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

(Full Version)

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
[2] AIT Austrian Institute of Technology
[3] Aalto University
[4] Max Planck Institute for Security and Privacy
[5] Carnegie Mellon University

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 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) [Kil92,Mic94] 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 [BCG+13,BCTV14b,PHGR13,BCC+09,CG08,GGM14]. More recently, SNARKs have found their way into real-world systems in the context of blockchain-based cryptocurrencies [BCG+14,KMS+16,BGH19,BDFG21,BMRS20].

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.

[*] 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.

[†] This work was supported by Protocol Labs under PL-RGP1-2021-050.

[‡] 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).

[§] This work was done at Friedrich-Alexander-Universität Erlangen-Nürnberg.

[¶] This work has been partially supported by the German Federal Ministry of Education and Research BMBF (grant 16K15K042, project 6GEM)

[‖] This work is supported in part by the DARPA SIEVE grant.

1.1 The Seascape of SNARKs[6]

To put our work into context, we give a brief outline of the current seascape of SNARK constructions. 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. [Gro16]) 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 [LMR19,BMM+21] and efficient recursive composition [BCTV14a]. 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) [Kil92], then compiling it into a non-interactive one using the Fiat-Shamir transformation [FS87] 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 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.[7] 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. [Reg05,GPV08], but also enable new powerful primitives [Gen09,GVW15,WZ17,GKW17], 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 [GMNO18,ISW21] or linear-time verification [ACK21,BCS21].

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

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

[7] Though we mention that there is recent progress [ARS+15,GKK+19] in crafting hash functions that are friendlier to multiparty computation and argument systems.

operations over a ring $\mathcal{R}$, and is therefore friendly to recursive composition. The soundness of our scheme is based on new lattice-based (knowledge) assumptions. The introduction of new knowledge assumptions is, to some extent, necessary: The work of Gentry and Wichs [GW11] 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 [LY10], 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 [CF13,LRY16,Fis19,LM19,GRWZ20,CFG⁺20]. 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[8] 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 [BF11,LPSS14] (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 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 [Agr20].

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 [Unr16]. 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.

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

(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 Section 7.2. 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 [Val08] and verifiable delay functions [BBBF18,Gro21].

The second application is the aggregation of GPV signatures [GPV08]. 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 Section 7.1. 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 [AEE$^+$21,EEE20,AME$^+$21]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.[9]

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 [LM19]. 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 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

[9] This generalises position binding.

$f(\mathbf{X}) := (\mathbf{A} \cdot (1, \mathbf{X})) \circ (\mathbf{B} \cdot (1, \mathbf{X})) - (\mathbf{C} \cdot (1, \mathbf{X}))$ and $\mathbf{y} = \mathbf{0}$. For a full description of our SNARK we refer the reader to Section 7.

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 $\mathcal{R}_q \coloneqq \mathcal{R}/q\mathcal{R}$ 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 $[u]_2 \coloneqq \sum_{g \in \mathcal{G}} c_g \cdot [g(\mathbf{v})]_2$ and present it to user B, who can then check the correctness of $[u]_2$ by checking

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

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 $\mathbf{u} \coloneqq \sum_{g \in \mathcal{G}} c_g \cdot \mathbf{u}_g$, although the coefficients c_g are now required to be short. The pairing-product check is then translated to checking

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

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 [CF13,LRY16,LM19] as an example.

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

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 $\left\langle \sum_{i\in\mathbb{Z}_w} x_i \cdot [v_i]_1, \sum_{j\in\mathbb{Z}_w} f_j \cdot [\bar{v}_j]_2 \right\rangle$.

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

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

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

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 $\langle\mathbf{a}, \mathbf{u}_{i,j}\rangle \equiv v_i/v_j \bmod q$, $\mathbf{u}_{i,j}$ are short
- Committing $\mathbf{x} \in \mathcal{R}^w$: $c \coloneqq \langle\mathbf{v}, \mathbf{x}\rangle \bmod q$
- Opening f: $\mathbf{u} \coloneqq \sum_{i,j\in\mathbb{Z}_w:i\neq j} f_j \cdot x_i \cdot \mathbf{u}_{i,j}$
- Verifying (f, y): $\langle\mathbf{a}, \mathbf{u}\rangle \overset{?}{\equiv} \left(\sum_{j\in\mathbb{Z}_w} f_j \cdot v_j^{-1}\right) \cdot c - y \bmod q$ 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 [AL21].

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 [GPV08]. As a bonus result, we also show how to build adaptor signatures [AEE+21] based on GPV signatures while preserving aggregatability. For more comprehensive details we refer the reader to Sections 6 and 7.2.

Aggregate GPV Signatures. GPV signatures [GPV08] are a lattice-based signature scheme paradigm of which an instantiation is a finalist in the NIST Post-Quantum Process (Falcon [PFH+20]). 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 [DGNW20]), 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 [BDN18]). Such aggregation can result in compact blocks in a blockchain as shown for the case of BLS signatures [BDN18], but now with post-quantum security.

Aggregate Adaptor Signatures. Adaptor signatures [AEE+21,EEE20,AME+21]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 [AEE+21,AME+21],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 [MP12] 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, $\mathbf{Y} := \mathbf{A} \cdot \mathbf{X} \bmod q$, is distributed statistically close to random. The signature is simply $(\mathbf{z}, \mathbf{c})$ such that during verification we have $[\mathbf{A}|\mathbf{G} + \mathbf{Y}] \cdot [\mathbf{z}|\mathbf{c}]^{\mathtt{T}} = H(m) \bmod q$ 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 $[\mathbf{A}|\mathbf{G} + \mathbf{Y}] \cdot [\hat{\mathbf{z}}|\mathbf{c}]^{\mathtt{T}} = H(m) - \mathbf{v}' \bmod q$ 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 $\mathbf{u}' := \mathbf{z} - \mathbf{z}'$. Similar to [EEE20] wehave 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 [LM19], VCs have found numerous applications, such as verifiable databases [CF13], verifiable decentralized storage [CFG+20], updatable zero-knowledge sets [MRK03,Lis05], keyless Proofs of Retrievability (PoR) [Fis18,Fis19], pseudonymous credentials [KZG10], and cryptocurrencies with stateless transaction validation [CPZ18]. Several works have studied various extensions to VC, with updatable commitments and proofs [CF13], aggregatable opening proofs for different commitments [GRWZ20], and incremental aggregatable proofs [CFG$^+$20].

Libert, Ramanna, and Yung [LRY16] 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 [LRY16], 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 [PPS21], Peikert, Pepin, and Sharp proposed a VC for positional openings based on the standard SIS assumption. Relative to our construction outlined in Section 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 [PPS21]. 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.[10]

In another recent work [AKSY21] Agrawal, Kirshanova, Stehlé, and Yadav constructed a blind signature scheme from a novel SIS-like assumption of the "one-more" flavour. Here the adversary can query ℓ arbitrary preimages for an ISIS instance and must then output $\ell + 1$ preimages of random images returned by an oracle. While this assumption is in the same "spirit" as those introduced in this work, they seem incomparable: being adaptive makes one-more-SIS potentially easier, requiring preimages of random images (hence without structure) seems to make it harder.

Prior to our work, all lattice-based SNARKs were in the designated-verifier setting. These constructions [GMNO18,ISW21] 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

[10] We stress that this does not contradict any of the claims made in [PPS21], but rather exemplifies the difference between their approach and ours.

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 $\gamma_\mathcal{R} := \max_{a,b \in \mathcal{R}} \frac{\|a \cdot b\|}{\|a\| \cdot \|b\|}$.*

Proposition 1 ([AL21]). *If $\mathcal{R} = \mathbb{Z}[\zeta_m]$ is a prime-power cyclotomic ring, then $\gamma_\mathcal{R} \leq 2\,n$. 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 $\Lambda^* = \{\mathbf{y} \in \mathbb{R}^n : \mathbf{y}^{\mathrm{T}} \cdot \Lambda \subseteq \mathbb{Z}\}$. 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.[11] 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 [SE94] with block size κ and sieving as the SVP oracle [BDGL16] (quantum algorithms do not promise a sufficiently substantial speed-up [Laa15,AGPS20]). Concretely, for $\lambda = 128$ we require $\kappa \geq 484$ and thus $\delta \leq 1.0034$.

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 [Mic07,SSTX09] or discrete Gaussian distribution [SS11,LPR13], we have that $\left(\mathbf{a}_0, \ldots, \mathbf{a}_{\ell-1}, \sum_{0 \leq i < \ell} \mathbf{a}_i \cdot r_i\right)$ 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 [GPV08,MP12,GM18]:

- $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^\eta, 1^\ell, q, \mathcal{R}, \beta)$ 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 $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$, 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 $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ outputs an element in $\mathbf{u} \in \mathcal{R}^\ell$ with norm bound $\beta' \geq \beta$. We have that $\mathbf{v} \coloneqq \mathbf{A} \cdot \mathbf{u} \bmod q$ is within $\mathsf{negl}(\lambda)$ statistical distance to $U(\mathcal{R}_q^\eta)$.
- $\mathbf{u} \leftarrow \mathsf{SampPre}(\mathsf{td}, \mathbf{v}, \beta')$ with $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ takes a trapdoor td, a vector $\mathbf{v} \in \mathcal{R}_q^\eta$, and a norm bound $\beta' \geq \beta$. It samples $\mathbf{u} \in \mathcal{R}^\ell$ satisfying $\mathbf{A} \cdot \mathbf{u} \equiv \mathbf{v} \bmod q$ and $\|\mathbf{u}\| \leq \beta'$. 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).

Proposition 3 (adapted from Lemma 5 of [AKSY21]). *For any $k > 1/\sqrt{2\,\pi}$,*

$$\Pr\left[\|\mathbf{z}\|_2 > k \cdot \sigma \cdot \sqrt{2\pi\,n};\ \mathbf{z} \leftarrow \mathcal{D}_{\mathbb{Z}^n,\sigma}\right] < \left(k \cdot \sqrt{2\pi}\right)^n \exp\left(\frac{n}{2} \cdot \left(1 - 2\pi\,k^2\right)\right),$$
$$\Pr\left[\|\mathbf{z}\|_\infty > k \cdot \sigma;\ \mathbf{z} \leftarrow \mathcal{D}_{\mathbb{Z}^n,\sigma}\right] < 2\,n \cdot \exp(-\pi k^2).$$

2.3 Rényi Divergence

Definition 2. *Let P and Q be any two discrete probability distributions such that* $\mathrm{Supp}(P) \subseteq \mathrm{Supp}(Q)$. *Then for $a \in (1, \infty)$, the Rényi Divergence (RD) of order a is defined by*

$$R_a(P\|Q) = \left(\sum_{x \in \mathrm{Supp}(P)} \frac{P(x)^a}{Q(x)^{a-1}}\right)^{\frac{1}{a-1}}.$$

Lemma 1 (in Lemma 2.9 of [BLR+18]). *Let P and Q be any two discrete probability distributions such that* $\mathrm{Supp}(P) \subseteq \mathrm{Supp}(Q)$ *and let $a \in (1, \infty)$.*

- *Let $E \subseteq \mathrm{Supp}(Q)$ be an arbitrary event, then $Q(E) \geq P(E)^{\frac{a}{a-1}} / R_a(P\|Q)$.*

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

– *Assume P and Q are two distributions of a pair of random variable (Y_0, Y_1). For $i \in \{0,1\}$ let P_i (resp. Q_i) denote the marginal distribution of Y_i under P (resp. Q), and let $P_{1|0}(\cdot|y_0)$ (resp. $Q_{1|0}(\cdot|y_0)$) denote the conditional distribution of Y_1 given that $Y_0 = y$. Then we have*

$$R_a(P||Q) = R_a(P_0||Q_0) \cdot R_a(P_1||Q_1) \text{ if } Y_0 \text{ and } Y_1 \text{ are independent.}$$

Lemma 2 ([BLR⁺18]). *For any n-dimensional lattice, $\Lambda \in \mathbb{R}^n$ and $\sigma > 0$, let P be the distribution $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}}$, and Q be the distribution $\mathcal{D}_{\Lambda,\sigma,\mathbf{c}'}$ for some fixed $\mathbf{c}, \mathbf{c}' \in \mathbb{R}^n$. If $\mathbf{c}, \mathbf{c}' \in \Lambda$, let $\epsilon = 0$. Otherwise fix $\epsilon \in (0,1)$ and assume that $\sigma > \eta_\epsilon(\Lambda)$. Then for any $a \in (1, \infty)$*

$$R_a(P\|Q) \in \left[\left(\frac{1-\epsilon}{1+\epsilon}\right)^{\frac{2}{a-1}}, \left(\frac{1+\epsilon}{1-\epsilon}\right)^{\frac{2}{a-1}}\right] \cdot \exp\left(a \cdot \pi \, \frac{\|\mathbf{c}-\mathbf{c}'\|_2^2}{\sigma^2}\right).$$

2.4 Hard Problems

The Short Integer Solution problem was introduced in the seminal work of Ajtai [Ajt96]. 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 [Mic07].

For both problems, it was shown [GPV08] that solving the problem for $q \geq \beta_2 \cdot \omega(\sqrt{n \cdot \log n})$ 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 $\beta_2 \geq \beta_\infty$, an appropriate choice of parameters is $n = \mathsf{poly}(\lambda)$, $q \geq \beta_\infty \cdot n \cdot \log n$ and $\ell \geq 2\, n \log_{\beta_\infty} q$. 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 [Mic07,PR06,LM06]; module variants in [LS15].

Definition 3 (M-SIS, adapted from [LS15]). *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 $\mathbf{u} \neq \mathbf{0} \in \mathcal{R}^\ell$ such that $\|\mathbf{u}\| \leq \beta^*$ 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 [LS15] it was shown that solving Module-SIS is as hard as finding a short basis in modules. In [LM06,PR06] 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 [Mic07]. 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 4 (k-M-SIS, generalised from [BF11,LPSS14]). *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. $\mathbf{A}\cdot\mathbf{u}_i \equiv \mathbf{0} \bmod q$ with $\|\mathbf{u}_i\| \leq \beta$. A solution to the problem is a nonzero vector $\mathbf{u} \in \mathcal{R}^\ell$ such that*

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

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 [BF11,LPSS14] 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-$\mathsf{SIS}_{\mathcal{R}_q,\ell,\beta,\beta^*} := k$-$M$-$\mathsf{SIS}_{\mathcal{R}_q,1,\ell,\beta,\beta^*}$.

2.5 Vector Commitments

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

Definition 5 (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 $s, w, t \in \mathbb{N}$.*
- $(c, \mathsf{aux}) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x})$*: 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 6 (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 $(c, \mathsf{aux}) \in \mathsf{Com}(\mathsf{pp}, \mathbf{x})$, 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 inconsistant function-image tuples.

Definition 7 (Extractability). *Let $\kappa : \mathbb{N}^4 \to [0, 1]$. A VC scheme for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is said to be $(\kappa, \mathcal{X}^*)$-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 $\kappa(\lambda, s, w, t)$:*

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

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

Definition 8 (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.*

2.6 Adaptor Signatures

Next, we recall the formal definitions of adaptor signatures [AEE$^+$21].

Definition 9 (Adaptor Signatures). *An adaptor signature scheme $\mathit{\Pi}_{\mathsf{AS}}$ w.r.t. a couple of hard relations $R, \widetilde{R}$, with $R \subseteq \widetilde{R}$, and a signature scheme $\mathit{\Pi}_{\mathcal{DS}} = (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Verify})$ consists of algorithms $(\mathsf{pSign}, \mathsf{Adapt}, \mathsf{PreVerify}, \mathsf{Ext})$ defined as:*

$\underline{\hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk}, m, Y)}$*: The pre-sign algorithm takes as input a secret key* sk*, message $m \in \{0,1\}^*$ and statement $Y \in L_R$, outputs a pre-signature $\hat{\sigma}$.*

$\underline{0/1 \leftarrow \mathsf{PreVerify}(\mathsf{pk}, m, Y, \hat{\sigma})}$*: The pre-verify algorithm takes as input a public key* pk*, message $m \in \{0,1\}^*$, statement $Y \in L_R$ and pre-signature $\hat{\sigma}$, outputs a bit b.*

$\underline{\sigma \leftarrow \mathsf{Adapt}(\hat{\sigma}, y)}$*: The adapt algorithm takes as input a pre-signature $\hat{\sigma}$ and witness y, outputs a signature σ.*

$\underline{y \leftarrow \mathsf{Ext}(\sigma, \hat{\sigma}, Y)}$*: The extract algorithm takes as input a signature σ, pre-signature $\hat{\sigma}$ and statement $Y \in L_R$, outputs a witness y such that $(Y, y) \in \widetilde{R}$, or $\perp$.*

The correctness definition of adaptor signatures is described below.

Definition 10 (Pre-signature Correctness). *An adaptor signature scheme $\mathit{\Pi}_{\mathsf{AS}}$ satisfies pre-signature correctness if for every $\lambda \in \mathbb{N}$, every message $m \in \{0,1\}^*$ and every statement/witness pair $(Y, y) \in R$, the following holds:*

$$\Pr\left[\begin{array}{l|l} \quad \mathsf{PreVerify}(\mathsf{pk}, m, Y, \hat{\sigma}) = 1 & (\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}(1^\lambda) \\ \wedge\ \mathsf{Verify}(\mathsf{pk}, m, \sigma) = 1 & \hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk}, m, Y) \\ & \sigma := \mathsf{Adapt}(\hat{\sigma}, y) \\ \wedge\ (Y, y') \in \widetilde{R} & y' := \mathsf{Ext}(\sigma, \hat{\sigma}, Y) \end{array}\right] = 1.$$

Next, we formally define the security properties of an adaptor signature scheme. We relax the definition of unforgeability, introduced in [AEE$^+$21], by restricting the adversary to query any given message $m \in \{0,1\}^*$ only once to one of the two oracle, either $\mathsf{Sign}\mathcal{O}(\cdot)$ or $\mathsf{pSign}\mathcal{O}(\cdot,\cdot)$. Looking ahead, we require this relaxation in order to prove the security of our adaptor signature scheme. Our instantiation is based on the GPV signature scheme [GPV08], and it is proven secure in the random oracle model, by relying on the programmability of the RO. The above restriction allows us to apply the same technique to prove the security of the adaptor signature scheme, as the random oracle needs to be programmed at most once for any given message m. However, this relaxation does not seem to lead to any practical security consequence as in real-world application, typical signed messages contain a time-stamp, and thus users never get to sign the same message more than once. Moreover, one could rely on the probabilistic FDH version of the GVP signature in order to overcome such drawback: every time the pSign or Sign algorithms are executed on input a message m, a fresh salt t is sampled, the message $m||t$ is signed, and t is appended to the so produced signature. This modification is in fact equivalent to the introduced restriction of the adversary as the introduced salt forces the adversary, with high probability, to only get signatures of different messages (i.e., different $(m||t)$).

Definition 11 (Weak Unforgeability). *An adaptor signature scheme $\mathit{\Pi}_{\mathsf{AS}}$ is* $\mathsf{aEUF\text{-}CMA}$ *secure if for every* PPT *adversary $\mathcal{A}$ there exists a negligible function* negl *such that:*

$$\Pr\left[\mathsf{aSigForge}_{\mathcal{A},\mathit{\Pi}_{\mathsf{AS}}}(\lambda) = 1\right] \leq \mathsf{negl}(\lambda)$$

where the experiment $\mathsf{aSigForge}_{\mathcal{A},\mathit{\Pi}_{\mathsf{AS}}}$ is defined as follows:

Definition 12 (Weak Pre-signature Adaptability). *An adaptor signature scheme $\mathit{\Pi}_{\mathsf{AS}}$ satisfies weak pre-signature adaptability if for any $\lambda \in \mathbb{N}$, any message $m \in \{0,1\}^*$, any statement/witness pair $(Y, y) \in R$, any key pair $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}(1^\lambda)$ and any pre-signature $\hat{\sigma} \leftarrow \{0,1\}^*$ with $\mathsf{PreVerify}(\mathsf{pk}, m, Y, \hat{\sigma}) = 1$ we have:*

$$\Pr[\mathsf{Verify}(\mathsf{pk}, m, \mathsf{Adapt}(\hat{\sigma}, y)) = 1] = 1$$

Definition 13 (Weak Witness Extractability). *An adaptor signature scheme $\mathit{\Pi}_{\mathsf{AS}}$ is* witness extractable *if for every* PPT *adversary $\mathcal{A}$, there exists a negligible function* negl *such that the following holds:*

$$\Pr[\mathsf{aWitExt}_{\mathcal{A},\mathit{\Pi}_{\mathsf{AS}}}(\lambda) = 1] \leq \mathsf{negl}(\lambda)$$

where the experiment $\mathsf{aWitExt}_{\mathcal{A},\mathit{\Pi}_{\mathsf{AS}}}$ is defined as follows

```
$\mathsf{aSigForge}_{\mathcal{A},\Pi_{\mathsf{AS}}}(\lambda)$
$\mathcal{Q} := \emptyset$
$(\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{KGen}(1^\lambda)$
$m \leftarrow \mathcal{A}^{\mathsf{Sign}\mathcal{O}(\cdot),\mathsf{pSign}\mathcal{O}(\cdot,\cdot)}(\mathsf{pk})$
$(Y,y) \leftarrow \mathsf{GenR}(1^\lambda)$
$\hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk},m,Y)$
$\sigma \leftarrow \mathcal{A}^{\mathsf{Sign}\mathcal{O}(\cdot),\mathsf{pSign}\mathcal{O}(\cdot,\cdot)}(\hat{\sigma},Y)$
return $(m \notin \mathcal{Q} \wedge \mathsf{Verify}(\mathsf{pk},m,\sigma))$

$\mathsf{Sign}\mathcal{O}(m)$
if $m \in \mathcal{Q}$
    return $\bot$
$\sigma \leftarrow \mathsf{Sign}(\mathsf{sk},m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\sigma$

$\mathsf{pSign}\mathcal{O}(m,Y)$
if $m \in \mathcal{Q}$
    return $\bot$
$\hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk},m,Y)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\hat{\sigma}$
```

Fig. 1. (Weak) Unforgeabiltiy experiment of adaptor signatures

```
$\mathsf{aWitExt}_{\mathcal{A},\Pi_{\mathsf{AS}}}(\lambda)$
$\mathcal{Q} := \emptyset$
$(\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{KGen}(1^\lambda)$
$(m,Y) \leftarrow \mathcal{A}^{\mathsf{Sign}\mathcal{O}(\cdot),\mathsf{pSign}\mathcal{O}(\cdot,\cdot)}(\mathsf{pk})$
$\hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk},m,Y)$
$\sigma \leftarrow \mathcal{A}^{\mathsf{Sign}\mathcal{O}(\cdot),\mathsf{pSign}\mathcal{O}(\cdot,\cdot)}(\hat{\sigma})$
$y' := \mathsf{Ext}(\mathsf{pk},\sigma,\hat{\sigma},Y)$
return $(m \notin \mathcal{Q} \wedge (Y,y') \notin \widetilde{R}$
 $\wedge\, \mathsf{Verify}(\mathsf{pk},m,\sigma))$

$\mathsf{Sign}\mathcal{O}(m)$
if $m \in \mathcal{Q}$
    return $\bot$
$\sigma \leftarrow \mathsf{Sign}(\mathsf{sk},m)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\sigma$

$\mathsf{pSign}\mathcal{O}(m,Y)$
if $m \in \mathcal{Q}$
    return $\bot$
$\hat{\sigma} \leftarrow \mathsf{pSign}(\mathsf{sk},m,Y)$
$\mathcal{Q} := \mathcal{Q} \cup \{m\}$
return $\hat{\sigma}$
```

Fig. 2. (Weak) Witness extractability experiment for adaptor signatures

2.7 Argument Systems

Definition 14 (Hard Relation). *For a relation R, with statement/witness (Y,y), let L_R be the associated language defined as $\{Y \mid \exists\ y \text{ s.t. } (Y,y) \in R\}$. We say that R is a hard relation if the following holds:*

i) There exists a PPT sampling algorithm GenR *that on input 1^λ outputs a statement/witness pair $(Y,y) \in R$,*
ii) The relation is poly-time decidable,
iii) For all PPT $\mathcal{A}$ the probability of $\mathcal{A}$ on input Y outputting a valid witness y is negligible.

We recall the definition of a non-interactive zero-knowledge proof of knowledge (NIZK-PoK) with online extractors as introduced in [Fis05].

Definition 15 (NIZK-PoK). *A tuple* $(\mathsf{Setup},\mathsf{Prove},\mathsf{Verify})$ *of PPT algorithms is called a NIZK with an online extractor for a relation R, and random oracle $\mathcal{H}$, if the following holds:*

i) Completeness: For all $\lambda \in \mathbb{N}$ and any $(Y,y) \in R$, it holds that

$$\mathsf{Verify}(\mathsf{pp},Y,\mathsf{Prove}(\mathsf{pp},Y,y)) = 1$$

except with negligible probability,

ii) Zero knowledge: If there exists a negligible function μ, a PPT simulator $\mathcal{S} = (\mathcal{S}_1,\mathcal{S}_2)$, such that for all $\lambda \in \mathbb{N}$, any $(Y,y) \in R$, and any PPT adversary $\mathcal{A}$, such that the following probability is bound by *a negligible function μ.*

$$\Pr\left[\begin{array}{c|l} & b \leftarrow \{0,1\} \\ & \textit{If } b = 0 \\ & \;\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda) \\ b' = \mathcal{A}(\mathsf{pp}, Y, \pi) & \;\pi \leftarrow \mathsf{Prove}(\mathsf{pp}, Y, y) \\ \wedge\, b = b' & \textit{else if } b = 1 \\ & \;(\mathsf{pp}, \mathsf{state}_0) \leftarrow \mathcal{S}_1(1^\lambda) \\ & \;\pi \leftarrow \mathcal{S}_2(\mathsf{pp}, \mathsf{state}_0, Y) \end{array}\right]$$

iii) Online Extractor: There exist a PPT online extractor K *with access to the sequence of queries to the random oracle and its answers, such that given (Y, π), the algorithm* K *can extract the witness y with $(Y, y) \in R$.*

2.8 SNARKs for Polynomial Map Satisfiability

We define the NP language of the satisfiability of systems of multivariate polynomials over $\mathcal{R}$ with bounded coefficients. It is straightforward to check that the language is NP-complete. In particular, it contains the NP-complete language of rank-1 constraint satisfiability (R1CS) over $\mathcal{R}$ [BCS21] as a subset.

Definition 16. *Let $d, \alpha \in \mathbb{N}$ with $d \geq 2$. The satisfiability of systems of degree-d polynomials over $\mathcal{R}$ with norm bound α is the language $\mathrm{PolySAT}_{\mathcal{R},d,\alpha} = \bigcup_{s,w,t \in \mathbb{N}} \mathcal{L}_{s,w,t}$ where*

$$\mathcal{L}_{s,w,t} := \{\, (f, y, \mathbf{z}) \in \mathcal{F}_{s,w,t} \times \mathcal{Y}_{s,t} \times \mathcal{X}^s : \exists\, \mathbf{x} \in \mathcal{X}^w,\;\; f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z}) \,\}\,.$$

where $\mathcal{F}_{s,w,t}$, $\mathcal{Y}_{s,t}$, and $\mathcal{X}$ are defined as in Table 1.

We recall the definition of succinct non-interactive arguments of knowledge (SNARKs). For concreteness, we state the definition with respect to the language $\mathrm{PolySAT}_{\mathcal{R},d,\alpha}$.

Definition 17 (Preprocessing Non-Interactive Arguments). *A preprocessing non-interactive argument system Π for $\mathrm{PolySAT}_{\mathcal{R},d,\alpha}$ is a tuple of PPT algorithms $(\mathsf{Setup}, \mathsf{Prove}, \mathsf{PreVerify}, \mathsf{Verify})$ 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 and size parameters $\lambda, s, w, t \in \mathbb{N}$.*
- $\pi \leftarrow \mathsf{Prove}(\mathsf{pp}, (f, y, \mathbf{z}), \mathbf{x})$*: The proving algorithm generates a proof π on input the public parameters* pp*, a statement $(f, y, \mathbf{z})$, and a witness $\mathbf{x}$.*
- $\mathsf{pp}_{f,y} \leftarrow \mathsf{PreVerify}(\mathsf{pp}, (f, y))$*: The preverification algorithm inputs the public parameters* pp *and a partial statement (f, y). It outputs the preprocessed public parameters* $\mathsf{pp}_{f,y}$*.*
- $b \leftarrow \mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, \pi)$*: The verification algorithm returns a bit b (denoting acceptance or rejection) on input the preprocessed public parameters* $\mathsf{pp}_{f,y}$ *and a proof π.*

In the following definitions, we use "a system" to refer to a preprocessing non-interactive argument system for $\mathrm{PolySAT}_{\mathcal{R},d,\alpha}$.

Definition 18 (Completeness). *A system Π is said to be complete 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 $(f, y, \mathbf{z}) \in \mathcal{F}_{s,w,t} \times \mathcal{Y}_{s,t} \times \mathcal{X}^s$ and $\mathbf{x} \in \mathcal{X}^w$ satisfying $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$, any $\pi \in \mathsf{Prove}(\mathsf{pp}, (f, y, \mathbf{z}), \mathbf{x})$, and any $\mathsf{pp}_{f,y} \in \mathsf{PreVerify}(\mathsf{pp}, (f, y))$, it holds that $\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, \pi) = 1$.*

Definition 19 (Succinctness). *A system Π is said to be succinct 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 $(f, y, \mathbf{z}) \in \mathcal{F}_{s,w,t} \times \mathcal{Y}_{s,t} \times \mathcal{X}^s$ and $\mathbf{x} \in \mathcal{X}^w$ satisfying $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$, any $\pi \in \mathsf{Prove}(\mathsf{pp}, (f, y, \mathbf{z}), \mathbf{x})$, and any $\mathsf{pp}_{f,y} \in \mathsf{PreVerify}(\mathsf{pp}, (f, y))$, the runtime of $\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, \pi)$ is upper-bounded by a fixed polynomial in $\mathsf{poly}(\lambda, s, \log w, \log t)$.*

Definition 20 (Knowledge Soundness). *Let $\kappa : \mathbb{N}^4 \to [0,1]$. A system Π is said to be $(\kappa, \mathcal{X}^*)$-knowledge-sound if for any PPT adversary $\mathcal{A}$ there exists a PPT extractor $\mathcal{E}_\mathcal{A}$ such that the following probability is at most $\kappa(\lambda)$:*

$$\Pr\left[\begin{array}{l|l} \big(\mathsf{Verify}(\mathsf{pp}_{f,y},\mathbf{z},\pi)=1\big)\wedge & \mathsf{pp}\leftarrow\mathsf{Setup}(1^\lambda,1^s,1^w,1^t)\\ \Big(((f,y,\mathbf{z})\notin \mathcal{F}_{s,w,t}\times\mathcal{Y}_{s,t}\times\mathcal{X}^s) & ((f,y,\mathbf{z}),\pi)\leftarrow\mathcal{A}(\mathsf{pp};r_\mathcal{A})\\ \vee(\mathbf{x}\notin(\mathcal{X}^*)^w) & \mathbf{x}\leftarrow\mathcal{E}_\mathcal{A}(\mathsf{pp},r_\mathcal{A})\\ \vee f(\mathbf{z},\mathbf{x})\neq y(\mathbf{z})\Big) & \mathsf{pp}_{f,y}\leftarrow\mathsf{PreVerify}(\mathsf{pp},(f,y))\end{array}\right]$$

We say that the SNARK is $\mathcal{X}^*$-knowledge-sound if it is $(\kappa, \mathcal{X}^*)$-knowledge-sound and $\kappa(\lambda, s, w, t)$ is a negligible in λ for any $s, w, t \in \mathsf{poly}(\lambda)$.

