# Verifiably-Extractable OWFs and Their Applications to Subversion Zero-Knowledge

Prastudy Fauzi[1], Helger Lipmaa[1,2], Janno Siim[2], Michał Zając[3], and Arne Tobias Ødegaard[1]

[1] Simula UiB, Bergen, Norway
[2] University of Tartu, Tartu, Estonia
[3] Clearmatics, London, UK

**Abstract.** An extractable one-way function (EOWF), introduced by Canetti and Dakdouk (ICALP 2008) and generalized by Bitansky et al. (SIAM Journal on Computing vol. 45), is an OWF that allows for efficient extraction of a preimage for the function. We study (generalized) EOWFs that have a public image verification algorithm. We call such OWFs verifiably-extractable and show that several previously known constructions satisfy this notion. We study how such OWFs relate to subversion zero-knowledge (Sub-ZK) NIZKs by using them to generically construct a Sub-ZK NIZK from a NIZK satisfying certain additional properties, and conversely show how to obtain them from any Sub-ZK NIZK. Prior to our work, the Sub-ZK property of NIZKs was achieved using concrete knowledge assumptions.

## 1 Introduction

Extractability is a way to formalize what an algorithm *knows*. It is a notion essential to modern cryptography which dates back to the works of Goldwasser et al. [34] who proposed *proofs of knowledge*, and later formalized for interactive proofs by Bellare and Goldreich [10].[4] For non-interactive proofs, Damgård [23] proposed knowledge-of-exponent assumptions, which are non-falsifiable assumptions[5] saying that any efficient algorithm that produces group elements that satisfy a specific relation must know their discrete logarithms.

Investigating extractable primitives, Canetti and Dakdouk [19] introduced the notion of extractable one-way functions (EOWFs). These are one-way functions $f$ such that any adversary who produces an image of $f$ must “know” its preimage. One formalizes this by saying that for every adversary $\mathcal{A}$ that outputs a value $y \in \text{image}(f)$, there exists an extractor $\mathsf{Ext}$ that, given $\mathcal{A}$'s auxiliary input and randomness, can output a preimage for $y$ under $f$. In the case of black-box (resp., non-black-box [7]) extractability, $\mathsf{Ext}$ is universal and has no access (resp., has access) to $\mathcal{A}$'s code.

[4] Extractability in interactive protocols is well-studied and involves a technique called *rewinding*. In this paper we focus on extractability for non-interactive protocols.

[5] Essentially, one cannot efficiently check if an adversary breaks the assumption.

Until the work of Bitansky *et al.* in [14], EOWFs were only known under very strong knowledge-of-exponent assumptions [13], making little attempt to justify how extraction would work. Bitansky *et al.* defined generalized extractable one-way functions (GEOWFs) and constructed a GEOWF based on sub-exponential learning with errors (or, alternatively, any delegation scheme) and non-black-box extraction, given that the auxiliary input of the adversary is bounded. They also prove that GEOWFs secure against auxiliary input of polynomially unbounded length do not exist assuming indistinguishability obfuscation (which seems an increasingly plausible assumption given recent progress [41, 56]).

*Extractability and SNARKs.* Extractability assumptions are widely used in various flavors of non-interactive zero-knowledge (NIZK) protocols, which are useful tools in ensuring privacy and correctness of cryptographic protocols. Succinct non-interactive zero-knowledge arguments of knowledge (zk-SNARKs, [30,36,37, 48]) are NIZKs that have sublinear-length proofs and are knowledge-sound (for any valid proof, the prover must "know" a witness). The knowledge-soundness property of a SNARK relies on being able to extract the witness from an adversary that outputs a valid argument. SNARKs are extremely popular due to practical applications such as verifiable computation and privacy-preserving cryptocurrencies (e.g., Zcash [11]).

An interesting question is which assumptions are necessary for SNARKs. Due to the impossibility result of Gentry and Wichs [32], any adaptively sound SNARK must rely on non-falsifiable assumptions. However, while non-falsifiable assumptions are necessary, they need not be knowledge assumptions. In fact, Bitansky et al. [13] showed that extractable collision-resistant hash functions (ECRHs) are necessary and sufficient to construct a SNARK that is adaptively sound and only privately verifiable. More precisely, they construct a designated verifier SNARK for $\mathsf{NP}$ from an ECRH and (an appropriate) private information retrieval, and construct a (specific variant of) ECRH from a designated verifier SNARK and a CRH. They also showed that ECRH implies EOWF.

*Extractability and Subversion Zero-knowledge.* Efficient SNARKs are typically defined in the common reference string (CRS) model, where one assumes that the prover and the verifier have access to a CRS generated by a trusted third party. However, in practice, such a party usually does not exist; this is important since a malicious CRS generator may cooperate with the prover to break soundness, or with the verifier to break zero-knowledge. Thus, it is preferable to construct SNARKs, and NIZKs in general, in weaker trust models than the CRS model.

The general notion of parameter subversion has been studied in [53]. Bellare et al. [9] defined subversion zero-knowledge (Sub-ZK), where zero-knowledge holds even in the case of a dishonestly generated CRS, and constructed a Sub-ZK NIZK argument. Subsequently, [1,3,27] constructed Sub-ZK SNARKs and [2] constructed succinct Sub-ZK quasi-adaptive NIZKs [42]. As noted in [2], Sub-ZK in the CRS model is equivalent to zero-knowledge in the minimal bare public key (BPK, [20]) model where the authority is only trusted to store the public key of each party. Since auxiliary-string non-black-box NIZK is impossible in the BPK model [33], one needs to use non-auxiliary-string non-black-box techniques to achieve Sub-ZK [2]. Existing Sub-ZK NIZKs extract a CRS trapdoor from the (possibly malicious) CRS generator, and then use the CRS trapdoor to simulate the NIZK argument. Prior to our work, extraction in Sub-ZK NIZKs was done using a concrete knowledge-of-exponent assumption.

As previously mentioned, the work of Bitansky et al. [13] established that extractable collision-resistant hash functions are necessary to obtain adaptive soundness of SNARKs. A natural extension of this question is then to ask:

> Which assumptions are necessary to obtain Sub-ZK for NIZKs and SNARKs? Are those assumptions stronger than the ones required to obtain adaptive soundness of SNARKs?

### 1.1 Our Contributions

Inspired by (G)EOWFs, we propose a new *generic assumption*[6]: the existence of verifiably-extractable (generalized) OWFs (VE(G)OWFs). We argue that VEGOWFs are a natural extension of GEOWFs introduced by Bitansky et al. [14], and show that in fact their GEOWF construction can easily be turned into a VEGOWF. Moreover, while Bitansky et al. [14] showed that a GEOWF can be transformed into a EOWF under certain assumptions, we similarly show that any VEGOWF can be transformed into a VEOWF with no further assumptions. To circumvent the impossibility result that EOWF and similar primitives do not exist assuming indistinguishability obfuscation, our definitions include non-black-box extractability as in [14] and assume a benign distribution of auxiliary inputs as suggested in [18].

Answering the first research question, we show that VEGOWFs are vital in understanding subversion zero-knowledge. Firstly, we show that VEGOWFs allow for the transformation of any perfect NIZK with a publicly verifiable CRS into a Sub-ZK NIZK. Secondly, we show the necessity of VEGOWFs by showing that the existence of a Sub-ZK NIZK with certain properties implies that the NIZK's CRS generation algorithm must be a VEOWF. We also prove that if a NIZK has perfect zero-knowledge and well-formedness of the CRS can be efficiently verified, then we automatically obtain a statistical two-message private-coin witness-indistinguishable argument. Obtaining statistical two-message witness-indistinguishable arguments (either public or private coin) was an open question until recently [6, 35, 49]. Similar observations were previously made about specific Sub-ZK SNARKs in [27].

We answer the second research question by showing that the assumption corresponding to this primitive seems weaker than that of extractable collision-resistant hash functions. In particular, we show that VEGOWFs can be built either from knowledge assumption or knowledge-sound NIZKs, and we also propose candidate VEGOWFs from various signature schemes.

[6] Generic assumptions postulate the existence of a cryptographic primitive, such as OWFs and one-way permutations. Meanwhile, concrete assumptions are used for concrete constructions, such as the RSA assumption [52] for the RSA cryptosystem.

By showing connections to Sub-ZK NIZK, our work further demonstrates the importance of extractable OWFs as an independent primitive. This tool, which has not been thoroughly studied, seems to lead the way to protocols that are otherwise difficult to achieve. We encourage further study into extractable functions under weaker (or different) assumptions as there are significant differences between various non-black-box techniques.

## 2 Technical Overview

Extending the notions of EOWF [19] and GEOWF [14], we define *Verifiably-Extractable Generalized One-Way Functions* (VEGOWFs), show several instantiations of these and show how it is related to subversion resistant zero-knowledge. Intuitively, an EOWF $f$ is a one-way function such that for any PPT adversary $\mathcal{A}$, there exists a PPT extractor $\mathsf{Ext}_{\mathcal{A}}$, such that if $\mathcal{A}$ outputs $y \in \text{image}(f)$, then $\mathsf{Ext}_{\mathcal{A}}$ (given access to $\mathcal{A}$'s auxiliary input) retrieves $x$ such that $f(x) = y$. Meanwhile, a GEOWF $g$ generalizes EOWFs by introducing a relation $\mathbf{RG}$ such that for every PPT $\mathcal{A}$, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}$, such that if $\mathcal{A}$ outputs $y \in \text{image}(g)$, then $\mathsf{Ext}_{\mathcal{A}}$ (given access to $\mathcal{A}$'s auxiliary input) returns $z$ such that $(y, z) \in \mathbf{RG}$. It is required that it is difficult for any adversary who is only given $y$ to compute such $z$, i.e., $\mathbf{RG}$ is a hard relation.

### 2.1 Verifiably-Extractable (Generalized) OWFs

A *Verifiably-Extractable Generalized OWF* (VEGOWF) $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}}\}_{\mathsf{e}}$ is a GEOWF which additionally allows one to efficiently check whether extraction will succeed for a given value $y$. More precisely, we define a relation $\mathbf{RG}_{\mathsf{e}}$ and a set $Y_{\mathsf{Ext}} \supseteq \text{image}(\mathsf{g}_{\mathsf{e}})$ such that

(i) given $y$ one can efficiently verify whether $y \in Y_{\mathsf{Ext}}$ and

(ii) if $y \in Y_{\mathsf{Ext}}$ then there exists an extractor $\mathsf{Ext}_{\mathcal{A}}$ that given non-black-box access to $\mathcal{A}$ extracts $z$ such that $(y, z) \in \mathbf{RG}_{\mathsf{e}}$.

Note that extraction should work even if $y \in Y_{\mathsf{Ext}} \setminus \text{image}(\mathsf{g}_{\mathsf{e}})$, and in general, it might be hard to decide if $y \in \text{image}(\mathsf{g}_{\mathsf{e}})$. We say that a VEGOWF is keyless if $\mathsf{e}$ is the security parameter $\lambda$; in this case we write $\mathbf{RG}$ instead of $\mathbf{RG}_{\mathsf{e}}$. The formal definition of VEGOWFs can be found in Section 4.1.

We denote both properties together as $\mathbf{RG}$-*verifiable-extractability.* The requirements for $\mathbf{RG}$-hardness remain the same as for GEOWFs. We introduce *verifiably-extractable OWFs* (VEOWF) as a special case of VEGOWFs where the corresponding relation is $\mathbf{RG}_{\mathsf{e}} = \{(\mathsf{g}_{\mathsf{e}}(x), x)\}$.

**Generic transformations.** We show that any VEGOWF can be transformed to a VEOWF with a simple technique that was first mentioned in [14], in a slightly different context. However, since the transformation incurs some efficiency loss, we still consider VEGOWFs to be a weaker primitive and base our subversion zero-knowledge application on VEGOWFs. We also give a construction of a VEGOWF from any GEOWF by evaluating the GEOWF on two different inputs and attaching a NIWI proof (in the plain model) that at least one of the functions was evaluated correctly. Together they give a surprising result that any GEOWF can be transformed to a VEOWF under the relatively mild assumptions (e.g., decisional linear assumption) required by the NIWI. We note that similar techniques have been previously used in specific applications. For example, [12] uses similar idea to obtain a 3-round zero-knowledge argument from any (non-verifiable) EOWF. We believe it is valuable to point out that this technique works as a general transformation. See Section 4.2 for more details.

**Robust Combiners.** We show that $n$ VEGOWFs can be combined to a new VEGOWF, which is secure if any $t > n/2$ of the initial functions is secure. A robust combiner [26, 40] for VEGOWFs is useful since many of the proposed VEGOWFs rely on strong assumptions. With combining we only need to trust that some of those strong assumptions hold without knowing which. Details are provided in Section 4.2.

We show several VEGOWFs and VEOWFs under various assumptions like bounded auxiliary input size, knowledge assumptions, and the random oracle.

**VEGOWF from the BCPR construction.** In the first construction, we show that the keyless GEOWF $\mathcal{G}$ from [14, Fig. 4] is, in fact, a VEGOWF against any adversary with bounded auxiliary input if we assume that the used delegation scheme has efficient public CRS-verifiability. We recall that a delegation scheme $\mathsf{DS}$ [5] allows one to prove statements of the form “a machine $\mathcal{M}$ outputs $y$ on input $x$ in time $t$”. A delegation proof $\pi_{\mathsf{DS}}$ must be faster to verify than the statement itself. The CRS-verifiability means that one can efficiently check if the $\mathsf{DS}$ CRS $\mathsf{crs}_{\mathsf{DS}}$ is a valid CRS.

In the BCPR construction, each function $\mathsf{g}_{\mathsf{e}}$ computes a CRS $\mathsf{crs}_{\mathsf{DS}}$ for a delegation scheme $\mathsf{DS}$, and then evaluates a PRG on a random value. The relation $\mathbf{RG}(y, z)$ holds for $y = (\mathsf{crs}_{\mathsf{DS}}, v)$ and $z = (\mathcal{A}, \pi_{\mathsf{DS}}, \mathsf{pad})$, if $\pi_{\mathsf{DS}}$ is a $\mathsf{DS}$-proof, using $\mathsf{crs}_{\mathsf{DS}}$ as the CRS, for the statement that $\mathcal{A}$ on input $1^\lambda$ outputs $v$. ($\mathsf{pad}$ is a padding.) The proof of $\mathbf{RG}$-hardness is as in [14], and follows from the security of the PRG together with an argument about Kolmogorov complexity. The $\mathbf{RG}$-verifiable-extractability follows from the CRS-verifiability and completeness of the delegation scheme. See Section 4.3 for more details.

We note that even if the delegation scheme is not CRS-verifiable, one could still make the BCPR EOWF a VEGOWF using the generic transformation presented in Section 4.2.

**VEGOWFs from knowledge-of-exponent assumptions.** Secondly, we show that many knowledge-of-exponent assumptions naturally imply VEGOWFs. For these VEGOWFs, the input key $\mathsf{e}$ consists of a bilinear group description and possibly some additional information.

We first construct of a VEOWF based on the Bilinear Diffie–Hellman Knowledge-of-Exponent (BDH-KE) assumption from [1] which states that if an adversary on input $\mathsf{p}$ (the asymmetric bilinear group description) outputs $([x]_1, [x]_2)$ for some $x$ then he knows $x$.[7] Here, $\mathsf{e} = \mathsf{p}$ and $\mathsf{g}_\mathsf{p}(x) = ([x]_1, [x]_2)$.

We also construct a VEGOWF based on the Diffie–Hellman Knowledge of Exponent (DH-KE) assumption introduced in [9]. The key is a description $\mathsf{p}$ of a symmetric bilinear group, and $\mathsf{g}_\mathsf{p}(x, y) = [x, y, xy]_1$. The DH-KE assumption states that is is possible to extract at least one of $x$ and $y$. This results in a VEGOWF with respect to the relation $\mathbf{RG}_\mathsf{p}([x, y, xy]_1, z) = 1$ iff $z = x$ or $z = y$.

We discuss these and other similar VE(G)OWF constructions in Section 4.4.

