Lattice-Based SNARKs: Publicly Verifiable, Preprocessing, and Recursively Composable

(Extended Abstract)

Martin R. Albrecht[1], Valerio Cini[2(✉)], Russell W. F. Lai[3], Giulio Malavolta[4], and Sri AravindaKrishnan Thyagarajan[5]

[1] Royal Holloway, University of London, Egham, UK
[2] AIT Austrian Institute of Technology, Seibersdorf, Austria
valerio.cini@ait.ac.at
[3] Aalto University, Espoo, Finland
[4] Max Planck Institute for Security and Privacy, Bochum, Germany
[5] Carnegie Mellon University, Pittsburgh, USA

Abstract. A succinct non-interactive argument of knowledge (SNARK) allows a prover to produce a short proof that certifies the veracity of a certain NP-statement. In the last decade, a large body of work has studied candidate constructions that are secure against quantum attackers. Unfortunately, no known candidate matches the efficiency and desirable features of (pre-quantum) constructions based on bilinear pairings.

In this work, we make progress on this question. We propose the first lattice-based SNARK that simultaneously satisfies many desirable properties: It (i) is tentatively post-quantum secure, (ii) is publicly-verifiable, (iii) has a logarithmic-time verifier and (iv) has a purely algebraic structure making it amenable to efficient recursive composition. Our construction stems from a general technical toolkit that we develop to translate pairing-based schemes to lattice-based ones. At the heart of our SNARK

M. R. Albrecht—The research of MA was supported by EPSRC grants EP/S020330/1, EP/S02087X/1 and by the European Union Horizon 2020 Research and Innovation Program Grant 780701.

M. R. Albrecht, V. Cini, R. W. F. Lai, G. Malavolta, S. A. Thyagarajan—This work was supported by Protocol Labs under PL-RGP1-2021-050.

V. Cini—This work was in part done while visiting Max Planck Institute for Security and Privacy. The research of VC was in part funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 830929 (CyberSec4Europe), No. 871473 (KRAKEN), and by the Austrian Science Fund (FWF) and netidee SCIENCE grant P31621-N38 (PROFET).

R. W. F. Lai—This work was done at Friedrich-Alexander-Universität Erlangen-Nürnberg.

G. Malavolta—This work has been partially supported by the German Federal Ministry of Education and Research BMBF (grant 16K15K042, project 6GEM).

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-15979-4_4.

Y. Dodis and T. Shrimpton (Eds.): CRYPTO 2022, LNCS 13508, pp. 102–132, 2022.
https://doi.org/10.1007/978-3-031-15979-4_4

is a new lattice-based vector commitment (VC) scheme supporting openings to constant-degree multivariate polynomial maps, which is a candidate solution for the open problem of constructing VC schemes with openings to beyond linear functions. However, the security of our constructions is based on a new family of lattice-based computational assumptions which naturally generalises the standard Short Integer Solution (SIS) assumption.

1 Introduction

A succinct non-interactive argument of knowledge (SNARK) [45,58] allows a prover to convince a verifier that they know a witness to an NP statement. The succinctness property demands that the size of the proof and, after preprocessing, the work of the verifier are sublinear in (ideally independent of) the time needed to check the validity of the witness. Over the last decade, SNARKs have witnessed a meteoric rise in their efficiency and applicability [9,11,13,22,30,62]. More recently, SNARKs have found their way into real-world systems in the context of blockchain-based cryptocurrencies [10,15,18,20,47].

The looming threat of quantum computers has given rise to a movement in the cryptographic community to investigate cryptographic constructions from assumptions that would plausibly withstand the presence of a quantum attacker. Unfortunately, present SNARKs based on post-quantum assumptions are in many ways inferior to pre-quantum constructions based on bilinear pairings. The goal of this work is to make progress in this area.

1.1 The Seascape of SNARKs

To put our work into context, we give a brief outline of the current seascape of SNARK constructions[1]. We split the schemes depending on the underlying cryptographic assumptions used as the source of hardness.

Bilinear Pairings. To date, the most efficient and feature-rich SNARKs are constructed over bilinear pairing groups (e.g. [42]) with a trusted setup. Typically, a pairing-based SNARK proof consists of only a small constant number of base group elements and is also publicly verifiable. Furthermore, offline preprocessing can often be performed, such that the online verification time is sublinear in the size of the statement being proved and the corresponding witness. Moreover, pairing-based SNARKs are favourable because of their algebraic structures that is known to enable proof batching [21,50] and efficient recursive composition [12]. However, due to their reliance on the hardness of problems related to discrete logarithms, pairing-based SNARKs are not sound against a cheating quantum prover.

Random Oracles. Promising post-quantum candidate for SNARKs are constructions based on Micali's CS proofs paradigm: They are obtained by first building an interactive argument using (generalisations of) probabilistically checkable proofs (PCP) [45], then compiling it into a non-interactive one using the Fiat-Shamir transformation [27] in the random oracle (RO) model.

A major difference between pairing-based and RO-based SNARKs, from both theoretical and practical perspectives, is the algebraic structure of the

[1] It can be succinctly verified that SNARKs, like sharks, are creatures of the sea.

verification algorithm. In RO-based SNARKs, the verification algorithms query the RO, which is a combinatorial object. This is especially important when recursively composing the SNARK: On the theoretical side, proving the knowledge of a valid RO-based SNARK proof requires specifying the circuit computing the RO. This makes it challenging to formally argue about soundness, even in the RO model. From a practical perspective, the RO is instantiated with cryptographic hash functions, which typically have high multiplicative degree.[2] Since the multiplicative degree of the relation being proven often dominates the prover computation complexity in SNARKs, proving the satisfiability of a cryptographic hash function becomes computationally expensive.

Lattices. A prominent source of hardness for post-quantum security are computational problems over lattices. Not only do lattice-based assumptions allow us to build most standard cryptographic primitives, e.g. [34,66], but also enable new powerful primitives [33,38,39,72], which are currently out of the reach of group-based assumptions. Unfortunately, in the context of SNARKs, lattices have yet to be established as competitive alternatives to group-based constructions. So far, lattice-based SNARKs either require designated verifiers [32,43] or linear-time verification [6,19].

Beyond their theoretical appeal, one additional motivation for constructing lattice-based SNARKs is that they are potentially more compatible with other basic lattice-based primitives when composing them to construct more advanced systems. More concretely, consider the task of proving the satisfiability of certain algebraic relations over a ring $\mathcal{R}$ by a solution vector of norm bounded by some δ, a language which arises naturally when composing lattice-based building blocks. Using an argument system for proving algebraic relations over a finite field without norm constraints, arithmetisation would be needed to express certain witness component in, say, binary representation and translate the bounded-norm condition to the satisfiability of a potentially-high-degree polynomial, depending on the choice of the norm and the norm bound δ. In contrast, the bounded-norm constraint could be proven natively if we have an argument system which supports proving the satisfiability of algebraic relations over $\mathcal{R}$ by solutions of norm bounded by some $\alpha \leq \delta$. This is done by expressing the solution vector in a likely more compact $O(\alpha)$-ary representation such that, if the representation has norm bounded by α, then the original solution has norm bounded by δ.

1.2 Our Contributions

In this work, we construct the first lattice-based SNARK for an NP-complete language defined over a ring $\mathcal{R}$. Specifically, the language being supported is the satisfiability of polynomial maps over $\mathcal{R}$ by bounded-norm solutions. Our construction qualitatively matches pairing-based SNARKs, i.e. it is publicly verifiable and can achieve sublinear verification time given preprocessing, while requiring a trusted setup. In addition, it is tentatively post-quantum secure. Furthermore, our construction uses only algebraic operations over a ring $\mathcal{R}$, and is therefore friendly to recursive composition. The soundness of our scheme is

[2] Though we mention that there is recent progress [5,40] in crafting hash functions that are friendlier to multiparty computation and argument systems.

based on new lattice-based (knowledge) assumptions. The introduction of new knowledge assumptions is, to some extent, necessary: The work of Gentry and Wichs [35] shows that the soundness of any SNARK cannot be based on falsifiable assumptions in a black-box manner. We summarise the main steps of our work in the following.

(1) Translation Technique. We put forward a new paradigm for translating pairing-based constructions to the lattice world. Our constructions stem from techniques from the literature on pairing-based cryptography [53], while simultaneously exploiting the ring structure offered by the lattice setting. We develop the necessary technical toolkit that helps us mimic operations of pairing-based VC constructions in the lattice setting. We view this translation strategy as a major conceptual contribution of our work and we expect it to be instrumental in enabling new applications of lattice-based cryptography.

(2) Vector Commitments for Constant-Degree Polynomials. A vector commitment (VC) allows a committer to commit to a vector of w values $\mathbf{x} := (x_0, \ldots, x_{w-1}) \in \mathcal{R}^w$ and then reveal selected portions of the input vector, or more generically a function $f : \mathcal{R}^w \to \mathcal{R}^t$ over the input vector, along with a proof π that can be publicly verified. We require both the commitment and the opening proof to be *compact*. In terms of security, we want to ensure an adversary cannot output a valid opening proof for an incorrect function evaluation of the input vector. VCs have been established as a central primitive in cryptography [23,24,29,37,49,52]. As a central technical contribution, we present the first (lattice-based) VC that supports openings beyond linear functions. Specifically, our VC commits to short vectors of ring elements $\mathbf{x} \in \mathcal{R}^w$ and supports openings to constant-degree d multivariate polynomial maps. We then show how this VC is sufficient to construct SNARKs for the satisfiability of degree-d polynomial maps (which is NP-complete for $d \geq 2$) by bounded-norm solutions.

(3) New Assumptions and Analysis. Our translation techniques (and consequently the resulting cryptographic schemes) rely on a new family of assumptions that we refer to as the *k-Ring-Inhomogenous Short Integer Solution* (or k-R-ISIS for short) assumptions. Roughly, a k-R-ISIS assumption says that it is hard to find a short preimage $\mathbf{u}_{g^*}$ satisfying $\langle \mathbf{a}, \mathbf{u}_{g^*} \rangle = g^*(\mathbf{v}) \bmod q$, where g^* is a Laurent monomial[3] and $\mathbf{v}$ is a random point, given short preimages of other Laurent monomials $\mathcal{G}$ evaluated on the same random point. Our new assumptions can be viewed as inhomogenous ring variants of the k-SIS assumption [17,54] (where the rational functions are zeros). The key difference to k-SIS is that we allow to hand out more preimages than the dimension of $\mathbf{a}$ but these preimages are all of different images.

In fact, the assumptions we introduce, k-M-ISIS, are slightly more general in being defined over modules rather than rings. Our generalisation to modules

[3] A Laurent monomial is a monomial where negative powers are allowed. Generally, one could consider k-R-ISIS problems for rational functions.

is motivated by the knowledge assumptions that we also introduce. In the knowledge assumptions images live in a moderately sized submodule.

We consider the introduction and study of the k-R-ISIS assumptions as a contribution to the programme of charting the territory between LWE and multilinear maps assumptions called for in [1].

To gain confidence in our newly introduced assumptions, we initiate their study. We show that certain subclasses of the k-R-ISIS problems (parameterised by the algebraic structure on the k-R-ISIS images) are as hard as the R-SIS problem. We show that, as expected, the k-M-ISIS problems are as hard as their k-R-ISIS counterparts, although the former have slightly skewed parameters. We also show that certain k-M-ISIS problems are as hard as the k-M-SIS problem, the natural module variant of the k-SIS problem, where the former have higher module ranks. Furthermore, we show that the k-M-ISIS problems for $(\mathcal{G}, g^*)$ is as hard as those for $(\mathcal{G}, 0)$, and that the hardness is preserved when scaling both $\mathcal{G}$ and g^* multiplicatively by any non-zero Laurent monomial.

However, since none of the reductions from well-established problems cover the case we rely upon in our constructions, we perform cryptanalysis to assess the hardness of general k-M-ISIS problems. While we did not identify any structural weaknesses, we encourage independent analysis to gain confidence in or invalidate our assumptions.

(4) Post-Quantum Security. As a contribution of independent interest, we show that our VC satisfies a strong notion of binding known as *collapsing* (as an ordinary commitment, not with respect to functional openings), a recently introduced security notion in the quantum setting [70]. For this, we introduce a new technique of embedding NTRU ciphertexts into the public parameters of our VC. To the best of our knowledge, this is the first VC not based on Merkle trees that is shown to satisfy such a notion.

(5) New Applications. Our SNARK supports proving the satisfiability of polynomial maps over $\mathcal{R}$ by bounded-norm solutions, a language which directly captures those statements which naturally arise in lattice-based cryptographic constructions. We highlight two native applications of our SNARK which do not rely on expensive conversions between different NP-complete languages.

The first application is the recursive composition of our SNARK, which refers to the process of using the SNARK to prove knowledge of another SNARK proof and the satisfiability of a polynomial map; for details see the full version. This is natively supported because the verification algorithm of our SNARK construction is itself checking the satisfiability of certain algebraic relations over $\mathcal{R}$ by a bounded-norm solution. Recursive composition of SNARKs is a general purpose technique for aggregating proofs or proving complex statements in a piece-by-piece fashion. The technique is also useful for constructing incremental verifiable computation [71] and verifiable delay functions [14,41].

The second application is the aggregation of GPV signatures [34]. While it is folklore that any signatures can be aggregated by a SNARK for an NP-complete language, we stress that the GPV verification algorithm, again, checks

the satisfiability of certain algebraic relations over $\mathcal{R}$ by a bounded-norm solution which our SNARK natively supports. We discuss how to handle relations in $\mathcal{R}_q$ in the full version of this work. Apart from obtaining short aggregated GPV signatures, in the setting where a set of n signers are signing a common message at a time, the verification of the aggregated signatures could be preprocessed, resulting in an online verification time *sublinear* in n. As a bonus result on GPV signatures, we further show how to construct lattice-based adaptor signatures [7] based on the GPV paradigm. Combining the two results, we obtain the first aggregatable adaptor signatures from any assumption.

Open Problems. Our work paves the way for what we believe to be an exciting line of research. As we initiate the study of inhomogenous variants of the k-SIS assumptions, we ask whether better (possibly quantum) algorithms can be found for solving this problem that exploit the additional algebraic structure. We also presume that for further families of rational functions the k-R-ISIS assumption can be shown to be as hard as standard hard lattice problems. Another compelling question is to study new cryptographic applications of the k-R-ISIS family. We expect that such an abstraction will be useful in transferring techniques from pairing-based cryptography into the lattice world.

1.3 Technical Overview

We give a concise overview of the process of obtaining our lattice-based SNARK.
From Vector Commitments to SNARKs. In this work, we are interested in VCs supporting openings to constant-degree-d w-variate t-output polynomial maps with bounded coefficients. The standard properties of interest for VCs are:

Compactness. Commitments and opening proofs are of size $\mathsf{poly}(\lambda, \log w, \log t)$.

Binding. It is infeasible to produce a commitment c and proofs for polynomials maps, such that the system of equations induced by them is not satisfiable.[4]

In addition, we require the following stronger notion of binding.

Extractability. To produce a commitment c and a proof that the image of a polynomial map f at the committed vector is $\mathbf{y}$, one must know a preimage $\mathbf{x}$ such that c is a commitment of $\mathbf{x}$ and $f(\mathbf{x}) = \mathbf{y}$.