Definition 21 (Preprocessing SNARKs). *A preprocessing non-interactive argument system Π is said to be a preprocessing SNARK if it is complete, succinct, and $\mathcal{X}^*$-knowledge-sound for some $\mathcal{X}^* \supseteq \mathcal{X}$.*

Sometimes SNARKs are required to be zero-knowledge (zk-SNARKs), in which case we also require the existence of a simulator that is able to generate valid proofs without knowing the witness. Contrary to standard zero-knowledge proofs, SNARKs are already non-trivial to construct without zero-knowledge, so we treat this aspect as tangential to our main result. We refer the reader to Definition 15 for a formal definition of this property.

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 $\mathbf{t} := \mathbf{A} \cdot \mathbf{u} \bmod q$. 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 sense since then $\{\, \mathbf{u}_i \,\}$ constitutes a trapdoor for $\mathbf{A}$ when the $\mathbf{u}_i$ are linearly independent [GPV08]. Formally, the problem as stated is impossible to solve since all vectors will be in $\mathbb{Q}\text{-}\operatorname{span}(\{\, \mathbf{u}_i \,\}_{0 \leq i < k})$, 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 independent 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 22 (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 23 (k-M-ISIS Assumptions). *Let $\ell, \eta \in \mathbb{N}$. Let q be a rational prime, $\mathcal{R}$ the m-th cyclotomic ring, and $\mathcal{R}_q := \mathcal{R}/q\mathcal{R}$ with $1/|\mathcal{R}_q| = \mathsf{negl}(\lambda)$. 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*

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

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

Definition 24 (*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" [KM10] we make the following, strong, meta assumption.

Definition 25 (*k*-*M*-ISIS Meta Assumption). *For any* k*-*M*-*ISIS*-admissible* $(\mathcal{G}, g^*)$*,* k*-*M*-*$\mathsf{ISIS}_{\mathsf{pp}}$ *with* $\mathsf{pp} := (\mathcal{R}_q, \eta, \ell, w, \mathcal{G}, g^*, \mathcal{D}, \mathcal{T}, \beta, \beta^*)$ *is hard if* M*-*$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,\beta^*}$ *is hard and if* R*-*$\mathsf{ISIS}_{\mathcal{R}_q,k,\beta^*}$ *is hard.*[12]

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 26 (Knowledge *k*-*M*-ISIS Assumption). *Adopt the notation from Definition 23, 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,* $\eta > 1$ *and* $\mathcal{T}$ *contains elements* $t \in \mathcal{R}_q^\eta$ *s.t.* $|\,\langle t\rangle\,|/|\mathcal{R}_q^\eta| = \mathsf{negl}(\lambda)$*. 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*

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

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 $\mathbf{A}\cdot\mathbf{u} \bmod q$ 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 $\mathbf{A}\cdot\mathbf{u} \bmod q$ 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.

[12] The second condition can be tightened somewhat, see discussion in "A Solution in $\mathsf{Span}_\mathcal{R}(\{\, \mathbf{u}_g \,\}_{g\in\mathcal{G}})$" in Section 4.1.

Definition 27 (Knowledge k-R-ISIS Assumption). *Let the parameters* pp *be as in Definition 23 except that* $\eta = 1$ *and* $\mathcal{T}$ *contains elements* $t \in \mathcal{R}_q$ *s.t.* $1/|\langle t \rangle| = \mathsf{negl}(\lambda)$ *and* $|\langle t \rangle|/|\mathcal{R}_q| = \mathsf{negl}(\lambda)$*. Furthermore, let* $\mathcal{S}_t \coloneqq \{s \in \mathcal{R}_q \mid s \cdot t \equiv 0 \bmod q\}$ *and let* $\mathcal{T}$ *be such that finding* $s' \in \mathcal{S}_t$ *with* $0 < \|s'\| \leq \alpha^*$ *is hard for* $t \leftarrow\$\, \mathcal{T}$*.*[13] *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*

$$\mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda) \coloneqq \Pr\left[\begin{array}{l|l} \quad\langle \mathbf{a}, \mathbf{u} \rangle \equiv c \cdot t \bmod q & \mathbf{a} \leftarrow\$\, \mathcal{R}_q^{\ell} \\ \wedge\ 0 < \|\mathbf{u}\| \leq \beta^* & t \leftarrow\$\, \mathcal{T};\ \ \mathbf{v} \leftarrow\$\, (\mathcal{R}_q^{\times})^{w} \\ \wedge\ \neg \left(\begin{array}{l} \quad c \equiv \sum\limits_{g \in \mathcal{G}} x_g \cdot g(\mathbf{v}) \bmod q \\ \wedge\ \left\|(x_g)_{g \in \mathcal{G}}\right\| \leq \alpha^* \end{array}\right) & \begin{array}{l} \mathbf{u}_g \leftarrow\$\, \mathcal{D}_{g,\mathbf{a},t,\mathbf{v}},\ \ \forall\ g \in \mathcal{G} \\ \Big((c, \mathbf{u}), (x_g)_{g \in \mathcal{G}}\Big) \\ \quad \leftarrow (\mathcal{A} \| \mathcal{E}_{\mathcal{A}})\,(\mathbf{a}, t, \{\, \mathbf{u}_{\mathcal{G}} \,\}, \mathbf{v}) \end{array} \end{array}\right].$$

Definition 28 (Knowledge k-M-ISIS Meta Assumption). *Let* $(\mathcal{G}, g^*)$ *be* k-M-ISIS*-admissible,* α^*, β^* *be reals with* $\alpha^* \geq \beta^* \geq 1$*, and* $\eta > 1$*. The knowledge* k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,w,\mathcal{G},\mathcal{D},\mathcal{T},\alpha^*,\beta,\beta^*}$ *assumption holds if the* k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,w,\mathcal{G},g^*,\mathcal{D},\mathcal{T},\beta,\beta^*}$ *assumption holds.*

Remark 4. We note that our meta assumption does not cover knowledge k-R-ISIS since it is not a true special case of k-M-ISIS as discussed above. See also discussion in Section 4.2.

4 Analysing the k-M-ISIS Assumption

We give reductions studying the properties of our new assumptions. We first show that there exist hard instances of the k-R-ISIS problem. In particular, in Lemmas 3 and 4 we show that k-R-ISIS (with $g^* \equiv 1$) is as hard as R-SIS when $w \geq k$ and when the system generated by $\mathcal{G}$ is efficiently invertible. In both lemmas, we use that $g(\mathbf{v}) \sim U((\mathcal{R}_q^{\times})^{w}) \approx U(\mathcal{R}_q^{w})$ and these reductions do not apply to k-M-ISIS, i.e. $\eta > 1$. In Theorem 1 we show that k-M-ISIS is at least as hard as k-R-ISIS. This, on the one hand, formalises the intuition that increasing the module rank does not make the problem easier but, on the other hand, also shows that the additional structure (restricting to multiples of $\mathbf{t} \coloneqq (1, 0, \ldots, 0)^{\mathsf{T}}$) preserves hardness. Using the same techniques, in Theorem 2 we also show that k-M-ISIS is a true generalisation of k-R-SIS. We stress, however, that none of the above reductions cover the case we use for our example application in Section 5.

We next study the relations between k-M-ISIS (but not just k-R-ISIS) problems for different choices of $(\mathcal{G}, g^*)$. In Lemma 6 we show that $(\mathcal{G}, g^*)$ is as hard as $(\mathcal{G}, 0)$ for any $\mathcal{G}$, formalising the intuition that the non-homogeneous variant is no easier than the homogeneous variant. Then, in Lemma 7 we show that scaling $(\mathcal{G}, g^*)$ multiplicatively by any non-zero Laurent monomial does not change the hardness, e.g. we may choose to normalise instances to $g^* \equiv 1$.

Finally, in Section 4.1, we investigate attacks on the k-M-ISIS problem. These attacks do not outperform standard attacks on SIS and we will use them to set parameters in Section 5.1.

Some k-R-ISIS $\geq$ R-SIS. First, we show that giving out up to w constraints and when $g^* \equiv 1$ then k-R-ISIS is no easier than R-SIS. Under this condition, we can simply sample random preimages and solve for the right $\mathbf{v}$ to satisfy the $\mathcal{G}$ constraints.

Lemma 3. *Let the parameters* pp *be as in Definition 23. Furthermore, let* $g^* \equiv 1$*,* $\mathcal{G} = \{g_i(\mathbf{X})\}_{i \in \mathbb{Z}_k} \subset \mathcal{R}(\mathbf{X})$ *be of size* $k \leq w$*, the number of variables, and* $\mathcal{D}$ *be such that the distribution*

$$\left\{(\mathbf{a}, t, \{\, \mathbf{u}_i \,\}, \mathbf{v}) \middle| \mathbf{a} \leftarrow\$\, \mathcal{R}_q^{\ell};\ t \leftarrow\$\, \mathcal{T};\ \mathbf{v} \leftarrow\$\, (\mathcal{R}_q^{\times})^{w};\ \mathbf{u}_i \leftarrow\$\, \mathcal{D}_{g_i,\mathbf{a},t,\mathbf{v}},\ \forall i \in \mathbb{Z}_k\right\}$$

is statistically close to the distribution

$$\left\{(\mathbf{a}, t, \{\, \mathbf{u}_i \,\}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{a} \leftarrow\$\, \mathcal{R}_q^{\ell};\ t \leftarrow\$\, \mathcal{T};\ \mathbf{v} \leftarrow\$\, (\mathcal{R}_q^{\times})^{w} \\ \mathbf{u}_i \leftarrow\$\, \mathsf{SampD}(1^1, 1^{\ell}, \mathcal{R}, \beta) : \langle \mathbf{a}, \mathbf{u}_i \rangle \equiv g_i(\mathbf{v}) \cdot t \bmod q,\ \forall i \in \mathbb{Z}_k \end{array}\right\}.$$

[13] 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 $\mathcal{R}$. See Section 4.2 for more discussion on the choices for $(\mathcal{R}_q, \mathcal{T})$.

Write $g_i(\mathbf{X}) = \mathbf{X}^{\mathbf{e}_i}$*,* $\mathbf{E} = (\mathbf{e}_i)_{i\in\mathbb{Z}_k} \in \mathbb{Z}^{k\times k}$*, and* $(g_i(\mathbf{v}))_{i\in\mathbb{Z}_k} = \mathbf{v}^{\mathbf{E}}$*. If* $\mathcal{R}_q$ *is a field, let* $\gcd(\det(\mathbf{E}), q^n - 1) = 1$*. Otherwise let* $\det(\mathbf{E}) = \pm 1$*. Then for any PPT adversary* $\mathcal{A}$ *against* k-R-$\mathsf{ISIS}_{\mathsf{pp}}$ *there exists a PPT adversary* $\mathcal{A}'$ *against* R-SIS *with*

$$\mathsf{Adv}^{\text{r-sis}}_{\mathcal{R}_q,\ell+1,\beta^*,\mathcal{A}'}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda).$$

Proof. Wlog we consider $k = w$ by simply only submitting a subset of our preimages to the adversary. Also wlog we assume $\mathcal{T} = \{1\}$ as discussed in Remark 2. We construct an R-SIS solver as follows: On input of an R-SIS instance $\mathbf{a}'$, write $\mathbf{a}' = (\bar{\mathbf{a}}, a')$ and set $\mathbf{a} = \frac{1}{a'} \cdot \bar{\mathbf{a}}$. If no a' is invertible in $\mathcal{R}_q$ the reduction aborts. By our choice of q, with non-negligible probability over the randomness of $\mathbf{a}'$ the reduction does not abort, and in which case $\mathbf{a}$ is uniformly distributed over $\mathcal{R}_q^\ell$. For $i \in \mathbb{Z}_k$ sample $\mathbf{u}_i \leftarrow\!\$\, \mathsf{SampD}(1^1, 1^\ell, \mathcal{R}, \beta)$ and compute $t_i = \langle \mathbf{a}, \mathbf{u}_i\rangle$. Since $\ell \geq \mathsf{lhl}(\mathcal{R}, 1, q, \beta)$, $\mathbf{t} \in \mathcal{R}_q^k$ is distributed within negligible statistical distance to uniform. By our choice of q, we have $\mathbf{t} \in (\mathcal{R}_q^\times)^k$ with non-negligible probability. Compute $\mathbf{v} = (\mathbf{t})^{\mathbf{E}^{-1}}$. We can write $\mathbf{E}^{-1}$ because $\mathbf{E}^{-1} = \mathbf{F}/r$ where $\mathbf{F}$ is over $\mathbb{Z}$ and $r := |\det(\mathbf{E})| \in \mathbb{Z}$. If $\det(\mathbf{E}) = \pm 1$ compute $\mathbf{v}$ directly. Otherwise, note that every element in a finite field of order q^n has an r-th root if $\gcd(r, q^n - 1) = 1$ and computing r-th roots can be accomplished by computing $r^{-1} \bmod (q^n - 1)$. Note that this implies r-th roots are unique under these conditions and the map is a bijection. Thus, the map defined by $g_i(\mathbf{X})$ is a bijection, implying our sampling procedure produces well distributed inputs.

Run the k-R-ISIS solver on $(\mathbf{a}, \{\mathbf{u}_i\}_{i\in\mathbb{Z}_k}, \mathbf{v})$ to obtain $(\mathbf{u}^*, s^*)$ satisfying $\langle \mathbf{a}, \mathbf{u}^*\rangle \equiv s^* \bmod q$. Output $\mathbf{u}' = (\mathbf{u}^*, -s^*)$. We observe that

$$\begin{aligned}\langle \mathbf{a}, \mathbf{u}^*\rangle &\equiv s^* \bmod q\\ \langle (\mathbf{a}, 1), (\mathbf{u}^*, -s^*)\rangle &\equiv 0 \bmod q\\ \langle a' \cdot (\mathbf{a}, 1), (\mathbf{u}^*, -s^*)\rangle &\equiv 0 \bmod q\\ \langle \mathbf{a}', \mathbf{u}'\rangle &\equiv 0 \bmod q\end{aligned}$$

Our R-SIS solver runs in time proportional to our k-R-ISIS solver. Finally, observe that $\|\mathbf{u}'\| \leq \beta^*$ if the k-M-ISIS adversary succeeded. □

Next, we show that for some additional forms of $\mathcal{G}$, too, k-R-ISIS is equivalent to R-SIS. Here we use the freedom to sample v_i to fix up images.

Lemma 4. *Let the parameters* pp *be as in Definition 23. Furthermore, let* $w = w' + k$ *for some* $w' \in \mathbb{N}$*,* $\mathcal{G}$ *be of the form*

$$\mathcal{G} = \{ g_i(\mathbf{X}) \}_{i\in\mathbb{Z}_k} = \{X_{w'+i} \cdot \prod_{j\in\mathbb{Z}_{w'}} X_j^{e_j}\}_{i\in\mathbb{Z}_k},$$

and $\mathcal{D}$ *be such that the distribution*

$$\left\{(\mathbf{a}, t, \{\mathbf{u}_i\}, \mathbf{v}) \middle| \mathbf{a} \leftarrow\!\$\, \mathcal{R}_q^\ell;\ t \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w;\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{g_i,\mathbf{a},t,\mathbf{v}},\ \forall i \in \mathbb{Z}_k\right\}$$

is statistically close to the distribution

$$\left\{(\mathbf{a}, t, \{\mathbf{u}_i\}, \mathbf{v}) \middle| \begin{array}{l}\mathbf{a} \leftarrow\!\$\, \mathcal{R}_q^\ell;\ t \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w\\ \mathbf{u}_i \leftarrow\!\$\, \mathsf{SampD}(1^1, 1^\ell, \mathcal{R}, \beta) : \langle \mathbf{a}, \mathbf{u}_i\rangle \equiv g_i(\mathbf{v}) \cdot t \bmod q,\ \forall i \in \mathbb{Z}_k\end{array}\right\}.$$

For any PPT adversary $\mathcal{A}$ *against* k-R-$\mathsf{ISIS}_{\mathsf{pp}}$ *there exists a PPT adversary* $\mathcal{A}'$ *against* R-SIS *with*

$$\mathsf{Adv}^{\text{r-sis}}_{\mathcal{R}_q,\ell,\beta^*,\mathcal{A}'}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda).$$

Proof. Let $\mathbf{a}$ be a R-$\mathsf{SIS}_{\mathcal{R}_q,\ell,\beta^*}$ instance. By assumption, $\mathbf{a}$ is uniformly distributed over $\mathcal{R}_q^\ell$. Sample $\mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^{w'}$ and $\mathbf{u}_i \leftarrow\!\$\, \mathsf{SampD}(1^1, 1^\ell, \mathcal{R}, \beta)$ for all $i \in \mathbb{Z}_k$. Compute $y_i \equiv \langle \mathbf{a}, \mathbf{u}_i\rangle \bmod q$ and $v_{w'+i} \equiv y_i \cdot \prod_{j\in\mathbb{Z}_{w'}} v_j^{-e_j} \bmod q$ for all $i \in \mathbb{Z}_k$.

If y_i is not invertible for any $i \in \mathbb{Z}_k$ the reduction aborts. If the reduction does not abort, which happens with non-negligible probability, since $\ell \geq \mathsf{lhl}(\mathcal{R}, 1, q, \beta)$, for each $i \in \mathbb{Z}_k$, y_i is uniformly distributed over $\mathcal{R}_q^\times$, and so is $v_{w'+i}$. We therefore conclude that $\mathbf{v}$ is uniformly distributed over $(\mathcal{R}_q^\times)^w$. Run the k-R-ISIS adversary on the input $(\mathbf{a}, \{\mathbf{u}_i\}_{i\in\mathbb{Z}_k}, \mathbf{v})$ to obtain $(s^*, \mathbf{u}_{g^*})$. By construction $\mathbf{u}_{g^*}$ satisfies $\langle \mathbf{a}, \mathbf{u}_{g^*}\rangle \equiv 0 \bmod q$ if the k-R-ISIS adversary succeeded. □

***k*-*M*-ISIS $\geq$ *k*-*R*-(I)SIS.** We show that k-M-ISIS is no easier than k-R-ISIS. The analogous reduction for M-ISIS and R-ISIS is trivial. Here we face the complication that we have to map the known preimages to k-M-ISIS while preserving a mapping back to make use of the returned k-M-ISIS solution in k-R-ISIS. We do this by constructing a lower-triangular matrix that satisfies our constraints and hide its structure by multiplying with a short upper triangular matrix (with a short inverse). We then use Rényi divergence arguments to break thus introduced dependencies. Our reduction has several limitations: (i) It requires $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ rather than $\ell > \mathsf{lhl}(\mathcal{R}, 1, q, \beta)$ for the input k-R-ISIS instance and (ii) it produces an output distribution $\mathcal{D}$ for k-M-ISIS that is non-spherical. For ease of exposition and because we do not require the more general case in this work, we give our reduction for $\eta = 2$.

Theorem 1. *Let the parameters* pp_M *and* pp_R *for* k-M-ISIS *and* k-R-ISIS *respectively be as in Definition 23, such that they share the same ring* $\mathcal{R}_q$*, number of variables* w*, and monomials* $(\mathcal{G}, g^*)$*. Differing parameters are distinguished by subscripts, e.g.* ℓ_M *and* ℓ_R*. Furthermore, let* $\eta_M = 2$, $\beta^*_\Delta \in \mathbb{R}$, $\sigma, \sigma_\Delta > \eta_\epsilon(\mathcal{R}) \in \mathbb{R}$, $\beta_x \geq \sigma_x$ *be s.t.* $u \sim \mathcal{D}_{\mathcal{R}^\ell, \sigma_x}$ *satisfy* $\|u\|_\infty \leq \beta_x$ *for* $x \in \{ R, M, \Delta \}$, $\ell_\Delta \coloneqq \ell_M - \ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \beta_\Delta)$, $\sigma_R > \gamma_\mathcal{R} \cdot (\ell_\Delta \cdot n)^{3/2} \cdot \gamma_\mathcal{R} \cdot \sigma_\Delta \cdot \sigma$, $\beta^*_R \geq 2\,\ell_\Delta \cdot \gamma_\mathcal{R} \cdot \sqrt{n} \cdot \sigma \cdot \beta^*_\Delta$, $\ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \sigma)$ *and* $\geq \mathsf{lhl}(\mathcal{R}, 2, q, \beta_R)$*. Let* $\mathcal{D}_R$ *be such that the distribution*

$$\left\{(\mathbf{a}, t, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \mathbf{a} \leftarrow\!\$\, \mathcal{R}_q^{\ell_R};\ t \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w;\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{g_i, \mathbf{a}, \mathbf{v}},\ \forall i \in \mathbb{Z}_k \quad \right\}$$

is statistically close to the distribution

$$\left\{(\mathbf{a}, t, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{a} \leftarrow\!\$\, \mathcal{R}_q^{\ell_R};\ t \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w \\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{\mathcal{R}^{\ell_R}, \sigma_R} : \langle \mathbf{a}, \mathbf{u}_i \rangle \equiv g_i(\mathbf{v}) \cdot t \bmod q,\ \forall i \in \mathbb{Z}_k \end{array} \right\}.$$

Let $\mathcal{D}_M$ *be such that the distribution*

$$\left\{(\mathbf{A}, \mathbf{t}, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \mathbf{A} \leftarrow\!\$\, \mathcal{R}_q^{\eta_M \times \ell_M};\ \mathbf{t} \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w;\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{g_i, \mathbf{A}, \mathbf{t}, \mathbf{v}},\ \forall i \in \mathbb{Z}_k \right\}$$

is statistically close to the distribution

$$\left\{(\mathbf{A}, \mathbf{t}, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{A} \leftarrow\!\$\, \mathcal{R}_q^{\eta_M \times \ell_M};\ \mathbf{t} \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w \\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{\mathcal{R}^{\ell_R}, \sigma_R} \times \mathcal{D}_{\mathcal{R}^{\ell_\Delta}, \sigma_\Delta} : \mathbf{A} \cdot \mathbf{u}_i \equiv g_i(\mathbf{v}) \cdot \mathbf{t} \bmod q,\ \forall i \in \mathbb{Z}_k \end{array} \right\}.$$

Let SampD *and* SampPre *output samples following a Discrete Gaussian distribution of appropriate width* σ_x *given* β_x*. For any PPT adversary* $\mathcal{A}$ *against* k-M-$\mathsf{ISIS}_{\mathsf{pp}_M}$ *there exists a PPT adversary* $\mathcal{A}'$ *against* k-R-$\mathsf{ISIS}_{\mathsf{pp}_R}$ *with*

$$\mathsf{Adv}^{\text{k-r-isis}}_{\mathsf{pp}_R, \mathcal{A}'}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp}_M, \mathcal{A}}(\lambda).$$

Using the same proof strategy, we show that some k-M-ISIS adversaries can break k-M-SIS. To ease readability, the formal statement below is for k-R-SIS, i.e. k-M-SIS with $\eta = 1$. The only non-trivial step is to argue that the output solution satisfies the additional constraint imposed by k-M-SIS. Here we use an unrelated R-SIS instances to argue that the adversary either broke R-SIS or the solution satisfies the required constraint that it is not in $\mathcal{K}\text{-}\operatorname{span}(\{ \mathbf{u}_i \}_{0 \leq i < k})$.

Theorem 2. *Let the parameters* pp_M *for* k-M-ISIS *be as in Definition 23. Furthermore, let* $\eta_M = 2$, $g^*_M = 0$, $\beta^*_\Delta \in \mathbb{R}$, $\sigma, \sigma_\Delta > \eta_\epsilon(\mathcal{R}) > 1 \in \mathbb{R}$, $\beta_x \geq \sigma_x$ *be s.t.* $u \sim \mathcal{D}_{\mathcal{R}, \sigma_x}$ *satisfy* $\|u\|_\infty \leq \beta_x$ *for* $x \in \{ R, M, \Delta \}$, $\ell_\Delta \coloneqq \ell_M - \ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \beta_\Delta)$, $\sigma_R > (4\,\gamma_\mathcal{R} \cdot \sigma_\Delta \cdot \sigma \cdot \ell_\Delta \cdot n)$, $\beta^*_R \geq 2\,\ell_\Delta \cdot \gamma_\mathcal{R} \cdot \sqrt{2\,\pi n} \cdot \sigma \cdot \beta^*_\Delta$, $\ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \sigma)$ *and* $\geq \mathsf{lhl}(\mathcal{R}, 2, q, \beta_R)$*. Let* $\mathcal{D}_M$ *be such that the distribution*

$$\left\{(\mathbf{A}, \mathbf{t}, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \mathbf{A} \leftarrow\!\$\, \mathcal{R}_q^{\eta_M \times \ell_M};\ \mathbf{t} \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w;\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{g_i, \mathbf{A}, \mathbf{t}, \mathbf{v}},\ \forall i \in \mathbb{Z}_k \right\}$$

is statistically close to the distribution

$$\left\{(\mathbf{A}, \mathbf{t}, \{ \mathbf{u}_i \}, \mathbf{v}) \middle| \begin{array}{l} \mathbf{A} \leftarrow\!\$\, \mathcal{R}_q^{\eta_M \times \ell_M};\ \mathbf{t} \leftarrow\!\$\, \mathcal{T};\ \mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w \\ \mathbf{u}_i \leftarrow\!\$\, \mathcal{D}_{\mathcal{R}^{\ell_R}, \sigma_R} \times \mathcal{D}_{\mathcal{R}^{\ell_\Delta}, \sigma_\Delta} : \mathbf{A} \cdot \mathbf{u}_i \equiv g_i(\mathbf{v}) \cdot \mathbf{t} \bmod q,\ \forall i \in \mathbb{Z}_k \end{array} \right\}.$$

Let SampD *and* $\mathsf{SampPre}$ *output samples following a Discrete Gaussian distribution of appropriate width* σ_x *given* β_x*. For any PPT adversary* $\mathcal{A}$ *against* k*-*M*-*$\mathsf{ISIS}_{\mathsf{pp}_M}$ *there exists a PPT adversary* $\mathcal{A}'$ *or* $\mathcal{A}''$ *against* k*-*M*-*$\mathsf{SIS}_{\mathcal{R}_q,1,\ell_R,\beta_R,\beta_R^*}$ *or* R*-*$\mathsf{SIS}_{\mathcal{R}_q,1,\ell_R,\beta_R^*}$ *respectively with*

$$\mathsf{Adv}^{\text{k-r-sis}}_{\mathcal{R}_q,\ell_R,\beta,\beta_R^*,\mathcal{A}'}(\lambda) + \mathsf{Adv}^{\text{r-sis}}_{\mathcal{R}_q,\ell_R,\beta_R^*,\mathcal{A}''}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp}_M}(\lambda).$$

We first state and prove a technical lemma that we will rely on in both proofs. It allows us to argue, using Rényi and statistical distance arguments, that the structured inputs we provide to the k-M-ISIS adversary are sufficiently close to what this adversary expects for it to succeed.

Lemma 5. *Consider*

$$\mathbf{A} \coloneqq \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} = \begin{pmatrix} \mathbf{a} & \mathbf{a} \cdot \mathbf{R} \\ \mathbf{r} & \mathbf{b} \end{pmatrix},$$
$$\mathbf{U} \coloneqq \begin{pmatrix} \mathbf{I} & -\mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{U}_R \\ \mathbf{W}_\Delta \end{pmatrix} = \begin{pmatrix} \mathbf{U}_R - \mathbf{R} \cdot \mathbf{W}_\Delta \\ \mathbf{W}_\Delta \end{pmatrix},$$

where $\mathbf{a}, \mathbf{r} \leftarrow\!\!\$\, \mathcal{R}_q^{\ell_R}$*,* $\mathbf{b} \leftarrow\!\!\$\, \mathcal{R}_q^{\ell_\Delta}$*,* $\mathbf{R} \in \mathcal{R}^{\ell_R \times \ell_\Delta}$ *with each entry sampled independently from* $\mathcal{D}_{\mathcal{R},\sigma}$*,* $\mathbf{U}_R \in \mathcal{R}^{\ell_R \times k}$ *with each entry sampled independently from* $\mathcal{D}_{\mathcal{R},\sigma_R}$*,* $\mathbf{W}_\Delta \in \mathcal{R}^{\ell_\Delta \times k}$ *with entry is sampled independently from* $\mathcal{D}_{\mathcal{R},\sigma_\Delta}$*.*

Let $\sigma, \sigma_\Delta > \eta_\epsilon(\mathcal{R}) \in \mathbb{R}$*,* $\beta_x \geq \sigma_x$ *be s.t.* $u \sim \mathcal{D}_{\mathcal{R}^\ell,\sigma_x}$ *satisfy* $\|u\|_\infty \leq \beta_x$ *for* $x \in \{R, M, \Delta\}$*,* $\ell_\Delta \coloneqq \ell_M - \ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \beta_\Delta)$*,* $\sigma_R > \gamma_{\mathcal{R}} \cdot (\ell_\Delta \cdot n)^{3/2} \cdot \gamma_{\mathcal{R}} \cdot \sigma_\Delta \cdot \sigma$*,* $\ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \sigma)$ *and* $\geq \mathsf{lhl}(\mathcal{R}, 2, q, \beta_R)$*.*

Let $\mathsf{pp}_{M'}$ *be as in Definition 23 except that* $\mathbf{A}$ *is sampled as above and* $\mathbf{u}_i$ *are sampled as the columns of* $\mathbf{U}$ *subject to* $\mathbf{A} \cdot \mathbf{U} \equiv \mathbf{G} \cdot \mathbf{t} \bmod q$ *where* $\mathbf{G} \coloneqq \mathcal{M}_{\mathcal{G}}(\mathbf{g})$*. Let* SampD *and* $\mathsf{SampPre}$ *output samples following a Discrete Gaussian distribution of appropriate width* σ_x *given* β_x*. Let* $\mathcal{A}$ *be a* k*-*M*-*ISIS *adversary solving instances sampled as in Definition 23 with non-negligible probability, then* $\mathcal{A}$ *also solves instances with* $\mathsf{pp}_{M'}$ *with non-negligible probability.*

Proof. We argue this by defining a series of hybrid experiments for sampling $(\mathbf{A}, \mathbf{t}, \mathbf{U}, \mathbf{v})$:

Hyb_0**:** The input $(\mathbf{A}_0, \mathbf{t}, \mathbf{U}_0, \mathbf{v})$ is sampled as above.
Hyb_1**:** In this experiment $(\mathbf{A}_1, \mathbf{t}, \mathbf{U}_1, \mathbf{v})$ is sampled such that $\mathbf{U}_1$ is sampled independent of $\mathbf{R}$, i.e. $\mathbf{u}_g \coloneqq (\mathbf{u}_g^{(R)}, \mathbf{u}_g^{(\Delta)})$ where $\mathbf{u}_g^{(R)} \sim \mathcal{D}_{\mathcal{R}^{\ell_R},\sigma_R}$ and $\mathbf{u}_g^{(\Delta)} \sim \mathcal{D}_{\mathcal{R}^{\ell_\Delta},\sigma_\Delta}$.
Hyb_2**:** In this experiment $(\mathbf{A}_2, \mathbf{t}, \mathbf{U}_2, \mathbf{v})$ is sampled as in the k-M-ISIS definition.