**VEGOWFs from knowledge sound NIZKs.** Thirdly, inspired by [22, 47], we build VEGOWFs using knowledge-sound NIZKs. Suppose that we have a knowledge-sound NIZK $\Pi$ for a relation $\mathbf{R}$ and that $\mathbf{R}$ has an efficient sampling algorithm $\mathcal{S}$ which produces instances that are hard on average. We define $\mathsf{g}_\mathsf{e}(r_\mathcal{S}, r_\pi)$ such that it samples $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(r_\mathcal{S})$, uses $r_\pi$ as random coins to generate a proof $\pi$ for $\mathsf{x}$, and outputs $(\mathsf{x}, \pi)$. The input $\mathsf{e}$ is either the CRS or a description of a hash function (in the random oracle model). We define $\mathbf{RG}_\mathsf{e}((\mathsf{x}, \pi), \mathsf{w}) = 1$ iff $\pi$ satisfies NIZK verification and $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$. Since $\Pi$ is knowledge-sound, we obtain $\mathbf{RG}$-verifiable-extractability by using $\Pi$'s verification on $(\mathsf{x}, \pi)$. $\mathbf{RG}$-hardness is satisfied since $\pi$ is simulatable and $\mathcal{S}$ produces hard instances on average.

As an interesting instantiation, if we let $\mathcal{S}$ output $([x], x)$ for a random $x$ and use Schnorr's $\Sigma$-protocol together with the Fiat-Shamir heuristic as a NIZK, we obtain a very efficient VEOWF $\mathsf{g}_\mathsf{e}(x, r) := (\mathsf{x} = [x], a = [r], z = H([x], [r]) \cdot x + r)$ where $H$ is a hash function and verification works by asserting that $H(\mathsf{x}, a)\mathsf{x} + a = [z]$. See Section 4.5 for more details.

**VEGOWFs from signature schemes.** Finally, we propose a novel heuristic for coming up with new VEGOWFs and knowledge-type assumptions in general. The intuition behind signature schemes is that only the one with (at least some) knowledge of the signing key $\mathsf{sk}$ can sign a message. Thus, it gives a very simple formula for looking for new VEGOWFs. Let $\Sigma = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ be a digital signature scheme. Then, $\mathsf{g}_\mathsf{p}(\mathsf{sk}) = (\mathsf{vk} = \mathsf{KGen}(\mathsf{sk}), \sigma = \mathsf{Sign}(\mathsf{sk}, m = 0))$ is a candidate for a VEGOWF where $\mathsf{p}$ is some parameter for the signature scheme, in particular when $\mathsf{vk} \in \mathsf{KGen}$ can be efficiently tested. Of course, this is just a heuristic since at least the standard notion of existential unforgeability does not require that the signer knows the secret key.

We then proceed by going over many concrete signatures schemes and investigate the security of the corresponding VEGOWF candidate. We see that in some cases the VEGOWF is insecure (e.g., Lamport's one-time signature [46] and RSA signature), in some cases it gives a VEGOWF that we already considered before (e.g., Schnorr's signature scheme [55] and Boneh-Boyen signature [16]) and in some cases we obtain (plausibly secure) VEGOWFs that have not been considered before. In the latter set is for example the DSA signature which gives

[7] We use the additive notation for bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ where $[x]_i$ denotes $xg_i$ using the fixed generator $g_i$ of $\mathbb{G}_i$ described in $\mathsf{p}$. A bilinear map $\bullet$ allows us to compute $[x]_1 \bullet [y]_2 = [xy]_T$.

quite a unique function in a non-pairing-based group and (and a slight modification of) the hash-and-sign lattice based signature scheme of [31], which gives the first lattice based VEGOWF candidate.

### 2.2 Constructing Sub-ZK NIZK from VEGOWF

We propose two generic constructions of a Sub-ZK NIZK. The first construction produces a knowledge-sound Sub-ZK NIZK from any knowledge-sound Sub-WI NIWI[8] and keyless VEGOWF. The second construction produces a sound Sub-ZK NIZK from a sound Sub-WI NIWI, a keyless extractable commitment, and a VEGOWF.

**Knowledge-sound Sub-ZK NIZK.** For the first construction, we propose a knowledge-sound Sub-ZK NIZK for any $\mathsf{NP}$-relation $\mathbf{R}$ using a variant of the well-known FLS disjunctive approach [25]. Namely, we use a knowledge-sound Sub-WI NIWI $\Pi_{wi}$ for the composite relation $\mathbf{R}'$, where $((\mathsf{x}, \widehat{y}), (\mathsf{w}, \widehat{z})) \in \mathbf{R}'$ iff either $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$ or $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$. Here $\mathcal{G} = \{\mathsf{g_e}\}$ is a keyless VEGOWF with respect to $\mathbf{RG}$ and $\widehat{y} \in Y_{\mathsf{Ext}}$ being added to $\Pi_{wi}$'s CRS. Knowledge-soundness of the new protocol will follow from the knowledge-soundness of $\Pi_{wi}$ together with the $\mathbf{RG}$-hardness of $\mathcal{G}$, and subversion zero-knowledge follows from the verifiable-extractability of $\mathcal{G}$ and the Sub-WI property of $\Pi_{wi}$. This construction preserves succinctness, and thus we obtain a Sub-ZK SNARK from a keyless VEGOWF and a Sub-WI SNARK. We later note that any perfectly zero-knowledge SNARK with efficient CRS verification is automatically a Sub-WI SNARK. See Section 5.1 for the full details of the construction.

**Sub-ZK NIZK.** Secondly, we construct a Sub-ZK NIZK $\Pi$ for any $\mathsf{NP}$-relation $\mathbf{R}$. It similarly uses the FLS approach with a keyless VEGOWF, but additionally uses a commitment to a trapdoor. Specifically, $\Pi$ implements a Sub-WI NIWI $\Pi_{wi}$ for the relation $\mathbf{R}'$, where $((\mathsf{x}, c, \widehat{y}), (\mathsf{w}, \widehat{z}, \widehat{r})) \in \mathbf{R}'$ iff $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$ or $c = \mathsf{C.Com}(\widehat{z}, \widehat{r})$ such that $\mathbf{RG}(\widehat{y}, \widehat{z}) = 1$, where $\mathcal{G}$ is a keyless VEGOWF with respect to $\mathbf{RG}$ and $\mathsf{C} = (\mathsf{Com}, \mathsf{Open}, \mathsf{Vf})$ is a keyless extractable commitment scheme.

A proof in $\Pi$ consists of a commitment $c$ and a proof in $\Pi_{wi}$, so this construction is less efficient than the previous one. However, this does not rely on $\Pi_{wi}$ being knowledge-sound, so the construction is still of interest. The soundness of $\Pi$ follows from the soundness of $\Pi_{wi}$ together with the $\mathbf{RG}$-hardness of $\mathcal{G}$ and the extractability of $\mathsf{C}$. Note that $\Pi_{wi}$ will already guarantee that $c$ is a valid commitment. Therefore, we do not need the commitment itself to have an efficient image verification procedure and can obtain it from any (even non-verifiable) injective EOWF. Sub-ZK follows from the verifiable-extractability of $\mathcal{G}$, the Sub-WI property of $\Pi_{wi}$ and the hiding property of $\mathsf{C}$. See Section 5.2 for the full details of the construction.

**Statistical ZAPRs with adaptive soundness.** We observe that if a NIZK has perfect zero-knowledge and CRS-verifiability, then we immediately obtain a sta-

[8] Although in the literature NIWI often refers to the plain model, in this context we allow for a CRS. A Sub-WI NIWI needs to remain witness indistinguishable even if the CRS is subverted. We note that any CRS-less NIWI is trivially a Sub-WI NIWI.

tistical two-message private-coin witness-indistinguishable argument. Obtaining statistical two-message witness-indistinguishable arguments that are public-coin (ZAP) or private-coin (ZAPR) was considered a significant open problem, until recent breakthroughs [6, 35, 49]. Note that existing Sub-ZK SNARKs [1, 27] are already statistical ZAPRs with adaptive soundness. Compared to previous statistical ZAP/ZAPR constructions, the soundness of SNARKs is based on less standard assumptions, but they have much better efficiency. Similar observations about Sub-ZK SNARKs were previously made by Fuchsbauer in [27].

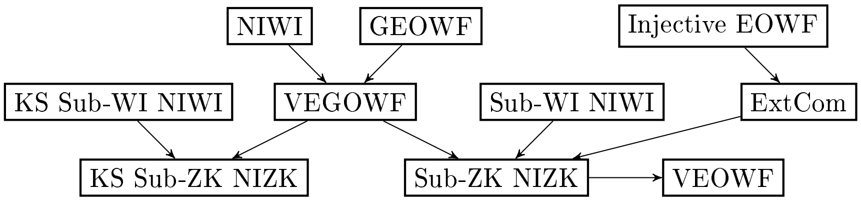

**Fig. 1.** Relations between argument systems and extractable functions. Multiple arrows pointing to the same node means that each source node is required to construct the destination node. KS denotes knowledge-sound.

**Instantiations.** The relations between our primitives are summarized in Fig. 1.

Table 1 shows a selection of instantiations for our generic constructions and compares them to previous work. We can achieve a keyless extractable commitment from any keyless injective VEOWF (or even from keyless injective EOWF if the commitment does not have to be image verifiable). In particular, this includes a VEOWF based on the symmetric discrete logarithm (SDL) assumption and the BDH-KE assumption, and a VEOWF based on the security of a non-interactive version of Schnorr's protocol.

We can construct a Sub-ZK NIZK by combining a keyless extractable commitment, a VEGOWF, and a Sub-WI NIWI. For example, we may use the Sub-WI NIWI of [39] based on DLIN or [15] based on *iO* and OWF. In comparison, [9] proposed a Sub-ZK NIZK which is based on the DLIN and DH-KE assumptions. We can obtain a KS Sub-ZK NIZK by combining a KS Sub-WI NIWI with a VEGOWF. In Table 1, we consider the case where we use [28] as the KS Sub-WI NIWI component, together with a VEGOWF which holds under the same assumptions. In Section 5.2, we also show that existing Sub-ZK SNARKs [1, 27] can be slightly modified to achieve Sub-ZK from any VEGOWF rather than a specific knowledge-of-exponent assumption.

### 2.3 Constructing VEOWF from Sub-ZK NIZK

It turns out that not only can Sub-ZK NIZK be constructed with the help of VEGOWF, but (under certain restrictions) Sub-ZK NIZK also implies a

| | **Soundness** | **Knowledge Soundness** | **Sub-ZK** |
|---|---|---|---|
| [9] | DH-KE + CDH | × | DH-KE + DLIN |
| Sec. 5.2 | injective VEOWF | × | injective VEOWF + DLIN |
| Sec. 5.2 | injective VEOWF | × | injective VEOWF + $iO$ |
| [1] | GGM | GGM | BDH-KE |
| [27, Sec. 4] | $q_1$-PDH + $q_2$-PKE | $q_1$-PDH + $q_2$-PKE | SKE |
| [27, Sec. 5] | $q_1$-PDH + $q_2$-PKE + $q_3$-SDH | $q_1$-PDH + $q_2$-PKE + $q_3$-TSDH | SKE |
| [27, Sec. 6] | GGM | GGM | SKE |
| Sec. 5.1 | DH-KE + DL | DH-KE + DL | DH-KE + DLIN |

**Table 1.** Instantiations of our generic constructions in comparison to previous work. SKE denotes the Square Knowledge-of-Exponent assumption, GGM denotes the generic group model, PDH denotes the Power Diffie-Hellman assumption, PKE denotes the Power Knowledge-of-Exponent assumption, and TSDH denotes the Target Strong Diffie-Hellman assumption.

VE(G)OWF. In that sense, VEGOWF is both a necessary and a sufficient condition for achieving Sub-ZK NIZKs, similar to how ECRH (also, under certain restrictions) is a necessary and a sufficient condition for achieving a SNARK.

More technically, we consider a CRS generation function $\mathsf{KGen}_{\mathbf{R},\mathsf{p}}$ of a Sub-ZK NIZK that takes as an input a randomly sampled trapdoor $\mathsf{td}$ and outputs a $\mathsf{crs}$. We show that this function has to be one-way if the NIZK is both computationally sound and computationally zero-knowledge. Intuitively, if one-wayness would not hold, the soundness adversary could recover $\mathsf{td}$ and use the simulator to construct a proof for a false statement. We additionally require that $\mathsf{KGen}_{\mathbf{R},\mathsf{p}}$ is injective to avoid the situation where one-wayness adversary computes $\mathsf{td}$ is which is particularly bad for simulation among all the possible preimages of $\mathsf{crs}$. Verifiable-extractability property follows straightforwardly from the Sub-ZK property of the NIZK since it requires that $\mathsf{td}$ must be extractable. However, here we also need to make some slight restrictions. Namely, the Sub-ZK extractor should be able to extract the complete $\mathsf{td}$, not only some part of it, which might still be sufficient for simulating the proof.

## 3 Preliminaries

Let PPT denote probabilistic polynomial-time. Let $\lambda \in \mathbb{N}$ be the security parameter. All adversaries are stateful. For an algorithm $\mathcal{A}$, let image$(\mathcal{A})$ be the image of $\mathcal{A}$ (the set of valid outputs of $\mathcal{A}$), let $\mathsf{RND}_\lambda(\mathcal{A})$ denote the random tape of $\mathcal{A}$, and let $r \leftarrow\!\!\$\, \mathsf{RND}_\lambda(\mathcal{A})$ denote the random choice of values from $\mathsf{RND}_\lambda(\mathcal{A})$. We write that $y \in \mathrm{range}(\mathcal{A}(x))$ if there is non-zero probability that the algorithm $\mathcal{A}$ outputs a value $y$ given the input $x$. We denote by $\mathsf{negl}(\lambda)$ an arbitrary negligible function and by $\mathsf{poly}(\lambda)$ an arbitrary polynomial function. We write $a(\lambda) \approx_\lambda b(\lambda)$ if $|a(\lambda) - b(\lambda)| = \mathsf{negl}(\lambda)$. For an $\mathsf{NP}$-relation $\mathbf{R} = \{(\mathsf{x}, \mathsf{w})\}$, let $\mathcal{L}_\mathbf{R} := \{\mathsf{x} : \exists \mathsf{w}, (\mathsf{x}, \mathsf{w}) \in \mathbf{R}\}$ be the corresponding language.

In the pairing-based setting, we use the standard bracket notation together with additive notation, i.e., we write $[a]_\iota$ to denote $a g_\iota$ where $g_\iota$ is a fixed generator of $\mathbb{G}_\iota$ and $a \in \mathbb{Z}_p$ for some prime $p$. Intuitively, pairings $\bullet : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ are efficient (one-way) functions that map $([a]_1, [b]_2)$ to $[a]_1 \bullet [b]_2 = [ab]_T$.

Let $A = \{A_\lambda\}_{\lambda \in \mathbb{N}}, B = \{B_\lambda\}_{\lambda \in \mathbb{N}}$ be collections of efficiently sampleable sets, such that $|B_\lambda| > |A_\lambda|$ for each $\lambda \in \mathbb{N}$. A polynomial-time function $\mathsf{PRG}\colon A_\lambda \to B_\lambda$ is a pseudorandom generator (PRG) if its output is computationally indistinguishable from a truly random one.

### 3.1 (Generalized) Extractable OWF

An extractable one-way function (EOWF, [19]) $\mathsf{g}$ is an OWF with the property that if $\mathcal{A}$ outputs a value in the image of $\mathsf{g}$, then one can extract its preimage. A generalized EOWF (GEOWF, [14]) is a function $\mathsf{g}$ with an associated hard relation $\mathbf{RG}$, such that given $\mathsf{g}(x)$, it is intractable to compute $z$ such that $\mathbf{RG}(\mathsf{g}(x), z) = 1$. However, given a machine (and its auxiliary input) that computes $\mathsf{g}(x)$, it is possible to extract $z$ such that $\mathbf{RG}(\mathsf{g}(x), z) = 1$. One obtains an EOWF when $\mathbf{RG} = \{(\mathsf{g}(x), z) : \mathsf{g}(z) = \mathsf{g}(x)\}$. Unless stated otherwise, we assume that $\mathbf{RG}$ is efficiently checkable.