It is well known that one can construct SNARKs from VCs supporting linear openings in the RO model [49]. However, in this work we take a different route and adopt a more structured approach to construct SNARKs. Specifically, recall that the satisfiability of systems of degree-d polynomials is NP-complete for any constant $d \geq 2$. As such, a SNARK can be trivially constructed from a compact and extractable VC for degree-d polynomials: The prover simply commits to the root of the system $(f, \mathbf{y})$ and immediately produces an opening proof for $(f, \mathbf{y})$. As a concrete example, a popular NP-complete language supported by existing SNARKs is rank-1 constraint satisfiability (R1CS). An R1CS instance consists of three matrices $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ over a field or in general a ring. The instance

[4] This generalises position binding.

is satisfied by a vector $\mathbf{x}$ if $(\mathbf{A} \cdot (1, \mathbf{x})) \circ (\mathbf{B} \cdot (1, \mathbf{x})) = (\mathbf{C} \cdot (1, \mathbf{x}))$, where $\circ$ denotes the Hardamard product. It is easy to see that an R1CS instance is a special case of an instance $(f, \mathbf{y})$ of degree-2 polynomial satisfiability where $f(\mathbf{X}) := (\mathbf{A} \cdot (1, \mathbf{X})) \circ (\mathbf{B} \cdot (1, \mathbf{X})) - (\mathbf{C} \cdot (1, \mathbf{X}))$ and $\mathbf{y} = \mathbf{0}$. For a full description of our SNARK we refer the reader to the full version of the paper.

Throughout the rest of this overview, we therefore focus on constructing lattice-based VCs supporting degree-d openings. Since known constructions are restricted to positional openings, we turn our attention to pairing-based schemes (which support linear openings) and develop a new strategy to translate them into lattice-based VCs and simultaneously to extend the degree to $d > 1$.

General Translation Strategy. Our strategy for constructing a lattice-based VC is a novel translation technique that lets us port techniques from the pairing-land to the lattice-land. We describe a general translation strategy for translating not only VC but also potentially other pairing-based constructions to the lattice setting. For the group setting, we adopt the implicit notation for bilinear groups $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_t$ of prime order q, i.e. the vector of elements in $\mathbb{G}_i$ with (entry-wise) discrete logarithm $\mathbf{x} \in \mathbb{Z}_q$ base an arbitrary fixed generator of $\mathbb{G}_i$ is denoted by $[\mathbf{x}]_i$, with group operations written additively, and the pairing product between $[\mathbf{x}]_1$ and $[\mathbf{y}]_2$ is written as $\langle [\mathbf{x}]_1, [\mathbf{y}]_2 \rangle$. For the lattice setting, we let $\mathcal{R}$ be a cyclotomic ring, $q \in \mathbb{N}$ be a large enough rational prime such that random elements in $\mathcal{R}_q := \mathcal{R}/q\mathcal{R}$ are invertible with non-negligible probability.

Consider a pairing-based construction where the elements $\{[1]_1, [g(\mathbf{v})]_t\}_{g \in \mathcal{G}}$ are publicly available to all parties, where $\mathcal{G}$ is a set of linearly-independent rational functions and $\mathbf{v}$ is a vector of secret exponents. An authority, knowing the secret exponents $\mathbf{v}$, is responsible for giving out secret elements $\{[g(\mathbf{v})]_2\}_{g \in \mathcal{G}}$ to user A. In turn, user A can compute $[u]_2 := \sum_{g \in \mathcal{G}} c_g \cdot [g(\mathbf{v})]_2$ and present it to user B, who can then check the correctness of $[u]_2$ by checking

$$\langle [1]_1, [u]_2 \rangle \stackrel{?}{=} \sum_{g \in \mathcal{G}} c_g \cdot [g(\mathbf{v})]_t.$$

Note that in this check one side of the pairing (i.e. $[1]_1$) is public, while the other side (i.e. $[u]_2$) is computed from secrets delegated by the authority to user A. This property will be crucial for our translation technique to apply.

The above structure can be seen in many pairing-based constructions. For example, the secret vector $\mathbf{v}$ could be a trapdoor, a master secret key of an identity-based encryption scheme, or a signing key; the delegated secrets $\{[g(\mathbf{v})]_2\}_{g \in \mathcal{G}}$ could be hints given alongside the public parameters of a VC, an identity-based secret key, or a signature; and the pairing-product check could be for opening proof verification, decryption, or signature verification.

Our strategy of translating the above to a lattice-based construction is as follows. First, the public elements $\{[1]_1, [g(\mathbf{v})]_t\}_{g \in \mathcal{G}}$ over $\mathbb{G}_1$ and $\mathbb{G}_t$ are translated to the public vector and elements $\{\mathbf{a},\ g(\mathbf{v})\}_{g \in \mathcal{G}}$, where $\mathbf{a}$ and $\mathbf{v}$ are random vectors over $\mathcal{R}_q$ and $\mathcal{R}_q^\times$ respectively. Since $\{g(\mathbf{v})\}_{g \in \mathcal{G}}$ does not necessarily hide $\mathbf{v}$ in the lattice setting (e.g. when $\mathcal{G}$ consists of many linear functions), the authority might as well publicly hand out the vectors $\{\mathbf{a}, \mathbf{v}\}$ directly. Next, the secret elements $\{[g(\mathbf{v})]_2\}_{g \in \mathcal{G}}$ are translated to the *short* secret vectors $\{\mathbf{u}_g\}_{g \in \mathcal{G}}$ satisfying

$\langle \mathbf{a}, \mathbf{u}_g \rangle = g(\mathbf{v}) \bmod q$. These short preimages can be sampled given a trapdoor of $\mathbf{a}$, which the authority should have generated alongside $\mathbf{a}$. Given $\{\mathbf{u}_g\}_{g\in\mathcal{G}}$, user A can similarly compute $\mathbf{u} := \sum_{g\in\mathcal{G}} c_g \cdot \mathbf{u}_g$, although the coefficients c_g are now required to be short. The pairing-product check is then translated to checking

$$\langle \mathbf{a}, \mathbf{u} \rangle \stackrel{?}{\equiv} \sum_{g\in\mathcal{G}} c_g \cdot g(\mathbf{v}) \bmod q \qquad \text{and} \qquad \mathbf{u} \text{ is short.}$$

The same strategy can also be used to translate (conjectured-)hard computational problems over bilinear groups to the lattice setting to obtain also seemingly-hard problems. For example, consider a variant of the ℓ-Diffie-Hellman Exponent problem, which asks to find $[v^\ell]_2$ given $([1]_1, [1]_2, [v]_2, \ldots, [v^{\ell-1}]_2)$. A natural lattice-counterpart of the problem is to find a short preimage $\mathbf{u}_\ell$ satisfying $\langle \mathbf{a}, \mathbf{u}_\ell \rangle \equiv v^\ell \bmod q$ given short preimages $(\mathbf{u}_i)_{i\in\mathbb{Z}_\ell}$ each satisfying $\langle \mathbf{a}, \mathbf{u}_i \rangle = v^i \bmod q$.

We remark that a direct translation of pairing-based constructions does not necessarily yield the most efficient lattice-based scheme. For this reason, it will be useful to generalise pairing-based constructions into a family parameterised by the function class $\mathcal{G}$. We will then have the freedom to pick $\mathcal{G}$ to optimise the efficiency of translated lattice-based scheme.

Translating Vector Commitments. We next demonstrate how the above translation strategy can be applied to translate pairing-based VCs, using the following pairing-based VC with openings to linear forms $f : \mathbb{Z}_q^w \to \mathbb{Z}_q$ adapted from [24,49,52] as an example.

- Public parameters: $\Big([1]_1, [1]_2, ([v_i]_1)_{i\in\mathbb{Z}_w}, ([\bar{v}_j]_2)_{j\in\mathbb{Z}_w}, ([v_i \cdot \bar{v}_j]_2)_{i,j\in\mathbb{Z}_w : i\neq j}, [\bar{v}]_t\Big)$ where $\bar{v} = \prod_{k\in\mathbb{Z}_w} v_k$ and $\bar{v}_j = \bar{v}/v_j$.
- Committing $\mathbf{x} \in \mathbb{Z}_q$: $[c]_1 := \sum_{i\in\mathbb{Z}_w} x_i \cdot [v_i]_1 = \langle [\mathbf{v}]_1, \mathbf{x} \rangle$
- Opening f : $[u]_2 := \sum_{i,j\in\mathbb{Z}_w : i\neq j} f_j \cdot x_i \cdot [v_i \cdot \bar{v}_j]_2$
- Verifying (f, y): $\langle [1]_1, [u]_2 \rangle \stackrel{?}{=} \Big\langle [c]_1, \sum_{j\in\mathbb{Z}_w} f_j \cdot [\bar{v}_j]_2 \Big\rangle - y \cdot [\bar{v}]_t$

The weak binding property of the scheme, i.e. the infeasibility of opening a commitment c to both (f, y) and (f, y') with $y \neq y'$, relies on the hardness of computing $[\bar{v}]_2$, whose exponent corresponds to evaluating the "target monomial" $\prod_{k\in\mathbb{Z}_w} X_k$ at $\mathbf{v}$. Notice that the target monomial is set up in such a way that $[\bar{v}]_t = [v_i]_1 \cdot [\bar{v}_i]_2$ holds for all $i \in \mathbb{Z}_w$, where $[\bar{v}_i]_2$ can be viewed as a "complement" of $[v_i]_1$. Consequently, the value $y = \langle \mathbf{f}, \mathbf{x} \rangle$ appears as the coefficient of $[\bar{v}]_t$ in the inner product $\Big\langle \sum_{i\in\mathbb{Z}_w} x_i \cdot [v_i]_1, \sum_{j\in\mathbb{Z}_w} f_j \cdot [\bar{v}_j]_2 \Big\rangle$.

While the above pairing-based scheme is ready to be translated to the lattice setting using our translation strategy, to prepare for our generalised scheme for higher-degree polynomials, we divide the target and complement monomials by $\prod_{k\in\mathbb{Z}_w} X_k$. The complement of X_i becomes X_i^{-1} and the target monomial becomes the constant 1. Concretely, we divide the opening and the verification

equation by $\bar{v}$ to obtain

$$[u']_2 := \sum_{i,j \in \mathbb{Z}_w : i \neq j} f_j \cdot x_i \cdot [v_i/v_j]_2$$

$$\langle [1]_1, [u']_2 \rangle \stackrel{?}{=} \left\langle [c]_1, \sum_{j \in \mathbb{Z}_w} f_j \cdot [v_j^{-1}]_2 \right\rangle - y \cdot [1]_t.$$

Recall that in the VC construction above we relied on the hardness of computing $[\bar{v}]_2$. What we have done here might seem absurd, since the element $[1]_2$ now is given in the group setting, but finding a short pre-image of a fixed image, say 1, is seemingly hard in the lattice setting. Indeed, translating the modified scheme, we derive the following lattice-based scheme.

- Public Parameters: $\left(\mathbf{a}, \mathbf{v}, (\mathbf{u}_{i,j})_{i \neq j \in \mathbb{Z}_w}\right)$ where $\langle \mathbf{a}, \mathbf{u}_{i,j} \rangle \equiv v_i/v_j$, $\mathbf{u}_{i,j}$ are short
- Committing $\mathbf{x} \in \mathcal{R}^w$: $c := \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$
- Opening f: $\mathbf{u} := \sum_{i,j \in \mathbb{Z}_w : i \neq j} f_j \cdot x_i \cdot \mathbf{u}_{i,j}$
- Verifying (f, y): $\langle \mathbf{a}, \mathbf{u} \rangle \stackrel{?}{=} \left(\sum_{j \in \mathbb{Z}_w} f_j \cdot v_j^{-1}\right) \cdot c - y \bmod q$ and $\mathbf{u}$ is short

For correctness, we require that the committed vector $\mathbf{x}$ and the function f both have short coefficients.

The weak binding property of the translated lattice-based scheme relies on the hardness of finding a short preimage of (a small multiple of) 1 given short preimages of v_i/v_j for all $i, j \in \mathbb{Z}_w$ with $i \neq j$ – a new computational assumption obtained by translating its pairing-counterpart, which belongs to a new family of assumptions called the k-R-ISIS assumption family.

Furthermore, the computation of $\sum_{j \in \mathbb{Z}_w} f_j \cdot v_j^{-1}$ in the verification equation can be preprocessed before knowing the commitment c and the opening proof $\mathbf{u}$, such that the online verification can be performed in time sublinear in w.

Supporting Higher-Degree Polynomials. Notice that in the group setting the (modified) verification algorithm can be seen as evaluating the linear form f at $([v_0^{-1}]_2 \cdot [c]_1, \ldots, [v_{w-1}^{-1}]_2 \cdot [c]_1)$ where $[c]_1$ supposedly encodes $\mathbf{x}$. In the group setting, f has to be linear since we cannot multiply two $\mathbb{G}_1$ elements together to get an encoding of the Kronecker product $\mathbf{x} \otimes \mathbf{x}$.

In the lattice setting, however, the commitment c is a ring element and thus we can evaluate a non-linear polynomial f at $(v_0^{-1} \cdot c, \ldots, v_{w-1}^{-1} \cdot c)$. Moreover, we notice that each degree-d monomial $\mathbf{x}^{\mathbf{e}}$ is encoded in c^d as (a factor of) the coefficient of $\mathbf{v}^{\mathbf{e}}$, which has a natural complement $\mathbf{v}^{-\mathbf{e}}$ satisfying $(\mathbf{v}^{\mathbf{e}}) \cdot (\mathbf{v}^{-\mathbf{e}}) = 1$, our modified target monomial. This suggests the possibility of generalising the translated lattice-based scheme above to support openings to higher-degree polynomials. Indeed, this technique allows us to generalise the scheme to support bounded-coefficient polynomials of degrees up to a constant, whose weak binding property is now based on another member of the k-R-ISIS assumption family.

Achieving Compactness and Extractability. The VC scheme obtained above achieves succinctness, i.e. commitments and opening proofs are of size

sublinear in w (not t), and weak binding, which fall short of the compactness and extractability required to construct a SNARK. Indeed, a black-box construction of SNARK using this VC is unlikely since, so far, we are only relying on falsifiable assumptions. To resolve this problem, we propose a knowledge version of the k-R-ISIS assumptions. For concreteness, we will use the following member of the knowledge k-R-ISIS assumption family:

> *Let* $\mathbf{a}' \leftarrow\!\!\$\ \mathcal{R}_q^{\ell}$ *and* $\mathbf{v} \leftarrow\!\!\$\ \mathcal{R}_q^{w}$ *be random vectors and* $t \leftarrow\!\!\$\ \mathcal{R}_q$ *be a random element such that* $|t \cdot \mathcal{R}_q|$ *is super-polynomial in* λ *and* $|t \cdot \mathcal{R}_q|/|\mathcal{R}_q|$ *is negligible in* λ*. If there exists an efficient algorithm* $\mathcal{A}$ *which, given short vectors* $\mathbf{u}_i'$ *satisfying* $\langle \mathbf{a}', \mathbf{u}_i' \rangle = v_i \cdot t \bmod q$ *for all* $i \in \mathbb{Z}_w$*, produces* $(c, \mathbf{u}')$ *such that* $\mathbf{u}'$ *is a short vector satisfying* $\langle \mathbf{a}', \mathbf{u}' \rangle = c \cdot t \bmod q$*, then there exists an efficient extractor* $\mathcal{E}_{\mathcal{A}}$ *which extracts a short vector* $\mathbf{x} \in \mathcal{R}^w$ *such that* $\langle \mathbf{v}, \mathbf{x} \rangle = c \bmod q$.