We first establish the closeness between the distributions Hyb_0 and Hyb_1.

Claim. The Rényi divergence between Hyb_0 and Hyb_1 is at most a constant.

Proof. We first show how we can sample from Hyb_1. Let $\mathbf{R}_1 \leftarrow\!\!\$\, \mathcal{R}^{\ell_R \times \ell_\Delta}$ be sampled as in Hyb_0. Sample $(\mathbf{X}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(2, \ell_R, q, \mathcal{R}, \beta_R)$, $\mathbf{y} \leftarrow\!\!\$\, \mathcal{R}_q^{\ell_M}$, write $\mathbf{x}_i$ for the i-th row of $\mathbf{X}$, and set

$$\mathbf{A}_1 \coloneqq \begin{pmatrix} \mathbf{x}_0 & \mathbf{0} \\ \mathbf{x}_1 & \mathbf{y} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R}_1 \\ \mathbf{0} & \mathbf{I} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_0 & \mathbf{x}_0 \cdot \mathbf{R}_1 \\ \mathbf{x}_1 & \mathbf{y} \end{pmatrix}.$$

Note that $\mathbf{x}_0$, $\mathbf{x}_1$, and $\mathbf{y}$ play the roles of $\mathbf{a}$, $\mathbf{r}$, and $\mathbf{b}$ in Hyb_0 respectively. Then, sample $\mathbf{W}_{\Delta,1} \leftarrow\!\!\$\, \mathcal{D}_{\mathcal{R}^{\ell_\Delta \times k},\sigma_\Delta}$ and $\mathbf{U}_{R,1} \leftarrow\!\!\$\, \mathsf{SampPre}\left(\mathsf{td}, \mathbf{G} \cdot \mathbf{t} - \begin{pmatrix} \mathbf{x}_0 \cdot \mathbf{R}_1 \\ \mathbf{y} \end{pmatrix} \mathbf{W}_{\Delta,1}, \beta_R\right)$ so that they satisfy

$$\mathbf{A}_1 \cdot \begin{pmatrix} \mathbf{U}_{R,1} \\ \mathbf{W}_{\Delta,1} \end{pmatrix} \equiv \mathbf{G} \cdot \mathbf{t} \bmod q.$$

We next argue about the closeness of Hyb_0 and Hyb_1. Write

$$\mathbf{U}_0 = ((\mathbf{U}_{R,0} + \mathbf{R}_0 \cdot \mathbf{W}_{\Delta,0})^{\mathsf{T}} \| \mathbf{W}_{\Delta,0}^{\mathsf{T}})^{\mathsf{T}}.$$

Since $\ell_R \geq \mathsf{lhl}(\mathcal{R}, 2, q, \beta_R)$ and by the properties of $\mathsf{TrapGen}$ we have that $\mathbf{A}_0$ and $\mathbf{A}_1$ are statistically close. We also note that $\mathbf{W}_{\Delta,0}$ and $\mathbf{W}_{\Delta,1}$ are identically distributed. Next, we consider the distribution

$\mathcal{D}_{\mathsf{Hyb}_1} := \mathcal{D}_{\mathcal{R}^{\ell_R \times k}, \sigma_R}$ of $\mathbf{U}_{R,1}$ and the distribution $\mathcal{D}_{\mathsf{Hyb}_0}$ of $\mathbf{U}_{R,0} + \mathbf{R}_0 \cdot \mathbf{W}_{\Delta,0}$, where we recall that $\mathbf{U}_{R,0} \sim \mathcal{D}_{\mathcal{R}^{\ell_R \times k}, \sigma_R}$, $\mathbf{W}_{\Delta,0} \sim \mathcal{D}_{\mathcal{R}^{\ell_\Delta \times k}, \sigma_\Delta}$ and $\mathbf{R}_0 \sim \mathcal{D}_{\mathcal{R}^{\ell_R \times \ell_\Delta}, \sigma}$. By Proposition 3 $\|\mathbf{W}_{\Delta,0}\| \leq \sqrt{\ell_\Delta \cdot n} \cdot \sigma_\Delta$, each column $\mathbf{r}$ of $\mathbf{R}_0$ satisfies $\|\mathbf{r}\| \leq \sqrt{\ell_\Delta \cdot n} \cdot \sigma$ and thus $\|\mathbf{R}_0 \cdot \mathbf{W}_{\Delta,0}\|_2 \leq (\ell_\Delta \cdot n)^{3/2} \cdot \gamma_\mathcal{R} \cdot \sigma_\Delta \cdot \sigma$. By Lemma 2, the Rényi divergence of order $a \in (1, \infty)$ is thus

$$R_a \left(\mathcal{D}_{\mathsf{Hyb}_1} \| \mathcal{D}_{\mathsf{Hyb}_0} \right) \leq \exp \left(a\, \pi \cdot \left((\ell_\Delta \cdot n)^{3/2} \cdot \gamma_\mathcal{R} \cdot \sigma_\Delta \cdot \sigma \right)^2 / (\sigma_R)^2 \right).$$

By assumption $\sigma_R > \gamma_\mathcal{R} \cdot (\ell_\Delta \cdot n)^{3/2} \cdot \gamma_\mathcal{R} \cdot \sigma_\Delta \cdot \sigma$ and thus the Rényi divergence $R_a \left(\mathcal{D}_{\mathsf{Hyb}_1} \| \mathcal{D}_{\mathsf{Hyb}_0} \right)$, and hence $R_a \left(\mathsf{Hyb}_1 \| \mathsf{Hyb}_0 \right)$, is bounded by a constant. □

Next, let E be the event that the k-M-ISIS adversary is successful when given $(\mathbf{A}, \mathbf{t}, \mathbf{U}, \mathbf{v})$, and denote the probability of this event happening when $(\mathbf{A}, \mathbf{t}, \mathbf{U}, \mathbf{v})$ is sampled from Hyb_1 by $\mathsf{Hyb}_1(E)$. By Lemma 1 we have that $\mathsf{Hyb}_0(E) \geq \mathsf{Hyb}_1(E)^{a/(a-1)} / R_a(\mathsf{Hyb}_0 \| \mathsf{Hyb}_1)$. Taking any constant $a > 1$ establishes that $(\mathbf{A}_0, \mathbf{t}, \mathbf{U}_0, \mathbf{v})$ sampled from Hyb_0 is sufficiently well distributed for the adversary to succeed if it does for $(\mathbf{A}_1, \mathbf{t}, \mathbf{U}_1, \mathbf{v})$ sampled from Hyb_1.

It remains to show that $\mathsf{Hyb}_1(E)$ and $\mathsf{Hyb}_2(E)$ are (statistically) indistinguishable. For this, we use that $\ell_R \geq \mathsf{lhl}(\mathcal{R}, 1, q, \sigma)$ and the distributions of $\mathbf{a}, \mathbf{b}, \mathbf{r}$ to conclude that $\mathbf{A}_1$ and $\mathbf{A}_2$ are statistically close, which implies that the distributions Hyb_1 and Hyb_2 are statistically close. The statistical indistinguishability between $\mathsf{Hyb}_1(E)$ and $\mathsf{Hyb}_2(E)$ follows. □

Proof (of Theorem 1). Let $(\mathbf{a}, t, \{ \mathbf{u}_\mathcal{G} \}, \mathbf{v}) \in \mathcal{R}_q^{\ell_R} \times \mathcal{R}_q \times \mathcal{R}_q^w \times \mathcal{R}^{\ell_R \times k}$ be a k-R-ISIS instance. Without loss of generality (Remark 2), suppose $t = 1$. Our reduction samples: $(\mathbf{b}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1, \ell_\Delta, q, \mathcal{R}, \beta_\Delta)$, $\mathbf{r} \leftarrow\!\!\$\, \mathcal{R}_q^{\ell_R}$ and a short matrix $\mathbf{R} \in \mathcal{R}^{\ell_R \times \ell_\Delta}$ where each entry is sampled independently from $\mathcal{D}_{\mathcal{R},\sigma}$.

Let $\mathbf{U} \in \mathcal{R}^{\ell_R \times k} := \mathcal{M}_\mathcal{G}(\{ \mathbf{u}_\mathcal{G} \})$. For each $g \in \mathcal{G}$, sample short preimages $\mathbf{w}_g \leftarrow\!\!\$\, \mathsf{SampPre}(\mathsf{td}, -\langle \mathbf{r}, \mathbf{u}_g \rangle, \beta_\Delta)$. Note that $0 \equiv \langle \mathbf{r}, \mathbf{u}_g \rangle + \langle \mathbf{b}, \mathbf{w}_g \rangle \bmod q$. Let $\mathbf{W} \in \mathcal{R}^{\ell_\Delta \times k} := \mathcal{M}_\mathcal{G}(\{ \mathbf{w}_\mathcal{G} \})$ and $\mathbf{G} \in \mathcal{R}_q^{1 \times k} := \mathcal{M}_\mathcal{G}(\{ g(\mathbf{v}) \})$. We construct

$$\mathbf{A}' := \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} = \begin{pmatrix} \mathbf{a} & \mathbf{a} \cdot \mathbf{R} \\ \mathbf{r} & \mathbf{b} \end{pmatrix},$$

$$\mathbf{U}' := \begin{pmatrix} \mathbf{I} & -\mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{U} \\ \mathbf{W} \end{pmatrix} = \begin{pmatrix} \mathbf{U} - \mathbf{R} \cdot \mathbf{W} \\ \mathbf{W} \end{pmatrix}.$$

Without loss of generality (Remark 2), suppose that $\mathbf{t} = (1, 0)^\mathsf{T}$. By construction we have

$$\mathbf{A}' \cdot \mathbf{U}' \equiv \mathbf{G} \cdot \mathbf{t} \bmod q$$

as required. Our reduction runs the k-M-ISIS adversary on $(\mathbf{A}', \mathbf{t}, \mathbf{U}', \mathbf{v})$. When the adversary returns a short preimage $\mathbf{u}^*$ of $s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t}$ we have

$$\begin{aligned} s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} &\equiv \mathbf{A}' \cdot \mathbf{u}^* \bmod q \\ &\equiv \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_0^* \\ \mathbf{u}_1^* \end{pmatrix} \bmod q \\ &\equiv \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^* \\ \mathbf{u}_1^* \end{pmatrix} \bmod q \\ s^* \cdot g^*(\mathbf{v}) &\equiv \langle \mathbf{a}, \mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^* \rangle \bmod q, \end{aligned}$$

i.e. $\mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^*$ is a solution for k-R-ISIS. By Proposition 3 the entries of $\mathbf{R}$ are bounded by $\sqrt{n} \cdot \sigma$ with overwhelming probability. Thus, $\|\mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^*\| \leq 2\, \ell_\Delta \cdot \gamma_\mathcal{R} \cdot \sqrt{2\, \pi n} \cdot \sigma \cdot \beta_M^* \leq \beta_R^*$.

Finally, to show that the input $(\mathbf{A}', \mathbf{t}, \mathbf{U}', \mathbf{v})$ to the k-M-ISIS adversary is (sufficiently) well distributed, we apply Lemma 5.

Proof (of Theorem 2). Let $(\mathbf{a}, \{ \mathbf{u}_i \}) \in \mathcal{R}_q^{\ell_R} \times \mathcal{R}^{\ell_R \times k}$ be a k-R-SIS instance. Our reduction samples: $\mathbf{v} \in (\mathcal{R}_q^\times)^w$, $(\mathbf{b}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1, \ell_\Delta, q, \mathcal{R}, \beta_\Delta)$, $\mathbf{r} \leftarrow\!\!\$\, \mathcal{R}_q^{\ell_R}$ and a short matrix $\mathbf{R} \in \mathcal{R}^{\ell_R \times \ell_\Delta}$ where each entry is sampled independently from $\mathcal{D}_{\mathcal{R},\sigma}$. Let $\mathbf{U} \in \mathcal{R}^{\ell_R \times k}$ be the matrix where $\mathbf{u}_i$ are the columns.

For each $0 \leq i < k$, sample short preimages $\mathbf{w}_{g_i} \leftarrow\!\!\$\ \mathsf{SampPre}(\mathsf{td}, -\langle \mathbf{r}, \mathbf{u}_i \rangle + g_i(\mathbf{v}), \beta_{\Delta})$. Note that $g_i(\mathbf{v}) \equiv \langle \mathbf{r}, \mathbf{u}_{g_i} \rangle + \langle \mathbf{b}, \mathbf{w}_{g_i} \rangle \bmod q$. Let $\mathbf{W} \in \mathcal{R}^{\ell_\Delta \times k} := \mathcal{M}_{\mathcal{G}}(\{\, \mathbf{w}_{\mathcal{G}} \,\})$ and $\mathbf{G} \in \mathcal{R}_q^{1 \times k} := \mathcal{M}_{\mathcal{G}}(\{\, g(\mathbf{v}) \,\})$. We construct

$$\begin{aligned} \mathbf{A}' &:= \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} = \begin{pmatrix} \mathbf{a} & \mathbf{a} \cdot \mathbf{R} \\ \mathbf{r} & \mathbf{b} \end{pmatrix}, \\ \mathbf{U}' &:= \begin{pmatrix} \mathbf{I} & -\mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{U} \\ \mathbf{W} \end{pmatrix} = \begin{pmatrix} \mathbf{U} - \mathbf{R} \cdot \mathbf{W} \\ \mathbf{W} \end{pmatrix}. \end{aligned}$$

Without loss of generality (Remark 2), suppose that $\mathbf{t} = (0, 1)^{\mathsf{T}}$. By construction we have

$$\mathbf{A}' \cdot \mathbf{U}' \equiv \mathbf{G} \cdot \mathbf{t} \bmod q$$

as required. Our reduction runs the k-M-ISIS adversary on $(\mathbf{A}', \mathbf{t}, \mathbf{U}', \mathbf{v})$. When the adversary returns a short preimage $\mathbf{u}^*$ of $s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t}$ we have

$$\begin{aligned} s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} &\equiv \mathbf{A}' \cdot \mathbf{u}^* \bmod q \\ &\equiv \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{I} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_0^* \\ \mathbf{u}_1^* \end{pmatrix} \bmod q \\ &\equiv \begin{pmatrix} \mathbf{a} & \mathbf{0} \\ \mathbf{r} & \mathbf{b} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^* \\ \mathbf{u}_1^* \end{pmatrix} \bmod q \\ 0 &\equiv \langle \mathbf{a}, \mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^* \rangle \bmod q, \end{aligned}$$

i.e. $\mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^*$ is a candidate solution for k-R-SIS. First, we bound its norm. By Proposition 3 the entries of $\mathbf{R}$ are bounded by $\sqrt{n} \cdot \sigma$ with overwhelming probability. Thus, $\|\mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^*\| \leq 2\, \ell_{\Delta} \cdot \gamma_{\mathcal{R}} \cdot \sqrt{2\, \pi n} \cdot \sigma \cdot \beta_M^* \leq \beta_R^*$.

Second, we establish that the solution is a valid k-R-SIS solution, i.e. not in the span of the $\mathbf{u}_i$. We distinguish two cases.

$\mathbf{u}_1^* = \mathbf{0}$. In this case we also have $\langle \mathbf{r}, \mathbf{u}_0^* \rangle \equiv g_M^*(\mathbf{v}) \equiv 0 \bmod q$, i.e. $\mathbf{u}_0^*$ is solution to the R-$\mathsf{SIS}_{\mathcal{R}_q, \ell_R, \beta_R^*}$ instance $\mathbf{r}$. In other words, if this case happens with non-negligible probability, we could construct a PPT algorithm for R-$\mathsf{SIS}_{\mathcal{R}_q, \ell_R, \beta_R^*}$.

$\mathbf{u}_1^* \neq \mathbf{0}$. It remains to be argued that $\mathbf{u}^* \notin \mathcal{K}\text{-}\operatorname{span}(\{\, \mathbf{u}_i \,\}_{0 \leq i < k})$ with non-negligible probability. First, note that $\mathbf{R}$ is information-theoretically hidden from the k-M-ISIS adversary. Now, suppose the contrary is true, i.e. that we have $\mathbf{u}_0^* + \mathbf{R} \cdot \mathbf{u}_1^* = \sum_{i \in \mathbb{Z}_k} a_i \cdot \mathbf{u}_i$ for some $a_i \in \mathcal{K}$. If this relation holds over $\mathcal{K}$ it must also hold mod 2. By [GPV08, Corollary 2.8], the distribution of $\mathbf{R}$ mod 2 is statistically close to $U(\mathcal{R}_2^{\ell_R \times \ell_\Delta})$ and thus $\mathbf{R} \cdot \mathbf{u}_1^*$ is uniform mod 2. Moreover in the worst case $\mathcal{R}_2$ splits into n copies of $\mathbb{Z}_2$. It suffices to consider only one copy. We thus may ask when $\sum_{i \in \mathbb{Z}_k} a_i \cdot \mathbf{u}_i \equiv \mathbf{R} \cdot \mathbf{u}_i^* \bmod 2$ for any $\mathbf{u}_i \in \mathbb{Z}_2^{\ell_R}$ has a solution $a_i \in \{0, 1\}^k$. Consider the matrix spanned by $\mathbf{u}_i$ and consider its echelon form. It has at most k pivot positions and thus at least $\ell_R - k$ non-pivot positions. Thus, the probability (over the randomness in $\mathbf{R}$) of satisfying the constraint is $\leq 1/2^{\ell_R - k} \leq 1/2$ since $k < \ell_R$. Thus with probability $> 1/2$ we have $\mathbf{u}^* \notin \mathcal{K}\text{-}\operatorname{span}(\{\, \mathbf{u}_i \,\}_{0 \leq i < k})$.

Finally, to show that the input $(\mathbf{A}', \mathbf{t}, \mathbf{U}', \mathbf{v})$ to the k-M-ISIS adversary is (sufficiently) well distributed, we apply Lemma 5. □

$(\mathcal{G}, g^*) \geq (\mathcal{G}, 0)$. The next lemma shows that solving for any $(\mathcal{G}, g^*)$ is as hard as solving for $(\mathcal{G}, 0)$.

Lemma 6. *Let the parameters* $\mathsf{pp} = (\mathcal{R}_q, \eta, \ell, w, \mathcal{G}, g^*, \mathcal{D}, \mathcal{T}, \beta, \beta^*)$ *be as in Definition 23. Furthermore, let* $\beta \leq \beta^*$, $g^* \neq 0$, *and* $\mathcal{D}$ *be such that* $H_\infty(\mathcal{D}_{g^*, \mathbf{A}, \mathbf{t}, \mathbf{v}}) \geq \lambda$ *for all* $(\mathbf{A}, \mathbf{t}, \mathbf{v})$. *Define* $\hat{\mathsf{pp}} = (\mathcal{R}_q, \eta, \ell, w, \mathcal{G} \cup \{g^*\}, 0, \hat{\mathcal{D}}, \mathcal{T}, \beta, \hat{\beta}^*)$ *where* $\hat{\mathcal{D}} = \mathcal{D} \cup \{\, \mathcal{D}_{g^*, \mathbf{A}, \mathbf{t}, \mathbf{v}} \,\}_{\mathbf{A}, \mathbf{t}, \mathbf{v}}$ *and* $\hat{\beta}^* = 2\, \gamma_{\mathcal{R}} \cdot (\beta^*)^2$. *For any PPT adversary* $\mathcal{A}$ *against* k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ *there exists a PPT adversary* $\mathcal{A}'$ *against* k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ *with*

$$\mathsf{Adv}^{\text{k-m-isis}}_{\hat{\mathsf{pp}}, \mathcal{A}'}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp}, \mathcal{A}}(\lambda).$$

Proof. Upon receiving a k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ instance $\Big(\mathbf{A}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G} \cup \{g^*\}}\Big)$, $\mathcal{A}'$ runs $\mathcal{A}$ on the k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance $\Big(\mathbf{A}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G}}\Big)$ and receives from it a vector $(s^*, \mathbf{u}'_{g^*})$.

Our algorithm $\mathcal{A}'$ then outputs $(1, \mathbf{u}'_{g^*} - s^* \cdot \mathbf{u}_{g^*})$. We argue that if $(s^*, \mathbf{u}'_{g^*})$ is a valid solution to the k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance then $(1, \mathbf{u}'_{g^*} - s^* \cdot \mathbf{u}_{g^*})$ is a valid solution to the k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ instance with non-negligible probability.

Clearly, the k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance given to $\mathcal{A}$ is well-distributed. By our assumption on $\mathcal{A}$, with non-negligible probability, it holds that $\mathbf{A} \cdot \mathbf{u}'_{g^*} \equiv s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} \bmod q$, $0 < \|s^*\| \leq \beta^*$, and $\left\|\mathbf{u}'_{g^*}\right\| \leq \beta^*$. Since $\mathbf{A} \cdot \mathbf{u}_{g^*} \equiv g^*(\mathbf{v}) \cdot \mathbf{t} \bmod q$, we have $\mathbf{A} \cdot (\mathbf{u}'_{g^*} - s^* \cdot \mathbf{u}_{g^*}) \equiv \mathbf{0} \bmod q$. Furthermore, by our assumption on $\mathcal{D}_{g^*,\mathbf{A},\mathbf{t},\mathbf{v}}$, we have $\|\mathbf{u}_{g^*}\| \leq \beta \leq \beta^*$. We therefore have $\left\|\mathbf{u}'_{g^*} - s^* \cdot \mathbf{u}_{g^*}\right\| \leq 2\,\gamma_{\mathcal{R}} \cdot (\beta^*)^2 = \hat{\beta}^*$. It remains to argue that $\mathbf{u}'_{g^*} - s^* \cdot \mathbf{u}_{g^*} \neq \mathbf{0}$ with non-negligible probability, which is immediate from $H_\infty\left(\mathcal{D}_{g^*,\mathbf{A},\mathbf{v}}\right) \geq \lambda$. □

$\boldsymbol{(\mathcal{G}, g^*) \geq (r \cdot \mathcal{G}, r \cdot g^*).}$ We show that the k-M-ISIS assumption is invariant under multiplication by any non-zero Laurent monomial $r(\mathbf{X})$.

Lemma 7. *Let the parameters* $\mathsf{pp} = (\mathcal{R}_q, \eta, \ell, w, \mathcal{G}, g^*, \mathcal{D}, \mathcal{T}, \beta, \beta^*)$ *be as in Definition 23. Let* $r(\mathbf{X}) \in \mathcal{R}(\mathbf{X})$ *be a non-zero Laurent monomial and denote* $r \cdot \mathcal{G} := \{\, r \cdot g : g \in \mathcal{G} \,\}$*. Define* $\hat{\mathsf{pp}} = (\mathcal{R}_q, \eta, \ell, w, r \cdot \mathcal{G}, r \cdot g^*, \mathcal{D}, \mathcal{T}, \beta, \beta^*)$*. For any PPT adversary* $\mathcal{A}$ *against* k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ *there exists a PPT adversary* $\mathcal{A}'$ *against* k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ *with*

$$\mathsf{Adv}^{\text{k-m-isis}}_{\hat{\mathsf{pp}},\mathcal{A}'}(\lambda) \geq \frac{1}{\mathsf{poly}(\lambda)} \cdot \mathsf{Adv}^{\text{k-m-isis}}_{\mathsf{pp},\mathcal{A}}(\lambda).$$

Proof. Upon receiving a k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ instance $\Big(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G}}\Big)$, $\mathcal{A}'$ sets $\mathbf{B} := r(\mathbf{v}) \cdot \mathbf{A}$, which is well-defined since $\mathbf{v} \in (\mathcal{R}_q^\times)^w$. It then runs $\mathcal{A}$ on the k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance $\Big(\mathbf{B}, \mathbf{t}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G}}\Big)$ and receives from it a tuple $(s^*, \mathbf{u}_{g^*})$. Our algorithm $\mathcal{A}'$ then outputs $(s^*, \mathbf{u}_{g^*})$. We argue that if $(s^*, \mathbf{u}_{g^*})$ is a valid solution to the k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance $\Big(\mathbf{B}, \mathbf{t}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G}}\Big)$, then it is also a valid solution to the k-M-$\mathsf{ISIS}_{\hat{\mathsf{pp}}}$ instance $\Big(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{\,\mathbf{u}_g\,\}_{g \in \mathcal{G}}\Big)$.

Note that $r(\mathbf{v}) \in \mathcal{R}_q^\times$ and $\mathbf{A}$ is uniformly random over $\mathcal{R}_q^{\eta \times \ell}$. Therefore $\mathbf{A}$ is also uniformly random over $\mathcal{R}_q^{\eta \times \ell}$. Next, note that $\mathbf{B} \cdot \mathbf{u}_g = r(\mathbf{v}) \cdot \mathbf{A} \cdot \mathbf{u}_g \equiv (r \cdot g)(\mathbf{v}) \cdot \mathbf{t} \bmod q$. The k-M-$\mathsf{ISIS}_{\mathsf{pp}}$ instance given to $\mathcal{A}$ is therefore well-distributed.

By our assumption on $\mathcal{A}$, with non-negligible probability, it holds that $\mathbf{B} \cdot \mathbf{u}_{g^*} \equiv s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} \bmod q$, $0 < \|s^*\| \leq \beta^*$, $\|\mathbf{u}_{g^*}\| \leq \beta^*$, and $(g^*, \mathbf{u}_{g^*}) \neq (0, \mathbf{0})$. The first equation implies

$$\begin{aligned} s^* \cdot (r \cdot g^*)(\mathbf{v}) \cdot \mathbf{t} &\equiv r(\mathbf{v}) \cdot \mathbf{A} \cdot \mathbf{u}_{g^*} \bmod q \\ &\equiv \mathbf{B} \cdot \mathbf{u}_{g^*} \bmod q. \end{aligned}$$

□

4.1 Attacks

Our first attack simply solves M-ISIS (more precisely ISIS). It thus simply ignores the algebraic dependencies among the $\{\, g(\cdot) \,\}_{g \in \mathcal{G}}$. Our further attacks attempt to find short linear combinations among the $\{\, g(\mathbf{v}) \,\}_{g \in \mathcal{G}}$.

Direct SIS Attack. First, we can reduce the problem of finding $\mathbf{u}_{g^*}$ s.t. $\mathbf{A} \cdot \mathbf{u}_{g^*} \equiv s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} \bmod q$ to finding a $\mathbf{A}' \cdot \mathbf{u}'_{g^*} \equiv 0 \bmod q$ with $\mathbf{A}' := (\mathbf{A}, -g^*(\mathbf{v}) \cdot \mathbf{t})$. Then the last entry of $\mathbf{u}'_{g^*}$ becomes s^*. The analysis here is completely standard.

We will write this as $\mathbf{A} \cdot \mathbf{u} \equiv 0 \bmod q$ with $\mathbf{A} \in \mathbb{Z}^{n \cdot \eta \times (n \cdot \eta \cdot (\ell+1))}$. This task is equivalent to finding a short vector in $\Lambda(\mathbf{L})$ with $\mathbf{A} \cdot \mathbf{L} \equiv 0 \bmod q$ and $\mathbf{L} \in \mathbb{Z}_q^{(n \cdot \eta \cdot (\ell+1)) \times (n \cdot \eta \cdot \ell)}$. Thus, we are trying to find a short vector in a $d \leq n \cdot \eta \cdot (\ell + 1)$ dimensional lattice with volume $\mathrm{Vol}(\Lambda) = q^{n \cdot \eta}$. Our problem formulation is for the infinity norm but lattice reduction naturally considers the ℓ_2 norm. We thus consider it a win if lattice reduction finds a vector of norm $\sqrt{d} \cdot \beta^*$, which is generous to the attacker. That is, we are trying to establish the root-Hermite factor δ s.t.

$$\sqrt{d} \cdot \beta^* \approx \delta^{d-1} \cdot \mathrm{Vol}(\Lambda)^{1/d}.$$

The minimum of the right hand side attained at $d \approx \sqrt{n \cdot \eta \cdot \log q / \log \delta}$.[14] Overall, we obtain a vector of norm $2^{2 \cdot \sqrt{n \cdot \eta \cdot \log(\delta) \cdot \log(q)} - \log(\delta)}$.

A Solution in $\mathsf{Span}_{\mathcal{R}}(\{\, \mathbf{u}_g \,\}_{g \in \mathcal{G}})$. We note that $\mathbf{v} \leftarrow\!\$\, \mathcal{R}_q^w$ is critical for security. If all v_i are small then e.g. $v_0/v_1 \cdot \mathbf{t} \equiv \mathbf{A} \cdot \mathbf{u}_{X_0/X_1}$ and $v_2/v_1 \cdot \mathbf{t} \equiv \mathbf{A} \cdot \mathbf{u}_{X_2/X_1}$ (which corresponds to the form of $\mathcal{G}$ which we will consider below) allows to compute $\mathbf{A} \cdot \left(v_2 \cdot \mathbf{u}_{X_0/X_1} - v_0 \cdot \mathbf{u}_{X_2/X_1}\right) \equiv 0 \bmod q$. If $k > \ell$ linearly independent such preimages of zero can be constructed then this constitutes a trapdoor for $\mathbf{A}$ and solves k-M-ISIS.

More generally and for $\mathbf{v} \leftarrow\!\$\, \mathcal{R}_q^w$, we may attempt to find a short $\mathbf{z} = (z_{g_0}, \ldots, z_{g_{k-1}})$ s.t.

$$\langle (g_0(\mathbf{v}), \ldots, g_{k-1}(\mathbf{v})), \mathbf{z} \rangle \equiv s^* \cdot g^*(\mathbf{v}) \bmod q,$$

for $g_i \in \mathcal{G}$. We then compute $\mathbf{u}_{g^*} = \sum_{g \in \mathcal{G}} z_g \cdot \mathbf{u}_g$ which gives

$$\mathbf{A} \cdot \mathbf{u}_{g^*} = \mathbf{A} \cdot \left(\sum_{g \in \mathcal{G}} z_g \cdot \mathbf{u}_g \right) = \sum_{g \in \mathcal{G}} z_g \cdot \mathbf{A} \cdot \mathbf{u}_i = \sum_{g \in \mathcal{G}} z_g \cdot g(\mathbf{v}) \cdot \mathbf{t} = s^* \cdot g^*(\mathbf{v}) \cdot \mathbf{t} \bmod q.$$

Write $\mathbf{G} = [\mathrm{rot}(g_0(\mathbf{v})) | \ldots | \mathrm{rot}(g_{k-1}(\mathbf{v})) | \mathrm{rot}(g^*(\mathbf{v}))] \in \mathbb{Z}_q^{n \times (n \cdot (k+1))}$. As above, we cost finding a short vector in $\Lambda(\mathbf{W})$ where $\mathbf{G} \cdot \mathbf{W} \equiv 0 \bmod q$. The analysis proceeds exactly as above.

One the one hand, the final solution will have a larger expected norm $\leq \sqrt{k} \cdot \gamma_R \cdot \max_{g \in \mathcal{G}}(\beta_g) \cdot \beta_{\mathbf{z}}$ when $\|\mathbf{z}\| \leq \beta_{\mathbf{z}}$: we are adding up k terms, each being the product of two elements, and consider the expected norm. On the other hand, note that this attack is independent of η. This implies that while k-M-ISIS is at least as hard as k-R-ISIS it cannot, in general, be strictly harder.

