Efficient Hybrid Exact/Relaxed Lattice Proofs and Applications to Rounding and VRFs

Muhammed F. Esgin[1,2], Ron Steinfeld[1], Dongxi Liu[2], and Sushmita Ruj[3]

[1] Monash University, Australia
[2] CSIRO's Data61, Australia
[3] University of New South Wales, Australia

Abstract. In this work, we study *hybrid exact/relaxed zero-knowledge proofs* from lattices, where the proved relation is exact in one part and relaxed in the other. Such proofs arise in important real-life applications such as those requiring verifiable PRF evaluation and have so far not received significant attention as a standalone problem.

We first introduce a general framework, LANES$^+$, for realizing such hybrid proofs efficiently by combining standard *relaxed* proofs of knowledge RPoK and the LANES framework (due to a series of works in Crypto'20, Asiacrypt'20, ACM CCS'20). The latter framework is a powerful lattice-based proof system that can prove exact linear and multiplicative relations. The advantage of LANES$^+$ is its ability to realize hybrid proofs more efficiently by exploiting RPoK for the high-dimensional part of the secret witness while leaving a low-dimensional secret witness part for the exact proof that is proven at a significantly lower cost via LANES. Thanks to the flexibility of LANES$^+$, other exact proof systems can also be supported.

We apply our LANES$^+$ framework to construct substantially shorter proofs of rounding, which is a central tool for *verifiable* deterministic lattice-based cryptography. Based on our rounding proof, we then design an efficient long-term verifiable random function (VRF), named LaV. LaV leads to the shortest VRF outputs among the proposals of standard (i.e., long-term and stateless) VRFs based on quantum-safe assumptions.

Of independent interest, we also present generalized results for challenge difference invertibility, a fundamental soundness security requirement for many proof systems.

Keywords: Lattice · Zero-Knowledge Proofs · Post-Quantum · Learning with Rounding · Verifiable Random Function

1 Introduction

Zero-knowledge proofs are fundamental tools for construction of privacy-preserving cryptographic protocols. Constructing such protocols with security against quantum attacks is an active research area, with lattice-based techniques a leading candidate. In such lattice-based privacy-preserving protocols, the desired protocol functionality boils down to constructing a zero-knowledge protocol for proving a relation of the form

$$\mathbf{Ar} + \mathbf{Bm} = \mathbf{t}, \tag{1}$$

over the underlying ring $\mathcal{R}_{q,d}$ (which may be $\mathbb{Z}_q$ or a d-dimensional polynomial ring modulo an integer q). In the above expression, $\mathbf{A}, \mathbf{B}$ are public matrices, $\mathbf{t}$ is a public vector and $(\mathbf{r}, \mathbf{m})$ is a pair of secret vectors constituting the prover's witness in the zero-knowledge proof, having *small* coordinates in some sets S_1, S_2 (e.g., $S_i = \{-1, 0, 1\}$). The witness vectors may also be required to satisfy additional constraints (e.g., linear relations). When the zero-knowledge protocol proves knowledge of such a witness satisfying (1) exactly and with coordinates guaranteed to be in the set S_i, it is said to be an *exact* proof. There is a line of work on constructing such exact lattice-based proofs, from long Stern-type [Ste93] proofs [LLNW17], to more compact algebraic proofs [BLS19, YAZ+19], culminating in the state-of-the-art, which we call the LANES framework, consisting of the combination of techniques developed in [ALS20, ENS20, LNS20] (the LANES acronym we use is derived from the initials of the authors of those latter works). However, even the state-of-the-art LANES framework for exact lattice-based proofs often results in relatively long proofs in practice. In contrast, some cryptographic functionalities, such as plain signatures [Lyu09, Lyu12, DLL+18], ring signatures and applications [ESLL19, EZS+19, ESZ22, LNS21b] and group signatures [dPLS18, EZS+19, ESZ22], have been shown to be realizable more compactly without resorting to exact proofs, replacing them with significantly shorter *relaxed (approximate) proofs of knowledge* RPoK, i.e., proofs of relations of the form

$$\mathbf{A}\mathbf{r}' + \mathbf{B}\mathbf{m}' = \bar{c}\mathbf{t}, \tag{2}$$

for a short "relaxation factor" $\bar{c} \in \mathcal{R}_{q,d}$, and also allowing some slack in the set S_i in which the coordinates of the witness vector $(\mathbf{r}', \mathbf{m}')$ are proved to be in.

In this paper, we focus on important cryptographic functionalities for which *exact* proofs are required for proving the well-formedness of *part* of the witness. In such *hybrid exact/relaxed proof* applications, it is crucial that the proof is exact for the portion $\mathbf{m}$ of the witness $(\mathbf{r}, \mathbf{m})$, in the sense that the coordinates of $\mathbf{m}$ are proved to exactly belong in some set S_i (and satisfy the appropriate additional, e.g., linear constraints), but the coordinates of $\mathbf{r}$ may have some soundness slack, and the relation to be satisfied is of the form

$$\mathbf{A}\mathbf{r}' + \bar{c}\mathbf{B}\mathbf{m} = \bar{c}\mathbf{t}. \tag{3}$$

Note that if $\bar{c}$ is invertible in $\mathcal{R}_{q,d}$, then (3) can be re-written as $\mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{m} = \mathbf{t}$ for $\mathbf{r} := \mathbf{r}'/\bar{c}$ so that it is exact for the $\mathbf{B}\mathbf{m}$ term while the relaxation factor $\bar{c}$ only affects the $\mathbf{r} = \mathbf{r}'/\bar{c}$ witness part (we remark that when the real witness $\mathbf{r}$ has unconstrained coordinates, this actually becomes an exact proof with extracted witness $\mathbf{r} = \mathbf{r}'/\bar{c}$; the relaxation factor only comes in when we require $\mathbf{r}$ to be short). Unfortunately, a limitation of the LANES framework for exact proofs is that it is not flexible enough to support such hybrid exact/relaxed relations *efficiently*. Namely, when using LANES for such hybrid relations, one is forced to prove an exact relation for the *whole* witness $(\mathbf{r}, \mathbf{m})$, which leads to long proofs, as the length of the LANES proof is proportional to the total length of the witness (we discuss this more precisely in 'Technical Overview' section). On

the other hand, compact relaxed proofs alone cannot be used due to the exact proof requirement on the $\mathbf{m}$ part of the witness.

A case in point of hybrid exact/relaxed relation that forms the central motivation of this paper is that of *rounding proofs*. Given a public matrix $\mathbf{A}$ and a vector $\mathbf{t}$ over $\mathbb{Z}_q$, and a rounding modulus p, a rounding proof proves knowledge of a secret vector $\mathbf{s}$ such that $\mathbf{t} = \lfloor \mathbf{As} \rfloor_p := \lfloor \frac{p}{q} \cdot \mathbf{As} \rfloor$, where the rounding is done coefficient-wise. Rounding proofs come up in protocols that prove the well-formedness of lattice-based Pseudo-Random Functions (PRFs) based on the Learning with Rounding (LWR) problem introduced in [BPR12] and several LWR-based constructions of PRFs are known [BPR12, BLMR13, BP14]. Proofs of correct PRF evaluation have applications in Verifiable Random Functions (VRFs) as constructed in this paper, along with privacy-preserving decentralized e-cash systems [BCG$^+$14, GM17, CGL$^+$17], stateful anonymous credentials [CGH09], n-times periodic anonymous authentication [CHK$^+$06], traceable ring signatures [FS07], anonymous survey systems [HMPS14], password-protected secret sharing [JKK14] and unlinkable pseudonyms for distributed databases [CL15] as stated in [LLNW17]. For $p \mid q$, the rounding relation $\mathbf{t} = \lfloor \mathbf{As} \rfloor_p$ can be written in the form $\frac{q}{p}\mathbf{t} = \mathbf{As} - \mathbf{e}$, where $\mathbf{e} \in [0, q/p - 1]^m$ is the rounding error. This has the form of (1), where the witness consists of $(\mathbf{s}, \mathbf{e})$. In rounding proofs, it is crucial that the proof is exact for the $\mathbf{e}$ part to ensure that its coordinates are in $[0, q/p - 1]$ for the correct rounding relation, whereas it turns out to be fine for applications to relax the proof requirement for the $\mathbf{s}$ portion of the witness. For example, a set of standard LWR samples does not require the secret $\mathbf{s}$ to be short. In typical applications, the dimension of the relaxed portion $\mathbf{s}$ of the witness is dictated by security constraints of the LWR problem, and is much longer than the dimension of the exact portion of the witness $\mathbf{e}$. Therefore, rounding proofs are a typical example of hybrid exact/relaxed proofs where the inflexibility limitation of the plain LANES framework would lead to long proofs, despite the short dimension of the exact portion of the witness.

The main application of rounding proofs we focus on in this paper is to the construction of lattice-based long-term (stateless) *Verifiable Random Functions* (VRFs). A VRF is a type of pseudorandom function whose output is both authenticated and publicly verifiable [MRV99]. VRFs based on quantum-insecure assumptions have been used in practice, for example, in the DNSSEC protocol [GNP$^+$15], WhatsApp's key transparency protocol [LL], and in blockchain Proof-of-Stake consensus protocols [GHM$^+$17,CM19,KRDO17]. Existing quantum-safe VRF constructions, on the other hand, fall into two classes. The first class are constructions in the standard model [GHKW17, Bit20, Yam17], which are relatively inefficient in practice but avoid the use of a common reference string or random oracle. The second class are constructions in the random oracle model [YAZ$^+$19, EKS$^+$21, BDE$^+$21]. The latter constructions are more practically oriented, but are limited due to the lack of compact rounding proofs or other reasons as discuss below. The lattice-based VRF construction sketched in [YAZ$^+$19] uses inefficient exact proofs of rounding that have lengths in the order of MBs. Even if improved using the LANES framework, such exact rounding

	Comm. Size	Key Hom.	Long Term	Stateless	Low Storage & Fast Keygen	Security
X-VRF [BDE+21]	3 KB	✗	✓	✗	✗	Hash
LB-VRF [EKS+21]	8.34 KB	✓	✗	✓	✓	Lattice
SL-VRF [BDE+21]	40 KB	✗	✓	✓	✓	Hash
LaV **(this work)**	10.3 KB	✓	✓	✓	✓	Lattice

Table 1. Comparison of (plausibly) post-quantum practical VRFs. ✓ means the property is partially satisfied. 'Key Hom.' means the underlying PRF is (approximately) 'key homomorphic'. For the communication size (Comm. Size) of LB-VRF, we consider the sum of proof size, VRF value and public key since the construction is one-time.

proofs would typically still be quite long, in the order of 100 KB[4]. The lattice-based VRF construction in [EKS+21] is compact (with proof sizes around 5-8 KB) but, to avoid the need for rounding proofs, it leaks an exact linear relation on the secret key with each VRF evaluation, which limits the number of times it can be evaluated to a small value (typically 1-5 evaluations), i.e., the construction in [EKS+21] is a *few-time* VRF rather than a full-fledged (practically unlimited-time) VRF as we construct in this paper.

In the application of VRF to Algorand's blockchain protocol, the few-time limitation on the VRF of [EKS+21] introduces modifications and additional overheads to the Algorand consensus protocol, in order to periodically refresh the VRF keys of the users [EKS+21]. Other applications of VRFs, such as the DNSSEC protocol [GNP+15], inherently require a long-term VRF. The authors of [EKS+21] stated that the main bottleneck to constructing an efficient long-term lattice-based VRF is the lack of an efficient rounding proof. We address this open problem in this paper.

Two VRF constructions based on symmetric-key primitives are given in [BDE+21], but also suffer from significant practical limitations. The first construction in [BDE+21], called X-VRF, achieves compact proofs (around 3 KB) but suffers from a *stateful* VRF algorithm and a key generation time and prover storage cost that increases in proportion to the number of allowed VRF evaluations (e.g., leading to days long key generation times for 2^{27} VRF evaluations). The second construction in [BDE+21], called SL-VRF, avoids stateful evaluation and long setup and memory costs, but suffers from long proofs in the order of 40KB (see Table 1).

Another significant consideration for higher-level applications of VRFs (or correct PRF evaluation proofs) is (an approximate) *key-homomorphism* of the underlying PRF (i.e., $\mathsf{PRF}_{\mathsf{sk}_0}(\mathsf{m}) + \mathsf{PRF}_{\mathsf{sk}_1}(\mathsf{m}) \approx \mathsf{PRF}_{\mathsf{sk}_0+\mathsf{sk}_1}(\mathsf{m})$), as this is an important property for various applications such as anonymous e-cash, distributed PRFs, symmetric-key proxy re-encryption and updatable encryption. The symmetric-key based proposals in [BDE+21] do not offer key-homomorphism.

[4] Even the optimized proof of 1024-dimensional LWE samples with *ternary* secret and error (i.e., $\mathbf{s}, \mathbf{e} \in \{-1, 0, 1\}^{1024}$) in [LNS21a] is at 33 KB. The magnitude of rounding error coefficients needs to be bigger for a VRF to circumvent algebraic attacks.

1.1 Our Contributions

LANES$^+$ framework: compact hybrid exact/relaxed proofs. We introduce a novel general framework called LANES$^+$ for constructing compact proofs for hybrid exact/relaxed relations, addressing the limitations of the LANES framework. LANES$^+$ combines the best of LANES and Relaxed Proofs of Knowledge (RPoK) to achieve much shorter proofs than LANES when the exact part of the witness is short compared to the full length of the witness. The LANES$^+$ framework proves relations of the form (3) and supports additional exact linear relations and polynomial constraints on the exact part $\mathbf{m}$ of the witness. Our LANES$^+$ framework is flexible enough to support different exact proof systems, including a concurrent work [LNP22] as discussed further in Sec. 1.2.

Compact lattice-based rounding proofs. We present an efficient instantiation of our LANES$^+$ framework applied to the design of compact rounding proofs for cryptographic protocols based on the LWR problem. Our rounding proof is substantially shorter than prior proposals [YAZ$^+$19, LLNW17] as they require communication in the order of MBs. We believe our compact rounding proof techniques will find future applications for the design of efficient correct PRF evaluation proofs in lattice-based privacy-preserving protocols such as anonymous e-cash [LLNW17]. We leave the application of our techniques to anonymous e-cash as future work.

LaV: Compact (long-term) lattice-based VRF. To demonstrate the utility of our new techniques, we present an efficient application of our LANES$^+$-based rounding proofs to the construction of a compact (long-term) lattice VRF, called LaV. Our construction is the *first practical* lattice-based VRF supporting practically unrestricted number (2^{128}) of VRF evaluations. For typical parameters, LaV achieves a VRF output size of about 10.3 KB, which is about 1.24× overhead over the communication size needed in [EKS$^+$21], while allowing for an arbitrary number of VRF evaluations (versus the 1-5 evaluation limitation of [EKS$^+$21]). In Table 1, we provide a comparison between practical post-quantum VRF proposals.

To support our new VRF construction and rounding proofs, we also introduce another technical contribution of potential independent interest as below.

Generalization of challenge difference invertibility bounds. RPoK part of our LANES$^+$ protocol requires the invertibility of challenge differences in the underlying polynomial ring and it is important for the practical efficiency of LaV that $\dim(\mathcal{R}_{q,d}) = d$ is small (e.g., $d = 32$). The latter requirement forces the protocol challenge c to have relatively large coefficients. To support this, we generalize the challenge difference invertibility bounds from [ALS20, ESZ22], which apply only to *ternary* challenge coordinates. In particular, we derive bounds for challenges with coefficients of infinity norm γ for any $\gamma \geq 1$. These generalized results are used to optimize the length of our rounding proofs in LaV, and we believe they will find further applications in future lattice proof systems. In general, compared to prior results applicable for $\gamma > 1$ such as [LS18], our new results allow to use a smaller modulus q and/or a highly-splitting ring $\mathcal{R}_{q,d}$.

1.2 Technical Overview

LANES$^+$ framework. We first explain in more detail the inflexibility limitations of the LANES framework. We recall that LANES uses a commitment scheme defined over a cyclotomic polynomial ring $\mathcal{R}_{q,\hat{d}} := \mathbb{Z}_q[X]/(X^{\hat{d}}+1)$ where $\hat{d}$ is a power of 2 and q is chosen so that $\mathcal{R}_{q,\hat{d}}$ splits into l subrings via the Chinese Remainder Theorem (CRT). We also use $\mathcal{R}_{q,d}$ to denote the ring where operations external to LANES are performed. In the following, for a vector $\mathbf{x} \in \mathcal{R}_{q,d}^n$, $\overrightarrow{\mathbf{x}} \in \mathbb{Z}_q^{dn}$ denotes the (concatenated) coefficient vector of $\mathbf{x}$ over $\mathbb{Z}_q$. In general, we will write $\overrightarrow{x}$ to denote vectors over $\mathbb{Z}_q$ and $\mathbf{x}$ to denote vectors over $\mathcal{R}_{q,d}$. Due to the way relations are proved in LANES, one cannot reduce the proof size by exploiting the *partial* exactness of the relation so that a relaxed proof of knowledge can be leveraged for the relaxed relation part. We elaborate more on this further below once we set out our target problem next.

Recall that the most common relations in lattice-based cryptography are of the form (1). We call $\mathbf{m}$ as "message" and $\mathbf{r}$ as "randomness" for ease of reference. As far as our framework is concerned, the distinction is merely that $\mathbf{m}$ is the secret vector part that goes into LANES, while $\mathbf{r}$ is the remaining part.

It is a common requirement to prove not just that (1) holds, but also that the message and/or the randomness satisfy certain properties (such as having small coefficients). Now suppose that we want to prove such a common relation along with some arbitrary linear relation $\mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v}$ for $(\mathbf{G}_1, \mathbf{G}_2, \overrightarrow{v})$ defined over $\mathbb{Z}_q$. First note that revealing $\overrightarrow{v}$ or $\mathbf{G}_1\overrightarrow{\mathbf{m}}$ in many cases would leak secret information (for example, when $\overrightarrow{v}$ is the binary decomposition of $\overrightarrow{\mathbf{m}}$). Hence, they need to be part of the prover's witness. Now, the way to prove these relations in LANES would be to write all of the relations in the following form

$$\underbrace{\begin{pmatrix} \mathsf{Rot}(\mathbf{A}) & \mathsf{Rot}(\mathbf{B}) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_1 & -\mathbf{G}_2 \end{pmatrix}}_{=:\mathbf{L}} \cdot \underbrace{\begin{pmatrix} \overrightarrow{\mathbf{r}} \\ \overrightarrow{\mathbf{m}} \\ \overrightarrow{v} \end{pmatrix}}_{=:\overrightarrow{x}} = \underbrace{\begin{pmatrix} \overrightarrow{\mathbf{t}} \\ \overrightarrow{0} \end{pmatrix}}_{=:\overrightarrow{t}}, \tag{4}$$

where $\mathsf{Rot}(\cdot)$ denotes the representative matrix of its input over $\mathbb{Z}_q$, and just prove this linear relation (along with additional multiplicative relations). However, the drawback of this approach is that the secret witness dimension here is $\dim(\overrightarrow{x}) = \dim(\overrightarrow{\mathbf{r}}) + \dim(\overrightarrow{\mathbf{m}}) + \dim(\overrightarrow{v})$. In many cases, the dimension of the randomness $\overrightarrow{\mathbf{r}}$ is lower-bounded by the security requirements (such as hiding and pseudorandomness) and thus cannot be very small. Indeed, there are applications where the dimension of the message $\overrightarrow{\mathbf{m}}$ is much smaller than that of the randomness, i.e., $\dim(\overrightarrow{\mathbf{m}}) \ll \dim(\overrightarrow{\mathbf{r}})$. Consider, for example, the case when we want to prove knowledge of a *single* LWR sample. Here, $\mathbf{r}$ being the secret key would typically have $\dim(\overrightarrow{\mathbf{r}}) \geq 1024$ while $\mathbf{m}$ being the rounding error would just have $\dim(\overrightarrow{\mathbf{m}}) = 1$. Since the size of a LANES proof output scales linearly in the dimension of the witness (see (12) in Sec. 2.4), it may not be ideal in such applications to use the LANES framework directly.