Equipped with this k-R-ISIS of knowledge assumption, we can upgrade our VC construction to achieve extractability as follows. First, we let the public parameters to additionally include $(\mathbf{a}', (\mathbf{u}_i')_{i \in \mathbb{Z}_w}, t)$. Here t generates an ideal that is small enough for random elements in $\mathcal{R}_q$ not to be contained within it, but big enough to provide sufficient entropy. Next, we let the committer also include $\mathbf{u}' = \sum_{i \in \mathbb{Z}_w} x_i \cdot \mathbf{u}_i'$ in an opening proof. Finally, we let the verifier additionally check that $\mathbf{u}'$ is short and $\langle \mathbf{a}', \mathbf{u}' \rangle = c \cdot t \bmod q$.

To see why the modified scheme is extractable, suppose an adversary is able to produce a commitment c and a valid opening proof for (f, y). By the k-R-ISIS of knowledge assumption, we can extract a short vector $\mathbf{x} \in \mathcal{R}^w$ such that $\langle \mathbf{v}, \mathbf{x} \rangle = c \bmod q$. Now, if $f(\mathbf{x}) = y' \neq y$, we can use the extracted $\mathbf{x}$ to compute a valid opening proof for (f, y'). However, being able to produce valid opening proofs for both (f, y) and (f, y') with $y \neq y'$ violates the weak binding property. We therefore conclude that $f(\mathbf{x}) = y$.

It remains to show how we can achieve compactness. Since our lattice-based VC schemes preserve the property of the original pairing-based schemes that the verification algorithm is linearly-homomorphic in the opening proofs, a natural strategy towards compactness is to aggregate multiple opening proofs into one using a random linear combination, with coefficients generated using a random oracle. The binding property of an aggregated opening proof can be proven using a classic rewinding argument which involves inverting a Vandermode matrix defined by the randomness used for aggregation. This strategy works particularly well in the prime-order group setting since scalars are field elements and Vandermonde matrices defined by distinct field elements are always invertible. In the lattice setting, however, the coefficients used for aggregation have to be chosen from a set where the difference between any pair of elements is (almost) invertible (over $\mathcal{R}$) for an analogous argument to go through. This is a severe limitation since sets satisfying this property cannot be too large [4].

To achieve compactness in the lattice setting, we are forced to use a different strategy. Specifically, the coefficients $\mathbf{h} = (h_i)_{i \in \mathbb{Z}_t} \in \mathcal{R}$ that we use to aggregate opening proofs are given by an instance of the R-SIS problem over $\mathcal{R}_p$ (taking

smallest $\mathcal{R}$-representatives of $\mathcal{R}_p$ elements) sampled as part of the public parameters, where p is chosen such that the R-SIS assumption is believed to hold over $\mathcal{R}_p$ while p is small relative to q.

To see why extractability still holds, suppose an adversary is able to produce a commitment c and a valid opening proof for (f, y) where $f = \sum_{i \in \mathbb{Z}_t} h_i \cdot f_i$ and $y = \sum_{i \in \mathbb{Z}_t} h_i \cdot y_i$. By our previous argument, we can extract $\mathbf{x}$ satisfying $f(\mathbf{x}) = y$. Suppose it is not the case that $f_i(\mathbf{x}) = y_i$ for all $i \in \mathbb{Z}_t$, then $(f_i(\mathbf{x}) - y_i)_{i \in \mathbb{Z}_t}$ is a short vector satisfying $\sum_{i \in \mathbb{Z}_t} h_i \cdot (f_i(\mathbf{x}) - y_i) = 0$ over $\mathcal{R}$, which implies $\sum_{i \in \mathbb{Z}_t} h_i \cdot (f_i(\mathbf{x}) - y_i) = 0 \bmod p$, breaking the R-SIS assumption over $\mathcal{R}_p$.

Discussion and Generalisations. We discuss the resulting VC scheme obtained through the aforementioned series of transformations. Our VC scheme supports openings to w-variate t-output constant-degree polynomial maps with bounded coefficients. The scheme achieves compactness and extractability, where the latter is based on the standard R-SIS assumption over $\mathcal{R}_p$ and our two new assumptions: k-R-ISIS and the k-R-ISIS of knowledge assumption over $\mathcal{R}_q$, where p is short relative to q. The construction uses only algebraic operations over $\mathcal{R}$ and $\mathcal{R}_q$. Furthermore, a major part of the verification equation can be precomputed, so that the online verification time is sublinear in w and t.

Our construction and the k-R-ISIS (of knowledge) assumption families admit natural generalisations to the module setting, where the vector $\mathbf{a}$ is replaced by a matrix $\mathbf{A}$ and other components are modified accordingly. Expectedly, we show that the module versions of the k-R-ISIS assumptions are at least as hard as the ring versions for certain parameter choices.

In many applications (e.g. aggregating signatures), often only a main part (e.g. a set of signature verification keys) of the function-image tuple (f, y) is known in advance, while the remaining small part (e.g. a message signed by all parties) is known when it comes the time to perform verification. It is desirable to preprocess the main part of (f, y) offline, so that the online verification cost is only dependent on the size of the small part. In our formal construction, we capture this flexibility by considering y itself to be a polynomial map, and allowing f and y to take an (additional, for f) public input $\mathbf{z}$. This allows the maps (f, y) to be preprocessed, such that the online cost depends mostly on $\mathbf{z}$.

1.4 Application

We highlight an application of interest of our VC, and in particular of the resulting SNARK, in aggregating GPV signatures [34]. As a bonus result, we also show how to build adaptor signatures [7] based on GPV signatures while preserving aggregatability. For more comprehensive details we refer the reader to the full version of the paper.

Aggregate GPV Signatures. GPV signatures [34] are a lattice-based signature scheme paradigm of which an instantiation is a finalist in the NIST Post-Quantum Process (Falcon [65]). On a high level, a GPV signature on a message m is a short vector $\mathbf{u}$ such that $\mathbf{A} \cdot \mathbf{u} \equiv \mathbf{v} \bmod q$, where $\mathbf{A}$ is the public key, $\mathbf{v} = H(m)$ with the hash function H modelled as a random oracle in

the security analysis. The verification is simply the check of the linear relation $\mathbf{A} \cdot \mathbf{u} \equiv \mathbf{v} \bmod q$ and that $\mathbf{u}$ is short.

Our SNARK can be used to prove knowledge of GPV signatures natively given the signature verification involves algebraic operations only. For instance, to aggregate n signatures $(\mathbf{u}_i)_{i \in \mathbb{Z}_n}$ on the same message m (a scenario that arises in a PoS consensus protocol [26]), the aggregator can compute a SNARK proof of knowledge of short $(\mathbf{u}_i)_{i \in \mathbb{Z}_n}$ satisfying $\mathbf{A}_i \cdot \mathbf{u}_i = \mathbf{v} \bmod q$, where $\mathbf{A}_i$ is the public key of the i-th signer. The aggregated signature i.e. the SNARK proof, can be verified in time sublinear in the number of signers and signatures n by first preprocessing the part of the verification equation depending on $(\mathbf{A}_i)_{i \in \mathbb{Z}_n}$. In fact, this preprocessing step is one-time for the given set of signers, and the online verification after knowing m is only logarithmic in n. If the signers sign different messages, a similar SNARK but now over the different messages results in a compact proof, but with verification time linear in n (similar to the case of BLS signatures [16]). Such aggregation can result in compact blocks in a blockchain as shown for the case of BLS signatures [16], but now with post-quantum security.

Aggregate Adaptor Signatures. Adaptor signatures [7] let a user generate an encryption $\hat{\sigma}$ of a signature σ on a message m with respect to an instance Y of a hard language $\mathcal{L}$. Here $\hat{\sigma}$ is also referred to as a *pre-signature*. Given the public key, it is efficient to verify if a given pre-signature $\hat{\sigma}$ is indeed valid with respect to the instance and the message. One can *adapt* the pre-signature $\hat{\sigma}$ into a valid signature σ given the witness y for the instance Y, and given $\hat{\sigma}$ and σ one can efficiently *extract* the witness y. The primitive has found itself useful in enhancing efficiency and privacy of conditional payments in cryptocurrencies [7], and aggregation of signatures adds clear benefits to this primitive. In the following we discuss how GPV signatures can be turned into adaptor signatures, which consequently implies that they can be aggregated via our newly constructed SNARK.

We consider the lattice trapdoor from [61] for our GPV signatures, and view the GPV signatures as follows. The public parameters are given by a uniformly random matrix $\mathbf{A}$, the signing key is $\mathsf{sk} := \mathbf{X}$, where $\mathbf{X}$ is a short norm matrix such that the public key, $\mathbf{Y} := \mathbf{A} \cdot \mathbf{X}$, is distributed statistically close to random. The signature is simply $(\mathbf{z}, \mathbf{c})$ such that during verification we have $[\mathbf{A}|\mathbf{G} + \mathbf{Y}] \cdot [\mathbf{z}|\mathbf{c}]^\intercal = H(m)$ and $\|(\mathbf{c}, \mathbf{z})\|$ is small as stipulated by GPV signatures. Here $\mathbf{G}$ is the gadget matrix. We choose the hard language

$$\mathcal{L} := \{(\mathbf{A}, \mathbf{v}') : \exists \ \mathbf{u}' \ s.t. \ \mathbf{A} \cdot \mathbf{u}' = \mathbf{v}' \wedge \|\mathbf{u}'\| \leq \beta^*\},$$

where $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$, $\mathbf{v}' \in \mathcal{R}_q^{\eta}$. A pre-signature $\hat{\sigma}$ is simply $(\mathbf{c}, \hat{\mathbf{z}})$ with $\mathbf{v}'$ as the hard instance, such that during pre-signature verification, it holds that $[\mathbf{A}|\mathbf{G} + \mathbf{Y}] \cdot [\hat{\mathbf{z}}|\mathbf{c}]^\intercal = H(m) - \mathbf{v}'$ and $\|(\mathbf{c}, \hat{\mathbf{z}})\|$ is small. It is easy to adapt $\hat{\sigma}$ given the witness $\mathbf{u}'$ by setting $\mathbf{z} := \hat{\mathbf{z}} + \mathbf{u}'$ and $\sigma := (\mathbf{c}, \mathbf{z})$. To extract a witness one can simply compute $\mathbf{u}' := \mathbf{z} - \mathbf{z}'$. We have that the extracted $\mathbf{u}'$ has a slightly higher norm than that was used to adapt the pre-signature. The security of our scheme only relies on the M-SIS problem and the RO model.

1.5 Related Work

Apart from applications to succinct arguments [49], VCs have found numerous applications, such as verifiable databases [24], verifiable decentralized storage [23], updatable zero-knowledge sets [55,59], keyless Proofs of Retrievability (PoR) [28,29], pseudonymous credentials [44], and cryptocurrencies with stateless transaction validation [25]. Several works have studied various extensions to VC, with updatable commitments and proofs [24], aggregatable opening proofs for different commitments [37], and incremental aggregatable proofs [23].

Libert, Ramanna, and Yung [52] showed that a VC for linear functions over $\mathbb{Z}_q$ implies a polynomial commitment for polynomials over $\mathbb{Z}_q$. The result was obtained by VC-committing to the coefficient vector of the polynomial and opening it to a linear function whose coefficients are evaluations of monomials at the evaluation point. Since our VC only allows committing to a short vector $\mathbf{x} \in \mathcal{R}^w$ and opening to a polynomial map f with short coefficients, we need to suitably tune the norm bound α of f and $\mathbf{x}$ to obtain similar applications. Concretely, by setting $\alpha \approx \delta^{d+1} \cdot \gamma_{\mathcal{R}}^d$ where $\gamma_{\mathcal{R}}$ is the ring expansion factor of $\mathcal{R}$, we obtain a polynomial commitment for degree-d multivariate polynomials with coefficients bounded by δ which supports evaluations at vectors of norm also bounded by δ. Note that only constant-degree polynomials are supported by our polynomial commitment since α depends exponentially on d.

In the same work [52], Libert, Ramanna, and Yung also showed that the polynomial commitment constructed from a VC for linear functions over $\mathbb{Z}_q$ implies an accumulator for $\mathbb{Z}_q$ elements, the construction requires committing to the polynomial $p(X) = \prod_{a \in A}(X - a)$ encoding the set A of elements to be accumulated. The polynomial commitment obtained via our VC unfortunately does not support committing to $p(X)$ since its degree is as large as $|A|$.

In a recent work [63], Peikert, Pepin, and Sharp proposed a VC for positional openings based on the standard SIS assumption. Relative to our construction outlined in Sect. 1.3, their construction can be interpreted as follows. Instead of handing out preimages $\mathbf{u}_{i,j}$ with $\langle \mathbf{a}, \mathbf{u}_{i,j} \rangle = v_j/v_i \bmod q$, they sample multiple $\mathbf{a}_i$ for $i \in \mathbb{Z}_w$ and let $\mathbf{u}_{i,j}$ satisfy $\langle \mathbf{a}_i, \mathbf{u}_{i,j} \rangle = v_j \bmod q$. To verify an opening to position i, the vector $\mathbf{a}_i$ is used. The removal of the non-linear term v_j/v_i allows proving security from the SIS assumption. On the flip side, using a different vector $\mathbf{a}_i$ to verify openings to different positions i forbids the standard technique of aggregating openings using a random linear combination. Furthermore, there seems to be no natural way of generalising their construction to support functional openings without significantly changing the VC model, e.g. introducing an authority responsible for issuing functional opening keys [63]. Even if we consider the model with an authority, the resulting VC only satisfies *weak binding* (using the terminology of our work) making it unsuitable to be transformed into a SNARG: There is in fact an explicit attack when compiling their VC (with authority) into a SNARG.[5]

[5] We stress that this does not contradict any of the claims made in [63], but rather exemplifies the difference between their approach and ours.

Prior to our work, all lattice-based SNARKs were in the designated-verifier setting. These constructions [32,43] are based on "linear-only" assumptions which are similar in spirit to the knowledge k-M-ISIS assumptions introduced in this work but with a key difference: While linear-only assumptions are with respect to specific encryption schemes, our assumptions are with respect to general rings. In terms of applications, linear-only encryption has always been used to construct designated-verifier primitives. In contrast, knowledge k-M-ISIS naturally leads to constructions of publicly verifiable primitives.

2 Preliminaries

Let $\lambda \in \mathbb{N}$ denote the security parameter. Define $\mathbb{N}_0 := \mathbb{N} \cup \{0\}$. Let $\mathcal{R}$ be a ring. We write $\mathcal{R}[\mathbf{X}]$ for the (multivariate) polynomial ring over $\mathcal{R}$ and $\mathcal{R}(\mathbf{X})$ for the ring of (multivariate) rational functions over $\mathcal{R}$ with intermediates $\mathbf{X} = (X_i : i \in \mathbb{Z}_w)$. We write $\langle \mathcal{G} \rangle$ for the ideal resp. module spanned by the elements of the set $\mathcal{G} \subset \mathcal{R}^\eta$ for $\eta \in \mathbb{N}$. When $\mathcal{G}$ is a singleton set we may suppress the $\{\cdot\}$ notation. We write $|\langle \mathcal{G} \rangle|$ for size of the ideal $\langle \mathcal{G} \rangle$ as a set.