A Solution in $\mathsf{Span}_{\mathcal{R}_q}(\{\, \mathbf{u}_g \,\}_{g \in \mathcal{G}})$. We can generalise the previous approach to finding any, i.e. not necessarily short, $\mathbf{z} \in \mathcal{R}_q^k$ s.t.

$$\sum z_i \cdot g_i(\mathbf{v}) \equiv s^* \cdot g^*(\mathbf{v}) \bmod q \quad \text{ and } \quad \sum z_i \cdot \mathbf{u}_{g_i} = \mathbf{u}_{\mathbf{z}} \text{ with } \|\mathbf{u}_{\mathbf{z}}\| \leq \beta^*.$$

$$\begin{aligned} \text{Write } \mathbf{G} &= \begin{pmatrix} \mathrm{rot}(g_0(\mathbf{v})) & \cdots & \mathrm{rot}(g_{k-1}(\mathbf{v})) \end{pmatrix} && \in \mathbb{Z}_q^{n \times (n \cdot k)} \\ \mathbf{U} &= \begin{pmatrix} \mathrm{rot}\left((\mathbf{u}_{g_0})_0\right) & \cdots & \mathrm{rot}\left((\mathbf{u}_{g_{k-1}})_0\right) \\ & \ddots & \\ \mathrm{rot}\left((\mathbf{u}_{g_0})_{\ell-1}\right) & \cdots & \mathrm{rot}\left((\mathbf{u}_{g_{k-1}})_{\ell-1}\right) \end{pmatrix} && \in \mathbb{Z}^{(n \cdot \ell) \times (n \cdot k)} \end{aligned}$$

and consider the lattice spanned by the columns of

$$\mathbf{S} := \begin{pmatrix} \tau & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{g}^* & q\,\mathbf{I}_n & \mathbf{0} & \mathbf{G} \\ \mathbf{0} & \mathbf{0} & q\,\mathbf{I}_{n \cdot \ell} & \mathbf{U} \end{pmatrix}$$

where τ is some “embedding factor” optimised by the solving algorithm (the reader may simply assume $\tau = 1$.) Then $\Lambda(\mathbf{S})$ contains a short vector $(-\tau \cdot s^*, \mathbf{0}^{\mathrm{T}}, \mathbf{u}_{\mathbf{z}}^{\mathrm{T}})^{\mathrm{T}}$. Computing the column Hermite normal form of $\mathbf{S}$ produces a basis in $\mathbb{Z}^{d \times d}$ with $d := \ell \cdot (n+1)$. Assuming full row rank of $[\mathbf{G}^{\mathrm{T}}, \mathbf{U}^{\mathrm{T}}] \bmod q$, the determinant of $\Lambda(\mathbf{S})$ is q^{T} with $t := (\ell - \min(k, \ell) + 1) \cdot n$.

Thus, by the Gaussian heuristic, i.e. assume the lattice generated behaves like a random lattice, we expect a shortest vector to have norm $\approx \sqrt{d/2\,\pi\,e} \cdot q^{t/d}$ and lattice reduction with root Hermite factor δ to find a vector of norm $\delta^{d-1} \cdot q^{t/d}$. This is minimised when $t = \varphi(m)$, i.e. when $k = \ell$.

[14] The minimum is $d = \sqrt{n \log q / \log \delta}$ for $\beta_\ell = \delta^d \cdot \mathrm{Vol}(\Lambda)^{1/d}$ which is what the literature typically considers. However, normalising δ by $d-1$ instead of d makes sense from the analysis of lattice algorithms. Note that $d \geq 1000$ and $\delta < 1.02$ so that discrepancy is tiny.

4.2 Knowledge Assumptions.

Finally, we consider the knowledge assumptions.

On the one hand, we evaluate these attack strategies with respect to the knowledge assumption. An adversary that succeeds with the direct SIS strategy breaks our knowledge assumption while also breaking the M-ISIS assumption. The second approach – finding a solution in $\mathsf{Span}_{\mathcal{R}}(\{\mathbf{u}_g\}_{g\in\mathcal{G}})$ immediately implies the extractor in Definition 26 by computing x_g directly. The third attack approach – finding a solution in $\mathsf{Span}_{\mathcal{R}_q}(\{\mathbf{u}_g\}_{g\in\mathcal{G}})$ – initially seems most promising to invalidate our knowledge assumption by generalising the attack to find large x_g such that $\mathbf{u}^* := \sum_{g\in\mathcal{G}} x_g \cdot \mathbf{u}_g$ is small. While finding such x_g given $\mathbf{u}^*$ and $\mathbf{u}_g$ is easy, finding a suitable target $\mathbf{u}^*$, i.e. one satisfying $c \cdot \mathbf{t} \equiv \mathbf{A} \cdot \mathbf{u}^* \bmod q$, seems hard, as outlined above.

On the other hand, we highlight a gap between the knowledge and "plain" k-R-ISIS assumption pair and the knowledge k-M-ISIS and plain k-M-ISIS pair. For k-R-ISIS, we can wlog pick $t = 1$ for the plain version but must pick $t \neq 1$ such that $1/|\langle t\rangle| = \mathsf{negl}(\lambda)$ and $|\langle t\rangle|/|\mathcal{R}_q| = \mathsf{negl}(\lambda)$ for the knowledge version. For k-M-ISIS, we may pick $\mathbf{t} = (1, 0, \ldots, 0)^T$ in both cases.

This distinction is not just aesthetic. For plain k-R-ISIS, the attack strategies we are aware of rely on finding short vectors in some modules of rank > 1. In contrast, in what follows, we sketch an attack on knowledge k-R-ISIS which relies on finding short vectors in ideals rather than in modules. Consider $g^* \equiv 0$ and consider $\mathcal{I} := \{s \in \mathcal{R} \mid s \cdot t \equiv 0 \bmod q\}$. Note that $\mathcal{I}$ is an ideal in $\mathcal{R}$ and that finding a sufficiently short element $s' \neq 0$ in $\mathcal{I}$ is a solution for k-R-ISIS with target $g^* \equiv 0$. Pick any of the provided $\mathbf{u}_i$ and return $s' \cdot \mathbf{u}_i$. Since it holds that $g_i(\mathbf{v}) \cdot t \equiv \langle \mathbf{a}, \mathbf{u}_i\rangle \bmod q$ and $s' \cdot t \equiv 0 \bmod q$ we have that $0 \equiv \langle \mathbf{a}, s' \cdot \mathbf{u}_i\rangle \bmod q$. Finding such an s' efficiently breaks the knowledge assumption since w.h.p. $s' \cdot g_i(\mathbf{v}) \neq 0 \bmod q$. This motivates our restriction in Definition 27.

To understand what choices of $(\mathcal{R}_q, \mathcal{T})$ may provide secure instantiations, we first note that a series of works [Ber14,CGS14,CDPR16,CDW17,PHS19,DPW19] reports quantum algorithms for finding short vectors in ideal latices that beat known algorithms for general lattices. In particular, finding vectors of norm $2^{\tilde{O}(\sqrt{m})}$ in an ideal lattice of dimension m can be done in quantum polynomial time. Thus, when $\alpha^* \approx 2^{\tilde{O}(\sqrt{m})}$ then knowledge k-R-ISIS is easy for a quantum adversary.[15]

Moreover, for some choices of ideals in a power-of-two cyclotomic ring $\mathcal{R}$ it has been shown in [PXWC21] that finding short vectors is easy. In our case, by construction, the ideals $\mathcal{S} \subseteq \mathcal{R}$ have algebraic norm $N(\mathcal{S}) = q^i$ for some integer i. The headline result of [PXWC21] thus implies that knowledge k-R-ISIS is easy when m is a power of two, $q \equiv \pm 3 \bmod 8$ and $\alpha^* \geq \sqrt{q/(m/2)}$.[16] On the other hand, note that the attack does not apply, for example, when $\mathcal{R}_q$ splits completely into $\phi(m)$ fields, e.g. when $q \equiv 1 \bmod m$ where m is a power of two. Then, sampling t as mentioned in the footnote to Definition 27, i.e. picking half the CRT components zero and the other half non-zero, produces ideals where, to the best of our knowledge, (approximate) ideal-SVP is not easier than the general case discussed above.

Yet, given that the status of ideal-SVP in $\mathcal{R}_q$ is still in flux, the above highlights that knowledge k-R-ISIS is a more risky assumption than (knowledge) k-M-ISIS or plain k-R-ISIS. In particular, we note that $\mathcal{I} := \{s \in \mathcal{R} \mid s \cdot t \equiv 0 \bmod q\} = \{0\}$ for plain k-R-ISIS when $t = 1$, and $\mathcal{I} := \{s \in \mathcal{R} \mid s \cdot \mathbf{t} \equiv \mathbf{0} \bmod q\} = \{0\}$ for both plain and knowledge k-M-ISIS when $\mathbf{t} := (1, 0, \ldots, 0)^T$, i.e. this line of attack is ruled out.

5 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.

5.1 Construction

A formal description of our VC construction is in Fig. 3 where important parameters and shorthands are listed and explained in Table 1.

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

[15] These quantum improvements may only matter, though, in practice for large values of m [DPW19].

[16] We consider the infinity norm but [PXWC21] considers the Euclidean norm.

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 3)	Norm bound for honestly generated opening proof $\mathbf{u}_i$
$\delta_0' \in \mathbb{R}$	$\mathsf{poly}(\lambda, s, w, t)$ (Theorem 5)	Norm bound for opening proof $\mathbf{u}_0'$ generated by knowledge extractor
$\delta_p \in \mathbb{R}$	$(s + w + d)^d \cdot \alpha^{d+1} \cdot \gamma_{\mathcal{R}}^d$	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
$\delta_p' \in \mathbb{R}$	$(s + w + d)^d \cdot \alpha^{d+1} \cdot (w \cdot \beta \cdot \gamma_{\mathcal{R}}^2)^d$	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 $w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}}$
$p \in \mathbb{N}$	$\geq \delta_p \cdot n \cdot \log n$	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 $\mathbf{A}_i$
$\mathcal{X} \subseteq \mathcal{R}$	$\{ x \in \mathcal{R} : \lVert x \rVert \leq \alpha \}$	$\mathcal{R}$ 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 $\mathcal{X}$
$\mathcal{E}_k \subseteq \mathbb{N}_0^w$	$\{ \mathbf{e} \in \mathbb{N}_0^w : \lVert \mathbf{e} \rVert_1 = k \}$	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 23)
$f_{i,\mathbf{e}}$		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

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 $c \coloneqq \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$.

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 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 $\|f(\mathbf{z}, \mathbf{x}) - y(\mathbf{z})\| \leq (s + w + d)^d \cdot \alpha^{d+1} \cdot \gamma_{\mathcal{R}}^d$. More generally, for arbitrary $\mathbf{x} \in \mathcal{R}^w$, we get $\|f(\mathbf{z}, \mathbf{x}) - y(\mathbf{z})\| \leq (s + w + d)^d \cdot \alpha \cdot \|\mathbf{x}\|^d \cdot \gamma_{\mathcal{R}}^d$. As mentioned in Section 2.4, 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.[17]

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

```
$\mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$
$\mathbf{v} \leftarrow\!\!\$ \, (\mathcal{R}_q^\times)^w$
$\mathbf{h} \leftarrow\!\!\$ \, \mathcal{R}_p^t$
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 $\mathsf{pp} := \begin{pmatrix} \mathbf{A}_0, \, \mathbf{t}_0, \, (\mathbf{u}_{0,g})_{g \in \mathcal{G}_0}, \\ \mathbf{A}_1, \, \mathbf{t}_1, \, (\mathbf{u}_{1,g})_{g \in \mathcal{G}_1}, \\ \mathbf{v}, \quad \mathbf{h} \end{pmatrix}$
```

```
$\mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$
$\mathbf{u}_0 := \sum_{i \in \mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e} \in \mathcal{E}_k} h_i \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{u}_{0,\mathbf{e}}$
return $\pi := (\mathbf{u}_0, \mathbf{u}_1)$
```

```
$\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi)$
$b_0 := \left( \mathbf{A}_0 \cdot \mathbf{u}_0 \overset{?}{\equiv} \hat{f}_y(\mathbf{z}, c) \cdot \mathbf{t}_0 \bmod q \right)$
$b_1 := \left( \mathbf{A}_1 \cdot \mathbf{u}_1 \overset{?}{\equiv} c \cdot \mathbf{t}_1 \bmod q \right)$
$b_2 := \left( \|\mathbf{u}_0\| \overset{?}{\leq} \delta_0 \right); b_3 := \left( \|\mathbf{u}_1\| \overset{?}{\leq} \delta_1 \right)$
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 $(c, \mathsf{aux})$
```

```
$\mathsf{PreVerify}(\mathsf{pp}, (f, y))$
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. 3. Our VC Construction.

Remark 5 (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 $O(\varDelta)$, where $\varDelta$ 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 $\binom{w}{k} - \binom{w-\varDelta}{k} = O(\varDelta^k)$ 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 $O(w^d \cdot \varDelta^d)$. 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.

Theorem 3. *For $d = O(1)$, $\ell_0 := \ell_1 := \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$,*

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

our VC construction in Fig. 3 is correct.

Proof. The multinomial theorem states that $(z_0 + \cdots + z_{w-1})^k = \sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}} \cdot \mathbf{z}^{\mathbf{e}}$. Let $(c, \mathsf{aux}) = \mathsf{Com}(\mathsf{pp}, \mathbf{x})$ so that $c = \langle \mathbf{v}, \mathbf{x} \rangle = v_0 \cdot x_0 + \cdots + v_{w-1} \cdot x_{w-1}$. Substituting $\mathbf{z} = (v_0 \cdot x_0, \ldots, v_{w-1} \cdot x_{w-1})$ we have $c^k = \sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}} \cdot \mathbf{v}^{\mathbf{e}} \cdot \mathbf{x}^{\mathbf{e}}$.

Fix any $f \in \mathcal{F}_{s,w,t}$ and any $y \in \mathcal{Y}_{s,t}$. Write $f(\mathbf{Z}, \mathbf{X}) = (\sum_{k=1}^{d} \sum_{\mathbf{e} \in \mathcal{E}_k} f_{i,\mathbf{e}}(\mathbf{Z}) \cdot \mathbf{X}^{\mathbf{e}})_{i \in \mathbb{Z}_t}$ and $y(\mathbf{Z}) = (y_i(\mathbf{Z}))_{i \in \mathbb{Z}_t}$. For $i \in \mathbb{Z}_t$, let

$$\bar{f}_{i,k}(\mathbf{Z}, C) \coloneqq \sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}}^{-1} \cdot f_{i,\mathbf{e}}(\mathbf{Z}) \cdot \mathbf{v}^{-\mathbf{e}} \cdot C^k$$

so that $\hat{f}_y(\mathbf{Z}, C) = \sum_{i \in \mathbb{Z}_t} h_i \cdot d! \cdot (\sum_{k=1}^{d} \hat{f}_{i,k}(\mathbf{Z}, C) - y_i(\mathbf{Z}))$.

For any $(\mathbf{z}, \mathbf{x}) \in \mathcal{X}^s \times \mathcal{X}^w$ and any $(c, \mathsf{aux}) \in \mathsf{Com}(\mathsf{pp}, \mathbf{x})$, we observe that

$$\begin{aligned}
\bar{f}_{i,k}(\mathbf{z}, c) &= \sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}}^{-1} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{v}^{-\mathbf{e}} \cdot c^k \\
&= \left(\sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}}^{-1} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{v}^{-\mathbf{e}} \right) \cdot \left(\sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}} \cdot \mathbf{x}^{\mathbf{e}} \cdot \mathbf{v}^{\mathbf{e}} \right) \\
&= \sum_{\mathbf{e}, \mathbf{e}' \in \mathcal{E}_k} \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{v}^{\mathbf{e}' - \mathbf{e}} \\
&= \sum_{\mathbf{e} \in \mathcal{E}_k} f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}} + \sum_{\mathbf{e}, \mathbf{e}' \in \mathcal{E}_k : \mathbf{e} \neq \mathbf{e}'} \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{v}^{\mathbf{e}' - \mathbf{e}}.
\end{aligned}$$

Suppose $y(\mathbf{z}) = f(\mathbf{z}, \mathbf{x})$. We have

$$\begin{aligned}
\hat{f}_y(\mathbf{z}, c) &= \sum_{i \in \mathbb{Z}_t} h_i \cdot d! \cdot \left(\sum_{k=1}^{d} \hat{f}_{i,k}(\mathbf{z}, c) - y_i(\mathbf{z}) \right) \\
&= \sum_{i \in \mathbb{Z}_t} h_i \cdot d! \cdot \left(\sum_{k=1}^{d} \sum_{\mathbf{e} \in \mathcal{E}_k} f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}} + \sum_{k=1}^{d} \sum_{\mathbf{e}, \mathbf{e}' \in \mathcal{E}_k : \mathbf{e} \neq \mathbf{e}'} \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{v}^{\mathbf{e}' - \mathbf{e}} - y_i(\mathbf{z}) \right) \\
&= \sum_{i \in \mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e}, \mathbf{e}' \in \mathcal{E}_k : \mathbf{e} \neq \mathbf{e}'} h_i \cdot d! \cdot \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{v}^{\mathbf{e}' - \mathbf{e}}.
\end{aligned}$$

Let $(\mathbf{u}_0, \mathbf{u}_1) \in \mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$. We have

$$\begin{aligned}
\mathbf{u}_0 &= \sum_{i \in \mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e} \neq \mathbf{e}' \in \mathcal{E}_k} h_i \cdot d! \cdot \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{u}_{X^{\mathbf{e}' - \mathbf{e}}} \text{ and} \\
\mathbf{u}_1 &= \sum_{X_i \in \mathcal{G}_1} x_i \cdot \mathbf{u}_{1, X_i}.
\end{aligned}$$

We check that the following indeed hold:

$$\begin{aligned}
\mathbf{A}_0 \cdot \mathbf{u}_0 &= \mathbf{A}_0 \cdot \left(\sum_{i \in \mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e} \neq \mathbf{e}' \in \mathcal{E}_k} h_i \cdot d! \cdot \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{u}_{X^{\mathbf{e}' - \mathbf{e}}} \right) \\
&\equiv \sum_{i \in \mathbb{Z}_t} \sum_{k=1}^{d} \sum_{\mathbf{e} \neq \mathbf{e}' \in \mathcal{E}_k} h_i \cdot d! \cdot \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot f_{i,\mathbf{e}}(\mathbf{z}) \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{v}^{\mathbf{e}' - \mathbf{e}} \cdot \mathbf{t}_0 \bmod q \\
&\equiv \hat{f}(\mathbf{z}, c) \cdot \mathbf{t}_0 \bmod q, \\
\mathbf{A}_1 \cdot \mathbf{u}_1 &= \mathbf{A}_1 \cdot \left(\sum_{X_i \in \mathcal{G}_1} x_i \cdot \mathbf{u}_{1, X_i} \right) \equiv \sum_{X_i \in \mathcal{G}_1} x_i \cdot v_i \cdot \mathbf{t}_1 \bmod q \equiv c \cdot \mathbf{t}_1 \bmod q.
\end{aligned}$$

We next analyse the norm of $\mathbf{u}_0$ and $\mathbf{u}_1$. Examining the form $\mathbf{u}_0$ and writing down an upper bound of the norm of each term, we have

$$\mathbf{u}_0 = \sum_{\underbrace{i\in\mathbb{Z}_t}_{t}} \sum_{\underbrace{k=1}_{d}}^{d} \sum_{\underbrace{\mathbf{e}\neq\mathbf{e}'\in\mathcal{E}_k}_{\binom{w+d}{d}^2}} \underbrace{h_i}_{p/2} \underbrace{d! \cdot \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}}}_{(d!)^2} \underbrace{f_{i,\mathbf{e}}(\mathbf{z})}_{(d+1)\cdot\binom{s+d}{d}\cdot\gamma_{\mathcal{R}}^{d}\cdot\alpha^{d+1}} \underbrace{\mathbf{x}^{\mathbf{e}'}}_{\gamma_{\mathcal{R}}^{d-1}\cdot\alpha^{d}} \underbrace{\mathbf{u}_{X^{\mathbf{e}'-\mathbf{e}}}}_{\beta} .$$

Using $\binom{w+d}{d}^2 \leq \frac{(w+d)^{2d}}{(d!)^2}$, $\binom{s+d}{d} \leq \frac{(s+d)^d}{d!}$, and $\frac{d+1}{(d-1)!} \leq 3$, and taking into account the expansion factor $\gamma_{\mathcal{R}}^3$ for multiplying 4 $\mathcal{R}$ elements, we have

$$\|\mathbf{u}_0\| \leq 2 \cdot p \cdot t \cdot (s+d)^d \cdot (w+d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} \leq \delta_0.$$

Similarly, examining $\mathbf{u}_1$, we have

$$\mathbf{u}_1 = \sum_{\underbrace{X_i\in\mathcal{G}_1}_{w}} \underbrace{x_i}_{\alpha} \underbrace{\mathbf{u}_{1,X_i}}_{\beta} .$$

Accounting for the expansion factor $\gamma_{\mathcal{R}}$ for multiplying 2 $\mathcal{R}$ elements, we have $\|\mathbf{u}_1\| \leq w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}} \leq \delta_1$. □

Theorem 4. *Let $\mathcal{X}^* := \{\, x \in \mathcal{R} : \|x\| \leq \alpha^* \,\}$. Our VC construction for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is $\mathcal{X}^*$-extractable if*

$$\begin{aligned}
&\ell_i \geq \mathsf{lhl}(\mathcal{R}, \eta_i, q, \beta) \text{ for } i \in \{0,1\},\\
&\alpha^* \geq \beta_1^* \geq \delta_1,\\
&\beta_0^* \geq 2 \cdot \max\{\, \delta_0, 2 \cdot p \cdot t \cdot (s+d)^d \cdot (w+d)^{2d} \cdot \alpha^{d+1} \cdot (\alpha^*)^d \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} \,\},\\
&\beta_p^* \geq 2 \cdot \max\{\, \delta_p, (s+w+d)^d \cdot \alpha \cdot (\alpha^*)^d \cdot \gamma_{\mathcal{R}}^{d} \,\},
\end{aligned}$$

and the k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta_0,\ell_0,w,\mathcal{G}_0,1,\mathcal{D}_0,\mathcal{T}_0,\beta,\beta_0^*}$ assumption, the knowledge k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta_1,\ell_1,w,\mathcal{G}_1,\mathcal{D}_1,\mathcal{T}_1,\alpha^*,\beta,\beta_1^*}$ assumption, and the R-$\mathsf{SIS}_{\mathcal{R}_p,t,\beta_p^*}$ 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

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

Proof. Suppose $\mathcal{A}$ is a PPT adversary which, on input honestly generated pp and some randomness, outputs $(f, y, \mathbf{z}, c, \pi)$. We construct an extractor $\mathcal{E}_{\mathcal{A}}$ which, on input pp and the same randomness given to $\mathcal{A}$, outputs $\mathbf{x}$.

For the sake of clarity of exposition, let us denote the public parameters pp of the vector commitment scheme as

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

where $\mathsf{pp}_0(\mathbf{v})$ and $\mathsf{pp}_1(\mathbf{v})$ are correlated in that they share the same $\mathbf{v}$.

We define an algorithm $\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]$ which has oracle access to $\mathcal{A}$ and is parameterised by an instance of the VC public parameters $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$. Our algorithm $\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]$ takes as input some $\mathsf{pp}_1'(\mathbf{v}') = (\mathbf{A}_1', \mathbf{t}_1', (\mathbf{u}_{1,g}')_{g\in\mathcal{G}_1}, \mathbf{v}')$ and some randomness $r_{\mathcal{A}}$. If $\mathbf{v}' \neq \mathbf{v}$, $\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]$ outputs some arbitrary $(c, \mathbf{u}_1)$. Otherwise, $\mathbf{v}' = \mathbf{v}$, and $\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]$ runs $\mathcal{A}$ on $(\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1'(\mathbf{v}), \mathbf{h})$ and the given randomness $r_{\mathcal{A}}$, and obtains $(f, y, \mathbf{z}, c, \pi)$. It parses π as $(\mathbf{u}_0, \mathbf{u}_1)$ and outputs $(c, \mathbf{u}_1)$.

Let $\mathcal{E}_{\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]}^{k\text{-}M\text{-}\mathsf{ISIS}}$ be a PPT extractor whose existence is guaranteed by the knowledge k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta_1,\ell_1,w,\mathcal{G}_1,\mathcal{D}_1,\mathcal{T}_1,\alpha^*,\beta,\beta_1^*}$ assumption. We construct our extractor $\mathcal{E}_{\mathcal{A}}$ as follows.

Our extractor $\mathcal{E}_{\mathcal{A}}$ takes as input some public parameters pp and some randomness $r_{\mathcal{A}}$. Parse $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$. It runs $\mathcal{E}^{k\text{-}M\text{-}\mathsf{ISIS}}_{\mathcal{B}^{\mathcal{A}}[\mathsf{pp}]}$ on input $\mathsf{pp}_1(\mathbf{v})$ and the given randomness $r_{\mathcal{A}}$, and obtains from them a vector $\mathbf{x}$. Finally, $\mathcal{E}_{\mathcal{A}}$ outputs $\mathbf{x}$.

We argue that for $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$, if $(f, y, \mathbf{z}, c, \pi) \leftarrow \mathcal{A}(\mathsf{pp}; r_{\mathcal{A}})$ satisfies $\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi) = 1$ with probability ρ, then the probability of $\mathcal{E}_{\mathcal{A}}(\mathsf{td}; r)$ not outputting $\mathbf{x}$ with $\|\mathbf{x}\| \leq \alpha^*$ such that $c = \mathsf{Com}(\mathsf{pp}, \mathbf{x})$ (for some aux suppressed from the output) and $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$ is at most $\kappa(\lambda, s, w, t) = \mathsf{negl}(\lambda)$, where the probabilities are taken over the randomness of Setup and that of $r_{\mathcal{A}}$.

Consider the following hybrid experiments for generating $(\mathsf{pp}, (f, y, \mathbf{z}, c, \pi), \mathbf{x})$ on input $(1^\lambda, 1^s, 1^w, 1^t; (r, r_{\mathcal{A}}))$:

Hyb_0**:** This is the "real" experiment with procedures as described above. Specifically, it runs $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t; r)$, $(f, y, \mathbf{z}, c, \pi) \leftarrow \mathcal{A}(\mathsf{pp}; r_{\mathcal{A}})$, and $\mathbf{x} \leftarrow \mathcal{E}_{\mathcal{A}}(\mathsf{pp}; r_{\mathcal{A}})$, and outputs $(\mathsf{pp}, (f, y, \mathbf{z}, c, \pi), \mathbf{x})$.

Hyb_1**:** This experiment is the same as Hyb_0 except that the $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$ passed to $\mathcal{A}$ and $\mathcal{E}_{\mathcal{A}}$ is replaced by $\mathsf{pp}' = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}'_1(\mathbf{v}), \mathbf{h})$ where $\mathsf{pp}'_1(\mathbf{v})$ is sampled as in the definition of $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\eta_1,\ell_1,w,\mathcal{G}_1,\mathcal{D}_1,\mathcal{T}_1,\alpha^*,\beta,\beta_1^*}$.

Hyb_2**:** This experiment is the same as Hyb_1 except that the $\mathsf{pp}' = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}'_1(\mathbf{v}), \mathbf{h})$ passed to $\mathcal{A}$ and $\mathcal{E}_{\mathcal{A}}$ is replaced by $\mathsf{pp}'' = (\mathsf{pp}'_0(\mathbf{v}), \mathsf{pp}'_1(\mathbf{v}), \mathbf{h})$ where $\mathsf{pp}'_0(\mathbf{v})$ is sampled as in the definition of $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\eta_0,\ell_0,w,\mathcal{G}_0,1,\mathcal{D}_0,\mathcal{T}_0,\beta,\beta_0^*}$.

By our assumption on $\mathcal{D}_0$, the distributions Hyb_0 and Hyb_1 are statistically close. Similarly, by our assumption on $\mathcal{D}_1$, the distributions Hyb_1 and Hyb_2 are statistically close. Since the distributions Hyb_0, Hyb_1, and Hyb_2 are all statistically close to each other, for any $i, j \in \mathbb{Z}_3$, if the output of Hyb_i satisfies certain properties with some probability, the output of Hyb_j also satisfies the same properties with similar probability.

The following lemma about the outputs of Hyb_1 is immediate by the knowledge $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\eta_1,\ell_1,w,\mathcal{G}_1,\mathcal{D}_1,\mathcal{T}_1,\alpha^*,\beta,\beta_1^*}$ assumption.

Lemma 8. *Let* $(\mathsf{pp}, (f, y, \mathbf{z}, c, \pi), \mathbf{x}) \leftarrow \mathsf{Hyb}_1(1^\lambda, 1^s, 1^w, 1^t; (r, r_{\mathcal{A}}))$. *Parse* $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$. *If the knowledge* $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\eta_1,\ell_1,w,\mathcal{G}_1,\mathcal{D}_1,\mathcal{T}_1,\alpha^*,\beta,\beta_1^*}$ *assumption holds, then* $c = \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$ *and* $\|\mathbf{x}\| \leq \alpha^*$ *except with negligible probability.*

The next lemma is about the outputs of Hyb_2.

Lemma 9. *Let* $(\mathsf{pp}, (f, y, \mathbf{z}, c, \pi), \mathbf{x}) \leftarrow \mathsf{Hyb}_2(1^\lambda, 1^s, 1^w, 1^t; (r, r_{\mathcal{A}}))$. *Parse* $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$. *If all of the following hold:*

- *the* $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q,\eta_0,\ell_0,w,\mathcal{G}_0,1,\mathcal{D}_0,\mathcal{T}_0,\beta,\beta_0^*}$ *assumption,*
- *the* $R\text{-}\mathsf{SIS}_{\mathcal{R}_p,t,\beta_p^*}$ *assumption,*
- $c = \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$*, and*
- $\|\mathbf{x}\| \leq \alpha^*$*,*

then $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$ *except with negligible probability.*

Proof. Parse pp to obtain $(\mathbf{A}_0, \mathbf{t}_0, \mathbf{v}, \mathbf{h})$ and parse π as $(\mathbf{u}_0, \mathbf{u}_1)$. We notice that $\mathbf{h}$ is distributed identically as $R\text{-}\mathsf{SIS}_{\mathcal{R}_p,t,\beta_p^*}$ instances. By our assumption on $\mathcal{A}$, with non-negligible probability, it holds that

$$\mathbf{A}_0 \cdot \mathbf{u}_0 \equiv \left(\hat{f}_0(\mathbf{z}, c) - d! \cdot \sum_{i \in \mathbb{Z}_t} h_i \cdot y_i(\mathbf{z}) \right) \cdot \mathbf{t}_0 \bmod q,$$

and $\|\mathbf{u}_0\| \leq \delta_0 \leq \beta_0^*/2$.

Suppose towards a contradiction that the event $f(\mathbf{z}, \mathbf{x}) = \mathbf{y}' \neq y(\mathbf{z})$ for some $\mathbf{y}'$ happens with non-negligible probability. Let $(c', \mathsf{aux}) = \mathsf{Com}(\mathsf{pp}, \mathbf{x})$. By assumption, $c' = c$. Let $(\mathbf{u}'_0, \mathbf{u}'_1) = \mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$. By a similar calculation as in the proof of correctness (Theorem 3), it holds that

$$\mathbf{A}_0 \cdot \mathbf{u}'_0 \equiv \left(\hat{f}_0(\mathbf{z}, c) - d! \cdot \sum_{i \in \mathbb{Z}_t} h_i \cdot y'_i \right) \cdot \mathbf{t}_0 \bmod q.$$

and

$$\|\mathbf{u}'_0\| \leq 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot \alpha^{d+1} \cdot (\alpha^*)^d \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} \leq \beta_0^*/2.$$

Let $\tilde{\mathbf{u}}_0 := \mathbf{u}_0 - \mathbf{u}_0'$. We have

$$\mathbf{A}_0 \cdot \tilde{\mathbf{u}}_0 \equiv d! \cdot \sum_{i \in \mathbb{Z}_t} h_i \cdot (y_i' - y_i(\mathbf{z})) \cdot \mathbf{t}_0 \bmod q.$$

and $\|\tilde{\mathbf{u}}_0\| \leq \beta_0^*$. One (or both) of the following two cases must be true: (i) $\sum_{i \in \mathbb{Z}_t} h_i \cdot (y_i' - y_i(\mathbf{z})) \equiv \mathbf{0} \bmod q$ with non-negligible probability, or (ii) $\sum_{i \in \mathbb{Z}_t} h_i \cdot (y_i' - y_i(\mathbf{z})) \not\equiv \mathbf{0} \bmod q$ with non-negligible probability.

Note that

$$\|\mathbf{y}'\| \leq (s + w + d)^d \cdot \alpha \cdot (\alpha^*)^d \cdot \gamma_{\mathcal{R}}^d \leq \beta_p^*/2$$

and $\|y(\mathbf{z})\| \leq \delta_p \leq \beta_p^*/2$ and hence $\|\mathbf{y}' - y(\mathbf{z})\| \leq \beta_p^*$. If Case (i) is true, we can construct a PPT algorithm for the $R\text{-}\mathsf{SIS}_{\mathcal{R}_p, t, \beta_p^*}$ problem which succeeds with non-negligible probability, which contradicts the $R\text{-}\mathsf{SIS}_{\mathcal{R}_p, t, \beta_p^*}$ assumption.

If Case (ii) is true, we can construct a PPT algorithm for the $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q, \eta_0, \ell_0, w, \mathcal{G}_0, 1, \mathcal{D}_0, \mathcal{T}_0, \beta, \beta_0^*}$ problem which succeeds with non-negligible probability, which contradicts the $k\text{-}M\text{-}\mathsf{ISIS}_{\mathcal{R}_q, \eta_0, \ell_0, w, \mathcal{G}_0, 1, \mathcal{D}_0, \mathcal{T}_0, \beta, \beta_0^*}$ assumption.

Since none of the two cases could be true, we must have $f(\mathbf{z}, \mathbf{x}) = \mathbf{y}' = y(\mathbf{z})$. □

Combining the two lemmas, we conclude that for $(\mathsf{pp}, (f, y, \mathbf{z}, c, \pi), \mathbf{x})$ generated by Hyb_0, where $\mathsf{pp} = (\mathsf{pp}_0(\mathbf{v}), \mathsf{pp}_1(\mathbf{v}), \mathbf{h})$, it holds that $c = \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$, $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$, and $\|\mathbf{x}\| \leq \delta_1$ except with negligible probability. □

Theorem 5. *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 3 and 4, the VC construction in Fig. 3 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,

$$\begin{aligned}
d, \eta_0, \eta_1 = O(1), \quad \beta \geq \alpha \\
\delta_0 &= 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2}, \\
\delta_0' &= 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot w^d \cdot \alpha^{2d+1} \cdot \beta^{d+1} \cdot \gamma_{\mathcal{R}}^{3d+2}, \\
\delta_1 &= w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}}, \\
\delta_p' &= (s + w + d)^d \cdot \alpha^{d+1} \cdot (w \cdot \beta \cdot \gamma_{\mathcal{R}}^2)^d \\
p &\approx \delta_p' \cdot n \cdot \log n, \quad q \approx \delta_0' \cdot n \cdot \log n, \text{ and} \\
\ell_0 = \ell_1 = \mathsf{lhl}(\mathcal{R}, 1, q, \beta) \approx 2 \log_\beta q,
\end{aligned}$$

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 $O(n \log s)$.

Proof. For the general case, we observe that a commitment $c \in \mathcal{R}_q$ is of description size $n \log q \in \mathsf{poly}(\lambda, \log s, \log w, \log t)$, and an opening proof $(\mathbf{u}_0, \mathbf{u}_1)$ is of description size $n \cdot (\ell_0 \log \delta_0 + \ell_1 \log \delta_1) \in \mathsf{poly}(\lambda, \log s, \log w, \log t)$.

For the concrete case, for honestly generated proofs, from Theorem 3, we have

$$\begin{aligned}
p &\approx \delta_p' \cdot n \cdot \log n = (s + w + d)^d \cdot \alpha^{d+1} \cdot (w \cdot \beta \cdot \gamma_{\mathcal{R}}^2)^d \cdot n \cdot \log n \\
&= O(s^{2d} \cdot \alpha^{d+1} \cdot \beta^d \cdot n^{2d+1} \cdot \log n), \\
\delta_0 &= 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} \\
&= O(s^d \cdot \alpha^{d+1} \cdot n^{d+1} \cdot \log n) \cdot O(s^{3d+1} \cdot \alpha^{2d+1} \cdot \beta \cdot n^{2d+2}) \\
&= O(s^{4d+1} \cdot \alpha^{3d+2} \cdot \beta \cdot n^{3d+3} \cdot \log n), \\
\delta_0' &= 2 \cdot p \cdot t \cdot (s + d)^d \cdot (w + d)^{2d} \cdot w^d \cdot \alpha^{2d+1} \cdot \beta^{d+1} \cdot \gamma_{\mathcal{R}}^{3d+2} \\
&= O(s^{2d} \cdot \alpha^{d+1} \cdot \beta^d \cdot n^{2d+1} \cdot \log n) \cdot O(s^{4d+1} \cdot \alpha^{2d+1} \cdot \beta^{d+1} \cdot n^{3d+2}) \\
&= O(s^{6d+1} \cdot \alpha^{3d+2} \cdot \beta^{2d+1} \cdot n^{5d+3} \cdot \log n), \\
\delta_1 &= w \cdot \alpha \cdot \beta \cdot \gamma_{\mathcal{R}} = O(s \cdot \alpha \cdot \beta \cdot n), \\
q &\approx \delta_0' \cdot n \cdot \log n = O(s^{6d+1} \cdot \alpha^{3d+2} \cdot \beta^{2d+1} \cdot n^{5d+4} \cdot \log^2 n),
\end{aligned}$$

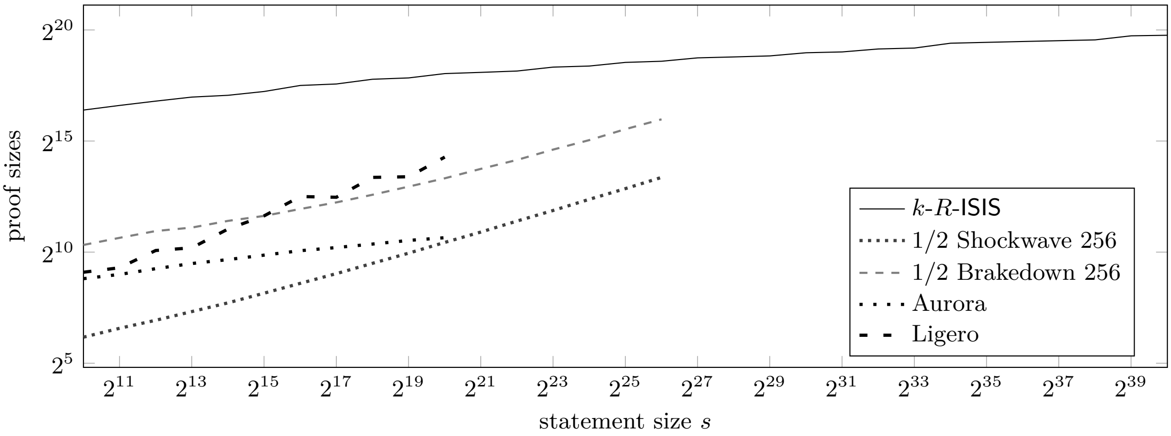

Fig. 4. Combined size (in KB) of a commitment and an opening proof for the concrete parameters chosen in Theorem 5, setting $\lambda = 128$, optimising for ρ and comparing with SNARK proof sizes in prior works [GLS+21, Fig. 5]. We picked $\alpha = s$.

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	$O(s^d + (\log s + \log w + \log t + \log \beta)/\log \beta)$

$$\begin{aligned}
\log \delta_0, \log \delta_1, \log q &= O(\log s + \log \alpha + \log \beta + \log n) = O(\log s + \log \beta),\\
\ell_0 = \ell_1 &= 2 \log q / \log \beta = O((\log s + \log \beta)/\log \beta),\\
|c| &= n \cdot \log q = O(n \log s), \text{ and}\\
|\mathbf{u}_i| &= n \cdot \ell_i \cdot \log \delta_i\\
&= n \cdot O((\log s + \log \beta)/\log \beta) \cdot O(\log s + \log \beta)\\
&= O(n \cdot (\log s + \log \beta)^2/\log \beta). \qquad \square
\end{aligned}$$

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. The factor of two arises from our reduction and the factor $\sqrt{n}$ translates between ℓ_∞ and ℓ_2. In Fig. 4 we report the concrete combined size (in KB) of a commitment and an opening proof for the concrete parameters chosen in Theorem 5, specifically setting $d = 2$, $\eta_0 = \eta_1 = 1$, and $\beta = s = w = t \in \{2^{10}, 2^{11}, \ldots, 2^{40}\}$.[18]

To analyse computation complexity, we assume the concrete parameter choices in Theorem 5 with the exception that s, w, t are treated as free variables 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 $O(n \log w + n \log t)$.

[18] Raw data and source code embedded.

6 GPV Adaptor Signatures

We consider hard languages of the form

$$\mathcal{L} := \{(\mathbf{A}, \mathbf{v}') \in \mathcal{R}_q^{\eta\times\ell} \times \mathcal{R}_q^{\eta} \mid \exists\ \mathbf{u}' \in \mathcal{R}^{\ell}\ s.t.\ , \mathbf{A}\cdot\mathbf{u}' = \mathbf{v}' \bmod q \wedge \|\mathbf{u}'\| \leq \beta^*\}.$$