To get around the above efficiency challenge, we introduce a hybrid framework that allows to combine a RPoK with LANES. Particularly, our goal is to

Algorithm 1 Standard Lattice-based Relaxed Proof of Knowledge (RPoK)

procedure $\mathsf{RPoK}((\mathbf{A},\mathbf{B},\mathbf{t});(\mathbf{r},\mathbf{m}))$:
  Sample short rand. masking $\mathbf{y}$
  Sample message masking $\mathbf{u}$
  $\mathbf{w}=\mathbf{A}\mathbf{y}+\mathbf{B}\mathbf{u}$ over $\mathcal{R}_{q,d}$
  $c\leftarrow\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{t},\mathbf{w})$ for a hash $\mathcal{H}$
  $\mathbf{z}=\mathbf{y}+c\cdot\mathbf{r}$
  $\mathbf{f}=\mathbf{u}+c\cdot\mathbf{m}$
  Rejection samp. on $\mathbf{z}$ (and $\mathbf{f}$ if req.)
  return proof $\pi=(c,\mathbf{z},\mathbf{f})$
end procedure

procedure $\mathsf{Verify}((\mathbf{A},\mathbf{B},\mathbf{t}),\pi)$:
  Parse $\pi=(c,\mathbf{z},\mathbf{f})$
  If $\mathbf{z}$ (and $\mathbf{f}$) is not sufficiently short, **return** 0
  $\mathbf{w}'=\mathbf{A}\mathbf{z}+\mathbf{B}\mathbf{f}-c\mathbf{t}$ over $\mathcal{R}_{q,d}$
  If $c\neq\mathcal{H}(\mathbf{A},\mathbf{B},\mathbf{t},\mathbf{w}')$, **return** 0
  return 1
end procedure

prove the relation in (1) using very efficient RPoK (as those used in ordinary signatures) shown in Alg. 1 and exploit LANES to prove the remaining linear (and multiplicative) relation (i.e., $\mathbf{G}_1\overrightarrow{\mathbf{m}}=\mathbf{G}_2\overrightarrow{v}$). This way, we will be combining the best of two worlds by (i) proving the (often low-dimensional) $\mathbf{G}_1\overrightarrow{\mathbf{m}}=\mathbf{G}_2\overrightarrow{v}$ linear relation *exactly* (via LANES), and (ii) using the efficient relaxed proofs whenever possible for the high-dimensional relations as in (1). A technical challenge here is that LANES protocol does not involve a masked opening of its input message (i.e., $\mathbf{m}$), preventing the utilization of standard EQ or AND protocol compositions that use the same masked opening in multiple proof parts.

Using a standard rewinding argument, we can show that RPoK in Alg. 1 proves knowledge of $(\bar{c},\bar{\mathbf{z}},\bar{\mathbf{f}})$ with short $(\bar{c},\bar{\mathbf{z}})$ (and possibly short $\bar{\mathbf{f}}$) such that

$$\mathbf{A}\bar{\mathbf{z}}+\mathbf{B}\bar{\mathbf{f}}=\bar{c}\mathbf{t}, \tag{5}$$

where $\bar{c},\bar{\mathbf{z}},\bar{\mathbf{f}}$ are the differences of rewinded protocol outputs $(c,\mathbf{z},\mathbf{f})$ and $(c',\mathbf{z}',\mathbf{f}')$. From Alg. 1, we can see that the masked message opening in RPoK is $\mathbf{f}=\mathbf{u}+c\cdot\mathbf{m}$. We exploit this to make a connection between the two proof parts (RPoK and LANES). Particularly, we prove via LANES that $\overrightarrow{\mathbf{f}}=\overrightarrow{\mathbf{u}}+\mathsf{Rot}(c)\cdot\overrightarrow{\mathbf{m}}$ over $\mathbb{Z}_q$, ensuring that $\mathbf{f}$ is indeed of the desired form, along with the low-dimensional linear relation $\mathbf{G}_1\overrightarrow{\mathbf{m}}=\mathbf{G}_2\overrightarrow{v}$ and any other polynomial constraints on the coordinates of $\overrightarrow{\mathbf{m}}$. From the LANES witness extractor, a similar relation holds for the rewinded transcript such that $\overrightarrow{\mathbf{f}'}=\overrightarrow{\mathbf{u}}+\mathsf{Rot}(c')\cdot\overrightarrow{\mathbf{m}}$ with the same $(\overrightarrow{\mathbf{u}},\overrightarrow{\mathbf{m}})$ by the binding of the LANES commitment. This gives that $\overrightarrow{\mathbf{f}}-\overrightarrow{\mathbf{f}'}=\mathsf{Rot}(\bar{c})\cdot\overrightarrow{\mathbf{m}}$, and thus $\bar{\mathbf{f}}=\bar{c}\mathbf{m}$ over $\mathcal{R}_{q,d}$. Plugging this in (5) gives the desired hybrid relation in (3) with $\mathbf{r}'=\bar{\mathbf{z}}$. With this approach, $\mathbf{r}$ is never involved in the LANES part and we can guarantee the use of the same witness $\overrightarrow{\mathbf{m}}$ in both LANES and RPoK.

Overall, the goal of LANES^+ is to prove knowledge of a tuple $(\bar{c},\mathbf{m},\mathbf{r},\overrightarrow{v})\in\mathcal{L}^+(\mathsf{mp},\mathsf{ulp})$ (i.e., $(\mathsf{ck},(\mathsf{mp},\mathsf{ulp}),(\bar{c},\mathbf{m},\mathbf{r},\overrightarrow{v}))\in R_{\mathsf{LANES}^+}$) such that

$$\mathcal{L}^+(\mathsf{mp},\mathsf{ulp})=\left\{(\bar{c},\mathbf{m},\mathbf{r},\overrightarrow{v}):\begin{array}{r}\mathbf{t}=\mathbf{A}\mathbf{r}+\mathbf{B}\mathbf{m}\text{ over }\mathcal{R}_{q,d}\ \wedge\ \mathbf{G}_1\overrightarrow{\mathbf{m}}=\mathbf{G}_2\overrightarrow{v}\bmod q\\ \wedge\ P(\overrightarrow{\mathbf{m}},\overrightarrow{v})=0\bmod q\ \forall P\in\mathsf{mp}\wedge\\ \|\bar{c}\mathbf{r}\|_\infty\leq\gamma_r\ \wedge\ \|\bar{c}\|_\infty\leq\gamma_c\text{ for }\gamma_r,\gamma_c\ll q\in\mathbb{Z}^+\end{array}\right\},$$

where mp is a set of multivariate polynomials in the coordinates of $(\overrightarrow{\mathbf{m}}, \overrightarrow{v})$[5] over $\mathbb{Z}_q$ (for example, enforcing the smallness of the witness coefficients via $P_i(\overrightarrow{\mathbf{m}}, \overrightarrow{v}) = v_i(v_i - 1)$ for $\overrightarrow{v} = (v_0, v_1, \ldots)$), $\mathsf{ulp} = ((\mathbf{A}, \mathbf{B}, \mathbf{t}), (\mathbf{G}_1, \mathbf{G}_2))$ is the collection of linear relations and γ_r, γ_c are some public norm-bounds. Note that the above language does not necessarily require $\mathbf{r}$ to be short, but $\bar{c}\mathbf{r}$ is short. Furthermore, the relation in (1) and the operations in LANES are not necessarily defined over the same polynomial ring. Particularly, LANES works internally over $\mathcal{R}_{q,\hat{d}}$ and proves relations over $\mathbb{Z}_q$, while (1) is over $\mathcal{R}_{q,d}$. In many cases, the relation proved by LANES is in fact over the integers (without mod q) and in those cases, we can use different moduli for the two rings $\mathcal{R}_{q,d}$ and $\mathcal{R}_{q,\hat{d}}$. This gives a lot of flexibility in choosing parameters and is critical for our rounding and VRF applications because the rounding/VRF relation requires a composite modulus while LANES works with a prime modulus.

Comparison with concurrent work. We further note that an approach from a concurrent and independent work [LNPS21] may also be adapted to solve our target problem. As we discuss next, our approach has the following advantages over a potential adaptation of [LNPS21] to our setting: (i) efficient and simple support for different system moduli, and (ii) better efficiency for applications with an expanding matrix $\mathbf{B}$. To use the techniques in [LNPS21] in our setting, one can use the term $\mathbf{w} := \mathbf{A}\mathbf{y}$ as a witness for LANES[6] and prove the relation $\mathbf{A}\mathbf{z} + c \cdot \mathbf{B}\mathbf{m} = \mathbf{w} + c \cdot \mathbf{t}$ via LANES. In this case, it is difficult to use different moduli for the LANES proof and the main relation $\mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{m} = \mathbf{t}$ because the main relation is being proven by LANES. It may be possible to overcome this by proving a relation of the form $\mathbf{A}\mathbf{z} + c \cdot \mathbf{B}\mathbf{m} = \mathbf{w} + c \cdot \mathbf{t} + v \cdot q$ over $\mathbb{Z}$ (without mod q) with LANES. This requires the LANES modulus to be significantly bigger, for example exceeding $\|v \cdot q\|_\infty$, which leads to a longer LANES output, larger communication and a more complicated protocol overall. Furthermore, since the LANES witness in a possible approach based on [LNPS21] is $(\mathbf{m}, \mathbf{w})$, the proof length will grow linearly with $\dim(\mathbf{w}) = \dim(\mathbf{t})$. Although our framework focuses on the case where $\mathbf{m}$ is small dimensional, we do not necessarily require/assume $\dim(\mathbf{t})$ to be small. For example, when $\mathbf{B}$ is an expanding matrix, we would have $\dim(\mathbf{t}) > \dim(\mathbf{m})$, implying a lower communication cost in our approach. Such expanding matrices are used in different contexts, for example, where a 'gadget' vector/matrix $\mathbf{G}$ multiplies a *scalar* message m, and hence, $\dim(\mathbf{G}m) \gg \dim(m) = 1$.

Thanks to the flexibility of our LANES^+ framework, we can support other exact proof systems. Particularly, another concurrent and independent work [LNP22] recently introduced a new exact proof system, that we call the LNP22 proof. Much like LANES, the LNP22 proof is also a commit-and-prove protocol. Since our LANES^+ framework makes black-box use of LANES, we can easily adapt LANES^+ to work with the LNP22 proof, where the commit-and-prove function-

[5] The polynomials need to obey certain restrictions depending on the structure of the underlying ring $\mathcal{R}_{q,d}$, which is explained formally in Sec. 2.4.

[6] Note that in this case, we need to hide $\mathbf{w} := \mathbf{A}\mathbf{y}$. Otherwise, everyone could compute $\mathbf{t} - \mathbf{w} = \mathbf{B}\mathbf{m}$, which leaks information on the secret $\mathbf{m}$.

alities of LANES would be replaced with those of the LNP22 proof. However, for our focus of *small* dimensional message vectors, we found that LANES still produces shorter proofs than the LNP22 proof. Particularly, in consultation with Nguyen [Ngu22] (an author of [LNP22]), we looked at the LNP22 proof size of the exact proof component needed for our VRF application and found it to be 11.2 KB. This is larger than our 7.1 KB proof size using LANES (see Sec. 6.5). More generally, Nguyen [Ngu22] confirmed that the LNP22 proof size lowerbound is at least 10 KB for any useful application. Therefore, for our focus applications with small-dimensional messages in this work, LANES is a better exact proof option than the LNP22 proof. However, for medium-sized message vectors, equipping LANES$^+$ with the LNP22 proof as discussed above may result in smaller proof sizes, extending the advantage of LANES$^+$ to a wider application domain.

Rounding proof technique. As explained above, our proof of rounding applies our LANES$^+$ framework to the rounding relation $\mathbf{t} = \lfloor \mathbf{Cs} \rfloor_p$ written in the form $\frac{q}{p}\mathbf{t} = \mathbf{Cs} - \mathbf{e}$, where $\mathbf{e} \in [0, q/p-1]^m$ is the rounding error vector and $\mathbf{C}$ is a matrix. Here, we invoke our LANES$^+$ proof with the witness $(\mathbf{m}, \mathbf{r}, \overrightarrow{v}) = (\mathbf{e}, \mathbf{s}, \overrightarrow{b})$, where $\mathbf{e}$ is the (typically short) part of the witness for which the exact proof is needed, $\mathbf{s}$ is the longer part of the witness for which a relaxed proof is sufficient, and $\overrightarrow{b}$ is a β-ary digit decomposition of $\mathbf{e}$ for some small β chosen to optimise the proof length. The main LANES$^+$ matrices are set as $(\mathbf{A}, \mathbf{B}) = (\mathbf{C}, -\mathbf{I})$ to enforce the rounding relation between $\mathbf{s}$ and $\mathbf{e}$, while the LANES$^+$ exact linear relation matrices are set as $(\mathbf{G}_1, \mathbf{G}_2) = (\mathbf{I}, \mathbf{G}_\beta)$, where $\mathbf{G}_\beta$ denotes the β-ary digit reconstruction gadget matrix (having powers of β along its rows) to enforce the β-ary reconstruction relation $\overrightarrow{\mathbf{e}} = \mathbf{G}_\beta \overrightarrow{b}$. We set the LANES$^+$ exact polynomial constraint $P(b_i) = \prod_{j \in [\beta]}(b_i - j) = 0$ to enforce the range $[0, \beta - 1]$ for the β-ary digits of $\mathbf{e}$ encoded as the coordinates b_i of $\overrightarrow{b}$. Consequently, the proof length of our call to LANES inside LANES$^+$ depends only on the length of the (short) witness part $\mathbf{e}$ and β, and not on the long witness part $\mathbf{s}$.

Generic folklore VRF construction and LaV. A natural way to construct a VRF is to combine a PRF function with a NIZK proof of correct PRF evaluation. In more detail, the VRF public key is a PRF output $\mathsf{pk} = \mathsf{PRF}_{\mathsf{sk}}(0)$ under a VRF/PRF secret key sk. To evaluate the VRF on a message m using secret key sk, we compute $v = \mathsf{PRF}_{\mathsf{sk}}(\mathsf{m})$ as the VRF value. Then, a NIZK proof, π, is generated to prove the well-formedness of values pk and v under the same secret key sk. Here, the pseudorandomness property of PRF is used to provide the VRF pseudorandomness. For the uniqueness of the VRF, we require the PRF to satisfy additional key-homomorphism (as defined in the introduction) and key-binding properties, where the latter ensures that if $\mathsf{PRF}_{\mathsf{sk}_0}(\mathsf{m}) = \mathsf{PRF}_{\mathsf{sk}_1}(\mathsf{m})$, then $\mathsf{sk}_0 = \mathsf{sk}_1$. The soundness of NIZK Π also contributes to uniqueness by ensuring that v is the only output that can pass the NIZK verification test.

We remark that this folklore VRF approach was informally sketched, e.g., in [Bit20, Sec. 1.2]. As discussed in Sec. 2.3, ECVRF [PWH$^+$17] and LB-VRF [EKS$^+$21] are examples of this paradigm. Our instantiation LaV in this work uses $\mathsf{PRF}_{\mathsf{sk}}(\mathsf{m}) := \lfloor \mathbf{A} \cdot \mathsf{sk} \rfloor_p = \mathbf{v}$ with $\mathbf{A} \leftarrow \mathcal{G}(\mathsf{m})$ for a random oracle $\mathcal{G}$

(where the PRF enjoys an approximate key homomorphism property), based on the Module LWR (MLWR) assumption.

In the context of LaV, the exact guarantee for the rounding error $\mathbf{e}$ in our LANES$^+$-based roundness proof NIZK is essential to guarantee the uniqueness of the VRF (as otherwise the adversary could pass the NIZK verification test with multiple errors $\mathbf{e}$ and break VRF uniqueness). LaV optimizes this generic construction by shrinking the vector $\mathbf{v}$ from a full PRF output to a portion of it (one ring element), and relaxing the NIZK requirement so it does not need to prove exact well-formedness of pk; a relaxed proof is sufficient. This is crucial to the efficiency of LaV as it allows us to use our LANES$^+$ framework as the NIZK Π, without including the long secret key sk in the underlying LANES exact proof.

2 Preliminaries

We use $[n] = \{0, \ldots, n-1\}$ for $n \in \mathbb{Z}^+$ and $\mathbb{Z}_q = [-(q-1)/2, (q-1)/2]$ for an odd modulus q. We utilize polynomial rings of the form $\mathcal{R}_{q,d} = \mathbb{Z}_q[X]/(X^d+1)$ for power-of-2 d and modulus $q \geq 2$. For a positive integer $c \leq q/2$, $\mathbb{S}_{c,d}$ denotes the set of polynomials in $\mathcal{R}_{q,d}$ with infinity norm at most c (w.r.t. the monomial (coefficient) basis). For a vector $\mathbf{x} \in \mathcal{R}_{q,d}^n$, $\overrightarrow{\mathbf{x}} \in \mathbb{Z}_q^{dn}$ denotes the (concatenated) coefficient vector of $\mathbf{x}$. In general, we will write $\overrightarrow{x}$ to denote vectors over $\mathbb{Z}_q$ and $\mathbf{x}$ to denote vectors over $\mathcal{R}_{q,d}$. We write $\lfloor \overrightarrow{x} \rceil_p$ to denote $\lfloor \frac{p}{q} \cdot \overrightarrow{x} \rceil$ for $\overrightarrow{x} \in \mathbb{Z}_q^m$, where the rounding is done coordinate-wise. The same notation extends analogously to vectors over $\mathcal{R}_{q,d}$ by applying the rounding to the coefficient vector. In this paper, we use the rounding down operation, but our results easily extend to the rounding up or to the closest integer operations. For an element of and a matrix over $\mathcal{R}_{q,d}$, we write $\mathsf{Rot}(f)$ and $\mathsf{Rot}(\mathbf{A})$, respectively, to denote its representative matrix over $\mathbb{Z}_q$. For vectors $\overrightarrow{x}$ and $\overrightarrow{y}$ over $\mathbb{Z}_q$, $\overrightarrow{x} \circ \overrightarrow{y}$ denotes coordinate-wise multiplication. We use $\bigcirc$ to denote coordinate-wise multiplication over a set of elements. $\mathsf{HW}(f)$ denotes the Hamming weight of the coefficient vector of $f \in \mathcal{R}_{q,d}$, and $\mathbb{D}_{\sigma,d}$ denotes the d-dimensional discrete Gaussian distribution with standard deviation σ and center 0. Some preliminaries including formal VRF definitions, MSIS/MLWR definitions, and rejection sampling are deferred to App. A.

The following fact plays an important role in our rounding proof and VRF.

Fact 1 (adapted from [LLNW17]). *Let $\overrightarrow{u} \in \mathbb{Z}_q^n$ and $\overrightarrow{v} \in \mathbb{Z}_p^n$ for $q > p$, where p divides q. Then, $\overrightarrow{v} = \lfloor \overrightarrow{u} \rceil_p$ if and only if there exists $\overrightarrow{e} \in \mathbb{Z}^n$ such that $\overrightarrow{e} \in [q/p]^n$ and $\overrightarrow{e} = \overrightarrow{u} - \frac{q}{p} \cdot \overrightarrow{v} \pmod q$.*

2.1 NIZK and Commit-and-Prove Protocols

We define a commit-and-prove (CP) protocol [Kil90, CLOS02] similar to the descriptions provided in [EG14]. Particularly, let ck, x and w denote a commitment key, a statement and a witness, respectively. Further, let $R_{\mathcal{L}}$ be a polynomial-time verifiable relation containing tuples (ck, x, w). We define a language $\mathcal{L}_{\mathsf{ck}}$ as the set of statements for which there exists a witness w with $(\mathsf{ck}, x, w) \in R_{\mathcal{L}}$. In general, a CP protocol allows one to commit to a sequence of messages $m = (m_1, \ldots, m_N)$ for $N \geq 1$ and prove certain statements about the committed

messages. For a commitment output, we will have a pair $(t; t')$ of public-secret outputs, where the latter needs to be retained by the prover for further steps.

Formally, a commit-and-prove protocol consists of four polynomial time algorithms $\Pi = (\Pi.\mathsf{Gen}, \Pi.\mathsf{Com}, \Pi.\mathsf{Prove}, \Pi.\mathsf{Ver})$ as follows.

$\mathsf{pp} \leftarrow \Pi.\mathsf{Gen}(1^\lambda)$**:** On input a security parameter λ, generate a commitment key ck, which also specifies a message space $\mathcal{S}_M$, a randomness space $\mathcal{S}_R$ and a commitment space $\mathcal{S}_C$. Generate further system parameters pp', if needed, and output $\mathsf{pp} = (\mathsf{ck}, \mathsf{pp}')$

$(t; t') \leftarrow \Pi.\mathsf{Com}_{\mathsf{pp}}(m; r)$**:** On input public parameters pp containing a commitment key ck, a message $m \in \mathcal{S}_M$ and a randomness $r \in \mathcal{S}_R$, output a commitment $t \in \mathcal{S}_C$ along with its secret opening t'.

$\pi \leftarrow \Pi.\mathsf{Prove}_{\mathsf{pp}}(x, (t; t'))$**:** On input a statement x and commitment output pair $(t; t')$, output a proof π.

$0/1 \leftarrow \Pi.\mathsf{Ver}_{\mathsf{pp}}(x, t, \pi)$**:** On input a statement x, a commitment t and a proof π, output 1 if the proof is accepted. Otherwise, output 0.

If a set of messages are committed in sequence, then we write $(\overrightarrow{t}; \overrightarrow{t'}) \leftarrow \Pi.\mathsf{Com}_{\mathsf{pp}}(\overrightarrow{m}; \overrightarrow{r})$ to denote $(t_i, t'_i) \leftarrow \Pi.\mathsf{Com}_{\mathsf{pp}}(m_i; r_i)$ where $\overrightarrow{m} = (m_1, \ldots, m_N)$, $\overrightarrow{r} = (r_1, \ldots, r_N)$, $\overrightarrow{t} = (t_1, \ldots, t_N)$ and $\overrightarrow{t'} = (t'_1, \ldots, t'_N)$. We next provide the properties of a CP protocol, which are similar to those in [EG14, LNS21a].