For $m \in \mathbb{N}$, let $\zeta_m \in \mathbb{C}$ be any fixed primitive m-th root of unity. Denote by $\mathcal{K} = \mathbb{Q}(\zeta_m)$ the cyclotomic field of order $m \geq 2$ and degree $n = \varphi(m)$, and by $\mathcal{R} = \mathbb{Z}[\zeta_m]$ its ring of integers, called a cyclotomic ring for short. We have $\mathcal{R} \cong \mathbb{Z}[x]/\langle \Phi_m(x) \rangle$, where $\Phi_m(x)$ is the m-th cyclotomic polynomial. If m is a power of 2, we call $\mathcal{R}$ a power-of-2 cyclotomic ring. If m is a prime-power, we call $\mathcal{R}$ a prime-power cyclotomic ring. Let $q \in \mathbb{N}$ be prime, we write $\mathcal{R}_q := \mathcal{R}/q\mathcal{R}$ and $\mathcal{R}_q^\times$ for all invertible elements in $\mathcal{R}_q$. We have that $\mathcal{R}_q$ splits into f fields of degree $\phi(m)/f$. We write $\mathrm{vec}(r) \in \mathbb{Z}^n$ for the coefficient vector of r (with the powerful basis). For any $r \in \mathcal{R}$ there exists a matrix $\mathrm{rot}(r) \in \mathbb{Z}^{n \times n}$ s.t. $\forall s \in \mathcal{R}$ we have $\mathrm{vec}(r \cdot s) = \mathrm{rot}(r) \cdot \mathrm{vec}(s)$. For elements $x \in \mathcal{R}$ we denote the infinity norm of its coefficient vector as $\|x\| := \|\mathrm{vec}(x)\|$. If $\mathbf{x} \in \mathcal{R}^\ell$ we write $\|\mathbf{x}\|$ for the infinity norm of $\mathbf{x}$. We write $\|\cdot\|_p$ for the ℓ_p-norm, e.g. $\|\cdot\|_2$ for the Euclidean norm. We write $\mathcal{M}_\mathcal{G}(\cdot)$ for a function that takes vectors indexed by $\mathcal{G}$ and returns a matrix where each column corresponds to one such vector. We write $\mathbf{I}_n$ for the identity matrix of dimension n over whatever ring is clear from context.

For $w \in \mathbb{N}$, $\mathbf{x} = (x_i : i \in \mathbb{Z}_w) \in \mathcal{R}^w$, and $\mathbf{e} = (e_i : i \in \mathbb{Z}_w) \in \mathbb{Z}^w$, we write $\mathbf{x}^\mathbf{e} := \prod_{i \in \mathbb{Z}_w} x_i^{e_i}$ whenever it is defined. For $\mathbf{v} = (v_i : i \in \mathbb{Z}_w) \in (\mathcal{R}_q^\times)^w$, we write $\bar{\mathbf{v}} := (v_i^{-1} : i \in \mathbb{Z}_w)$ for the entry-wise inverse of $\mathbf{v}$. A Laurent monomial $g(\mathbf{X}) \in \mathcal{R}(\mathbf{X})$ is an expression $g(\mathbf{X}) = \mathbf{X}^\mathbf{e} := \prod_{i \in \mathbb{Z}_w} X_i^{e_i}$ with exponent vector $\mathbf{e} = (e_i : i \in \mathbb{Z}_w) \in \mathbb{Z}^w$.

We may suppress arbitrary subscripts and superscripts from problem and advantage notations when those are clear from context. We write $x \leftarrow \mathcal{D}$ for sampling from the distribution $\mathcal{D}$ and $x \leftarrow\!\!\$\ \mathcal{S}$ to sample an element from the finite space $\mathcal{S}$ uniformly at random. We write $U(\mathcal{S})$ for the uniform distribution over $\mathcal{S}$ and $\{\mathbf{u}_\mathcal{G}\} := \{\mathbf{u}_g\}_{g \in \mathcal{G}}$.

Definition 1 (Ring Expansion Factor). *Let $\mathcal{R}$ be a ring. The expansion factor of $\mathcal{R}$, denoted by $\gamma_\mathcal{R}$, is $\gamma_\mathcal{R} := \max_{a,b \mathcal{R}} \frac{\|a \cdot b\|}{\|a\| \cdot \|b\|}$.*

Proposition 1 ([4]). *If $\mathcal{R} = \mathbb{Z}[\zeta_m]$ is a prime-power cyclotomic ring, then $\gamma_{\mathcal{R}} \leq 2n$. If $\mathcal{R} = \mathbb{Z}[\zeta_m]$ is a power-of-2 cyclotomic ring, then $\gamma_{\mathcal{R}} \leq n$.*

Proposition 2. *Let $q = \omega((w \cdot f)^{f/\phi(m)})$ be a rational prime such that $\mathcal{R}_q$ splits into f fields each of size $q^{\varphi(m)/f}$. For $\mathbf{v} \leftarrow\!\$\ \mathcal{R}_q^w$, we have $\mathbf{v} \in (\mathcal{R}_q^\times)^w$ with non-negligible probability.*

Proof. The probability that $\mathbf{v} \in (\mathcal{R}_q^\times)^w$ is $(1 - 1/q^{\varphi(m)/f})^{w \cdot f} \geq 1 - (w \cdot f)/q^{\varphi(m)/f}$ which is non-negligible. □

For the rest of this work, we implicitly assume q is large enough so that a uniformly random $\mathbf{v} \leftarrow\!\$\ \mathcal{R}_q^w$ satisfies $\mathbf{v} \in (\mathcal{R}_q^\times)^w$ with non-negligible probability.

2.1 Lattices

We write $\Lambda(\mathbf{B})$ for the Euclidean lattice generated by the columns of $\mathbf{B} \in \mathbb{Z}^{n \times d} = [\mathbf{b}_0 | \ldots \mathbf{b}_{d-1}]$, i.e. $\{z_i \cdot \mathbf{b}_i \mid z_i \in \mathbb{Z}\}$. When $\mathbf{B}$ has full rank we call it a basis and when $n = d$ we say that $\Lambda(\mathbf{B})$ has full rank. The determinant of a full rank lattice is the absolute value of the determinant of any of its bases. Minkowski's theorem implies that there is a vector $\mathbf{x} \in \Lambda \subset \mathbb{R}^d$ of (infinity) norm $\|\mathbf{x}\| \leq \det(\Lambda)^{1/d}$ when Λ has full rank. The Gaussian heuristic predicts that a random full-rank lattice Λ contains a shortest vector of (Euclidean) norm $\approx \sqrt{\frac{d}{2\pi e}} \cdot \det(\Lambda)^{1/d}$.

For any $\mathbf{c} \in \mathbb{R}^n$ and any real $\sigma > 0$, the (spherical) Gaussian function with standard deviation parameter σ and centre $\mathbf{c}$ is:

$$\forall \mathbf{x} \in \mathbb{R}^n, \rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp\left(-\frac{\pi \cdot \|\mathbf{x} - \mathbf{c}\|_2^2}{\sigma^2}\right).$$

The Gaussian distribution is $\mathcal{D}_{\sigma,\mathbf{c}}(\mathbf{x}) = \rho_{\sigma,\mathbf{c}}(\mathbf{x})/\sigma^n$. The (spherical) discrete Gaussian distribution over a lattice $\Lambda \in \mathbb{R}^n$, with standard deviation parameter $\sigma > 0$ and centre $\mathbf{c}$ is:

$$\forall \mathbf{x} \in \Lambda, \mathcal{D}_{\Lambda,\sigma,\mathbf{c}} = \frac{\rho_{\sigma,\mathbf{c}}(\mathbf{x})}{\rho_{\sigma,\mathbf{c}}(\Lambda)},$$

where $\rho_{\sigma,\mathbf{c}}(\Lambda) := \sum_{\mathbf{x} \in \Lambda} \rho_{\sigma,\mathbf{c}}(\mathbf{x})$. When $\mathbf{c} = \mathbf{0}$ we omit the subscript $\mathbf{c}$. We may write $\mathcal{D}_{\mathcal{R},\sigma}$ where we interpret $\mathcal{R}$ to be the lattice spanned by $\mathcal{R}$.

The dual of a lattice Λ is defined by $\Lambda^* = \{\mathbf{y} \in \mathbb{R}^n : \mathbf{y}^\mathsf{T} \cdot \Lambda \subseteq \mathbb{Z}\}$. The smoothing parameter of an n-dimensional lattice Λ with respect to $\epsilon > 0$, denoted $\eta_\epsilon(\Lambda)$, is the smallest $\sigma > 0$, such that $\rho_{1/\sigma}(\Lambda^* \setminus \{0\}) \leq \epsilon$.

Lattice reduction with parameter κ returns a vector of Euclidean norm $\approx \delta^{d-1} \cdot \det(\Lambda)^{1/d}$ where δ is the root Hermite factor δ and a function of κ.[6] A root Hermite factor $\delta \approx \left(\frac{\kappa}{2\pi e}\right)^{1/(2\kappa)}$ can be achieved in time $2^{0.292\,\kappa + o(\kappa)}$ classically using the BKZ algorithm [67] with block size κ and sieving as the SVP oracle [8] (quantum algorithms do not promise a sufficiently substantial speed-up [3,48]). Concretely, for $\lambda = 128$ we require $\kappa \geq 484$ and thus $\delta \leq 1.0034$.

[6] The literature routinely simplifies the first expression to $\approx \delta^d \cdot \det(\Lambda)^{1/d}$.

2.2 Sampling Algorithms

The following relies on analogues of the Leftover Hash Lemma over rings attesting that given $\mathbf{a}_i \leftarrow\!\$\ U(\mathcal{R}_q^\eta)$ and $r_i \leftarrow\!\$\ \mathcal{D}$ where $\mathcal{D}$ is a small uniform [60,69] or discrete Gaussian distribution [57,68], we have that $\left(\mathbf{a}_0, \ldots, \mathbf{a}_{\ell-1}, \sum_{0 \le i < \ell} \mathbf{a}_i \cdot r_i\right)$ is close to uniform. In what follows, we will write $\mathsf{lhl}(\mathcal{R}, \eta, q, \mathcal{D})$ for an algorithm that outputs a minimal $\ell \in \mathbb{N}$ ensuring that the resulting distribution is within $\mathsf{negl}(\lambda)$ to uniform. We may also write $\mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ for some $\mathcal{D}$ outputting elements bounded by β (with overwhelming probability). In many cases the reader may think $\ell \in O(\eta \log_\beta(q))$. Let $(\mathsf{TrapGen}, \mathsf{SampD}, \mathsf{SampPre})$ be PPT algorithms with the following syntax and properties [31,34,61]:

- $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^\eta, 1^\ell, q, \mathcal{R}, \beta)$ takes dimensions $\eta, \ell \in \mathbb{N}$, a modulus $q \in \mathbb{N}$, a ring $\mathcal{R}$, and a norm bound $\beta \in \mathbb{R}$. It generates a matrix $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$ and a trapdoor td. For any $n \in \mathsf{poly}(\lambda)$ and $\ell \ge \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$, the distribution of $\mathbf{A}$ is within $\mathsf{negl}(\lambda)$ statistical distance of $U(\mathcal{R}_q^{\eta \times \ell})$.
- $\mathbf{u} \leftarrow \mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, \beta')$ with $\ell \ge \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ outputs an element in $\mathbf{u} \in \mathcal{R}^\ell$ with norm bound $\beta' \ge \beta$. We have that $\mathbf{v} := \mathbf{A} \cdot \mathbf{u} \bmod q$ is within $\mathsf{negl}(\lambda)$ statistical distance to $U(\mathcal{R}_q^\eta)$.
- $\mathbf{u} \leftarrow \mathsf{SampPre}(\mathsf{td}, \mathbf{v}, \beta')$ with $\ell \ge \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ takes a trapdoor td, a vector $\mathbf{v} \in \mathcal{R}_q^\eta$, and a norm bound $\beta' \ge \beta$. It samples $\mathbf{u} \in \mathcal{R}^\ell$ satisfying $\mathbf{A} \cdot \mathbf{u} \equiv \mathbf{v} \bmod q$ and $\|\mathbf{u}\| \le \beta'$. Furthermore, $\mathbf{u}$ is within $\mathsf{negl}(\lambda)$ statistical distance to $\mathbf{u} \leftarrow \mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, \beta')$ conditioned on $\mathbf{v} \equiv \mathbf{A} \cdot \mathbf{u} \bmod q$. The syntax can be extended in the natural way for $\mathsf{SampPre}$ to take a matrix $\mathbf{V}$ as input, in which case $\mathsf{SampPre}$ is run on each column of $\mathbf{V}$ and the output vectors are concatenated column-wise to form a matrix.

For all algorithms we may replace β by $\mathcal{D}$ where it is understood that $\mathcal{D}$ outputs samples bounded by β (with overwhelming probability).

2.3 Hard Problems

The Short Integer Solution problem was introduced in the seminal work of Ajtai [2]. It asks to find a short element (of Euclidean norm β_2) in the kernel of a random matrix mod q. An inhomogeneous version, asking to find a short solution to a linear algebra problem mod q was formalised later [60].

For both problems, it was shown [34] that solving the problem for $q \ge \beta_2 \cdot \omega(\sqrt{n \cdot \log n})$ implies solving certain presumed hard lattice problems (finding a short basis) to within approximation factor $\beta_2 \cdot \tilde{\mathcal{O}}(\sqrt{n})$. Thus, since $\beta_2 \ge \beta_\infty$, an appropriate choice of parameters is $n = \mathsf{poly}(\lambda)$, $q \ge \beta_\infty \cdot n \cdot \log n$ and $\ell \ge 2\, n \log_{\beta_\infty} q$. An algorithm solving ISIS can be used to solve SIS (by making one of the columns of $\mathbf{A}$ the target) and solving ISIS twice allows to solve SIS by considering the difference of these solutions. Ring variants were introduced in [56,60,64]; module variants in [51].

Definition 2 (M-SIS, adapted from [51]). *Let $\mathcal{R}, \eta, q, \ell, \beta$ depend on λ. The Module-SIS (or M-SIS) problem, denoted M-$\mathsf{SIS}_{\mathcal{R}_q, \eta, \ell, \beta^*}$, is: Given a uniform $\mathbf{A} \leftarrow\!\$\ \mathcal{R}_q^{\eta \times \ell}$, $\mathbf{t} \equiv 0 \bmod q$ find some $\mathbf{u} \ne \mathbf{0} \in \mathcal{R}^\ell$ such that $\|\mathbf{u}\|_\infty \le \beta^*$ and $\mathbf{A} \cdot$*

$\mathbf{u} \equiv \mathbf{t} \bmod q$. *We write* $\mathsf{Adv}^{\text{m-sis}}_{\mathcal{R}_q,\eta,\ell,\beta^*}(\lambda)$ *for the advantage of any algorithm* $\mathcal{A}$ *in solving* M-$\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,\beta^*}$. *We assume* $\mathsf{Adv}^{\text{m-sis}}_{\mathcal{R}_q,\eta,\ell,\beta^*,\mathcal{A}}(\lambda) \leq \mathsf{negl}(\lambda)$ *for appropriately chosen* $\mathcal{R}_q, \eta, \ell, \beta^*$ *and PPT* $\mathcal{A}$. *When* $\mathbf{t} \neq 0$ *we speak of the Module-ISIS or M-ISIS problem, denoted* M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,\beta^*}$. *When* $\eta = 1$ *we speak of Ring-(I)SIS or R-(I)SIS, denoted* R-$\mathsf{SIS}_{\mathcal{R}_q,\ell,\beta^*}$ *or* R-$\mathsf{ISIS}_{\mathcal{R}_q,\ell,\beta^*}$.

