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.

Keywords: Generalized EOWF, NIZK, NIWI, subversion zero-knowledge, VEGOWF

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. [GMR85] who proposed *proofs of knowledge*, and later formalized for interactive proofs by Bellare and Goldreich [BG93].[4] For non-interactive proofs, Damgård [Dam92] 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 [CD08] 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 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 [Bar01, BGGL01]) extractability, Ext is universal and has no access (resp., has access) to $\mathcal{A}$'s code.

Until the work of Bitansky *et al.* in [BCPR16], EOWFs were only known under very strong knowledge-of-exponent assumptions [BCCT12], 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 [JLS20, WW20]).

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, [Gro10, Lip12, GGPR13, Gro16]) 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

[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.

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 [BCG$^+$14]).

An interesting question is which assumptions are necessary for SNARKs. Due to the impossibility result of Gentry and Wichs [GW11], 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. [BCCT12] 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 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 [RTYZ16]. Bellare et al. [BFS16] 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, [ABLZ17, Fuc18, ALSZ20b] constructed Sub-ZK SNARKs and [ALSZ20a] constructed succinct Sub-ZK quasi-adaptive NIZKs [JR13]. As noted in [ALSZ20a], Sub-ZK in the CRS model is equivalent to zero-knowledge in the minimal bare public key (BPK, [CGGM00]) 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 [GO94], one needs to use non-auxiliary-string non-black-box techniques to achieve Sub-ZK [ALSZ20a]. 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. [BCCT12] 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. [BCPR16], and show that in fact their GEOWF construction can easily be turned into a VEGOWF. Moreover, while Bitansky et al. [BCPR16] 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 [BCPR16] and assume a benign distribution of auxiliary inputs as suggested in [BP15b].

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

[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 [RSA78] for the RSA cryptosystem.

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 [BFJ+20, GJJM20, LVW20]. Similar observations were previously made about specific Sub-ZK SNARKs in [Fuc18].

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.

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 [CD08] and GEOWF [BCPR16], 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 $y \in \mathrm{image}(f)$, 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 $y \in \mathrm{image}(g)$, 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) $\mathcal{G} = \{\mathsf{g_e}\}_{\mathsf{e}}$ 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 $Y_{\mathsf{Ext}} \supseteq \mathrm{image}(\mathsf{g_e})$ 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 $y \in Y_{\mathsf{Ext}} \setminus \mathrm{image}(\mathsf{g_e})$, and in general, it might be hard to decide if $y \in \mathrm{image}(\mathsf{g_e})$. We say that a VEGOWF is keyless if e is the security parameter λ; 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 $\mathbf{RG}_{\mathsf{e}} = \{(\mathsf{g_e}(x), x)\}$.

Generic transformations. We show that any VEGOWF can be transformed to a VEOWF with a simple technique that was first mentioned in [BCPR16], 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, [BCC+17] 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 [HKN$^+$05, Her05, FLP08] 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 [BCPR16, 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 DS [ABOR00] allows one to prove statements of the form "a machine $\mathcal{M}$ outputs y on input x in time t". A delegation proof π_{DS} must be faster to verify than the statement itself. The CRS-verifiability means that one can efficiently check if the DS CRS $\mathsf{crs}_{\mathsf{DS}}$ is a valid CRS.

In the BCPR construction, each function $\mathsf{g_e}$ computes a CRS $\mathsf{crs}_{\mathsf{DS}}$ for a delegation scheme 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 π_{DS} is a DS-proof, using $\mathsf{crs}_{\mathsf{DS}}$ as the CRS, for the statement that $\mathcal{A}$ on input 1^λ outputs v. (pad is a padding.) The proof of $\mathbf{RG}$-hardness is as in [BCPR16], 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 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 [ABLZ17] which states that if an adversary on input 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 $\mathsf{g_p}(x) = ([x]_1, [x]_2)$. See Appendix C for more details.

We also construct a VEGOWF based on the Diffie–Hellman Knowledge of Exponent (DH-KE) assumption introduced in [BFS16]. The key is a description p of a symmetric bilinear group, and $\mathsf{g_p}(x, y) = [x, y, xy]_1$. 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 [Dak09, Lep02], we build VEGOWFs using knowledge-sound NIZKs. Suppose that we have a knowledge-sound NIZK Π 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 $\mathsf{g_e}(r_{\mathcal{S}}, r_\pi)$ such that it samples $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}(r_{\mathcal{S}})$, uses r_π as random coins to generate a proof π for x, and outputs (x, π). The input 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 π satisfies NIZK verification and $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$. Since Π is knowledge-sound, we obtain $\mathbf{RG}$-verifiable-extractability by using Π's verification on (x, π). $\mathbf{RG}$-hardness is satisfied since π 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 Σ-protocol together with the Fiat-Shamir heuristic as a NIZK, we obtain a very efficient VEOWF $\mathsf{g_e}(x, r) := (\mathsf{x} = [x], a = [r], z = H([x], [r]) \cdot x + r)$ 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.

[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 p. A bilinear map $\bullet$ allows us to compute $[x]_1 \bullet [y]_2 = [xy]_T$.

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 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 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 [Lam79] and RSA signature), in some cases it gives a VEGOWF that we already considered before (e.g., Schnorr's signature scheme [Sch90] and Boneh-Boyen signature [BB04]) 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 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 [GPV08], 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 NP-relation $\mathbf{R}$ using a variant of the well-known FLS disjunctive approach [FLS90]. Namely, we use a knowledge-sound Sub-WI NIWI Π_{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 $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}}\}$ is a keyless VEGOWF with respect to $\mathbf{RG}$ and $\widehat{y} \in Y_{\mathsf{Ext}}$ being added to Π_{wi}'s CRS. Knowledge-soundness of the new protocol will follow from the knowledge-soundness of Π_{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 Π_{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 Π for any NP-relation $\mathbf{R}$. It similarly uses the FLS approach with a keyless VEGOWF, but additionally uses a commitment to a trapdoor. Specifically, Π implements a Sub-WI NIWI Π_{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 $c = \mathsf{C}.\mathsf{Com}(\widehat{z}, \widehat{r})$ 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 Π consists of a commitment c and a proof in Π_{wi}, so this construction is less efficient than the previous one. However, this does not rely on Π_{wi} being knowledge-sound, so the construction is still of interest. The soundness of Π follows from the soundness of Π_{wi} together with the $\mathbf{RG}$-hardness of $\mathcal{G}$ and the extractability of C. Note that Π_{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 Π_{wi} and the hiding property of 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 statistical two-message private-coin witness-indistinguishable

[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.

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 [BFJ$^+$20, GJJM20, LVW20]. Note that existing Sub-ZK SNARKs [ABLZ17, Fuc18] 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 [Fuc18].

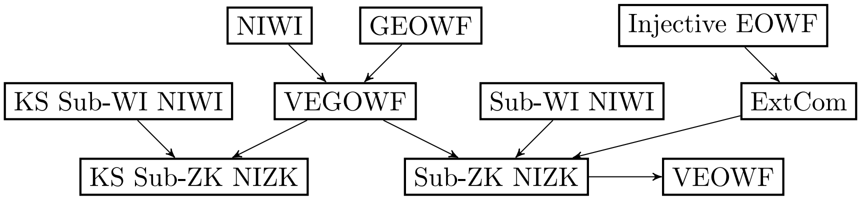

Fig. 1. Relations between argument systems and extractable functions. Multiple arrows pointing to the same node means that each source node is required to construct the destination node. KS denotes knowledge-sound.

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 [GOS06] based on DLIN or [BP15a] based on *iO* and OWF. In comparison, [BFS16] 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 [FO18] 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 [ABLZ17, Fuc18] can be slightly modified to achieve Sub-ZK from any VEGOWF rather than a specific knowledge-of-exponent assumption.

	Soundness	**Knowledge Soundness**	**Sub-ZK**
[BFS16]	DH-KE + CDH	×	DH-KE + DLIN
Sec. 5.2	injective VEOWF	×	injective VEOWF + DLIN
Sec. 5.2	injective VEOWF	×	injective VEOWF + *iO*
[ABLZ17]	GGM	GGM	BDH-KE
[Fuc18, Sec. 4]	q_1-PDH + q_2-PKE	q_1-PDH + q_2-PKE	SKE
[Fuc18, Sec. 5]	q_1-PDH + q_2-PKE + q_3-SDH	q_1-PDH + q_2-PKE + q_3-TSDH	SKE
[Fuc18, 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.

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 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 td and outputs a 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 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 td is which is particularly bad for simulation among all the possible preimages of crs. Verifiable-extractability property follows straightforwardly from the Sub-ZK property of the NIZK since it requires that 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 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 $\mathrm{image}(\mathcal{A})$ 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 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_ι 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, [CD08]) g is an OWF with the property that if $\mathcal{A}$ outputs a value in the image of g, then one can extract its preimage. A generalized EOWF (GEOWF, [BCPR16]) is a function 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.* [BCPR16] 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 [BCPR16]. 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* aux *sampled from a benign distribution of* $\mathsf{poly}(\lambda)$*-bit strings*

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

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,*

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

The function is publicly verifiable *if there exists a polynomial-time tester $\mathcal{T}$ such that for any (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 λ, there is only one key $\mathsf{e} = 1^\lambda$. For ease of notation, we simply write g_λ 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.* [BCPR16] 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 [KRR14, KPY19]. In a universal delegation scheme (cf. Appendix A), one delegates computation of some circuit M on input x to a prover, who must compute $M(x)$ and provide a proof π 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 [KRR14] and even falsifiable assumptions [KPY19] for languages in BPP.

BCPR GEOWF. We briefly describe the construction from [BCPR16] 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 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 DS, one can construct a succinct proof π_DS of length $\mathsf{DS.plen}(\lambda)$ that a Turing machine M on input 1^λ outputs some value v in time $T(\lambda)$, where $T(\lambda) \in (2^{\omega(\log \lambda)}, 2^{\mathsf{poly}(\lambda)})$ is some superpolynomial function. DS must satisfy that the proof verification complexity is linear in M's size and polylogarithmic in 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 ([BCPR16, 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 DS and PRG. In addition, it uses a Barak-type [Bar01] extractability paradigm (namely, the machine M is the adversary who outputs y). It is worth noting that a similar approach with a number of extra steps [BCPR16, 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 NP relation $\mathbf{R}$ and a parameter p (e.g., the group description). An argument system Ψ is a tuple of PPT algorithms $(\mathsf{K}, \mathsf{P}, \mathsf{V})$. The CRS generation algorithm K takes in $(\mathbf{R}, \mathsf{p})$ and outputs a crs and a trapdoor td (which may be $\perp$ if the argument does not have zero-knowledge). The prover algorithm P takes in $\mathbf{R}, \mathsf{p}, \mathsf{crs}$ and $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$ and outputs a proof π. The verifier algorithm 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 Sim that takes in $(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{td}, \mathsf{x})$ and outputs a simulated proof π for the statement x. Furthermore, a subversion resistant argument will have a CRS verification algorithm 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 [Gro16]). *A non-interactive argument Ψ is* perfectly complete for *$\mathcal{R}$, if for all λ, all $(\mathbf{R}, \mathsf{p}) \in \operatorname{range}(\mathcal{R}(1^\lambda))$, and $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$,*

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

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

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

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

Definition 5 (Computational Knowledge Soundness). *Ψ 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*

$$\Pr\left[\begin{array}{l}(\mathbf{R}, \mathsf{p}) \leftarrow \mathcal{R}(1^\lambda), (\mathsf{crs}, \mathsf{td}) \leftarrow \mathsf{K}(\mathbf{R}, \mathsf{p}), r \leftarrow_{\$} \mathsf{RND}_\lambda(\mathcal{A}), \\ (\mathsf{x}, \pi) \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{p}, \mathsf{crs}; r), \mathsf{w} \leftarrow \mathsf{Ext}_{\mathcal{A}}(\mathbf{R}, \mathsf{p}, \mathsf{crs}; r) : \\ (\mathsf{x}, \mathsf{w}) \notin \mathbf{R} \wedge \mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1\end{array}\right] \leq \mathsf{negl}(\lambda) \ .$$

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 [BCPR16].