Definition 1 (Correctness). *A commit-and-prove protocol* $\Pi = (\Pi.\mathsf{Gen}, \Pi.\mathsf{Com}, \Pi.\mathsf{Prove}, \Pi.\mathsf{Ver})$ *has statistical correctness if the following probability is negligible in* λ *for all adversaries* $\mathcal{A}$

$$\Pr\left[\begin{array}{ll} \mathsf{pp} \leftarrow \Pi.\mathsf{Gen}(1^\lambda); (x, \overrightarrow{m}, \overrightarrow{r}) \leftarrow \mathcal{A}(\mathsf{pp}); & \\ (\overrightarrow{t}; \overrightarrow{t'}) \leftarrow \Pi.\mathsf{Com}_{\mathsf{pp}}(\overrightarrow{m}; \overrightarrow{r}); & : \ \Pi.\mathsf{Ver}_{\mathsf{pp}}(x, \overrightarrow{t}, \pi) = 0 \\ \pi \leftarrow \Pi.\mathsf{Prove}_{\mathsf{pp}}(x, (\overrightarrow{t}; \overrightarrow{t'})) & \end{array}\right],$$

where $\mathcal{A}$ *outputs* $\overrightarrow{m} \in \mathcal{S}_M^N$ *and* $\overrightarrow{r} \in \mathcal{S}_R^N$ *for some* $N \geq 1$ *with* $(ck, x, \overrightarrow{m}) \in R_{\mathcal{L}}$.

Since our protocols rely on LANES, we define simulatability as in [LNS21a], where the randomness for the commitment is sampled properly (from χ) as it would be in the real-world protocol.

Definition 2 (Simulatability). *A commit-and-prove protocol* $\Pi = (\Pi.\mathsf{Gen}, \Pi.\mathsf{Com}, \Pi.\mathsf{Prove}, \Pi.\mathsf{Ver})$ *is simulatable if for all PPT adversaries* $\mathcal{A}$*, there exist PPT simulators* SimC *and* SimP *such that the following holds*

$$\begin{aligned} &\Pr\left[\begin{array}{ll} \mathsf{pp} = (\mathsf{ck}, \mathsf{pp}') \leftarrow \Pi.\mathsf{Gen}(1^\lambda); (x, \overrightarrow{m}) \leftarrow \mathcal{A}(\mathsf{pp}); & \\ \overrightarrow{r} \leftarrow \chi^N; (\overrightarrow{t}, \overrightarrow{t'}) \leftarrow \Pi.\mathsf{Com}_{\mathsf{pp}}(\overrightarrow{m}; \overrightarrow{r}); & : \ \begin{array}{l}(\mathsf{ck}, x, \overrightarrow{m}) \in R_{\mathcal{L}} \\ \wedge \mathcal{A}(\overrightarrow{t}, \pi) = 1\end{array} \\ \pi \leftarrow \Pi.\mathsf{Prove}_{\mathsf{pp}}(x, (\overrightarrow{t}; \overrightarrow{t'})) & \end{array}\right] \\ \approx &\Pr\left[\begin{array}{ll} \mathsf{pp} = (\mathsf{ck}, \mathsf{pp}') \leftarrow \Pi.\mathsf{Gen}(1^\lambda); (x, \overrightarrow{m}) \leftarrow \mathcal{A}(\mathsf{pp}); & \\ \overrightarrow{t} \leftarrow \mathsf{SimC}_{\mathsf{pp}}(x); & : \ \begin{array}{l}(\mathsf{ck}, x, \overrightarrow{m}) \in R_{\mathcal{L}} \\ \wedge \mathcal{A}(\overrightarrow{t}, \pi) = 1\end{array} \\ \pi \leftarrow \mathsf{SimP}_{\mathsf{pp}}(x, \overrightarrow{t}) & \end{array}\right], \end{aligned}$$

where χ *is a probability distribution on* $\mathcal{S}_R$.

Definition 3 (Knowledge Soundness). *A commit-and-prove protocol* $\Pi = (\Pi.\mathsf{Gen}, \Pi.\mathsf{Com}, \Pi.\mathsf{Prove}, \Pi.\mathsf{Ver})$ *satisfies knowledge soundness if for all PPT adversaries* $\mathcal{A}$*, there exists an expected polynomial time extractor* $\mathcal{E}$ *such that the following probability is negligible in* λ

$$\Pr\left[\begin{array}{l|l} \mathsf{pp} = (\mathsf{ck}, \mathsf{pp}') \leftarrow \Pi.\mathsf{Gen}(1^\lambda); & \Pi.\mathsf{Ver}_{\mathsf{pp}}(x, \overrightarrow{t}, \pi) = 1 \wedge \\ (x, \overrightarrow{t}, \pi) \leftarrow \mathcal{A}(\mathsf{pp}; \rho); & \left((\mathsf{ck}, x, \overrightarrow{m}) \notin R_{\mathcal{L}} \vee \Pi.\mathsf{Com}_{\mathsf{pp}}(\overrightarrow{m}^*; \overrightarrow{r}^*) \neq \overrightarrow{t}\right) \\ (\overrightarrow{m}^*; \overrightarrow{r}^*) \leftarrow \mathcal{E}(\mathsf{pp}, \rho) & \end{array}\right],$$

where $\mathcal{E}$ *outputs* $\overrightarrow{m}^* \in \mathcal{S}_M^N$ *and* $\overrightarrow{r}^* \in \mathcal{S}_R^N$ *for some* $N \geq 1$*.*

Our soundness definition is similar to the special soundness of Sigma protocols since our application protocols in this work are of the form of a Sigma protocol, but made non-interactive using the Fiat-Shamir transformation. LANES protocol has actually 5 moves with an additional 'randomization' move, but still relies on the standard rewinding arguments for soundness. When proving knowledge soundness of our proposals, we will similarly use standard rewinding arguments where the extractor rewinds the adversary to a specific point and, e.g., provides a different random oracle output.

For efficient lattice-based proofs, it is necessary to relax the soundness requirement and have $(\mathsf{ck}, x, \overrightarrow{m}) \in \bar{R}_{\mathcal{L}}$ for $R_{\mathcal{L}} \subseteq \bar{R}_{\mathcal{L}}$. We adopt the same relaxation as in many prior works, e.g., [ESLL19, ESS+19, LNS21a]. Therefore, while correctness and simulatability are defined w.r.t. to a *base* relation $R_{\mathcal{L}}$, the soundness only guarantees the extraction of a witness for an *extended* relation $\bar{R}_{\mathcal{L}}$. An honest prover's witness is in $R_{\mathcal{L}}$ (i.e., an honest run of Π uses a witness from $R_{\mathcal{L}}$).

As discussed in [EG14], a CP protocol is a generalization of a standard non-interactive zero-knowledge (NIZK) proof, where the same commitment outputs can be used across multiple NIZKs. Therefore, when considering a NIZK, we use the same syntax above while omitting $\Pi.\mathsf{Com}$, the commitment key ck in the elements of $R_{\mathcal{L}}$ and the commitment output t (and t') in $\Pi.\mathsf{Prove}$ and $\Pi.\mathsf{Ver}$.

2.2 Desired PRF Properties

A Pseudorandom Function (PRF) is a function that maps an input message m to a random-looking output v under a secret key sk, i.e., $v = \mathsf{PRF}_{\mathsf{sk}}(\mathsf{m})$. We denote the key space by $\mathcal{K}$, and output space by $\mathcal{T}$. We require a PRF to satisfy the standard pseudorandomness property where no polynomial-time adversary having adaptive oracle access to the PRF function can distinguish PRF outputs (under a random key) from independent uniformly random elements in $\mathcal{T}$ with an advantage non-negligible in the security parameter λ. We let κ be the number of oracle queries allowed in the pseudorandomness game. We sometimes write κ-pseudorandomness to explicitly denote the number of PRF oracle queries allowed in the pseudorandomness game. Some prior VRF constructions such as [EKS+21] only allow a small κ value. As our lattice-based PRF in this work satisfies the standard pseudorandomness and allows for up to $\kappa = 2^\lambda$ evaluations (where $\lambda = 128$ for our parameter settings, see Sec. 6), it results in a standard VRF construction. For the folklore VRF construction based on a PRF, we additionally

require the following PRF properties. We note that these properties are defined w.r.t. an *extended key space* $\mathcal{K}'$ to accommodate for the relaxed soundness of efficient lattice-based proofs.

Key-binding. A PRF is *statistically key-binding w.r.t. extended key space* $\mathcal{K}'$ with $\mathcal{K} \subseteq \mathcal{K}'$ if the following probability over the randomness of an adversary $\mathcal{A}$ is negligible

$$\Pr\left[(\mathsf{m}, k_0, k_1) \leftarrow \mathcal{A} : k_1 \neq k_0 \in \mathcal{K}' \wedge \mathsf{PRF}_{k_1}(\mathsf{m}) = \mathsf{PRF}_{k_0}(\mathsf{m})\right].$$

If the adversary $\mathcal{A}$ is assumed to be PPT, then the PRF is said to be *computationally key-binding*.

Additive key-homomorphism. The extended key space $\mathcal{K}'$ is a subset of a module with operations $(+, \cdot)$ over some underlying commutative scalar ring $\mathfrak{R}$, the output space $\mathcal{T}$ is a subset of a module with operations $(\oplus, \otimes)$ over $\mathfrak{R}$, and there exists a 'homomorphism' space $S \subseteq \mathfrak{R}$ of scalars such that for any keys $k_0, k_1 \in \mathcal{K}'$, message m and scalar $\alpha \in S$, we have $\mathsf{PRF}_{\alpha \cdot k_0 + k_1}(\mathsf{m}) = \alpha \otimes \mathsf{PRF}_{k_0}(\mathsf{m}) \oplus \mathsf{PRF}_{k_1}(\mathsf{m})$.

2.3 Folklore VRF from PRF and NIZK

We now present the folklore approach to constructing a VRF based on a PRF and a NIZK proof. Our treatment is a bit more general than the traditional idea to accommodate for the relaxations in efficient lattice-based NIZKs. We note that our PRF and NIZK instantiations in this work are in the random oracle model. We also remark that the PRF in this section can also be viewed as a commitment scheme by interpreting $\mathsf{PRF}_k(\mathsf{m}) = \mathsf{Com}_{\mathsf{ck}}(k)$, where $\mathsf{ck} \leftarrow \mathcal{G}(\mathsf{m})$ for a random oracle $\mathcal{G}$ mapping messages to commitment keys and the key k serves as the commitment randomness.

Let PRF be a PRF defined as in Sec. 2.2 and Π be a NIZK, proving the following relation

$$R_{\text{vrf}} = \left\{ ((\mathsf{m}, \mathsf{pk}, v), (f, k)) \ : \ \begin{array}{c} f \otimes \mathsf{pk} = \mathsf{PRF}_k(0) \ \wedge \ f \otimes v = \mathsf{PRF}_k(\mathsf{m}) \\ \wedge f \in F \ \wedge \ f' \cdot k \in \mathcal{K}' \quad \forall f' \in F \end{array} \right\}, \quad (6)$$

for a message m, a public key pk, a PRF output v, and a set $F \subseteq \mathfrak{R}$ of "relaxation factors". To allow for the use of *efficient* lattice-based zero-knowledge proofs, it is necessary to relax the relation guaranteed by the NIZK and, therefore, we introduce a *relaxation factor* f. For standard NIZKs outside of the lattice setting, we simply have $f = 1$ (and $F = \{1\}$), but efficient proofs in the lattice setting have a *soundness gap*, where the proved relation has the additional relaxation factor while an *honest* prover would simply use $f = 1$. Hence, we allow the existence of such a relaxation factor as, e.g., in [ESS+19, ESLL19]. We show in the uniqueness proof of the VRF how to handle this relaxation (see Thm 1). We next describe the generic VRF construction.

$\mathsf{V.ParamGen}(1^\lambda, \mathsf{PRF}, \Pi)$: Generate NIZK public parameters $\mathsf{pp} \leftarrow \Pi.\mathsf{Gen}(1^\lambda)$, and output pp.

$\mathsf{V.KeyGen}(\mathsf{pp})$: Sample a randomness $k \xleftarrow{\$} \mathcal{K}$ and compute $\mathsf{pk} \leftarrow \mathsf{PRF}_k(0)$. Return $(\mathsf{pk}, \mathsf{sk})$ for $\mathsf{sk} = k$.

$\mathsf{V.Eval}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{sk}, \mathsf{m})$: Given the message m, together with the key pair pk and $\mathsf{sk} = k$, proceed as follows:

- Compute $v \leftarrow \mathsf{PRF}_k(\mathsf{m})$.
- Run the NIZK proof system to generate a proof for the relation in (6).

$$\pi \leftarrow \Pi.\mathsf{Prove}_{\mathsf{pp}}((\mathsf{m}, \mathsf{pk}, v), (v; k)).$$

- Output v as the VRF value and π as the proof.

$\mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{m}, v, \pi)$: This algorithm verifies the VRF value v as below.

- Return $\Pi.\mathsf{Ver}_{\mathsf{pp}}(\mathsf{m}, \mathsf{pk}, v, \pi)$.

Existing examples. There are already example instantiations of the general VRF framework above. Particularly, ECVRF [PWH+17] is an example where the PRF is instantiated as $\mathsf{PRF}_k(\mathsf{m}) = g^k$ with $g \leftarrow \mathcal{G}(\mathsf{m})$ for a random oracle $\mathcal{G}$ (using multiplicative group notation) and the NIZK proof is a standard proof of equality of discrete log secrets as $\log_{g_0}(\mathsf{pk}) = \log_{g_1}(v)$. The PRF in this case satisfies statistical key-binding with (practically) unlimited pseudorandomness (i.e., $\kappa = 2^\lambda$ for a security parameter λ), and hence we get an unconditionally unique VRF with (practically) unlimited executions per key pair.

Another example is the few-time lattice-based VRF proposal in [EKS+21], where the PRF is instantiated as $\mathsf{PRF}_k(\mathsf{m}) = \mathbf{Ar}$ with $\mathbf{A} \leftarrow \mathcal{G}(\mathsf{m})$ for a random oracle $\mathcal{G}$ and the secret PRF key $k = \mathbf{r}$ is a short random vector. The NIZK proof in this case is a relaxed proof of knowledge that proves $f \cdot \mathsf{pk} = \mathbf{Ar}'$ and $f \cdot v = \mathbf{Br}'$ for some relaxation factor f and short vector $\mathbf{r}'$. The PRF in this case satisfies computational binding (based on MSIS) with κ-time pseudorandomness for a very small $\kappa \leq 5$, and hence we get a computationally unique κ-time VRF.

Security analysis. The provability of the folklore VRF in Sec. 2.3 follows from the correctness of the NIZK. The VRF pseudorandomness, on the other hand, follows from simulatability of Π and pseudorandomness of PRF. We prove below that the above VRF framework satisfies uniqueness and pseudorandomness (defined in App. A.1).

Theorem 1. *If the NIZK proof* Π *is statistically (resp. computationally) sound, the PRF* PRF *is statistically (resp. computationally) key-binding with respect to extended key space* $\mathcal{K}'$ *and additively key-homomorphic, the set of relaxation factors* F *is a subset of the homomorphism space (i.e.,* $F \subseteq S$*), and any element of* F *is invertible in* $\mathfrak{R}$*, then the generic VRF constructed over* (PRF, Π) *in Sec. 2.3 satisfies unconditional (resp. computational) uniqueness.*

Proof. Suppose that an adversary $\mathcal{A}$ produces $(\mathsf{m}, \mathsf{pk}, v_1, \pi_1, v_2, \pi_2)$ such that $\mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{m}, v_1, \pi_1) = \mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{m}, v_2, \pi_2) = 1$. We want to show that $v_1 = v_2$.

Now, we use the extractor $\mathcal{E}$ of Π to extract (f_1^*, k_1^*) and (f_2^*, k_2^*) such that $((\mathsf{m}, \mathsf{pk}, v_1), (f_1^*, k_1^*)) \in R_{\mathrm{vrf}}$ and $((\mathsf{m}, \mathsf{pk}, v_2), (f_2^*, k_2^*)) \in R_{\mathrm{vrf}}$. If Π is *computationally* sound, then the extraction works against a PPT $\mathcal{A}$ (except for a negligible probability). Then, we get the following expressions

$$f_1^* \otimes \mathsf{pk} = \mathsf{PRF}_{k_1^*}(0) \implies f_2^* \otimes f_1^* \otimes \mathsf{pk} = \mathsf{PRF}_{f_2^* \cdot k_1^*}(0), \tag{7}$$

$$f_1^* \otimes v_1 = \mathsf{PRF}_{k_1^*}(\mathsf{m}), \tag{8}$$

$$f_2^* \otimes \mathsf{pk} = \mathsf{PRF}_{k_2^*}(0) \implies f_1^* \otimes f_2^* \otimes \mathsf{pk} = \mathsf{PRF}_{f_1^* \cdot k_2^*}(0), \tag{9}$$

$$f_2^* \otimes v_2 = \mathsf{PRF}_{k_2^*}(\mathsf{m}). \tag{10}$$

By the statistical (resp. computational) key-binding property of PRF, $\mathfrak{R}$ being commutative, and (7) and (9), we must have $f_2^* \cdot k_1^* = f_1^* \cdot k_2^*$ over $\mathfrak{R}$ against the (resp. PPT) adversary $\mathcal{A}$ except for a negligible probability.

Then, by (8) and (10), and the key-homomorphism of PRF, we get

$$f_2^* \otimes f_1^* \otimes v_1 = \mathsf{PRF}_{f_2^* \cdot k_1^*}(\mathsf{m}) = \mathsf{PRF}_{f_1^* \cdot k_2^*}(\mathsf{m}) = f_1^* \otimes f_2^* \otimes v_2,$$

where the middle equality follows from the fact that $f_2^* \cdot k_1^* = f_1^* \cdot k_2^*$ over $\mathfrak{R}$. Hence, we get $f_2^* \otimes f_1^* \otimes v_1 = f_1^* \otimes f_2^* \otimes v_2$, and thus $v_1 = v_2$ by the relaxation factor invertibility property. □

Remark 1. Note in the above uniqueness proof that, the key-binding property of the PRF is only applied on pk, and not on (v_1, v_2). Hence, it is in fact sufficient if v is generated via a weaker PRF evaluation *without* a key-binding property, which is one of the optimizations we employ in LaV in Sec. 6.3.

Theorem 2. *If the PRF* PRF *has* κ*-pseudorandomness for* $\kappa \geq 1$*, and* Π *is simulatable, then the generic VRF constructed over* (PRF, Π) *in Sec. 2.3 is* κ*-pseudorandom.*

Proof (Theorem 2). Let $\mathsf{pp} \leftarrow \mathsf{V.ParamGen}(1^\lambda)$ and $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{V.KeyGen}(\mathsf{pp})$.
Simulation of $\mathcal{O}_{\mathrm{VEval}}$ queries. For a $\mathsf{V.Eval}$ output (v, π), the simulator Sim uses the simulator of Π to generate π. For the simulation of v, Sim samples $v \xleftarrow{\$} \{0,1\}^{m(\lambda)}$.

Since $\mathcal{A}$ is restricted making at most $\kappa - 1$ queries to $\mathcal{O}_{\mathrm{VEval}}$, and Π is simulatable, the output of Sim is (computationally) indistinguishable from a real output of $\mathsf{V.Eval}$.

Using a standard hybrid argument where $\mathcal{O}_{\mathrm{VEval}}$ queries are simulated as above and v_0 at Step 4 of $\mathsf{Exp\text{-}PRand}$ is replaced by $v_0 \xleftarrow{\$} \{0,1\}^{m(\lambda)}$, we conclude that PPT $\mathcal{A}$ has a negligible probability over $\frac{1}{2}$ of outputting $b' = b$ by the fact that total number of calls to $\mathsf{V.Eval}$ or its oracle did not exceed κ. □

2.4 LANES Framework

In this section, we recall the LANES framework [ALS20, ENS20, LNS20] without going into its technical details as we will use it as a black-box. Our description is similar to that in [LNS20]. The framework allows one to prove (unstructured)

linear and multiplicative relations over $\mathbb{Z}_q$ about a committed message without leaking the secret message information. The zero-knowledge proof is performed over a polynomial ring $\mathcal{R}_{q,d} = \mathbb{Z}_q[X]/(X^d+1)$ for a power-of-2 d while allowing $\mathcal{R}_{q,d}$ to split into l sub-rings for a parameter $2 \leq l \leq d$ by choosing a prime modulus q with $q \equiv 2l+1 \pmod{4l}$. We stress here that even though the proof is performed over $\mathcal{R}_{q,d}$, the proved relations hold over $\mathbb{Z}_q$.[7] Suppose that the prover $\mathcal{P}$ has a vector $\overrightarrow{m} = (\overrightarrow{m}_1, \ldots, \overrightarrow{m}_N)$ with $\overrightarrow{m}_i \in \mathbb{Z}_q^l$ for $N \geq 1$ and wants to prove the satisfiability of a public set, mp, of polynomials in N variables (for multiplicative proof) $P_i : (\mathbb{Z}_q^l)^N \to \mathbb{Z}_q^{\gamma_i l}$ with maximal degree α and $\gamma_i \geq 1$, where addition and multiplication are done component-wise. Further, we let $\mathsf{ulp} = (\mathbf{A}, \overrightarrow{u}) \in \mathbb{Z}_q^{vl \times Nl} \times \mathbb{Z}_q^{vl}$ denote the public statement of the linear relation the prover wants to prove (i.e., $\mathbf{A}\overrightarrow{m} = \overrightarrow{u}$). One simply pads zero rows, if needed, to make sure that the number of rows of $\mathbf{A}$ is a multiple of l. We also define k as the smallest positive integer such that $q^{-kd/l}$ is negligible.