```
$\mathsf{Setup}(1^\lambda)$
$\mathbf{A} \leftarrow\$\ \mathcal{R}_q^{\eta\times\ell}$
$\tilde{\mathsf{pp}} \leftarrow \mathit{\Pi}.\mathsf{Setup}(1^\lambda)$
return $\mathsf{pp} := (\mathbf{A}, \tilde{\mathsf{pp}})$

$\mathsf{KGen}(\mathsf{pp})$
$\mathbf{X}^{\mathrm{T}} \leftarrow (\mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, d))^\ell$
$\mathsf{sk} := \mathbf{X}$
$\mathsf{pk} := \mathbf{Y} := \mathbf{A}\cdot\mathbf{X} \bmod q$
return $(\mathsf{pk}, \mathsf{sk})$

$\mathsf{Sign}(\mathsf{sk}, m \in \mathcal{M})$
$\mathbf{u} \leftarrow \mathcal{D}_{\mathcal{R}^\ell,\rho}$
$\mathbf{v} := \mathbf{A}\cdot\mathbf{u} \bmod q$
$\mathbf{c} := \mathbf{G}^{-1}(\mathbf{v} - H(m))$
$\mathbf{z} := \mathbf{u} + \mathbf{X}\cdot\mathbf{c}$
return $\sigma := (\mathbf{c}, \mathbf{z})$

$\mathsf{Verify}(\mathsf{pk}, m, \sigma)$
return $\begin{cases} \mathbf{A}\cdot\mathbf{z} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} \overset{?}{=} H(m) \bmod q \\ \|(\mathbf{c},\mathbf{z})\| \leq \gamma_2 \end{cases}$
```

```
$\mathsf{pSign}(\mathsf{sk}, m, Y = (\mathbf{v}', \pi))$
if $\mathit{\Pi}.\mathsf{Verify}(\tilde{\mathsf{pp}}, \mathbf{v}', \pi) = 0$
    return $\perp$
$\mathbf{u} \leftarrow \mathcal{D}_{\mathcal{R}^\ell,\rho}$
$\mathbf{v} := \mathbf{A}\cdot\mathbf{u} \bmod q$
$\mathbf{c} := \mathbf{G}^{-1}(\mathbf{v} + \mathbf{v}' - H(m))$
$\hat{\mathbf{z}} := \mathbf{u} + \mathbf{X}\cdot\mathbf{c}$
return $\hat{\sigma} := (\mathbf{c}, \hat{\mathbf{z}})$

$\mathsf{PreVerify}(\mathsf{pk}, m, \mathbf{v}', \hat{\sigma})$
return $\begin{cases} \mathbf{A}\cdot\hat{\mathbf{z}} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} \overset{?}{=} H(m) - \mathbf{v}' \bmod q \\ \|(\mathbf{c},\hat{\mathbf{z}})\| \leq \gamma_1 \end{cases}$

$\mathsf{Adapt}(\hat{\sigma}, \mathbf{u}')$
$\mathbf{z} := \hat{\mathbf{z}} + \mathbf{u}'$
return $\sigma := (\mathbf{c}, \mathbf{z})$

$\mathsf{Ext}(\sigma, \hat{\sigma}, \mathbf{v}')$
return $\mathbf{u}' := \mathbf{z} - \hat{\mathbf{z}}$
```

Fig. 5. GPV based adaptor signatures using a NIZK-PoK $\mathit{\Pi}$.

We will consider the following hard relations R, $\widetilde{R}$, that capture witnesses used to adapt and extracted witnesses respectively, are given by

$$R_{\mathbf{A}} := \{(\mathbf{v}', \mathbf{u}') \in \mathcal{R}_q^{\eta} \times \mathcal{R}^{\ell} \mid \mathbf{v}' = \mathbf{A}\cdot\mathbf{u}' \bmod q \wedge \|\mathbf{u}'\| \leq \beta\},$$

and

$$\widetilde{R}_{\mathbf{A}} := \{(\mathbf{v}', \mathbf{u}') \in \mathcal{R}_q^{\eta} \times \mathcal{R}^{\ell} \mid \mathbf{v}' = \mathbf{A}\cdot\mathbf{u}' \bmod q \wedge \|\mathbf{u}'\| \leq \widetilde{\beta}\},$$

where $\beta \leq \widetilde{\beta}$. As done in Aumayr et. al. in [AEE$^+$21], we slightly modify the hard relation for which the adaptor signature is defined in order to be able to extract the corresponding witness in the security experiments. Let $\mathit{\Pi} = (\mathit{\Pi}.\mathsf{Setup}, \mathit{\Pi}.\mathsf{Prove}, \mathit{\Pi}.\mathsf{Verify})$ be a NIZK-PoK with online extractor for the relation $R_{\mathbf{A}}$, as defined in Definition 15. We will consider the relation $R_{\mathbf{A}}^{+}$, whose statements are pairs $(\mathbf{v}', \pi)$, where $(\mathbf{v}', \mathbf{u}') \in R_{\mathbf{A}}$, and $\pi \leftarrow \mathit{\Pi}.\mathsf{Prove}(\mathsf{pp}, \mathbf{v}', \mathbf{u}')$, for $\mathsf{pp} \leftarrow \mathit{\Pi}.\mathsf{Setup}(1^\lambda)$. That is

$$R_{\mathbf{A}}^{+} := \{((\mathbf{v}', \pi), \mathbf{u}') \mid \mathbf{v}' = \mathbf{A}\cdot\mathbf{u}' \bmod q \wedge \|\mathbf{u}'\| \leq \beta \wedge \mathit{\Pi}.\mathsf{Verify}(\mathsf{pp}, \mathbf{v}', \pi) = 1\}.$$

Since $R_{\mathbf{A}}$ is a hard relation, so is $R_{\mathbf{A}}^{+}$. In order to ease readability and avoid introducing too many different notations, in our construction we replace $R_{\mathbf{A}}$ with $R_{\mathbf{A}}^{+}$.

Parameters. The scheme parameters

- $\rho \geq (d\ell\sqrt{\ell} + \beta)\sqrt{Q}$, where Q is the maximum number of oracle queries allowed in the experiment,
- β, witness norm bound,
- $\gamma_1 \geq \rho\sqrt{\ell}$, norm bound for pre-signature,
- $\gamma_2 \geq \gamma_1 + \beta$, norm bound for signature,
- $\beta \geq \gamma_1 + \gamma_2$, norm bound of extracted witnesses,

have to be chosen so that $M\text{-}\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,2\gamma_2+2d\ell\sqrt{\ell}}$ and $M\text{-}\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,\gamma_1+\gamma_2+\beta+2d\ell\sqrt{\ell}}$ are hard.

6.1 Security Analysis

Pre-signature correctness follows via a straightforward investigation, using the fact that $\mathbf{A}\cdot\mathbf{u}' = \mathbf{v}' \bmod q$.

Lemma 10 (Weak Pre-signature Adaptability). *The adaptor signature scheme described in Fig. 5 satisfies weak pre-signature adaptability with respect to the relation $R_{\mathbf{A}}$.*

Proof. Let $\hat{\sigma} = (\mathbf{c}, \hat{\mathbf{z}})$ be a valid pre-signature with $\mathsf{PreVerify}(\mathsf{pk}, \mathbf{m}, \mathbf{v}', \hat{\sigma}) = 1$, and $\mathbf{u}'$, with $\|\mathbf{u}'\| \leq \beta$ be a witness corresponding to $\mathbf{v}'$. Since $\hat{\sigma}$ is valid, we have $\|\hat{\mathbf{z}}\| \leq \gamma_1$. Then, $\mathsf{Adapt}(\hat{\sigma}, \mathbf{u}') = (\mathbf{c}, \hat{\mathbf{z}} + \mathbf{u}') = \sigma$. Therefore, we have

$$\|\mathbf{z}\| = \|\hat{\mathbf{z}} + \mathbf{u}'\| \leq \|\hat{\mathbf{z}}\| + \|\mathbf{u}'\| \leq \gamma_1 + \beta \leq \gamma_2.$$

We further have

$$\begin{aligned}\mathbf{A}\cdot\mathbf{z} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} &= \mathbf{A}\cdot(\hat{\mathbf{z}} + \mathbf{u}') - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} \\ &= (\mathbf{A}\cdot\hat{\mathbf{z}} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c}) + \mathbf{A}\cdot\mathbf{u}' \\ &= (H(m) - \mathbf{v}') + \mathbf{v}' = H(m) \bmod q.\end{aligned}$$

From the above two equations, it follows that σ is a valid signature for message $\mathbf{m}$, i.e., $\mathsf{Verify}(\mathsf{pk}, \mathbf{m}, \sigma) = 1$. □

Lemma 11 (Weak Unforgeability). *Let $\mathit{\Pi} = (\mathit{\Pi}.\mathsf{Setup}, \mathit{\Pi}.\mathsf{Prove}, \mathit{\Pi}.\mathsf{Verify})$, used in the construction on the adaptor signature from Fig. 5, be a NIZK-PoK with online extractor for the relation $R_{\mathbf{A}}$. Assuming $M\text{-}\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,2\gamma_2+2d\ell}$ and that $\rho \geq (d\ell + \beta)\sqrt{Q}$, where Q is the maximum number of oracle queries an attacker can make, the adaptor signature from Fig. 5 is weakly unforgeable in the random oracle model.*

Proof. We prove the unforgeability of the adaptor signature scheme by reduction to the M-SIS problem. Let $(\mathbf{A}, \mathbf{v}^*)$ be the given M-SIS instance. Consider the following sequence of hybrids. In all of them let $\sigma^* = (\mathbf{c}^*, \mathbf{z}^*)$ be the forgery signature output by $\mathcal{A}$ on message m^*. Without loss of generality, we can assume that the adversary always queries the random oracle H on every message m before making a presigning/signing query on m.

- Hybrid Hyb_0: This is identical to the real experiment.
- Hybrid Hyb_1: This is identical to the real experiment except that the public parameters $\tilde{\mathsf{pp}}$ of the NIZK-PoK are generated by the simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$, whose existence is guaranteed by the zero-knowledge property of the NIZK-PoK $\mathit{\Pi}$ (Definition 15), i.e., $(\tilde{\mathsf{pp}}, \mathsf{state}_0) \leftarrow \mathcal{S}_1(1^\lambda)$. Moreover whenever the adversary outputs a challenge message m^*, the challenger samples $(\mathbf{v}^*, \mathbf{u}^*) \leftarrow \mathsf{GenR}(1^\lambda)$, runs the simulator S on input $(\tilde{\mathsf{pp}}, \mathsf{state}_0, \mathbf{v}^*)$, to obtain a simulated proof π^*, i.e., $\pi^* \leftarrow \mathsf{S}(\tilde{\mathsf{pp}}, \mathsf{state}_0, \mathbf{v}^*)$, and returns $(\hat{\sigma}, (\mathbf{v}^*, \pi^*))$ to the adversary.
- Hybrid Hyb_2: Here the simulator $\mathcal{S}$ works as follows:
  - The simulator records a list $\mathcal{Q}$ of all H queries made by $\mathcal{A}$ with their responses. Let $Q = |\mathcal{Q}|$ be the number of hash queries made by $\mathcal{A}$.
  - Whenever $\mathcal{A}$ queries the random oracle H on input m, the simulator samples $\mathbf{v} \leftarrow \mathcal{R}_q^\ell$, $\mathbf{z} \leftarrow \mathcal{D}_{\mathcal{R}^\ell,\rho}$, sets $\mathbf{c} := \mathbf{G}^{-1}(\mathbf{v})$, programs the random oracle $H(m) := \mathbf{A}\cdot\mathbf{z} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} \bmod q$, and returns $\sigma = (\mathbf{c}, \mathbf{z})$, stores $(m, H(m), \sigma)$ in $\mathcal{Q}$, and returns $H(m)$ to the adversary.
  - Whenever the adversary queries the $\mathsf{Sign}\mathcal{O}(\cdot)$ oracle on input m, the simulator finds the corresponding entry $(m, H(m), \sigma)$ in $\mathcal{Q}$, and returns σ to the adversary.
  - Whenever the adversary queries the $\mathsf{pSign}\mathcal{O}(\cdot)$ oracle on input $(m, (\mathbf{v}', \pi))$, the simulator checks the validity of $(\mathbf{v}', \pi))$, extracts the witness $\mathbf{u}'$ of $\mathbf{v}'$ from the proof π, finds the corresponding entry $(m, H(m), \sigma = (\mathbf{c}, \mathbf{z}))$ in $\mathcal{Q}$, and returns $\hat{\sigma} = (\mathbf{c}, \hat{\mathbf{z}} := (\mathbf{z} - \mathbf{u}'))$ to the adversary.

  - Whenever the adversary outputs a challenge message m^*, the simulator finds the corresponding entry $(m^*, H(m^*), \sigma = (\mathbf{c}, \mathbf{z}))$ in $\mathcal{Q}$, runs $(\mathbf{v}^*, \mathbf{u}^*) \leftarrow \mathsf{GenR}(1^\lambda)$, and returns $(\hat{\sigma} := (\mathbf{c}, \hat{\mathbf{z}} := \mathbf{z} - \mathbf{u}^*), (\mathbf{v}^*, \pi^*))$ to the adversary, where π^* is the corresponding simulated NIZK-PoK proof.

– Hybrid Hyb_3: This is identical to hybrid Hyb_2, except that this time the simulator samples $\mathbf{X}^\mathsf{T} \leftarrow (\mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, d))^\ell$, and sets $\mathbf{Y} := \mathbf{A} \cdot \mathbf{X} - \mathbf{G} \bmod q$.

Let δ_i denote the probability of an adversary winning in hybrid Hyb_i. Hybrids Hyb_0 and Hyb_1 only differ in the way the proof π^* is generated. By the zero-knowledge property of the proof system NIZK-PoK $\mathit{\Pi}$, one has that the distribution of simulated proofs is computationally indistinguishable to the distribution of real ones. Therefore, we obtain

$$\delta_0 \leq \delta_1 + \mathsf{negl}(\lambda).$$

Claim. If there is an adversary that makes at most Q oracle queries and can win the game in hybrid Hyb_1 with probability δ_1, then its probability of winning in hybrid Hyb_2 is polynomial in δ_1, if $\rho \geq (d\ell + \beta)\sqrt{Q}$.

Proof. The only difference between the two hybrids is in the value of $\mathbf{z}$ or $\hat{\mathbf{z}}$. For $i \in [Q]$, in hybrid Hyb_1 we have $\mathbf{z}_i$ or $\hat{\mathbf{z}}_i$ equal to $\mathbf{u}_i + \mathbf{X} \cdot \mathbf{c}_i$ with $\mathbf{u}_i \leftarrow \mathcal{D}_{\mathcal{R}^\ell, \rho}$, while in hybrid Hyb_2, we have $\mathbf{z}_i \leftarrow \mathcal{D}_{\mathcal{R}^\ell, \rho}$ and $\hat{\mathbf{z}}_i \leftarrow \mathcal{D}_{\mathcal{R}^\ell, \rho, -\mathbf{u}'}$. Let us refer to the joint distribution of all $\mathbf{z}$ and $\hat{\mathbf{z}}$ in Hyb_1 as D_1 and that in Hyb_2 as D_2. Let E denote the event that the adversary wins the game. Then, by our assumptions, we have $D_1(E) = \delta_1$. From the probability preservation property (Lemma 1) of the Rényi divergence, we get

$$D_2(E) \geq \frac{\delta_1^{\frac{a}{a-1}}}{R_a(D_1 || D_2)}, \quad \text{for any } a \in (1, \infty).$$

In order to compute $R_a(D_1 || D_2)$, notice that, for $i \in [Q]$, the vectors $\mathbf{z}_i$ or $\hat{\mathbf{z}}$ are drawn from distribution $D_{1i} = \mathcal{D}_{\mathcal{R}^\ell, \rho, \mathbf{X} \cdot \mathbf{c}_i}$ in hybrid Hyb_1, and from distribution $D_{2i} = \mathcal{D}_{\mathcal{R}^\ell, \rho}$ or $D_{2i} = \mathcal{D}_{\mathcal{R}^\ell, \rho, -\mathbf{u}'}$ in hybrid Hyb_2. Notice that $D_1 = (D_{11}, \ldots, D_{1Q})$, and $D_2 = (D_{21}, \ldots, D_{2Q})$. By Lemma 2, we have

$$R_a(D_{1i} || D_{2i}) \leq \exp\left(a\pi \frac{(\|\mathbf{X} \cdot \mathbf{c}_i\| + \|\mathbf{u}'\|)^2}{\rho^2}\right), \quad \text{for any } a \in (1, \infty).$$

Since each row of $\mathbf{X}$ has norm bounded by d, and $\|\mathbf{c}_i\| \leq \sqrt{\ell}$, we have $\|\mathbf{X} \cdot \mathbf{c}_i\| \leq d\ell$. Moreover, the extracted witness must have $\|\mathbf{u}'\| \leq \beta$ as the NIZK-PoK proof π verifies correctly. Using the multiplicativity property of the Rényi divergence (Lemma 1), we get

$$R_a(D_1 || D_2) \leq \exp\left(a\pi \frac{Q(d\ell + \beta)^2}{\rho^2}\right).$$

Using the assumption $\rho \geq (d\ell + \beta)\sqrt{Q}$, we get that $R_a(D_1 || D_2) \leq \exp(a\pi)$. Therefore, we obtain that $\delta_2 := D_2(E) \geq \delta_1^{\frac{a}{a-1}} / \exp(a\pi)$. Taking any value of $a > 1$ yields the result. □

Hybrids Hyb_2 and Hyb_3 only differ in the way the public key $\mathbf{Y}$ is generated. By the properties of SampD, we have that $\mathbf{A} \cdot \mathbf{X} \bmod q$ is statistically close to uniform. Thus, the same holds for $\mathbf{A} \cdot \mathbf{X} - \mathbf{G} \bmod q$, which implies that

$$\delta_2 \leq \delta_3 + \mathsf{negl}(\lambda).$$

Claim. If there is an adversary $\mathcal{A}$ that makes at most Q oracle queries, and succeeds in forging a valid signature with probability δ_3 is hybrid Hyb_3, then we can define an algorithm $\mathcal{B}$ which given $\mathbf{A} \leftarrow \mathcal{R}_q^{\eta \times \ell}$, finds a non-zero short $\mathbf{u} \in \mathcal{R}^\ell$ such that $\|\mathbf{u}\| \leq 2\gamma_2 + 2d\ell$ and $\mathbf{A} \cdot \mathbf{u} = \mathbf{0} \mod q$.

Proof. Let $\sigma^* = (\mathbf{c}^*, \mathbf{z}^*)$ be the forgery signature output by $\mathcal{A}$ on message m^*, $\mathbf{v}^*$ the challenge statement provided to $\mathcal{A}$, and $\sigma = (\mathbf{c}, \mathbf{z})$ the corresponding signature created when the adversary queried the random oracle on message m^*. Let $\mathbf{u} := \mathbf{z}^* - \mathbf{z} - \mathbf{X}(\mathbf{c}^* - \mathbf{c})$. We have

$$\begin{aligned} \mathbf{A} \cdot (\mathbf{z}^* - \mathbf{z} + \mathbf{X} \cdot (\mathbf{c}^* - \mathbf{c})) &= \mathbf{A} \cdot (\mathbf{z}^* - \mathbf{z}) - (\mathbf{G} + \mathbf{Y}) \cdot (\mathbf{c}^* - \mathbf{c}) \\ &= H(m) - H(m) \\ &= \mathbf{0} \bmod q. \end{aligned}$$

Moreover, since $\|\mathbf{z}^*\| \le \gamma_2$, $\|\mathbf{z}\| \le \gamma_2$, and $\|\mathbf{X}\cdot\mathbf{c}^*\|, \|\mathbf{X}\cdot\mathbf{c}\| \le d\ell$, we obtain $\|\mathbf{u}\| \le 2\gamma_2 + 2d\ell$. It remain to argue that $\mathbf{u} \neq \mathbf{0}$. We distinguish 2 cases:

Case 1: $\mathbf{c}^* = \mathbf{c}$. In this case we have $\mathbf{A}\cdot(\mathbf{z}^* - \mathbf{z}) = \mathbf{0}$. Recall that presignature given to $\mathcal{A}$ corresponding to statement $\mathbf{v}^*$ was of the form $\hat{\sigma} = (\mathbf{c}, \mathbf{z} - \mathbf{u}^*)$. We obtain

$$\begin{aligned}\mathbf{A}\cdot(\mathbf{z}^* - \mathbf{z}) &= \mathbf{A}\cdot(\mathbf{z}^* - (\hat{\mathbf{z}} + \mathbf{u}^*))\\ &= \mathbf{A}\cdot((\mathbf{z}^* - \hat{\mathbf{z}}) - \mathbf{u}^*)\\ &= \mathbf{0} \bmod q,\end{aligned}$$

which implies that $\mathbf{A}\cdot(\mathbf{z}^* - \hat{\mathbf{z}}) = \mathbf{v}^* = \mathbf{A}\cdot\mathbf{u}^* \bmod q$. As argued by Gentry et. al in [GPV08], the min-entropy of $\mathbf{u}^*$ given $\mathbf{v}^*$ (and also $\hat{\mathbf{z}}$ in our case) is $\omega(\log k)$. Thus, $\mathbf{z}^* - \hat{\mathbf{z}} \neq \mathbf{u}^*$, except with negligible probability.

Case 2: $\mathbf{c}^* \neq \mathbf{c}$. In this case, we can apply the same arguments as in Lemma 5.4 of [Lyu12], to get that $\mathbf{u} \neq \mathbf{0}$ with high probability. This proves the claim and thus, by showing that $\delta_3 \le \mathsf{negl}(\lambda)$, finishes the proof. □

Lemma 12 (Witness Extractability). *Let $\Pi = (\Pi.\mathsf{Setup}, \Pi.\mathsf{Prove}, \Pi.\mathsf{Verify})$, used in the construction on the adaptor signature from Fig. 5, be a NIZK-PoK with online extractor for the relation $R_{\mathbf{A}}$. Assuming M-$\mathsf{SIS}_{\mathcal{R}_q,\eta,\ell,\gamma_1+\gamma_2+\beta+2d\ell}$ and that $\rho \ge (d\ell+\beta)\sqrt{Q}$, where Q is the maximum number of oracle queries an attacker can make, the adaptor signature from Fig. 5 is witness extractable in the random oracle model.*

Proof. We prove the witness extractibility of the adaptor signature scheme by reduction to the M-SIS problem. Let $\mathbf{A}$ be the given M-SIS instance. The proof is very similar to that of Lemma 11. Consider the following sequence of hybrids. In all of them let $\sigma^* = (\mathbf{c}^*, \mathbf{z}^*)$ be the forgery signature output by $\mathcal{A}$ on message m^*, and let $(\mathbf{v}^*, \pi^*)$ be the challenge statement. Without loss of generality, we can assume that the adversary always queries the random oracle H on every message m before making a presigning/signing query on m.

- Hybrid Hyb_0: This is identical to the real experiment.
- Hybrid Hyb_1: Here the challenger works as follows:
  - The simulator records a list $\mathcal{Q}$ of all H queries made by $\mathcal{A}$ with their responses. Let $Q = |\mathcal{Q}|$ be the number of hash queries made by $\mathcal{A}$.
  - Whenever $\mathcal{A}$ queries the random oracle H on input m, the simulator samples $\mathbf{v} \leftarrow \mathcal{R}_q^\ell$, $\mathbf{z} \leftarrow \mathcal{D}_{\mathcal{R}^\ell,\rho}$, sets $\mathbf{c} := \mathbf{G}^{-1}(\mathbf{v})$, programs the random oracle $H(m) := \mathbf{A}\cdot\mathbf{z} - (\mathbf{G}+\mathbf{Y})\cdot\mathbf{c} \bmod q$, and returns $\sigma = (\mathbf{c}, \mathbf{z})$, stores $(m, H(m), \sigma)$ in $\mathcal{Q}$, and returns $H(m)$ to the adversary.
  - Whenever the adversary queries the $\mathsf{Sign}\mathcal{O}(\cdot)$ oracle on input m, the simulator finds the corresponding entry $(m, H(m), \sigma)$ in $\mathcal{Q}$, and returns σ to the adversary.
  - Whenever the adversary queries the $\mathsf{pSign}\mathcal{O}(\cdot)$ oracle on input $(m, (\mathbf{v}', \pi))$, the simulator checks the validity of $(\mathbf{v}', \pi))$, extracts the witness $\mathbf{u}'$ of $\mathbf{v}'$ from the proof π, finds the corresponding entry $(m, H(m), \sigma = (\mathbf{c}, \mathbf{z}))$ in $\mathcal{Q}$, and returns $\hat{\sigma} = (\mathbf{c}, \hat{\mathbf{z}} := (\mathbf{z} - \mathbf{u}'))$ to the adversary.
  - Whenever the adversary outputs a challenge message-statement tuple $(m^*, (\mathbf{v}^*, \pi^*))$, the simulator works as if it was responding to a presignature query: it makes use of the extractor K, whose existence is guaranteed by the extractability property of the NIZK-PoK Π (Definition 15). In order to extract the witness $\mathbf{u}^*$ corresponding to the statement $\mathbf{v}^*$, it runs extractor K, with access to the random oracle and its answers, on input $(\mathbf{v}^*, \pi^*)$, i.e., $\mathbf{u}^* \leftarrow \mathsf{K}(\mathbf{v}^*, \pi^*)$. Then, it finds the entry $(m^*, H(m^*), \sigma = (\mathbf{c}, \mathbf{z}))$ in $\mathcal{Q}$ corresponding to m^*, and returns $\hat{\sigma} = (\mathbf{c}, \hat{\mathbf{z}} := (\mathbf{z} - \mathbf{u}^*))$ to the adversary.
- Hybrid Hyb_2: This is identical to hybrid Hyb_1, except that this time the simulator samples $\mathbf{X}^\intercal \leftarrow (\mathsf{SampD}(1^\eta, 1^\ell, \mathcal{R}, d))^\ell$, and sets $\mathbf{Y} := \mathbf{A}\cdot\mathbf{X} - \mathbf{G} \bmod q$.

Let δ_i denote the probability of an adversary winning in hybrid Hyb_i.

Claim. If there is an adversary that makes at most Q oracle queries and can win the game in hybrid Hyb_0 with probability δ_0, then its probability of winning in hybrid Hyb_1 is polynomial in δ_0, if $\rho \ge (d\ell+\beta)\sqrt{Q}$.

Proof. The proof is identical to that of the analogous claim used in the proof of Lemma 11. □

Hybrids Hyb_1 and Hyb_2 only differ in the way the public key $\mathbf{Y}$ is generated. By the properties of SampD, we have that $\mathbf{A} \cdot \mathbf{X} \bmod q$ is statistically close to uniform. Thus, the same holds for $\mathbf{A} \cdot \mathbf{X} - \mathbf{G} \bmod q$, which implies that

$$\delta_1 \leq \delta_2 + \mathsf{negl}(\lambda).$$

Claim. If there is an adversary $\mathcal{A}$ that makes at most Q oracle queries, and succeeds winning with probability δ_2 in hybrid Hyb_2, then we can define an algorithm $\mathcal{B}$ which given $\mathbf{A} \leftarrow \mathcal{R}_q^{\eta \times \ell}$, finds a non-zero short $\mathbf{u}^*$ such that $\|\mathbf{u}^*\| \leq \gamma_1 + \gamma_2 + \beta + 2d\ell$ and $\mathbf{A} \cdot \mathbf{u}^* = 0 \bmod q$.

Proof. Let $\sigma^* = (\mathbf{c}^*, \mathbf{z}^*)$ be the forged signature output by $\mathcal{A}$. We distinguish 2 cases:

Case 1: $\mathbf{c}^* = \mathbf{c}$. Since both pre-signature and signature verify, we have that

$$\mathbf{A} \cdot \mathbf{z}^* - (\mathbf{G} + \mathbf{Y}) \cdot \mathbf{c} = H(m) \bmod q \qquad \text{and} \qquad \mathbf{A} \cdot \hat{\mathbf{z}} - (\mathbf{G} + \mathbf{Y}) \cdot \mathbf{c} = H(m) - \mathbf{v}^* \bmod q,$$

from which we obtain that

$$\mathbf{A} \cdot (\mathbf{z}^* - \hat{\mathbf{z}}) = \mathbf{v}^* \bmod q.$$

As $\|\mathbf{z}^* - \hat{\mathbf{z}}\| \leq \gamma_1 + \gamma_2 \leq \widetilde{\beta}$, the output of the Ext algorithm $\mathbf{u}^* := \mathbf{z}^* - \hat{\mathbf{z}}$ is a valid witness for $\mathbf{v}^*$.

Case 2: $\mathbf{c}^* \neq \mathbf{c}$. In this case, we make use of the extractability property of the zero-knowledge proof π^*, in order to extract $\mathbf{u}^*$ and obtain from the forged signature a M-SIS solution. Let $\mathbf{u}^* \leftarrow \mathsf{K}(\mathbf{v}, \pi, \mathcal{H})$, where $\mathcal{H}$ is the list of random oracle queries made by $\mathcal{A}$. With high probability, it holds that $((\mathbf{v}^*, \pi^*), \mathbf{u}^*) \in R_{\mathbf{A}}$. Using that

$$\mathbf{A} \cdot \mathbf{z}^* - (\mathbf{G} + \mathbf{Y}) \cdot \mathbf{c}^* = H(m) \bmod q \qquad \text{and} \qquad \mathbf{A} \cdot \hat{\mathbf{z}} - (\mathbf{G} + \mathbf{Y}) \cdot \mathbf{c} = H(m) - \mathbf{v}^* \bmod q,$$

we obtain

$$[\mathbf{A} | \mathbf{A}\mathbf{X}] \cdot \begin{bmatrix} \mathbf{z}^* - \hat{\mathbf{z}} + \mathbf{u}^* \\ \mathbf{c}^* - \mathbf{c} \end{bmatrix} = 0 \bmod q,$$

which leads to the non-zero M-SIS solution $\mathbf{r} := \mathbf{z}^* - \hat{\mathbf{z}} + \mathbf{u}^* + \mathbf{X} \cdot (\mathbf{c}^* - \mathbf{c})$, with $\|\mathbf{r}\| \leq \gamma_1 + \gamma_2 + \beta + 2d\ell$, by relying again on the analysis done in Lemma 5.4 of [Lyu12]. □

7 SNARK for Polynomial Maps Satisfiability

We construct a SNARK Π for $\mathrm{PolySAT}_{\mathcal{R},d,\alpha}$ in Fig. 6, based on the vector commitment Γ for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ that we developed in Section 5. The following theorem establishes the properties of our construction.