In [51] it was shown that solving Module-SIS is as hard as finding a short basis in modules. In [56,64] it was shown that solving Ring-SIS is as hard as find a short vector in any ideal in $\mathcal{R}$. A similar result was established for Ring-ISIS [60]. From a cryptanalytic perspective, no known algorithm solves Ring/Module-(I)SIS significantly faster than those solving (I)SIS. Our assumption is a generalisation and adaptation to more general rings of the k-SIS assumption.

Definition 3 (*k-M-SIS, generalised from* [17,54])**.** *For any integer* $k \geq 0$, *an instance of the* k-M-$\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,\beta,\beta^*}$ *problem is a matrix* $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$ *and a set of* k *vectors* $\mathbf{u}_0, \ldots \mathbf{u}_{k-1}$ *s.t.* $\mathbf{A} \cdot \mathbf{u}_i \equiv \mathbf{0} \bmod q$. *A solution to the problem is a nonzero vector* $\mathbf{u} \in \mathcal{R}^\ell$ *such that*

$$\|\mathbf{u}\|_\infty \leq \beta, \quad \mathbf{A} \cdot \mathbf{u} \equiv \mathbf{0}, \quad \textit{and} \quad \mathbf{u} \notin \mathcal{K}\text{-span}(\{\mathbf{u}_i\}_{0 \leq i < k}).$$

If $\mathcal{B}$ *is an algorithms that takes as input a matrix* $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$ *and vectors* $\mathbf{u}_i \in \mathcal{R}^\ell$ *for* $0 \leq i < k$, *we define* $\mathsf{Adv}^{\text{k-m-sis}}_{\mathcal{R}_q,\eta,\ell,\beta,\beta^*,\mathcal{B}}(\lambda)$ *to be the probability that* $\mathcal{B}$ *outputs a solution to the* k-M-$\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,\beta,\beta^*}$ *problem instance* $\mathbf{A}, \mathbf{u}_0, \ldots, \mathbf{u}_{k-1}$ *over uniformly random* $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$ *and* $\mathbf{u}_i$ *drawn from* $\mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, \beta)$ *conditioned on* $\mathbf{A} \cdot \mathbf{u}_i \equiv \mathbf{0} \bmod q$.

In [17,54] it is shown that if SIS is hard for $\mathbb{Z}_q^{n \times (\ell - k)}$ and norm bound β then k-M-$\mathsf{SIS}_{\mathbb{Z}_q,n,\ell,\beta',\beta''}$ is hard for any $k < \ell$, and certain $\beta', \beta'' \in \mathsf{poly}(\beta)$. Looking ahead, here we are interested in k-R-$\mathsf{SIS}_{\mathcal{R}_q,\ell,\beta,\beta^*} := k$-$M$-$\mathsf{SIS}_{\mathcal{R}_q,1,\ell,\beta,\beta^*}$.

3 The *k*-*M*-ISIS Assumption

We first introduce a family of assumptions over modules – k-M-ISIS – which we then specialise to rings to obtain k-R-ISIS mentioned above.

We note that the most immediate candidate notion for k-ISIS, i.e. generalising k-SIS, is to simply hand out short preimages of random images and then ask the adversary to solve ISIS. This notion is trivially equivalent to ISIS since short preimages of random images can be efficiently sampled by sampling short $\mathbf{u} \in \mathbb{Z}^\ell$ and computing $\mathbf{t} := \mathbf{A} \cdot \mathbf{u}$. The same reasoning can be lifted to $\mathcal{R}$. On the other hand, k-SIS is trivially insecure when $k \geq \ell$ in the intuitive sence since then $\{\mathbf{u}_i\}$ constitutes a trapdoor for $\mathbf{A}$ when the $\mathbf{u}_i$ are linearly independent [34]. Formally, the problem as stated is impossible to solve since all vectors will be in $\mathbb{Q}$-span$(\{\mathbf{u}_i\}_{0 \leq i < k})$, i.e. there are no valid solutions.

Our variants are neither trivially equivalent to M-ISIS nor immediately broken when $k > \ell$ by imposing on the images an algebraic structure which is inde-

pendent of the challenge matrix $\mathbf{A}$. Before stating our family of assumptions, we define a notion of admissibility to formally rule out trivial wins.

Definition 4 (k-M-ISIS-Admissible). *Let $g(\mathbf{X}) \in \mathcal{R}(\mathbf{X})$ be a Laurent monomial, i.e. $g(\mathbf{X}) = \mathbf{X}^{\mathbf{e}} := \prod_{i \in \mathbb{Z}_w} X_i^{e_i}$ for some exponent vector $\mathbf{e} = (e_i : i \in \mathbb{Z}_w) \in \mathbb{Z}^w$. Let $\mathcal{G} \subset \mathcal{R}(\mathbf{X})$ be a set of Laurent monomials with $k := |\mathcal{G}|$ and let $\boldsymbol{\mathcal{G}}$ be a vector of those monomials. Let $g^* \in \mathcal{R}(\mathbf{X})$ be a target Laurent monomial. We call a family $\mathcal{G}$ k-M-ISIS-admissible if (i) all $g \in \mathcal{G}$ have constant degree, i.e. $\|\mathbf{e}\|_1 \in O(1)$; (ii) all $g \in \mathcal{G}$ are distinct, i.e. $\mathcal{G}$ is not a multiset; and (iii) $0 \notin \mathcal{G}$. We call a family $(\mathcal{G}, g^*)$ k-M-ISIS-admissible if $\mathcal{G}$ is k-M-ISIS-admissible, g^* has constant degree, and $g^* \notin \mathcal{G}$.*

Remark 1. Condition (i) rules out monomials that depend on the ring $\mathcal{R}$, such as $X^{\phi(m)}$. Condition (ii) rules out that trivial linear combinations of known preimages produce a preimage for the target. Condition (iii) rules out trivially producing multiple preimages of the same image. On the other hand, we do not target full generality here but restrict ourselves to a slight generalisation of what we require in this work. It is plausible that we can replace Laurent monomials by Laurent "terms", i.e. with coefficients $\neq 1$ in $\mathcal{R}_q$, or rational functions.

Definition 5 (k-M-ISIS Assumptions). *Let $\ell, \eta \in \mathbb{N}$. Let q be a rational prime, $\mathcal{R}$ the m-th cyclotomic ring, and $\mathcal{R}_q := \mathcal{R}/q\mathcal{R}$. Let $\mathcal{T} \subset \mathcal{R}_q^\eta$ be such that, for any $\mathbf{t} = (t_i)_{i \in \mathbb{Z}_\eta} \in \mathcal{T}$, $\langle \{t_i\} \rangle = \mathcal{R}_q$. Let $\mathcal{G} \subset \mathcal{R}(\mathbf{X})$ be a set of w-variate Laurent monomial. Let $g^* \in \mathcal{R}(\mathbf{X})$ be a target Laurent monomial. Let $(\mathcal{G}, g^*)$ be k-M-ISIS-admissible. Let $\bar{\mathcal{G}} := \mathcal{G} \cup \{g^*\}$. Let $\beta \geq 1$ and $\beta^* \geq 1$ be reals. For $\eta, \ell \in \mathbb{N}$, $g \in \bar{\mathcal{G}}$, $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$, $\mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}$, $\mathbf{t} \in \mathcal{T}$, and $\mathbf{v} \in (\mathcal{R}_q^\times)^w$, let $\mathcal{D}_{g,\mathbf{A},\mathbf{t},\mathbf{v}}$ be a distribution over*

$$\{\mathbf{u}_g \in \mathcal{R}^\ell : \mathbf{A} \cdot \mathbf{u}_g \equiv g(\mathbf{v}) \cdot \mathbf{t} \bmod q, \|\mathbf{u}_g\| \leq \beta\}.$$

Let $\mathcal{D} := \{\mathcal{D}_{g,\mathbf{A},\mathbf{t},\mathbf{v}} : \eta, \ell \in \mathbb{N}, g \in \bar{\mathcal{G}}, \mathbf{A} \in \mathcal{R}_q^{\eta \times \ell}, \mathbf{v} \in (\mathcal{R}_q^\times)^w\}$ be the family of these distributions. Write $\mathsf{pp} := (\mathcal{R}_q, \eta, \ell, w, \mathcal{G}, g^*, \mathcal{D}, \mathcal{T}, \beta, \beta^*)$. The k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ assumption states that for any PPT adversary $\mathcal{A}$ we have $\mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) \leq \mathsf{negl}(\lambda)$, where

$$\mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) := \Pr\left[\begin{array}{l|l} \mathbf{A} \cdot \mathbf{u}_{g^*} \equiv s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} & \mathbf{A} \leftarrow\!\$\, \mathcal{R}_q^{\eta \times \ell} \bmod q \\ \wedge\ 0 < \|s^*\| \leq \beta^* & \mathbf{t} \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w \\ \wedge\ \|\mathbf{u}_{g^*}\| \leq \beta^* & \mathbf{u}_g \leftarrow\!\$\, \mathcal{D}_{g,\mathbf{A},\mathbf{t},\mathbf{v}},\ \forall\ g \in \mathcal{G} \\ \wedge\ (g^*, \mathbf{u}_{g^*}) \neq (0, \mathbf{0}) & (s^*, \mathbf{u}_{g^*}) \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{t}, \{\mathbf{u}_{\mathcal{G}}\}, \mathbf{v}) \end{array}\right].$$

Remark 2. Since for any $\mathbf{t}' \in \mathcal{T}$ there exist matrices $\mathbf{X}, \mathbf{Y}$ s.t. $\mathbf{X} \cdot \mathbf{Y} \equiv \mathbf{I}$, $\mathbf{X} \cdot \mathbf{t}' \equiv (1, 0, \ldots, 0)^\intercal \bmod q$ and $\mathbf{Y} \cdot (1, 0, \ldots, 0)^\intercal \equiv \mathbf{t}' \bmod q$, we can assume that $\mathcal{T} = \{(1, 0, \ldots, 0)^\intercal\}$ without loss of generality.

Definition 6 (k-R-ISIS). *When $\eta = 1$ we may write*

$$k\text{-}R\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\ell,w,\mathcal{G},g^*,\mathcal{D},\mathcal{T},\beta,\beta^*} := k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,1,\ell,w,\mathcal{G},g^*,\mathcal{D},\mathcal{T},\beta,\beta^*}.$$

Remark 3. Analogous to the ℓ-Diffie-Hellman exponent assumption, an example of $(w, \mathcal{G}, g^*)$ is $w = 1$, $\mathcal{G} = \{1, X, \ldots, X^{\ell}, X^{\ell+2}, \ldots, X^{2\ell}\}$, and $g^*(X) = X^{\ell+1}$ for some $\ell \in \mathbb{N}$.

As written above we have a separate assumption for each family of $(\mathcal{G}, g^*)$ which are application dependent. As we will show below, there are $(\mathcal{G}, g^*)$ that are as hard as M-ISIS and our discussion of admissibility indicates that some $(\mathcal{G}, g^*)$ are trivially insecure. However, to encourage analysis and to avoid "bodacious assumptions" [46] we make the following, strong, meta assumption.

Definition 7 (k-M-ISIS Meta Assumption). *For any k-M-ISIS-admissible $(\mathcal{G}, g^*)$, k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ is hard.*

3.1 Knowledge Variants

We next propose a "knowledge" version of the k-M-ISIS assumption. It captures the intuition that if the images are restricted to scalar multiples of $\mathbf{t}$ then the only way to produce preimages of them under $\mathbf{A}$ is to perform a linear combination of the given preimages under $\mathbf{A}$ with small coefficients.

Definition 8 (Knowledge k-M-ISIS Assumption). *Adopt the notation from Definition 5, but let $\mathsf{pp} := (\mathcal{R}_q, \eta, \ell, w, \mathcal{G}, \mathcal{D}, \mathcal{T}, \alpha, \beta, \beta^*)$ where $\alpha \geq 1$ is real and $\eta > 1$. The knowledge k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ assumption states that for any PPT adversary $\mathcal{A}$ there exists a PPT extractor $\mathcal{E}_{\mathcal{A}}$ such that $\mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) \leq \mathsf{negl}(\lambda)$, where*

$$\mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) := \mathsf{Pr}\left[\begin{array}{l|l} \mathbf{A} \cdot \mathbf{u} \equiv c \cdot \mathbf{t} \bmod q & \mathbf{A} \hookleftarrow\$\ \mathcal{R}_q^{\eta \times \ell} \\ \wedge\ \|\mathbf{u}\| \leq \beta^* & \mathbf{t} \hookleftarrow\$\ \mathcal{T};\ \mathbf{v} \hookleftarrow\$\ (\mathcal{R}_q^{\times})^w \\ \wedge\ \neg \left(\begin{array}{c} c \equiv \sum_{g \in \mathcal{G}} x_g \cdot g(\mathbf{v}) \\ \wedge\ \|(x_g)_{g \in \mathcal{G}}\| \leq \alpha \end{array} \right) & \mathbf{u}_g \hookleftarrow\$\ \mathcal{D}_{g,\mathbf{A},\mathbf{t},\mathbf{v}},\ \forall\ g \in \mathcal{G} \\ & \left((c, \mathbf{u}), (x_g)_{g \in \mathcal{G}}\right) \\ & \quad \leftarrow (\mathcal{A} \| \mathcal{E}_{\mathcal{A}})\,(\mathbf{A}, \mathbf{t}, \{\mathbf{u}_{\mathcal{G}}\}, \mathbf{v}) \end{array}\right]$$

where the notation $(\mathcal{A} \| \mathcal{E}_{\mathcal{A}})$ means that $\mathcal{A}$ and $\mathcal{E}_{\mathcal{A}}$ are run on the same input including the randomness, and $(c, \mathbf{u})$ and $(x_g)_{g \in \mathcal{G}}$ are the outputs of $\mathcal{A}$ and $\mathcal{E}_{\mathcal{A}}$ respectively.