Overall, the LANES framework proves knowledge of $\overrightarrow{m} \in \mathcal{L}(\mathsf{mp}, \mathsf{ulp})$ for

$$\mathcal{L}(\mathsf{mp}, \mathsf{ulp}) = \left\{ \overrightarrow{m} \in \mathbb{Z}_q^{Nl} \,:\, \forall P \in \mathsf{mp},\, P(\overrightarrow{m}) = \overrightarrow{0} \bmod q \,\wedge\, \mathbf{A}\overrightarrow{m} = \overrightarrow{u} \bmod q \right\}.$$

That is, the target relation R_{LANES} for a commitment key ck is the following

$$(\mathsf{ck}, (\mathsf{mp}, \mathsf{ulp}), \overrightarrow{m}) \in R_{\mathsf{LANES}} \iff \overrightarrow{m} \in \mathcal{L}(\mathsf{mp}, \mathsf{ulp}). \tag{11}$$

Let us present LANES as a CP protocol as described in Section 2.1, where the commitment scheme is instantiated using the BDLOP commitment [BDL+18].

$\mathsf{pp} \leftarrow \mathsf{LANES.Gen}(1^\lambda)$: generate a commitment key ck for the BDLOP commitment, specifying the message, randomness and commitment spaces. Generate further systems parameters pp', if needed. Output $\mathsf{pp} = (\mathsf{ck}, \mathsf{pp}')$.

$(t; t') \leftarrow \mathsf{LANES.Com}_{\mathsf{pp}}(\overrightarrow{m})$: sample a randomness $\mathbf{r} \in \mathbb{S}_1^{n+\ell+N+\alpha}$ for the BDLOP commitment and commit to the message $\hat{\mathbf{m}} = (\hat{\mathbf{m}}_1, \ldots, \hat{\mathbf{m}}_N) \in \mathcal{R}_{q,d}^N$ where $\hat{\mathbf{m}}_i$ is the polynomial in $\mathcal{R}_{q,d}$ whose CRT coefficient vector is $\overrightarrow{m}_i$ for $i = 1, \ldots, N$. Output the commitment t and the secret state information t'.

$\pi \leftarrow \mathsf{LANES.Prove}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), (t; t'))$: run a NIZK proof (see, e.g., [LNS20, Fig. 8]) to prove relation (11) for $\overrightarrow{m}$. Output a proof π.

$0/1 \leftarrow \mathsf{LANES.Ver}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), t, \pi)$: Check that π is a valid proof of knowledge for the relation (11).

The LANES output (t, π) requires (without compression) a total communication of $(n+N+\alpha+1)d \log q + k \cdot (n+\ell+N+\alpha)d \log(12\mathfrak{s})$ bits, where $\mathfrak{s}$ denotes the standard deviation of the discrete Gaussian distribution that the masked randomness follows. Note that the communication size only depends on the maximal polynomial degree α, not the individual degrees of P_i's. With the

[7] We note here that for $l < d$, the proved relations actually hold over $\mathbb{F}_{q^{d/l}}$. However, with a shortness proof of the form $P_i(x) = \prod_{j \in [\beta]}(x-j)$ for some $\beta < q \in \mathbb{Z}^+$, the proved relation is restricted to $\mathbb{Z}_q \subseteq \mathbb{F}_{q^{d/l}}$. This is explained further in [ENS20, App. A]. We have such a shortness proof for all of our applications in this work, and therefore, our description is focused on $\mathbb{Z}_q$.

compression techniques in [BG14, DLL+18] and considering the entropy of the discrete Gaussian (instead of a worst-case tail bound), the output size can be reduced to about (neglecting the size of very small "hints")

$$nd(\log q - D) + (N + \alpha + 1)d \log q + k \cdot (\ell + N + \alpha)d \log(4.13 \cdot \mathfrak{s}) \quad \text{bits}, \quad (12)$$

where D denotes the number of least significant bits dropped from commitment (a.k.a. *commitment compression*). A typical choice of D is around 13. The constant 4.13 is the result of our empirical tests that showed the entropy of a discrete Gaussian variable with standard deviation $\mathfrak{s}$ is very close to $\log(4.13 \cdot \mathfrak{s})$ for a wide range of parameters. A reasonable choice of the standard deviation would be $\mathfrak{s} \approx w\sqrt{k(\ell + N + \alpha)d}$ when using the optimized rejection sampling in [LNS21a], where w is an upper-bound on the ℓ_1-norm of the challenge c used in the protocol (see, e.g., the fourth move of [ENS20, Fig. 3]). Alternatively, we can use the very recent results of [KLSS23] to set $\mathfrak{s} \approx 2\sqrt{2}w\mathfrak{s}_0$ for $\mathfrak{s}_0 = \sqrt{\ln(2d(1 + 1/\varepsilon))}/\pi$ with, e.g., $\varepsilon = 2^{-100}$. The advantages in the latter case are (i) $\mathfrak{s}$ is independent of the (dimension) parameters (k, ℓ, N, α, d), (ii) no rejection sampling (inside LANES) is needed, and (iii) the security argument relies on the standard MLWE assumption (instead of the "Extended-MLWE" assumption in [LNS21a]).

It is important to note that the commitment phase LANES.Com does not rely on the multiplicative-linear relations (mp, ulp), which we will exploit in Section 4. The soundness and zero-knowledge/simulatability properties of this framework were established in [ALS20, ENS20, LNS20] and we refer the reader to them for more details.

A classical use-case of LANES is to prove knowledge of a message $\overrightarrow{m}$ with small coordinates, say in $[0, T-1]$ with $T < q$, that also satisfies a linear relation $\mathbf{A}\overrightarrow{m} = \overrightarrow{u}$.[8] Using base-$\beta$ integer decomposition (a.k.a. 'gadget') matrices, the latter relation can easily be transformed into an equivalent relation $\mathbf{A}'\overrightarrow{m}' = \overrightarrow{u}$, where $T = \beta^r$ and $\overrightarrow{m}'$ is r times bigger than $\overrightarrow{m}$ (i.e., $\dim(\overrightarrow{m}') = r \cdot \dim(\overrightarrow{m})$). In this case, it is sufficient to prove that $m_i(m_i - 1) \cdots (m_i - (\beta - 1)) = 0$ for each coordinate m_i of $\overrightarrow{m}'$. This is a multiplicative relation of degree $\alpha = \beta$ that will contribute to mp. Looking now at the proof length in (12), for such protocols, the LANES framework performs the best by choosing α that minimizes $\dim(\overrightarrow{m}') + \alpha = N \cdot r + \alpha = N \cdot \log_\alpha(T) + \alpha$.

In the rest of the paper, we will use hatted notations like $\hat{d}, \hat{q}$ to distinguish the parameters of LANES from the rest of the protocol (if they are indeed different).

[8] We note here that one does not necessarily need to consider positive ranges $[0, T-1]$. It is straightforward to "shift" the range to support a more general range $[a, b]$ with $a \leq b \in \mathbb{Z}$. For example, proving knowledge of $\overrightarrow{m} \in [a, b]^N$ with $\mathbf{A}\overrightarrow{m} = \overrightarrow{u}$ is equivalent to proving knowledge of $\overrightarrow{m}' \in [0, b - a]^N$ such that $\mathbf{A}\overrightarrow{m}' = \overrightarrow{u}'$ for $\overrightarrow{u}' := \overrightarrow{u} - \mathbf{A}\overrightarrow{a}^N$ and $\overrightarrow{a}^N := (a, \ldots, a) \in \mathbb{Z}^N$. Hence, the important part is the width, T, of the range.

3 Generalized Challenge Difference Invertibility Results

In this section, we generalize recent results [ESZ22, ALS20] on invertibility of challenge differences in polynomial rings based on Fourier analysis. Our generalization extends the *Partition-and-Sample* (PaS) challenge distribution of [ESZ22] and the results of [ALS20] to allow challenge polynomials of infinity norm γ for any $\gamma \geq 1$, extending the case $\gamma = 1$ in [ESZ22, ALS20]. We require the $\gamma > 1$ case for our efficient VRF construction in Sec. 6.

Let $l \leq d$ be powers of 2 and $q \equiv 2l+1 \pmod{4l}$ and $\delta := d/l$. Fix a primitive $2l$'th root of unity ζ in $\mathbb{Z}_q$. Then, the polynomial X^d+1 factors into l irreducible polynomials $g_i(X) := X^\delta + \zeta_i$ modulo q, where for $i \in [l]$, $\zeta_i := \zeta^{2i+1}$ are the primitive $(2l)$-th roots of unity in $\mathbb{Z}_q$.

For $a(X) \in \mathcal{R}_{q,d}$ and $i \in [l]$, we denote by $a\{i\}(X) := a(X) \bmod g_i(X)$ the i'th CRT slot of $a(X)$. Let $\mathbb{S}^{(\delta)}_{\gamma,d}$ be the set of polynomials in $\mathbb{S}_{\gamma,d}$ of the form $f(X) = f_0 + f_\delta X^\delta + \cdots + f_{(l-1)\delta} X^{(l-1)\delta}$. Our bounds apply to the challenge set $\mathcal{C}$, defined as

$$\mathcal{C} = \left\{ \tilde{c}_0 + \tilde{c}_1 X + \cdots + \tilde{c}_{\delta-1} X^{\delta-1} : \tilde{c}_i \in \mathbb{S}^{(\delta)}_{\gamma,d} \wedge \mathsf{HW}(\tilde{c}_i) \leq \tilde{w} \right\}. \tag{13}$$

Note that challenges $c(X) = \sum_{k=0}^{\delta-1} \tilde{c}_i(X) X^k$ in $\mathcal{C}$ have total Hamming weight $w \leq \delta\tilde{w}$ with non-zero coefficients in $[-\gamma, +\gamma]$, and the coefficient index set $S_k := \{j \in [d] : j = k \bmod \delta\}$ appearing in $\tilde{c}_k(X)$ has weight $\leq \tilde{w}$ for each $k \in [\delta]$. We consider the challenge probability distribution $\mathfrak{C}$ on $\mathcal{C}$ defined as follows: for each $k \in [\delta]$, we choose a uniformly random subset $T_k \subset S_k$ of size $|T_k| = \tilde{w}$ and independently sample each challenge coefficient in T_k to be zero with probability p_z and uniformly random on $[-\gamma, +\gamma]\backslash 0$ with probability $1-p_z$.

Lemma 1 (Generalization of [ESZ22, Le.1] and [ALS20, Le.3.3]). *Let P_2 denote the probability distribution of the coefficient $\tilde{c}_{i,k}$ of X^k in the i'th CRT slot $c\{i\} = c(X) \bmod g_i(X)$ of a challenge $c(X)$ sampled from the distribution $\mathfrak{C}$ on $\mathcal{C}$ defined above. Then, for $\eta := \frac{l^{\tilde{w}}(l-\tilde{w})!}{l!}$ and all $i \in [l]$ and $k \in [\delta]$, we have:*

$$\max_y P_2(y) \leq \min(M_2, N_2), \tag{14}$$

$$M_2 := \frac{\eta}{q}\left(1 + 2l \sum_{j \in \mathbb{Z}_q^*/<\zeta_i>} |\hat{\mu}(j)|^{\tilde{w}}\right), \tag{15}$$

$$N_2 := \frac{1}{q}\left(1 + 2l \sum_{j \in \mathbb{Z}_q^*/<\zeta_i>} |\hat{P}_2(j)|\right), \tag{16}$$

and for $j \in \mathbb{Z}_q^*/ < \zeta_i >$, we define

$$\hat{\mu}(j) := \frac{1}{l} \sum_{k \in [l]} \hat{\mu}_k(j), \tag{17}$$

$$\hat{P}_2(j) := \frac{1}{\binom{l}{\tilde{w}}} \sum_{S\subset[l],|S|=\tilde{w}} \prod_{k\in S} \hat{\mu}_k(j), \tag{18}$$

$$\hat{\mu}_k(j) := p_z + \frac{1-p_z}{\gamma} \sum_{b\in[1,\gamma]} \cos(2\pi j b \zeta_i^k/q). \tag{19}$$

Proof (Lemma 1). The proof of the bound M_2 follows the same arguments as in the proof of [ESZ22, Le.1] in the case $\gamma = 1$, while the proof of the bound N_2 is a generalization of [ALS20, Le.3.3] to the case $\gamma > 1$ and $\tilde{w} \leq l$ so we only summarise the differences here.

Observe that P_2 is the distribution of the random variable $Y_2 := \sum_{j\in[\tilde{w}]} h_j \zeta_i^{k_j}$ over $\mathbb{Z}_q$, with $(\mathbf{h} = (h_1, ..., h_w), \mathbf{k} = (k_1, ..., k_w))$ sampled from a distribution D_2 as follows: h_j's are identically and independently distributed (iid) with probability p_z to be zero and probability $(1-p_z)/(2\gamma)$ to take each value in $[-\gamma, +\gamma] \setminus 0$, and $\mathbf{k}$ is sampled uniformly at random from the set of all w-tuples from $[r]$ with *distinct* coordinates (i.e. $k_j \neq k_{j'}$ for $j \neq j'$).

We first derive the bound M_2. Similarly to [ESZ22, Le.1], our first step (below) is to compute a bound M_1 on a slightly different distribution P_1 of random variable $Y_1 := \sum_{j\in[w]} h_j \zeta_i^{k_j}$ defined similarly to Y_2 except that in the distribution D_1 of $(\mathbf{k}, \mathbf{h})$, the k_j's are sampled iid from the uniform distribution on $[l]$ (i.e. without the distinct coordinate requirement). The second step follows the same Rényi divergence of order ∞ argument as in [ESZ22, Le.1] to give $M_2 \leq \eta \cdot M_1$.

To complete our first step and prove the Lemma, it therefore suffices to bound M_1. For this, we generalise the Fourier analysis approach of [ESZ22, Le.1]. Writing $P_1 : \mathbb{Z}_q \to [0, 1]$ in terms of its Fourier transform $\hat{P}_1$ over $\mathbb{Z}_q$ (with respect to the orthogonal Fourier basis $\{\chi_j(x) = \exp(-2\pi\imath jx/q)\}_{j\in\mathbb{Z}_q}$, where $\imath := \sqrt{-1}$) to get:

$$P_1(y) = \frac{1}{q} + \frac{1}{q} \sum_{j\in\mathbb{Z}_q^*} \hat{P}_1(j) \cdot \exp(-2\pi\imath jy/q). \tag{20}$$

As the coordinates of $\mathbf{h}$ and $\mathbf{k}$ are iid, $\hat{P}_1$ is the $\tilde{w}$-fold self-convolution of the distribution μ of each term $h_j\zeta_i^{k_j}$ in Y_1. We have $\mu(0) = p_z$. We now study the distribution of $h_j\zeta_i^{k_j}$ conditioned on h_j being non-zero, which happens with probability $(1-p_z)$. In this case, we can write $h_j = m_j \cdot s_j$ where m_j is uniformly random on $[1, \gamma]$ and s_j is uniformly random on $\{-1, 1\}$. Since $\zeta_i^l = -1$, $v := s\zeta_i^k$ runs through all elements of the group $< \zeta_i >$ of $2l$'th roots of unity in $\mathbb{Z}_q^*$ as (s, k) run through $\{-1, +1\} \times [l]$. Therefore the random variable $s_j\zeta_i^{k_j}$ is uniformly random on $< \zeta_i >$ and therefore, for each fixed $m \in [1, \gamma]$, the random variable $ms_j\zeta_i^{k_j}$ is uniformly random on the coset $m \cdot < \zeta_i >$ of $< \zeta_i >$ containing m. Since m_j is uniformly random on $[1, \gamma]$, it follows that $\mu(0) = p_z$ and $\mu(b) = (1 - p_z)n_b/(2l\gamma)$ for $b \in \bigcup_{b'\in[1,\gamma]} b' < \zeta_i >$, where we denote by n_b the number of $b' \in [1, \gamma]$ in the same coset as b (i.e. satisfying $b'b^{-1} \in < \zeta_i >$). So from the convolution property of the Fourier transform, we have $\hat{P}_1(j) = \hat{\mu}(j)^{\tilde{w}}$.

Computing the Fourier transform $\hat{\mu}$ of μ, we get for each $j \in \mathbb{Z}_q^*$ that

$$\begin{aligned}\hat{\mu}(j) &:= p_z + \frac{1-p_z}{2l\gamma} \sum_{v \in \bigcup_{b' \in [1,\gamma]} b' < \zeta_i >} n_v \exp(2\pi\imath jv/q) \\ &= p_z + \frac{1-p_z}{2l\gamma} \sum_{b \in [1,\gamma]} \sum_{k \in [2l]} \exp(2\pi\imath jb\zeta_i^k/q) \\ &= p_z + \frac{1-p_z}{l\gamma} \sum_{b \in [1,\gamma]} \sum_{k \in [l]} \cos(2\pi jb\zeta_i^k/q),\end{aligned}$$

where we have used $\zeta_i^l = -1$, $\cos(.)$ is even, and $\sin(.)$ odd.

The rest of the proof is identical to [ESZ22, Le.1]:

$$\begin{aligned}P_1(y) &= \frac{1}{q}\left(1 + \sum_{j \in \mathbb{Z}_q^*} \hat{\mu}(j)^{\tilde{w}} \exp(-2\pi\imath jy/q)\right) \\ &\leq \frac{1}{q}\left(1 + \sum_{j \in \mathbb{Z}_q^*} |\hat{\mu}(j)|^{\tilde{w}}\right) = \frac{1}{q}\left(1 + 2l \sum_{j \in \mathbb{Z}_q^*/<\zeta_i>} |\hat{\mu}(j)|^{\tilde{w}}\right),\end{aligned}$$

where the inequality uses the triangle inequality (taking magnitude) and the equality uses the fact that $\hat{\mu}(j) = \hat{\mu}(j')$ for j, j' in the same coset of $< \zeta_i >$ in $\mathbb{Z}_q^*$ and that the size of each coset is $2l$. The last bound is M_1, as claimed.

For the bound N_2, we directly bound the distribution P_2 of $Y_2 = \sum_{k \in S} h_k \zeta_i^k$ similarly to [ALS20, Le.3.3]. For a subset $S \subseteq [l]$ of size $|S| = \tilde{w}$, let $P_2(\cdot|S)$ denote the conditional distribution of Y_2 over the choice of the h_j's, conditioned on $\{k_1, \ldots, k_{\tilde{w}}\} = S$. Since $\{k_1, \ldots, k_{\tilde{w}}\}$ is a uniformly random subset of $[l]$ of size $\tilde{w}$, we have $P_2(x) = \frac{1}{\binom{l}{\tilde{w}}} \sum_{S \subset [l], |S| = \tilde{w}} P_2(x|S)$. Let $\hat{P}_2(\cdot|S)$ and $\hat{P}_2$ denote the Fourier transform of $P_2(\cdot|S)$ and P_2, respectively. By linearity of the Fourier transform, we therefore have: $\hat{P}_2(j) = \frac{1}{\binom{l}{\tilde{w}}} \sum_{S \subset [l], |S| = \tilde{w}} \hat{P}_2(j|S)$. Now, for each fixed S, the $\tilde{w}$ terms in the sum $Y_2 = \sum_{k \in S} h_k \zeta_i^k$ are independent, so the distribution $P_2(\cdot|S)$ is a $\tilde{w}$-fold convolution of the distributions μ_k of $h_k \zeta_i^k$ for $k \in S$, and by the convolution property of Fourier transform, $\hat{P}_2(j|S) = \prod_{k \in S} \hat{\mu}_k(j)$. Since h_k is zero with probability p_z and conditioned on h_k being non-zero, μ_k is uniformly random over $([-\gamma, \gamma] \setminus 0) \cdot \zeta_i^k$, we find that $\hat{\mu}_k(j)$ and $\hat{P}_2(j)$ are given by Eqs. (19) and (18), respectively.