Bitansky *et al.* [14] show that, assuming the existence of indistinguishability obfuscation, there do not exist EOWFs or GEOWFs with common auxiliary-input of unbounded polynomial length. However, the result does not rule out their existence when the common auxiliary input comes from some natural distribution, such as the uniform distribution. Thus, nowadays zk-SNARKs explicitly assume that the auxiliary input is benign, i.e., with overwhelming probability it does not encode a malicious obfuscation. We also make the same assumption: if no bound for the auxiliary input is given, then we assume that it is taken from a benign distribution.

We present a slight modification of the GEOWF definition of [14]. Note that hardness is required to hold even against poly-length auxiliary inputs.

**Definition 1 (GEOWFs).** *Let $\mathcal{X} = \{X_\lambda\}_\lambda$, $\mathcal{Y} = \{Y_\lambda\}_\lambda$, $\mathcal{Z} = \{Z_\lambda\}_\lambda$ and $\mathcal{K} = \{K_\lambda\}_\lambda$ be collections of sets indexed by $\lambda \in \mathbb{N}$. An efficiently computable family of functions $\mathcal{G} = \{\mathsf{g}_\mathsf{e}\colon X_\lambda \to Y_\lambda \mid \mathsf{e} \in K_\lambda, \lambda \in \mathbb{N}\}$ associated with an efficient (probabilistic) key sampler* $\mathsf{KeySamp}$*, is a* GEOWF *with respect to a relation $\mathbf{RG}_\mathsf{e}(y, z)$ on triples $(\mathsf{e}, y, z) \in K_\lambda \times Y_\lambda \times Z_\lambda$ if it is:*

**RG-hard:** *for any PPT adversary $\mathcal{A}$ and any* $\mathsf{aux}$ *sampled from a benign distribution of* $\mathsf{poly}(\lambda)$*-bit strings*

$$\Pr_{\substack{\mathsf{e} \leftarrow \mathsf{KeySamp}(1^\lambda) \\ x \leftarrow\$ X_\lambda}} [z \leftarrow \mathcal{A}(\mathsf{e}, \mathsf{g}_\mathsf{e}(x), \mathsf{aux}) : \mathbf{RG}_\mathsf{e}(\mathsf{g}_\mathsf{e}(x), z) = 1] \le \mathsf{negl}(\lambda) \enspace .$$

**RG-extractable:** *For any PPT adversary $\mathcal{A}$, there exists a PPT extractor* $\mathsf{Ext}_\mathcal{A}$*, s.t. for any benign distribution $\mathcal{D}_\lambda$ of* $\mathsf{poly}(\lambda)$*-bit strings,*

$$\Pr_{\substack{\mathsf{e} \leftarrow \mathsf{KeySamp}(1^\lambda) \\ \mathsf{aux} \leftarrow \mathcal{D}_\lambda}} \left[\begin{matrix} y \leftarrow \mathcal{A}(\mathsf{e}; \mathsf{aux}), z \leftarrow \mathsf{Ext}_\mathcal{A}(\mathsf{e}; \mathsf{aux}) : \\ y \in \mathrm{image}(\mathsf{g}_\mathsf{e}) \wedge \mathbf{RG}_\mathsf{e}(y, z) \ne 1 \end{matrix}\right] \le \mathsf{negl}(\lambda) \enspace .$$

*The function is* publicly verifiable *if there exists a polynomial-time tester $\mathcal{T}$ such that for any* $(\mathsf{e}, x, z)$*,* $\mathbf{RG}_{\mathsf{e}}(\mathsf{g}_{\mathsf{e}}(x), z) = \mathcal{T}(\mathsf{e}, \mathsf{g}_{\mathsf{e}}(x), z)$*.*

*We say that a GEOWF is keyless if, for each security parameter $\lambda$, there is only one key* $\mathsf{e} = 1^{\lambda}$*. For ease of notation, we simply write* $\mathsf{g}_{\lambda}$ *and* $\mathbf{RG}$ *in this case. A GEOWF is an EOWF if* $\mathbf{RG}_{\mathsf{e}}(\mathsf{g}_{\mathsf{e}}(x), z) = \{(\mathsf{e}, \mathsf{g}_{\mathsf{e}}(x), z) : \mathsf{g}_{\mathsf{e}}(x) = \mathsf{g}_{\mathsf{e}}(z)\}$*.*

**Bounded auxiliary input.** We also consider GEOWFs where the auxiliary input in $\mathbf{RG}$-extractability holds for any $\mathsf{aux} \in \{0,1\}^{\mathfrak{b}(\lambda)}$ (not just for a benign distribution) for some fixed polynomial $\mathfrak{b}$. We call these $\mathfrak{b}$-bounded GEOWFs.

### 3.2 BCPR GEOWF and EOWF

Bitansky *et al.* [14] show that if the common auxilliary string of the adversary and the extractor has an a priori bounded length $\mathfrak{b}(\lambda)$, then one can implement extractable one-way functions (EOWF) based on a pseudorandom generator and a universal delegation scheme [43,44]. In a universal delegation scheme, one delegates computation of some circuit $M$ on input $x$ to a prover, who must compute $M(x)$ and provide a proof $\pi$ that he computed it correctly; any verifier that is given $(M, x, M(x), \pi)$ must be able to verify the proof in less time than computing $M(x)$ itself. One can construct universal delegation schemes under the subexponential learning with errors assumption [44] and even falsifiable assumptions [43] for languages in $\mathsf{BPP}$.

**BCPR GEOWF.** We briefly describe the construction from [14] of a GEOWF secure against an adversary with $(\mathfrak{b}(\lambda) - \omega(1))$-bounded auxiliary input.

Fix a polynomial $\mathfrak{b}(\lambda)$. Let $\mathsf{PRG}\colon \{0,1\}^{\lambda} \to \{0,1\}^{\mathfrak{b}(\lambda)+\lambda}$ be a PRG. Let $\mathsf{DS}$ be a *universal* delegation scheme that consists of a CRS generator $\mathsf{DS.K}$, a prover $\mathsf{DS.P}$, and a verifier $\mathsf{DS.V}$. We assume that using $\mathsf{DS}$, one can construct a succinct proof $\pi_{\mathsf{DS}}$ of length $\mathsf{DS.plen}(\lambda)$ that a Turing machine $\mathsf{M}$ on input $1^{\lambda}$ outputs some value $v$ in time $T(\lambda)$, where $T(\lambda) \in (2^{\omega(\log \lambda)}, 2^{\mathsf{poly}(\lambda)})$ is some superpolynomial function. $\mathsf{DS}$ must satisfy that the proof verification complexity is linear in $\mathsf{M}$'s size and polylogarithmic in $\mathsf{M}$'s execution time $T$.

We define the function $\mathsf{g}_{\lambda}\colon (s, r) \mapsto (\mathsf{crs}_{\mathsf{DS}}, v)$ and the corresponding relation $\mathbf{RG}(y, z)$ as in Fig. 2, where $y = (\mathsf{crs}_{\mathsf{DS}}, v)$ and $z = (\mathsf{M}, \pi_{\mathsf{DS}}, \mathsf{pad})$ with $|z| = \mathfrak{l}(\lambda)$.

**Proposition 1 ( [14, Theorem 14]).** $\mathcal{G} = \{\mathsf{g}_{\lambda}\}_{\lambda \in \mathbb{N}}$*, depicted in Fig. 2, is a GEOWF with respect to* $\mathbf{RG}$*, against* $(\mathfrak{b}(\lambda) - \omega(1))$*-bounded auxiliary input.*

This proposition relies on the security of $\mathsf{DS}$ and $\mathsf{PRG}$. In addition, it uses a Barak-type [7] extractability paradigm (namely, the machine $\mathsf{M}$ is the adversary who outputs $y$). It is worth noting that a similar approach with a number of extra steps [14, Theorem 14] also allows one to construct a function family which is an EOWF against $(\mathfrak{b}(\lambda) - \omega(1))$-bounded auxiliary-input. We will see an adaptation of this approach in Section 4.2.

```
$\mathsf{g}_\lambda(s, r)$
$(\mathsf{crs}_{\mathsf{DS}}, \tau) \leftarrow \mathsf{DS.K}(1^\lambda; r)$;    // the generator for universal delegation
return $(\mathsf{crs}_{\mathsf{DS}}, v \leftarrow \mathsf{PRG}(s))$;

$\mathbf{RG}(y, z)$
parse $y = (\mathsf{crs}_{\mathsf{DS}}, v), z = (\mathsf{M}, \pi_{\mathsf{DS}}, \mathsf{pad})$;
    // $|\mathsf{M}| = \mathfrak{b}(\lambda), |\pi_{\mathsf{DS}}| = \mathsf{DS.plen}(\lambda), |\mathsf{pad}| = \mathfrak{l}(\lambda) - \mathfrak{b}(\lambda) - \mathsf{DS.plen}(\lambda)$;
find the verification state $\tau$ corresponding to the reference string $\mathsf{crs}_{\mathsf{DS}}$;
verify the statement "$\mathsf{M}(1^\lambda)$ outputs $v$ in $T(\lambda)$ steps" by using $\pi_{\mathsf{DS}}$ (DS proof);
return 1 iff the DS verifier accepts $\pi_{\mathsf{DS}}$;
```

**Fig. 2.** BCPR GEOWF $\mathcal{G}$ (above) and the relation $\mathbf{RG}(y, z)$ (below).

### 3.3 NIZK and NIWI Arguments

We recall the definition of NIZK and NIWI arguments and their security properties. We assume that $\mathcal{R}$ is a relation generator that output an $\mathsf{NP}$ relation $\mathbf{R}$ and a parameter $\mathsf{p}$ (e.g., the group description). An argument system $\Psi$ is a tuple of PPT algorithms $(\mathsf{K}, \mathsf{P}, \mathsf{V})$. The CRS generation algorithm $\mathsf{K}$ takes in $(\mathbf{R}, \mathsf{p})$ and outputs a $\mathsf{crs}$ and a trapdoor $\mathsf{td}$ (which may be $\perp$ if the argument does not have zero-knowledge). The prover algorithm $\mathsf{P}$ takes in $\mathbf{R}, \mathsf{p}, \mathsf{crs}$ and $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$ and outputs a proof $\pi$. The verifier algorithm $\mathsf{V}$ takes in $(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$ and outputs either 0 (rejecting the proof) or 1 (accepting the proof). A NIZK argument system will additionally have a simulator $\mathsf{Sim}$ that takes in $(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}, \mathsf{x})$ and outputs a simulated proof $\pi$ for the statement $\mathsf{x}$. Furthermore, a subversion resistant argument will have a CRS verification algorithm $\mathsf{CV}$ that take in $(\mathbf{R}, \mathsf{p}, \mathsf{crs})$ and output either 0 (by rejecting the CRS) or 1 (by accepting the CRS).

**Definition 2 (Perfect Completeness [37]).** *A non-interactive argument $\Psi$ is* perfectly complete for $\mathcal{R}$, *if for all $\lambda$, all* $(\mathbf{R}, \mathsf{p}) \in \mathrm{range}(\mathcal{R}(1^\lambda))$, *and* $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$,

$$\Pr\left[\mathsf{crs} \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p}) : \mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \mathsf{P}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \mathsf{w})) = 1\right] = 1 \enspace .$$

**Definition 3 (Perfect CRS Verifiability).** *A non-interactive (subversion-resistant) argument $\Psi$ is* perfectly CRS-verifiable for $\mathcal{R}$, *if for all $\lambda$ and all* $(\mathbf{R}, \mathsf{p}) \in \mathrm{range}(\mathcal{R}(1^\lambda))$, $\Pr\left[(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p}) : \mathsf{CV}(\mathbf{R}, \mathsf{p}, \mathsf{crs}) = 1\right] = 1$.

**Definition 4 (Computational Soundness).** *$\Psi$ is computationally (adaptively)* sound for $\mathcal{R}$, *if for every PPT $\mathcal{A}$,*

$$\Pr\left[\begin{matrix}(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda), (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p}), (\mathsf{x}, \pi) \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}) : \\ \mathsf{x} \notin \mathcal{L}_{\mathbf{R}} \wedge \mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1\end{matrix}\right] \leq \mathsf{negl}(\lambda) \enspace .$$

**Definition 5 (Computational Knowledge Soundness).** *$\Psi$ is computationally (adaptively)* knowledge-sound for $\mathcal{R}$, *if for every PPT $\mathcal{A}$, there exists a PPT* *extractor* $\mathsf{Ext}_{\mathcal{A}}$*, such that*

$$\Pr\left[\begin{array}{l}(\mathbf{R},\mathsf{p}) \leftarrow \mathcal{R}(1^\lambda), (\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R},\mathsf{p}), r \leftarrow\!\!\$\ \mathsf{RND}_\lambda(\mathcal{A}),\\ (\mathsf{x},\pi) \leftarrow \mathcal{A}(\mathbf{R},\mathsf{p},\mathsf{crs};r), \mathsf{w} \leftarrow \mathsf{Ext}_{\mathcal{A}}(\mathbf{R},\mathsf{p},\mathsf{crs};r) :\\ (\mathsf{x},\mathsf{w}) \not\in \mathbf{R} \wedge \mathsf{V}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{x},\pi) = 1\end{array}\right] \le \mathsf{negl}(\lambda)\ .$$

Above we assume that the input $(\mathbf{R},\mathsf{p},\mathsf{crs};r)$ comes from a benign distribution and thus avoids the impossibility result of [14].

**Definition 6 (Statistically Composable ZK).** *$\Psi$ is* statistically composable zero-knowledge for *$\mathcal{R}$, if for all $(\mathbf{R},\mathsf{p}) \in \mathrm{range}(\mathcal{R}(1^\lambda))$, and all computationally unbounded $\mathcal{A}$, $\varepsilon_0^{comp} \approx_\lambda \varepsilon_1^{comp}$, where $\varepsilon_b^{comp} =$*

$$\Pr\left[\begin{array}{l}(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R},\mathsf{p}), (\mathsf{x},\mathsf{w}) \leftarrow \mathcal{A}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{td}); \pi_0 \leftarrow \mathsf{P}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{x},\mathsf{w});\\ \pi_1 \leftarrow \mathsf{Sim}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{td},\mathsf{x}) : (\mathsf{x},\mathsf{w}) \in \mathbf{R} \wedge \mathcal{A}(\pi_b) = 1\end{array}\right]\ .$$

*$\Psi$ is* perfectly composable ZK for *$\mathcal{R}$ if one requires that $\varepsilon_0^{comp} = \varepsilon_1^{comp}$. In Theorem 8 we also consider a computational version of this definition, that is $\mathcal{A}$ is a PPT adversary and the input* $\mathsf{td}$ *is not given as input to $\mathcal{A}$.*

**Definition 7 (Statistically Composable Sub-ZK [1]).** *$\Psi$ is* statistically composable subversion ZK (Sub-ZK) for *$\mathcal{R}$, if for any PPT subverter $\mathcal{Z}$ there exists a PPT* $\mathsf{Ext}_{\mathcal{Z}}$*, such that for all $\mathbf{R} \in \mathrm{range}(\mathcal{R}(1^\lambda))$, and all computationally unbounded $\mathcal{A}$, $\varepsilon_0^{comp} \approx_\lambda \varepsilon_1^{comp}$, where $\varepsilon_b^{comp} =$*

$$\Pr\left[\begin{array}{l}r \leftarrow\!\!\$\ \mathsf{RND}_\lambda(\mathcal{Z}), (\mathsf{crs},\mathsf{aux}_{\mathcal{Z}}) \leftarrow \mathcal{Z}(\mathbf{R},\mathsf{p};r), \mathsf{td} \leftarrow \mathsf{Ext}_{\mathcal{Z}}(\mathbf{R},\mathsf{p};r)\\ (\mathsf{x},\mathsf{w}) \leftarrow \mathcal{A}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{td},\mathsf{aux}_{\mathcal{Z}}), \pi_0 \leftarrow \mathsf{P}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{x},\mathsf{w});\\ \pi_1 \leftarrow \mathsf{Sim}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{td},\mathsf{x}) : (\mathsf{x},\mathsf{w}) \in \mathbf{R} \wedge \mathsf{CV}(\mathbf{R},\mathsf{p},\mathsf{crs}) = 1 \wedge \mathcal{A}(\pi_b,\mathsf{aux}_{\mathcal{Z}}) = 1\end{array}\right]\ .$$