The knowledge k-M-ISIS assumption, as stated, only makes sense for $\eta \geq 2$, i.e. not for k-R-ISIS. To see this, consider an adversary $\mathcal{A}$ which does the following: First, it samples random short $\mathbf{u}$ and checks whether $\mathbf{A} \cdot \mathbf{u}$ is in the submodule of $\mathcal{R}_q^{\eta}$ generated by $\mathbf{t}$. If not, $\mathcal{A}$ aborts. If so, it finds c such that $\mathbf{A} \cdot \mathbf{u} = c \cdot \mathbf{t} \bmod q$ and outputs $(c, \mathbf{u})$. When $\eta = 1$ and assuming without loss of generality that $\mathcal{T} = \{(1, 0, \ldots, 0)^{\mathsf{T}}\}$, we observe that $t = 1$ generates $\mathcal{R}_q$, which means $\mathcal{A}$ never aborts. Clearly, when $\mathcal{A}$ does not abort, it has no "knowledge" of how c can be expressed as a linear combination of $\{g(\mathbf{v})\}_{g \in \mathcal{G}}$. Note that when $\eta \geq 2$ the adversary $\mathcal{A}$ aborts with overwhelming probability since $\mathbf{A} \cdot \mathbf{u}$ is close to uniform over $\mathcal{R}_q^{\eta}$ but the submodule generated by $\mathbf{t}$ is only a negligible faction of $\mathcal{R}_q^{\eta}$. However, in order to be able to pun about "crises of knowledge", we also define a ring version of the knowledge assumption. In the ring setting, we consider proper ideals rather than submodules.

Definition 9 (Knowledge k-R-ISIS Assumption). *Let the parameters* pp *be as in Definition 5 except that* $\eta = 1$ *and* $\mathcal{T}$ *contains elements* $t \in \mathcal{R}_q$ *s.t.* $1/|\langle t \rangle| = \mathsf{negl}(\lambda)$ *and* $|\langle t \rangle|/|\mathcal{R}_q| = \mathsf{negl}(\lambda)$.[7] *The knowledge* k-R-$\mathsf{ISIS}_{\mathsf{pp}}$ *assumption states that for any PPT adversary* $\mathcal{A}$ *there exists a PPT extractor* $\mathcal{E}_{\mathcal{A}}$ *such that* $\mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) \leq \mathsf{negl}(\lambda)$, *where*

$$\mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) := \mathsf{Pr}\left[\begin{array}{l} \langle \mathbf{a}, \mathbf{u} \rangle \equiv c \cdot t \bmod q \\ \wedge\ \|\mathbf{u}\| \leq \beta^* \\ \wedge\ \neg \left(\begin{array}{l} c \equiv \sum_{g \in \mathcal{G}} x_g \cdot g(\mathbf{v}) \\ \wedge\ \|(x_g)_{g \in \mathcal{G}}\| \leq \alpha \end{array} \right) \end{array} \middle| \begin{array}{l} \mathbf{a} \leftarrow\!\$\ \mathcal{R}_q^{\ell} \\ t \leftarrow\!\$\ \mathcal{T};\ \mathbf{v} \leftarrow\!\$\ (\mathcal{R}_q^{\times})^w \\ \mathbf{u}_g \leftarrow\!\$\ \mathcal{D}_{g,\mathbf{a},t,\mathbf{v}},\ \forall\ g \in \mathcal{G} \\ \left((c, \mathbf{u}), (x_g)_{g \in \mathcal{G}}\right) \\ \quad \leftarrow (\mathcal{A} \| \mathcal{E}_{\mathcal{A}})(\mathbf{a}, t, \{\mathbf{u}_{\mathcal{G}}\}, \mathbf{v}) \end{array} \right].$$

Definition 10 (k-M-ISIS Meta Knowledge Assumption). *For any* k-M-ISIS-*admissible* $\mathcal{G}$, *the knowledge* k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ *assumption holds.*

We provide reductions for some parameter regimes and some preliminary cryptanalysis of our assumption in the full version of this work.

4 Compact Extractable Vector Commitments

We construct compact extractable vector commitments with openings to constant-degree multivariate polynomial maps from the knowledge k-M-ISIS assumption.

4.1 Definitions

We define a non-interactive variant of vector commitments with preprocessing.

Definition 11 (Vector Commitments (VC)). *A (preprocessing non-interactive) vector commitment (VC) scheme is parameterised by the families*

$$\mathcal{F} = \{\mathcal{F}_{s,w,t} \subseteq \{f : \mathcal{R}^s \times \mathcal{R}^w \to \mathcal{R}^t\}\}_{s,w,t \in \mathbb{N}} \text{ and}$$
$$\mathcal{Y} = \{\mathcal{Y}_{s,t} \subseteq \{y : \mathcal{R}^s \to \mathcal{R}^t\}\}_{s,t \in \mathbb{N}}$$

of functions over $\mathcal{R}$ *and an input alphabet* $\mathcal{X} \subseteq \mathcal{R}$. *The parameters* s, w, *and* t *are the dimensions of public inputs, secret inputs, and outputs of* f *respectively. The VC scheme consists of the PPT algorithms* $(\mathsf{Setup}, \mathsf{Com}, \mathsf{Open}, \mathsf{PreVerify}, \mathsf{Verify})$ *defined as follows:*

- $\mathsf{pp} \leftarrow \mathsf{Setup}(1^{\lambda}, 1^s, 1^w, 1^t)$*: The setup algorithm generates the public parameters on input the security parameter* $\lambda \in \mathbb{N}$ *and the size parameters* $s, w, t \in \mathbb{N}$.

[7] Concretely, let $\mathcal{T}$ be the set of all $\mathcal{R}_q$ elements t where half of the components of t in the Chinese remainder theorem (CRT) representation are zero and the other half are non-zero. Note that this is well-defined only when $\langle q \rangle$ is not a prime ideal in $\mathcal{R}$.

- $(c, \mathsf{aux}) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x})$*: The commitment algorithm generates a commitment* c *of a given vector* $\mathbf{x} \in \mathcal{X}^w$ *with some auxiliary opening information* aux*.*
- $\pi \leftarrow \mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$*: The opening algorithm generates a proof* π *for* $f(\mathbf{z}, \cdot)$ *for the public input* $\mathbf{z} \in \mathcal{X}^s$ *and function* $f \in \mathcal{F}_{s,w,t}$*.*
- $\mathsf{pp}_{f,y} \leftarrow \mathsf{PreVerify}(\mathsf{pp}, (f, y))$*: Given functions* $f \in \mathcal{F}_{s,w,t}$ *and* $y \in \mathcal{Y}_{s,t}$*, the verification preprocessing algorithm generates the preprocessed public parameters* $\mathsf{pp}_{f,y}$ *for verifying proofs for* (f, y)*.*
- $b \leftarrow \mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi)$*: The verification algorithm inputs a preprocessed public parameters* $\mathsf{pp}_{f,y}$*, a public input* $\mathbf{z} \in \mathcal{X}^s$*, a commitment* c*, and an opening proof* π*. It outputs a bit* b *deciding whether to accept or reject that the vector* $\mathbf{x}$ *committed in* c *satisfies* $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$*.*

Definition 12 (Correctness). *A VC scheme for* $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ *is said to be correct if for any* $\lambda, s, w, t \in \mathbb{N}$*, any* $\mathsf{pp} \in \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$*, any* $(f, \mathbf{z}, \mathbf{x}, y) \in \mathcal{F}_{s,w,t} \times \mathcal{X}^s \times \mathcal{X}^w \times \mathcal{Y}_{s,t}$ *satisfying* $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$*, any* $(c, \mathsf{aux}) \in, (\mathsf{pp}, \mathbf{x})$*, any* $\pi \in \mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$*, and any* $\mathsf{pp}_{f,y} \in \mathsf{PreVerify}(\mathsf{pp}, (f, y))$*, it holds that* $\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi) = 1$*.*

Informally, a VC scheme is extractable if, whenever an adversary $\mathcal{A}$ is able to produce a commitment c and a valid opening proof π for some $(f(\mathbf{z}, \cdot), y(\mathbf{z}))$, then it must "know" a preimage $\mathbf{x}$ which is committed in c and satisfies $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$. Clearly, an extractable VC must also be binding, i.e. it is infeasible to open a commitment c to a set $\{(f_i(\mathbf{z}_i, \cdot), y_i(\mathbf{z}_i))\}_i$ of inconsistent function-image tuples.

Definition 13 (Extractability). *Let* $\kappa : \mathbb{N}^4 \to [0, 1]$*. A VC scheme for* $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ *is said to be* κ*-extractable if for any PPT adversary* $\mathcal{A}$ *there exists a PPT extractor* $\mathcal{E}_\mathcal{A}$ *such that the following probability is at most* $\kappa(\lambda, s, w, t)$*:*

$$\Pr\left[\begin{array}{l|l} & \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t) \\ \big(\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi) = 1\big) & (f, y, \mathbf{z}, c, \pi) \leftarrow \mathcal{A}(\mathsf{pp}; r_\mathcal{A}) \\ \wedge\ ((f, \mathbf{z}, \mathbf{x}, y) \notin \mathcal{F}_{s,w,t} \times \mathcal{X}^s \times \mathcal{X}^w \times \mathcal{Y}_{s,t} & (\mathbf{x}, r) \leftarrow \mathcal{E}_\mathcal{A}(\mathsf{pp}; r_\mathcal{A}) \\ \quad \vee\, c' \neq c \vee f(\mathbf{z}, \mathbf{x}) \neq y(\mathbf{z})) & (c', \mathsf{aux}') \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}; r) \\ & \mathsf{pp}_{f,y} \leftarrow \mathsf{PreVerify}(\mathsf{pp}, (f, y)) \end{array}\right].$$

In case Com *is deterministic, we suppress the output* r *of* $\mathcal{E}_\mathcal{A}$*. We say that the scheme is extractable if it is* κ*-extractable and* $\kappa(\lambda, s, w, t)$ *is negligible in* λ *for any* $s, w, t \in \mathsf{poly}(\lambda)$*.*

Definition 14 (Compactness). *A VC scheme for* $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ *is said to be compact if there exists* $p(\lambda, s, w, t) \in \mathsf{poly}(\lambda, \log s, \log w, \log t)$ *such that for any* $\lambda, s, w, t \in \mathbb{N}$*, any* $\mathsf{pp} \in \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$*, any* $(f, \mathbf{z}, \mathbf{x}, y) \in \mathcal{F}_{s,w,t} \times \mathcal{X}^s \times \mathcal{X}^w \times \mathcal{Y}_{s,t}$*, any* $(c, \mathsf{aux}) \in \mathsf{Com}(\mathsf{pp}, \mathbf{x})$*, and any* $\pi \in \mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$*, it holds that* $\max\{|c|, |\pi|\} \leq p(\lambda, s, w, t)$*, where* $|\cdot|$ *denotes the description size.*

4.2 Construction

A formal description of our VC construction is in Fig. 1 where important parameters and shorthands are listed and explained in Table 1.

Table 1. Parameters and shorthands with λ as security parameter.

$s \in \mathbb{N}$		Dimension of public input $\mathbf{z}$
$w \in \mathbb{N}$		Dimension of $\mathbf{v}$ and secret input $\mathbf{x}$
$t \in \mathbb{N}$		Number of outputs
$d \in \mathbb{N}$	$O(1)$	Degree of polynomial maps
$n \in \mathbb{N}$	$\mathsf{poly}(\lambda)$	Degree of $\mathcal{R}$
$\alpha \in \mathbb{R}$	$\mathsf{poly}(\lambda)$	Norm bound for f and $\mathbf{x}$
$\beta \in \mathbb{R}$	$\mathsf{poly}(\lambda)$	Norm bound for public preimages
$\delta_i \in \mathbb{R}$	$\mathsf{poly}(\lambda, s, w, t)$ (Theorem 1)	Norm bound for opening proof $\mathbf{u}_i$
$\delta_p \in \mathbb{R}$	$(s+w+d)^d\, \alpha^{d+1}\, \gamma^d\, n$	Norm bound of evaluation of a degree-d $(s+w)$-variate polynomial with coefficients of norm bounded by α at a point of norm bounded by α
$p \in \mathbb{N}$	$\geq \delta_p\, n \log n$	Moduli for $\mathcal{R}_p$
$q \in \mathbb{N}$	$\geq \max\{\delta_0, \delta_1\} \cdot n \log n$	Moduli for $\mathcal{R}_q$
$\eta_i \in \mathbb{N}$	$O(1)$	Number of rows of $\mathbf{A}_i$
$\ell_i \in \mathbb{N}$	$\geq \mathsf{lhl}(\mathcal{R}, \eta_i, q, \beta)$	Number of columns of $\mathbf{A}_i$
$\mathcal{X} \subseteq \mathcal{R}$	$\{x \in \mathcal{R} : \|x\| \leq \alpha\}$	$\mathcal{R}$ elements with norm bound α
$\mathcal{F}_{s,w,t}$		Degree-d $(s+w)$-variate t-output homogeneous polynomial maps over $\mathcal{X}$
$\mathcal{Y}_{s,t}$		s-variate t-output polynomial maps over $\mathcal{X}$
$\mathcal{E}_k \subseteq \mathbb{N}_0^w$	$\{\mathbf{e} \in \mathbb{N}_0^w : \|\mathbf{e}\|_1 = k\}$	Non-negative integer vectors of 1-norm k, for $k \in [d]$
$\mathcal{G}_0 \subseteq \mathcal{R}(\mathbf{X})$	$\bigcup_{k=1}^{d} \{\mathbf{X}^{\mathbf{e}'-\mathbf{e}} : \mathbf{e}' \neq \mathbf{e} \in \mathcal{E}_k\}$	Laurent monomials expressible as ratios of distinct degree-k monomials, for $k \in [d]$
$\mathcal{G}_1 \subseteq \mathcal{R}(\mathbf{X})$	$\{X_i : i \in \mathbb{Z}_w\}$	Degree-1 monomials
$\binom{k}{\mathbf{e}}$	$\binom{k}{e_0, \ldots, e_{w-1}}$	Multinomial coefficient, for $\mathbf{e} \in \mathcal{E}_k$ and $k \in [d]$
$\mathcal{T}_i$		Subset of $\mathcal{R}_q^{\eta_i}$ (Definition 5)
$f_{i,\mathbf{e}}$		For $f(\mathbf{Z}, \mathbf{X}) \in \mathcal{F}_{s,w,t}$, $f_{i,\mathbf{e}}(\mathbf{Z})$ is the coefficient of the monomial $\mathbf{X}^{\mathbf{e}}$ of the i-th output

The public parameters consists of a k-M-ISIS instance $(\mathbf{A}_0, \mathbf{t}_0, \mathbf{v}, (\mathbf{u}_{0,g})_{g \in \mathcal{G}_0})$ over $\mathcal{R}_q$, a correlated k-M-ISIS of knowledge instance $(\mathbf{A}_1, \mathbf{t}_1, \mathbf{v}, (\mathbf{u}_{1,g})_{g \in \mathcal{G}_1})$ over $\mathcal{R}_q$ sharing the same $\mathbf{v}$ as the k-M-ISIS instance, and a R-SIS instance $\mathbf{h}$ over $\mathcal{R}_p$, where p is short relative to q. Intuitively, the k-M-ISIS instance is for weak binding, the knowledge k-M-ISIS instance is for upgrading weak binding to extractability, and the R-SIS instance is for compactness. The commitment c to a vector $\mathbf{x}$ is simply $c := \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$.