The rest of the proof is similar to the one for M_1:

$$\begin{aligned}P_2(y) &= \frac{1}{q}\left(1 + \sum_{j \in \mathbb{Z}_q^*} \hat{P}_2(j) \exp(-2\pi\imath jy/q)\right) \\ &\leq \frac{1}{q}\left(1 + \sum_{j \in \mathbb{Z}_q^*} |\hat{P}_2(j)|\right) = \frac{1}{q}\left(1 + 2l \sum_{j \in \mathbb{Z}_q^*/<\zeta_i>} |\hat{P}_2(j)|\right),\end{aligned}$$

q	d	l	$\tilde{w}$	γ	$\log_2 p_{\text{inv}}$	$\lvert\mathcal{C}\rvert$
61	32	2	2	16	-91.5	2^{160}
13	64	2	2	2	-99	2^{128}

Table 2. Sample challenge space parameters and challenge difference invertibility bounds over $\mathcal{R}_{q,d}$. Here, q and γ are minimised subject to challenge invertibility probability bound $p_{\text{inv}} \leq 2^{-90}$ computed using Corollary 1.

where the inequality uses the triangle inequality (taking magnitude) and the equality uses the fact that the size of each coset is $2l$ and $\hat{P}_2(j) = \hat{P}_2(j')$ for j, j' in the same coset of $< \zeta_i >$ in $\mathbb{Z}_q^*$. The last fact holds because, writing $j' = j\zeta_i^c$ for some c, we have $j'\zeta_i^k = j\zeta_i^{k+c}$. So, for any S, $\prod_{k\in S} \hat{\mu}_k(j) = \prod_{k\in S'} \hat{\mu}_k(j')$ with $S' := S - c \bmod l$ (i.e. the set S' is obtained by subtracting c mod l from each element in S). As the mapping $S \mapsto S' = S - c$ is one-to-one on the collections of subsets of $[l]$ of size $\tilde{w}$, the sum over S in $\hat{P}_2$ remains unchanged for j, j' in the same coset of $< \zeta_i >$ in $\mathbb{Z}_q^*$. The last bound above is N_2, as claimed. □

Using the independence of the δ coefficients of each CRT slot, and the fact that a challenge difference $c(X) - c'(X)$ is non-invertible in $\mathcal{R}_{q,d}$ if and only if one of its CRT slots is 0, we immediately get the following corollary.

Corollary 1 (Generalization of [ESZ22, Cor.1]). *Let $c(X), c'(X)$ denote a pair of challenges independently sampled from distribution $\mathfrak{C}$. The probability that $c(X) - c'(X)$ is not invertible in $\mathcal{R}_{q,d}$ is upper bounded by $p_{\text{inv}} := l \min(M_2, N_2)^\delta$, where M_2, N_2 are the bounds from Lemma 1.*

We remark that as in [ESZ22], we can split the computation of the invertibility bound of Cor. 1 into two phases. In the longer pre-computation step that does not depend on w, we compute a table of $\hat{\mu}$ and in the faster post-computation step, we compute the bound M_2 using this table. The computation time cost $O(q/l)$ of our post-computation step is similar to that in [ESZ22]. However, our table pre-computation step computation time cost is $O(\gamma q/l)$, which is $O(\gamma)$ times larger than the table computation time in [ESZ22] in the case $\gamma = 1$. Table 2 shows the resulting computed bounds for two sets of challenge space parameter choices. Our actual optimised VRF parameter set in Sec. 6.5 uses the parameters in the first row of the table ($d = 32$).

4 LANES$^+$: A Framework for Hybrid Exact/Relaxed Lattice-Based Proofs

We recall from Sec. 1.2, that the goal of LANES$^+$ is to prove knowledge of a tuple $(\bar{c}, \mathbf{m}, \mathbf{r}, \overrightarrow{v}) \in \mathcal{L}^+(\mathsf{mp}, \mathsf{ulp})$ (i.e., $(\mathsf{ck}, (\mathsf{mp}, \mathsf{ulp}), (\bar{c}, \mathbf{m}, \mathbf{r}, \overrightarrow{v})) \in R_{\mathsf{LANES}^+}$) such that

$$\mathcal{L}^+(\mathsf{mp}, \mathsf{ulp}) = \left\{ (\bar{c}, \mathbf{m}, \mathbf{r}, \overrightarrow{v}) \; : \; \begin{array}{r} \mathbf{t} = \mathbf{A}\mathbf{r} + \mathbf{B}\mathbf{m} \text{ over } \mathcal{R}_{q,d} \wedge \mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v} \bmod q \\ \wedge P(\overrightarrow{\mathbf{m}}, \overrightarrow{v}) = 0 \bmod q \; \forall P \in \mathsf{mp} \wedge \\ \|\bar{c}\mathbf{r}\|_\infty \leq \gamma_r \wedge \|\bar{c}\|_\infty \leq \gamma_c \text{ for } \gamma_r, \gamma_c \ll q \in \mathbb{Z}^+ \end{array} \right\}. \tag{21}$$

where $\mathsf{ulp} = ((\mathbf{A}, \mathbf{B}, \mathbf{t}), (\mathbf{G}_1, \mathbf{G}_2))$ and mp is a set of polynomials over $\mathbb{Z}_q$ as in Sec. 2.4. By setting $d = 1$, the whole relation becomes over $\mathbb{Z}_q$. Hence, there is no loss of generality and we stick to the naming 'unstructured' linear relation for $(\mathbf{A}, \mathbf{B}, \mathbf{t})$. Often the relation is over a polynomial ring for better efficiency.

As discussed in Section 1.2, the approach of LANES$^+$ to proving the hybrid exact/relaxed relation (21) is to use an efficient RPoK to prove the (typically) high-dimensional relation $\mathbf{t} = \mathbf{Ar} + \mathbf{Bm}$, and use the costly exact LANES framework only to prove the (typically) low-dimensional relations $\mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v} \bmod q$ and $P(\overrightarrow{\mathbf{m}}, \overrightarrow{v}) = 0 \bmod q$, along with the well-formedness of the RPoK masked message relation $\mathbf{f} = \mathbf{u} + c\mathbf{m}$ that links the RPoK and LANES proofs.

We provide the full LANES$^+$ protocol as a commit-and-prove protocol in Alg. 2, where $\mathsf{ulp} = ((\mathbf{A}, \mathbf{B}, \mathbf{t}), (\mathbf{G}_1, \mathbf{G}_2))$ as before. We write the steps relating to LANES in purple colour to make it easy to distinguish them from RPoK steps. The flag $\mathsf{flag}_{\mathsf{rs}}$ is used to specify if a rejection sampling on $\mathbf{m}$ is done.

4.1 Security Analysis

The analysis of our LANES$^+$ framework is fairly intuitive. Correctness follows straightforwardly from the completeness of a standard RPoK and the correctness of LANES. The simulatability (or zero-knowledge) property follows from the simulatability properties of a standard RPoK and LANES. The more difficult part is the soundness, which we look at more closely next.

Theorem 3. LANES$^+$ *protocol in Alg. 2 is*

1. *correct if* LANES *is correct,*
2. *simulatable if* LANES *is simulatable, and*
3. *knowledge sound if* LANES *is knowledge sound and any non-zero difference of challenges in* $\mathcal{C}$ *is invertible in* $\mathcal{R}_{q,d}$.

Proof. The correctness of LANES$^+$ follows straightforwardly. The simulation of LANES$^+$ output $(t_L, (\pi_L, \hat{\pi}))$ also follows via standard arguments as discussed next. By assumption, LANES is simulatable and thus (t_L, π_L) can be simulated using the simulator of LANES, given the public input $(\mathbf{f}, c)$ to the LANES prove algorithm. Here, $\mathbf{f}$ and c must be simulated first using the simulator for the remaining proof part $\hat{\pi} = (c, \mathbf{z}, \mathbf{f})$, which follows from the rejection sampling. In particular, if the 'uniform' rejection sampling in [Lyu09] is used for $\mathbf{z}$ (and $\mathbf{f}$), then simulation of $\mathbf{z}$ (and $\mathbf{f}$) is done by sampling each coefficient from a known uniform distribution. If the 'Gaussian' rejection sampling in [Lyu12] is used for $\mathbf{z}$ (and $\mathbf{f}$), which is what is described in Alg. 2, then simulation of $\mathbf{z}$ (and $\mathbf{f}$) is done by sampling each coefficient from a known discrete Gaussian distribution (i.e., $\mathbf{z} \xleftarrow{\$} \mathbb{D}^{\dim(\mathbf{r})}_{\phi\eta,d}$ and $\mathbf{u} \xleftarrow{\$} \mathbb{D}^{V}_{\phi_m\eta_m,d}$). If no rejection sampling is used, then each coordinate in $\mathbf{f}$ are simply sampled as a uniformly random element of $\mathcal{R}_{q,d}$. The simulator picks $c \xleftarrow{\$} \mathcal{C}$ and then programs the random oracle $\mathcal{H}$ such that $\mathcal{H}(\mathsf{pp}, \mathsf{mp}, \mathsf{ulp}, t_L, \mathbf{Az} + \mathbf{Bf} - c\mathbf{t}; \rho) = c$. This concludes the simulatability proof.

We now investigate soundness, which is the more critical property. Using a standard rewinding argument (e.g., [BN06]), we get two accepting protocol

Algorithm 2 LANES^+ : Framework for Hybrid Exact/Relaxed Proofs

procedure $\mathsf{LANES}^+.\mathsf{Gen}(1^\lambda)$
  Pick $\mathcal{H} : \{0,1\}^* \to \mathcal{C} \subseteq \mathcal{R}_{q,d}$
  $\mathsf{pp}_L \leftarrow \mathsf{LANES.Gen}(1^\lambda)$
  return $\mathsf{pp} = (\mathsf{pp}_L, \mathcal{H})$
end procedure

procedure $\mathsf{LANES}^+.\mathsf{Com}_{\mathsf{pp}}(\mathbf{m}, \mathbf{r}, \overrightarrow{v})$ ▷ $(\mathbf{m}, \overrightarrow{v}) \in \mathcal{R}_{q,d}^V \times \mathbb{Z}_q^{Ml}$ and $\mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v}$
  Set public params $\eta, \eta_m, \phi, \phi_m$ s.t. $\eta \geq \|c\mathbf{r}\|$ and $\eta_m \geq \|c\mathbf{m}\|$ for any $c \in \mathcal{C}$
  Sample msg masking $\mathbf{u} \xleftarrow{\$} \mathbb{D}_{\phi_m\eta_m,d}^V$ if $\mathsf{flag}_{\mathsf{rs}} = \mathsf{true}$; otherwise $\mathbf{u} \xleftarrow{\$} \mathcal{R}_{q,d}^V$
  $\overrightarrow{s} = (\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{m}}, \overrightarrow{v}) \in \mathbb{Z}_q^{2Vd+Ml}$
  $(t_L; t'_L) \leftarrow \mathsf{LANES.Com}_{\mathsf{pp}_L}(\overrightarrow{s})$
  return $(t; t') = (t_L; (t'_L, \mathbf{m}, \mathbf{r}, \overrightarrow{v}, \mathbf{u}))$ ▷ t is public and t' is secret
end procedure

procedure $\mathsf{LANES}^+.\mathsf{Prove}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), (t; t'); \rho)$ ▷ ρ is optional; only used as $\mathcal{H}$ input
  Parse $(t; t') = (t_L; (t'_L, \mathbf{m}, \mathbf{r}, \overrightarrow{v}, \mathbf{u}))$
  Sample short randomness masking $\mathbf{y} \xleftarrow{\$} \mathbb{D}_{\phi\eta,d}^{\dim(\mathbf{r})}$
  Compute $\mathbf{w} = \mathbf{A}\mathbf{y} + \mathbf{B}\mathbf{u}$
  $c \leftarrow \mathcal{H}(\mathsf{pp}, \mathsf{mp}, \mathsf{ulp}, t, \mathbf{w}; \rho)$
  $\mathbf{z} = \mathbf{y} + c \cdot \mathbf{r}$
  $\mathbf{f} = \mathbf{u} + c \cdot \mathbf{m} \in \mathcal{R}_{q,d}^V$
  Restart if $\mathsf{Rej}(\mathbf{z}, c\mathbf{r}, \phi, \eta)$
  Restart if $\mathsf{flag}_{\mathsf{rs}} = \mathsf{true}$ and $\mathsf{Rej}(\mathbf{f}, c\mathbf{m}, \phi_m, \eta_m)$
  $\mathsf{ulp}' = \left(\mathbf{L}, \begin{pmatrix} \overrightarrow{\mathbf{f}} \\ \overrightarrow{0} \end{pmatrix}\right)$ where $\mathbf{L} := \begin{pmatrix} \mathbf{I}_{Vd} & \mathbf{I}_V \otimes \mathsf{Rot}(c) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_1 & -\mathbf{G}_2 \end{pmatrix}$
  $\pi_L \leftarrow \mathsf{LANES.Prove}_{\mathsf{pp}_L}((\mathsf{mp}, \mathsf{ulp}'), (t_L; t'_L))$
  return the proof $\pi = (\pi_L, \hat{\pi})$ with $\hat{\pi} = (c, \mathbf{z}, \mathbf{f})$
end procedure

procedure $\mathsf{LANES}^+.\mathsf{Ver}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), t, \pi; \rho)$ ▷ ρ is an optional argument
  Parse $\pi = (\pi_L, (c, \mathbf{z}, \mathbf{f}))$
  If $\|\mathbf{z}\|_\infty > 6\phi\eta$ or ($\mathsf{flag}_{\mathsf{rs}} = \mathsf{true}$ and $\|\mathbf{f}\|_\infty > 6\phi_m\eta_m$), **return** 0
  Compute $\mathbf{w}' = \mathbf{A}\mathbf{z} + \mathbf{B}\mathbf{f} - c\mathbf{t}$
  If $c \neq \mathcal{H}(\mathsf{pp}, \mathsf{mp}, \mathsf{ulp}, t, \mathbf{w}'; \rho)$, **return** 0
  Set ulp' as in $\mathsf{LANES}^+.\mathsf{Prove}$
  return $\mathsf{LANES.Ver}_{\mathsf{pp}_L}((\mathsf{mp}, \mathsf{ulp}'), t_L, \pi_L)$
end procedure

outputs $\pi = (\pi_L, (c, \mathbf{z}, \mathbf{f}))$ and $\pi' = (\pi'_L, (c', \mathbf{z}', \mathbf{f}'))$ for $c \neq c'$ w.r.t. the same hash input $(\mathsf{pp}, \mathsf{mp}, \mathsf{ulp}, t, \mathbf{w}; \rho)$. From the verification Step 29, we have

$$\bar{c}\mathbf{t} = \mathbf{A}\bar{\mathbf{z}} + \mathbf{B}\bar{\mathbf{f}} \text{ over } \mathcal{R}_{q,d}, \tag{22}$$

where $\bar{c} := c - c'$, $\bar{\mathbf{z}} := \mathbf{z} - \mathbf{z}'$ and $\bar{\mathbf{f}} := \mathbf{f} - \mathbf{f}'$. Note that $\|\bar{\mathbf{z}}\|_\infty \leq 12\phi\eta$ and $\|\bar{\mathbf{f}}\|_\infty \leq 12\phi_m\eta_m$ (if $\mathsf{flag}_{\mathsf{rs}} = \mathsf{true}$) by Step 28.

Now, we will use the extractor $\mathcal{E}_0$ of LANES, which itself also relies on a standard rewinding, as in [ENS20, Theorem 4.1] to extract a witness $\overrightarrow{s}^*$. First, it is important to observe that the commitment phase LANES.Com is performed *before* the challenge computation at Step 17. The special soundness of LANES requires this commitment to be binding and thus a PPT adversary cannot find two distinct openings. As a result, when running $\mathcal{E}_0$ on both sets of transcripts w.r.t. c and c', the commitment opening returned by $\mathcal{E}_0$ will be the same for both cases, except with negligible probability.

With the above in mind, we use $\mathcal{E}_0$ to extract a witness $\overrightarrow{s}^* := (\overrightarrow{u}^*, \overrightarrow{m}^*, \overrightarrow{v}^*) \in \mathbb{Z}_q^{2Vd+Ml}$ for $\mathsf{ulp} = \left(\mathbf{L}, \begin{pmatrix} \overrightarrow{\mathbf{f}} \\ \overrightarrow{0} \end{pmatrix}\right)$ where $\mathbf{L} := \begin{pmatrix} \mathbf{I}_{Vd} & \mathbf{I}_V \otimes \mathsf{Rot}(c) & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_1 & -\mathbf{G}_2 \end{pmatrix}$ such that

$$P(\overrightarrow{s}^*) = 0 \bmod q \quad \text{for all } P \in \mathsf{mp}, \text{ and} \tag{23}$$

$$\mathbf{L} \cdot \begin{pmatrix} \overrightarrow{u}^* \\ \overrightarrow{m}^* \\ \overrightarrow{v}^* \end{pmatrix} = \begin{pmatrix} \overrightarrow{\mathbf{f}} \\ \overrightarrow{0} \end{pmatrix} \bmod q, \tag{24}$$

which is equivalent to

$$\mathbf{f} = \mathbf{u}^* + c \cdot \mathbf{m}^* \text{ over } \mathcal{R}_{q,d}, \text{ and} \tag{25}$$

$$\mathbf{G}_1 \overrightarrow{m}^* = \mathbf{G}_2 \overrightarrow{v}^* \text{ over } \mathbb{Z}_q, \tag{26}$$

where $\mathbf{u}^*$ and $\mathbf{m}^*$ are the vectors of polynomials in $\mathcal{R}_{q,d}$ corresponding to $\overrightarrow{u}^*$ and $\overrightarrow{m}^*$, respectively (i.e., $\overrightarrow{\mathbf{u}}^* = \overrightarrow{u}^*$ and $\overrightarrow{\mathbf{m}}^* = \overrightarrow{m}^*$).

From the above discussion for the same witness $\overrightarrow{s}^* = (\overrightarrow{u}^*, \overrightarrow{m}^*, \overrightarrow{v}^*)$, we similarly use $\mathcal{E}_0$ to obtain

$$\mathbf{f}' = \mathbf{u}^* + c' \cdot \mathbf{m}^* \text{ over } \mathcal{R}_{q,d}. \tag{27}$$

Plugging (25) and (27) into (22), we get

$$\bar{c}\mathbf{t} = \mathbf{A}\bar{\mathbf{z}} + \bar{c}\mathbf{B}\mathbf{m}^* \text{ over } \mathcal{R}_{q,d}, \tag{28}$$

for $\overrightarrow{s}^* = (\overrightarrow{u}^*, \overrightarrow{m}^*, \overrightarrow{v}^*)$ and $\overrightarrow{\mathbf{m}}^* = \overrightarrow{m}^*$. By assumption, $\bar{c}$ is invertible in $\mathcal{R}_{q,d}$, and hence the extractor can compute $\mathbf{r}^* := \bar{\mathbf{z}}/\bar{c} \bmod q$ such that (21) holds w.r.t. $(\bar{c}, \mathbf{m}^*, \mathbf{r}^*, \overrightarrow{v}^*)$. This concludes the proof. □

Remark 2. Note that the extracted randomness $\mathbf{r}^*$ in the proof of Theorem 3 is not proven to be short, but this is not needed for our applications. In our rounding proof and VRF applications, the shortness proof will be done using LANES for the message part, which will correspond to an error term. Moreover, we do also prove a relaxed relation as in (28), where the randomness $\bar{\mathbf{z}}$ is short.

Remark 3 (Using different system moduli). Suppose that we want to use different moduli, e.g., $\hat{q}$ in LANES and q in RPoK. To achieve this, we need to focus on the components that are used both in LANES and RPoK. In particular, we need to assume the following

1. $\|\overrightarrow{s}\|_\infty < \hat{q}/2$,
2. q is large enough that $\mathbf{f} = \mathbf{u} + c\mathbf{m}$ holds without mod q, (i.e. $\|\mathbf{f}\|_\infty < q/2$),
3. $\|\mathbf{f}\|_\infty, \|c\mathbf{m}^*\|_\infty < \hat{q}/4$,
4. $\hat{q}$ is large enough that $\mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v}$ holds without mod $\hat{q}$.