Definition 6 (Statistically Composable ZK [Gro06]). *Ψ is* statistically composable zero-knowledge *for $\mathcal{R}$, if for all $(\mathbf{R}, \mathsf{p}) \in \operatorname{range}(\mathcal{R}(1^\lambda))$, and all computationally unbounded $\mathcal{A}$, $\varepsilon_0^{comp} \approx_\lambda \varepsilon_1^{comp}$, where $\varepsilon_b^{comp} =$*

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

Ψ *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* td *is not given as input to* $\mathcal{A}$*.*

Definition 7 (Statistically Composable Sub-ZK [ABLZ17]). Ψ *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* $\varepsilon_b^{comp} =$

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

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

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

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

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

Definition 9 (Sub-WI [BFS16]). Ψ *is* computationally Sub-WI for $\mathcal{R}$*, if for any PPT subverter* $\mathcal{Z}$*,* $\varepsilon_0^{wi} \approx_\lambda \varepsilon_1^{wi}$*, where* $\varepsilon_b^{wi} =$

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

Ψ *is* perfectly Sub-WI for $\mathcal{R}$ *if one requires that* $\varepsilon_0^{wi} = \varepsilon_1^{wi}$ *for an unbounded* $\mathcal{Z}$*. In case* Ψ *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 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_{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 image(PRG) which contradicts the security of 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 DS CRS and v is *any* string output by an adversary with bounded auxiliary input.

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

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

```
$\mathsf{ImV}_{\mathsf{f}}(\mathsf{e}; y)$
  return $\mathsf{EV}_{\mathsf{g}}(\mathsf{e}; y) \vee y = \perp$;
```

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

RG-verifiably-extractable with respect to Y_{Ext}: Let image(g_{e}) $\subseteq Y_{\mathsf{Ext}} \subseteq Y_{\lambda}$, and let 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 Ext, s.t. for any benign distribution $\mathcal{D}_{\lambda}$ of $\mathsf{poly}(\lambda)$-bit strings,

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

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

If there exists an algorithm ImV that decides membership in image(g_{e}), then the GEOWF is *image-verifiable.* Clearly, any image-verifiable GEOWF is also verifiably-extractable with respect to $Y_{\mathsf{Ext}} =$ image(g_{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 $y \notin$ image(g_{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 idea is to include to a VEGOWF g_{e} a branch input $i \in \{0,1\}^{\lambda}$. If $i \neq 0^{\lambda}$, which happens with an overwhelming probability, then g_{e} works as usual and outputs $\mathsf{g}_{\mathsf{e}}(x)$. However, on a trapdoor branch $i = 0^{\lambda}$, the function uses its two extra inputs y and z. If y satisfies $\mathsf{EV}_{\mathsf{g}}(\mathsf{e}; y)$ 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 $y \in$ image(g_{e}) and the preimage has to contain either x such that $\mathsf{g}_{\mathsf{e}}(x) = y$ 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 **RG**-hardness. On the other hand, the VEOWF extractor can use the VEGOWF extractor to recover z from y when $\mathsf{EV}_{\mathsf{g}}(\mathsf{e}; y) = 1$ and then return a preimage $(0^{\lambda}, \perp, y, z)$.

A similar transformation was introduced in [BCPR16] 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 $y \notin$ image(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 $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}}\}_{\mathsf{e}}$ is* **RG**-*hard and* **RG**-*verifiably-extractable, then $\mathcal{F} = \{\mathsf{f}_{\mathsf{e}}\}_{\mathsf{e}}$ 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 π 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 [BOV03, GOS06, BP15a].