We next explain the opening and verification mechanism. Suppose for the moment that $f(\mathbf{z}, \cdot)$ is a single-output polynomial, i.e. $t = 1$. Consider the commitment c of $\mathbf{x}$ and the evaluation of $f(\mathbf{z}, \cdot)$ at $(v_0^{-1} \cdot c, \ldots, v_w^{-1} \cdot c)$ as polynomials in $\mathbf{v}$. The value $f(\mathbf{z}, \mathbf{x})$ is encoded as the constant term in the evaluation polynomial. To open the commitment c of $\mathbf{x}$ to a function $f(\mathbf{z}, \cdot)$, the committer computes the coefficient of each non-zero Laurent monomial $g \in \mathcal{G}_0$ in the evaluation polynomial, and use these coefficients to compute a linear combination

$\mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$

$\mathbf{v} \leftarrow\$ (\mathcal{R}_q^\times)^w$

$\mathbf{h} \leftarrow\$ \mathcal{R}_p^t$

for $i \in \{0,1\}$ **do**

$(\mathbf{A}_i, \mathsf{td}_i) \leftarrow \mathsf{TrapGen}(1^{\eta_i}, 1^{\ell_i}, q, \mathcal{R}, \beta)$

$\mathbf{t}_i \leftarrow\$ \mathcal{T}_i$

$\mathbf{u}_{i,g} \leftarrow \mathsf{SampPre}(\mathsf{td}_i, g(\mathbf{v}) \cdot \mathbf{t}_i, \beta),\ \forall g \in \mathcal{G}_i$

return $\mathsf{pp} := \begin{pmatrix} \mathbf{A}_0,\ \mathbf{t}_0,\ (\mathbf{u}_{0,g})_{g\in\mathcal{G}_0}, \\ \mathbf{A}_1,\ \mathbf{t}_1,\ (\mathbf{u}_{1,g})_{g\in\mathcal{G}_1}, \\ \mathbf{v},\qquad \mathbf{h} \end{pmatrix}$

$\mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$

$\mathbf{u}_0 := \sum_{i\in\mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e}\in\mathcal{E}_k} h_i \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{u}_{0,\mathbf{e}}$

return $\pi := (\mathbf{u}_0, \mathbf{u}_1)$

$\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi)$

$b_0 := \left(\mathbf{A}_0 \cdot \mathbf{u}_0 \stackrel{?}{\equiv} \hat{f}_y(\mathbf{z}, c) \cdot \mathbf{t}_0 \bmod q\right)$

$b_1 := \left(\mathbf{A}_1 \cdot \mathbf{u}_1 \stackrel{?}{\equiv} c \cdot \mathbf{t}_1 \bmod q\right)$

$b_2 := \left(\|\mathbf{u}_0\| \stackrel{?}{\leq} \delta_0\right); b_3 := \left(\|\mathbf{u}_1\| \stackrel{?}{\leq} \delta_1\right)$

return $b_0 \wedge b_1 \wedge b_2 \wedge b_3$

$\mathsf{Com}(\mathsf{pp}, \mathbf{x})$

$c := \langle \mathbf{v}, \mathbf{x} \rangle \bmod q; \quad \mathbf{u}_1 := \sum_{X_i \in \mathcal{G}_1} x_i \cdot \mathbf{u}_{1,X_i}$

for $\mathbf{e} \in \bigcup_{k\in[d]} \mathcal{E}_k$ **do** $\mathbf{u}_{0,\mathbf{e}} := d! \cdot \sum_{\mathbf{e}'\in\mathcal{E}_k\setminus\{\mathbf{e}\}} \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{u}_{0,\mathbf{X}^{\mathbf{e}'-\mathbf{e}}}$

$\mathsf{aux} := \left((\mathbf{u}_{0,\mathbf{e}})_{\mathbf{e}\in\bigcup_{k\in[d]}\mathcal{E}_k}, \mathbf{u}_1\right)$

return (c, aux)

$\mathsf{PreVerify}(\mathsf{pp}, (f, y))$

if $(f, y) \notin \mathcal{F}_{s,w,t} \times \mathcal{Y}_{s,t}$ **then return** $\perp$

$\hat{f}_y(\mathbf{Z}, C) := d! \cdot \left(\sum_{i\in\mathbb{Z}_t} h_i \cdot \left(\sum_{k=1}^{d} \sum_{\mathbf{e}\in\mathcal{E}_k} \binom{k}{\mathbf{e}}^{-1} \cdot f_{i,\mathbf{e}}(\mathbf{Z}) \cdot \mathbf{v}^{-\mathbf{e}} \cdot C^k - y_i(\mathbf{Z})\right)\right)$

$\mathsf{pp}_{f,y} := \left(\mathbf{A}_0, \mathbf{t}_0, \mathbf{A}_1, \mathbf{t}_1, \hat{f}_y\right)$

return $\mathsf{pp}_{f,y}$

Fig. 1. Our VC Construction.

of $(\mathbf{u}_{0,g})_{g\in\mathcal{G}_0}$ to produce $\mathbf{u}_0$. In general, for $t \geq 1$, the committer further compresses the multiple instances of $\mathbf{u}_0$ into a single one using a linear combination with coefficients given by $\mathbf{h}$. To enable extraction (in the security proof), the committer also provides $\mathbf{u}_1$ which is a linear combination of $(\mathbf{u}_{1,g})_{g\in\mathcal{G}_1}$ using $\mathbf{x}$ as coefficients. Given the above, the meaning behind the verification algorithm is immediate.

Finally, we explain the choice of p and q in Table 1. First, p is chosen such that the element $f(\mathbf{z}, \mathbf{x}) - y(\mathbf{z})$ is considered short (in the context of R-SIS

problems) relative to p for all $f \in \mathcal{F}_{s,w,t}$, $y \in \mathcal{Y}_{s,t}$, $\mathbf{z} \in \mathcal{X}^s$, and $\mathbf{x} \in \mathcal{X}^w$. By some routine calculations, we can see that for such choice of $(f, \mathbf{z}, \mathbf{x}, y)$, we have $\|f(\mathbf{z}, \mathbf{x}) - y(\mathbf{z})\| \leq (s + w + d)^d \cdot \alpha^{d+1} \cdot \gamma_{\mathcal{R}}^d$. A standard choice for R-SIS problems over $\mathcal{R}_p$ is for p to be at least $n \log n$ times the norm bound; we thus simply pick this. Similarly, q is chosen such that δ_0 and δ_1 are both considered short relative to q, concretely by setting q to be $n \log n$ times the maximum among them.[8]

Remark 4 (Updating Commitments and Opening Proofs). We discuss the cost of updating a commitment of $\mathbf{x}$ to that of $\mathbf{x}'$, and an opening proof for $f(\mathbf{z}, \mathbf{x})$ to that of $f'(\mathbf{z}', \mathbf{x}')$, omitting fixed $\mathsf{poly}(\lambda)$ factors. Due to the linearity of the commitment $c = \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$ and opening proof component $\mathbf{u}_1 = \sum_{i \in \mathbb{Z}_w} x_i \cdot \mathbf{u}_{1,X_i}$ in the committed vector $\mathbf{x}$, they can be updated for a new committed vector $\mathbf{x}'$ easily by adding $\langle \mathbf{v}, \mathbf{x}' - \mathbf{x} \rangle \bmod q$ and $\sum_{i \in \mathbb{Z}_w} (x'_i - x_i) \cdot \mathbf{u}_{1,X_i}$ respectively. The computation complexity of the update is $O(\Delta)$, where Δ is the Hamming distance between $\mathbf{x}$ and $\mathbf{x}'$. Updating the $\mathbf{u}_{0,\mathbf{e}}$ terms is more computationally expensive due to its non-linearity in $\mathbf{x}$. The cost of computing the difference term for $\mathbf{u}_{0,\mathbf{e}}$ is linear in $\binom{w}{k} - \binom{w-\Delta}{k} = O(\Delta^k)$ for each $\mathbf{e} \in \mathcal{E}_k$ and each $k \in [d]$. The total work needed for updating $\{\mathbf{u}_{0,\mathbf{e}}\}_{\mathbf{e} \in \mathcal{E}_k, k \in [d]}$ is thus $O(w^d \cdot \Delta^d)$. For fixed $\mathbf{x}$ and hence fixed $\{\mathbf{u}_{0,\mathbf{e}}\}_{\mathbf{e} \in \mathcal{E}_k, k \in [d]}$, updating $\mathbf{u}_0$ by the same method costs computation linear in the Hamming distance between the coefficient vector of $f(\mathbf{z}, \cdot)$ and that of $f'(\mathbf{z}', \cdot)$.

We show that our VC construction is correct, extractable under a knowledge k-M-ISIS assumption, and compact. The formal analysis of the theorems are deferred to the full version.

Theorem 1. *For $d = O(1)$, $\ell_0 := \ell_1 := \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$,*

$$\delta_0 = 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} \quad \text{and} \quad \delta_1 = w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}},$$

our VC construction in Fig. 1 is correct.

Theorem 2. *Our VC construction for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is extractable if it is correct, $\beta \geq \alpha$, $\ell_i \geq \mathsf{lhl}(\mathcal{R}, \eta_i, q, \beta)$ for $i \in \{0, 1\}$, and the k-M-$\mathsf{ISIS}_{\mathcal{R}_q, \eta_0, \ell_0, w, \mathcal{G}_0, 1, \mathcal{D}_0, \mathcal{T}_0, \beta, 2\delta_0}$ assumption, the knowledge version k-M-$\mathsf{ISIS}_{\mathcal{R}_q, \eta_1, \ell_1, w, \mathcal{G}_1, \mathcal{D}_1, \mathcal{T}_1, \alpha, \beta, \delta_1}$ assumption, and the R-$\mathsf{SIS}_{\mathcal{R}_p, t, 2\delta_p}$ assumption hold, where $\mathcal{D}_i$ is such that the distribution*

$$\left\{ (\mathbf{A}_i, \mathbf{t}_i, \{\mathbf{u}_{\mathcal{G}_i}\}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{A}_i \leftarrow\$ \mathcal{R}_q^{\eta_i \times \ell_i};\ \mathbf{t}_i \leftarrow\$ \mathcal{T}_i;\ \mathbf{v} \leftarrow\$ (\mathcal{R}_q^\times)^w \\ \mathbf{u}_g \leftarrow\$ \mathcal{D}_{0,g,\mathbf{A}_i,\mathbf{t}_i,\mathbf{v}},\ \forall g \in \mathcal{G}_i \end{array} \right\}$$

is statistically close to the distribution

$$\left\{ (\mathbf{A}_i, \mathbf{t}_i, \{\mathbf{u}_{\mathcal{G}_i}\}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{A}_i \leftarrow\$ \mathcal{R}_q^{\eta_i \times \ell_i};\ \mathbf{t}_i \leftarrow\$ \mathcal{T}_i;\ \mathbf{v} \leftarrow\$ (\mathcal{R}_q^\times)^w \\ \mathbf{u}_g \leftarrow\$ \mathsf{SampD}(1^{\eta_i}, 1^{\ell_i}, \mathcal{R}, \beta) : \mathbf{A}_i \cdot \mathbf{u}_g \equiv g(\mathbf{v}) \cdot \mathbf{t}_i \bmod q,\ \forall g \in \mathcal{G}_i \end{array} \right\}.$$

[8] In practice the gap may be smaller or larger and when picking parameters we optimise over these gaps.

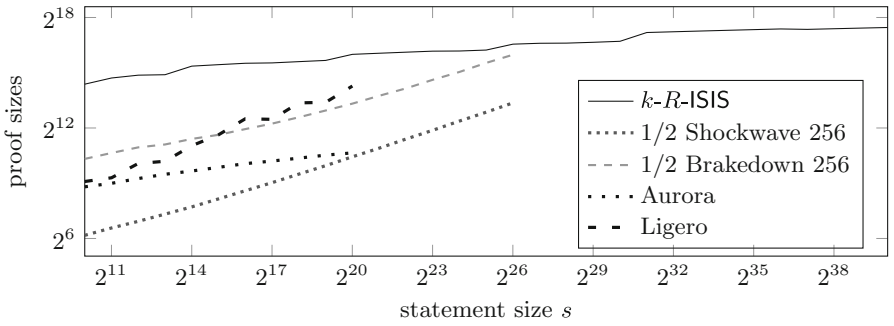

Fig. 2. Combined size (in KB) of a commitment and an opening proof for the concrete parameters chosen in Theorem 3, setting $\lambda = 128$, optimising for ρ and comparing with SNARK proof sizes in prior works [36, Fig. 5]. We picked $\alpha = s$.

Theorem 3. *For $n \in \mathsf{poly}(\lambda)$, $q, \delta_0, \delta_1 \in \mathsf{poly}(\lambda, s, w, t)$, and $\ell_0, \ell_1 \in \Theta(\log q) = \mathsf{polylog}(\lambda, s, w, t)$, covering the choices of parameters in Theorems 1 and 2, the VC construction in Fig. 1 is compact.*

Concretely, let $\mathcal{R}$ be a power-of-2 cyclotomic ring so that $\gamma_{\mathcal{R}} = n$. For $s = w = t \geq n$ and for the following choices of parameters,

$$\begin{aligned} d, \eta_0, \eta_1 &= O(1), \quad \beta \geq \alpha \\ \delta_0 &= 2 \cdot p \cdot t \cdot (s+d)^d \cdot (w+d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2}, \\ \delta_1 &= w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}}, \\ p &\approx \delta_p \cdot n \cdot \log n, \quad q \approx \delta_0 \cdot n \cdot \log n, \text{ and} \\ \ell_0 = \ell_1 &= \mathsf{lhl}(\mathcal{R}, 1, q, \beta) \approx 2 \log_\beta q, \end{aligned}$$

a commitment and openings are of size $O(n \log s)$, and $O(n \cdot (\log s + \log \beta)^2 / \log \beta)$, respectively. The minimum is attained at $\beta = \Theta(s)$, where an opening proof is of size $O(n \log s)$.

To translate these into concrete sizes we need to pick n such that solving k-R-ISIS and R-SIS costs $\approx 2^\lambda$ operations. Here it can be beneficial to set $q = \delta_0^\rho \cdot n \cdot \log n$ for some parameter $\rho \in \mathbb{N}$. Specifically, we require that $R\text{-SIS}_{\mathcal{R}_q, \ell_0, 2\cdot\sqrt{n}\cdot\delta_0}$, $R\text{-SIS}_{\mathcal{R}_q, \ell_1, 2\cdot\sqrt{n}\cdot\delta_1}$ and $R\text{-SIS}_{\mathcal{R}_p, t, 2\cdot\sqrt{n}\cdot\delta_p}$ are hard where $\delta_p := (s+w+d)^d \cdot \alpha^{d+1} \cdot \gamma_{\mathcal{R}}^d$. The factor of two arises from our reduction and the factor $\sqrt{n}$ translates between ℓ_∞ and ℓ_2. In Fig. 2 we report the concrete combined size (in KB) of a commitment and an opening proof for the concrete parameters chosen in Theorem 3, specifically setting $d = 2$, $\eta_0 = \eta_1 = 1$, and $\beta = s = w = t \in \{2^{10}, 2^{11}, \ldots, 2^{40}\}$.

To analyse computation complexity, we assume the concrete parameter choices in Theorem 3 with the exception that s, w, t are treated as free vari-

Table 2. Computation complexities (in number of $\mathcal{R}$ or $\mathcal{R}_q$ operations) of our VC.