With the above assumptions, the witness $\overrightarrow{s} = (\overrightarrow{\mathbf{u}}, \overrightarrow{\mathbf{m}}, \overrightarrow{v})$ of LANES is a vector over $\mathbb{Z}$ with coordinates in $[-(\hat{q}-1)/2, (\hat{q}-1)/2]$, and hence can be seen as $\mathbb{Z}_{\hat{q}}$ elements without any change. Also, no coefficient of the expression $\mathbf{f} = \mathbf{u} + c\mathbf{m}$ exceeds q or $\hat{q}$, and it can be proven without any change in the two proof parts. Particularly, LANES will prove that $\overrightarrow{\mathbf{f}} = \overrightarrow{u}^* + \mathbf{I}_V \otimes \mathsf{Rot}(c) \cdot \overrightarrow{m}^* \bmod \hat{q}$ and $\overrightarrow{\mathbf{f}'} = \overrightarrow{u}^* + \mathbf{I}_V \otimes \mathsf{Rot}(c') \cdot \overrightarrow{m}^* \bmod \hat{q}$. Hence, $\overrightarrow{\mathbf{f}} - \overrightarrow{\mathbf{f}'} = \mathbf{I}_V \otimes (\mathsf{Rot}(c) - \mathsf{Rot}(c')) \cdot \overrightarrow{m}^* \bmod \hat{q}$. By the above infinity-norm assumptions, we get $\overrightarrow{\mathbf{f}} - \overrightarrow{\mathbf{f}'} = \mathbf{I}_V \otimes (\mathsf{Rot}(c) - \mathsf{Rot}(c')) \cdot \overrightarrow{m}^*$ over $\mathbb{Z}$, which implies that $\bar{\mathbf{f}} = \bar{c}\mathbf{m}^*$ over $\mathcal{R}_d := \mathbb{Z}[X]/(X^d+1)$ (without mod q or $\hat{q}$), as needed. Finally, the linear relation proven by LANES now holds over the ring $\mathcal{R}_d$ and hence it also holds over the ring $\mathcal{R}_{q,d}$ (with mod q).

The above assumptions in Remark 3 naturally hold for our application to VRFs because the message $\mathbf{m}$ will be an error term with coefficients much less than q and $\hat{q}$. Hence, we can also easily construct $\mathbf{f}$ via rejection sampling to make sure that it has relatively small coefficients. The linear relation $(\mathbf{G}_1, \mathbf{G}_2)$ will represent an integer decomposition of the error coefficients and hence $\mathbf{G}_1\overrightarrow{\mathbf{m}} = \mathbf{G}_2\overrightarrow{v}$ will readily hold over $\mathbb{Z}$. As a result, we will have more flexibility in choosing concrete parameters in our application without imposing aggressive conditions.

The total average number of repetitions for LANES^+ will be about $\mu(\phi) \cdot \mu(\phi_m) \cdot M_L$ (and $\mu(\phi) \cdot M_L$ if no rejection sampling is done for $\mathbf{m}$), where M_L denotes the average number of repetitions in LANES and $\mu(\phi) = e^{12/\phi + 1/(2\phi^2)}$ as defined in Alg. 5. Recall that $M_L = 1$ if the results of [KLSS23] are used.

5 Proof of Rounding

In this section, we describe our protocol that allows proving knowledge of a vector satisfying a rounding relation of the form

$$R_{\mathrm{rnd}} = \left\{ ((\mathbf{B}, \mathbf{v}); \mathbf{s}) \ : \ \mathbf{s} \in \mathcal{R}_{q,d}^m \wedge \mathbf{v} = \lfloor \mathbf{B}\mathbf{s} \rceil_p \bmod p \right\}. \tag{29}$$

In the rest of the paper, q is assumed to be a multiple of p so that we can use Fact 1. Typical applications would require that $(\mathbf{B}, \mathbf{v})$ does not leak information about $\mathbf{s}$ since otherwise it may not make sense to prove the rounding relation in zero-knowledge. However, we do not necessarily assume $\mathbf{B}$ to be binding.

The proof relies on the observation in Fact 1. Particularly, given public $(\mathbf{B}, \mathbf{v})$, the prover proves knowledge of $(\mathbf{s}, \mathbf{e})$ satisfying the following relation

$$R'_{\mathrm{rnd}} = \left\{ ((\mathbf{B}, \mathbf{v}); (\mathbf{s}, \mathbf{e})) \ : \ \mathbf{s} \in \mathcal{R}_{q,d}^m \wedge \mathbf{e} = \mathbf{B}\mathbf{s} - \frac{q}{p}\mathbf{v} \bmod q \wedge \overrightarrow{\mathbf{e}} \in [q/p]^{Vd} \right\}, \tag{30}$$

which is equivalent to proving (29). To prove this relation, we make use of LANES^+ such that the knowledge of $\mathbf{s}$ is proven efficiently via RPoK while having

Algorithm 3 Proof of Correct Rounding

procedure $\mathsf{R.Gen}(1^\lambda)$
  return $\mathsf{pp} \leftarrow \mathsf{LANES}^+.\mathsf{Gen}(1^\lambda)$
end procedure

procedure $\mathsf{R.Prove}(\mathsf{pp}, (\mathbf{B}, \mathbf{v}), \mathbf{s}; \rho)$ ▷ ρ is an optional argument
  $\mathbf{e} = \mathbf{B}\mathbf{s} - \frac{q}{p} \cdot \mathbf{v}$
  Set (β, r) s.t. $q/p = \beta^r$
  Compute $\overrightarrow{b} \in \mathbb{Z}^{dVr}$ as the base-β digits of the coefficients in $\mathbf{e}$
  $P(\overrightarrow{\mathbf{e}}, \overrightarrow{b}) = \bigcirc_{i \in [\beta]}(\overrightarrow{b} - \overrightarrow{i})$ for $\overrightarrow{i} := (i, \ldots, i)$, where $\bigcirc$ denotes coordinate-wise multiplication over a set of elements
  $\mathsf{mp} := \{P\}$
  $\mathbf{G} = \mathbf{I}_{Vd} \otimes \mathbf{g}$ with $\mathbf{g} = (1, \beta, \ldots, \beta^{r-1})$ ▷ $\overrightarrow{\mathbf{e}} = \mathbf{G}\overrightarrow{b}$
  $\mathsf{ulp} = \left(\left(\mathbf{B}, -\mathbf{I}_{Vd}, \frac{q}{p}\mathbf{v}\right), (\mathbf{I}_{Vd}, \mathbf{G})\right)$
  $(t; t') \leftarrow \mathsf{LANES}^+.\mathsf{Com}_{\mathsf{pp}}(\mathbf{e}, \mathbf{s}, \overrightarrow{b})$
  $\pi \leftarrow \mathsf{LANES}^+.\mathsf{Prove}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), (t; t'); \rho)$
  return (t, π)
end procedure

procedure $\mathsf{R.Ver}(\mathsf{pp}, (\mathbf{B}, \mathbf{v}), (t_L, \pi); \rho)$ ▷ ρ is an optional argument
  Set mp and ulp as in R.Prove
  return $\mathsf{LANES}^+.\mathsf{Ver}_{\mathsf{pp}}((\mathsf{mp}, \mathsf{ulp}), t, \pi; \rho)$
end procedure

small coefficients for $\mathbf{e}$ is proven via LANES. Note that we do not necessarily need to prove that $\mathbf{s}$ is short and hence an RPoK is an ideal solution for that part. However, LANES^+ already proves knowledge of an f such that $f \cdot \mathbf{s}$ is short (which is not made explicit in the above relation). Now, we set $q/p = \beta^r$ and run the commitment step of LANES^+ with input $(\mathbf{e}, \mathbf{s}, \overrightarrow{b})$, where $\overrightarrow{b}$ denotes the base-β representation of the coefficient vector of $\mathbf{e}$. We can then prove in LANES that the coordinates of $\overrightarrow{b}$ are in $[\beta]$ using a multiplicative relation of the form $b_i(b_i - 1) \cdots (b_i - (\beta - 1)) = 0$ and also prove that they re-construct the coefficients of $\mathbf{e}$ via a linear relation such that the coefficients remain in the desired range. As a result, we prove (30), and hence (29). The full rounding protocol is presented in Alg. 3.

In certain cases (as our VRF application), we may not be able to set $q/p = \beta^r$ for $2 \leq \beta < q$, e.g., since q/p needs to be prime. In such cases, we can set $\beta^r \geq q$, which raises the issue that proof of being in the range $[\beta^r]$ is not equivalent to that of being in $[q/p]$. However, we can get around it by proving that certain digits in the decomposition satisfy a lower-order multiplicative relation of the form $P_a(X) = X \cdot (X - 1) \cdots (X - a) = 0$ for $a \leq \beta - 1$ so that reconstructed integer coefficients of $\mathbf{e}$ are really in $[q/p]$, not $[\beta^r]$. This is possible in LANES as long as the digits satisfying the $P_a(X)$ for the same a are packed within the same ring element of $\mathcal{R}_{\hat{q},\hat{d}}$. As discussed in Sec. 2.4, the communication size of LANES only depends on the maximal degree $\alpha = \beta$.

It is easy to see that the size of a proof output $\sigma = (t, \pi) = (t_L, (\pi_L, c, \mathbf{z}, \mathbf{f}))$ for Alg. 3 can be approximated by (ignoring the very small size of c)

$$|\sigma| \approx \underbrace{|t_L| + |\pi_L|}_{\text{size of LANES}} + \underbrace{|\mathbf{z}| + |\mathbf{f}|}_{\text{size of RPoK}} \ . \tag{31}$$

The advantage of our proof comes from (i) minimizing the entropy of the secret witness of LANES, and (ii) exploiting the efficient lattice-based RPoK for the high-entropy secret witness part. Particularly, the dimension over $\mathbb{Z}$ of the secret witness $\overrightarrow{s}$ in LANES is equal to $2Vd + Vdr = Vd(2 + r)$. In the case of a single module LWR sample, we have $V = 1$. We can also reasonably assume that $d \leq \hat{d}$, where $\hat{d} = 128$ in LANES is the default choice. Let us take $d = 32$ as in the concrete parameters of our VRF proposal. Finally, if we take $q/p = 2^4$ and $\beta = 4$ as an example, then we end up with $r = 2$. Hence, $\dim(\overrightarrow{s}) = Vd(2+r) = 128$. On the other hand, if we directly apply the LANES framework to prove knowledge of a single module LWR sample (i.e., $(\mathbf{s}, e)$ such that $\frac{q}{p}v = \langle \mathbf{b}, \mathbf{s} \rangle + e$), we would have the same cost for decomposition of the error e plus the much bigger dimension of $\overrightarrow{\mathbf{s}}$ compared to $\dim(\overrightarrow{e}) = d$. In practice, we would likely need $\dim(\overrightarrow{\mathbf{s}}) \geq 1024$, hence the total dimension of the secret witness in LANES would be 1088 using the same $(V, d, r) = (1, 32, 2)$, which pushes LANES to its less efficient realm where multiple proof responses need to be sent.

Theorem 4. *Assume that* LANES$^+$ *is correct, simulatable and knowledge sound as in Theorem 3, and uses a prime modulus* $\hat{q}$ *for* LANES *and another modulus* q *for* RPoK *with* $p \mid q$. *Further assume that any non-zero difference of challenges in* $\mathcal{C}$ *is invertible in* $\mathcal{R}_{q,d}$ *and that the assumptions in Remark 3 hold. Then, the protocol in Alg. 3 is correct, simulatable and sound w.r.t. the relation in* (29).

Proof (Theorem 4). Correctness and simulatability properties follow from correctness and simulatability of LANES$^+$.

For the knowledge soundness, running the extractor $\mathcal{E}_L$ of LANES$^+$ as in the proof of Thm. 3, we obtain $(\mathbf{e}^*, \mathbf{s}^*, \overrightarrow{b^*})$ such that

$$\frac{q}{p} \cdot \mathbf{v} = \mathbf{B}\mathbf{s}^* - \mathbf{e}^* \text{ over } \mathcal{R}_{q,d}, \tag{32}$$

$$\overrightarrow{\mathbf{e}^*} = \mathbf{G}\overrightarrow{b^*} \bmod \hat{q}, \text{ and} \tag{33}$$

$$\bigcirc_{i \in [\beta]}(\overrightarrow{b} - \overrightarrow{i}) = 0 \bmod \hat{q} \text{ for } \overrightarrow{i} := (i, \ldots, i). \tag{34}$$

Since $\hat{q}$ is prime by assumption, (34) implies that $\overrightarrow{b^*} \in [\beta]^{Vdr}$. Then, by the structure of $\mathbf{G}$, (33) gives that $\overrightarrow{\mathbf{e}^*} \in [q/p]^{Vd}$. Since $\mathbf{v} \in \mathcal{R}_{p,d}^V$, we conclude that $\mathbf{v} = \lfloor \mathbf{B}\mathbf{s}^* \rceil_p$ by Fact 1. □

6 LaV: Our Efficient Long-Term Lattice-Based VRF

In this section, we first describe our concrete instantiations of the PRF and the NIZK from lattices to realize the general VRF framework from Sec. 2.3. Then, we optimize over this proposal and describe our final VRF scheme, LaV.

6.1 Instantiation of the PRF

We describe our MLWR-based PRF below that is parametrized by η with $q > \eta \geq 1$. The PRF is then defined as

$$\mathsf{PRF}_k(\mathsf{m}) = \lfloor \mathbf{Ar} \rceil_p, \text{ where } \mathbf{A} \leftarrow \mathcal{G}(\mathsf{m}) \text{ and } k = \mathbf{r}, \tag{35}$$

for a short vector $\mathbf{r} \in \mathcal{K} := \mathbb{S}^{\ell}_{\eta,d}$ and a random oracle $\mathcal{G} : \{0,1\}^* \to \mathcal{R}^{n\times\ell}_{q,d}$. The output space is therefore $\mathcal{T} = \mathcal{R}^n_{q,d}$. We also define an extended key space of our PRF as in Sec. 2.2. In particular, for some parameter $\eta_1 \geq \eta$, the extended key space of the PRF is $\mathbb{S}^{\ell}_{\eta_1,d}$, which may be larger than the key space $\mathbb{S}^{\ell}_{\eta,d}$ for honest PRF executions. This property is useful for efficient lattice-based zero-knowledge proofs, and we want to make sure that the PRF is key-binding for the extended key space (which includes the key space).

Lemma 2. *The PRF defined above is computationally key-binding w.r.t. the extended key space $\mathbb{S}^{\ell}_{\eta_1,d}$ (see Sec. 2.2) if $\mathsf{MSIS}^{\infty}_{n,d,n+\ell,q,\beta_{\mathrm{SIS}}}$ is hard for $\beta_{\mathrm{SIS}} = \max\{2\eta_1, 2q/p\}$. It also satisfies computational κ-pseudorandomness (see Sec. 2.2) if $\mathsf{MLWR}_{\ell,d,n\kappa,q,p,\eta}$ is hard and p divides q.*

Proof. **Key-binding.** Let $\mathbf{r}, \mathbf{r}' \in \mathbb{S}^{\ell}_{\eta_1,d}$ be valid keys for a PRF output $\mathbf{v}$ with (i) $\mathbf{r} \neq \mathbf{r}'$ over $\mathcal{R}_{q,d}$, and (ii) $\|\mathbf{r}\|_\infty, \|\mathbf{r}'\|_\infty \leq \eta_1$. We want to show that such two keys lead to an MSIS solution. We have $\mathbf{v} = \lfloor \mathbf{Ar} \rceil_p = \lfloor \mathbf{Ar}' \rceil_p \pmod p$. Defining $\mathbf{e} := \mathbf{Ar} - \frac{q}{p} \cdot \mathbf{v} \bmod q$ and $\mathbf{e}' := \mathbf{Ar}' - \frac{q}{p} \cdot \mathbf{v} \bmod q$, we have the coefficients of $\mathbf{e}, \mathbf{e}'$ in $[q/p]$. Then, consider the following

$$\frac{q}{p}\mathbf{v} = \mathbf{Ar} - \mathbf{e} = \mathbf{Ar}' - \mathbf{e}' \pmod q, \tag{36}$$

$$\iff \quad \big(\mathbf{I}_n \ \mathbf{A}\big) \cdot \underbrace{\begin{pmatrix} \mathbf{e}' - \mathbf{e} \\ \mathbf{r} - \mathbf{r}' \end{pmatrix}}_{=:\mathbf{s}} = \mathbf{0} \pmod q. \tag{37}$$

Since $\mathbf{r} \neq \mathbf{r}'$ over $\mathcal{R}_{q,d}$, $\mathbf{s} \neq \mathbf{0}$ yields a solution to $\mathsf{MSIS}^{\infty}_{n,d,n+\ell,q,\beta_{\mathrm{SIS}}}$ for $\beta_{\mathrm{SIS}} = \max\{2\eta_1, 2q/p\}$.

κ-pseudorandomness. It is easy to see that each PRF output is an instance of $\mathsf{MLWR}_{\ell,d,n,q,p,\eta}$. So, κ PRF outputs will be an instance of $\mathsf{MLWR}_{\ell,d,n\kappa,q,p,\eta}$. Hence, the collection of such κ outputs will be indistinguishable from a uniformly random element of $\mathcal{R}^{n\kappa}_{p,d}$ if $\mathsf{MLWR}_{\ell,d,n\kappa,q,p,\eta}$ is hard and p divides q. □

6.2 Instantiation of the NIZK

We first define the set of relaxation factors as $F := \{c - c' \ : \ c, c' \in \mathcal{C} \wedge c \neq c'\}$ for $\mathcal{C}$ defined in (13). For the invertibility of relaxation factors, we rely on our results from Sec. 3 and set the parameters accordingly. We define $\zeta := 2w\gamma$ with $w = \delta\tilde{w}$. Then, we say that $(f, \mathbf{r})$ is a valid opening of a PRF output $\mathbf{v}$ if, for some parameter $\bar{\eta} < q$,

- $\|\mathbf{r}\|_\infty \leq \bar{\eta}$, $\|f\|_1 \leq \zeta$ with $f \in F$, and

– $\mathbf{v} = \lfloor \mathbf{A}(\mathbf{r}/f) \rfloor_p$, where division is done mod q.

We require the NIZK to prove knowledge of such a valid opening and also set $\eta_1 = \zeta\bar{\eta}$ so that $f' \cdot \mathbf{r}$ falls in the extended key space for any $f' \in F$. Now, let $\mathbf{A} \leftarrow \mathcal{G}(0)$ and $\mathbf{B} \leftarrow \mathcal{G}(\mathsf{m})$ be two matrices output by a random oracle $\mathcal{G}$. Denote $\mathsf{pk} = \mathbf{t}$ as the public key and $v = \mathbf{v}$ as the VRF value. Recall that we are interested in proving (6), which corresponds to proving the following relation for our concrete PRF instantiation

$$R_{\text{lbvrf}} = \left\{ ((\mathbf{A}, \mathbf{B}, \mathbf{t}, \mathbf{v}), (f, \mathbf{r})) \ : \ \begin{array}{c} \mathbf{t} = \lfloor \mathbf{A}(\mathbf{r}/f) \rfloor_p \wedge \mathbf{v} = \lfloor \mathbf{B}(\mathbf{r}/f) \rfloor_p \\ \wedge \, \|f\|_1 \leq \zeta \wedge \|\mathbf{r}\|_\infty \leq \bar{\eta} \end{array} \right\}. \quad (38)$$

The above itself is equivalent to proving the following $\begin{pmatrix} \mathbf{t} \\ \mathbf{v} \end{pmatrix} = \left\lfloor \begin{pmatrix} \mathbf{A} \\ \mathbf{B} \end{pmatrix} (\mathbf{r}/f) \right\rfloor_p$, which can be easily done using our rounding proof from Sec. 5. So, the NIZK for the above rounding relation together with the PRF from Sec. 6.1 is enough to instantiate the generic VRF proposal from Sec. 2.3. However, the scheme in this case is sub-optimal and, in the next section, we introduce a more efficient protocol that leads to our final long-term VRF proposal, LaV.

We remark that our MLWR-based PRF satisfies an approximate variant of additive key-homomorphism (as also observed in [BLMR13]) which suffices for the VRF uniqueness argument in the proof of Theorem 1 to go through, exploiting the fact that the key-binding property of our PRF also holds up to some approximation error in the PRF output. In more detail, observe that a PRF output $\mathbf{v}$ with a relaxed opening $(f, \mathbf{r})$ satisfies $\frac{q}{p} \cdot \mathbf{v} = \mathbf{A}\mathbf{r}/f - \mathbf{e} \bmod q$ where $\mathbf{e}$ is the rounding error. In this case, for small scaling factor α, we have $\alpha \cdot \frac{q}{p} \cdot \mathbf{v} = \mathbf{A}\alpha\mathbf{r}/f - \alpha\mathbf{e}$ which is approximately a PRF evaluation under $\alpha\mathbf{r}$ up to small error $\alpha\mathbf{e}$. The binding-based argument used in the proof of Theorem 1 still holds for our PRF in the presence of such small errors using an MSIS-based argument with respect to the matrix $[\, \mathbf{I}_n \,\|\, \mathbf{A} \,]$ as discussed in Sec. 6.4.

6.3 Final Unrolled VRF Scheme

We employ several optimizations over the general VRF framework instantiation. First, one can observe that a user is bound to a particular opening $(f, \mathbf{r})$ by the opening proof of the public key $\mathsf{pk} = \mathbf{t}$ (see Remark 1). Therefore, the VRF value $\mathbf{v}$ need not be a full-sized PRF output and we shrink it to a single $\mathcal{R}_{q,d}$ element. That is, we set $v = \lfloor \langle \mathbf{b}, \mathbf{s} \rangle \rfloor_p$ for a user secret key $\mathsf{sk} = \mathbf{s}$ and $\mathbf{b} \leftarrow \mathcal{G}(\mathsf{m})$.