The intuition is that we create a new function $g(x, y, r) = (f(x), f(y), \pi)$ where π 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.

```
$\mathsf{g}_{\mathsf{e}}(x_1, x_2, r)$
    $y_1 \leftarrow \mathsf{f}_{\mathsf{e}}(x_1); y_2 \leftarrow \mathsf{f}_{\mathsf{e}}(x_2);$
    $\pi \leftarrow \mathsf{P}\left(\mathbf{R}_{\mathsf{e}}, (\mathsf{f}_{\mathsf{e}}(x_1), \mathsf{f}_{\mathsf{e}}(x_2)), (x_1, x_2); r\right);$
    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 $\mathcal{F} = \{\mathsf{f}_{\mathsf{e}}\}_{\mathsf{e}}$ to a VEGOWF $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}}\}_{\mathsf{e}}$.

Consider a GEOWF $\mathcal{F} = \{\mathsf{f}_{\mathsf{e}}\}_{\mathsf{e}}$ with an associated relation $\mathbf{RG}$. Let $\mathit{\Pi} = (\mathsf{P}, \mathsf{V})$ be a perfectly sound NIWI, and let the relation $\mathbf{R}_{\mathsf{e}}((y_1, y_2), (x_1', x_2'))$ hold iff $y_1 = \mathsf{f}_{\mathsf{e}}(x_1')$ or $y_2 = \mathsf{f}_{\mathsf{e}}(x_2')$. We define a VEGOWF $\mathcal{G} = \{\mathsf{g}_{\mathsf{e}}\}_{\mathsf{e}}$ with an extraction verification algorithm EV in Fig. 4 and define the hardness relation:

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

Similar techniques have been used before in conjunction with EOWFs (e.g, 3-round ZK in [BCC+17]) but not, up to our knowledge, as a generic transformation.

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 $\mathbf{RG}'$.*

Proof. **Verifiable-extractability:** Suppose an adversary $\mathcal{A}$ outputs $y = (y_1, y_2, \pi)$ such that $\mathsf{EV}(\mathsf{e}; (y_1, y_2, \pi)) = 1$. This means that the verifier V accepts the proof, and by perfect soundness this shows that $y_1 \in \mathrm{image}(\mathsf{f}_{\mathsf{e}})$ or $y_2 \in \mathrm{image}(\mathsf{f}_{\mathsf{e}})$. Based on $\mathcal{A}$, we can create adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$, which output y_1 and y_2, respectively. WLOG, assume that $y_1 \in \mathrm{image}(\mathsf{f}_{\mathsf{e}})$. By the definition of GEOWFs, there exists an extractor $\mathsf{Ext}_{\mathcal{A}_1}$ which is able to extract z_1 such that $\mathbf{RG}_{\mathsf{e}}(y_1, z_1)$ holds, and therefore we are able to extract $z = (z_1, z_2)$ such that $\mathbf{RG}'_{\mathsf{e}}(y, z)$ holds, which establishes verifiable-extractability.

$\mathbf{RG}'$-hardness: Suppose there exists an adversary $\mathcal{A}$ against $\mathbf{RG}'$-hardness. Therefore, given $(\mathsf{f}_{\mathsf{e}}(x_1), \mathsf{f}_{\mathsf{e}}(x_2), \pi = \mathsf{P}(\mathbf{R}_{\mathsf{e}}, (\mathsf{f}_{\mathsf{e}}(x_1), \mathsf{f}_{\mathsf{e}}(x_2)), (x_1, x_2); r)$ with x, y, r chosen uniformly at random, it outputs (z_1, z_2) such that, with noticeable probability $\varepsilon(\lambda)$, $\mathbf{RG}'_{\mathsf{e}}((\mathsf{f}_{\mathsf{e}}(x_1), \mathsf{f}_{\mathsf{e}}(x_2), \pi), (z_1, z_2))$ holds.

We will now use $\mathcal{A}$ to break $\mathbf{RG}$-hardness of f_{e}. Suppose we are given as input $\mathsf{f}_{\mathsf{e}}(x)$ for a randomly chosen x, and we wish to find z such that $\mathbf{RG}_{\mathsf{e}}(\mathsf{f}_{\mathsf{e}}(x), z) = 1$. To do this, we choose x' and r at random, and run $\mathcal{A}$ on one of the two inputs, chosen at random with probability $1/2$:

$$\begin{gathered}(\mathsf{f}_{\mathsf{e}}(x), \mathsf{f}_{\mathsf{e}}(x'), \mathsf{P}(\mathbf{R}_{\mathsf{e}}, (\mathsf{f}_{\mathsf{e}}(x), \mathsf{f}_{\mathsf{e}}(x')), (\perp, x'); r), \\ (\mathsf{f}_{\mathsf{e}}(x'), \mathsf{f}_{\mathsf{e}}(x), \mathsf{P}(\mathbf{R}_{\mathsf{e}}, (\mathsf{f}_{\mathsf{e}}(x'), \mathsf{f}_{\mathsf{e}}(x)), (x', \perp); r).\end{gathered}$$

Here $\perp$ is some arbitrary element of the input space. By the witness-indistinguishability of $\mathit{\Pi}$, these two inputs are indistinguishable from honestly generated values $\mathsf{g}_{\mathsf{e}}(x, x', r)$ and $\mathsf{g}_{\mathsf{e}}(x', x, r)$, respectively, and these two values are equally distributed. Therefore, $\mathcal{A}$ will output z such that $\mathbf{RG}(f(x), z)$ with probability $\varepsilon(\lambda)/2 - \mathsf{negl}(\lambda)$, which is a noticeable probability, and therefore contradicts the assumption that $\mathcal{F}$ is $\mathbf{RG}$-hard. □

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

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

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 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 [BCPR16]; we have reproduced it for the sake of completeness.

Theorem 3. *Let* 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* aux.

Proof. **RG-hardness.** Identical to the proof of Theorem 14 in [BCPR16].

RG-verifiable-extractability. Since DS is CRS-verifiable, there exists an algorithm 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 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 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^λ, 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^λ, after at most $t_{\mathcal{A}} < T(\lambda)$ steps, it outputs $\mathcal{A}_{\mathsf{aux}}(1^\lambda) := y = (\mathsf{crs}_{\mathsf{DS}}, v) \in \{0,1\}^{\mathfrak{l}'(\lambda)}$. (Recall $Y_{\mathsf{Ext}} \subseteq \{0,1\}^{\mathfrak{l}'(\lambda)}$.) 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 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 [Mic00, ABLZ17, Fuc18] 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 P. Thus, even a tautological security assumption (the corresponding SNARK is sound for BPP) would be falsifiable. Such an assumption about [Mic00] was made say in [CLP13]. Secondly, some recent suggestions for delegation schemes [KPY19, KNYY19] 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 [Dak09, Section 3.3.1.1]. In each case, the key is a description 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 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.* [ABLZ17]. 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) [BCN$^+$10] assumption, and extractability is equivalent to the BDH-KE assumption introduced in [ABLZ17]. 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 Appendix C. 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 [BFS16] 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 [Dam92].
- $\mathsf{g}_\mathsf{p}(x) = ([1, x, \ldots, x^q]_1, [1, x, \ldots, x^q]_2)$ is a OWF under the q-PDL assumption [Lip12] and verifiably-extractable under the q-PKE assumption [DFGK14].
- $\mathsf{g}_\mathsf{p}(x) = ([x, x^2]_1, [x]_2)$ is a OWF under a well-known variant of the discrete logarithm assumption and verifiably-extractable under the square knowledge of exponent assumption [Fuc18].
- $\mathsf{g}_\mathsf{p}(x) = ([x]_1, [1/x]_2)$ is a OWF under the inverse-exponent assumption [SS01] 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 [Dak09, Section 3.3.3.2] observed that EOWFs can be constructed from the proof of knowledge (PoK) assumption of Lepinski [Lep02] which states that a specific non-interactive Σ-protocol described in [Lep02] is secure. We generalize this idea, and show how to use knowledge-sound NIZKs to build VEGOWFs.

Suppose that $\mathbf{R}$ is an NP relation with a sampler $\mathcal{S}_{\mathbf{R},\mathsf{p}}$ that outputs (x, w), such that (i) it is efficient to verify that (x, w) is a possible output of $\mathcal{S}_{\mathbf{R},\mathsf{p}}$, and (ii) with an overwhelming probability it is computationally hard to guess w given 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. KGen is the "key" generation algorithm, such that $\mathsf{KGen}(\mathbf{R}, \mathsf{p})$ produces an auxiliary input aux_Π, provided to P, V and Sim. If the NIZK uses a random oracle, then aux_Π may contain the description of a hash function instantiating the random oracle. If the NIZK is CRS-based, then aux_Π 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* $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{S}_\mathbf{R}(r_\mathcal{S})$*,* π *produced by* Π*'s prover* P *for* x, w*, and then outputs* (x, π)*. Define the corresponding relation as* $\mathbf{RG}_{\mathbf{R},\mathsf{p},\mathsf{aux}_\Pi} :=$

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

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

Proof. **RG-hardness:** Let $\mathcal{B}$ be an adversary that given $\widehat{y} = (\mathsf{x}, \pi)$, where π is a proof for (x, 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 Sim and gets a simulated proof π_{Sim}. Since Π 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., Π's verifier accepts. We use the knowledge-soundness extractor Ext from Π 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 Π's verifier accepts. $\mathsf{Ext}_{\mathcal{G}}$ runs 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 Π-extractor Ext outputs 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 Ext. □

For the sake of concreteness we instantiate the above result as follows. Let Σ 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 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 Σ is knowledge-sound in the random oracle model and the algebraic group model [FPS20]. If Σ is zero-knowledge in the random oracle model and the discrete logarithm problem is hard, g_{e} is also $\mathbf{RG}_{\mathsf{e}}$-hard. Moreover, Σ 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 (vk, σ) 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) 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 [CL06], 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.

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 σ 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 [BB04] 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 [BLS04] 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. [GPV08], 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 $\mathsf{sk} = T$.

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 σ 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 $L_p^{\perp}(A)$.

Negative example: Lamport's one-time signature. We briefly remind the idea of Lamport's signature scheme [Lam79]. Let $f : X \to Y$ be a one-way function and suppose we want to sign n-bit messages. The secret key sk is a $2 \times n$ matrix where $\mathsf{sk}_{b,i} \leftarrow\!\$\, X$ and the public key vk is also a $2 \times n$ matrix where $\mathsf{vk}_{b,i} = f(\mathsf{sk}_{b,i})$. In order to sign a message $m \in \{0,1\}^n$, signer reveals $\mathsf{sk}_{m_i+1,i}$ for $i = 1, \ldots\ldots, n$. That is, the signer reveals half of the secret key that corresponds to the bit-representation of m.

However, this one does not seem to be a good candidate for a VEGOWF. An adversary could easily compute vk that contains a valid preimage for secret keys that correspond to $\mathsf{Sign}(0)$ and rest of the vk is computed obliviously, for example, by hashing into a group in case $f(x) = g^x$.

Schnorr's signature. We recall the signature scheme from [Sch90]. Let $H_K : \{0,1\}^* \to \mathbb{Z}_p$ be a collision-resistant hash function with a key K and let $\mathsf{p} = (\mathbb{G}, p, g)$ be a description of a prime p order group with a generator g. We assume that the discrete logarithm assumption holds in $\mathbb{G}$. The verification key $\mathsf{vk} = g^x$ where $\mathsf{sk} = x \leftarrow\!\$\, \mathbb{Z}_p$ and $\mathsf{Sign}(\mathsf{sk}, M \in \{0,1\}^*) = (g^y, z = ex + y) = \sigma$ where $e = H_K(g^y, M)$ and $y \leftarrow\!\$\, \mathbb{Z}_p$. Verification is done by checking that $g^{\sigma} = (g^x)^e \cdot g^y$. The candidate VEOWF we obtain is $\mathsf{g}_{\mathsf{p},K}(x, y) = (g^x, g^y, H_K(g^y, 0)x + y)$.

Due to a well-known weakness of the Fiat-Shamir paradigm, $\mathsf{g}_{\mathsf{p},K}(x, y)$ is insecure. The adversary can sample $g^y \leftarrow\!\$\, \mathbb{G}$ without necessarily knowing y, sample $z \leftarrow \mathbb{Z}_p$, and then compute $g^x = (g^z/g^y)^{1/(H_K(g^y,0))}$. Hence the verification holds by definition but the adversary may not know x, y. We can easily fix this by instead computing $H_K(g^x, g^y, 0)$, which will give the VEOWF mentioned in Section 4.5.

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 $\mathit{\Pi}$ by using a knowledge-sound Sub-WI

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

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

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

$\mathsf{V}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \pi)$
    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 $\mathbf{RG}$.

NIWI Π_{wi} and $\mathcal{G}$. To prove that $\mathsf{x} \in \mathcal{L}$, we use Π_{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 Π, 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 Π 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 Π_{wi}, and subversion zero-knowledge is achieved by the $\mathbf{RG}$-verifiable-extractability of $\mathcal{G}$ as well as the subversion witness-indistinguishability of Π_{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 Π is the same as the efficiency of Π' for the modified circuit of size $k + l$. Note also that l is independent of $\mathbf{R}$.

Theorem 5 (Knowledge-sound Sub-WI NIWI + VEGOWF $\Longrightarrow$ Knowledge-sound Sub-ZK NIZK). *Let Π_{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 Π_{wi} is complete then Π is complete.

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

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

(4) If Π_{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 Π is a Sub-ZK SNARK.

Proof. **Completeness:** Straightforward.

Knowledge Soundness: Since Π_{wi} is knowledge-sound, for every Π_{wi}-adversary $\mathcal{B}$ there exists a knowledge-soundness extractor $\mathsf{Ext}^{wi}_{\mathcal{B}}$ such that if $\mathcal{B}(\mathbf{R}', \mathsf{crs}'; r)$ returns an acceptable instance–proof pair $(\mathsf{x}_{\mathbf{R}'} = (\mathsf{x}, \widehat{y}), \pi)$ then $\mathsf{Ext}^{wi}_{\mathcal{B}}(\mathbf{R}', \mathsf{crs}'; r)$ returns $\mathsf{w}_{\mathbf{R}'} = (\mathsf{w}, \widehat{z})$ which satisfies $(\mathsf{x}_{\mathbf{R}'}, \mathsf{w}_{\mathbf{R}'}) \in \mathbf{R}'$ with all but negligible probability ε_{ksnd}.

Since $\mathcal{G}$ is $\mathbf{RG}$-hard, for every $\mathcal{G}$-adversary $\mathcal{C}$, the probability that for random $\widehat{x}$, $\mathcal{C}(1^\lambda, \widehat{y} = \mathsf{g}_\lambda(\widehat{x}), \mathsf{aux}) = \widehat{z}$ such that $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$ is bounded by some negligible ε_{hard}.

Let $\mathcal{A}$ be an adversary for the knowledge-soundness of Π that succeeds with probability $\varepsilon_{\mathcal{A}}$. That is, for $r \leftarrow_\$ \mathsf{RND}_\lambda(\mathcal{A})$, $\mathcal{A}(\mathbf{R}, \mathsf{crs}; r)$ returns with probability $\varepsilon_{\mathcal{A}}$ an instance x and proof π such that the Π-verifier V accepts but no extractor equipped with the code and randomness of $\mathcal{A}$ returns w such that $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$.

We construct the two adversaries $\mathcal{B}$ (against the knowledge-soundness of Π_{wi}) and $\mathcal{C}$ (against the $\mathbf{RG}$-hardness of $\mathcal{G}$), see Fig. 6. We denote by bad the event that $\mathsf{Ext}^{wi}_{\mathcal{B}}$ fails, i.e., for $\mathsf{x}_{\mathbf{R}'}$ output by $\mathcal{B}$ and $\mathsf{w}_{\mathbf{R}'}$ output by $\mathsf{Ext}^{wi}_{\mathcal{B}}$, $(\mathsf{x}_{\mathbf{R}'}, \mathsf{w}_{\mathbf{R}'}) \notin \mathbf{R}'$. Clearly, $\mathcal{B}$ succeeds if (i) $\mathcal{A}$ succeeds (from which it follows that $\mathcal{A}$ returns (x, π)

```
$\mathcal{B}(\mathbf{R}', \mathsf{crs}'; r)$
    $\widehat{x} \leftarrow\!\$\, X_\lambda; \widehat{y} \leftarrow \mathsf{g}_\lambda(\widehat{x});$
    $\mathsf{crs} \leftarrow (\mathsf{crs}', \widehat{y});$
    $(\mathsf{x}, \pi) \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{crs}; r);$
    $\mathbf{return}\ ((\mathsf{x}, \widehat{y}), \pi);$