*$\Psi$ is* perfectly composable Sub-ZK for *$\mathcal{R}$ if one requires that $\varepsilon_0^{comp} = \varepsilon_1^{comp}$.*

**Definition 8 (Witness Indistinguishability).** *$\Psi$ is* computationally witness indistinguishable (WI) for *$\mathcal{R}$, if for any PPT $\mathcal{A}$, $\varepsilon_0^{wi} \approx_\lambda \varepsilon_1^{wi}$, where $\varepsilon_b^{wi} =$*

$$\Pr\left[\begin{array}{l}(\mathsf{crs},\mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R},\mathsf{p}), (\mathsf{x},\mathsf{w}_0,\mathsf{w}_1) \leftarrow \mathcal{A}(\mathbf{R},\mathsf{p},\mathsf{crs}), \pi_b \leftarrow \mathsf{P}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{x},\mathsf{w}_b) :\\ (\mathsf{x},\mathsf{w}_0) \in \mathbf{R} \wedge (\mathsf{x},\mathsf{w}_1) \in \mathbf{R} \wedge \mathcal{A}(\pi_b) = 1\end{array}\right]\ .$$

*$\Psi$ is* perfectly WI for *$\mathcal{R}$ if one requires that $\varepsilon_0^{wi} = \varepsilon_1^{wi}$ for unbounded $\mathcal{A}$. Note that* $\mathsf{td}$ *above might be $\perp$ if $\Psi$ is not zero-knowledge.*

**Definition 9 (Sub-WI [9]).** *$\Psi$ is* computationally Sub-WI for *$\mathcal{R}$, if for any PPT subverter $\mathcal{Z}$, $\varepsilon_0^{wi} \approx_\lambda \varepsilon_1^{wi}$, where $\varepsilon_b^{wi} =$*

$$\Pr\left[\begin{array}{l}(\mathsf{crs},\mathsf{x},\mathsf{w}_0,\mathsf{w}_1,\mathsf{aux}_{\mathcal{Z}}) \leftarrow \mathcal{Z}(\mathbf{R},\mathsf{p}), \pi_b \leftarrow \mathsf{P}(\mathbf{R},\mathsf{p},\mathsf{crs},\mathsf{x},\mathsf{w}_b) :\\ (\mathsf{x},\mathsf{w}_0) \in \mathbf{R} \wedge (\mathsf{x},\mathsf{w}_1) \in \mathbf{R} \wedge \mathsf{CV}(\mathbf{R},\mathsf{p},\mathsf{crs}) = 1 \wedge \mathcal{Z}(\pi_b,\mathsf{aux}_{\mathcal{Z}}) = 1\end{array}\right]\ .$$

*$\Psi$ is* perfectly Sub-WI for *$\mathcal{R}$ if one requires that $\varepsilon_0^{wi} = \varepsilon_1^{wi}$ for an unbounded $\mathcal{Z}$. In case $\Psi$ does not utilise any common string we assume $\mathsf{CV}(\mathbf{R},\mathsf{p},\varepsilon) = 1$.*

# 4 Verifiably-Extractable Generalized OWFs

## 4.1 Definition

We study GEOWFs $\mathcal{G} = \{\mathsf{g}_\mathsf{e}\}$ that come with an efficient (public) algorithm that decides whether or not extraction is going to be successful. That is, we require that there exists an extraction verification algorithm $\mathsf{EV}$, such that $\mathsf{EV}(\mathsf{e}, y)$ decides whether $y \in Y_\mathsf{Ext} \supseteq \mathrm{image}(\mathsf{g}_\mathsf{e})$, where extraction succeeds for any $y \in Y_\mathsf{Ext}$. We also require that, with overwhelming probability, extraction is successful for any adversary which outputs a value in $Y_\mathsf{Ext}$. (Extraction *may* succeed even if $y \notin Y_\mathsf{Ext}$.) We call GEOWFs with such properties *Verifiably-Extractable Generalized OWFs* (VEGOWFs).

Although for some VEGOWFs it may hold that $Y_\mathsf{Ext} = \mathrm{image}(\mathsf{g}_\mathsf{e})$, it is not necessarily the case. For example in the BCPR GEOWF, one is not able to decide if $y \in \mathrm{image}(\mathsf{g}_\lambda)$, because any such algorithm can be used to decide membership in $\mathrm{image}(\mathsf{PRG})$ which contradicts the security of $\mathsf{PRG}$. However, as we will show, extraction is successful for any $y = (\mathsf{crs}_\mathsf{DS}, v)$, where $\mathsf{crs}_\mathsf{DS}$ is a valid $\mathsf{DS}$ CRS and $v$ is *any* string output by an adversary with bounded auxiliary input.

Define VEGOWFs as GEOWFs where the $\mathbf{RG}$-extractability property has been substituted with the following, stronger one. (It makes an implicit assumption that $\mathsf{EV}$ exists.)

**RG-verifiably-extractable with respect to $Y_\mathsf{Ext}$:** Let $\mathrm{image}(\mathsf{g}_\mathsf{e}) \subseteq Y_\mathsf{Ext} \subseteq Y_\lambda$, and let $\mathsf{EV}$ be an efficient algorithm such that $\mathsf{EV}(\mathsf{e}; y) = 1$ iff $y \in Y_\mathsf{Ext}$. For any PPT adversary $\mathcal{A}$, there exists a PPT extractor $\mathsf{Ext}$, s.t. for any benign distribution $\mathcal{D}_\lambda$ of $\mathsf{poly}(\lambda)$-bit strings,

$$\Pr_{\substack{\mathsf{e} \leftarrow \mathsf{KeySamp}(1^\lambda) \\ \mathsf{aux} \leftarrow \mathcal{D}_\lambda}} \begin{bmatrix} y \leftarrow \mathcal{A}(\mathsf{e}; \mathsf{aux}), z \leftarrow \mathsf{Ext}(\mathsf{e}; \mathsf{aux}) : \\ y \in Y_\mathsf{Ext} \wedge (y, z) \notin \mathbf{RG}_\mathsf{e} \end{bmatrix} \leq \mathsf{negl}(\lambda) \enspace .$$

If this definition holds for adversaries with auxiliary input length bounded by some polynomial $\mathfrak{b}(\lambda)$, we say that that the GEOWF is $\mathbf{RG}$*-verifiably-extractable against $\mathfrak{b}$-bounded adversaries with respect to $Y_\mathsf{Ext}$*.

We also require that there is a PPT algorithm $t$, such that for any $x \in X_\lambda$, $(\mathsf{g}_\mathsf{e}(x), t(x)) \in \mathbf{RG}_\mathsf{e}$, that is, given $x$, $t$ computes the "witness" for $\mathsf{g}_\mathsf{e}(x)$ in $\mathbf{RG}$.

If there exists an algorithm $\mathsf{ImV}$ that decides membership in $\mathrm{image}(\mathsf{g}_\mathsf{e})$, then the GEOWF is *image-verifiable*. Clearly, any image-verifiable GEOWF is also verifiably-extractable with respect to $Y_\mathsf{Ext} = \mathrm{image}(\mathsf{g}_\mathsf{e})$. Furthermore, for an EOWF, $\mathbf{RG}_\mathsf{e}$ only consists of pairs $(\mathsf{g}_\mathsf{e}(x), x)$ so extraction is not possible if one is given $y \notin \mathrm{image}(\mathsf{g}_\mathsf{e})$. Hence, for an EOWF, verifiable-extractability is the same as image-verifiability.

## 4.2 Generic transformations

**VEGOWF ⇒ VEOWF.** Surprisingly, any VEGOWF can be transformed to a VEOWF with the transformation in Fig. 3 that adds very little overhead. The

```
$\mathsf{f_e}(i \in \{0,1\}^\lambda, x \in X_\lambda, y \in Y_\lambda, z \in X_\lambda)$
    if $i \neq 0^\lambda$ then return $\mathsf{g_e}(x)$;
    elseif $(y,z) \in \mathbf{RG}_\mathsf{e} \wedge \mathsf{EV_g}(\mathsf{e}; y)$ then return $y$;
    else return $\perp$;

$\mathsf{ImV_f}(\mathsf{e}; y)$
    return $\mathsf{EV_g}(\mathsf{e}; y) \vee y = \perp$;
```

**Fig. 3.** Transformation from a VEGOWF $\mathcal{G} = \{\mathsf{g_e}\}_\mathsf{e}$ to a VEOWF $\mathcal{F} = \{\mathsf{f_e}\}_\mathsf{e}$.

idea is to include to a VEGOWF $\mathsf{g_e}$ a branch input $i \in \{0,1\}^\lambda$. If $i \neq 0^\lambda$, which happens with an overwhelming probability, then $\mathsf{g_e}$ works as usual and outputs $\mathsf{g_e}(x)$. However, on a trapdoor branch $i = 0^\lambda$, the function uses its two extra inputs $y$ and $z$. If $y$ satisfies $\mathsf{EV_g}(\mathsf{e}; y)$ and $(y,z) \in \mathbf{RG}_\mathsf{e}$, it outputs $y$ (or $\perp$ if the condition is not met). One-wayness follows since with overwhelming probability the function outputs $y \in \text{image}(\mathsf{g_e})$ and the preimage has to contain either $x$ such that $\mathsf{g_e}(x) = y$ or $z$ such that $(y,z) \in \mathbf{RG}_\mathsf{e}$. By outputting either $t(x)$ (in the first case) or $z$ (in the other case), one breaks $\mathbf{RG}$-hardness. On the other hand, the VEOWF extractor can use the VEGOWF extractor to recover $z$ from $y$ when $\mathsf{EV_g}(\mathsf{e}; y) = 1$ and then return a preimage $(0^\lambda, \perp, y, z)$.

A similar transformation was introduced in [14] to obtain EOWFs from GEOWFs. However, they observed that an adversary can pick as input $(0^\lambda, \perp, y, z)$ with $(y,z) \in \mathbf{RG}_\mathsf{e}$, but $y \notin \text{image}(\mathsf{g_e})$. This makes the extraction impossible. Our construction does not run into this issue since we assume that extraction is possible when $\mathsf{EV}(\mathsf{e}; y) = 1$.

**Theorem 1.** *If $\mathcal{G} = \{\mathsf{g_e}\}_\mathsf{e}$ is $\mathbf{RG}$-hard and $\mathbf{RG}$-verifiably-extractable, then $\mathcal{F} = \{\mathsf{f_e}\}_\mathsf{e}$ in Fig. 3 is a VEOWF.*

**GEOWFs ⇒ VEGOWF.** We now consider a generic transformation from a GEOWF to a VEGOWF. One approach is to simply append a NIZK proof $\pi$ which proves that the given value was computed correctly. A problem with this approach is that it would require a CRS computed by a trusted third party, which might not be desirable in a number of settings. We therefore give a modification of this approach, where we instead rely on a NIWI, which are known to exist in the plain model under various assumptions [8,15,39].

The intuition is that we create a new function $g(x,y,r) = (f(x), f(y), \pi)$ where $\pi$ is a NIWI proof (created using randomness $r$) showing that either $f(x)$ or $f(y)$ belongs to the image of $f$ (in which case extraction will be possible). Verifiable-extractability follows from extractability of the GEOWF as well as perfect soundness of the NIWI, and hardness will follow from the hardness of $f$ and witness-indistinguishability of the NIWI.

Consider a GEOWF $\mathcal{F} = \{\mathsf{f_e}\}_\mathsf{e}$ with an associated relation $\mathbf{RG}$. Let $\Pi = (\mathsf{P}, \mathsf{V})$ be a perfectly sound NIWI, and let the relation $\mathbf{R}_\mathsf{e}((y_1, y_2), (x'_1, x'_2))$ hold iff $y_1 = \mathsf{f_e}(x'_1)$ or $y_2 = \mathsf{f_e}(x'_2)$. We define a VEGOWF $\mathcal{G} = \{\mathsf{g_e}\}_\mathsf{e}$ with an extraction verification algorithm $\mathsf{EV}$ in Fig. 4 and define the hardness relation:

$$\mathbf{RG}'_\mathsf{e}((y_1, y_2, \pi), (z_1, z_2)) := \mathbf{RG}_\mathsf{e}(y_1, z_1) \vee \mathbf{RG}_\mathsf{e}(y_2, z_2).$$

```
$\mathsf{g_e}(x_1, x_2, r)$
$y_1 \leftarrow \mathsf{f_e}(x_1); y_2 \leftarrow \mathsf{f_e}(x_2)$;
$\pi \leftarrow \mathsf{P}\left(\mathbf{R}_\mathsf{e}, (\mathsf{f_e}(x_1), \mathsf{f_e}(x_2)), (x_1, x_2); r\right)$;
else return $(y_1, y_2, \pi)$;

$\mathsf{EV}(\mathsf{e}; (y_1, y_2, \pi))$
return $\mathsf{V}(\mathbf{R}_\mathsf{e}, (y_1, y_2), \pi)$;
```

**Fig. 4.** Transformation from a GEOWF $\mathcal{F} = \{\mathsf{f_e}\}_\mathsf{e}$ to a VEGOWF $\mathcal{G} = \{\mathsf{g_e}\}_\mathsf{e}$.

Similar techniques have been used before in conjunction with EOWFs (e.g, 3-round ZK in [12]) but not, up to our knowledge, as a generic transformation. The proof of the following theorem is deferred to the full version of our paper.

**Theorem 2.** *If $\mathcal{F}$ is a GEOWF with respect to* $\mathbf{RG}$*, then $\mathcal{G}$ in Fig. 4 is a VEGOWF with respect to* $\mathbf{RG}'$*.*

**A robust combiner.** Additionally, a simple robust combiner is possible for VEGOWFs (or even for GEOWFs). Let us suppose that $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}_1}\}_{\mathsf{e}_1}$, $\mathcal{F} = \{\mathsf{f}_{\mathsf{e}_2}\}_{\mathsf{e}_2}$, and $\mathcal{H} = \{\mathsf{h}_{\mathsf{e}_3}\}_{\mathsf{e}_3}$ are candidate VEGOWFs for the respective relations $\mathbf{RG}^\mathsf{g}$, $\mathbf{RG}^\mathsf{f}$, and $\mathbf{RG}^\mathsf{h}$. We do assume that the associated extraction verification algorithm always accepts when given a value in the image of each candidate, but we make no other assumption about the security of the candidates.

We define a new VEGOWF $\mathcal{T} = \{\mathsf{t_e}\}_\mathsf{e}$ by $\mathsf{t_e}(x, y, z) := (\mathsf{g}_{\mathsf{e}_1}(x), \mathsf{f}_{\mathsf{e}_2}(y), \mathsf{h}_{\mathsf{e}_3}(z))$ where $\mathsf{e} = (\mathsf{e}_1, \mathsf{e}_2, \mathsf{e}_3)$ and the relation $\mathbf{RG}_\mathsf{e}$ is

$$\left\{ \begin{array}{l} \left((a,b,c),(z_1,z_2)\right) : \left((a,z_1) \in \mathbf{RG}^\mathsf{g}_{\mathsf{e}_1} \land (b,z_2) \in \mathbf{RG}^\mathsf{f}_{\mathsf{e}_2}\right) \lor \\ \left((a,z_1) \in \mathbf{RG}^\mathsf{g}_{\mathsf{e}_1} \land (c,z_2) \in \mathbf{RG}^\mathsf{h}_{\mathsf{e}_3}\right) \lor \left((b,z_1) \in \mathbf{RG}^\mathsf{f}_{\mathsf{e}_2} \land (c,z_2) \in \mathbf{RG}^\mathsf{h}_{\mathsf{e}_3}\right) \end{array} \right\} .$$

We define the new extraction verification algorithm to accept when all individual extraction verification algorithms accept.