The second optimization arises from the fact that we do not need to prove the well-formedness of the public key *exactly*, and can just bind the user to a *short* secret key $\mathsf{sk}' = (\mathbf{s}', \mathbf{e}')$ such that $\bar{c} \cdot \frac{q}{p}\mathbf{t} = \mathbf{A}\mathbf{s}' - \mathbf{e}'$ for a relaxation factor $\bar{c}$ using a RPoK. From an MSIS-based binding argument, it is computationally hard to find another triple $(\bar{c}_1, \mathbf{s}'_1, \mathbf{e}'_1)$ such that $\bar{c}_1 \cdot \frac{q}{p}\mathbf{t} = \mathbf{A}\mathbf{s}'_1 - \mathbf{e}'_1$ with $\mathbf{s}'_1/\bar{c}_1 \neq \mathbf{s}'/\bar{c}$. Hence, proving that $v = \lfloor \langle \mathbf{b}, \mathbf{s}'/\bar{c} \rangle \rfloor_p$ is sufficient to ensure uniqueness. This is further discussed in Sec. 6.4.

Lastly, we make use of the Bai-Galbraith compression technique [BG14] at Step 26 of LaV.Eval. In Alg. 4, we describe the full LaV VRF scheme, where the

Algorithm 4 LaV : Our long-term lattice-based VRF construction

1: **procedure** LaV.ParamGen(1^λ)
2: $\mathsf{pp}' \leftarrow \mathsf{R.Gen}(1^\lambda)$
3: Pick random $\mathcal{G} : \{0,1\}^* \to \mathcal{R}_{q,d}^{\ell}$
4: $\mathbf{A} \xleftarrow{\$} \mathcal{R}_{q,d}^{n \times \ell}$
5: **return** $\mathsf{pp} = (\mathsf{pp}', \mathbf{A}, \mathcal{G})$
6: **end procedure**

7: **procedure** LaV.KeyGen(pp)
8: $\mathbf{s} \xleftarrow{\$} \mathbb{S}_{\mathcal{B},d}^{\ell}$
9: $\mathbf{t} = \lfloor \mathbf{As} \rceil_p$
10: **return** $(\mathsf{pk}, \mathsf{sk}) = (\mathbf{t}, \mathbf{s})$
11: **end procedure**

12: **procedure** $\mathsf{LaV.Eval}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{sk}, \mathsf{m})$
13: $\mathbf{b} \leftarrow \mathcal{G}(\mathsf{m})$ and let $\mathbf{t} = \mathsf{pk}$
14: $v = \lfloor \langle \mathbf{b}, \mathbf{s} \rangle \rceil_p$ and $e' = \langle \mathbf{b}, \mathbf{s} \rangle - \frac{q}{p} v$
15: Sample $\mathbf{y}$ for Step 15 of Alg. 2
16: $\mathbf{w}_2 = [\![\mathbf{Ay}]\!]_K$ for $2^K \approx w\gamma \cdot q/p \cdot nd$
17: $(t, \pi) \leftarrow \mathsf{R.Prove}(\mathsf{pp}', (\mathbf{b}, v), \mathbf{s}; \mathbf{w}_2)$
18: Parse $\pi = (\pi_L, (c, \mathbf{z}, \mathbf{f}))$
19: $\hat{\mathbf{w}}_2 = \mathbf{Az} - c \cdot \frac{q}{p} \mathbf{t} \bmod 2^K$
20: **if** $\|\hat{\mathbf{w}}_2\|_\infty > 2^{K-1} - w\gamma\frac{q}{p}$, **then** Restart
21: **return** VRF value v and proof $\sigma = (t, \pi)$
22: **end procedure**

23: **procedure** $\mathsf{LaV.Verify}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{m}, v, \sigma)$
24: Parse $\sigma = (t, (\pi_L, (c, \mathbf{z}, \mathbf{f})))$
25: $\mathbf{b} \leftarrow \mathcal{G}(\mathsf{m})$ and let $\mathbf{t} = \mathsf{pk}$
26: $\mathbf{w}_2' = [\![\mathbf{Az} - c \cdot \frac{q}{p} \mathbf{t}]\!]_K$
27: **return** $\mathsf{R.Ver}(\mathsf{pp}', (\mathbf{b}, v), \sigma; \mathbf{w}_2')$
28: **end procedure**

challenge space $\mathcal{C}$ is instantiated as in (13) and $[\![\mathbf{x}]\!]_K$ denotes dropping $K \geq 1$ least-significant bits of each coefficient in $\mathbf{x}$. For simplicity, we sample the matrix $\mathbf{A}$ at random in Step 4, instead of generating it via a random oracle.

Remark 4. The NIZK proof in LaV.Eval can also be seen as executing LANES^+ with $\mathsf{ulp} = ((\mathbf{A}', \mathbf{b}', \mathbf{t}'), (\mathbf{I}_d, \mathbf{G}))$ where $\mathbf{G}$ is the integer reconstruction matrix for e' as in R.Prove, $\mathbf{A}' = \begin{pmatrix} \mathbf{A} & -\mathbf{I}_n \\ \mathbf{b}^\top & \mathbf{0}^\top \end{pmatrix}$, $\mathbf{b}' = \begin{pmatrix} \mathbf{0} \\ -1 \end{pmatrix}$, and $\mathbf{t}' = \begin{pmatrix} \frac{q}{p} \cdot \mathbf{t} \\ \frac{q}{p} \cdot v \end{pmatrix}$. The secret witness for LANES^+ (i.e., input of $\mathsf{LANES}^+.\mathsf{Com}$) is then $\left(\begin{pmatrix} \mathbf{s} \\ \mathbf{e} \end{pmatrix}, e', \overrightarrow{b} \right)$, where $\overrightarrow{b}$ is the base-β decomposition of the coefficients of e'.

The total average number of restarts in LaV is approximately equal to $\mu(\phi) \cdot \mu(\phi_m) \cdot \exp(1) \cdot M_L$ for $2^K \approx w\gamma \cdot q/p \cdot nd$, where M_L denotes the average number of repetitions in LANES and $\mu(\phi) = e^{12/\phi + 1/(2\phi^2)}$ as defined in Alg. 5. Recall that $M_L = 1$ if we use the results of [KLSS23] in LANES since that approach does not require rejection sampling. We can perform a single rejection sampling on the concatenated vector $(\mathbf{z}, \mathbf{f})$ if $\mathcal{B} \approx q/p$. In this case, we would have $\phi_m = \phi$ and the total average number of repetitions $\approx \mu(\phi) \cdot \exp(1) \cdot M_L$.

We list in Assumption 1, the assumptions needed to establish a secure VRF from Alg. 4. We refer to each requirement in Assumption 1 as 'Sub-Assumption i'. We discuss in Sec. 6.4 that our optimizations do not harm the security of LaV.

Assumption 1 *We assume the following to establish security of* LaV *with (at most)* κ *evaluations per key pair.*

1. *Any non-zero difference of challenges in* $\mathcal{C}$ *is invertible in* $\mathcal{R}_{q,d}$.
2. $\hat{q} > \max\{24\phi_m\eta_m, w\gamma\beta^r\}$ *and* $q > 12\phi_m\eta_m$ *(these assumptions ensure that those in Remark 3 are satisfied).*

3. $q > \beta_{\mathrm{SIS}}$ *and* $\mathsf{MSIS}^{\infty}_{n,d,n+\ell,q,\beta_{\mathrm{SIS}}}$ *for* $\beta_{\mathrm{SIS}} = 4w\gamma \cdot \max\{12\phi\eta, 2^K\}$ *is hard.*
4. $\mathsf{MLWR}_{\ell,d,n+\kappa,q,p,\mathcal{B}}$ *is hard.*
5. *Internal parameters for* LANES *are set properly.*

6.4 Security Discussion of LaV

As we have already formally proved the security of the generic VRF construction in Sec. 2.3 and the required properties of the concrete lattice-based instantiation, we now discuss the impact of our optimizations.

Assume that a user creates at most κ VRF outputs per key pair. Since the underlying NIZK used in LaV.Eval is zero-knowledge (or simulatable), for pseudorandomness, it is sufficient to consider the information leaked by the public key pk and the VRF values v_i's for $1 \leq i \leq \kappa$. The difference of Alg. 4 from the generic approach is that each VRF output leaks a single MLWR sample rather than n samples. As a result, in LaV, $n + \kappa$ MLWR samples are produced after κ VRF outputs. Hence, it is sufficient to assume Sub-Assumption 4.

For the uniqueness property of LaV, the intuition is that we do not need to prove the well-formedness of the public key *exactly*, and can just bind the user to a *short* secret key $\mathsf{sk}' = (\mathbf{s}', \mathbf{e}')$ such that $\bar{c} \cdot \frac{q}{p}\mathbf{t} = \mathbf{A}\mathbf{s}' - \mathbf{e}' \bmod q$ for a relaxation factor $\bar{c}$ using a RPoK. Let us discuss the uniqueness of LaV in more detail.

Uniqueness of LaV. Let $(v, (t, (\pi_L, (c, \mathbf{z}, \mathbf{f}))))$ and $(v', (t', (\pi'_L, (c', \mathbf{z}', \mathbf{f}'))))$ be two valid VRF outputs for the same message m and public key $\mathsf{pk} = \mathbf{t}$. We want to show that $v = v'$. Similar to [EKS+21], we use a double rewinding argument.
Rewind 1: We rewind w.r.t. to the first output and obtain another accepting output $(v, (t, (\pi_L^{(0)}, (c^{(0)}, \mathbf{z}^{(0)}, \mathbf{f}^{(0)}))))$. Define $\bar{\mathbf{z}} := \mathbf{z} - \mathbf{z}^{(0)}$, and $\bar{c} := c - c^{(0)}$. Then, by Step 26 of LaV.Verify (note that $\mathbf{w}'_2$ goes as an input to the random oracle $\mathcal{H}$ and thus must not change between rewindings), we get

$$[\![\mathbf{A}\mathbf{z} - c \cdot \frac{q}{p}\mathbf{t}]\!]_K = [\![\mathbf{A}\mathbf{z}^{(0)} - c^{(0)} \cdot \frac{q}{p}\mathbf{t}]\!]_K \tag{39}$$

$$\iff \quad \bar{c} \cdot \frac{q}{p}\mathbf{t} = \mathbf{A}\bar{\mathbf{z}} - \bar{\mathbf{e}} =: \mathbf{A}' \cdot \bar{\mathbf{s}} \pmod q, \tag{40}$$

for some $\bar{\mathbf{e}}$ with $\|\bar{\mathbf{e}}\|_\infty \leq 2^K$, $\bar{\mathbf{s}} := \begin{pmatrix} \bar{\mathbf{z}} \\ \bar{\mathbf{e}} \end{pmatrix}$ and $\mathbf{A}' := [\,\mathbf{A} \,\|\, -\mathbf{I}_n\,]$. Note that $\|\bar{\mathbf{s}}\|_\infty \leq \max\{12\phi\eta, 2^K\}$.
Rewind 2: We do a similar rewinding w.r.t. to the second output and obtain the following

$$\bar{c}' \cdot \frac{q}{p}\mathbf{t} = \mathbf{A}' \cdot \bar{\mathbf{s}}' \pmod q. \tag{41}$$

Again, we have $\|\bar{\mathbf{s}}'\|_\infty \leq \max\{12\phi\eta, 2^K\}$. Multiplying (40) by $\bar{c}'$ and (41) by $\bar{c}$ to equalize the left-hand sides of both expressions, and then subtracting the results, we get

$$\mathbf{A}' \cdot (\bar{c}'\bar{\mathbf{s}} - \bar{c}\bar{\mathbf{s}}') = \mathbf{0} \pmod q. \tag{42}$$

Observe that $\|\bar{c}'\bar{\mathbf{s}} - \bar{c}\bar{\mathbf{s}}'\|_\infty \leq 4w\gamma \cdot \max\{12\phi\eta, 2^K\} =: \beta_{\text{SIS}}$. By the hardness of MSIS^∞ in Sub-Assumption 3, we conclude that

$$\bar{c}'\bar{\mathbf{s}} = \bar{c}\bar{\mathbf{s}}'. \tag{43}$$

Note that q must be strictly bigger than $\beta_{\text{SIS}} > \|\bar{c}'\bar{\mathbf{s}}\|_\infty, \|\bar{c}\bar{\mathbf{s}}'\|_\infty$ to ensure MSIS^∞ hardness. Hence, the above equality holds without mod q.

Now, by the soundness of $\mathsf{R.Prove}$, we have that $v = \lfloor \langle \mathbf{b}, \mathbf{s}^* \rangle \rfloor_p$ and $v' = \lfloor \langle \mathbf{b}, \mathbf{s}'^* \rangle \rfloor_p$, where $\mathbf{s}^* := \bar{\mathbf{z}}/\bar{c} \bmod q$ and $\mathbf{s}'^* := \bar{\mathbf{z}}'/\bar{c}' \bmod q$ as shown at the end of the soundness proof of Thm. 3. Since $\bar{c}' \cdot \bar{\mathbf{z}} = \bar{c} \cdot \bar{\mathbf{z}}' \bmod q$ by (43), we can use the fact that $\bar{c}, \bar{c}'$ are invertible mod q to conclude that $\bar{\mathbf{z}}/\bar{c} = \bar{\mathbf{z}}'/\bar{c}' \bmod q$ and hence $\mathbf{s}^* = \mathbf{s}'^*$ and $v = v'$.

6.5 Parameter Setting

As noted as a footnote in Sec. 2.4, it is easy to shift the range for the NIZK proof so that it is centred at zero. Hence, we can apply it (for free in communication) so that the error e' has coefficients in $\left[-\frac{q}{2p}, \frac{q}{2p}\right) \cap \mathbb{Z}$ to save a factor 2 when bounding $\|e'\|_\infty$. In MSIS and $\mathsf{MLWE}/\mathsf{MLWR}$ problems, it is also often the case that the solution coefficients are centred at zero. Hence, we assume the same shifting of the range when estimating their hardness.

Setting parameters external to LANES. One of the most critical assumptions that restrict our choice of parameters is Sub-Assumption 1. This is because we need q to be composite so that $p \mid q$ and we can use Fact 1. If we have $q = q_0 \cdot p$ for prime values q_0 and p, then $\mathcal{R}_{q,d} \cong \mathcal{R}_{q_0,d} \times \mathcal{R}_{p,d}$ does not split further w.r.t. the integer modulus q. As a result, Sub-Assumption 1 is satisfied if and only if challenge differences are invertible in $\mathcal{R}_{q_0,d}$ and $\mathcal{R}_{p,d}$. That is, we need to guarantee the results from Sec. 3 in both $\mathcal{R}_{q_0,d}$ and $\mathcal{R}_{p,d}$. Since we want to minimize q_0 to reduce the entropy of the input message, e', for LANES, this task itself reduces to focusing on $\mathcal{R}_{q_0,d}$. As a result, we looked at the smallest d we can set while satisfying Sub-Assumption 1 and found that $d = 32$ is the best choice. Otherwise, we need $q_0 > 2^{12}$, which is quite large. Hence, we choose $d = 32$ first.

Having fixed $d = 32$, the smallest $q_0 = q/p$ while satisfying Sub-Assumption 1 is $q_0 = 61$ from the results of Sec. 3. In this case, the assumption holds with probability at least $1 - 2^{-91.5}$. We also set $(w, \gamma) = (32, 16)$ from the results in Table 2, where $w = \delta\tilde{w} = 16 \cdot 2$ is the full weight of a challenge in (13).

Now, since $q_0 = q/p$ is prime, we cannot exactly have $q_0 = q/p = \beta^r$ for $2 \leq \beta < q_0$. Instead, we choose $(\beta, r) = (3, 4)$ such that $\beta^r \geq q/p$. As discussed before in Sec. 5, this choice is still fine. In particular, our choice of parameters for LANES have $l = 32 = d$ as the optimal option to minimize LANES communication size. Now, let $\overrightarrow{e'} = (e_0, \ldots, e_{d-1})$ and $e_i = (e_{i,0}, \ldots, e_{i,r-1})$ in base β. Then, since $l = d$, we can store in each ring element $\hat{m}_i \in \mathcal{R}_{\hat{q},\hat{d}}$ exactly $l = d$ values using the CRT slots. Particularly, we can set $\hat{m}_i = \mathsf{CRT}^{-1}(e_{0,i}, \ldots, e_{d-1,i})$, storing the i-th digit of the integers in the same ring element. Now, instead of proving that $\hat{m}_i \cdot (\hat{m}_i - 1) \cdots (\hat{m}_i - (\beta - 1)) = 0$ for all $i \in [r]$ in the multiplicative proof of LANES, we can instead prove that $\hat{m}_i \cdot (\hat{m}_i - 1) \cdots (\hat{m}_i - a) = 0$ for some $a \leq \beta - 1$

and a specific set of indices i to make sure that the integer reconstruction from the digits does not exceed q/p.

We also set $\phi = \phi_m = 12$ as a typical choice and $\mathcal{B} = 1$ to minimize the communication size. In terms of (η, η_m) (the ℓ_2-norm bounds in Alg. 2), they are computed as $\eta = w\gamma\mathcal{B}\sqrt{\ell d}$ and $\eta_m = w\gamma\lfloor q_0/2 \rfloor \sqrt{d}$ (recall that the coefficients of e' are centred at zero and hence $\|e'\|_\infty \leq \lfloor q_0/2 \rfloor$).

Finally, we look at the practical MSIS/MLWR requirements against known attacks to set the module ranks n and ℓ (for MSIS^∞ and MLWR, respectively) and the modulus $q = q_0 \cdot p$. When estimating the security of these problems against lattice attacks, we consider the "root Hermite factor (RHF)", a common metric used to measure the practical hardness of MSIS and MLWE/MLWR problems, and aim for RHF ≈ 1.0045 as in, e.g., [EZS+19, ALS20, ENS20, LNS20]. For MSIS^∞, we used "Asymmetric-MSIS" scripts of [ESZ22] and found that setting $n = 48$ and $q \approx 2^{37}$ (i.e., $p \approx 2^{31}$) leads to a RHF of 1.0045. Since challenge difference invertibility requirement is satisfied for a much smaller modulus $q_0 \ll p$, finding a suitable prime p is easy.

For MLWR with $(d, q, p, \mathcal{B}) \approx (32, 2^{37}, 2^{31}, 1)$, we set $\ell = 40$ to achieve a root Hermite Factor ≈ 1.0045 against lattice attacks, estimated using the LWE estimator [APS15] BKZ quantum sieve model for LWE with a ternary coordinate secret distribution. We also estimated using the LWE estimator the complexity of algebraic Gröbner Base (GB) attacks against MLWR with $\kappa + n$ samples over $\mathcal{R}_{q,d}$, assuming semi-regularity of the system, based on the model in [ACF+15]. The system of equations in the nd secret coordinates over $\mathbb{Z}_q$ includes $d(\kappa + n)$ equations of degree $q_0 = 61$ (the rounding error interval size) and also nd equations of degree $2\mathcal{B} + 1 = 3$ (the secret coordinate interval size). However, with our parameter set $(d, q, p, \mathcal{B}) \approx (32, 2^{37}, 2^{31}, 1)$ the estimated GB attack complexity always exceeded the lattice attack complexity, for any number of MLWR samples $\kappa \leq 2^{128}$, indicating that the LaV VRF with our parameter set is secure against known attacks with an essentially unbounded number of outputs.

Setting internal parameters for LANES. One of the advantages of our proposal is that we have the flexibility to minimize the dimension (and entropy) of the input message for LANES so as to push it towards its more efficient realm. In particular, from the above setting of $\beta = 3$, we get the maximal polynomial degree in mp as $\alpha = \beta = 3$. Furthermore, we can use the *partition-and-sample* technique in [ESZ22] (i.e., $\gamma = 1$ case of the results in Sec. 3) to have $\mathcal{R}_{\hat{q},\hat{d}}$ split into $l = 32$ factors with $\hat{d} = 128$ while also keeping the ℓ_1-norm of the challenge c_L used in LANES (see the fourth move of [ENS20, Fig. 3]) small. In this case, we can set $k = 1$ and the challenge differences will be invertible with overwhelming probability. Particularly, we set $\|c_L\|_1 \leq \hat{w} = 44$, which leads to a challenge space of size about 2^{152} for LANES. With the choice of $(d, l, r) = (32, 32, 4)$, we end up with $N = d(2 + r)/l = 6$ as the input message dimension over $\mathbb{Z}_{\hat{q}}^l$.