```
$\Pi.\mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$
return $\mathsf{pp} \leftarrow \Gamma.\mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$

$\Pi.\mathsf{PreVerify}(\mathsf{pp}, (f, y))$
return $\mathsf{pp}_{f,y} \leftarrow \Gamma.\mathsf{PreVerify}(\mathsf{pp}, (f, y))$

$\Pi.\mathsf{Prove}(\mathsf{pp}, (f, y, \mathbf{z}), \mathbf{x})$
$(c, \mathsf{aux}) \leftarrow \Gamma.\mathsf{Com}(\mathsf{pp}, \mathbf{x})$
$\pi' \leftarrow \Gamma.\mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$
return $\pi := (c, \pi')$

$\Pi.\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, \pi)$
return $\Gamma.\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi')$
```

Fig. 6. Construction of SNARK Π for $\mathrm{PolySAT}_{\mathcal{R},d,\beta}$ from a VC Γ for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$.

Theorem 6. *If Γ is correct, then the SNARK Π presented in Fig. 6 is complete. If Γ is $\mathcal{X}^*$-extractable, then Π is $\mathcal{X}^*$-knowledge-sound. If the computation complexity of $\Gamma.\mathsf{Verify}$ is in $\mathsf{poly}(\lambda, s, \log w, \log t)$ (implying that Γ is compact), then Π is succinct.*

Proof. (Sketch) Completeness and succinctness are immediate. For knowledge soundness, by the extractability of $\mathit{\Gamma}$, for any adversary $\mathcal{A}$ producing a commitment c and a valid opening proof for $(f, y, \mathbf{z})$, there exists an efficient procedure to extract from $\mathcal{A}$ a short vector $\mathbf{x}$ such that $f(\mathbf{z}, \mathbf{x}) = y(\mathbf{z})$, except with negligible probability.

7.1 Proving Relations over $\mathcal{R}_q$

In Section 7, we constructed a SNARK for proving knowledge of a short vector $\mathbf{x}$ with $\|\mathbf{x}\| \leq \alpha$ satisfying $f(\mathbf{x}) = \mathbf{y}$, where the polynomial map f and vector $\mathbf{y}$ both have coefficients of norm also at most α.[19] There are, however, natural applications where we want to prove algebraic relations which involve $\mathcal{R}$ elements of high norm ($> \alpha$) and where arithmetic is performed modulo q.

For example, the verification equation of a GPV signature [GPV08] is of the form $\mathbf{A}\cdot\mathbf{u} = H(m) \bmod q$, where $\mathbf{A}$ is a random public key matrix over $\mathcal{R}_q$, $H(m)$ is a random vector over $\mathcal{R}_q$ encoding the public message m, and the signature $\mathbf{u}$ is a short vector over $\mathcal{R}$ satisfying the relation. The verification equation of our VC and SNARK constructions $\mathbf{A}_0 \cdot \mathbf{u}_0 \overset{?}{=} \hat{f}(c) \cdot \mathbf{t}_0 \bmod q$ have a more complicated form involving the evaluation of a polynomial $\hat{f}$ with large coefficients at a large $\mathcal{R}_q$ element c.

In general, consider the task of proving

$$\{\, (f, \mathbf{y}) : \exists\ (\mathbf{x}, \mathbf{c}) \in \mathcal{R}^w \times \mathcal{R}_q^\ell,\ f(\mathbf{x}, \mathbf{c}) = \mathbf{y} \bmod q \wedge \|\mathbf{x}\| \leq \delta \,\}$$

for some $\delta \in \mathbb{R}$ and $q \in \mathbb{N}$, where the polynomial map f and the vector $\mathbf{y}$ have coefficients of norm at most q. Here, $\mathbf{c}$ represent part of the witness which is not necessarily short, e.g. a commitment in our VC construction. We outline a series of transformations on $(f, \mathbf{y})$ and $(\mathbf{x}, \mathbf{c})$ to obtain slightly relaxed[20] statement and witness respectively satisfying a relation natively supported by our SNARK.

We will assume that there exists an odd rational integer $p \in \mathbb{N}$ with $p \leq 2\alpha + 1$ and either $\delta \leq \alpha$ or $2\delta + 1$ is a power of p. Since α for our VC and SNARK and (usually) δ for the application can be chosen freely from a wide range of values, we view this as a mild assumption. The resulting statement and witness will be larger than their original counterparts by a multiplicative factor of $\mathsf{poly}(\log_\alpha q)$.

Handling Modular Reduction. To remove the modular reduction step, let $r \in \mathcal{R}$ be such that $f(\mathbf{x}, \mathbf{c}) + q \cdot r = \mathbf{y}$. Let $q' \in \mathbb{N}$ be the smallest such that $r \in \mathcal{R}_{q'}$. By absorbing r into $\mathbf{c}$ and renaming q' to q, we obtain an equivalent language of the form

$$\{\, (f, \mathbf{y}) : \exists\ (\mathbf{x}, \mathbf{c}) \in \mathcal{R}^w \times \mathcal{R}_q^\ell,\ f(\mathbf{x}, \mathbf{c}) = \mathbf{y} \wedge \|\mathbf{x}\| \leq \delta \,\}.$$

Handling Long Witness Components. Next, we transform the witness $(\mathbf{x}, \mathbf{c})$ into an equivalent witness of norm at most α. Write $d = 2\delta + 1$. Let $p \in \mathbb{N}$ be an odd rational integer satisfy the following conditions: (i) $p \leq 2\alpha + 1$, (ii) if $\delta \leq \alpha$ then $p \leq 2\delta + 1$, and (iii) if $\delta > \alpha$ then $p^k = 2\delta + 1 = d$ for some $k \in \mathbb{N}$.

For any $x, h \in \mathbb{N}$, define the p-ary 'Gadget" matrix $\mathbf{G}_{x,h} = (p^i)^{\mathtt{T}}_{i \in \mathbb{Z}_{\lceil \log_p x \rceil}} \otimes \mathbf{I}_h \in \mathcal{R}_q^{h \times h \cdot \lceil \log_p x \rceil}$. Let $\mathbf{G}^{-1}(\cdot)$ denote the component-wise balanced p-ary decomposition, i.e. it outputs a vector with entries in $\{\, -(p-1)/2, \ldots, 0, \ldots, (p-1)/2 \,\}$. Note that $\mathbf{x} = \mathbf{G}_{d,w'} \cdot \mathbf{G}^{-1}(\mathbf{x})$ and $\mathbf{c} = \mathbf{G}_{q,\ell} \cdot \mathbf{G}^{-1}(\mathbf{c})$. By construction, if $\left\|\mathbf{G}^{-1}(\mathbf{x})\right\| \leq \alpha$, then we must have $\|\mathbf{x}\| \leq \delta$. Given a polynomial map $f(\mathbf{X}, \mathbf{C})$, define

$$f'(\mathbf{X}', \mathbf{C}') := f(\mathbf{G}_{d,w} \cdot \mathbf{X}, \mathbf{G}_{q,\ell} \cdot \mathbf{C}).$$

By renaming f' to f and absorbing $\mathbf{c}$ into $\mathbf{x}$, we obtain an equivalent language of the form

$$\{\, (f, \mathbf{y}) : \exists\ \mathbf{x} \in \mathcal{R}^w,\ f(\mathbf{x}) = \mathbf{y} \wedge \|\mathbf{x}\| \leq \alpha \,\}$$

Note that unlike the previous and the original languages f likely contains large coefficients not contained in $\mathcal{R}_q$.

[19] We dropped the public input $\mathbf{z}$ for the ease of exposition.

[20] In the sense that the norm of the transformed witness has a looser upper bound which is polynomial in the original.

Handling Long Coefficients in Statements. It remains to transform $(f, \mathbf{y})$ with long coefficients to an equivalent statement containing only coefficients of norm at most α. Let $q' \in \mathbb{N}$ be the smallest such that all coefficients of f and $\mathbf{y}$ are contained in $\mathcal{R}_{q'}$. We first replace $f, \mathbf{y}$ by $(f', \mathbf{y}') := (\mathbf{G}^{-1}(f), \mathbf{G}^{-1}(\mathbf{y}))$ where $\mathbf{G}^{-1}(f)$ denotes the coefficient-wise balanced p-ary decomposition of f by viewing f as a linear map on monomials with coefficient vectors in $\mathcal{R}_{q'}^t$. Note that if $\mathbf{x}$ were to satisfy $f'(\mathbf{x}) = \mathbf{y}'$, then it also satisfies $f(\mathbf{x}) = \mathbf{y}$ because

$$\begin{aligned} f'(\mathbf{x}) &= \mathbf{y}' \\ \mathbf{G}^{-1}(f)(\mathbf{x}) &= \mathbf{G}^{-1}(\mathbf{y}) \\ \mathbf{G}_{q',t} \cdot \mathbf{G}^{-1}(f)(\mathbf{x}) &= \mathbf{G}_{q',t} \cdot \mathbf{G}^{-1}(\mathbf{y}) \\ f(\mathbf{x}) &= \mathbf{y}. \end{aligned}$$

However, this transformation is not complete as $f(\mathbf{x}) = \mathbf{y}$ does not necessarily imply $f'(\mathbf{x}) = \mathbf{y}'$.

To address above the issue, we consider any parity-check matrix $\mathbf{H}$ of $\mathbf{G}_{q',t}$, i.e. $\mathbf{G}_{q',t} \cdot \mathbf{H} = \mathbf{0}$ and $\mathbf{H}$ is full-rank. Suppose $\mathbf{x}$ satisfies $f(\mathbf{x}) = \mathbf{y}$. Consider $\mathbf{w} := f'(\mathbf{x}) - \mathbf{y}'$. We have $\mathbf{G} \cdot \mathbf{w} = \mathbf{G} \cdot f'(\mathbf{x}) - \mathbf{G} \cdot \mathbf{y}' = f(\mathbf{x}) - \mathbf{y} = 0$. Therefore there exists unique $\mathbf{z}$ such that $\mathbf{w} = \mathbf{H} \cdot \mathbf{z}$.

With the above observation, we pick a specific $\mathbf{H}$ which has p on the main diagonal, -1 in the entries just below the diagonal and zero everywhere else, and define

$$f''(\mathbf{X}, \mathbf{Z}) := f'(\mathbf{X}) - \mathbf{H} \cdot \mathbf{Z}.$$

By the previous argument, with the knowledge $\mathbf{x}$ satisfying $f(\mathbf{x}) = \mathbf{y}$, one could find a unique $\mathbf{z}$ satisfying $f''(\mathbf{x}, \mathbf{z}) = \mathbf{y}'$. Conversely, suppose $(\mathbf{x}, \mathbf{z})$ satisfies $f''(\mathbf{x}, \mathbf{z}) = \mathbf{y}'$. We have

$$\begin{aligned} f'(\mathbf{x}) - \mathbf{H} \cdot \mathbf{z} &= \mathbf{y}' \\ \mathbf{G}_{q',t} \cdot f'(\mathbf{x}) - \underbrace{\mathbf{G}_{q',t} \cdot \mathbf{H}}_{\mathbf{0}} \cdot \mathbf{z} &= \mathbf{G}_{q',t} \cdot \mathbf{y}' \\ f(\mathbf{x}) &= \mathbf{y}. \end{aligned}$$

Note that the coefficients of f'' and the entries of $\mathbf{y}'$ have norm upper-bounded by α by construction. It remains to upper-bound $\|\mathbf{z}\|$ given that $\|\mathbf{x}\| \leq \alpha$ and $f''(\mathbf{x}, \mathbf{z}) = \mathbf{y}'$. Let $\mathbf{w} := f'(\mathbf{x}) - \mathbf{y}'$ so that $\mathbf{H} \cdot \mathbf{z} = \mathbf{w}$ and $\|\mathbf{w}\| \leq \alpha' := (w + d)^d \cdot \alpha^{d+1} \cdot \gamma_{\mathcal{R}}^d$. By the construction of $\mathbf{H}$, we have $w_0 = p \cdot z_0$ and $w_i = p \cdot z_i - z_{i-1}$ for $i > 0$. Consequently, we have $\|z_0\| < \|w_0\| \leq \alpha'$ and $\|z_i\| \leq (\|w_i\| + \|z_{i-1}\|)/2 \leq \alpha'$ for $i > 0$.

By renaming f'' to f, $\mathbf{y}'$ to $\mathbf{y}$, and α' to α, and absorbing $\mathbf{z}$ into $\mathbf{x}$, we obtain a relaxed language of the form

$$\{\, (f, \mathbf{y}) : \exists\, \mathbf{x} \in \mathcal{R}^w,\ f(\mathbf{x}) = \mathbf{y} \wedge \|\mathbf{x}\| \leq \alpha \,\}$$

which is natively supported by our SNARK. Note that the resulting language is relaxed in the sense that it only requires $\|\mathbf{x}\|$ to be upper-bounded by $\alpha' = \gamma_{\mathcal{R}}^d \cdot \alpha^{d+1}$ instead of by α required in the original language.

7.2 Applications

Although a SNARK for an NP-complete language can in principle be used to prove any NP relation, the computation and verification of the proof may not be concretely efficient due to NP reductions. In the following, we highlight languages which are natively supported by our SNARK construction.

Aggregating GPV Signatures. GPV [GPV08] is a lattice-based signature scheme paradigm of which an instantiation is in the process of being standardised [PFH+20]. GPV signatures are a prime candidate for aggregation as it is unclear how to perform aggregation efficiently in other lattice-signature paradigms based on Schnorr-like paradigms, due to how the random oracle is used there and how it is typically instantiated with hash functions of high multiplicative degree (when viewed as an arithmetic circuit) [DHSS20,BR21,BK20]. On a high level, GPV signatures work as follows. A signature is a short vector $\mathbf{u}$, with respect to a public key $\mathbf{A}$. To verify the signature, the verifier computes $\mathbf{v} = H(m)$, checks that the linear relation $\mathbf{A} \cdot \mathbf{u} \equiv \mathbf{v} \bmod q$ holds and that $\mathbf{u}$ is short, where H is modeled as a random oracle.

As motivated in Section 7.1, our SNARK construction can be used to prove knowledge of GPV signatures natively given the verification is a linear relation. The high level idea is to use our SNARK construction to prove knowledge of n signatures where each of them are short vectors satisfying a linear relation. Consider the scenario where the same set of signers, identified with the public keys $(\mathbf{A}_i)_{i \in \mathbb{Z}_n}$, periodically issue signatures $(\mathbf{u}_{i,j})_{i \in \mathbb{Z}_n}$ on a common message m_j with $\mathbf{v}_j = H(m_j)$ at each time j.[21] An aggregator can aggregate the n signatures issued at each time j by computing a SNARK proof for the knowledge of short $(\mathbf{u}_{i,j})_{i \in \mathbb{Z}_n}$ satisfying $\mathbf{A}_i \cdot \mathbf{u}_{i,j} \equiv \mathbf{v}_j \bmod q$. 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 $(\mathbf{A}_i)_{i \in \mathbb{Z}_n}$. This preprocessing step only needs to be done once for the same set of signers. Then, when the message m_j becomes known at or after time j, the online verification time is only logarithmic in n.

The above idea can also be extended to the case where multiple signers sign different messages. In this case, one can still preprocess the public keys $(\mathbf{A}_i)_{i \in \mathbb{Z}_n}$ of users if they are known ahead of time. The verification time is linear in n since we have to check relations with respect to different messages. However, we are still set to gain from the compactness of the SNARK proof. Such aggregation can aid in the blockchain setting, where an aggregator can aggregate signatures on different transactions included in a block; resulting in smaller blocks to mitigate the effects on the ever-growing size of blockchains.

Recursive SNARK Composition. Since our SNARK construction is purely algebraic over $\mathcal{R}$ and $\mathcal{R}_q$, it can be used to natively prove knowledge of a committed witness and a SNARK proof that satisfy the verification equation. Furthermore, since the verification time of our SNARK construction is sublinear after preprocessing, our SNARK construction can be recursively composed without blowing up the proof size. This makes our SNARK construction suitable for the constructions of verifiable delay functions [BBBF18] and incrementally verifiable computation [Val08] based on the recursive composition of SNARKs. Below, we outline a naive recursive composition strategy which only achieves provable soundness for constant-depth composition. We refer to the literature [Val08,BCCT13,BBBF18] for more advanced composition strategies to support higher-depth composition.

Consider a long computation which involves iteratively applying a computation C on an initial input $\mathbf{x}_0$ for t times to obtain $\mathbf{x}_t = C^t(\mathbf{x}_0)$, where $\mathbf{x}_{i+1} = C(\mathbf{x}_i)$ for $i \in \mathbb{Z}_t$. Let pp be the public parameters sampled by the SNARK construction of a sufficient for the following language $L = L_{\mathsf{pp},C}$ with relation $R = R_{\mathsf{pp},C}$: A statement in L consists of a vector $\mathbf{x}'$. A witness is of the form $(\pi, \mathbf{x})$ where $\mathbf{x}$. The relation R is satisfied if

$$\begin{cases} \pi = \mathbf{x} \ \vee \ \mathsf{Verify}(\mathsf{pp}_R, \mathbf{x}, \pi) = 1 \\ \mathbf{x}' = C(\mathbf{x}) \end{cases}$$

where $\mathsf{pp}_R = \mathsf{PreVerify}(\mathsf{pp}, R)$.

To prove that a statement $(C, \mathbf{x}_t)$ and a witness $\mathbf{x}_0$ satisfy $\mathbf{x}_t = C^t(\mathbf{x}_0)$, the prover computes:

- Set $\mathsf{wit}_0 := \mathbf{x}_0$.
- For $i \in \mathbb{Z}_t$:
  - Compute $\mathbf{x}_{i+1} = C(\mathbf{x}_i)$.
  - Compute $\pi_{i+1} \leftarrow \mathsf{Prove}(\mathsf{pp}, (R, \mathbf{x}_{i+1}), \mathsf{wit}_i)$.
  - Set $\mathsf{wit}_{i+1} := (\pi_{i+1}, \mathbf{x}_{i+1})$
- Output π_t.

The proof can then be verified by checking that $\mathsf{Verify}(\mathsf{pp}_R, \mathbf{x}_t, \pi_t) = 1$.

To show succinctness, we observe that the computation required for checking the relation R given pp_R is of size $\mathsf{polylog}(|R|, |C|, \lambda) \cdot \mathsf{poly}(\lambda) + |C|$, and the computation required for verifying a SNARK proof of the satisfiability of R given pp_R is of size $\mathsf{polylog}(|R|, |C|, \lambda) \cdot \mathsf{poly}(\lambda)$. However, this composition strategy is not known to be provably sound for large t, say $t = \Omega(\lambda)$, since the knowledge extractor

[21] Signing the same message twice produces a solution for M-SIS on $\mathbf{A}_i$, so we may assume a deterministic signature scheme here to avoid this issue.

may run in time exponential in t (unless the underlying SNARK has a very efficient extractor $\mathcal{E}_\mathcal{A}$ which runs only an additive factor longer than the the runtime of $\mathcal{A}$). Fortunately, this issue is discussed and circumvented in many prior works (e.g. [Val08,BCCT13,BBBF18]) where the techniques should also be applicable to the recursive composition of our SNARK construction.

References

ACK21. Thomas Attema, Ronald Cramer, and Lisa Kohl. A compressed Σ-protocol theory for lattices. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part II*, volume 12826 of *LNCS*, pages 549–579, Virtual Event, August 2021. Springer, Heidelberg. 2

AEE$^+$21. Lukas Aumayr, Oguzhan Ersoy, Andreas Erwig, Sebastian Faust, Kristina Hostáková, Matteo Maffei, Pedro Moreno-Sanchez, and Siavash Riahi. Generalized channels from limited blockchain scripts and adaptor signatures. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 635–664. Springer, 2021. 4, 8, 14, 35

AGHS13. Shweta Agrawal, Craig Gentry, Shai Halevi, and Amit Sahai. Discrete Gaussian leftover hash lemma over infinite domains. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 97–116. Springer, Heidelberg, December 2013. 48

AGPS20. Martin R. Albrecht, Vlad Gheorghiu, Eamonn W. Postlethwaite, and John M. Schanck. Estimating quantum speedups for lattice sieves. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part II*, volume 12492 of *LNCS*, pages 583–613. Springer, Heidelberg, December 2020. 11

Agr20. Shweta Agrawal. Unlikely friendships: The fruitful interplay of cryptography assumptions. Invited talk at ASIACRYPT 2020, December 2020. https://youtu.be/Owz8UuWTsqg. 3

Ajt96. Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *28th ACM STOC*, pages 99–108. ACM Press, May 1996. 12

AKSY21. Shweta Agrawal, Elena Kirshanova, Damien Stehle, and Anshu Yadav. Can round-optimal lattice-based blind signatures be practical? Cryptology ePrint Archive, Report 2021/1565, 2021. https://eprint.iacr.org/2021/1565. 9, 11

AL21. Martin R. Albrecht and Russell W. F. Lai. Subtractive sets over cyclotomic rings - limits of Schnorr-like arguments over lattices. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part II*, volume 12826 of *LNCS*, pages 519–548, Virtual Event, August 2021. Springer, Heidelberg. 7, 10, 52

AME$^+$21. Lukas Aumayr, Matteo Maffei, Oğuzhan Ersoy, Andreas Erwig, Sebastian Faust, Siavash Riahi, Kristina Hostáková, and Pedro Moreno-Sanchez. Bitcoin-compatible virtual channels. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 901–918. IEEE, 2021. 4, 8

ARS$^+$15. Martin R. Albrecht, Christian Rechberger, Thomas Schneider, Tyge Tiessen, and Michael Zohner. Ciphers for MPC and FHE. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 430–454. Springer, Heidelberg, April 2015. 2

ARU14. Andris Ambainis, Ansis Rosmanis, and Dominique Unruh. Quantum attacks on classical proof systems: The hardness of quantum rewinding. In *55th FOCS*, pages 474–483. IEEE Computer Society Press, October 2014. 53

BBBF18. Dan Boneh, Joseph Bonneau, Benedikt Bünz, and Ben Fisch. Verifiable delay functions. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part I*, volume 10991 of *LNCS*, pages 757–788. Springer, Heidelberg, August 2018. 4, 42, 43

BCC$^+$09. Mira Belenkiy, Jan Camenisch, Melissa Chase, Markulf Kohlweiss, Anna Lysyanskaya, and Hovav Shacham. Randomizable proofs and delegatable anonymous credentials. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 108–125. Springer, Heidelberg, August 2009. 1

BCCT13. Nir Bitansky, Ran Canetti, Alessandro Chiesa, and Eran Tromer. Recursive composition and bootstrapping for SNARKS and proof-carrying data. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 111–120. ACM Press, June 2013. 42, 43

BCG$^+$13. Eli Ben-Sasson, Alessandro Chiesa, Daniel Genkin, Eran Tromer, and Madars Virza. SNARKs for C: Verifying program executions succinctly and in zero knowledge. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 90–108. Springer, Heidelberg, August 2013. 1

BCG$^+$14. Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society Press, May 2014. 1

BCS21. Jonathan Bootle, Alessandro Chiesa, and Katerina Sotiraki. Sumcheck arguments and their applications. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 742–773, Virtual Event, August 2021. Springer, Heidelberg. 2, 16

BCTV14a. Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Scalable zero knowledge via cycles of elliptic curves. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 276–294. Springer, Heidelberg, August 2014. 2

BCTV14b. Eli Ben-Sasson, Alessandro Chiesa, Eran Tromer, and Madars Virza. Succinct non-interactive zero knowledge for a von neumann architecture. In Kevin Fu and Jaeyeon Jung, editors, *USENIX Security 2014*, pages 781–796. USENIX Association, August 2014. 1

BDFG21. Dan Boneh, Justin Drake, Ben Fisch, and Ariel Gabizon. Halo infinite: Proof-carrying data from additive polynomial commitments. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 649–680, Virtual Event, August 2021. Springer, Heidelberg. 1

BDGL16. Anja Becker, Léo Ducas, Nicolas Gama, and Thijs Laarhoven. New directions in nearest neighbor searching with applications to lattice sieving. In Robert Krauthgamer, editor, *27th SODA*, pages 10–24. ACM-SIAM, January 2016. 11

BDN18. Dan Boneh, Manu Drijvers, and Gregory Neven. Compact multi-signatures for smaller blockchains. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 435–464. Springer, Heidelberg, December 2018. 8

Ber14. Daniel J. Bernstein. A subfield-logarithm attack against ideal lattices: Computational algebraic number theory tackles lattice-based cryptography. The cr.yp.to blog, https://blog.cr.yp.to/20140213-ideal.html, 2014. 27

BF11. Dan Boneh and David Mandell Freeman. Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures. In Dario Catalano, Nelly Fazio, Rosario Gennaro, and Antonio Nicolosi, editors, *PKC 2011*, volume 6571 of *LNCS*, pages 1–16. Springer, Heidelberg, March 2011. 3, 12

BGH19. Sean Bowe, Jack Grigg, and Daira Hopwood. Halo: Recursive proof composition without a trusted setup. Cryptology ePrint Archive, Report 2019/1021, 2019. https://eprint.iacr.org/2019/1021. 1

BK20. Dan Boneh and Sam Kim. One-time and interactive aggregate signatures from lattices. https://crypto.stanford.edu/~skim13/agg_ots.pdf, 2020. 41

BLR+18. Shi Bai, Tancrède Lepoint, Adeline Roux-Langlois, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance. *Journal of Cryptology*, 31(2):610–640, April 2018. 11, 12

BMM+21. Benedikt Bünz, Mary Maller, Pratyush Mishra, Nirvan Tyagi, and Psi Vesely. Proofs for inner pairing products and applications. In Mehdi Tibouchi and Huaxiong Wang, editors, *ASIACRYPT 2021, Part III*, volume 13092 of *LNCS*, pages 65–97. Springer, Heidelberg, December 2021. 2

BMRS20. Joseph Bonneau, Izaak Meckler, Vanishree Rao, and Evan Shapiro. Coda: Decentralized cryptocurrency at scale. *Cryptology ePrint Archive*, 2020. 1

BR21. Katharina Boudgoust and Adeline Roux-Langlois. Compressed linear aggregate signatures based on module lattices. Cryptology ePrint Archive, Report 2021/263, 2021. https://eprint.iacr.org/2021/263. 41

CDH+20. Cong Chen, Oussama Danba, Jeffrey Hoffstein, Andreas Hulsing, Joost Rijneveld, John M. Schanck, Peter Schwabe, William Whyte, Zhenfei Zhang, Tsunekazu Saito, Takashi Yamakawa, and Keita Xagawa. NTRU. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 53