$\mathcal{C}(\mathbf{R}', \widehat{y}, \mathsf{aux}; r)$
    $\mathsf{crs}' \leftarrow \mathsf{K}'(\mathbf{R}');$
    $(\mathsf{w}, \widehat{z}) \leftarrow \mathsf{Ext}^{wi}_{\mathcal{B}}(\mathbf{R}', \mathsf{crs}'; r);$
    $\mathbf{return}\ \widehat{z};$
```

Fig. 6. $\mathcal{B}$, $\mathcal{C}$ in the knowledge-soundness proof of Theorem 5.

such that V accepts (x, π) and thus V' accepts $(\mathsf{x}_{\mathbf{R}'}, \pi)$) and (ii) bad holds, since then the Π_{wi}-verifier V accepts a statement for which an extractor cannot extract a witness. Thus, $\Pr[\mathcal{B} \text{ succeeds}] \geq \Pr[\mathcal{A} \text{ succeeds} \wedge \mathsf{bad}]$.

On the other hand, $\mathcal{C}$ succeeds if (i) $\mathcal{A}$ succeeds (i.e., $\mathcal{A}$ returns (x, π) such that V accepts (x, π)) and for w returned by any $\mathsf{Ext}_{\mathcal{A}}$, $(\mathsf{x}, \mathsf{w}) \notin \mathbf{R}$, and (ii) bad does not hold (i.e., for $\mathsf{w}_{\mathbf{R}'}$ extracted by $\mathsf{Ext}^{wi}_{\mathcal{B}}$, $(\mathsf{x}_{\mathbf{R}'}, \mathsf{w}_{\mathbf{R}'}) \in \mathbf{R}'$, which means that, since $(\mathsf{x}, \mathsf{w}) \notin \mathbf{R}$, $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$), and therefore $\mathcal{C}$ has output $\widehat{z}$ such that $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$. Thus, $\Pr[\mathcal{C} \text{ succeeds}] \geq \Pr\big[\mathcal{A} \text{ succeeds} \wedge \overline{\mathsf{bad}}\big]$. Clearly,

$$\begin{aligned} \varepsilon_{\mathcal{A}} = \Pr[\mathcal{A} \text{ succeeds}] &= \Pr[\mathcal{A} \text{ succeeds} \wedge \mathsf{bad}] + \Pr\big[\mathcal{A} \text{ succeeds} \wedge \overline{\mathsf{bad}}\big] \\ &\leq \Pr[\mathcal{B} \text{ succeeds}] + \Pr[\mathcal{C} \text{ succeeds}] \\ &\leq \varepsilon_{ksnd} + \varepsilon_{hard} \enspace . \end{aligned}$$

Hence any adversary against the knowledge-soundness of Π has a negligible probability of succeeding, so Π is knowledge-sound.

Sub-ZK: Let $\mathcal{Z}$ be any subverter that, on input $(\mathbf{R}'; r_{\mathcal{Z}})$, creates $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$ such that $\mathsf{CV}(\mathbf{R}, \mathsf{crs}) = 1$, along with some auxiliary information $\mathsf{aux}_{\mathcal{Z}}$. We construct the following adversary $\mathcal{B}$:

```
$\mathcal{B}(\mathbf{R}'; r_{\mathcal{Z}})$
    $(\mathsf{crs}', \widehat{y}), \mathsf{aux}_{\mathcal{Z}} \leftarrow \mathcal{Z}(\mathbf{R}'; r_{\mathcal{Z}}); \mathbf{return}\ \widehat{y};$
```

Since $\mathsf{CV}(\mathbf{R}, \mathsf{crs}) = 1$, we have that $\widehat{y} \in Y_{\mathsf{Ext}}$. Since $\mathcal{G}$ is $\mathbf{RG}$-verifiably-extractable, there exists an extractor $\mathsf{Ext}_{\mathcal{B}}$ such that, with overwhelming probability $1 - \varepsilon_{ext}$, $\widehat{z} \leftarrow \mathsf{Ext}_{\mathcal{B}}(\mathbf{R}; r_{\mathcal{Z}})$ where $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$.

The extractor $\mathsf{Ext}_{\mathcal{Z}}$ for $\mathcal{Z}$ works as follows: Given $\mathcal{Z}$, construct $\mathcal{B}$ as above, and output the result of $\mathsf{Ext}_{\mathcal{B}}$. Clearly, $\mathsf{Ext}_{\mathcal{Z}}$ succeeds precisely when $\mathsf{Ext}_{\mathcal{B}}$ does. Let bad denote the event where extraction fails, i.e. the $\widehat{z}$ returned by $\mathsf{Ext}_{\mathcal{B}}(\mathbf{R}; r_{\mathcal{Z}})$ does not satisfy $(\widehat{y}, \widehat{z}) \in \mathbf{RG}$. We then have that $\Pr[\mathsf{bad}] = \varepsilon_{ext}$.

Consider an adversary $\mathcal{A}_{sub\text{-}zk}$ against the Sub-ZK property of Π. Based on $\mathcal{Z}$ and $\mathcal{A}_{sub\text{-}zk}$, we construct an adversary $\mathcal{A}_{sub\text{-}wi}$ which succeeds precisely when $\mathcal{A}_{sub\text{-}zk}$ succeeds and extraction is successful.

```
$\mathcal{A}_{sub\text{-}wi}(\mathbf{R}', r_{\mathcal{Z}})$
    $((\mathsf{crs}', \widehat{y}), \mathsf{aux}_{\mathcal{Z}}; \widehat{z}) \leftarrow (\mathcal{Z}; \mathsf{Ext}_{\mathcal{Z}})(\mathbf{R}', r_{\mathcal{Z}});$
    $(\mathsf{x}, \mathsf{w}) \leftarrow \mathcal{A}_{sub\text{-}zk}(\mathbf{R}', (\mathsf{crs}', \widehat{y}), \mathsf{aux}_{\mathcal{Z}})$
    $\pi_b \leftarrow \mathsf{O}_{wi}(\mathsf{crs}', (\mathsf{x}, \widehat{y}), (\mathsf{w}, \perp), (\perp, \widehat{z}), \mathsf{aux}_{\mathcal{Z}});$
    $\mathbf{return}\ \mathcal{A}_{sub\text{-}zk}(\pi_b, \mathsf{aux}_{\mathcal{Z}});$
```

Here O_{wi} is the challenger of the Sub-WI game that takes a NIWI instance x along with two witnesses w_0, w_1 and returns a proof π_b which uses w_b for a randomly sampled $b \leftarrow\!\$\, \{0, 1\}$.

$\mathcal{A}_{sub\text{-}wi}$ succeeds in the Sub-WI game when both i) $(\perp, \widehat{z})$ is a valid witness (this happens precisely when $\mathsf{Ext}_{\mathcal{Z}}$ succeeds, i.e. when $\overline{\mathsf{bad}}$ holds) and ii) it distinguishes $(\mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, \widehat{y}), (\mathsf{w}, \perp)), \mathsf{aux}_{\mathcal{Z}})$ and $(\mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, \widehat{y}), (\perp, \widehat{z})), \mathsf{aux}_{\mathcal{Z}})$. By the definition of P and Sim, these distributions equal $(\mathsf{P}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \mathsf{w}), \mathsf{aux}_{\mathcal{Z}})$ and $(\mathsf{Sim}(\mathbf{R}, \mathsf{crs}, \widehat{z}, \mathsf{x}), \mathsf{aux}_{\mathcal{Z}})$, respectively. Thus $\mathcal{A}_{sub\text{-}wi}$ succeeds when both $\overline{\mathsf{bad}}$ holds and $\mathcal{A}_{sub\text{-}zk}$ succeeds, so $\Pr[\mathcal{A}_{sub\text{-}wi} \text{ succeeds}] \geq \Pr[\mathcal{A}_{sub\text{-}zk} \text{ succeeds} \wedge \overline{\mathsf{bad}}]$.

Let $\varepsilon_{sub\text{-}wi}$ be the maximal advantage any adversary has in breaking Sub-WI of Π_{wi}. Then

$$\Pr[\mathcal{A}_{sub\text{-}zk} \text{ succeeds}] = \Pr[\mathcal{A}_{sub\text{-}zk} \text{ succeeds} \wedge \mathsf{bad}] + \Pr[\mathcal{A}_{sub\text{-}zk} \text{ succeeds} \wedge \overline{\mathsf{bad}}]$$

$$\leq \Pr[\mathsf{bad}] + \Pr[\mathcal{A}_{sub\text{-}wi} \text{ succeeds}]$$
$$\leq \varepsilon_{ext} + \varepsilon_{sub\text{-}wi}$$

Note that while computational (respectively, statistical) Sub-WI implies computational (respectively, statistical) Sub-ZK, perfect Sub-WI implies perfect Sub-ZK only if there is no chance of extraction failure for the VEGOWF, otherwise we get statistical Sub-ZK.

Sub-ZK SNARK: Suppose $\mathit{\Pi}_{wi}$ is succinct for $\mathbf{R}'$, then there is some polynomial poly and some $c < 1$ such that the size of any proof π_{wi} from $\mathit{\Pi}_{wi}$ is bounded by $\mathsf{poly}(\lambda) \cdot (|(\mathsf{x}, \widehat{y})| + |(\mathsf{w}, \widehat{z})|)^c = \mathsf{poly}(\lambda) \cdot (|\mathsf{x}| + |\mathsf{w}| + |\widehat{y}| + |\widehat{z}|)^c \leq \mathsf{poly}'' \cdot (\lambda)(|\mathsf{x}| + |\mathsf{w}|)^c$, for some polynomial poly'' since $|\widehat{y}|, |\widehat{z}|$ are bounded by a polynomial. Since any proof in $\mathit{\Pi}'$ for $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$ is a proof in $\mathit{\Pi}_{wi}$ for $((\mathsf{x}, \widehat{y}), (\mathsf{w}, \widehat{z})) \in \mathbf{R}'$, this shows that any proof in $\mathit{\Pi}'$ is succinct. The remaining properties of being a Sub-ZK SNARK have been proved above. □

5.2 Constructing Sub-ZK NIZK

Next, we propose a Sub-ZK NIZK $\mathit{\Pi}$ which only relies on $\mathit{\Pi}_{wi}$ being sound, not knowledge-sound, but $\mathit{\Pi}$ 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 Appendix B 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*

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

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

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

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,*

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

In some cases, we may have an efficient commitment verification function ComV_λ *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 $\mathit{\Pi}_{wi}$ be a NIWI argument for the relation $\mathbf{R}' = \{((\mathsf{x}, c, \widehat{y}), (\mathsf{w}, \widehat{z}, \widehat{r})) : \quad ((\mathsf{x}, \mathsf{w}) \in \mathbf{R}) \vee (c = \mathsf{Com}(\widehat{z}; \widehat{r}) \wedge (\widehat{y}, \widehat{z}) \in \mathbf{RG})\}$.

$$\mathbf{R}' = \left\{((\mathsf{x}, c, \widehat{y}), (\mathsf{w}, \widehat{z}, \widehat{r})) : \quad ((\mathsf{x}, \mathsf{w}) \in \mathbf{R}) \vee (c = \mathsf{Com}(\widehat{z}; \widehat{r}) \wedge (\widehat{y}, \widehat{z}) \in \mathbf{RG})\right\} .$$

We set $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$, where crs' is the CRS of the underlying NIWI $\mathit{\Pi}_{wi}$ for $\mathbf{R}'$ and crs is the CRS of the NIZK for $\mathbf{R}$. The argument consists of the commitment c and the $\mathit{\Pi}_{wi}$-argument π; see Fig. 7.

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

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

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

```
$\mathsf{Sim}(\mathbf{R}, \mathsf{crs}, \mathsf{td} = \widehat{z}, \mathsf{x})$
    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 $\pi \leftarrow (c, \pi')$
```

```
$\mathsf{P}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \mathsf{w})$
    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 $\pi \leftarrow (c, \pi')$;
```

```
$\mathsf{V}(\mathbf{R}, \mathsf{crs}, \mathsf{x}, \pi)$
    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. 7. 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, C is an extractable commitment scheme, and $\mathcal{G} = \{\mathsf{g}_\lambda\}_{\lambda \in \mathbb{N}}$ is a GEOWF.

```
$\mathcal{A}_{snd}(\mathbf{R}', \mathsf{crs}')$
    $\widehat{x} \leftarrow\!\!\$\; X_\lambda; \widehat{y} \leftarrow \mathsf{g}_\lambda(\widehat{x})$;
    $\mathsf{crs} \leftarrow (\mathsf{crs}', \widehat{y})$;
    $(\mathsf{x}, (c, \pi')) \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{crs})$;
    return $((\mathsf{x}, c, \widehat{y}), \pi')$;
```

```
$\mathcal{C}(\mathbf{R}, \mathsf{crs})$
    $(\mathsf{x}, (c, \pi)) \leftarrow \mathcal{A}(\mathbf{R}, \mathsf{crs})$;
    return $c$;
```

```
$\mathcal{A}_{hard}(\mathbf{R}', \widehat{y})$
    $\mathsf{crs}' \leftarrow \mathsf{K}(\mathbf{R}')$;
    $\mathsf{crs} \leftarrow (\mathsf{crs}', \widehat{y})$;
    $\widehat{z} \leftarrow \mathsf{Ext}_{\mathcal{C}}(\mathbf{R}, \mathsf{crs})$;
    return $\widehat{z}$;
```

Fig. 8. $\mathcal{A}_{snd}$, $\mathcal{C}$, $\mathcal{A}_{hard}$ in the soundness proof of Theorem 6.