Then, using the "Hint-MLWE" approach of [KLSS23], we looked at the possible choices of $(\log \hat{q}, \hat{n}, \hat{\ell})$ for LANES with our small-dimensional input message and found that choosing $(\log \hat{q}, \hat{n}, \hat{\ell}) = (26, 6, 7)$ leads to a RHF ≈ 1.0045, which is similar to the choices in [ALS20, ENS20, LNS20]. Since we do not have any

additional condition (over those needed in LANES) on the shape of $\hat{q}$, it can be set as a suitable prime with $\hat{q} \equiv 2l + 1 \pmod{4l}$. Note also that both moduli q and $\hat{q}$ are sufficiently large to satisfy Sub-Assumption 2. We also assume that $D = 13$ for commitment compression in LANES ([ENS20, LNS20] use $D = 14$).

The above parameter setting for LANES leads to a total communication size of $|t_L| + |\pi_L| \approx 7.1$ KB (using (12) with $\mathfrak{s} \approx 2\sqrt{2}w\mathfrak{s}_0$ defined in Sec. 2.4) for the LANES part of LaV output.

Overall, the above parameter setting leads to 3.18 KB for RPoK, and the total proof size of LaV is $|\sigma| \approx 10.27$ KB. The VRF value v is 124 bytes and the public key size is about 5.81 KB. One could apply the public key compression technique in Dilithium [DLL+18] to reduce the public key size further (which may come at a cost in proof size). Since communication of a public key in (long-term) VRF is often a one-time task, we consider the proof cost as the major factor.

Acknowledgements. This research was supported in part by ARC Discovery Project grants DP180102199 and DP220101234.

References

ACF+15. Martin R. Albrecht, Carlos Cid, Jean-Charles Faugère, Robert Fitzpatrick, and Ludovic Perret. Algebraic algorithms for LWE problems. *ACM Commun. Comput. Algebra*, 49(2):62, 2015.

ALS20. Thomas Attema, Vadim Lyubashevsky, and Gregor Seiler. Practical product proofs for lattice commitments. In *CRYPTO (2)*, LNCS, pages 470–499. Springer, 2020.

APS15. Martin R. Albrecht, Rachel Player, and Sam Scott. On the concrete hardness of learning with errors. *J. Math. Cryptol.*, 9(3):169–203, 2015. Code available at `https://bitbucket.org/malb/lwe-estimator/src/master/`.

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 *IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society, 2014.

BDE+21. Maxime Buser, Rafael Dowsley, Muhammed F. Esgin, Shabnam Kasra Kermanshahi, Veronika Kuchta, Joseph K. Liu, Raphael Phan, and Zhenfei Zhang. Post-quantum verifiable random function from symmetric primitives in pos blockchain. *IACR Cryptol. ePrint Arch.*, page 302, 2021.

BDL+18. Carsten Baum, Ivan Damgård, Vadim Lyubashevsky, Sabine Oechsner, and Chris Peikert. More efficient commitments from structured lattice assumptions. In *SCN*, volume 11035 of *LNCS*, pages 368–385. Springer, 2018.

BG14. Shi Bai and Steven D. Galbraith. An improved compression technique for signatures based on learning with errors. In *CT-RSA*, volume 8366 of *LNCS*, pages 28–47. Springer, 2014.

Bit20. Nir Bitansky. Verifiable random functions from non-interactive witness-indistinguishable proofs. *J. Cryptol.*, 33(2):459–493, 2020.

BLMR13. Dan Boneh, Kevin Lewi, Hart William Montgomery, and Ananth Raghunathan. Key homomorphic prfs and their applications. In *CRYPTO (1)*, volume 8042 of *LNCS*, pages 410–428. Springer, 2013.

BLS19. Jonathan Bootle, Vadim Lyubashevsky, and Gregor Seiler. Algebraic techniques for short(er) exact lattice-based zero-knowledge proofs. In *CRYPTO (1)*, volume 11692 of *LNCS*, pages 176–202. Springer, 2019.

BN06. Mihir Bellare and Gregory Neven. Multi-signatures in the plain public-key model and a general forking lemma. In *ACM CCS*, pages 390–399. ACM, 2006.

BP14. Abhishek Banerjee and Chris Peikert. New and improved key-homomorphic pseudorandom functions. In *CRYPTO (1)*, volume 8616 of *LNCS*, pages 353–370. Springer, 2014.

BPR12. Abhishek Banerjee, Chris Peikert, and Alon Rosen. Pseudorandom functions and lattices. In *EUROCRYPT*, volume 7237 of *LNCS*, pages 719–737. Springer, 2012.

CGH09. Scott E. Coull, Matthew Green, and Susan Hohenberger. Controlling access to an oblivious database using stateful anonymous credentials. In *Public Key Cryptography*, volume 5443 of *LNCS*, pages 501–520. Springer, 2009.

CGL+17. Alessandro Chiesa, Matthew Green, Jingcheng Liu, Peihan Miao, Ian Miers, and Pratyush Mishra. Decentralized anonymous micropayments. In *EUROCRYPT (2)*, volume 10211 of *LNCS*, pages 609–642, 2017.

CHK+06. Jan Camenisch, Susan Hohenberger, Markulf Kohlweiss, Anna Lysyanskaya, and Mira Meyerovich. How to win the clonewars: efficient periodic n-times anonymous authentication. In *ACM CCS*, pages 201–210. ACM, 2006.

CL15. Jan Camenisch and Anja Lehmann. (un)linkable pseudonyms for governmental databases. In *ACM CCS*, pages 1467–1479. ACM, 2015.

CLOS02. Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *STOC*, pages 494–503. ACM, 2002.

CM19. Jing Chen and Silvio Micali. Algorand: A secure and efficient distributed ledger. *Theor. Comput. Sci.*, 777:155–183, 2019.

DLL+18. Léo Ducas, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. Crystals–Dilithium: Digital signatures from module lattices. In *CHES*, volume 2018-1, 2018.

dPLS18. Rafaël del Pino, Vadim Lyubashevsky, and Gregor Seiler. Lattice-based group signatures and zero-knowledge proofs of automorphism stability. In *ACM CCS*, pages 574–591. ACM, 2018.

EG14. Alex Escala and Jens Groth. Fine-tuning groth-sahai proofs. In *Public Key Cryptography (PKC)*, volume 8383 of *LNCS*, pages 630–649. Springer, 2014.

EKS+21. Muhammed F. Esgin, Veronika Kuchta, Amin Sakzad, Ron Steinfeld, Zhenfei Zhang, Shifeng Sun, and Shumo Chu. Practical post-quantum few-time verifiable random function with applications to algorand. In *Financial Cryptography and Data Security (2)*, volume 12675 of *LNCS*, pages 560–578. Springer, 2021. (Full version at `ia.cr/2020/1222`).

ENS20. Muhammed F. Esgin, Ngoc Khanh Nguyen, and Gregor Seiler. Practical exact proofs from lattices: New techniques to exploit fully-splitting rings. In *ASIACRYPT (2)*, volume 12492 of *LNCS*, pages 259–288. Springer, 2020. Full version at `ia.cr/2020/518`.

ESLL19. Muhammed F. Esgin, Ron Steinfeld, Joseph K. Liu, and Dongxi Liu. Lattice-based zero-knowledge proofs: New techniques for shorter and faster constructions and applications. In *CRYPTO (1)*, volume 11692 of *LNCS*, pages 115–146. Springer, 2019.

ESS+19. Muhammed F. Esgin, Ron Steinfeld, Amin Sakzad, Joseph K. Liu, and Dongxi Liu. Short lattice-based one-out-of-many proofs and applications to ring signatures. In *ACNS*, volume 11464 of *LNCS*, pages 67–88. Springer, 2019.

ESZ22. Muhammed F. Esgin, Ron Steinfeld, and Raymond K. Zhao. MatRiCT+: More efficient post-quantum private blockchain payments. In *IEEE Symposium on Security and Privacy (S&P)*, pages 1281–1298. IEEE, 2022. (Full version at `ia.cr/2021/545`).

EZS+19. Muhammed F. Esgin, Raymond K. Zhao, Ron Steinfeld, Joseph K. Liu, and Dongxi Liu. MatRiCT: Efficient, scalable and post-quantum blockchain confidential transactions protocol. In *ACM CCS*, pages 567–584. ACM, 2019.

FS07. Eiichiro Fujisaki and Koutarou Suzuki. Traceable ring signature. In *Public Key Cryptography*, volume 4450 of *LNCS*, pages 181–200. Springer, 2007.

GHKW17. Rishab Goyal, Susan Hohenberger, Venkata Koppula, and Brent Waters. A generic approach to constructing and proving verifiable random functions. In *TCC (2)*, volume 10678 of *LNCS*, pages 537–566. Springer, 2017.

GHM+17. Yossi Gilad, Rotem Hemo, Silvio Micali, Georgios Vlachos, and Nickolai Zeldovich. Algorand: Scaling byzantine agreements for cryptocurrencies. In *SOSP*, pages 51–68. ACM, 2017.

GM17. Matthew Green and Ian Miers. Bolt: Anonymous payment channels for decentralized currencies. In *ACM CCS*, pages 473–489. ACM, 2017.

GNP+15. Sharon Goldberg, Moni Naor, Dimitrios Papadopoulos, Leonid Reyzin, Sachin Vasant, and Asaf Ziv. NSEC5: provably preventing DNSSEC zone enumeration. In *NDSS*. The Internet Society, 2015.

HMPS14. Susan Hohenberger, Steven A. Myers, Rafael Pass, and Abhi Shelat. ANONIZE: A large-scale anonymous survey system. In *IEEE Symposium on Security and Privacy*, pages 375–389. IEEE Computer Society, 2014.

JKK14. Stanislaw Jarecki, Aggelos Kiayias, and Hugo Krawczyk. Round-optimal password-protected secret sharing and T-PAKE in the password-only model. In *ASIACRYPT (2)*, volume 8874 of *LNCS*, pages 233–253. Springer, 2014.

Kil90. Joe Kilian. *Uses of randomness in algorithms and protocols*. MIT Press, 1990.

KLSS23. Duhyeong Kim, Dongwon Lee, Jinyeong Seo, and Yongsoo Song. Toward practical lattice-based proof of knowledge from hint-mlwe. Cryptology ePrint Archive, Paper 2023/623, 2023. `https://eprint.iacr.org/2023/623`.

KRDO17. Aggelos Kiayias, Alexander Russell, Bernardo David, and Roman Oliynykov. Ouroboros: A provably secure proof-of-stake blockchain protocol. In *CRYPTO (1)*, volume 10401 of *LNCS*, pages 357–388. Springer, 2017.

LL. Sean Lawlor and Kevin Lewi. Deploying key transparency at WhatsApp. `https://engineering.fb.com/2023/04/13/security/whatsapp-key-transparency/`. Accessed: 2023-05-16.

LLNW17. Benoît Libert, San Ling, Khoa Nguyen, and Huaxiong Wang. Zero-knowledge arguments for lattice-based prfs and applications to e-cash. In *ASIACRYPT (3)*, volume 10626 of *LNCS*, pages 304–335. Springer, 2017.

LNP22. Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Maxime Plançon. Lattice-based zero-knowledge proofs and applications: Shorter, simpler, and more

general. In *CRYPTO (2)*, volume 13508 of *LNCS*, pages 71–101. Springer, 2022.

LNPS21. Vadim Lyubashevsky, Ngoc Khanh Nguyen, Maxime Plançon, and Gregor Seiler. Shorter lattice-based group signatures via "almost free" encryption and other optimizations. In *ASIACRYPT (4)*, volume 13093 of *LNCS*, pages 218–248. Springer, 2021.

LNS20. Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler. Practical lattice-based zero-knowledge proofs for integer relations. In *ACM CCS*, pages 1051–1070. ACM, 2020.

LNS21a. Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler. Shorter lattice-based zero-knowledge proofs via one-time commitments. In *Public Key Cryptography (1)*, volume 12710 of *LNCS*, pages 215–241. Springer, 2021.

LNS21b. Vadim Lyubashevsky, Ngoc Khanh Nguyen, and Gregor Seiler. SMILE: set membership from ideal lattices with applications to ring signatures and confidential transactions. In *CRYPTO (2)*, volume 12826 of *LNCS*, pages 611–640. Springer, 2021.

LS18. Vadim Lyubashevsky and Gregor Seiler. Short, invertible elements in partially splitting cyclotomic rings and applications to lattice-based zero-knowledge proofs. In *EUROCRYPT (1)*, volume 10820 of *LNCS*, pages 204–224. Springer, 2018.

Lyu09. Vadim Lyubashevsky. Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In *ASIACRYPT*, pages 598–616. Springer, 2009.

Lyu12. Vadim Lyubashevsky. Lattice signatures without trapdoors. In *EUROCRYPT*, pages 738–755. Springer, 2012. (Full version).

MRV99. Silvio Micali, Michael O. Rabin, and Salil P. Vadhan. Verifiable random functions. In *FOCS*, pages 120–130. IEEE Computer Society, 1999.

Ngu22. Ngoc Khanh Nguyen. Private communication, 2022.

PWH+17. Dimitrios Papadopoulos, Duane Wessels, Shumon Huque, Moni Naor, Jan Včelák, Leonid Reyzin, and Sharon Goldberg. Making NSEC5 practical for DNSSEC. Cryptology ePrint Archive, Report 2017/099, 2017. `https://eprint.iacr.org/2017/099`.

Ste93. Jacques Stern. A new identification scheme based on syndrome decoding. In *CRYPTO*, volume 773 of *LNCS*, pages 13–21. Springer, 1993.

Yam17. Shota Yamada. Asymptotically compact adaptively secure lattice ibes and verifiable random functions via generalized partitioning techniques. In *CRYPTO (3)*, volume 10403 of *LNCS*, pages 161–193. Springer, 2017.

YAZ+19. Rupeng Yang, Man Ho Au, Zhenfei Zhang, Qiuliang Xu, Zuoxia Yu, and William Whyte. Efficient lattice-based zero-knowledge arguments with standard soundness: Construction and applications. In *CRYPTO (1)*, volume 11692 of *LNCS*, pages 147–175. Springer, 2019.

A Further Preliminaries

A.1 Verifiable Random Function (VRF)

A Verifiable Random Function (VRF) comprises the following four polynomial time algorithms [MRV99].

$\mathsf{V.ParamGen}(1^\lambda)$**:** Given the security parameter λ, this algorithm generates public parameters pp.

$\mathsf{V.KeyGen}(\mathsf{pp})$**:** With the parameters pp, this algorithms generates the private key sk and the corresponding public key pk.

$\mathsf{V.Eval}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{sk}, \mathsf{m})$**:** Given the message m and the private key sk, this algorithm generates the VRF value $v \in \{0,1\}^{m(\lambda)}$ and a proof π.

$\mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk}, \mathsf{m}, v, \pi)$**:** This algorithm returns 1 or 0, indicating whether v can be verified with the remaining parameters.

We next define the properties a VRF should satisfy. We adopt the κ-pseudorandomness and (full) uniqueness properties from [EKS+21].

Provability: This property requires the following condition to hold for all valid messages m.

$$\Pr\left[\begin{array}{l}\mathsf{pp} \leftarrow \mathsf{V.ParamGen}(1^\lambda),\\ (\mathsf{sk},\mathsf{pk}) \leftarrow \mathsf{V.KeyGen}(\mathsf{pp}),\\ (v,\pi) \leftarrow \mathsf{V.Eval}_{\mathsf{pp}}(\mathsf{pk},\mathsf{sk},\mathsf{m})\end{array} : \mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk},\mathsf{m},\pi,v) = 1\right] = 1.$$

κ-Pseudorandomness: Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be a polynomial-time adversary playing the following experiment $\mathsf{Exp\text{-}PRand}$:

1. $\mathsf{pp} \leftarrow \mathsf{V.ParamGen}(1^\lambda)$
2. $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{V.KeyGen}(\mathsf{pp})$
3. $(\mathsf{m},\mathsf{st}) \leftarrow \mathcal{A}_1^{\mathcal{O}_{\mathrm{VEval}}(\cdot)}(\mathsf{pk})$
4. $(v_0,\pi_0) \leftarrow \mathsf{V.Eval}_{\mathsf{pp}}(\mathsf{pk},\mathsf{sk},\mathsf{m})$
5. $v_1 \xleftarrow{\$} \{0,1\}^{m(\lambda)}$
6. $b \xleftarrow{\$} \{0,1\}$
7. $b' \leftarrow \mathcal{A}_2^{\mathcal{O}_{\mathrm{VEval}}(\cdot)}(v_b,\mathsf{st})$

where $\mathcal{O}_{\mathrm{VEval}}(\cdot)$ is an oracle (that can be queried at most $\kappa - 1$ times by the adversary)[9] that on input a value m outputs the VRF value v and the corresponding proof of correctness $\pi(\mathsf{sk},\mathsf{m})$. A VRF scheme is said to satisfy κ-pseudorandomness if the following holds for any PPT adversary $\mathcal{A}$ that did not issue any queries to $\mathcal{O}_{\mathrm{VEval}}$ on the value m:

$$\Pr[b = b' \,|\, \mathcal{A} \text{ runs } \mathsf{Exp\text{-}PRand}] \leq \frac{1}{2} + \mathrm{negl}(\lambda).$$

[9] Note that together with the challenge query to $\mathsf{V.Eval}(\cdot)$ in the pseudorandomness experiment, a total of κ $\mathsf{V.Eval}(\cdot)$ queries can be made in total in the experiment.

(Full) Uniqueness: A VRF scheme satisfies (full) uniqueness if the following probability is negligible in λ for any adversary $\mathcal{A}$.

$$\Pr\left[\begin{array}{l}\mathsf{pp} \leftarrow \mathsf{V.ParamGen}(1^\lambda),\\ (\mathsf{m},\mathsf{pk},v_1,\pi_1,v_2,\pi_2) \leftarrow \mathcal{A}(\mathsf{pp})\end{array} : \begin{array}{l}\mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk},\mathsf{m},v_1,\pi_1)=1 \wedge\\ \mathsf{V.Verify}_{\mathsf{pp}}(\mathsf{pk},\mathsf{m},v_2,\pi_2)=1 \wedge\\ v_1 \neq v_2\end{array}\right]$$

If the adversary $\mathcal{A}$ is assumed to be PPT, then we call this property *computational* (full) uniqueness.

Note that the adversary has full control over the generation of the public key in the above uniqueness experiment.

A.2 Security Assumptions

Definition 4 ($\mathsf{MSIS}^{\infty}_{n,d,m,q,\beta}$)**.** *For positive integer parameters (n, m, q, β) with $m > n$, given $\mathbf{A} = [\,\mathbf{I}_n \,\|\, \mathbf{A}'\,] \in \mathcal{R}_{q,d}^{n\times m}$ with $\mathbf{A}' \xleftarrow{\$} \mathcal{R}_{q,d}^{n\times(m-n)}$, the* MSIS *problem asks to find a* short non-zero *vector $\mathbf{v} \in \mathcal{R}^m$ such that $\mathbf{A}\mathbf{v} = \mathbf{0} \in \mathcal{R}_{q,d}^n$ and $\|\mathbf{v}\|_\infty \leq \beta$.*

We define the module variant of LWR problem introduced in [BPR12], with the generalization that the secret coefficients can be sampled from a narrower distribution rather than just uniform over $\mathcal{R}_{q,d}$.

Definition 5 ($\mathsf{MLWR}_{\ell,d,m,q,p,\mathcal{B}}$)**.** *For positive integer parameters $(\ell, m, q, p, \mathcal{B})$ with $p < q$, the* MLWR *problem asks to distinguish between the following two cases: (i) $(\mathbf{A}, \lfloor\mathbf{u}\rceil_p)$ for $(\mathbf{A}, \mathbf{u}) \xleftarrow{\$} \mathcal{R}_{q,d}^{m\times\ell} \times \mathcal{R}_{p,d}^m$, and (ii) $(\mathbf{A}, \lfloor\mathbf{A}\mathbf{s}\rceil_p)$ for $\mathbf{A} \xleftarrow{\$} \mathcal{R}_{q,d}^{m\times\ell}$ and $\mathbf{s} \xleftarrow{\$} \mathbb{S}_{\mathcal{B},d}^{\ell}$.*

In the case that p divides q, $\lfloor\mathbf{u}\rceil_p$ is itself uniform over $\mathcal{R}_{p,d}^m$.

A.3 Rejection Sampling

Our proposals make use of a standard 'Gaussian' rejection sampling technique [Lyu12]. We describe the rejection sampling function in Alg. 5 and refer the reader to [Lyu12] for further details. As a shortcut, we add the last infinity-norm check '$\|\mathbf{z}\|_\infty > 6\sigma$' to make sure that no coefficient is too large.

Algorithm 5 $\mathsf{Rej}(\mathbf{z}, \mathbf{c}, \phi, T)$

1: $\sigma = \phi T$; $\mu(\phi) = e^{12/\phi + 1/(2\phi^2)}$; $u \leftarrow [0, 1)$
2: **if** $u > \frac{1}{\mu(\phi)} \cdot \exp\left(\frac{-2\langle\mathbf{z},\mathbf{c}\rangle + \|\mathbf{c}\|^2}{2\sigma^2}\right)$, **then return** 1
3: **if** $\|\mathbf{z}\|_\infty > 6\sigma$, **then return** 1
4: **else return** 0