If any two of the candidates are hard for their respective relations, then $\mathcal{T}$ is $\mathbf{RG}$-hard. Similarly, if any two are extractable, then $\mathcal{T}$ is $\mathbf{RG}$-extractable. The idea can be generalized to $n$ VEGOWFs for an arbitrary constant $n$, where it is sufficient that more than $n/2$ are secure. An interesting open question is to construct a robust combiner where fewer functions have to be secure.

### 4.3 The BCPR GEOWF is Verifiably-Extractable

We show that if a delegation scheme $\mathsf{DS}$ is CRS-verifiable, then the BCPR GEOWF from Fig. 2 is verifiably-extractable with respect to $Y_{\mathsf{Ext}} = \mathrm{image}(\mathsf{DS.K}(1^\lambda)) \times \{0,1\}^{\mathfrak{b}(\lambda)+\lambda}$. That is, $z$ contains the code of an adversary and the DS argument, independently of whether or not $y \in \mathrm{image}(\mathsf{g}_\lambda)$.

The proof of the following theorem is very similar to the proof of Theorem 14 from [14]; we have reproduced it for the sake of completeness.

**Theorem 3.** *Let* $\mathsf{DS}$ *be a delegation scheme that has publicly verifiable proofs and CRS, and let* $\mathsf{PRG} : \{0,1\}^\lambda \to \{0,1\}^{\mathfrak{b}(\lambda)+\lambda}$ *be a PRG. Let $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ and* $\mathbf{RG}$ *be as in Fig. 2. $\mathcal{G}$ is a VEGOWF for* $\mathbf{RG}$ *with respect to* $Y_{\mathsf{Ext}} = \mathrm{image}(\mathsf{DS.K}(1^\lambda)) \times \{0,1\}^{\mathfrak{b}(\lambda)+\lambda}$ *and* $(\mathfrak{b}(\lambda) - \omega(1))$*-bounded* $\mathsf{aux}$.

*Proof.* **RG-hardness.** Identical to the proof of Theorem 14 in [14].

**RG-verifiable-extractability.** Since $\mathsf{DS}$ is CRS-verifiable, there exists an algorithm $\mathsf{CV}$ which decides if $\mathsf{crs}_{\mathsf{DS}} \in \mathrm{image}(\mathsf{DS.K}(1^\lambda))$. On input $y = (\mathsf{crs}_{\mathsf{DS}}, v)$, the new extraction verification algorithm $\mathsf{EV}$ returns 1 if $\mathsf{CV}(\mathsf{crs}_{\mathsf{DS}}) = 1$ and $|v| = \mathfrak{b}(\lambda) + \lambda$.

We show that there is one universal PPT extractor $\mathsf{Ext}$ that can handle any PPT adversary $\mathcal{A}$ with advice of size at most $\mathfrak{b}(\lambda) - \omega(1)$. For an adversary $\mathcal{A}$ (a Turing machine) and advice $\mathsf{aux} \in \{0,1\}^{\mathfrak{b}(\lambda)-\omega(1)}$, denote by $\mathcal{A}_{\mathsf{aux}}$ the machine that, on input $1^\lambda$, runs $\mathcal{A}(1^\lambda; \mathsf{aux})$. Assume that (i) $\mathcal{A}_{\mathsf{aux}}$ has description size at most $\mathfrak{b}(\lambda)$ and that (i) on input $1^\lambda$, after at most $t_{\mathcal{A}} < T(\lambda)$ steps, it outputs $\mathcal{A}_{\mathsf{aux}}(1^\lambda) := y = (\mathsf{crs}_{\mathsf{DS}}, v) \in \{0,1\}^{\ell'(\lambda)}$. (Recall $Y_{\mathsf{Ext}} \subseteq \{0,1\}^{\ell'(\lambda)}$.) The extractor $\mathsf{Ext}(\mathcal{A}, \mathsf{aux}, 1^{t_{\mathcal{A}}})$ works as follows:

```
$\mathsf{Ext}(\mathcal{A}, \mathsf{aux}, 1^{t_{\mathcal{A}}})$
Construct $\mathcal{A}_{\mathsf{aux}}$;
$(\mathsf{crs}_{\mathsf{DS}}, v) \leftarrow \mathcal{A}_{\mathsf{aux}}(1^\lambda)$; if $\mathsf{EV}((\mathsf{crs}_{\mathsf{DS}}, v)) = 0$ then return $\perp$; fi ;
Compute a $\mathsf{DS}$-argument $\pi_{\mathsf{DS}}$ for the fact that "$\mathcal{A}_{\mathsf{aux}}(1^\lambda) = (\mathsf{crs}_{\mathsf{DS}}, v)$";
return $z \leftarrow (\mathcal{A}_{\mathsf{aux}}, \pi_{\mathsf{DS}}, \mathsf{pad})$;
```

It follows directly from the perfect completeness of $\mathsf{DS}$ that $\mathbf{RG}(y, z) = 1$. Since this holds for any $(\mathsf{crs}_{\mathsf{DS}}, v) \in Y_{\mathsf{Ext}}$ output by an adversary with $(\mathfrak{b}(\lambda) - \omega(1))$-bounded auxiliary input, we get $\mathbf{RG}$-verifiable-extractability. By the relative prover efficiency of the delegation scheme, the extractor's running time is polynomial in the running time $t_{\mathcal{A}}$ of the adversary. □

To instantiate the construction, we need a delegation scheme with public CRS and proof verification. Firstly, SNARKs in [1, 27, 51] satisfy both properties and have succinct proofs. All of them are based on non-falsifiable assumptions, however, here it is only needed that they are sound for the class $\mathsf{P}$. Thus, even a tautological security assumption (the corresponding SNARK is sound for $\mathsf{BPP}$) would be falsifiable. Secondly, some recent suggestions for delegation schemes [43,45] with public proof-verification are based on non-tautological falsifiable assumptions. Unfortunately, it is not immediately evident if those schemes also have CRS-verifiability. We leave the latter as an important open problem.

### 4.4 VEGOWFs from Knowledge-of-Exponent Assumptions

Next, we construct VEGOWFs based on knowledge-of-exponent (KE) assumptions, a logical direction partially motivated by [22, Section 3.3.1.1]. In each case, the key is a description $\mathsf{p}$ of an asymmetric or symmetric (in the latter case, we state it explicitly) bilinear group generated by a group generator algorithm $\mathsf{Pgen}(1^\lambda)$. Note that if the group generator $\mathsf{Pgen}$ is deterministic, i.e., each security parameter corresponds to a unique group, this is a keyless EOWF.

**The ABLZ VEOWF from BDH-KE.** The ABLZ VEOWF is based on an idea from Abdolmaleki *et al.* [1]. We define $\mathsf{g}_{\mathsf{p}}(x) := ([x]_1, [x]_2)$. The one-way property of the ABLZ EOWF is equivalent to the Symmetric Discrete Logarithm

(SDL) assumption, and extractability is equivalent to the BDH-KE assumption introduced in [1]. Finally, one can verify if $([x]_1, [y]_2) \in \mathrm{image}(\mathsf{g}_\mathsf{p})$ by checking that $[x]_1 \bullet [1]_1 = [1]_1 \bullet [y]_2$. We give a formal proof that this is a VEOWF in the full version of the paper. Note that this VEOWF is injective.

**VEGOWF from DH-KE.** Some KE assumptions from the literature lead to VEGOWFs rather than VEOWFs. The Diffie-Hellman KE (DH-KE) assumption introduced in [9] states that any adversary which produces a DDH triple $[x, y, xy]_1$ must know at least one of $x$ and $y$. Given a symmetric bilinear group, this gives rise to the following VEGOWF. Define $\mathsf{g}_\mathsf{p}(x, y) := [x, y, xy]_1$ and the relation $\mathbf{RG}_\mathsf{p}([x, y, xy]_1, z) = 1$ iff $z = x$ or $z = y$. We can verify if $[x, y, w]_1 \in \mathrm{image}(\mathsf{g}_\mathsf{p})$ by checking that $[x]_1 \bullet [y]_1 = [w]_1 \bullet [1]_1$. This function is $\mathbf{RG}$-hard if the discrete logarithm problem is hard and is verifiably-extractable if the DH-KE assumption holds.

**Further examples.** There are also a number of other knowledge of exponent assumptions in the literature, and these give rise to the following verifiably-extractable injective OWFs:

- $\mathsf{g}_{(\mathsf{p},[1,\alpha]_1)}(x) := [x, x\alpha]_1$ is a OWF under the discrete logarithm assumption and verifiably-extractable for symmetric pairings under the knowledge-of-exponent assumption [23].
- $\mathsf{g}_\mathsf{p}(x) = ([1, x, \dots, x^q]_1, [1, x, \dots, x^q]_2)$ is a OWF under the $q$-PDL assumption [48] and verifiably-extractable under the $q$-PKE assumption [24].
- $\mathsf{g}_\mathsf{p}(x) = ([x, x^2]_1, [x]_2)$ is a OWF under a well-known variant of the discrete logarithm assumption and verifiably-extractable under the square knowledge of exponent assumption [27].
- $\mathsf{g}_\mathsf{p}(x) = ([x]_1, [1/x]_2)$ is a OWF under the inverse-exponent assumption [54] and verifiably-extractable under the tautological assumption, which we call *inverse-KE*, that it is hard to compute $[x]_1$, $[1/x]_2$ without knowing $x$.

### 4.5 VEGOWFs from Knowledge-Sound NIZK

Dakdouk [22, Section 3.3.3.2] observed that EOWFs can be constructed from the proof of knowledge (PoK) assumption of Lepinski [47] which states that a specific non-interactive $\Sigma$-protocol described in [47] is secure. We generalize this idea, and show how to use knowledge-sound NIZKs to build VEGOWFs.

Suppose that $\mathbf{R}$ is an $\mathsf{NP}$ relation with a sampler $\mathcal{S}_{\mathbf{R},\mathsf{p}}$ that outputs $(\mathsf{x}, \mathsf{w})$, such that (i) it is efficient to verify that $(\mathsf{x}, \mathsf{w})$ is a possible output of $\mathcal{S}_{\mathbf{R},\mathsf{p}}$, and (ii) with an overwhelming probability it is computationally hard to guess $\mathsf{w}$ given $\mathsf{x}$. Then we say that this relation is $\mathcal{S}_{\mathbf{R},\mathsf{p}}$-hard. Such samplers (and relations) are common in cryptography, e.g., the discrete logarithm problem ($\mathsf{x} = [x]_1, \mathsf{w} = x$ for a uniformly random $x$) and the short integer solution problem ($\mathsf{x} = A$ is a random matrix and $\mathsf{w} = \vec{x}$ is a short integer vector such that $A\vec{x} = 0$).

Consider a knowledge-sound NIZK $\Pi = (\mathsf{KGen}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ for a $\mathcal{S}_{\mathbf{R},\mathsf{p}}$-hard relation $\mathbf{R}$, where $\mathsf{P}, \mathsf{V}, \mathsf{Sim}$ are the prover, the verifier, and the simulator. $\mathsf{KGen}$ is the "key" generation algorithm, such that $\mathsf{KGen}(\mathbf{R}, \mathsf{p})$ produces an auxiliary input $\mathsf{aux}_\Pi$, provided to $\mathsf{P}, \mathsf{V}$ and $\mathsf{Sim}$. If the NIZK uses a random oracle, then $\mathsf{aux}_\Pi$ may contain the description of a hash function instantiating the random oracle. If the NIZK is CRS-based, then $\mathsf{aux}_\Pi$ contains the CRS. The following theorem shows how to construct a VEGOWF given a knowledge-sound NIZK.

**Theorem 4.** *Define* $\mathcal{G} := \{\mathsf{g}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi}\}_{\mathbf{R}\in\mathcal{R}(1^\lambda),\mathsf{p}\leftarrow\mathsf{Pgen}(1^\lambda),\mathsf{aux}_\Pi\in\mathsf{KGen}(\mathbf{R},\mathsf{p})}$, *where* $\mathsf{g}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi}(r_\mathcal{S}, r_\Pi)$ *sets* $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}_\mathbf{R}(r_\mathcal{S})$, $\pi$ *produced by* $\Pi$*'s prover* $\mathsf{P}$ *for* $\mathsf{x}, \mathsf{w}$, *and then outputs* $(\mathsf{x}, \pi)$. *Define the corresponding relation as* $\mathbf{RG}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi} :=$

$$\{(\widehat{y}, \widehat{z}) : \widehat{y} = (\mathsf{x}, \pi) \wedge \widehat{z} = \mathsf{w} \wedge \Pi.\mathsf{V} \textit{ accepts } \pi \wedge (\mathsf{x}, \mathsf{w}) \in \mathbf{R}\} . \qquad (1)$$

*If* $\mathbf{R}$ *is* $\mathcal{S}_\mathbf{R}$*-hard and* $\Pi$ *is zero-knowledge, then* $\mathcal{G}$ *is* $\mathbf{RG}$*-hard. If* $\Pi$ *is a proof of knowledge, then* $\mathcal{G}$ *is* $\mathbf{RG}$*-verifiably-extractable.*

*Proof.* **RG-hardness:** Let $\mathcal{B}$ be an adversary that given $\widehat{y} = (\mathsf{x}, \pi)$, where $\pi$ is a proof for $(\mathsf{x}, \mathsf{w})$ returns $\widehat{z}$, such that $\mathbf{RG}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi}(\widehat{y}, \widehat{z})$ holds with non-negligible probability. We construct an adversary $\mathcal{B}$ that breaks $\mathcal{S}_\mathbf{R}$-hardness. On input $(\mathbf{R}, \mathsf{x})$, $\mathcal{B}$ sets $\mathsf{aux}_\Pi \leftarrow \mathsf{KGen}(\mathbf{R}, \mathsf{p})$, runs the simulator $\mathsf{Sim}$ and gets a simulated proof $\pi_\mathsf{Sim}$. Since $\Pi$ is zero-knowledge, $\mathcal{B}$ outputs the same $\widehat{z} = \mathsf{w}$ (with overwhelming probability) when run on $\widehat{y} = (\mathsf{x}, \pi)$ and $\widehat{y} = (\mathsf{x}, \pi_\mathsf{Sim})$. Thus, $\mathcal{B}$ breaks the $\mathcal{S}_{\mathbf{R},\mathsf{p}}$-hardness of $\mathbf{R}$ with non-negligible probability.

**RG-verifiable-extractability:** Clearly, one can verify that $\widehat{y} \in \mathrm{image}(\mathsf{g}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi})$ by checking that the NIZK verifier accepts $\widehat{y} = (\mathsf{x}, \pi)$, i.e., $\Pi$'s verifier accepts. We use the knowledge-soundness extractor $\mathsf{Ext}$ from $\Pi$ to build a $\mathcal{G}$ extractor $\mathsf{Ext}_\mathcal{G}$. Let $\mathcal{A}_\mathsf{ext}$ be an algorithm that on input $(\mathbf{R}, \mathsf{p}, \mathsf{aux}_\Pi)$ outputs $\widehat{y} \in \mathrm{image}(\mathsf{g}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi})$. Since $\widehat{y} \in \mathrm{image}(\mathsf{g}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi})$, then $\widehat{y} = (\mathsf{x}, \pi)$ and $\Pi$'s verifier accepts. $\mathsf{Ext}_\mathcal{G}$ runs $\mathsf{Ext}$ on the same input $(\mathbf{R}, \mathsf{p}, \mathsf{aux}_\Pi)$ given to $\mathcal{A}_\mathsf{ext}$. By knowledge-soundness, with an overwhelming probability, the $\Pi$-extractor $\mathsf{Ext}$ outputs $\mathsf{w}$, such that $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$. $\mathsf{Ext}_\mathcal{G}$ sets $\widehat{z} \leftarrow \mathsf{w}$, and succeeds with the same probability as $\mathsf{Ext}$. □

For the sake of concreteness we instantiate the above result as follows. Let $\Sigma$ be the non-interactive version (e.g., by using the Fiat-Shamir transform) of the well-known Schnorr's protocol for proving the knowledge of the discrete logarithm of $\mathsf{x} = [x]_1$. Let the VEGOWF key be $\mathsf{e} = (\mathbf{R}, \mathsf{p}, \mathsf{aux}_\Pi = H)$, where $\mathsf{p}$ is the system parameters (group description). Define $\mathsf{g}_\mathsf{e}(x, r) := ([x]_1, a = [r]_1, z = cx + r) = \widehat{y}$, where $c = H([x]_1, [r]_1)$. The verifier recomputes $c$ and accepts if $[z]_1 = c\mathsf{x} + a$ and $c = H(\mathsf{x}, a)$. Then $\mathbf{RG}_\mathsf{e}$-verifiable-extractability holds since $\Sigma$ is knowledge-sound in the random oracle model and the algebraic group model [29]. If $\Sigma$ is zero-knowledge in the random oracle model and the discrete logarithm problem is hard, $\mathsf{g}_\mathsf{e}$ is also $\mathbf{RG}_\mathsf{e}$-hard. Moreover, $\Sigma$ can be used to get an injective VEOWF since after the extractor extracts the witness $x$, it can also compute $r \leftarrow z - cx$.