(1) If Π_{wi} is perfectly complete then Π is perfectly complete.
(2) If Π_{wi} is sound, C is keyless and extractable, and $\mathcal{G}$ is $\mathbf{RG}$-hard then Π is sound.
(3) If Π_{wi} is Sub-WI, $\mathcal{G}$ is $\mathbf{RG}$-verifiably-extractable, and C is keyless and hiding, then Π is Sub-ZK.

Proof. **Perfect completeness:** obvious.

Soundness: Consider an adversary $\mathcal{A}$ against the soundness of Π, which gets as input a relation $\mathbf{R}$ as well as an honestly computed CRS crs, and outputs (x, π) such that $x \notin \mathcal{L}$ but π is accepted by the verifier V. This means that $\mathsf{V}(\mathbf{R}, \mathsf{crs}, x, \pi) = \mathsf{V}'(\mathbf{R}', \mathsf{crs}', (x, c, \widehat{y}), \pi')$ returns 1. We use $\mathcal{A}$ to construct two adversaries, one against the soundness of Π_{wi} and one against the $\mathbf{RG}$-hardness of $\mathcal{G}$, see Fig. 8. Let in denote the event that $(x, c, \widehat{y}) \in \mathcal{L}'$. Let ext denote the event that $\mathsf{Ext}_{\mathcal{C}}$ succeeds in extraction.

Note that, if $\mathcal{A}$ succeeds in breaking the soundness of Π by returning $(\mathsf{x}, (c, \pi'))$ such that $(\mathsf{x}, c, \widehat{y}) \notin \mathcal{L}'$, then $\mathcal{A}_{snd}$ succeeds in breaking the soundness of Π_{wi}. Therefore $\Pr[\mathcal{A} \text{ succeeds} \wedge \overline{\mathsf{in}}] \leq \Pr[\mathcal{A}_{snd} \text{ succeeds}]$.

We also see that, if $\mathcal{A}$ succeeds in breaking the soundness of Π by returning $(\mathsf{x}, (c, \pi'))$ such that $(\mathsf{x}, c, \widehat{y}) \in \mathcal{L}'$ and the extractor $\mathsf{Ext}_{\mathcal{C}}$ is successful, then $\mathcal{A}_{hard}$ succeeds in breaking the $\mathbf{RG}$-hardness of $\mathcal{G}$. Hence $\Pr[\mathcal{A} \text{ succeeds} \wedge \mathsf{in} \wedge \mathsf{ext}] \leq \Pr[\mathcal{A}_{hard} \text{ succeeds}]$.

We have that

$$\begin{aligned}\Pr[\mathcal{A} \text{ succeeds}] &\leq \Pr[\mathcal{A} \text{ succeeds} \wedge \overline{\mathsf{in}}] + \Pr[\mathcal{A} \text{ succeeds} \wedge \mathsf{in} \wedge \mathsf{ext}] + \Pr[\overline{\mathsf{ext}}] \\ &\leq \Pr[\mathcal{A}_{snd} \text{ succeeds}] + \Pr[\mathcal{A}_{hard} \text{ succeeds}] + \Pr[\overline{\mathsf{ext}}]\end{aligned}$$

By the soundness of Π, the $\mathbf{RG}$-hardness of $\mathcal{G}$ and the extractability of C, respectively, we have thatPr$[\mathcal{A}_{snd} \text{ succeeds}]$, $\Pr[\mathcal{A}_{hard} \text{ succeeds}]$ and $\Pr[\overline{\mathsf{ext}}]$ are all negligible. Hence $\Pr[\mathcal{A} \text{ succeeds}]$ is negligible so Π is sound.

Sub-ZK: Let $\mathcal{Z}(\mathbf{R}; r_{\mathcal{Z}})$ be a subverter that outputs $\mathsf{crs} = (\mathsf{crs}', \widehat{y})$ accepted by CV, as well as some auxiliary information $\mathsf{aux}_{\mathcal{Z}}$. We first construct a PPT algorithm $\mathcal{A}(\mathbf{R}; r_{\mathcal{Z}})$ which invokes $\mathcal{Z}(\mathbf{R}; r_{\mathcal{Z}})$, obtains

crs, and then outputs $\widehat{y}$. Since $\mathsf{CV}(\mathsf{crs}) = 1$, $\mathcal{A}$ outputs a value in Y_{Ext}. Since $\mathcal{G}$ is $\mathbf{RG}$-extractable, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}(\mathbf{R}; r_{\mathcal{Z}})$ that extracts $\widehat{z}$ such that $\mathbf{RG}(\widehat{y}, \widehat{z}) = 1$ with overwhelming probability $1 - \varepsilon_{ext}$. This $\widehat{z}$ is given to the simulator Sim, which acts as described in Fig. 7.

Let $\mathcal{A}$ be a PPT adversary which, on input $\mathbf{R}, \mathsf{crs}, \widehat{z}$ and $\mathsf{aux}_{\mathcal{Z}}$, outputs $(\mathsf{x}, \mathsf{w}) \in \mathbf{R}$. We prove that $\mathcal{A}$ cannot distinguish between an honest proof generated with w and a simulated proof using $\widehat{z}$ by providing a sequence of indistinguishable games. Let r_{Sim} and r_{P} be the randomness used by Sim and P, respectively, and let r be some independent randomness. Let x_λ be a random element of X_λ. Let ε_{sw} be the maximal probability of breaking Sub-WI of $\Pi_{wi} = (\mathsf{K}', \mathsf{CV}', \mathsf{P}', \mathsf{V}')$, and let ε_{hide} be the maximal probability of breaking the hiding property of C (see Definition 10). By the Sub-WI property of Π_{wi} and the hiding property of C, ε_{sw} and ε_{hide} are negligible.

$\boxed{\text{Game 1:}}$ This is the output of the simulator Sim in the Sub-ZK game: $(c_{\mathsf{Sim}}, \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c_{\mathsf{Sim}}, \widehat{y}), (\perp, \widehat{z}, r_{\mathsf{Sim}})), \mathsf{aux})$.

$\boxed{\text{Game 2:}}$ We change the witness used by the prover P' to use an actual witness for x, and output $(c_{\mathsf{Sim}}, \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c_{\mathsf{Sim}}, \widehat{y}), (\mathsf{w}, x_\lambda, r)), \mathsf{aux})$. Because Π_{wi} is Sub-WI, this is distinguishable from Game 1 with probability at most ε_{sw}.

$\boxed{\text{Game 3:}}$ We now replace c_{Sim} with c_{P} in Game 2 and output $(c_{\mathsf{P}}, \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c_{\mathsf{P}}, \widehat{y}), (\mathsf{w}, x_\lambda, r)), \mathsf{aux})$. By the hiding property of C, this is distinguishable from Game 2 with probability at most ε_{hide}.

$\boxed{\text{Game 4:}}$ We now simply switch from using randomness r in the witness, to using randomness r_{P}, which gives us the output of the honest prover in the Sub-ZK game: $(c_{\mathsf{P}}, \mathsf{P}'(\mathbf{R}', \mathsf{crs}', (\mathsf{x}, c_{\mathsf{P}}, \widehat{y}), (\mathsf{w}, x_\lambda, r_{\mathsf{P}})), \mathsf{aux})$. By the Sub-WI property of Π_{wi}, this is distinguishable from Game 3 with probability at most ε_{sw}.

Hence Game 1 and Game 4, which are the simulator and prover's output in the Sub-ZK game, respectively, are distinguishable with probability at most $2\varepsilon_{sw} + \varepsilon_{hide}$ which is negligible, hence Π 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 [FO18] 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 [FO18] 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 Ψ is perfectly WI for relation $\mathbf{R}$ and there exists an efficient CRS validation algorithm CV. Then Ψ is Sub-WI.*

Proof. Definition 8 for perfect WI states that for all honestly generated CRS crs (i.e., CRS in the image of $\mathsf{K}(\mathbf{R})$), instances 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 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 crs and $\mathsf{aux}_{\mathcal{Z}}$. Hence assuming perfect WI, verifying that a subverter-generated CRS crs is in fact in the image of $\mathsf{K}(\mathbf{R})$ is enough to assure that perfect subversion WI holds. □

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

Therefore, the efficient SNARK constructions in [ABLZ17, Fuc18], the updatable SNARKs in [GKM+18, MBKM19], and the shuffle argument in [AFK+20] are all Sub-WI. The same observation about Sub-ZK SNARKs was already made by Fuchsbauer in [Fuc18]. 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 [ABLZ17, Fuc18] are statistical ZAPRs with adaptive soundness. The only other pairing-based ZAPR is [LVW20] which is less efficient and uses much more advanced tools, but relies on weaker assumptions for soundness. Secondly, if we use the SNARKs of [ABLZ17, Fuc18] 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* CV *and there exists a VEGOWF. Then there exists a Sub-ZK SNARK.*

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

6 Characterising Sub-ZK NIZKs

We show that the CRS generation algorithm 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 td and maps it deterministically to a 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* ImV*, then Π 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 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 td. Since Π 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 td does not allow simulation, which the subverter could output.

Following [Gro16], 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 $|\varepsilon_0 - \varepsilon_1| \le \varepsilon_{\mathcal{S}}$, 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 [ALSZ20b].

Definition 11 (Trapdoor-Extractability [ALSZ20b]). *A subversion-resistant argument Ψ 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 λ 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 $\implies$ VEOWF $\mathcal{G}_\mathsf{K}$). *Assume Π 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 $\implies$ One-Wayness.** Suppose there exists a PPT adversary $\mathcal{A}$ that breaks one-wayness of $\mathcal{G}_\mathsf{K}$ with probability ε_{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 td' such that $\mathsf{K}_{\mathbf{R},\mathsf{p}}(\mathsf{td}') = \mathsf{crs}$ with probability ε_{owf}.

We are going to construct a PPT adversary $\mathcal{B}$ that internally runs $\mathcal{A}$ together with an auxiliary input 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 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. 9 to quantify the probability that $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1$ in the soundness game.

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 $\mathsf{V}(\mathbf{R}, \mathsf{p}, \mathsf{crs}, \mathsf{x}, \pi) = 1$.

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

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

Game 3: Finally, instead of simulating the proof π, we use the witness w to create an honest proof.

Let us denote the probability of Game i outputting 1 by ε_i. Firstly, it is clear that ε_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 Game 0 and Game 1 succeeds with probability at most $\varepsilon_{\mathcal{S}}$.