CDPR16. Ronald Cramer, Léo Ducas, Chris Peikert, and Oded Regev. Recovering short generators of principal ideals in cyclotomic rings. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 559–585. Springer, Heidelberg, May 2016. 27

CDW17. Ronald Cramer, Léo Ducas, and Benjamin Wesolowski. Short stickelberger class relations and application to ideal-SVP. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 324–348. Springer, Heidelberg, April / May 2017. 27

CF13. Dario Catalano and Dario Fiore. Vector commitments and their applications. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *PKC 2013*, volume 7778 of *LNCS*, pages 55–72. Springer, Heidelberg, February / March 2013. 3, 5, 8, 9, 49

CFG+20. Matteo Campanelli, Dario Fiore, Nicola Greco, Dimitris Kolonelos, and Luca Nizzardo. Incrementally aggregatable vector commitments and applications to verifiable decentralized storage. In Shiho Moriai and Huaxiong Wang, editors, *ASIACRYPT 2020, Part II*, volume 12492 of *LNCS*, pages 3–35. Springer, Heidelberg, December 2020. 3, 8, 9

CG08. Jan Camenisch and Thomas Groß. Efficient attributes for anonymous credentials. In Peng Ning, Paul F. Syverson, and Somesh Jha, editors, *ACM CCS 2008*, pages 345–356. ACM Press, October 2008. 1

CGS14. Peter Campbell, Michael Groves, and Dan Shepherd. Soliloquy: A cautionary tale. Available at http://docbox.etsi.org/Workshop/2014/201410_CRYPTO/S07_Systems_and_Attacks/S07_Groves_Annex.pdf, 2014. 27

CLMQ21. Yilei Chen, Alex Lombardi, Fermi Ma, and Willy Quach. Does fiat-shamir require a cryptographic hash function? In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part IV*, volume 12828 of *LNCS*, pages 334–363, Virtual Event, August 2021. Springer, Heidelberg. 48

CMSZ22. A. Chiesa, F. Ma, N. Spooner, and M. Zhandry. Post-quantum succinct arguments: Breaking the quantum rewinding barrier. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 49–58, Los Alamitos, CA, USA, feb 2022. IEEE Computer Society. 53

CPZ18. Alexander Chepurnoy, Charalampos Papamanthou, and Yupeng Zhang. Edrax: A cryptocurrency with stateless transaction validation. Cryptology ePrint Archive, Report 2018/968, 2018. https://eprint.iacr.org/2018/968. 9

DGNW20. Manu Drijvers, Sergey Gorbunov, Gregory Neven, and Hoeteck Wee. Pixel: Multi-signatures for consensus. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 2093–2110. USENIX Association, August 2020. 8

DHSS20. Yarkın Doröz, Jeffrey Hoffstein, Joseph H. Silverman, and Berk Sunar. MMSAT: A scheme for multimessage multiuser signature aggregation. Cryptology ePrint Archive, Report 2020/520, 2020. https://eprint.iacr.org/2020/520. 41

DPW19. Léo Ducas, Maxime Plançon, and Benjamin Wesolowski. On the shortness of vectors to be found by the ideal-SVP quantum algorithm. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 322–351. Springer, Heidelberg, August 2019. 27

EEE20. Muhammed F Esgin, Oğuzhan Ersoy, and Zekeriya Erkin. Post-quantum adaptor signatures and payment channel networks. In *European Symposium on Research in Computer Security*, pages 378–397. Springer, 2020. 4, 8

Fis05. Marc Fischlin. Communication-efficient non-interactive proofs of knowledge with online extractors. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 152–168. Springer, Heidelberg, August 2005. 15

Fis18. Ben Fisch. PoReps: Proofs of space on useful data. Cryptology ePrint Archive, Report 2018/678, 2018. https://eprint.iacr.org/2018/678. 9

Fis19. Ben Fisch. Tight proofs of space and replication. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part II*, volume 11477 of *LNCS*, pages 324–348. Springer, Heidelberg, May 2019. 3, 9

FS87. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, August 1987. 2

Gen09. Craig Gentry. Fully homomorphic encryption using ideal lattices. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 169–178. ACM Press, May / June 2009. 2

GGM14. Christina Garman, Matthew Green, and Ian Miers. Decentralized anonymous credentials. In *NDSS 2014*. The Internet Society, February 2014. 1

GKK$^+$19. Lorenzo Grassi, Daniel Kales, Dmitry Khovratovich, Arnab Roy, Christian Rechberger, and Markus Schofnegger. Starkad and Poseidon: New hash functions for zero knowledge proof systems. Cryptology ePrint Archive, Report 2019/458, 2019. https://eprint.iacr.org/2019/458. 2

GKW17. Rishab Goyal, Venkata Koppula, and Brent Waters. Lockable obfuscation. In Chris Umans, editor, *58th FOCS*, pages 612–621. IEEE Computer Society Press, October 2017. 2

GLS$^+$21. Alexander Golovnev, Jonathan Lee, Srinath Setty, Justin Thaler, and Riad S. Wahby. Brakedown: Linear-time and post-quantum SNARKs for R1CS. Cryptology ePrint Archive, Report 2021/1043, 2021. https://eprint.iacr.org/2021/1043. 34

GM18. Nicholas Genise and Daniele Micciancio. Faster Gaussian sampling for trapdoor lattices with arbitrary modulus. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part I*, volume 10820 of *LNCS*, pages 174–203. Springer, Heidelberg, April / May 2018. 11

GMNO18. Rosario Gennaro, Michele Minelli, Anca Nitulescu, and Michele Orrù. Lattice-based zk-SNARKs from square span programs. In David Lie, Mohammad Mannan, Michael Backes, and XiaoFeng Wang, editors, *ACM CCS 2018*, pages 556–573. ACM Press, October 2018. 2, 9

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008. 2, 4, 8, 11, 12, 14, 17, 24, 38, 40, 41

Gro16. Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Heidelberg, May 2016. 2

Gro21. Jonathan Gross. Practical snark based vdf, 2021. https://zkproof.org/2021/11/24/practical-snark-based-vdf/. 4

GRWZ20. Sergey Gorbunov, Leonid Reyzin, Hoeteck Wee, and Zhenfei Zhang. Pointproofs: Aggregating proofs for multiple vector commitments. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 2007–2023. ACM Press, November 2020. 3, 9, 52

GVW15. Sergey Gorbunov, Vinod Vaikuntanathan, and Hoeteck Wee. Predicate encryption for circuits from LWE. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 503–523. Springer, Heidelberg, August 2015. 2

GW11. Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 99–108. ACM Press, June 2011. 3, 52

HPS96. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A new high speed public key cryptosystem, 1996. Draft Distributed at Crypto'96, available at http://web.securityinnovation.com/hubfs/files/ntru-orig.pdf. 53

HPS98. Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In *ANTS*, pages 267–288, 1998. 53

HW15. Pavel Hubacek and Daniel Wichs. On the communication complexity of secure function evaluation with long output. In Tim Roughgarden, editor, *ITCS 2015*, pages 163–172. ACM, January 2015. 52, 54

ISW21. Yuval Ishai, Hang Su, and David J. Wu. Shorter and faster post-quantum designated-verifier zkSNARKs from lattices. In Giovanni Vigna and Elaine Shi, editors, *ACM CCS 2021*, pages 212–234. ACM Press, November 2021. 2, 9

Kil92. Joe Kilian. A note on efficient zero-knowledge proofs and arguments (extended abstract). In *24th ACM STOC*, pages 723–732. ACM Press, May 1992. 1, 2

KM10. Neal Koblitz and Alfred Menezes. The brave new world of bodacious assumptions in cryptography. *Notices of the American Mathematical Society*, 57(3):357–365, 2010. 18

KMS$^+$16. Ahmed E. Kosba, Andrew Miller, Elaine Shi, Zikai Wen, and Charalampos Papamanthou. Hawk: The blockchain model of cryptography and privacy-preserving smart contracts. In *2016 IEEE Symposium on Security and Privacy*, pages 839–858. IEEE Computer Society Press, May 2016. 1

KZG10. Aniket Kate, Gregory M. Zaverucha, and Ian Goldberg. Constant-size commitments to polynomials and their applications. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 177–194. Springer, Heidelberg, December 2010. 9

Laa15. Thijs Laarhoven. *Search problems in cryptography: From fingerprinting to lattice sieving*. PhD thesis, Eindhoven University of Technology, 2015. 11

Lis05. Moses Liskov. Updatable zero-knowledge databases. In Bimal K. Roy, editor, *ASIACRYPT 2005*, volume 3788 of *LNCS*, pages 174–198. Springer, Heidelberg, December 2005. 9

LM06. Vadim Lyubashevsky and Daniele Micciancio. Generalized compact Knapsacks are collision resistant. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *ICALP 2006, Part II*, volume 4052 of *LNCS*, pages 144–155. Springer, Heidelberg, July 2006. 12

LM19. Russell W. F. Lai and Giulio Malavolta. Subvector commitments with application to succinct arguments. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part I*, volume 11692 of *LNCS*, pages 530–560. Springer, Heidelberg, August 2019. 3, 4, 5, 8, 49, 52

LMR19. Russell W. F. Lai, Giulio Malavolta, and Viktoria Ronge. Succinct arguments for bilinear group arithmetic: Practical structure-preserving cryptography. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 2057–2074. ACM Press, November 2019. 2

LPR13. Vadim Lyubashevsky, Chris Peikert, and Oded Regev. A toolkit for ring-LWE cryptography. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 35–54. Springer, Heidelberg, May 2013. 11

LPSS14. San Ling, Duong Hieu Phan, Damien Stehlé, and Ron Steinfeld. Hardness of k-LWE and applications in traitor tracing. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 315–334. Springer, Heidelberg, August 2014. 3, 12

LRY16. Benoît Libert, Somindu C. Ramanna, and Moti Yung. Functional commitment schemes: From polynomial commitments to pairing-based accumulators from simple assumptions. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *ICALP 2016*, volume 55 of *LIPIcs*, pages 30:1–30:14. Schloss Dagstuhl, July 2016. 3, 5, 9, 49

LS15. Adeline Langlois and Damien Stehlé. Worst-case to average-case reductions for module lattices. *Designs, Codes and Cryptography*, 75(3):565–599, June 2015. 12

LY10. Benoît Libert and Moti Yung. Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 499–517. Springer, Heidelberg, February 2010. 3

Lyu12. Vadim Lyubashevsky. Lattice signatures without trapdoors. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 738–755. Springer, Heidelberg, April 2012. 38, 39

Ma20. Fermi Ma. Quantum-secure commitments and collapsing hash functions. https://www.cs.princeton.edu/~fermim/talks/collapse-binding.pdf, April 2020. 55

Mic94. Silvio Micali. CS proofs (extended abstracts). In *35th FOCS*, pages 436–453. IEEE Computer Society Press, November 1994. 1

Mic07. Daniele Micciancio. Generalized compact knapsacks, cyclic lattices, and efficient one-way functions. *Comput. Complex.*, 16(4):365–411, 2007. 11, 12

MP12. Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 700–718. Springer, Heidelberg, April 2012. 8, 11

MRK03. Silvio Micali, Michael O. Rabin, and Joe Kilian. Zero-knowledge sets. In *44th FOCS*, pages 80–91. IEEE Computer Society Press, October 2003. 9

MS04. Thom Mulders and Arne Storjohann. Certified dense linear system solving. *J. Symb. Comput.*, 37(4):485–510, 2004. 52

PFH$^+$20. Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. FALCON. Technical report, National Institute of Standards and Technology, 2020. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions. 8, 41

PHGR13. Bryan Parno, Jon Howell, Craig Gentry, and Mariana Raykova. Pinocchio: Nearly practical verifiable computation. In *2013 IEEE Symposium on Security and Privacy*, pages 238–252. IEEE Computer Society Press, May 2013. 1

PHS19. Alice Pellet-Mary, Guillaume Hanrot, and Damien Stehlé. Approx-SVP in ideal lattices with preprocessing. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part II*, volume 11477 of *LNCS*, pages 685–716. Springer, Heidelberg, May 2019. 27

PPS21. Chris Peikert, Zachary Pepin, and Chad Sharp. Vector and functional commitments from lattices. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part III*, volume 13044 of *LNCS*, pages 480–511. Springer, Heidelberg, November 2021. 9

PR06. Chris Peikert and Alon Rosen. Efficient collision-resistant hashing from worst-case assumptions on cyclic lattices. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 145–166. Springer, Heidelberg, March 2006. 12

PXWC21. Yanbin Pan, Jun Xu, Nick Wadleigh, and Qi Cheng. On the ideal shortest vector problem over random rational primes. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part I*, volume 12696 of *LNCS*, pages 559–583. Springer, Heidelberg, October 2021. 27

Reg05. Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005. 2

SE94. Claus-Peter Schnorr and Michael Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Math. Program.*, 66:181–199, 1994. 11

SS11. Damien Stehlé and Ron Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices. In Kenneth G. Paterson, editor, *EUROCRYPT 2011*, volume 6632 of *LNCS*, pages 27–47. Springer, Heidelberg, May 2011. 11, 53

SSTX09. Damien Stehlé, Ron Steinfeld, Keisuke Tanaka, and Keita Xagawa. Efficient public key encryption based on ideal lattices. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 617–635. Springer, Heidelberg, December 2009. 11

Unr16. Dominique Unruh. Computationally binding quantum commitments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 497–527. Springer, Heidelberg, May 2016. 3, 52, 53

Val08. Paul Valiant. Incrementally verifiable computation or proofs of knowledge imply time/space efficiency. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 1–18. Springer, Heidelberg, March 2008. 4, 42, 43

WZ17. Daniel Wichs and Giorgos Zirdelis. Obfuscating compute-and-compare programs under LWE. In Chris Umans, editor, *58th FOCS*, pages 600–611. IEEE Computer Society Press, October 2017. 2

A On Achieving (Functional) Hiding

We discuss potential approaches to modify the VC construction in Section 5 to achieve hiding and functional hiding.

Definition 29 ((Functional) Hiding). *A VC scheme for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is said to be statistically/computationally hiding if for any $\lambda, w, t \in \mathbb{N}$, any $\mathsf{pp} \in \mathsf{Setup}(1^\lambda, 1^w, 1^t)$, and any $\mathbf{x}, \mathbf{x}' \in \mathcal{X}^w$, the distributions*

$$\{\, c : (c, \mathsf{aux}) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}) \,\} \qquad \textit{and} \qquad \{\, c : (c, \mathsf{aux}) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}') \,\}$$

are statistically/computationally indistinguishable.

A VC scheme for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is said to be statistically/computationally functional hiding if there exists a tuple of PPT simulators $\mathcal{S} = (\mathcal{S}_0, \mathcal{S}_1)$ such that, for any $\lambda, w, t \in \mathbb{N}$ and any $(f, \mathbf{x}, y) \in \mathcal{F}_{w,t} \times \mathcal{X}^w \times \mathcal{Y}^t$ satisfying $f(\mathbf{x}) = y$, the distributions

$$\left\{ (\mathsf{pp}, c, \pi) : \begin{array}{r} \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^w, 1^t) \\ (c, \mathsf{aux}) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}) \\ \pi \leftarrow \mathsf{Open}(\mathsf{pp}, f, \mathsf{aux}) \end{array} \right\} \qquad \textit{and} \qquad \left\{ (\mathsf{pp}, c, \pi) : \begin{array}{r} (\mathsf{pp}, \mathsf{td}) \leftarrow \mathcal{S}_0(1^\lambda, 1^w, 1^t) \\ (c, \pi) \leftarrow \mathcal{S}_1(\mathsf{td}, f, y) \end{array} \right\}$$

are statistically/computationally indistinguishable.

In the VC construction in Fig. 3, a commitment of $\mathbf{x}$ is of the form $\langle \mathbf{v}, \mathbf{x} \rangle \bmod q$ as in essentially every lattice-based commitment schemes. A well-known technique for achieving hiding is to commit instead to the concatenation of $\mathbf{x}$ and a short random vector $\mathbf{r}$. If the randomness vector $\mathbf{r}$ has sufficiently many dimensions one could argue that $\langle \mathbf{v}, (\mathbf{x}, \mathbf{r}) \rangle \bmod q$ is statistically close to uniform. This can be done, relying on the regularity lemmas discussed in Section 2.2. Achieving functional hiding requires more work. In the following, we discuss three (potential) approaches on top of introducing $\mathbf{r}$.

Notice that the verification algorithm in Fig. 3 is simply checking that an opening proof $(\mathbf{u}_0, \mathbf{u}_1)$ satisfies two SIS relations. An approach of achieving functional hiding is therefore to replace the opening proof $(\mathbf{u}_0, \mathbf{u}_1)$ with a zero-knowledge proof of knowledge of $(\mathbf{u}_0, \mathbf{u}_1)$. This can be done efficiently using Schnorr-like proofs in the random oracle model, without affecting compactness since the witness $(\mathbf{u}_0, \mathbf{u}_1)$ and the relation that it satisfies are of size independent of (f, y). Due to the use of a random oracle, the resulting scheme may no longer be purely algebraic (depending on how the random oracle is heuristically instantiated) and therefore might not be recursively composed natively. However, in applications where a single party performs the entire recursive composition, it is possible to first recursively compose the non-functional-hiding scheme in Fig. 3, and finish off with a zero-knowledge proof of the final opening proof.

Another approach, related to the first and inspired by [CLMQ21], is to (provably) instantiate the random oracle in a Schnorr-like proof with a function that outputs short preimages of the inputs with respect to a linear function. While this technique preserves the algebraic structure of the scheme, it requires each of the witness components $\mathbf{u}_0$ and $\mathbf{u}_1$ to be a short square matrix instead of a short vector. In other words, to achieve functional hiding using this approach, we need to either introduce dummy relations or prove ℓ openings in batch.

The third approach is to argue directly that $(\mathbf{u}_0, \mathbf{u}_1)$ leaks no information about $\mathbf{x}$. This is intuitively plausible since both $\mathbf{u}_0$ and $\mathbf{u}_1$ consists of linear combinations of Gaussian vectors with coefficients depending on $\mathbf{r}$. Indeed, for $d = 1$, we could apply a Gaussian-version of the Leftover Hash Lemma [AGHS13] and rejection sampling to argue this formally. For $d \geq 2$, unfortunately, the distributions of $\mathbf{u}_0$ and $\mathbf{u}_1$ become much more complicated, making generalising the argument for $d = 1$ to $d \geq 2$ difficult. Furthermore, we remark that this approach relies on making the variance of $\mathbf{u}_0$ and $\mathbf{u}_1$ super-polynomially wide to "smudge" the contribution of $\mathbf{x}$. This means the modulus q would also need to be super-polynomially large.

B Vector Commitments without Knowledge Assumptions

We strip off components for compactness and extractability from our main VC construction in Section 5. The resulting scheme supports the same class of openings. It achieves the weaker notions of succinctness and weak binding but does not rely on any non-falsifiable assumption.

B.1 Definitions

Since our goal is to achieve succinctness, we fix $t = 1$ everywhere and omit it from the syntax. The definition of correctness is modified accordingly. Next, we formalise (weak) binding and succinctness.

Definition 30 ((Weak) Binding). *Let $\rho : \mathbb{N}^3 \to [0, 1]$. A VC scheme for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is said to be weakly ρ-binding if for any pair of PPT adversary $\mathcal{A}$ and any $s, w \in \mathsf{poly}(\lambda)$ it holds that the following expression is upper-bounded by $\rho(\lambda, s, w)$:*

$$\Pr\left[\begin{array}{l|l} \quad\forall\ i \in \{0,1\}, & \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w) \\ \qquad \mathsf{Verify}(\mathsf{pp}_{f_i,y_i}, \mathbf{z}_i, c, \pi_i) = 1, & \big(c, (f_i, \mathbf{z}_i, y_i, \pi_i)_{i=0}^{1}\big) \leftarrow \mathcal{A}(\mathsf{pp}) \\ \wedge\ f_0(\mathbf{z}_0, \cdot) = f_1(\mathbf{z}_1, \cdot) & \forall\ i \in \{0,1\}, \\ \wedge\ y_0(\mathbf{z}_0) \neq y_1(\mathbf{z}_1) & \quad \mathsf{pp}_{f_i,y_i} \leftarrow \mathsf{PreVerify}(\mathsf{pp}, (f_i, y_i)) \end{array}\right].$$

We say that the scheme is weakly binding if it is weakly ρ-binding and $\rho(\lambda, s, w)$ is negligible in λ for any $s, w \in \mathsf{poly}(\lambda)$.

The scheme is said to be ρ-binding if for any PPT adversary $\mathcal{A}$ and $w, t = \mathsf{poly}(\lambda)$ it holds that the following expression is upper-bounded by $\rho(\lambda)$:

$$\Pr\left[\begin{array}{c|l} & \mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w) \\ \big(\forall\ i \in I,\ \mathsf{Verify}(\mathsf{pp}_{f_i,y_i}, \mathbf{z}_i, c, \pi_i) = 1\big) & \big(c, I, (f_i, \mathbf{z}_i, y_i, \pi_i)_{i \in \mathbb{Z}_t}\big) \leftarrow \mathcal{A}(\mathsf{pp}) \\ \wedge\ \neg\left(\exists\ \mathbf{x} \in \mathcal{K}^w,\ \forall\ i \in I,\ f_i(\mathbf{z}_i, \mathbf{x}) = y_i(\mathbf{z}_i)\right) & \forall\ i \in I \\ & \quad \mathsf{pp}_{f_i,y_i} \leftarrow \mathsf{PreVerify}(\mathsf{pp}, (f_i, y_i)) \end{array}\right].$$

We say that the scheme is binding if it is ρ-binding and $\rho(\lambda, s, w)$ is negligible in λ for any $s, w \in \mathsf{poly}(\lambda)$.

Note that in the binding definition the existence of $\mathbf{x}$ is checked over the base field $\mathcal{K}$ rather than the ring $\mathcal{R}$. The reason for this choice will become clear when we discuss the binding property of our construction.

For positional openings [CF13] weak binding and binding are trivially equivalent. Using linear algebra, it is also not difficult to that the equivalence also holds for openings to linear functions over finite fields [LRY16,LM19].[22] The equivalence does not seem to hold, however, for openings to linear functions over rings nor for high-degree openings over rings or fields.

Definition 31 (Succinctness). *A VC scheme for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ is said to be succinct if there exists $p(\lambda, s, w) \in \mathsf{poly}(\lambda, \log s, \log w)$ such that for any $\lambda, s, w \in \mathbb{N}$, any $\mathsf{pp} \in \mathsf{Setup}(1^\lambda, 1^s, 1^w)$, any $(f, \mathbf{z}, \mathbf{x}, y) \in \mathcal{F}_{s,w} \times \mathcal{X}^s \times \mathcal{X}^w \times \mathcal{Y}_s$, 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 $|c| \leq p(\lambda, s, w)$ and $|\pi| \leq p(\lambda, s, w)$, where $|\cdot|$ denotes the description size.*

B.2 Construction

A formal description of the stripped-down construction is in Fig. 7. The proof of correctness is completely analogous to that of Theorem 3 and is therefore omitted.

Theorem 7. *For $d = O(1)$, $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$ and*

$$\delta = 3 \cdot (s + d)^d \cdot (w + d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2},$$

the VC construction in Fig. 7 is correct.

Theorem 8. *The construction of vector commitments for $(\mathcal{F}, \mathcal{X}, \mathcal{Y})$ in Fig. 7 is weakly-binding if $\ell \geq \mathsf{lhl}(\mathcal{R}, \eta, q, \beta)$, $\beta \geq \alpha$, and the k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,w,\mathcal{G},1,\mathcal{D},\mathcal{T},\beta,2\delta}$ assumption holds, where $\mathcal{D}$ is such that the distribution*

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

[22] In [LM19], the generic group model is used to prove the binding property of the compact linear map commitment construction. If the compactness requirement is dropped, binding could be proven in the plain model.