### 4.6 VEGOWFs from Signature Schemes

We propose the following heuristic approach for finding new candidates for VEGOWFs. Suppose that $\Sigma = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Vf})$ is a digital signature scheme. If an adversary outputs $(\mathsf{vk}, \sigma)$ such that $\mathsf{vk} \in \mathsf{KGen}$ and $\mathsf{Vf}(\mathsf{vk}, \sigma, m = 0) = 1$, then there exists an extractor that can recover (some part of) $\mathsf{sk}$. In other words, we follow the intuition that if someone can sign a message (say $m = 0$ for simplicity), then she must possess the secret key. Moreover, if $\mathsf{vk} \in \mathsf{KGen}$ can be efficiently verified, then we might be able obtain a VEGOWF.

*Remark 1.* Note that unforgeability of a signature scheme does not require that the signer *knows* the secret key. It is only important that the adversary cannot produce valid signatures for previously unsigned messages. A stronger notion of knowledge has been formalized by signatures of knowledge [21], where the signer can sign messages under any statement $\mathsf{x} \in \mathcal{L}$ if it knows the corresponding witness. In general this is a very strong notion and implies, e.g., simulation-extractable NIZKs. Therefore, we will not focus on those constructions here.

There are signature schemes which do give believable VEGOWF candidates, but there are also cases where it clearly fails. We will mention some of them here, and defer others to the full version of our paper.

**Negative example: RSA signatures.** Let $H$ be a hash function, $\mathsf{sk} = d$ be the secret key and $\mathsf{vk} = (n, e)$ be a public key such that $de \equiv 1 \pmod{n}$. A signature of an integer $m$ is then $\sigma = H(m)^d \mod n$, and a signature $\sigma$ of a message $m$ is valid if $\sigma^e \equiv H(m) \pmod{n}$. However, RSA signatures are also not good candidates for a VEGOWF. The adversary could easily compute $\mathsf{vk} = (n, 3)$ such that $H(0) \mod n$ is a perfect cube, then output $(\mathsf{vk}, (H(0) \mod n)^{1/3})$.

**Positive example: Boneh-Boyen signatures.** Boneh-Boyen [16] is a pairing-based signature scheme where $\mathsf{vk} = [x]_2$ and $\mathsf{sk} = x \leftarrow_\$ \mathbb{Z}_p$ and $\mathsf{Sign}(\mathsf{sk}, m) = [1/(x + m)]_1$. In fact, $\mathsf{g}_\mathsf{p}(x) = (\mathsf{vk}, \mathsf{Sign}(0)) = ([x]_2, [1/x]_1)$ is an asymmetric version of a VEOWF candidate mentioned in Section 4.4. In particular, it is verifiably-extractable under a similar tautological assumption.

**Positive example: BLS signatures.** BLS [17] is another pairing-based signature scheme where $\mathsf{vk} = [x]_2$, $\mathsf{sk} = x \leftarrow_\$ \mathbb{Z}_p$, and $\mathsf{Sign}(\mathsf{sk}, m) = xH(m) = [\sigma]_1$ where $H$ hashes into $\mathbb{G}_1$. Verification is done by checking that $[\sigma]_1[1]_2 = H(m)[x]_2$. This gives us a VEOWF candidate $\mathsf{g}_\mathsf{p}(x) = ([x]_2, xH(0))$.

**Positive example: DSA.** In the DSA signature scheme,[9] we again have some discrete logarithm secure group $\mathsf{p} = (\mathbb{G}, p, g)$. The verification key is $\mathsf{vk} = g^x$ for $\mathsf{sk} = x \leftarrow_\$ \mathbb{Z}_p$, $\sigma = \mathsf{Sign}(\mathsf{sk}, M \in \{0,1\}^*; r) = (u = g^r \bmod p, v = r^{-1}(H_K(m) + xu) \bmod p)$, and the verifier checks that $0 < u, v < p$ and $u = (g^{H_K(M)}\mathsf{vk}^u)^{v^{-1}} \bmod p$. DSA results in a candidate VEOWF $\mathsf{g}_{\mathsf{p},K}(x, r) = (g^x, g^r \bmod p, r^{-1}(H_K(m) + xu) \bmod p)$.

**Hash-and-sign lattice signatures.** We recall hash-and-sign lattice-based signatures introduced by Gentry et al. [31], which relies on the hardness of the short integer solution problem. Let $p$ be a prime, $H$ be a hash function, and let $A \in \mathbb{Z}_p^{m \times n}$ be a randomly generated matrix. Define $L_p^\perp(A) := \{y | Ay = 0 \mod p\}$, and let $T$ be a basis of $L_p^\perp(A)$ with short vectors. The trapdoor can be used to compute short vectors $s$ s.t. $As = b$, for any vector $b$. Set $\mathsf{vk} = A$ and $\mathsf{sk} = T$.

[9] https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf

To sign a message $m$, one first computes $b = H(m)$, then outputs a short $s = \sigma_A(b)$ such that $As = b$. A signature $\sigma$ of a message $m$ is valid if it is short and if $A\sigma = H(m)$. However, this does not work as a VEGOWF. The adversary could easily compute $s$ with a nice structure (e.g., a unit vector), then choose $A$ such that $As = H(\vec{0})$. An easy fix is to set $b = H(A, m)$ to prevent choosing $A$ after setting $s$. This results in a candidate VEOWF $\mathsf{g}_p(x) = (A, \sigma_A(H(A, \vec{0})))$, where $x$ is a short basis of $L_p^{\perp}(A)$.

## 5 Sub-ZK NIZKs Based on VEGOWFs

We give a generic construction of a knowledge-sound Sub-ZK NIZK from any VEGOWF and any knowledge-sound Sub-WI NIWI in the CRS model. We also give a generic construction of a sound Sub-ZK NIZK from any VEGOWF, any keyless extractable commitment and any Sub-WI NIWI in the CRS model. Later, we show some interesting instantiations of these constructions.

### 5.1 Constructing Knowledge-sound Sub-ZK NIZK

Let $\mathcal{G} = \{\mathsf{g}_\lambda : X_\lambda \to Y_\lambda \mid \lambda \in \mathbb{N}\}$ be a keyless VEGOWF with respect to a publicly testable relation $\mathbf{RG}$ on triples $(1^\lambda, \widehat{y}, \widehat{z})$. We construct a knowledge-sound Sub-ZK NIZK $\Pi$ by using a knowledge-sound Sub-WI NIWI $\Pi_{wi}$ and $\mathcal{G}$. To prove that $\mathsf{x} \in \mathcal{L}$, we use $\Pi_{wi}$ to prove that $(\mathsf{x}, \widehat{y}) \in \mathcal{L}'$, where $\widehat{y} \in Y_{\mathsf{Ext}}$ is a new element in the CRS for $\Pi$, and $\mathbf{R}' := \{(\mathsf{x}_{\mathbf{R}'} = (\mathsf{x}, \widehat{y}), \mathsf{w}_{\mathbf{R}'} = (\mathsf{w}, \widehat{z})) : (\mathsf{x}, \mathsf{w}) \in \mathbf{R} \vee (\widehat{y}, \widehat{z}) \in \mathbf{RG}\}$ where $\mathcal{L} = \{\mathsf{x} \mid \exists \mathsf{w} : (\mathsf{x}, \mathsf{w}) \in \mathbf{R}\}$ and $\mathcal{L}' = \{\mathsf{x}_{\mathbf{R}'} \mid \exists \mathsf{w}_{\mathbf{R}'} : (\mathsf{x}_{\mathbf{R}'}, \mathsf{w}_{\mathbf{R}'}) \in \mathbf{R}'\}$. We assume that $\mathbf{R}$ is generated by a relation generator $\mathcal{R}(1^\lambda)$. The full construction of $\Pi$ can be found in Fig. 5.

The construction yields a knowledge-sound Sub-ZK NIZK, where knowledge-soundness follows from the $\mathbf{RG}$-hardness of $\mathcal{G}$ and the knowledge-soundness of $\Pi_{wi}$, and subversion zero-knowledge is achieved by the $\mathbf{RG}$-verifiable-extractability of $\mathcal{G}$ as well as the subversion witness-indistinguishability of $\Pi_{wi}$.

Note that if $\mathbf{R}$ is implemented by a circuit of size $k$ and $\mathbf{RG}$ is implemented by a circuit of size $l$, then the efficiency of $\Pi$ is the same as the efficiency of $\Pi'$ for the modified circuit of size $k + l$. Note also that $l$ is independent of $\mathbf{R}$. The proof of the following theorem is deferred to the full version of our paper.

**Theorem 5 (Knowledge-sound Sub-WI NIWI + VEGOWF $\Longrightarrow$ Knowledge-sound Sub-ZK NIZK).** *Let $\Pi_{wi}$ be a non-interactive argument for $\mathbf{R}'$ and let $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ be a keyless function family with a corresponding publicly testable relation $\mathbf{RG}$.*

*(1) If $\Pi_{wi}$ is complete then $\Pi$ is complete.*

*(2) If $\Pi_{wi}$ is knowledge-sound for $\mathbf{R}'$ and $\mathcal{G}$ is $\mathbf{RG}$-hard then $\Pi$ is knowledge-sound for $\mathbf{R}$.*

*(3) If $\Pi_{wi}$ is Sub-WI for $\mathbf{R}'$ and $\mathcal{G}$ is $\mathbf{RG}$-verifiably-extractable, then $\Pi$ is Sub-ZK for $\mathbf{R}$.*

```
$\mathsf{K}(\mathbf{R})$
    $\widehat{x} \leftarrow_{\$} X_\lambda$;
    $\widehat{y} \leftarrow \mathsf{g}_\lambda(\widehat{x})$
    $\mathsf{crs}' \leftarrow \mathsf{K}'(\mathbf{R})$
    $\mathsf{crs} \leftarrow (\mathsf{crs}', \widehat{y})$
    $\mathsf{td} \leftarrow t(\widehat{x})$
    return $(\mathsf{crs}, \mathsf{td})$

$\mathsf{CV}(\mathbf{R}, \mathsf{crs})$
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    if $\mathsf{CV}(\mathbf{R}', \mathsf{crs}') = 1 \wedge \widehat{y} \in Y_{\mathsf{Ext}}$
     then return $1$
     else return $0$

$\mathsf{Sim}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \mathsf{td})$
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    return $\mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, \widehat{y}), (\bot, \mathsf{td}))$

$\mathsf{P}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \mathsf{w})$
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$
    return $\pi \leftarrow \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, \widehat{y}), (\mathsf{w}, \bot))$;

$\mathsf{V}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \pi)$
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    return $\mathsf{V}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, \widehat{y}), \pi)$
```

**Fig. 5.** The Sub-ZK KS NIZK $\Pi = (\mathsf{K}, \mathsf{CV}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$, where $\Pi_{wi} = (\mathsf{K}', \mathsf{CV}', \mathsf{P}', \mathsf{V}')$ is a Sub-WI KS argument, and $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ is a VEGOWF. Recall that $t$ computes the "witness" for $\mathsf{g}_\lambda(\widehat{x})$ in $\mathbf{RG}$.

*(4) If $\Pi_{wi}$ is a Sub-WI SNARK and $\mathcal{G}$ is a VEGOWF with respect to a relation* $\mathbf{RG}$ *which takes inputs of polynomial size, then $\Pi$ is a Sub-ZK SNARK.*

### 5.2 Constructing Sub-ZK NIZK

Next, we propose a Sub-ZK NIZK $\Pi$ which only relies on $\Pi_{wi}$ being sound, not knowledge-sound, but $\Pi$ will also not be knowledge-sound. As part of this construction, we rely on a keyless extractable commitment scheme. We now give the definition of a keyless extractable commitment scheme, and in the full version of our paper we show how this can be constructed based on injective EOWFs.

**Definition 10.** *We say that $\mathsf{Com}_\lambda \colon \mathcal{M}_\lambda \times \mathcal{R}_\lambda \to \mathcal{C}_\lambda$ is a keyless extractable commitment if it satisfies the following properties.*

**Computational hiding:** *For any PPT adversary $\mathcal{A}$, $\varepsilon_0 \approx_\lambda \varepsilon_1$, where*

$$\varepsilon_b := \Pr\left[\begin{matrix}(m_1, m_2) \leftarrow \mathcal{A}(1^\lambda), r \leftarrow_{\$} \mathcal{R}_\lambda, c \leftarrow \mathsf{Com}_\lambda(m_b; r) : \\ m_1, m_2 \in \mathcal{M}_\lambda \wedge \mathcal{A}(c) = 1\end{matrix}\right] \ .$$

**Perfect binding:** *For any adversary $\mathcal{A}$ and $\lambda \in \mathbb{N}$,*

$$\Pr\left[\begin{matrix}(m_1, r_1, m_2, r_2) \leftarrow \mathcal{A}(1^\lambda) : \\ \mathsf{Com}_\lambda(m_1; r_1) = \mathsf{Com}_\lambda(m_2; r_2) \wedge m_1 \neq m_2\end{matrix}\right] = 0 \ .$$

**Non-black-box extractability:** *Let $\mathcal{D}$ be a family $\{D_\lambda\}_\lambda$ of efficiently sampleable distributions. We say that $\mathsf{Com}_\lambda \colon \mathcal{M}_\lambda \times \mathcal{R}_\lambda \to \mathcal{C}_\lambda$ is non-black-box extractable with respect to auxiliary distribution $\mathcal{D}$ if for any PPT adversary $\mathcal{A}$, there exists a PPT extractor $\mathsf{Ext}_\mathcal{A}$ such that,*

$$\Pr\left[\begin{matrix}\mathsf{aux} \leftarrow_{\$} D_\lambda, c \leftarrow \mathcal{A}(1^\lambda, \mathsf{aux}), m \leftarrow \mathsf{Ext}_\mathcal{A}(1^\lambda, \mathsf{aux}), \\ c \in \mathrm{image}(\mathsf{Com}_\lambda) : c = \mathsf{Com}_\lambda(m; r) \textit{ for some } r \in \mathcal{R}_\lambda;\end{matrix}\right] \geq 1 - \mathsf{negl}(\lambda) \ .$$

```
K(R)
    $\widehat{x} \leftarrow_\$ X_\lambda$;
    $\widehat{y} \leftarrow \mathsf{g}_\lambda(\widehat{x})$;
    $\mathsf{crs}' \leftarrow \mathsf{K}'(\mathbf{R}')$;
    $\mathsf{crs} \leftarrow (\mathsf{crs}', \widehat{y})$;
    $\mathsf{td} \leftarrow t(\widehat{x})$;
    return $(\mathsf{crs}, \mathsf{td})$;

CV(R, crs)
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    if $\mathsf{CV}'(\mathbf{R}', \mathsf{crs}') = 1 \wedge y \in Y_{\mathsf{Ext}}$;
        then return $1$
        else return $0$

Sim(R, crs, td = $\widehat{z}$, x)
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    $r \leftarrow_\$ \mathsf{RND}_\lambda(\mathsf{Com})$;
    $c \leftarrow \mathsf{Com}(\widehat{z}; r)$;
    $\pi' \leftarrow \mathsf{P}(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c, \widehat{y}), (\perp, \widehat{z}, r))$;
    return $\pi \leftarrow (c, \pi')$

P(R, crs, x, w)
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    $r \leftarrow \mathsf{RND}_\lambda(\mathsf{Com})$;
    $c \leftarrow \mathsf{Com}(x_\lambda; r)$ where $x_\lambda \leftarrow_\$ X_\lambda$;
    $\pi' \leftarrow \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c, \widehat{y}), (\mathsf{w}, x_\lambda, r))$;
    return $\pi \leftarrow (c, \pi')$;

V(R, crs, x, $\pi$)
    parse $\pi = (c, \pi')$;
    parse $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$;
    return $\mathsf{V}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c, \widehat{y}), \pi')$;
```

**Fig. 6.** The Sub-ZK NIZK $\Pi = (\mathsf{K}, \mathsf{CV}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$, where $\Pi_{wi} = (\mathsf{K}', \mathsf{CV}', \mathsf{P}', \mathsf{V}')$ is a Sub-WI NIWI, $\mathsf{C}$ is an extractable commitment scheme, and $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ is a GEOWF.