Lemma 2. *For the probabilities ε_0 and ε_1 defined as above, $|\varepsilon_0 - \varepsilon_1| \leq \varepsilon_{\mathcal{S}}$.*

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 x_1 is generated by $\mathcal{S}$ and 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 x_0 is ε_0 and given x_1 is ε_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 td' for simulation and the other uses 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 ε_{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 $|\varepsilon_1 - \varepsilon_2| \leq 1 - \varepsilon_{owf}$. □

```
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. 9. Security games for Theorem 8.

The final game transition is based on the zero-knowledge property.

Lemma 4. *Let ε_{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 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 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 aux, outputs with an overwhelming 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

ABLZ17. Behzad Abdolmaleki, Karim Baghery, Helger Lipmaa, and Michal Zajac. A subversion-resistant SNARK. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part III*, volume 10626 of *LNCS*, pages 3–33. Springer, Heidelberg, December 2017. doi:10.1007/978-3-319-70700-6_1.

ABOR00. William Aiello, Sandeep N. Bhatt, Rafail Ostrovsky, and Sivaramakrishnan Rajagopalan. Fast verification of any remote procedure call: Short witness-indistinguishable one-round proofs for NP. In Ugo Montanari, José D. P. Rolim, and Emo Welzl, editors, *ICALP 2000*, volume 1853 of *LNCS*, pages 463–474. Springer, Heidelberg, July 2000. doi:10.1007/3-540-45022-X_39.

AFK+20. Antonis Aggelakis, Prastudy Fauzi, Georgios Korfiatis, Panos Louridas, Foteinos Mergoupis-Anagnou, Janno Siim, and Michal Zajac. A non-interactive shuffle argument with low trust assumptions. In Stanislaw Jarecki, editor, *CT-RSA 2020*, volume 12006 of *LNCS*, pages 667–692. Springer, Heidelberg, February 2020. doi:10.1007/978-3-030-40186-3_28.

ALSZ20a. Behzad Abdolmaleki, Helger Lipmaa, Janno Siim, and Michal Zajac. On QA-NIZK in the BPK model. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020, Part I*, volume 12110 of *LNCS*, pages 590–620. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45374-9_20.

ALSZ20b. Behzad Abdolmaleki, Helger Lipmaa, Janno Siim, and Michał Zając. On subversion-resistant SNARKs. Cryptology ePrint Archive, Report 2020/668, 2020. https://eprint.iacr.org/2020/668.

Bar01. Boaz Barak. How to go beyond the black-box simulation barrier. In *42nd FOCS*, pages 106–115. IEEE Computer Society Press, October 2001. doi:10.1109/SFCS.2001.959885.

BB04. Dan Boneh and Xavier Boyen. Short signatures without random oracles. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 56–73. Springer, Heidelberg, May 2004. doi:10.1007/978-3-540-24676-3_4.

BCC$^+$17. Nir Bitansky, Ran Canetti, Alessandro Chiesa, Shafi Goldwasser, Huijia Lin, Aviad Rubinstein, and Eran Tromer. The hunting of the SNARK. *Journal of Cryptology*, 30(4):989–1066, October 2017. doi:10.1007/s00145-016-9241-9.

BCCT12. Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. From extractable collision resistance to succinct non-interactive arguments of knowledge, and back again. In Shafi Goldwasser, editor, *ITCS 2012*, pages 326–349. ACM, January 2012. doi:10.1145/2090236.2090263.

BCG$^+$14. Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society Press, May 2014. doi:10.1109/SP.2014.36.

BCG$^+$15. Eli Ben-Sasson, Alessandro Chiesa, Matthew Green, Eran Tromer, and Madars Virza. Secure sampling of public parameters for succinct zero knowledge proofs. In *2015 IEEE Symposium on Security and Privacy*, pages 287–304. IEEE Computer Society Press, May 2015. doi:10.1109/SP.2015.25.

BCN$^+$10. Patrik Bichsel, Jan Camenisch, Gregory Neven, Nigel P. Smart, and Bogdan Warinschi. Get shorty via group signatures without encryption. In Juan A. Garay and Roberto De Prisco, editors, *SCN 10*, volume 6280 of *LNCS*, pages 381–398. Springer, Heidelberg, September 2010. doi:10.1007/978-3-642-15317-4_24.

BCPR16. Nir Bitansky, Ran Canetti, Omer Paneth, and Alon Rosen. On the Existence of Extractable One-Way Functions. *SIAM J. Comput.*, 45(5):1910–1952, 2016.

BFJ$^+$20. Saikrishna Badrinarayanan, Rex Fernando, Aayush Jain, Dakshita Khurana, and Amit Sahai. Statistical ZAP arguments. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 642–667. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45727-3_22.

BFS16. Mihir Bellare, Georg Fuchsbauer, and Alessandra Scafuro. NIZKs with an untrusted CRS: Security in the face of parameter subversion. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 777–804. Springer, Heidelberg, December 2016. doi:10.1007/978-3-662-53890-6_26.

BG93. Mihir Bellare and Oded Goldreich. On defining proofs of knowledge. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 390–420. Springer, Heidelberg, August 1993. doi:10.1007/3-540-48071-4_28.

BGGL01. Boaz Barak, Oded Goldreich, Shafi Goldwasser, and Yehuda Lindell. Resettably-sound zero-knowledge and its applications. In *42nd FOCS*, pages 116–125. IEEE Computer Society Press, October 2001. doi:10.1109/SFCS.2001.959886.

BLS04. Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the Weil pairing. *Journal of Cryptology*, 17(4):297–319, September 2004. doi:10.1007/s00145-004-0314-9.

BOV03. Boaz Barak, Shien Jin Ong, and Salil P. Vadhan. Derandomization in cryptography. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 299–315. Springer, Heidelberg, August 2003. doi:10.1007/978-3-540-45146-4_18.

BP15a. Nir Bitansky and Omer Paneth. ZAPs and non-interactive witness indistinguishability from indistinguishability obfuscation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 401–427. Springer, Heidelberg, March 2015. doi:10.1007/978-3-662-46497-7_16.

BP15b. Elette Boyle and Rafael Pass. Limits of extractability assumptions with distributional auxiliary input. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part II*, volume 9453 of *LNCS*, pages 236–261. Springer, Heidelberg, November / December 2015. doi:10.1007/978-3-662-48800-3_10.

CD08. Ran Canetti and Ronny Ramzi Dakdouk. Extractable perfectly one-way functions. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 449–460. Springer, Heidelberg, July 2008. doi:10.1007/978-3-540-70583-3_37.

CD09. Ran Canetti and Ronny Ramzi Dakdouk. Towards a theory of extractable functions. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 595–613. Springer, Heidelberg, March 2009. doi:10.1007/978-3-642-00457-5_35.

CGGM00. Ran Canetti, Oded Goldreich, Shafi Goldwasser, and Silvio Micali. Resettable zero-knowledge (extended abstract). In *32nd ACM STOC*, pages 235–244. ACM Press, May 2000. doi:10.1145/335305.335334.

CL06. Melissa Chase and Anna Lysyanskaya. On signatures of knowledge. In Cynthia Dwork, editor, *CRYPTO 2006*, volume 4117 of *LNCS*, pages 78–96. Springer, Heidelberg, August 2006. doi:10.1007/11818175_5.

CLP13. Kai-Min Chung, Huijia Lin, and Rafael Pass. Constant-round concurrent zero knowledge from P-certificates. In *54th FOCS*, pages 50–59. IEEE Computer Society Press, October 2013. doi:10.1109/FOCS.2013.14.

Dak09. Ramzi Ronny Dakdouk. *Theory and Application of Extractable Functions.* PhD thesis, Yale University, 2009.

Dam92. Ivan Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 445–456. Springer, Heidelberg, August 1992. doi:10.1007/3-540-46766-1_36.

DFGK14. George Danezis, Cédric Fournet, Jens Groth, and Markulf Kohlweiss. Square span programs with applications to succinct NIZK arguments. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 532–550. Springer, Heidelberg, December 2014. doi:10.1007/978-3-662-45611-8_28.

FLP08. Marc Fischlin, Anja Lehmann, and Krzysztof Pietrzak. Robust multi-property combiners for hash functions revisited. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 655–666. Springer, Heidelberg, July 2008. doi:10.1007/978-3-540-70583-3_53.

FLS90. Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *31st FOCS*, pages 308–317. IEEE Computer Society Press, October 1990. doi:10.1109/FSCS.1990.89549.

FO18. Georg Fuchsbauer and Michele Orrù. Non-interactive zaps of knowledge. In Bart Preneel and Frederik Vercauteren, editors, *ACNS 18*, volume 10892 of *LNCS*, pages 44–62. Springer, Heidelberg, July 2018. doi:10.1007/978-3-319-93387-0_3.

FPS20. Georg Fuchsbauer, Antoine Plouviez, and Yannick Seurin. Blind schnorr signatures and signed ElGamal encryption in the algebraic group model. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part II*, volume 12106 of *LNCS*, pages 63–95. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45724-2_3.

Fuc18. Georg Fuchsbauer. Subversion-zero-knowledge SNARKs. In Michel Abdalla and Ricardo Dahab, editors, *PKC 2018, Part I*, volume 10769 of *LNCS*, pages 315–347. Springer, Heidelberg, March 2018. doi:10.1007/978-3-319-76578-5_11.

GGPR13. Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct NIZKs without PCPs. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 626–645. Springer, Heidelberg, May 2013. doi:10.1007/978-3-642-38348-9_37.

GJJM20. Vipul Goyal, Abhishek Jain, Zhengzhong Jin, and Giulio Malavolta. Statistical zaps and new oblivious transfer protocols. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 668–699. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45727-3_23.

GKM$^+$18. Jens Groth, Markulf Kohlweiss, Mary Maller, Sarah Meiklejohn, and Ian Miers. Updatable and universal common reference strings with applications to zk-SNARKs. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 698–728. Springer, Heidelberg, August 2018. doi:10.1007/978-3-319-96878-0_24.

GL89. Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In *21st ACM STOC*, pages 25–32. ACM Press, May 1989. doi:10.1145/73007.73010.

GMR85. Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems (extended abstract). In *17th ACM STOC*, pages 291–304. ACM Press, May 1985. doi:10.1145/22145.22178.

GO94. Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. *Journal of Cryptology*, 7(1):1–32, December 1994. doi:10.1007/BF00195207.

GOS06. Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for NIZK. In Cynthia Dwork, editor, *CRYPTO 2006*, volume 4117 of *LNCS*, pages 97–111. Springer, Heidelberg, August 2006. doi:10.1007/11818175_6.

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008. doi:10.1145/1374376.1374407.

Gro06. Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 444–459. Springer, Heidelberg, December 2006. doi:10.1007/11935230_29.

Gro10. Jens Groth. Short pairing-based non-interactive zero-knowledge arguments. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 321–340. Springer, Heidelberg, December 2010. doi:10.1007/978-3-642-17373-8_19.

Gro16. Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Heidelberg, May 2016. doi:10.1007/978-3-662-49896-5_11.

GW11. Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 99–108. ACM Press, June 2011. doi:10.1145/1993636.1993651.

Her05. Amir Herzberg. On tolerant cryptographic constructions. In Alfred Menezes, editor, *CT-RSA 2005*, volume 3376 of *LNCS*, pages 172–190. Springer, Heidelberg, February 2005. doi:10.1007/978-3-540-30574-3_13.

HKN$^+$05. Danny Harnik, Joe Kilian, Moni Naor, Omer Reingold, and Alon Rosen. On robust combiners for oblivious transfer and other primitives. In Ronald Cramer, editor, *EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 96–113. Springer, Heidelberg, May 2005. doi:10.1007/11426639_6.

JLS20. Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from well-founded assumptions. Cryptology ePrint Archive, Report 2020/1003, 2020. https://eprint.iacr.org/2020/1003.

JR13. Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 1–20. Springer, Heidelberg, December 2013. doi:10.1007/978-3-642-42033-7_1.

KNYY19. Shuichi Katsumata, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. Exploring constructions of compact NIZKs from various assumptions. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part III*, volume 11694 of *LNCS*, pages 639–669. Springer, Heidelberg, August 2019. doi:10.1007/978-3-030-26954-8_21.

KPY19. Yael Tauman Kalai, Omer Paneth, and Lisa Yang. How to delegate computations publicly. In Moses Charikar and Edith Cohen, editors, *51st ACM STOC*, pages 1115–1124. ACM Press, June 2019. doi:10.1145/3313276.3316411.

KRR14. Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. How to delegate computations: the power of no-signaling proofs. In David B. Shmoys, editor, *46th ACM STOC*, pages 485–494. ACM Press, May / June 2014. doi:10.1145/2591796.2591809.

Lam79. Leslie Lamport. Constructing digital signatures from a one-way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory, October 1979.

Lep02. Matthew Lepinski. On the existence of 3-round zero-knowledge proofs. Master's thesis, MIT, USA, 2002.

Lip12. Helger Lipmaa. Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 169–189. Springer, Heidelberg, March 2012. doi:10.1007/978-3-642-28914-9_10.

LVW20. Alex Lombardi, Vinod Vaikuntanathan, and Daniel Wichs. Statistical ZAPR arguments from bilinear maps. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 620–641. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45727-3_21.

Mau05. Ueli M. Maurer. Abstract models of computation in cryptography (invited paper). In Nigel P. Smart, editor, *10th IMA International Conference on Cryptography and Coding*, volume 3796 of *LNCS*, pages 1–12. Springer, Heidelberg, December 2005.

MBKM19. Mary Maller, Sean Bowe, Markulf Kohlweiss, and Sarah Meiklejohn. Sonic: Zero-knowledge SNARKs from linear-size universal and updatable structured reference strings. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 2111–2128. ACM Press, November 2019. doi:10.1145/3319535.3339817.

Mic00. Silvio Micali. Computationally Sound Proofs. *SIAM J. Comput.*, 30(4):1253–1298, 2000.

RSA78. Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the Association for Computing Machinery*, 21(2):120–126, 1978.

RTYZ16. Alexander Russell, Qiang Tang, Moti Yung, and Hong-Sheng Zhou. Cliptography: Clipping the power of kleptographic attacks. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 34–64. Springer, Heidelberg, December 2016. `doi:10.1007/978-3-662-53890-6_2`.

Sch90. Claus-Peter Schnorr. Efficient identification and signatures for smart cards. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 239–252. Springer, Heidelberg, August 1990. `doi:10.1007/0-387-34805-0_22`.

SS01. Ahmad-Reza Sadeghi and Michael Steiner. Assumptions related to discrete logarithms: Why subtleties make a real difference. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 244–261. Springer, Heidelberg, May 2001. `doi:10.1007/3-540-44987-6_16`.

WW20. Hoeteck Wee and Daniel Wichs. Candidate obfuscation via oblivious LWE sampling. Cryptology ePrint Archive, Report 2020/1042, 2020. `https://eprint.iacr.org/2020/1042`.

A Universal Delegation for Deterministic Computations

The following definition is taken from [BCPR16].

Let $\mathcal{L}_U$ be the universal language consisting of all tuples (M, x, t), such that machine M accepts x within time t. Let $\mathcal{L}_U(T)$ be the set of all pairs (M, x), such that $(M, x, T) \in \mathcal{L}_U$.

Let $T(\lambda) \in (2^{\omega(\log \lambda)}, 2^{\mathsf{poly}(\lambda)})$ be a computable superpolynomial function. A universal delegation system for $\mathbf{DTIME}(T)$ consists of three algorithms $(\mathsf{K}, \mathsf{P}, \mathsf{V})$, where

- The probabilistic generator K, given 1^λ, outputs a CRS $\mathsf{crs}_{\mathsf{DS}}$ and corresponding verification state τ. K is independent of any statement, proven later.
- The honest prover $\mathsf{P}(\mathsf{crs}_{\mathsf{DS}}; M, x)$ produces a certificate π that $(M, x) \in \mathcal{L}_U(T)$.
- The verifier $\mathsf{V}(\mathsf{crs}_{\mathsf{DS}}, \tau; M, x; \pi)$ verifies the validity of $(M, x) \in \mathcal{L}_U(T)$.

A universal delegation system $(\mathsf{K}, \mathsf{P}, \mathsf{V})$ for $\mathbf{DTIME}(T)$ is secure if it satisfies the following conditions:

Perfect completeness: for any λ, $(M, x) \in \mathcal{L}_u(T(\lambda))$, $(\mathsf{crs}_{\mathsf{DS}}, \tau) \leftarrow \mathsf{K}(1^\lambda)$, $\pi \leftarrow \mathsf{P}(\mathsf{crs}_{\mathsf{DS}}; M, x)$, it holds that $\mathsf{V}(\mathsf{crs}_{\mathsf{DS}}, \tau; M, x; \pi) = 1$.

Adaptive soundness for a bounded number of statements: there is a polynomial $\mathfrak{b}$, such that for any poly-size P^*, and any set of at most $2^{\mathfrak{b}(\lambda)}$ false statements $S \subseteq \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \mathcal{L}_U(T(\lambda))$,

$$\Pr\left[\begin{matrix}(\mathsf{crs}_{\mathsf{DS}}, \tau) \leftarrow \mathsf{K}(1^\lambda), (M, x, \pi) \leftarrow \mathsf{P}^*(\mathsf{crs}_{\mathsf{DS}}) \text{ s.t. } (M, x) \in S : \\ \mathsf{V}(\mathsf{crs}_{\mathsf{DS}}, \tau; M, x; \pi) = 1\end{matrix}\right] \leq \mathsf{negl}(\lambda) \enspace .$$

Efficiency: there exists a polynomial p, such that for every λ, $t \leq T(\lambda)$, and $(M, x) \in \mathcal{L}_U(t)$,

- K runs in time $p(\lambda)$,
- V runs in time $p(\lambda + |M| + |x|)$,
- P runs in time $p(\lambda + |M| + |x| + t)$.

The scheme is *publicly verifiable* if soundness is maintained when the malicious prover is also given the verification state τ. In this case, we will assume that τ appears in the clear in the reference string $\mathsf{crs}_{\mathsf{DS}}$.

For example, [KPY19] is a delegation scheme based on falsifiable assumptions.

B Keyless Non-Black-Box Extractable Commitments

Inspired by Canetti and Dakdouk [CD09], we show how to construct a non-interactive keyless extractable commitment from an injective (verifiably-)extractable keyless OWF g_λ. Clearly, black-box extraction is impossible since there is no secret key for the extractor to use. Thus, we consider a non-black-box extractability definition where the extractor can depend on the adversary.

Let us observe that an EOWF is essentially a trapdoor function where the trapdoor is the adversary's code and the auxiliary input is aux. There are well-known constructions for obtaining public key encryption from