Com	$O(w^{2d} \cdot (\log s + \log w + \log t + \log \beta)/\log \beta)$
Open	$O(t \cdot (s+w)^d \cdot (\log s + \log w + \log t + \log \beta)/\log \beta)$
PreVerify	$O(t \cdot (s+w)^d)$
Verify	$O(s^d + (\log s + \log w + \log t + \log \beta)/\log \beta)$

ables for more fine-grained complexity measures and to highlight the benefits of preprocessing. For simplicity, we assume $\max\{s, w, t\} \geq n$. The computation complexities (in number of $\mathcal{R}$ or $\mathcal{R}_q$ operations) of Com, Open, PreVerify, and Verify are reported in Table 2. Note that each $\mathcal{R}$ or $\mathcal{R}_q$ operation takes at most $\mathsf{poly}(\lambda, \log s, \log w, \log t)$ time. In summary, the combined time needed to commit to $\mathbf{x}$ and open to $f(\mathbf{z}, \cdot)$ is quasi-quadratic in the time needed to compute $f(\mathbf{z}, \mathbf{x})$, and the time needed to pre-verify (f, y) is quasi-linear in the time needed to compute $f(\mathbf{z}, \mathbf{x})$. We highlight that the online verification cost, i.e. the computation complexity of Verify, is dominated additively by s^d where s is the dimension of the public input. In applications where $s^d = O(\log w + \log t)$ and setting $\beta = \Theta(w + t)$, the online verification cost (in number of bit operations) is $O(n \log w + n \log t)$.

References

1. Agrawal, S.: Unlikely friendships: the fruitful interplay of cryptography assumptions. Invited talk at ASIACRYPT 2020, December 2020. https://youtu.be/Owz8UuWTsqg
2. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: 28th ACM STOC, pp. 99–108. ACM Press, May 1996. https://doi.org/10.1145/237814.237838
3. Albrecht, M.R., Gheorghiu, V., Postlethwaite, E.W., Schanck, J.M.: Estimating quantum speedups for lattice sieves. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part II. LNCS, vol. 12492, pp. 583–613. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-64834-3_20
4. Albrecht, M.R., Lai, R.W.F.: Subtractive sets over cyclotomic rings. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part II. LNCS, vol. 12826, pp. 519–548. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84245-1_18
5. Albrecht, M.R., Rechberger, C., Schneider, T., Tiessen, T., Zohner, M.: Ciphers for MPC and FHE. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part I. LNCS, vol. 9056, pp. 430–454. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46800-5_17
6. Attema, T., Cramer, R., Kohl, L.: A compressed Σ-protocol theory for lattices. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part II. LNCS, vol. 12826, pp. 549–579. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84245-1_19
7. Aumayr, L., et al.: Generalized channels from limited blockchain scripts and adaptor signatures. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021. LNCS, vol. 13091, pp. 635–664. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-92075-3_22

8. Becker, A., Ducas, L., Gama, N., Laarhoven, T.: New directions in nearest neighbor searching with applications to lattice sieving. In: Krauthgamer, R. (ed.) 27th SODA, pp. 10–24. ACM-SIAM, January 2016. https://doi.org/10.1137/1.9781611974331.ch2
9. Belenkiy, M., Camenisch, J., Chase, M., Kohlweiss, M., Lysyanskaya, A., Shacham, H.: Randomizable proofs and delegatable anonymous credentials. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 108–125. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-03356-8_7
10. Ben-Sasson, E., et al.: Zerocash: decentralized anonymous payments from bitcoin. In: 2014 IEEE Symposium on Security and Privacy, pp. 459–474. IEEE Computer Society Press, May 2014. https://doi.org/10.1109/SP.2014.36
11. Ben-Sasson, E., Chiesa, A., Genkin, D., Tromer, E., Virza, M.: SNARKs for C: verifying program executions succinctly and in zero knowledge. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 90–108. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40084-1_6
12. Ben-Sasson, E., Chiesa, A., Tromer, E., Virza, M.: Scalable zero knowledge via cycles of elliptic curves. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 276–294. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44381-1_16
13. Ben-Sasson, E., Chiesa, A., Tromer, E., Virza, M.: Succinct non-interactive zero knowledge for a von neumann architecture. In: Fu, K., Jung, J. (eds.) USENIX Security 2014, pp. 781–796. USENIX Association, August 2014
14. Boneh, D., Bonneau, J., Bünz, B., Fisch, B.: Verifiable delay functions. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 757–788. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96884-1_25
15. Boneh, D., Drake, J., Fisch, B., Gabizon, A.: Halo Infinite: proof-carrying data from additive polynomial commitments. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 649–680. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84242-0_23
16. Boneh, D., Drijvers, M., Neven, G.: Compact multi-signatures for smaller blockchains. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 435–464. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03329-3_15
17. Boneh, D., Freeman, D.M.: Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures. In: Catalano, D., Fazio, N., Gennaro, R., Nicolosi, A. (eds.) PKC 2011. LNCS, vol. 6571, pp. 1–16. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19379-8_1
18. Bonneau, J., Meckler, I., Rao, V., Shapiro, E.: Coda: decentralized cryptocurrency at scale. Cryptology ePrint Archive (2020)
19. Bootle, J., Chiesa, A., Sotiraki, K.: Sumcheck arguments and their applications. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part I. LNCS, vol. 12825, pp. 742–773. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-84242-0_26
20. Bowe, S., Grigg, J., Hopwood, D.: Halo: recursive proof composition without a trusted setup. Cryptology ePrint Archive, Report 2019/1021 (2019). https://eprint.iacr.org/2019/1021
21. Bünz, B., Maller, M., Mishra, P., Tyagi, N., Vesely, P.: Proofs for inner pairing products and applications. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part III. LNCS, vol. 13092, pp. 65–97. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-92078-4_3

22. Camenisch, J., Groß, T.: Efficient attributes for anonymous credentials. In: Ning, P., Syverson, P.F., Jha, S. (eds.) ACM CCS 2008, pp. 345–356. ACM Press, October 2008. https://doi.org/10.1145/1455770.1455814
23. Campanelli, M., Fiore, D., Greco, N., Kolonelos, D., Nizzardo, L.: Incrementally aggregatable vector commitments and applications to verifiable decentralized storage. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020. LNCS, vol. 12492, pp. 3–35. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-64834-3_1
24. Catalano, D., Fiore, D.: Vector commitments and their applications. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 55–72. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36362-7_5
25. Chepurnoy, A., Papamanthou, C., Zhang, Y.: Edrax: a cryptocurrency with stateless transaction validation. Cryptology ePrint Archive, Report 2018/968 (2018). https://eprint.iacr.org/2018/968
26. Drijvers, M., Gorbunov, S., Neven, G., Wee, H.: Pixel: multi-signatures for consensus. In: 29th USENIX Security Symposium (USENIX Security 2020), pp. 2093–2110. USENIX Association, August 2020. https://www.usenix.org/conference/usenixsecurity20/presentation/drijvers
27. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987). https://doi.org/10.1007/3-540-47721-7_12
28. Fisch, B.: PoReps: proofs of space on useful data. Cryptology ePrint Archive, Report 2018/678 (2018). https://eprint.iacr.org/2018/678
29. Fisch, B.: Tight proofs of space and replication. In: Ishai, Y., Rijmen, V. (eds.) EUROCRYPT 2019, Part II. LNCS, vol. 11477, pp. 324–348. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-17656-3_12
30. Garman, C., Green, M., Miers, I.: Decentralized anonymous credentials. In: NDSS 2014. The Internet Society, February 2014
31. Genise, N., Micciancio, D.: Faster gaussian sampling for trapdoor lattices with arbitrary modulus. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part I. LNCS, vol. 10820, pp. 174–203. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-78381-9_7
32. Gennaro, R., Minelli, M., Nitulescu, A., Orrù, M.: Lattice-based zk-SNARKs from square span programs. In: Lie, D., Mannan, M., Backes, M., Wang, X. (eds.) ACM CCS 2018, pp. 556–573. ACM Press, October 2018. https://doi.org/10.1145/3243734.3243845
33. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: Mitzenmacher, M. (ed.) 41st ACM STOC, pp. 169–178. ACM Press, May/June 2009. https://doi.org/10.1145/1536414.1536440
34. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC, pp. 197–206. ACM Press, May 2008. https://doi.org/10.1145/1374376.1374407
35. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd ACM STOC, pp. 99–108. ACM Press, June 2011. https://doi.org/10.1145/1993636.1993651
36. Golovnev, A., Lee, J., Setty, S., Thaler, J., Wahby, R.S.: Brakedown: linear-time and post-quantum SNARKs for R1CS. Cryptology ePrint Archive, Report 2021/1043 (2021). https://eprint.iacr.org/2021/1043
37. Gorbunov, S., Reyzin, L., Wee, H., Zhang, Z.: Pointproofs: aggregating proofs for multiple vector commitments. In: Ligatti, J., Ou, X., Katz, J., Vigna, G. (eds.) ACM CCS 2020, pp. 2007–2023. ACM Press, November 2020. https://doi.org/10.1145/3372297.3417244

38. Gorbunov, S., Vaikuntanathan, V., Wee, H.: Predicate encryption for circuits from LWE. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015, Part II. LNCS, vol. 9216, pp. 503–523. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-48000-7_25
39. Goyal, R., Koppula, V., Waters, B.: Lockable obfuscation. In: Umans, C. (ed.) 58th FOCS, pp. 612–621. IEEE Computer Society Press, October 2017. https://doi.org/10.1109/FOCS.2017.62
40. Grassi, L., Kales, D., Khovratovich, D., Roy, A., Rechberger, C., Schofnegger, M.: Starkad and Poseidon: New hash functions for zero knowledge proof systems. Cryptology ePrint Archive, Report 2019/458 (2019). https://eprint.iacr.org/2019/458
41. Gross, J.: Practical SNARK based VDF (2021). https://zkproof.org/2021/11/24/practical-snark-based-vdf/
42. Groth, J.: On the size of pairing-based non-interactive arguments. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 305–326. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_11
43. Ishai, Y., Su, H., Wu, D.J.: Shorter and faster post-quantum designated-verifier zkSNARKs from lattices. In: Vigna, G., Shi, E. (eds.) ACM CCS 2021, pp. 212–234. ACM Press, November 2021. https://doi.org/10.1145/3460120.3484572
44. Kate, A., Zaverucha, G.M., Goldberg, I.: Constant-size commitments to polynomials and their applications. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 177–194. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-17373-8_11
45. Kilian, J.: A note on efficient zero-knowledge proofs and arguments (extended abstract). In: 24th ACM STOC, pp. 723–732. ACM Press, May 1992. https://doi.org/10.1145/129712.129782
46. Koblitz, N., Menezes, A.: The brave new world of bodacious assumptions in cryptography. Not. Am. Math. Soc. **57**(3), 357–365 (2010)
47. Kosba, A.E., Miller, A., Shi, E., Wen, Z., Papamanthou, C.: Hawk: the blockchain model of cryptography and privacy-preserving smart contracts. In: 2016 IEEE Symposium on Security and Privacy, pp. 839–858. IEEE Computer Society Press, May 2016. https://doi.org/10.1109/SP.2016.55
48. Laarhoven, T.: Search problems in cryptography: from fingerprinting to lattice sieving. Ph.D. thesis, Eindhoven University of Technology (2015)
49. Lai, R.W.F., Malavolta, G.: Subvector commitments with application to succinct arguments. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part I. LNCS, vol. 11692, pp. 530–560. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26948-7_19
50. Lai, R.W.F., Malavolta, G., Ronge, V.: Succinct arguments for bilinear group arithmetic: Practical structure-preserving cryptography. In: Cavallaro, L., Kinder, J., Wang, X., Katz, J. (eds.) ACM CCS 2019, pp. 2057–2074. ACM Press, November 2019. https://doi.org/10.1145/3319535.3354262
51. Langlois, A., Stehlé, D.: Worst-case to average-case reductions for module lattices. Des. Codes Crypt. **75**(3), 565–599 (2014). https://doi.org/10.1007/s10623-014-9938-4
52. Libert, B., Ramanna, S.C., Yung, M.: Functional commitment schemes: from polynomial commitments to pairing-based accumulators from simple assumptions. In: Chatzigiannakis, I., Mitzenmacher, M., Rabani, Y., Sangiorgi, D. (eds.) ICALP 2016. LIPIcs, vol. 55, pp. 30:1–30:14. Schloss Dagstuhl, July 2016. https://doi.org/10.4230/LIPIcs.ICALP.2016.30

53. Libert, B., Yung, M.: Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 499–517. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-11799-2_30
54. Ling, S., Phan, D.H., Stehlé, D., Steinfeld, R.: Hardness of k-LWE and applications in traitor tracing. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 315–334. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44371-2_18
55. Liskov, M.: Updatable zero-knowledge databases. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 174–198. Springer, Heidelberg (2005). https://doi.org/10.1007/11593447_10
56. Lyubashevsky, V., Micciancio, D.: Generalized compact knapsacks are collision resistant. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006, Part II. LNCS, vol. 4052, pp. 144–155. Springer, Heidelberg (2006). https://doi.org/10.1007/11787006_13
57. Lyubashevsky, V., Peikert, C., Regev, O.: A toolkit for ring-LWE cryptography. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 35–54. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38348-9_3
58. Micali, S.: CS proofs (extended abstracts). In: 35th FOCS, pp. 436–453. IEEE Computer Society Press, November 1994. https://doi.org/10.1109/SFCS.1994.365746
59. Micali, S., Rabin, M.O., Kilian, J.: Zero-knowledge sets. In: 44th FOCS, pp. 80–91. IEEE Computer Society Press, October 2003. https://doi.org/10.1109/SFCS.2003.1238183
60. Micciancio, D.: Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. Comput. Complex. **16**(4), 365–411 (2007)
61. Micciancio, D., Peikert, C.: Trapdoors for lattices: simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_41
62. Parno, B., Howell, J., Gentry, C., Raykova, M.: Pinocchio: nearly practical verifiable computation. In: 2013 IEEE Symposium on Security and Privacy, pp. 238–252. IEEE Computer Society Press, May 2013. https://doi.org/10.1109/SP.2013.47
63. Peikert, C., Pepin, Z., Sharp, C.: Vector and functional commitments from lattices. Cryptology ePrint Archive, Report 2021/1254 (2021). https://ia.cr/2021/1254
64. Peikert, C., Rosen, A.: Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 145–166. Springer, Heidelberg (2006). https://doi.org/10.1007/11681878_8
65. Prest, T., et al.: FALCON. Technical report, National Institute of Standards and Technology (2020). https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions
66. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Gabow, H.N., Fagin, R. (eds.) 37th ACM STOC, pp. 84–93. ACM Press, May 2005. https://doi.org/10.1145/1060590.1060603
67. Schnorr, C., Euchner, M.: Lattice basis reduction: improved practical algorithms and solving subset sum problems. Math. Program. **66**, 181–199 (1994)

68. Stehlé, D., Steinfeld, R.: Making NTRU as secure as worst-case problems over ideal lattices. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 27–47. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20465-4_4
69. Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 617–635. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10366-7_36
70. Unruh, D.: Computationally binding quantum commitments. In: Fischlin, M., Coron, J.-S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 497–527. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49896-5_18
71. Valiant, P.: Incrementally verifiable computation or proofs of knowledge imply time/space efficiency. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 1–18. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78524-8_1
72. Wichs, D., Zirdelis, G.: Obfuscating compute-and-compare programs under LWE. In: Umans, C. (ed.) 58th FOCS, pp. 600–611. IEEE Computer Society Press, October 2017. https://doi.org/10.1109/FOCS.2017.61