*In some cases, we may have an efficient commitment verification function* $\mathsf{ComV}_\lambda$ *that outputs 1 on input c if and only if* $c \in \mathrm{image}(\mathsf{Com}_\lambda)$.

Let $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ be a function family with associated relation $\mathbf{RG}$. Let $\mathsf{C} = (\mathsf{Com}, \mathsf{Open}, \mathsf{Vf})$ be an extractable commitment scheme. Let $\Pi_{wi}$ be a NIWI argument for the relation We set $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$, where $\mathsf{crs}'$ is the CRS of the underlying NIWI $\Pi_{wi}$ for $\mathbf{R}'$ and $\mathsf{crs}$ is the CRS of the NIZK for $\mathbf{R}$. The argument consists of the commitment $c$ and the $\Pi_{wi}$-argument $\pi$; see Fig. 6. The proof of the following theorem is deferred to the full version of our paper.

**Theorem 6 (Sub-WI NIWI + VEGOWF + ExtCom $\Longrightarrow$ Sub-ZK NIZK).** *Let $\Pi_{wi}$ be a non-interactive argument, $\mathsf{C}$ be a commitment scheme, and $\mathcal{G}$ be a function family with associated publicly testable relation $\mathbf{RG}$.*

*(1) If $\Pi_{wi}$ is perfectly complete then $\Pi$ is perfectly complete.*

*(2) If $\Pi_{wi}$ is sound, $\mathsf{C}$ is keyless and extractable, and $\mathcal{G}$ is $\mathbf{RG}$-hard then $\Pi$ is sound.*

*(3) If $\Pi_{wi}$ is Sub-WI, $\mathcal{G}$ is $\mathbf{RG}$-verifiably-extractable, and $\mathsf{C}$ is keyless and hiding, then $\Pi$ is Sub-ZK.*

### 5.3 Instantiations and Statistical ZAPR

We show some interesting instantiations of the above construction and also make a simple, but significant, connection between Sub-ZK NIZK and ZAPs with private random coin (ZAPRs).

Firstly, we argue that there is a knowledge-sound Sub-ZK NIZK based on the DLin and DH-KE assumptions. To the best of our knowledge, the only known knowledge-sound Sub-ZK NIZKs are Sub-ZK SNARKs. Our construction therefore relies arguably on weaker assumptions.

**Proposition 2.** *There exists a knowledge-sound Sub-ZK NIZK based on the DLin and DH-KE assumptions with 3 group elements as the CRS and with a proof size of $\mathcal{O}(\lambda(k+l))$ where $k$ is the circuit size and $l$ is size of a circuit verifying the image of the DH-KE GEOWF.*

*Proof.* In [28] it is proven that there exists a knowledge-sound NIWI in the plain model based on the DLin and DH-KE assumptions. Since it has no CRS, it is also Sub-WI. From Section 4.4, there exists a VEGOWF based on the DH-KE and discrete logarithm (DL) assumptions (note that DLIN implies DL). We now apply our construction in Fig. 5 using the knowledge-sound NIWI from [28] and the VEGOWF from Section 4.4. It then follows from Theorem 5 that the resulting protocol is a knowledge-sound Sub-ZK NIZK. □

Let us next prove a helpful lemma that shows when NIWI is Sub-WI. The corollary follows since perfect zero knowledge implies perfect WI.

**Lemma 1.** *Suppose $\Psi$ is perfectly WI for relation $\mathbf{R}$ and there exists an efficient CRS validation algorithm $\mathsf{CV}$. Then $\Psi$ is Sub-WI.*

*Proof.* Definition 8 for perfect WI states that for all honestly generated CRS $\mathsf{crs}$ (i.e., CRS in the image of $\mathsf{K}(\mathbf{R})$), instances $\mathsf{x}$, and corresponding witnesses $\mathsf{w}_0, \mathsf{w}_1$, no unbounded adversary can distinguish a proof generated using either $(\mathsf{crs}, \mathsf{x}, \mathsf{w}_0)$ or $(\mathsf{crs}, \mathsf{x}, \mathsf{w}_1)$. Note that if a subverter can create a valid $\mathsf{crs}$ such that $\mathcal{A}$ breaks Sub-WI with probability at least $\varepsilon > 0$, the same $\mathcal{A}$ can break WI with probability at least $\varepsilon/(|\mathsf{crs}|+|\mathsf{aux}_{\mathcal{Z}}|) > 0$ by simply guessing $\mathsf{crs}$ and $\mathsf{aux}_{\mathcal{Z}}$. Hence assuming perfect WI, verifying that a subverter-generated CRS $\mathsf{crs}$ is in fact in the image of $\mathsf{K}(\mathbf{R})$ is enough to assure that perfect subversion WI holds. □

**Corollary 1.** *If $\Psi$ is perfectly zero-knowledge and there exist an efficient CRS validation algorithm, then $\Psi$ is Sub-WI.*

Therefore, the efficient SNARK constructions in [1, 27], the updatable SNARKs in [38, 50], and the shuffle argument in [4] are all Sub-WI. The same observation about Sub-ZK SNARKs was already made by Fuchsbauer in [27]. These arguments have a CRS validation algorithm and were already known to be Sub-ZK under a knowledge assumption. However, the above result shows that they are perfect Sub-WI *without any assumptions*. Moreover, any NIWI without a CRS is trivially Sub-WI.

Firstly, it means that [1, 27] are statistical ZAPRs with adaptive soundness. The only other pairing-based ZAPR is [49] which is less efficient and uses much more advanced tools, but relies on weaker assumptions for soundness. Secondly, if we use the SNARKs of [1, 27] in Fig. 5, we have Sub-ZK SNARKs from any VEGOWF rather than from a specific knowledge assumption.

**Proposition 3.** *Suppose there exists a perfectly zero-knowledge SNARK with an efficient CRS validation algorithm $\mathsf{CV}$ and there exists a VEGOWF. Then there exists a Sub-ZK SNARK.*

*Proof.* Since the given SNARK $\Pi$ is perfectly ZK and has a $\mathsf{CV}$ algorithm, it follows from Corollary 1 that it is perfectly Sub-WI. Applying our construction in Section 5.1 to $\Pi$ and the VEGOWF $\mathcal{G}$ to construct a new SNARK $\Pi'$, it then follows from part (4) of Theorem 5 that $\Pi'$ is a Sub-ZK SNARK, as desired. □

## 6 Characterising Sub-ZK NIZKs

We show that the CRS generation algorithm $\mathsf{K}$ of a NIZK is a VEOWF if and only if the NIZK is Sub-ZK. Let $\mathcal{R}$ be a relation generator, and let $\Pi = (\mathsf{K}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ be a NIZK argument for $\mathcal{R}$. We define a family of functions $\mathcal{G}_{\mathsf{K}} = \left\{\mathsf{K}_{\mathbf{R},\mathsf{p}} \colon \{\mathsf{td}\} \to \{\mathsf{crs}\} \mid (\mathbf{R}, \mathsf{p}) \in \mathcal{R}(1^\lambda), \lambda \in \mathbb{N}\right\}$ where $\mathsf{K}_{\mathbf{R},\mathsf{p}}$ takes in a uniformly sampled trapdoor $\mathsf{td}$ and maps it deterministically to a $\mathsf{crs}$. We assume that the distribution $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{KGen}(\mathbf{R}, \mathsf{p})$ is the same as $(\mathsf{crs} \leftarrow \mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}), \mathsf{td} \leftarrow\$ \{\mathsf{td}\})$. We use both notations interchangeably in this section.

Let us start by establishing the following straightforward connection.

**Theorem 7 (VEOWF $\mathcal{G}_{\mathsf{K}} \implies$ Sub-ZK).** *Suppose* $\Pi = (\mathsf{K}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ *is a perfect NIZK argument. If* $\mathcal{G}_{\mathsf{K}}$ *is a VEOWF with image verification algorithm* $\mathsf{ImV}$*, then* $\Pi$ *is statistically composable Sub-ZK with respect to the CRS verification algorithm* $\mathsf{CV} = \mathsf{ImV}$.

*Proof.* Consider a subverter $\mathcal{Z}$ which outputs a CRS $\mathsf{crs}$. We only need to consider the case where $\mathsf{CV}(\mathsf{crs}) = 1$ and thus $\mathsf{crs} \in \mathrm{image}(\mathsf{K}_{\mathbf{R},\mathsf{p}})$. Since $\mathsf{K}_{\mathbf{R},\mathsf{p}}$ is a VEOWF and the subverter $\mathcal{Z}$ outputs an image of $\mathsf{K}_{\mathbf{R},\mathsf{p}}$, we know that there exists an extractor $\mathsf{Ext}_{\mathcal{Z}}$ which with overwhelming probability outputs a simulation trapdoor $\mathsf{td}$. Since $\Pi$ is perfect zero-knowledge, proofs $\pi_0 \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{td}, \mathsf{crs}, \mathsf{x})$ and $\pi_1 \leftarrow \mathsf{P}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \mathsf{w})$ are identically distributed. □

*Remark 2.* The same result does not hold for statistical (or computational) NIZK since there might be a negligible number of CRSs where $\mathsf{td}$ does not allow simulation, which the subverter could output.

Following [37], we say that the relation generator $\mathcal{R}$ *has a* $\varepsilon_{\mathcal{S}}$*-hard decisional problem* if there exist two samplers $\mathcal{S}$ and $\mathcal{S}'$ such that for $(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$ (1) sampler $\mathcal{S}(\mathbf{R}, \mathsf{p})$ produces $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$, and (2) $\mathcal{S}'(\mathbf{R}, \mathsf{p})$ produces $\mathsf{x} \notin \mathcal{L}_{\mathbf{R}}$. Furthermore, for some negligible $\varepsilon_{\mathcal{S}}$, it holds for all PPT adversaries $\mathcal{A}$ that $|\varepsilon_0 - \varepsilon_1| \leq \varepsilon_{\mathcal{S}}$, where $\varepsilon_b = \Pr\left[(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda), (\mathsf{x}_0, \mathsf{w}_0) \leftarrow \mathcal{S}(\mathbf{R}, \mathsf{p}), \mathsf{x}_1 \leftarrow \mathcal{S}'(\mathbf{R}, \mathsf{p}) : \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{x}_b) = 1\right]$.

A simple example of this is the language of Diffie-Hellman tuples where $\mathsf{p} = (\mathbb{G}, g, p) \leftarrow \mathcal{R}(1^\lambda)$ is a group description, $\mathcal{S}$ outputs $(\mathsf{x} = (g^x, g^y, g^{xy}), \mathsf{w} = (x, y))$ for random $x, y \leftarrow\$ \mathbb{Z}_p$, and $\mathcal{S}'$ outputs $g^x, g^y, g^z$ for random $x, y \leftarrow\$ \mathbb{Z}_p$ and $z \leftarrow\$ \mathbb{Z}_p \setminus \{xy\}$.

Now let us establish the opposite connection between VEOWF and Sub-ZK. In general, the extractor in subversion zero-knowledge definition does not need to extract the whole preimage of the CRS function. It just needs to extract something which allows for simulation of proofs. For example, this could be only a small part of the full trapdoor. Due to this, we restrict ourselves slightly and lend the following notion from [3].

**Definition 11 (Trapdoor-Extractability [3]).** *A subversion-resistant argument $\Psi$ for a relation $\mathcal{R}$ has trapdoor-extractability if for any PPT subverter $\mathcal{Z}$ there exists a PPT extractor $\mathsf{Ext}_{\mathcal{Z}}$, s.t. for all $\lambda$ and $(\mathbf{R}, \mathsf{p}) \in \mathcal{R}(1^\lambda)$,*

$$\Pr\left[\begin{matrix} r \leftarrow_\$ \mathsf{RND}_\lambda(\mathcal{Z}), \mathsf{crs} \leftarrow \mathcal{Z}(\mathbf{R}, \mathsf{p}; r), \mathsf{td} \leftarrow \mathsf{Ext}_{\mathcal{Z}}(\mathbf{R}, \mathsf{p}; r) : \\ \mathsf{CV}(\mathbf{R}, \mathsf{p}, \mathsf{crs}) = 1 \wedge \mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}) \neq \mathsf{crs} \end{matrix}\right] \leq \mathsf{negl}(\lambda) \enspace .$$

**Theorem 8 (Sub-ZK $\Longrightarrow$ VEOWF $\mathcal{G}_\mathsf{K}$).** *Assume $\Pi$ is a NIZK argument for $\mathcal{R}$, which has $\varepsilon_{\mathcal{S}}$-hard decisional problems. Let $\mathcal{G}_\mathsf{K}$ be as defined above. Assume the distribution $\mathcal{D}_\lambda$ is benign. Then*

1. *if (i) $\Pi = (\mathsf{K}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ is perfectly complete, computationally sound, and computationally zero-knowledge, and (ii) $\mathsf{K}_{\mathbf{R},\mathsf{p}}$ is injective, then $\mathcal{G}_\mathsf{K}$ is a one-way function;*
2. *if $\Pi = (\mathsf{K}, \mathsf{P}, \mathsf{V}, \mathsf{Sim}, \mathsf{CV})$ is a statistically composable Sub-ZK argument with trapdoor extractability, then $\mathcal{G}_\mathsf{K}$ is verifiably-extractable with $\mathcal{G}_\mathsf{K}.\mathsf{ImV} = \Pi.\mathsf{CV}$ respect to auxiliary inputs $(\mathbf{R}, \mathsf{p}, r)$ where $(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$, $r \leftarrow_\$ \{0,1\}^{\mathsf{poly}(\lambda)}$.*

*Proof.* **Soundness + ZK $\Longrightarrow$ One-Wayness.** Suppose there exists a PPT adversary $\mathcal{A}$ that breaks one-wayness of $\mathcal{G}_\mathsf{K}$ with probability $\varepsilon_{owf}$. That is, for a random $(\mathbf{R}, \mathsf{p}) \leftarrow \mathsf{KeySamp}_{\mathcal{G}}(1^\lambda)$, $\mathsf{td} \leftarrow_\$ \{\mathsf{td}\}, \mathsf{aux} \leftarrow_\$ \mathcal{D}_\lambda$, the $\mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs} = \mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}), \mathsf{aux})$ outputs $\mathsf{td}'$ such that $\mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}') = \mathsf{crs}$ with probability $\varepsilon_{owf}$.

We are going to construct a PPT adversary $\mathcal{B}$ that internally runs $\mathcal{A}$ together with an auxiliary input $\mathsf{aux}$. We build the soundness adversary $\mathcal{B}$ as follows:

1. $\mathcal{B}$ gets $(\mathbf{R}, \mathsf{p}, \mathsf{crs})$ as an input;
2. $\mathcal{B}$ samples $\mathsf{aux}' \leftarrow_\$ \mathcal{D}_\lambda$ and computes $\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$;
3. $\mathcal{B}$ outputs $\mathsf{x}$ such that $\mathsf{x} \leftarrow \mathcal{S}'(\mathbf{R}, \mathsf{p})$ (i.e. $\mathsf{x} \notin \mathcal{L}_\mathbf{R}$) along with a simulated proof $\pi \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}', \mathsf{x})$.