```
$\mathsf{Com}_\lambda(b)$    // $b \in \{0,1\}$
    $r \leftarrow\!\$\, \{0,1\}^{2\ell}$;
    return $(\mathsf{g}'_\lambda(r), B(r) \oplus b)$
```

```
$\mathsf{ComV}_\lambda(c)$
    parse $c = (\widehat{y}, \widehat{b})$;
    return $\widehat{y} \in \mathrm{image}(\mathsf{g}'_\lambda) \wedge \widehat{b} \in \{0,1\}$
```

Fig. 10. Keyless extractable bit commitment.

any trapdoor function and a hard-core predicate. We are going to use a similar approach to obtain a keyless extractable commitment from a (verifiably-)extractable OWF and a hard-core predicate. In particular, if the EOWF is verifiable, then the commitment will have an efficient verification algorithm ComV_λ. We recall the definition of a hard-core predicate of a one-way function.

Definition 12. *We say that an OWF $\mathsf{g}\colon X \to Y$ has a hard-core predicate $B\colon X \to \{0,1\}$, if for any PPT adversary $\mathcal{A}$, $\varepsilon_0 \approx_\lambda \varepsilon_1$, where*

$$\varepsilon_b := \Pr[r \leftarrow\!\$\, X, u_0 = B(r), u_1 \leftarrow\!\$\, \{0,1\} : \mathcal{A}(\mathsf{g}(r), u_b) = 1] \enspace .$$

Let $\mathsf{g}_\lambda\colon \{0,1\}^\ell \to \{0,1\}^{\ell'}$ be a verifiably-extractable injective OWF. (See Sections 4.4 and 4.5 for examples of such functions.) Using the Goldreich–Levin construction [GL89], we can obtain a verifiably-extractable OWF g'_λ with a hard-core predicate B. The Goldreich–Levin construction defines $\mathsf{g}'_\lambda(x, y) := (\mathsf{g}_\lambda(x), y)$ where $|x| = |y| = \ell$ and $B(x, y) = \sum_{i=1}^{\ell} x_i \cdot y_i \bmod 2$. Note that g'_λ is still a verifiably-extractable OWF since x can be extracted from $\mathsf{g}_\lambda(x)$ and y is public. We use g'_λ and B to construct a simple non-black-box extractable bit commitment scheme that is shown in Fig. 10.

Theorem 9. *The bit commitment scheme in Fig. 10 has the following properties:*

1. *If g'_λ is injective, then it is perfectly binding.*
2. *If g'_λ is an OWF and B is its hard-core predicate, then g'_λ is computationally hiding.*
3. *If g'_λ is verifiably-extractable with respect to an auxiliary distribution $\mathcal{D}$ and injective, then it is non-black-box extractable with respect to the same distribution.*

Proof. **Perfect Binding:** Suppose an adversary outputs two valid openings (r_1, b_1) and (r_2, b_2) to the same commitment, i.e., $(\mathsf{g}'_\lambda(r_1); B(r_1) \oplus b_1) = (\mathsf{g}'_\lambda(r_2); B(r_2) \oplus b_2)$. Since g'_λ is injective, then $\mathsf{g}'_\lambda(r_1) = \mathsf{g}'_\lambda(r_2)$ implies $r_1 = r_2$. Then $B(r_1) = B(r_2)$ so $b_1 = b_2$.

Computational hiding: Let $\mathcal{A}$ be a PPT adversary that tries to break computational hiding. Suppose $\mathcal{A}$ outputs messages $m_0, m_1 \in \{0,1\}$. Let ε_i denote the probability that $\mathcal{A}$ wins in $\mathsf{Game}\ i$.

Game 0: This is the original computational hiding game where $c = \mathsf{Com}(m_0; r) = (\mathsf{g}'_\lambda(r); m_0 \oplus B(r))$ for $r \leftarrow\!\$\, \{0,1\}^{2\ell}$.

Game 1: Now instead of using $B(r)$, we sample a random bit $u \leftarrow\!\$\, \{0,1\}$ and compute the commitment as $c = (\mathsf{g}'_\lambda(r); m_0 \oplus u)$. Clearly, $|\varepsilon_0 - \varepsilon_1| \le \varepsilon_{hc}$ where ε_{hc} denotes the advantage a PPT adversary has in distinguishing $B(r)$ from a uniform bit.

Game 2: We change the previous game and commit to m_1 instead, that is, $c = (\mathsf{g}'_\lambda(r); m_1 \oplus u)$. Since u is uniformly random, we have that $\varepsilon_1 = \varepsilon_2$.

Game 3: We change c back to a real commitment, i.e., $c = (\mathsf{g}'_\lambda(r); m_1 \oplus B(r))$. Clearly again, $|\varepsilon_2 - \varepsilon_3| \le \varepsilon_{hc}$.

Using the triangle inequality, we get that the advantage $\mathcal{A}$ has in breaking computational hiding is bounded by $2\varepsilon_{hc} \approx_\lambda 0$.

Non-black-box extractability: Let $\mathcal{A}(1^\lambda, \mathsf{aux})$ be a PPT adversary that outputs a commitment $c = (\widehat{y}, \widehat{b})$ such that $\mathsf{ComV}_\lambda(c) = 1$. Then $\widehat{y} \in \mathrm{image}(\mathsf{g}'_\lambda)$ and $\widehat{b} \in \{0,1\}$. According to the definition of verifiable-extractability, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{g}'_\lambda}$ that given $\mathcal{A}$'s auxiliary input aux, outputs r such that $\widehat{y} = \mathsf{g}'_\lambda(r)$ with overwhelming probability. Thus we can construct an extractor $\mathsf{Ext}_{\mathcal{A}}(1^\lambda, \mathsf{aux})$ that runs $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{g}'_\lambda}(1^\lambda, \mathsf{aux})$ to recover a unique r, then returns $\widehat{b} \oplus B(r)$. Extractor $\mathsf{Ext}_{\mathcal{A}}$ succeeds with the same probability as $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{g}'_\lambda}$. □