```
$\mathsf{Setup}(1^\lambda, 1^s, 1^w)$
    $\mathbf{v} \leftarrow\!\$\, (\mathcal{R}_q^\times)^w$
    $(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^\eta, 1^\ell, q, \mathcal{R}, \beta)$
    $\mathbf{t} \leftarrow\!\$\, \mathcal{R}_q^\eta$
    $\mathbf{u}_g \leftarrow \mathsf{SampPre}(\mathsf{td}, g(\mathbf{v}) \cdot \mathbf{t}, \beta),\ \forall g \in \mathcal{G}$
    return $\mathsf{pp} := \left(\mathbf{A}, \mathbf{t}, (\mathbf{u}_g)_{g \in \mathcal{G}}, \mathbf{v}\right)$

$\mathsf{Com}(\mathsf{pp}, \mathbf{x})$
    $c := \langle \mathbf{v}, \mathbf{x} \rangle \bmod q$
    for $\mathbf{e} \in \bigcup_{k \in [d]} \mathcal{E}_k$ do
        $\mathbf{u}_\mathbf{e} := d! \cdot \sum_{\mathbf{e}' \in \mathcal{E}_k \setminus \{\mathbf{e}\}} \frac{\binom{k}{\mathbf{e}'}}{\binom{k}{\mathbf{e}}} \cdot \mathbf{x}^{\mathbf{e}'} \cdot \mathbf{u}_{\mathbf{X}^{\mathbf{e}'-\mathbf{e}}}$
    $\mathsf{aux} := (\mathbf{u}_\mathbf{e})_{\mathbf{e} \in \bigcup_{k \in [d]} \mathcal{E}_k}$
    return $(c, \mathsf{aux})$

$\mathsf{PreVerify}(\mathsf{pp}, (f, y))$
    if $(f, y) \notin \mathcal{F}_{s,w} \times \mathcal{Y}_s$ then return $\perp$
    $\hat{f}(\mathbf{Z}, C) := d! \cdot \left( \sum_{k=1}^{d} \sum_{\mathbf{e} \in \mathcal{E}_k} \binom{k}{\mathbf{e}}^{-1} \cdot f_\mathbf{e}(\mathbf{Z}) \cdot \mathbf{v}^{-\mathbf{e}} \cdot C^k - y(\mathbf{Z}) \right)$
    $\mathsf{pp}_{f,y} := \left(\mathbf{A}, \mathbf{t}, \hat{f}\right)$
    return $\mathsf{pp}_{f,y}$

$\mathsf{Open}(\mathsf{pp}, f, \mathbf{z}, \mathsf{aux})$
    $\mathbf{u} := \sum_{k=1}^{d} \sum_{\mathbf{e} \in \mathcal{E}_k} f_\mathbf{e}(\mathbf{z}) \cdot \mathbf{u}_\mathbf{e}$
    return $\pi := \mathbf{u}$

$\mathsf{Verify}(\mathsf{pp}_{f,y}, \mathbf{z}, c, \pi)$
    $b_0 := \left( \mathbf{A}\mathbf{u} \overset{?}{=} \hat{f}(\mathbf{z}, c) \cdot \mathbf{t} \bmod q \right)$
    $b_1 := \left( \|\mathbf{u}\| \overset{?}{\leq} \delta \right)$
    return $b_0 \wedge b_1$
```

Fig. 7. Stripped-Down VC Construction.

is statistically close to the distribution

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

Proof. Let $\mathcal{A}$ be an adversary against the weakly binding property of the construction in Fig. 7. We construct an algorithm $\mathcal{B}$ for the k-M-$\mathsf{ISIS}_{\mathcal{R}_q,\eta,\ell,w,\mathcal{G},1,\mathcal{D},\mathcal{T},\beta,2\delta}$ problem. Our algorithm $\mathcal{B}$ inputs a problem instance $\left(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{\, \mathbf{u}_g \,\}_{g \in \mathcal{G}}\right)$, sets $\mathsf{pp} := \left(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{\, \mathbf{u}_g \,\}_{g \in \mathcal{G}}\right)$, and runs $\mathcal{A}$ on pp to obtain a tuple $(c, (f_i, \mathbf{z}_i, y_i, \mathbf{u}_i)_{i=0}^{1})$. Our algorithm $\mathcal{B}$ outputs $(s^*, \mathbf{u}_{g^*}) = (d! \cdot (y_1(\mathbf{z}_1) - y_0(\mathbf{z}_0)), \mathbf{u}_0 - \mathbf{u}_1)$.

Suppose $\mathcal{A}$ is a successful adversary against the weak-binding property of our VC construction. By our assumption on $\mathcal{D}$, the distribution of the public parameters pp passed to $\mathcal{A}$ by $\mathcal{B}$ is statistically close to that generated by Setup. Therefore, with non-negligible probability, the tuple that $\mathcal{A}$ returns to $\mathcal{B}$ satisfies

$$\begin{cases} \mathbf{A} \cdot \mathbf{u}_i = \hat{f}_i(\mathbf{z}_i, c) \cdot \mathbf{t} \bmod q, \\ \|\mathbf{u}_i\| \leq \delta. \end{cases}$$

for $i \in \{0, 1\}$ with $f_0(\mathbf{z}_0, \cdot) = f_1(\mathbf{z}_1, \cdot)$ but $y_1(\mathbf{z}_1) \neq y_0(\mathbf{z}_0)$, which implies $\mathbf{A} \cdot \mathbf{u}_{g^*} = s^* \cdot \mathbf{t} \bmod q$, $0 < \|s^*\| \leq 2\,\delta$, and $\|\mathbf{u}_{g^*}\| \leq 2\,\delta$. □

Theorem 9. *For $n \in \mathsf{poly}(\lambda)$, $q, \delta \in \mathsf{poly}(\lambda, s, w)$, and $\ell \in \Theta(\log q) = \mathsf{polylog}(\lambda, s, w)$, covering the choices of parameters in Theorems 7 and 8, the VC construction in Fig. 7 is succinct.*

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

$$d = O(1), \quad \beta \geq \alpha,$$

$$\begin{aligned}
&\delta = 3 \cdot (s+d)^d \cdot (w+d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2},\\
&q \approx \delta \cdot n \cdot \log n, \ \textit{and}\\
&\ell = \mathsf{lhl}(\mathcal{R}, \eta, q, \beta) \approx 2 \log_\beta q,
\end{aligned}$$

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

Proof. For the general case, we observe that a commitment $c \in \mathcal{R}_q$ is of description size $n \log q \in \mathsf{poly}(\lambda, \log s, \log w)$, and an opening proof $\mathbf{u}$ is of description size $n \cdot \ell \cdot \log \delta \in \mathsf{poly}(\lambda, \log s, \log w)$.

For the concrete case, we have

$$\begin{aligned}
\delta &= 3 \cdot (s+d)^d \cdot (w+d)^{2d} \cdot \alpha^{2d+1} \cdot \beta \cdot \gamma_{\mathcal{R}}^{2d+2} = O(s^{3d} \cdot \alpha^{2d+1} \cdot \beta \cdot n^{2d+2}),\\
q &= \delta \cdot n \cdot \log n = O(s^{3d} \cdot \alpha^{2d+1} \cdot \beta \cdot n^{2d+3} \cdot \log n),\\
\log \delta, \log q &= O(\log s + \log \beta),\\
\ell &= 2 \log q / \log \beta = O((\log s + \log \beta)/\log \beta),\\
|c| &= n \cdot \log q = O(n \log s), \ \text{and}\\
|\mathbf{u}| &= n \cdot \ell \cdot \log \delta\\
&= n \cdot O((\log s + \log \beta)/\log \beta) \cdot O(\log s + \log \beta)\\
&= O(n \cdot (\log s + \log \beta)^2/\log \beta).
\end{aligned}$$

□

B.3 On Binding

We study to what extent binding can be achieved without relying on non-falsifiable assumptions.

In the following informal discussion we omit the public input $\mathbf{z}$ for readability. As mentioned previously, in the case where $\mathcal{F}$ consists of only position maps, then weak binding is trivially equivalent to binding. This is because, if f_i are position maps, i.e. $f_i(\mathbf{x}) = x_i$, for $i \in I$ then the only way to force that no $\mathbf{x} \in \mathcal{K}^w$ satisfies $f_i(\mathbf{x}) = y_i$ for all $i \in I$ is to set $f_{i'} = f_{i''}$ but $y_{i'} \neq y_{i''}$ for some distinct $i', i'' \in I$.

In fact, even if $\mathcal{F}$ consists of only linear maps, i.e. $d = 1$, the equivalence between weak binding and binding still holds without considering the norm bound constraint, e.g. when the linear maps are defined over a finite field. Indeed, suppose that $f_i(\mathbf{x}) = y_i$ for all $i \in I$ is not satisfiable by any $\mathbf{x} \in \mathcal{K}^w$, then by Gaussian elimination one can find a coefficient vector $\mathbf{r}' \in \mathcal{K}^I$ such that $\sum_{i \in I} r'_i f_i \equiv 0$ and $\sum_{i \in I} r'_i y_i = 1$. Multiplying $\mathbf{r}'$ by the least common multiple Δ of the denominators in $\mathbf{r}'$ to obtain $\mathbf{r} \in \mathcal{R}^I$, we have $\sum_{i \in I} r_i f_i \equiv 0$ and $\sum_{i \in I} r_i y_i = \Delta$. Since the verification algorithm Verify is linear in (f, y), we obtain openings for (f_i, y_i) and $(f_i, y_i + \Delta)$.

In the lattice setting, however, we need to argue that Δ is not too large relative to (a large enough) q, so that we can use the technique in the proof of Theorem 8 to turn an adversary against binding into an algorithm for solving certain k-M-ISIS problems. The following theorem states that binding can be achieved for $d = 1$ and an exponentially large q.

Theorem 10. *In addition to the assumptions made in Theorem 8, let $d = 1$, $\delta^* := |I| \cdot \gamma_{\mathcal{R}} \cdot \delta \cdot \nu^\nu \cdot \alpha^{2\nu}$, and $q \geq \omega\left(|I| \cdot \gamma_{\mathcal{R}} \cdot \delta \cdot \nu^\nu \cdot \alpha^{2\nu}\right)$ where $\nu := (w+1) \cdot n$. If the VC construction in Fig. 7 is weakly-binding for δ^* then it is also binding for δ.*

Proof. Suppose there exists a PPT adversary $\mathcal{A}$ against binding, we construct a PPT adversary $\mathcal{B}$ against weak binding as follows. Our adversary $\mathcal{B}$ receives the public parameters $\left(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{\, \mathbf{u}_g \,\}_{g \in \mathcal{G}}\right)$ and forwards it to $\mathcal{A}$. By assumption, $\mathcal{A}$ outputs a tuple $\left(c, I, \{\, f_i, \mathbf{z}_i, y_i, \mathbf{u}_i \,\}_{i \in I}\right)$ which satisfies the following with non-negligible probability:

(i) For all $i \in I$, $\mathbf{A} \cdot \mathbf{u}_i \equiv \hat{f}_i(\mathbf{z}_i, c) \bmod q$.
(ii) For all $i \in I$, $\|\mathbf{u}_i\| \leq \delta$.
(iii) There does not exist $\mathbf{x} \in \mathcal{K}^w$ such that, for all $i \in I$, $f_i(\mathbf{z}_i, \mathbf{x}) = y_i$.

Since f_i is a homogeneous linear polynomial, we have $\hat{f}_i(\mathbf{z}_i, \cdot) = f_i(\mathbf{z}_i, \cdot) - y(\mathbf{z}_i)$ and $f_i(\mathbf{z}_i, \cdot)$ can be represented by a vector $\mathbf{f}_i \in \mathcal{R}^w$ such that $f_i(\mathbf{z}_i, \mathbf{x}) = \langle \mathbf{f}_i, \mathbf{x} \rangle$ for any $\mathbf{x} \in \mathcal{K}^w$. Let $\mathbf{F}$ be the matrix with the i-th column being $\mathbf{f}_i$, $\mathbf{U}$ be the matrix with the i-th column being $\mathbf{u}_i$, and $\mathbf{y} = (y_i(\mathbf{z}_i))_{i\in I}$. Consequently, we can rewrite the equations in Item 1 above as

$$\mathbf{A}^{\mathrm{T}} \cdot \mathbf{U} \equiv \mathbf{t} \cdot \left(c \cdot \bar{\mathbf{v}}^{\mathrm{T}} \; -1\right) \begin{pmatrix} \mathbf{F} \\ \mathbf{y}^{\mathrm{T}} \end{pmatrix} \bmod q.$$

By assumption there exists an $\mathbf{r}' \in \mathcal{K}^{|I|}$ s.t. $\begin{pmatrix} \mathbf{F} \\ \mathbf{y}^{\mathrm{T}} \end{pmatrix} \cdot \mathbf{r}' = (0, \ldots, 0, 1)$. Thus, we have $\mathbf{r} := \mathit{\Delta} \cdot \mathbf{r}' \in \mathcal{R}^{|I|}$ s.t. $\begin{pmatrix} \mathbf{F} \\ \mathbf{y}^{\mathrm{T}} \end{pmatrix} \cdot \mathbf{r} = (0, \ldots, 0, \mathit{\Delta})$ where $\mathit{\Delta}$ is the least common multiple of the denominators in $\mathbf{r}'$. Note that a solution in $\mathcal{R}$ maps to a solution over $\mathbb{Z}$ by the map $g \mapsto \mathrm{rot}(g)$. To bound $\|\mathbf{r}\|$ and $\mathit{\Delta}$, assume $|I| = w + 1$, which represents the worst case, and apply known bounds for solutions over $\mathbb{Z}$ [MS04, Fact 25]: $\mathit{\Delta} \leq \nu^{\nu/2} \cdot \alpha^{\nu}$ and $\|\mathbf{r}\| \leq \nu^{\nu/2} \cdot \alpha^{\nu-1} \cdot \mathit{\Delta}$.

Let $\mathbf{u}_0' := \mathbf{U} \cdot \mathbf{r}$. We have

$$\begin{aligned} \mathsf{Verify}(\mathsf{pp}_{f_0,y_0}, \mathbf{z}_0, c, \mathbf{u}_0) &= 1 \\ \mathsf{Verify}(\mathsf{pp}_{f_0,y_0+\mathit{\Delta}}, \mathbf{z}_0, c, \mathbf{u}_0') &= 1 \\ \|\mathbf{u}_0\| < \|\mathbf{u}_{|I|}\| &\leq \delta^* \end{aligned}$$

but $y_0 \neq y_0 + \mathit{\Delta}$. □

We next discuss why proving binding in the case $d > 1$ from falsifiable assumptions seems unlikely. Indeed, if we were given a compact and binding VC for degree-d openings for $d \geq 2$, we can construct a SNARG for the NP-complete language of degree-d polynomial maps satisfiability (Section 7), where a SNARG is almost a SNARK but only satisfies soundness instead of knowledge soundness. Due to the impossibility result of Gentry and Wichs [GW11], who showed that certain flavour of SNARG requires non-falsifiable assumption or non-black-box reduction, we obtain the same impossibility for compact and binding VC with openings to non-linear polynomial maps.

B.4 On Compactness

We discuss the difficulty of achieving compactness without relying on the knowledge k-M-ISIS assumption.

For VC constructions where the verification equation is linear in the opening proof, such as the constructions presented in Section 5.1 and Appendix B.2, a natural strategy to achieve compactness is to aggregate multiple opening proofs using a random linear combination. Instantiating the strategy involves deciding how the random coefficients of the linear combination are generated.

For schemes where the verification equation is defined over prime-order cyclic groups, provably binding ways of instantiating the strategy includes (i) embedding the random coefficients in the public parameters and prove soundness in the generic [LM19] or algebraic [GRWZ20] group model, (ii) making the verification interactive and let the verifier sample the coefficients, or (iii) generate the coefficients using a random oracle. The proofs of binding in all three approaches rely crucially on the fact that Vandermonde matrices defined by distinct elements in a finite field are always invertible.

In the lattice setting, the random coefficients need to be chosen from a subtractive set, i.e. a set where the difference between any pairs of distinct elements is always invertible, for a similar proof strategy to work (see, e.g. [AL21]). Unfortunately, it has been shown [AL21] that over many cyclotomic rings $\mathcal{R}$, the size of (even relaxed variants of) subtractive sets is at most $O(n)$, which is insufficient for aggregating an unbounded polynomial number t of opening proofs into a single proof of size poly-logarithmic in t.

B.5 Post-Quantum Security

We analyse the security of our stripped-down construction against quantum attackers. We show that our construction, viewed as an ordinary commitment scheme, satisfies the notion of collapsing [Unr16]. This is done is two steps: First, we show that our VC scheme satisfies the notion of somewhere statistically binding (SSB) [HW15]. Next, we rely on a previous result, reproduced in Appendix B.5 that an SSB VC is also collapsing.

```
$\mathsf{KGen}(1^\lambda)$
$f' \leftarrow \mathsf{SampD}(1^1, 1^1, \mathcal{R}, \beta); f := p \cdot f' + 1$
if $f \notin \mathcal{R}_q^\times$ resample
$g \leftarrow \mathsf{SampD}(1^1, 1^1, \mathcal{R}, \beta)$if $g \notin \mathcal{R}_q^\times$ resample
return $\mathsf{pk} := h = p \cdot g/f$, $\mathsf{sk} := f$
```

```
$\mathsf{Enc}(\mathsf{pk}, m \in \mathcal{R}_p)$
$(s, e) \leftarrow \mathsf{SampD}(1^1, 1^2, \mathcal{R}, \beta')$
return $c := h \cdot s + p \cdot e + m$

$\mathsf{Dec}(\mathsf{sk}, c)$
return $m := (f \cdot c \bmod q) \bmod p$
```

Fig. 8. NTRU Encryption. n, q, p, β, β' are parameters $\in \mathsf{poly}(\lambda)$.

Assumptions. For showing post-quantum security, we will rely on the pseudorandomness and correctness of the NTRU encryption scheme [HPS96,HPS98].

Definition 32 (NTRU Encryption Assumption). *Consider the NTRU encryption scheme parameterised by $n, q, p, \beta, \beta' \in \mathsf{poly}(\lambda)$ as given in Fig. 8.*

(i) We say that NTRU ciphertexts are w-pseudorandom if the following expression is negligible in λ for any PPT $\mathcal{A}$, arbitrary $m_i \in \mathcal{R}_p$ for $i \in \mathbb{Z}_w$, and $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$.

$$\Pr\left[\mathcal{A}(\mathsf{pk}, \{ c_i \}_{i \in \mathbb{Z}_w}) = 1 \middle| c_i \leftarrow \mathsf{Enc}(\mathsf{pk}, m_i)\right] - \Pr\left[\mathcal{A}(\mathsf{pk}, \{ u_i \}_{i \in \mathbb{Z}_w}) = 1 \middle| u_i \leftarrow\!\$\, \mathcal{R}_q\right].$$

(ii) Let $w, \alpha \in \mathsf{poly}(\lambda)$ be additional parameters. We say that NTRU decryption is (w, α)-correct if the following expression is negligible in λ for any PPT $\mathcal{A}$, $m_i \in \{0, 1\}$ for $i \in \mathbb{Z}_w$, and $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$.

$$\Pr\left[\begin{matrix} x_i \leftarrow \mathcal{A}(\mathsf{pk}, \{ c_i \}_{i \in \mathbb{Z}_w}); \\ \forall i \in \mathbb{Z}_w,\ \|x_i\| \leq \alpha \wedge \mathsf{Dec}(\mathsf{sk}, \sum x_i \cdot c_i) \neq \sum\limits_{i \in \mathbb{Z}_w} x_i \cdot m_i \end{matrix} \middle| c_i \leftarrow \mathsf{Enc}(\mathsf{pk}, m_i)\right].$$

The NTRU encryption assumption holds for the parameters n, q, w, α if there exist $p, \beta, \beta' \in \mathsf{poly}(\lambda)$ such that NTRU ciphertexts are w-pseudorandom and NTRU decryption is (w, α)-correct for these parameters.

The pseudorandomness of NTRU ciphertexts can be reduced to the decision NTRU assumption (asserting that NTRU public keys are pseudorandom) and the Ring-LWE assumption [CDH+20]. The decisional NTRU assumption can be dropped when $\beta \approx \sqrt{q}$ [SS11]. For any $\alpha \in \mathsf{poly}(\lambda)$, there exist parameters $n, q, p, \beta, \beta' \in \mathsf{poly}(\lambda)$ such that NTRU decryption is unconditionally correct [CDH+20].

Quantum Information. A (pure) quantum state is a unit vector $|\psi\rangle$ in a complex Hilbert space $\mathcal{H}$. Hilbert spaces are commonly divided into registers, e.g., $\mathcal{H} = \mathcal{H}_0 \otimes \mathcal{H}_1$. A unitary operation is represented by a complex matrix U such that $UU^\dagger = \mathbf{I}$. The operation U transforms the pure state $|\psi\rangle$ to the pure state $U|\psi\rangle$. In this work, a quantum adversary is a family of quantum circuits $\{\mathcal{A}_\lambda\}_{\lambda \in \mathbb{N}}$ represented classically using some standard universal gate set. A quantum adversary is polynomial-size if there exists a polynomial p and some $\lambda_0 \in \mathbb{N}$ such that for all $\lambda > \lambda_0$ it holds that $|\mathcal{A}_\lambda| \leq p(\lambda)$.

Collapsing. It is well known that the classical (computational) notion of binding is not meaningful against quantum attackers [Unr16,ARU14]. For compressing commitment schemes, where statistical binding is simply impossible, a more useful notion is that of *collapsing*. In the following, we adapt the definition of collapsing for hash functions [Unr16] to one for VCs. Essentially, our definition requires the commitment algorithm of the VC to be collapsing when viewed as a hash function. Note that our definition is weaker than that of [CMSZ22], who requires the collapsing property to hold with respect to positional openings.

Definition 33 (Collapsing). *A VC scheme Γ is said to be collapsing if for any QPT adversary $\mathcal{A}$ and any $w = \mathsf{poly}(\lambda)$ it holds that*

$$\left| \begin{matrix} \Pr\left[\mathsf{CollapsExp}^0_{\Gamma,\mathcal{A}}(1^\lambda, 1^s, 1^w, 1^t) = 1\right] \\ - \Pr\left[\mathsf{CollapsExp}^1_{\Gamma,\mathcal{A}}(1^\lambda, 1^s, 1^w, 1^t) = 1\right] \end{matrix} \right| \leq \mathsf{negl}(\lambda).$$

where the experiment $\mathsf{CollapsExp}^b_{\Gamma,\mathcal{A}}(1^\lambda, 1^s, 1^w, 1^t)$ is defined as follows:

- *The challenger samples* $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$ *and sends it to* $\mathcal{A}$.
- $\mathcal{A}$ *replies with a classical message* c *(a commitment) and a quantum register* $\mathcal{V}$*, which contains strings* $\mathbf{x} \in \mathcal{Z}^w$.
- *Let* U *be the unitary that acts on* $\mathcal{V}$ *and some ancilla register and computes the bit* $\left(c \overset{?}{=} \mathsf{Com}(\mathsf{pp}, \mathcal{V})\right)$*, where the auxiliary output* aux *is suppressed. The challenger applies* U *to* $\mathcal{V}$ *and measures the ancilla register containing the output bit in the computational basis. If such bit is* 0 *abort the experiment, else apply* $U^\dagger$.
- *If* $b = 0$ *the challenger does nothing. If* $b = 1$ *the challenger measures the register* $\mathcal{V}$ *in the computational basis.*
- *Return the (possibly measured) register* $\mathcal{V}$ *to* $\mathcal{A}$.
- $\mathcal{A}$ *returns a bit which is also the output of the experiment.*

Somewhere Statistically Binding (SSB) We introduce the notion of somewhere statistically binding (SSB) [HW15] for VCs. Similar to the treatment for collapsing above, our definition of SSB essentially requires that the commitment algorithm of the VC to be SSB as an ordinary commitment.

Definition 34 (Somewhere Statistically Binding (SSB)). *A VC scheme* Γ *is said to be somewhere statistically binding (SSB) if there exists a binding setup algorithm* $\mathsf{pp} \leftarrow \mathsf{BSetup}(1^\lambda, 1^s, 1^w, 1^t, i)$*, which takes an additional input* $i \in \mathbb{Z}_w$*, such that the following properties are satisfied:*

- *(Mode Indistinguishability) For all* $\lambda \in \mathbb{N}$*, all* $s, w, t = \mathsf{poly}(\lambda)$*, and all* $i \in \mathbb{Z}_w$ *the following distributions are computationally indistinguishable*
$$\mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t) \approx \mathsf{BSetup}(1^\lambda, 1^w, 1^t, i).$$
- *(SSB) For all* $\lambda \in \mathbb{N}$*,* $s, w, t = \mathsf{poly}(\lambda)$*,* $i \in \mathbb{Z}_w$*, and* $\mathsf{pp} \in \mathsf{BSetup}(1^\lambda, 1^s, 1^w, 1^t, i)$*,*
$$\Pr\left[\exists \mathbf{x}_0, \mathbf{x}_1 \in \mathcal{X}^w : \begin{array}{r}(c_0, \mathsf{aux}_0) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}_0)\\ \wedge\ (c_1, \mathsf{aux}_1) \leftarrow \mathsf{Com}(\mathsf{pp}, \mathbf{x}_1)\\ \wedge\ c_0 = c_1 \qquad\qquad\qquad\\ \wedge\ x_{0,i} \neq x_{1,i} \qquad\qquad\quad\end{array}\right] \leq \mathsf{negl}(\lambda).$$

Our central technique of achieving SSB is to replace entries of the public vector $\mathbf{v}$ with ciphertexts of (the provable variant of) the NTRU encryption scheme. Concretely, we construct $\mathsf{BSetup}(1^\lambda, 1^s, 1^w, 1^t, i)$ by setting v_i to be an NTRU ciphertext encrypting 1, while setting v_j to be an NTRU ciphertext encrypting 0 for all $j \neq i$. Since NTRU ciphertexts are indistinguishable from uniformly random $\mathcal{R}_q$ elements, mode indistinguishability follows. For the main SSB property, we notice that if two vectors $\mathbf{x}_0, \mathbf{x}_1 \in \mathcal{X}^w$ generate the same commitment, we have $\langle \mathbf{v}, \mathbf{x}_0 \rangle = \langle \mathbf{v}, \mathbf{x}_1 \rangle$. Since the NTRU encryption scheme is linearly homomorphic, the left-hand-side is a ciphertext encrypting $x_{0,i}$, while the right-hand-side is encrypting $x_{1,i}$. The correctness of NTRU then forces $x_{0,i} = x_{1,i}$.

Theorem 11. *If the NTRU encryption assumption (Definition 32) holds for* n, q, w, α*, the VC construction* Γ *in Fig. 7 is SSB.*

Proof. Following the treatment in Appendix B, we assume without loss of generality that $t = 1$ and omit the input 1^t to the setup algorithms. We begin by constructing the binding setup algorithm $\mathsf{BSetup}(1^\lambda, 1^s, 1^w, i)$ as follows, where $\mathbf{m}_i$ denotes the i-th unit vector.

Mode Indistinguishability. Fix any $i \in \mathbb{Z}_w$. To show that $\mathsf{Setup}(1^\lambda, 1^s, 1^w) \approx \mathsf{BSetup}(1^\lambda, 1^s, 1^w, i)$ it suffices to show that the distributions of $\mathbf{v}$ induced by the two algorithms are indistinguishable, which is immediately implied by the assumption that NTRU ciphertexts are w-pseudorandom.

SSB. Fix any $i \in \mathbb{Z}_w$ and $\mathsf{pp} \in \mathsf{BSetup}(1^\lambda, 1^s, 1^w, i)$. We show that if $\mathbf{x}_0, \mathbf{x}_1 \in \mathcal{X}^w$ satisfy $\mathsf{Com}(\mathsf{pp}, \mathbf{x}_0) = \mathsf{Com}(\mathsf{pp}, \mathbf{x}_1)$ (suppressing aux), then it holds that $x_{0,i} = x_{1,i}$. Let sk be the NTRU secret key generated when generating the pp. Since $\mathbf{x}_0, \mathbf{x}_1 \in \mathcal{X}^w$, we have that $\|\mathbf{x}_0\| \leq \alpha$ and $\|\mathbf{x}_1\| \leq \alpha$. Let $c := \mathsf{Com}(\mathsf{pp}, \mathbf{x}_b) = \langle \mathbf{v}, \mathbf{x}_b \rangle \bmod q$. By the assumption that NTRU decryption is (w, α)-correct, it holds that $\mathsf{Dec}(\mathsf{sk}, c) = \langle \mathbf{m}_i, \mathbf{x}_i \rangle = x_{b,i}$ for $b \in \{0, 1\}$. Consequently, $x_{0,i} = x_{1,i}$. □

```
$\mathsf{BSetup}(1^\lambda, 1^s, 1^w, i)$
$(\mathbf{A}, \mathsf{td}) \leftarrow \mathsf{TrapGen}(1^\eta, 1^\ell, q, \mathcal{R}, \beta)$
$\mathbf{t} \leftarrow \mathcal{T}$
$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KGen}(1^\lambda)$
$v_i \leftarrow \mathsf{Enc}(\mathsf{pk}, 1)$
$v_j \leftarrow \mathsf{Enc}(\mathsf{pk}, 0),\ \forall j \in \mathbb{Z}_w \setminus \{ i \}$
$\mathbf{v} := (v_j : j \in \mathbb{Z}_w)$
$\mathbf{u}_g \leftarrow \mathsf{SampPre}(\mathsf{td}, g(\mathbf{v}) \cdot \mathbf{t}, \beta),\ \forall g \in \mathcal{G}$
return $\mathsf{pp} := \big(\mathbf{A}, \mathbf{t}, \mathbf{v}, \{ \mathbf{u}_g \}_{g \in \mathcal{G}}\big)$
```

SSB Implies Collapsing. We now show that an SSB VC is also collapsing. This implication was first shown in an oral presentation of Ma [Ma20] but, to the best of our knowledge, it does not formally appear in any prior work. For completeness, we present the proof below.

Theorem 12. *An SSB VC Γ is collapsing.*

Proof. Let $\mathcal{V} = \mathcal{V}_0 \otimes \cdots \otimes \mathcal{V}_{w-1}$ denote the registers sent by the attacker in the collapsing experiment. The proof consists of a hybrid argument where we define the hybrids H_i for $i \in \{ 0, 1, \dots, w \}$ to be the same experiment as $\mathsf{CollapsExp}^b_{\Gamma,\mathcal{A}}$ except that the challenger measures the registers $(\mathcal{V}_0, \dots, \mathcal{V}_{i-1})$. Note that the hybrid H_0 corresponds to the original experiment with the bit $b = 0$, whereas hybrid H_w is identical to the original experiment with the bit set to $b = 1$. It therefore suffices to show that for all $i = [w]$ the hybrids H_{i-1} and H_i produce distributions that are computationally close. This is done by defining the following intermediate distributions:

- Hybrid G_0: This experiment is identical to H_{i-1}.
- Hybrid G_1: In this hybrid we compute the public parameters as $\mathsf{pp} \leftarrow \mathsf{BSetup}(1^\lambda, 1^s, 1^w, 1^t, i)$. By the mode indistinguishability of the setup algorithm, we can conclude that the view of the adversary is computationally indistinguishable from that induced by the previous hybrid.
- Hybrid G_2: This hybrid is identical to the previous one, except that the challenger additionally measures the i-th register $\mathcal{V}_i$. Let us analyse the content of the registers after the third step of the experiment. If the challenger aborts, then the adversary is not returned any register and therefore the views are trivially identical. On the other hand, if the challenger does not abort, then the state in the $\mathcal{V}$ register consists of
$$\chi = \sum_{\mathbf{x} \text{ s.t. } c = \mathsf{Com}(\mathsf{pp}, \mathbf{x})} \alpha_{\mathbf{x}} |\mathbf{x}\rangle$$
where the amplitudes are suitably normalized and c is the classical string returned by $\mathcal{A}$. By the SSB property of the VC, it holds that, except with negligible probability, all pre-images of c have the same i-bit $\mathbf{x}_i$. Thus we can rewrite (up to a rearrangement of the registers)
$$\chi = \sum_{\mathbf{x} \text{ s.t. } c = \mathsf{Com}(\mathsf{pp}, \mathbf{x})} \alpha_{\mathbf{x}} |\mathbf{x}\rangle = |\mathbf{x}_i\rangle \otimes \sum_{\mathbf{x} \text{ s.t. } c = \mathsf{Com}(\mathsf{pp}, \mathbf{x})} \alpha_{\mathbf{x}} |\mathbf{x}_{-i}\rangle$$
where $\mathbf{x}_{-i}$ denotes the vector $\mathbf{x}$ without the i-th bit $\mathbf{x}_i$. It follows that measuring the register $\mathcal{V}_i$ returns $\mathbf{x}_i$ with probability 1 and it does not disturb the state. Thus the adversary's view of this hybrid is statistically close to that of the previous one.
- Hybrid G_3: This is identical to the previous experiment, except that we undo the modification done in the G_1 (i.e., we sample the public parameters as $\mathsf{pp} \leftarrow \mathsf{Setup}(1^\lambda, 1^s, 1^w, 1^t)$). Computational indistinguishability follows by the same argument.

The proof is concluded by observing that the experiment G_3 is identical to H_i.