Since $\mathsf{x} \notin \mathcal{L}_\mathbf{R}$ by definition, it means that $\mathcal{B}$ wins the soundness game if $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1$. We use games in Fig. 7 to quantify the probability that $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1$ in the soundness game.

$\boxed{\mathsf{Game\ 0}:}$ This is the original soundness game without the condition $\mathsf{x} \notin \mathcal{L}_\mathbf{R}$ with the adversary $\mathcal{B}$ inlined. The winning condition is just $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1$.

$\boxed{\mathsf{Game\ 1}:}$ We change $\mathsf{Game}$ 0 such that $\mathcal{B}$ samples a true statement-witness pair $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(\mathbf{R}, \mathsf{p})$ instead.

$\boxed{\mathsf{Game\ 2}:}$ We modify $\mathsf{Game}$ 1 such that the simulator gets the real trapdoor $\mathsf{td}$ as an input rather than the trapdoor $\mathsf{td}'$ extracted by $\mathcal{A}$.

$\boxed{\mathsf{Game\ 3}:}$ Finally, instead of simulating the proof $\pi$, we use the witness $\mathsf{w}$ to create an honest proof.

Let us denote the probability of $\mathsf{Game}$ $i$ outputting 1 by $\varepsilon_i$. Firstly, it is clear that $\varepsilon_0$ is the probability of $\mathcal{B}$ winning (that is, outputting 1) in the soundness game since, although, we do not check the condition $\mathsf{x} \notin \mathcal{L}_\mathbf{R}$, it always holds for the adversary $\mathcal{B}$. We now prove that distinguishing $\mathsf{Game}$ 0 and $\mathsf{Game}$ 1 succeeds with probability at most $\varepsilon_{\mathcal{S}}$.

**Lemma 2.** *For the probabilities $\varepsilon_0$ and $\varepsilon_1$ defined as above, $|\varepsilon_0 - \varepsilon_1| \leq \varepsilon_{\mathcal{S}}$.*

```
Game 0:
$(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$;
$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p})$; $\mathsf{aux}' \leftarrow\!\!\$\, \mathcal{D}_\lambda$;
$\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$;
$\mathsf{x} \leftarrow \mathcal{S}'(\mathbf{R}, \mathsf{p})$;
$\pi \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}', \mathsf{x})$;
return $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$;
```

```
Game 1:
$(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$;
$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p})$; $\mathsf{aux}' \leftarrow\!\!\$\, \mathcal{D}_\lambda$;
$\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$;
$(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(\mathbf{R}, \mathsf{p})$;
$\pi \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}', \mathsf{x})$;
return $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$;
```

```
Game 2:
$(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$;
$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p})$; $\mathsf{aux}' \leftarrow\!\!\$\, \mathcal{D}_\lambda$;
$\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$;
$(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(\mathbf{R}, \mathsf{p})$;
$\pi \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}, \mathsf{x})$;
return $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$;
```

```
Game 3:
$(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$;
$(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p})$; $\mathsf{aux}' \leftarrow\!\!\$\, \mathcal{D}_\lambda$;
$\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$;
$(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(\mathbf{R}, \mathsf{p})$;
$\pi \leftarrow \mathsf{P}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \mathsf{w})$;
return $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$;
```

**Fig. 7.** Security games for Theorem 8.

*Proof.* Consider the following adversary $\mathcal{C}$ against the $\varepsilon_{\mathcal{S}}$-hardness. Firstly, $\mathcal{C}$ gets as an input $(\mathbf{R}, \mathsf{p}, \mathsf{x}_b)$ where $\mathsf{x}_1$ is generated by $\mathcal{S}$ and $\mathsf{x}_0$ is generated by $\mathcal{S}'$. Then, $\mathcal{C}$ samples $(\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p})$ and $\mathsf{aux}' \leftarrow\!\!\$\, \mathcal{D}_\lambda$, computes $\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux}')$, and simulates the proof $\pi \leftarrow \mathsf{Sim}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}', \mathsf{x})$. It returns the answer of $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi)$.

By construction, the probability that $\mathcal{C}$ outputs 1 given $\mathsf{x}_0$ is $\varepsilon_0$ and given $\mathsf{x}_1$ is $\varepsilon_1$. It thus follows that $|\varepsilon_0 - \varepsilon_1| \leq \varepsilon_{\mathcal{S}}$. □

**Lemma 3.** *Assuming that $\mathsf{K}_{\mathbf{R},\mathsf{p}}$ is injective, $|\varepsilon_1 - \varepsilon_2| \leq 1 - \varepsilon_{owf}$.*

*Proof.* The only difference between Game 1 and Game 2 is that one uses $\mathsf{td}'$ for simulation and the other uses $\mathsf{td}$. If $\mathcal{A}$ is successful in breaking one-wayness, then $\mathsf{td} = \mathsf{td}'$ (since $\mathsf{K}_{\mathbf{R},\mathsf{p}}$ is injective) and output distributions of both games are the same. That happens with probability $\varepsilon_{owf}$. Outputs distributions of games can differ only when $\mathcal{A}$ fails in breaking one-wayness, which happens at most with the probability $1 - \varepsilon_{owf}$. We conclude that $|\varepsilon_1 - \varepsilon_2| \leq 1 - \varepsilon_{owf}$. □

**Lemma 4.** *Let $\varepsilon_{zk}$ denote the maximum advantage that any PPT adversary wins in the zero-knowledge game. Then, $|\varepsilon_2 - \varepsilon_3| \leq \varepsilon_{zk}$.*

*Proof.* Consider the verifier $\mathsf{V}$ as the adversary in the zero-knowledge game. From this perspective Game 2 is the zero-knowledge game with the simulator and Game 3 is the zero-knowledge game with the honest prover given that we ignore the line $\mathsf{td}' \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{aux})$. It follows that $|\varepsilon_2 - \varepsilon_3| \leq \varepsilon_{zk}$. □

Using the triangle inequality, we now get that $|\varepsilon_0 - \varepsilon_3| \leq \varepsilon_{\mathcal{S}} + (1 - \varepsilon_{owf}) + \varepsilon_{zk}$. Since the argument system is perfectly complete, $\varepsilon_3 = 1$ and therefore $|\varepsilon_0 - \varepsilon_3| =$

$|\varepsilon_0 - 1| = 1 - \varepsilon_0$. Putting equations together, we get $1 - \varepsilon_0 \leq \varepsilon_{\mathcal{S}} + (1 - \varepsilon_{owf}) + \varepsilon_{zk}$, which can be simplified to $\varepsilon_{owf} \leq \varepsilon_0 + \varepsilon_{\mathcal{S}} + \varepsilon_{zk}$, which is negligible. □

**Sub-ZK $\Longrightarrow$ verifiable-extractability.** This part of the proof is essentially tautological. Let $\mathcal{A}$ be an adversary in the verifiable extractability game and let $\mathsf{aux} = (\mathbf{R}, \mathsf{p}, r)$ where $(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda)$ and $r \leftarrow_\$ \{0, 1\}^{\mathsf{poly}(\lambda)}$. Suppose that $\mathcal{A}$ is Sub-ZK subverter that outputs $\mathsf{crs}$ such that $\mathsf{CV}(\mathbf{R}, \mathsf{p}, \mathsf{crs}) = 1$. Then according to the trapdoor extractability property, there exists a PPT extractor $\mathsf{Ext}_{\mathcal{A}}$ that on input $\mathsf{aux}$, outputs with an overwhelming $\mathsf{td}$ such that $\mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}) = \mathsf{crs}$. Thus, verifiable extractability holds. □

**Acknowledgements.** Janno Siim and Helger Lipmaa were partially supported by the Estonian Research Council grant (PRG49).

## References

1. Abdolmaleki, B., Baghery, K., Lipmaa, H., Zajac, M.: A subversion-resistant SNARK. In: ASIACRYPT 2017, Part III. LNCS, vol. 10626, pp. 3–33
2. Abdolmaleki, B., Lipmaa, H., Siim, J., Zajac, M.: On QA-NIZK in the BPK model. In: PKC 2020, Part I. LNCS, vol. 12110, pp. 590–620
3. Abdolmaleki, B., Lipmaa, H., Siim, J., Zając, M.: On subversion-resistant SNARKs. Cryptology ePrint Archive, Report 2020/668 (2020) `https://eprint.iacr.org/2020/668`.
4. Aggelakis, A., Fauzi, P., Korfiatis, G., Louridas, P., Mergoupis-Anagnou, F., Siim, J., Zajac, M.: A non-interactive shuffle argument with low trust assumptions. In: CT-RSA 2020. LNCS, vol. 12006, pp. 667–692
5. Aiello, W., Bhatt, S.N., Ostrovsky, R., Rajagopalan, S.: Fast verification of any remote procedure call: Short witness-indistinguishable one-round proofs for NP. In: ICALP 2000. LNCS, vol. 1853, pp. 463–474
6. Badrinarayanan, S., Fernando, R., Jain, A., Khurana, D., Sahai, A.: Statistical ZAP arguments. In: EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 642–667
7. Barak, B.: How to go beyond the black-box simulation barrier. In: 42nd FOCS, pp. 106–115
8. Barak, B., Ong, S.J., Vadhan, S.P.: Derandomization in cryptography. In: CRYPTO 2003. LNCS, vol. 2729, pp. 299–315
9. Bellare, M., Fuchsbauer, G., Scafuro, A.: NIZKs with an untrusted CRS: Security in the face of parameter subversion. In: ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 777–804
10. Bellare, M., Goldreich, O.: On defining proofs of knowledge. In: CRYPTO'92. LNCS, vol. 740, pp. 390–420
11. 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
12. Bitansky, N., Canetti, R., Chiesa, A., Goldwasser, S., Lin, H., Rubinstein, A., Tromer, E.: The hunting of the SNARK. Journal of Cryptology **30**(4) (2017) pp. 989–1066
13. Bitansky, N., Canetti, R., Chiesa, A., Tromer, E.: From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In: ITCS 2012, pp. 326–349

14. Bitansky, N., Canetti, R., Paneth, O., Rosen, A.: On the Existence of Extractable One-Way Functions. SIAM J. Comput. **45**(5) (2016) pp. 1910–1952
15. Bitansky, N., Paneth, O.: ZAPs and non-interactive witness indistinguishability from indistinguishability obfuscation. In: TCC 2015, Part II. LNCS, vol. 9015, pp. 401–427
16. Boneh, D., Boyen, X.: Short signatures without random oracles. In: EUROCRYPT 2004. LNCS, vol. 3027, pp. 56–73
17. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. Journal of Cryptology **17**(4) (2004) pp. 297–319
18. Boyle, E., Pass, R.: Limits of extractability assumptions with distributional auxiliary input. In: ASIACRYPT 2015, Part II. LNCS, vol. 9453, pp. 236–261
19. Canetti, R., Dakdouk, R.R.: Extractable perfectly one-way functions. In: ICALP 2008, Part II. LNCS, vol. 5126, pp. 449–460
20. Canetti, R., Goldreich, O., Goldwasser, S., Micali, S.: Resettable zero-knowledge (extended abstract). In: 32nd ACM STOC, pp. 235–244
21. Chase, M., Lysyanskaya, A.: On signatures of knowledge. In: CRYPTO 2006. LNCS, vol. 4117, pp. 78–96
22. Dakdouk, R.R.: Theory and Application of Extractable Functions. PhD thesis, Yale University (2009)
23. Damgård, I.: Towards practical public key systems secure against chosen ciphertext attacks. In: CRYPTO'91. LNCS, vol. 576, pp. 445–456
24. Danezis, G., Fournet, C., Groth, J., Kohlweiss, M.: Square span programs with applications to succinct NIZK arguments. In: ASIACRYPT 2014, Part I. LNCS, vol. 8873, pp. 532–550
25. Feige, U., Lapidot, D., Shamir, A.: Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In: 31st FOCS, pp. 308–317
26. Fischlin, M., Lehmann, A., Pietrzak, K.: Robust multi-property combiners for hash functions revisited. In: ICALP 2008, Part II. LNCS, vol. 5126, pp. 655–666
27. Fuchsbauer, G.: Subversion-zero-knowledge SNARKs. In: PKC 2018, Part I. LNCS, vol. 10769, pp. 315–347
28. Fuchsbauer, G., Orrù, M.: Non-interactive zaps of knowledge. In: ACNS 18. LNCS, vol. 10892, pp. 44–62
29. Fuchsbauer, G., Plouviez, A., Seurin, Y.: Blind schnorr signatures and signed ElGamal encryption in the algebraic group model. In: EUROCRYPT 2020, Part II. LNCS, vol. 12106, pp. 63–95
30. 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
31. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: 40th ACM STOC, pp. 197–206
32. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: 43rd ACM STOC, pp. 99–108
33. Goldreich, O., Oren, Y.: Definitions and properties of zero-knowledge proof systems. Journal of Cryptology **7**(1) (1994) pp. 1–32
34. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems (extended abstract). In: 17th ACM STOC, pp. 291–304
35. Goyal, V., Jain, A., Jin, Z., Malavolta, G.: Statistical zaps and new oblivious transfer protocols. In: EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 668–699
36. Groth, J.: Short pairing-based non-interactive zero-knowledge arguments. In: ASIACRYPT 2010. LNCS, vol. 6477, pp. 321–340

37. Groth, J.: On the size of pairing-based non-interactive arguments. In: EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 305–326
38. Groth, J., Kohlweiss, M., Maller, M., Meiklejohn, S., Miers, I.: Updatable and universal common reference strings with applications to zk-SNARKs. In: CRYPTO 2018, Part III. LNCS, vol. 10993, pp. 698–728
39. Groth, J., Ostrovsky, R., Sahai, A.: Non-interactive zaps and new techniques for NIZK. In: CRYPTO 2006. LNCS, vol. 4117, pp. 97–111
40. Harnik, D., Kilian, J., Naor, M., Reingold, O., Rosen, A.: On robust combiners for oblivious transfer and other primitives. In: EUROCRYPT 2005. LNCS, vol. 3494, pp. 96–113
41. Jain, A., Lin, H., Sahai, A.: Indistinguishability obfuscation from well-founded assumptions. Cryptology ePrint Archive, Report 2020/1003 (2020) `https://eprint.iacr.org/2020/1003`.
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., Raz, R., Rothblum, R.D.: How to delegate computations: the power of no-signaling proofs. In: 46th ACM STOC, pp. 485–494
45. Katsumata, S., Nishimaki, R., Yamada, S., Yamakawa, T.: Exploring constructions of compact NIZKs from various assumptions. In: CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 639–669
46. Lamport, L.: Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory (1979)
47. Lepinski, M.: On the existence of 3-round zero-knowledge proofs. Master's thesis, MIT, USA (2002)
48. Lipmaa, H.: Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In: TCC 2012. LNCS, vol. 7194, pp. 169–189
49. Lombardi, A., Vaikuntanathan, V., Wichs, D.: Statistical ZAPR arguments from bilinear maps. In: EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 620–641
50. Maller, M., Bowe, S., Kohlweiss, M., Meiklejohn, S.: Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings. In: ACM CCS 2019, pp. 2111–2128
51. Micali, S.: Computationally Sound Proofs. SIAM J. Comput. **30**(4) (2000) pp. 1253–1298
52. Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the Association for Computing Machinery **21**(2) (1978) pp. 120–126
53. Russell, A., Tang, Q., Yung, M., Zhou, H.S.: Cliptography: Clipping the power of kleptographic attacks. In: ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 34–64
54. Sadeghi, A.R., Steiner, M.: Assumptions related to discrete logarithms: Why subtleties make a real difference. In: EUROCRYPT 2001. LNCS, vol. 2045, pp. 244–261
55. Schnorr, C.P.: Efficient identification and signatures for smart cards. In: CRYPTO'89. LNCS, vol. 435, pp. 239–252
56. Wee, H., Wichs, D.: Candidate obfuscation via oblivious LWE sampling. Cryptology ePrint Archive, Report 2020/1042 (2020) `https://eprint.iacr.org/2020/1042`.