed CRS since we do not know how the distinguishing adversary may pick the valid CRS. Moreover, with computational zero-knowledge it may be even possible to extract the witness from a proof π which would make distinguishing $\mathcal{A}_{slow}$ and $\mathcal{A}_{fast}$ trivial. This is the reason why zero-knowledge has to be perfect (or statistical). It follows now that outputs of $\mathcal{A}_{fast}$ and $\mathcal{A}_{slow}$ are computationally indistinguishable.

Since $R^{\mathcal{A}_{slow}}$ can break the assumption C, then so does $R^{\mathcal{A}_{fast}}$ which means that the assumption C is insecure. Hence, it is impossible to base adaptively sound perfect zero-knowledge argument on a falsifiable assumption using a black-box reduction.

G Non-Adaptive SNARGs With Perfect ZK Based on iO

We show how for the non-adaptive case, none of [56] and [32] results hold. We do so by the following observation: assuming that indistinguishability obfuscation (iO) can be build from falsifiable assumptions (see [62]), the perfect NIZK arguments of Sahai and Waters [58], instantiated with a puncturable PRF (PPRF) that satisfies *succinctness* property, is a non-adaptive SNARG with perfect ZK for all NP languages in the CRS model. While this can be seen as a feasibility result, proposing a construction with more standard assumptions (i.e., without iO) is still an interesting open question.

We now recall the NIZK arguments of Sahai and Waters [58].

NIZK arguments of Sahai and Waters. The idea is very simple: the proof system consists of two obfuscated programs put in the CRS. The first program is the proving algorithm that inputs a statement x and witness w and outputs a signature on x if $(\mathsf{x}, \mathsf{w}) \in \mathcal{R}$. The signature is realized by a PRF in the construction. The second program is the verification algorithm that is just the signature verification and verifies the proof by checking the validity of the signature on x.

Let PRF be a puncturable PRF that inputs ℓ-bit long strings and outputs λ bits (where λ is the security parameter). Let $f(\cdot)$ be a one way function. The NIZK argument $\mathit{\Pi} = (\mathsf{Setup}, \mathsf{P}, \mathsf{V})$ for language $\mathcal{L}$ with relation $\mathcal{R}$ is as follows:

- $\mathsf{Setup}(1^\lambda)$ first selects a puncturable PRF key K for PRF. Next, it creates an obfuscation of programs P' and V' as depicted in Figure 16. The CRS crs consists of the two obfuscated programs.
- $\mathsf{P}(\mathsf{crs}, \mathsf{x}, \mathsf{w})$ runs the obfuscated program P' on input (x, w) and returns the proof π if $(\mathsf{x}, \mathsf{w}) \in \mathcal{R}$.
- $\mathsf{V}(\mathsf{crs}, \mathsf{x}, \pi)$ runs the obfuscated program V' on input (x, π) and returns a bit indicating accept or reject.

Theorem 12. *[58] The argument system Π is perfectly zero-knowledge. Moreover, if the obfuscation scheme is indistingishuably secure,* PRF *is a secure punctured PRF with succinctness property, and $f(\cdot)$ is an injective one way function, then Π is a non-adaptive SNARG.*

Remark 4. While SNARGs with non-adaptive security can be seen as interactive two-message arguments by thinking of the CRS as the verifier's message, the type of non-adaptivity in the resulting argument is still "strong" in the sense that the verifier's message does not depend on the prover's (fixed) statement. One can also define a weaker notion of non-adaptivity for two-message arguments where the first message is *statement-dependent* (See [7] for example). We note that while the above iO-based construction satisfies the stronger notion, giving a construction for the weaker notion of non-adaptivity based on seemingly weaker tools is not a hard task. Namely, the verifier can use a witness encryption scheme to encrypt a succinct random value r under the prover's statement and ask the prover to return r.