```
$\mathsf{Com}(m)$    // $m \in \{0,1\}^{tk}$
parse $m = m_1 \| \dots \| m_k$    // $m_i \in \{0,1\}^t$;
for $i = 1, \dots, k$ :    $r_i \leftarrow_\$ \{0,1\}^{2\ell+t-1}$;
  $c_i = (\mathsf{g}^*_\lambda(r_i), f(r_i) \oplus m_i)$;
return $(c_1, \dots, c_k)$;
```

```
$\mathsf{ComV}(c)$
parse $c = (c_1, \dots, c_k)$
for $i \in [1 \mathinner{..} k]$
  parse $c_i = (\widehat{y}_i, \widehat{s}_i)$;
  if $\widehat{y}_i \notin \mathrm{image}(\mathsf{g}^*_\lambda) \vee \widehat{s}_i \notin \{0,1\}^t$ then
return 0 else return 1
```

Fig. 11. Keyless extractable commitment for bit strings.

Remark 3. Note that if we have the usual notion of extractability, then ComV_λ might be inefficient.

The bit commitment scheme above can be extended to arbitrary-length messages by simply committing to each bit of the message. However, this would be very inefficient. Instead of the hard-core predicate B one can use a hard-core function f, i.e., a function which produces t hard-core bits and thus allows to commit to messages of length t. Goldreich and Levin proposed the following construction. Let $\mathsf{g}_\lambda \colon \{0,1\}^\ell \to \{0,1\}^{\ell'}$ be a one-way function, then so is $\mathsf{g}^*_\lambda(x,y) := (\mathsf{g}(x), y)$ where $|y| = \ell + t(\ell) - 1$ and $t := t(\ell) := \min\{\ell, c \cdot \lceil \log_2 \ell \rceil\}$ for any constant $c > 0$. In general, then $t(\ell) = \mathcal{O}(\log_2 \ell)$. They define the hard-core function as $f(x,y) := (b_1, \dots, b_t)$ where $b_i = B(x, (y_i, \dots, y_{i+\ell-1})) = \sum_{j=1}^{\ell} x_j y_{i+j-1} \bmod 2$. The construction of the multi-bit commitment scheme is given in Fig. 11. The security proof is similar to the bit commitment case.

Theorem 10. *The commitment scheme in Fig. 11 has the following properties:*

1. *If g^*_λ is injective, then it is perfectly binding.*
2. *If g^*_λ is a OWF and f is its hard-core function, then it is computationally hiding.*
3. *If g^*_λ is verifiably-extractable respect to an auxiliary distribution $\mathcal{D}$ and injective, then it is non-black-box extractable respect to the same distribution*

C ABLZ EOWF

We recall that the ABLZ EOWF (based on [ABLZ17]) is defined as $\mathsf{g}_\mathsf{p}(x) := ([x]_1, [x]_2)$, where p is an asymmetric bilinear group (esp. we assume that $([1]_1, [1]_2) \in \mathsf{p}$), and we recall the definitions of the SDL and BDH-KE assumptions.

The one-way property of the ABLZ EOWF relies on the *Symmetric Discrete Logarithm (SDL) [BCN+10] assumption* which holds relative to Pgen, if for any PPT $\mathcal{A}$,

$$\Pr\left[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); x \leftarrow_\$ \mathbb{Z}_p : \mathcal{A}(\mathsf{p}, [x]_1, [x]_2) = x\right] \leq \mathsf{negl}(\lambda) \ .$$

The *BDH-KE* assumption [ABLZ17] states that for any PPT adversary $\mathcal{A}$, there exists a PPT extractor $\mathsf{Ext}_\mathcal{A}$, such that

$$\Pr\begin{bmatrix} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); r \leftarrow \mathsf{RND}_\lambda(\mathcal{A}); ([a]_1, [a']_2) \leftarrow \mathcal{A}(\mathsf{p}; r); \\ a'' \leftarrow \mathsf{Ext}_\mathcal{A}(\mathsf{p}; r) : [a]_1 \bullet [1]_2 = [1]_1 \bullet [a']_2 \wedge a \neq a'' \end{bmatrix} \leq \mathsf{negl}(\lambda) \ .$$

Lemma 5. *Let $\mathcal{G} = \{\mathsf{g}_\mathsf{p}\}_{\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda), \lambda \in \mathbb{N}}$ be the ABLZ EOWF. The family $\mathcal{G}$ is a VEOWF.*

Proof. **One-wayness:** Assume that $\mathcal{B}$ is an adversary that given $([x]_1, [x]_2)$, for a random $x \leftarrow_\$ \mathbb{Z}_p$, outputs x with non-negligible probability. Then $\mathcal{B}$ breaks the SDL assumption.

Verifiable-extractability with respect to aux of length $\mathfrak{b}(\lambda)$: Let $\mathcal{A}_{ve}$ be an adversary that breaks verifiable-extractability with some probability upper-bounded by ε_{ve}. We use security of the BDH-KE assumption to show that ε_{ve} is negligible. Let ε_{bdh} be the upper-bound for any PPT adversary $\mathcal{A}_{bdh}$ (that does not take any advice) to break the BDH-KE assumption. That is, an extractor $\mathsf{Ext}_{bdh}(\mathsf{p})$ fails with probability at most ε_{bdh}. Assume $\mathcal{A}_{bdh}$ proceeds as follows.

First it samples the auxiliary input $\mathsf{aux} \leftarrow \mathcal{D}_\lambda$ then runs $\mathcal{A}_{ve}(\mathsf{p}, \mathsf{aux})$ and gets $([x]_1, [x]_2)$ which it outputs. Since with probability ε_{ve} no extractor can reveal x we get that $\varepsilon_{bdh} \geq \varepsilon_{ve}$. Since ε_{ve} is non-negligible, ε_{bdh} is also non-negligible. □

Remark 4 (Securing BDH-KE assumption against malicious aux). The limiting restriction on the adversary's auxiliary input can be lifted if we assume that it is set prior to the choice of the groups generators. That is, first the adversary gets its advice, then the generators are picked. Intuitively, in this setting no auxiliary input could given an advantage in breaking the assumption, even if $\mathcal{A}$ gets $\mathfrak{g}^a \in \mathbb{G}_1, \mathfrak{h}^a \in \mathbb{G}_2$. Since the parameters are fixed afterwards and group generators are picked randomly we have $[1]_1 = \mathfrak{g}^\alpha, [1]_2 = \mathfrak{h}^\beta$, for some random α, β; furthermore $\mathfrak{g}^a = [a/\alpha]_1$ and $\mathfrak{h}^a = [a/\beta]_2$. Thus with overwhelming probability, $[a/\alpha]_1 \bullet [1]_2 \neq [1]_1 \bullet [a/\beta]_2$.

Note that the generic (bilinear) group model (G(B)GM) allows for such a setup. As proposed by Maurer in [Mau05], in G(B)GM an adversary $\mathcal{A}$ is given access to an oracle O that performs group operations on behalf of $\mathcal{A}$, who does not see any information about the structure nor binary representation of group elements at all. That is, the only information $\mathcal{A}$ gets about the group elements is in which cells of memory of O the group elements are stored. Unfortunately, in real life assuming that the adversary knows nothing about (say) a group generator before the protocol starts may be unreasonable.

From ABLZ to a more general class of functions. In general, one can replace the ABLZ function $\mathcal{G}$ with any similar function from integers to group elements, where one can argue in the generic group model extractability, and where one can efficiently verify whether some element belongs to the image of $\mathcal{G}$ or not. In the pairing-based setting, similar functions have been studied in say [BCG+15] (in the context of generating SNARK CRSs by using MPC; however, here extractability is not needed) and [ABLZ17] (in the context of Sub-ZK SNARKs).

One can have $\mathcal{G} = \{\mathsf{g}_\mathsf{e}\}_\lambda$ from the following class $\mathsf{BDHClass}$ of function families, where x_i are secret trapdoors and $\mathsf{e} = \mathsf{p}$. Here, we require that

$$\mathsf{g}_\mathsf{e}(\vec{x}) = ([\varrho(\vec{x}) : \varrho \in \mathcal{R}]_1, [\sigma(\vec{x}) : \sigma \in \mathcal{S}]_2, [\tau(\vec{x}) : \tau \in \mathcal{T}]_T)$$

for three sets of polynomials $\mathcal{R}$, $\mathcal{S}$, $\mathcal{T}$ from $\mathbb{Z}_p[\vec{X}]$, such that

(i) $X_i \in \mathcal{R} \cap \mathcal{S}$ for each i,
(ii) If $f(\vec{X}) \in \mathcal{R} \cup \mathcal{S} \cup \mathcal{T}$ has degree $d > 0$, then there exist polynomials ϱ, σ of degree $< d$, such that $f(\vec{X}) = \varrho(\vec{X})\sigma(\vec{X})$ and $\varrho(\vec{X}) \in \mathcal{R}, \sigma(\vec{X}) \in \mathcal{S}$.

Here, Item i guarantees that one can use the BDH-KE assumption for extractability, and Item ii guarantees verifiability. One can replace Item i with some other bootstrapping requirement if one wishes to use a different knowledge assumption. Most generally, one can omit Item i and rely on a tautological knowledge assumption (that then has to be proven secure in the generic group model).