The Price of Verifiability: Lower Bounds for Verifiable Random Functions

Nicholas Brandt, Dennis Hofheinz, Julia Kastner, and Akin Ünal

Department of Computer Science
ETH Zurich
Zurich, Switzerland
{nicholas.brandt,hofheinz,julia.kastner,akin.uenal}@inf.ethz.ch

Abstract. Verifiable random functions (VRFs) are a useful extension of pseudorandom functions for which it is possible to generate a *proof* that a certain image is indeed the correct function value (relative to a public verification key). Due to their strong soundness requirements on such proofs, VRFs are notoriously hard to construct, and existing constructions suffer either from complex proofs (for function images), or rely on complex and non-standard assumptions.

In this work, we attempt to explain this phenomenon. We show that for a large class of pairing-based VRFs, it is not possible to obtain short proofs *and* a reduction to a simple assumption simultaneously. Since the class of "consecutively verifiable" VRFs we consider contains in particular the VRF of Lysyanskaya and that of Dodis-Yampolskiy, our results explain the large proof size, resp. the complex assumption of these VRFs.

1 Introduction

Verifiable Random Functions. Pseudorandomness, and in particular pseudorandom generators [7, 51] and pseudorandom functions (PRFs, [22]) have proven to be immensely useful and universal cryptographic building blocks. A PRF takes as input a short seed (or key) sk, and an input x, and outputs a function value $\mathbf{y} = \mathsf{prf}_{\mathsf{sk}}(x)$. The distinguishing feature of a PRF is that for a fixed but random sk, oracle access to $\mathsf{prf}_{\mathsf{sk}}(\cdot)$ cannot be distinguished from oracle access to a truly random function. This allows to use prf as a compact drop-in replacement for a truly random function.

In this work, we focus on a special class of PRFs whose image can be *proven* to be correct (relative to a public key vk that fixes prf's behavior). Indeed, a verifiable random function (VRF [38]) vrf is a PRF for which it is possible to generate proofs π (from a given sk and x) that show that a given $\mathbf{y}$ really satisfies $\mathbf{y} = \mathsf{vrf}_{\mathsf{sk}}(x)$. We want such proofs to be sound in a very strong sense: We require that for *any* vk and x, no two $\mathbf{y} \neq \mathbf{y}'$ can both be proven to be $\mathsf{vrf}_{\mathsf{sk}}(x)$. This property, dubbed "unique provability", is crucial for most applications of VRFs, and is the main reason why constructing VRFs is difficult. For instance, unique provability cannot be achieved by using non-interactive zero-knowledge proofs on a given PRF. (This would require a trusted common reference string, which we cannot assume in the VRF setting.) We do note, however, that (non-straightforward) solutions with non-interactive witness-indistinguishable (NIWI) proofs are possible [6, 25].

VRFs have a number of interesting applications. These include signatures with very strong verifiability guarantees [24], resettable zero-knowledge proofs [39], lottery systems [40], transaction escrow schemes [29], updatable zero-knowledge databases [35], and e-cash systems [2, 4].

Existing Constructions of VRFs. There are a variety of constructions of VRFs already [1, 6, 9, 16, 17, 25–28, 32–34, 36, 38, 44, 46, 50]. These constructions are diverse in the used techniques and the resulting features: For instance, some constructions (such as Lysyanskaya's VRF [36] and its variants [9, 26–28, 46]) are based on the specific algebraic properties of the Naor-Reingold PRF [41], while others (such as [6, 25]) are based on more generic primitives such as NIWI proofs. However, none of the above VRF constructions achieves all of the following useful features simultaneously:

- its input space is large (i.e., exponential in the security parameter),
- its proofs π are short (i.e., comprise a constant number of group elements),
- its security proof is based on a "simple" (i.e., non-interactive and compact[1]) assumption.

We do note that some of the constructions come close: E.g., Kohl's VRF [32] achieves all of the above properties, except that proofs π comprise $\omega(1)$ group elements. Conversely, the VRF of Dodis and Yampolskiy [17] enjoys very compact proofs, but relies on a complex hardness assumption (with challenges as large as the input space). While there exists work on the difficulty of achieving VRFs (e.g., from trapdoor one-way functions [19], cf. [12], or in a tightly secure way [44]), the proof size and necessary assumptions for VRFs are generally not well-understood.

Our Contribution. In this work, we are concerned with the reason *why* it is difficult, even after a plethora of different approaches and 20 years of research, to construct useful and compact VRFs from standard assumptions. In order to give a meaningful answer, we will restrict ourselves to particular classes of VRFs (which however cover many of the previous VRF constructions), and give lower bounds.

Specifically, we restrict ourselves to VRFs vrf in the standard model (i.e., that do not use random oracles or generic groups) that are algebraic over a group, such that secret keys sk are comprised of exponents, and public keys vk, images $\mathbf{y}$, and proofs π are all comprised of group elements. We do allow pairings, however, such that in particular images may be elements of a target group.

Furthermore, we require that verification (of a proof π for an image $\mathbf{y}$) operates in a specific and "consecutive" way. We give more details on the conditions on verification below in the technical overview. We stress, however, that we believe that this way to verify is natural, and in fact many existing VRFs support consecutive verification, including Lysyanskaya's VRF [36], the VRF of Dodis and Yampolskiy [17], and many more (see Fig. 1). A convenient consequence of this type of consecutive verification is that the function image $\mathbf{y}$ has a specific form: We can deduce that $\mathbf{y} = \mathsf{vrf}_{\mathsf{sk}}(x)$ is of the form $\mathbf{g}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})}$, where

[1] With a non-interactive and compact assumption, we mean one in which the adversary gets a constant number of group elements as challenge and is then supposed to output a solution (e.g., a decision bit).

- $\mathbf{g}$ is a fixed group generator,
- σ_x and ρ_x are multivariate polynomials (that depend in any efficiently computable way on the preimage), and
- $\overrightarrow{\vartheta}$ is the vector of discrete logarithms of the verification key vk.

We finally assume a large (i.e., superpolynomial in the security parameter) input space. Again, while this of course severely restricts the VRFs we consider, many previous constructions fall into this class.[2]

For such algebraic VRFs with consecutive verification, we show necessary relations between the size of proofs π and the "size" of the underlying assumption (i.e., the size of the challenge in group elements in a non-interactive hardness assumption). To develop and express these relations, it is useful to consider what we call the *evaluation degree* of the VRF. Formally, this degree is simply the maximum of the degrees of the (multivariate) polynomials σ_x and ρ_x from the image $\mathbf{y} = \mathbf{g}^{\sigma_x(\overrightarrow{\vartheta})/\rho_x(\overrightarrow{\vartheta})}$ above (and for this exposition, we assume that these degrees do not depend on x).

We show that for any VRF vrf that matches all of our formal requirements,

(a) if the size of π (in group elements) is small, then so is the degree of vrf,
(b) if vrf's degree is small, then vrf cannot be proven secure with a generic reduction to a constant-size non-interactive hardness assumption. (We note that almost all existing cryptographic reductions are generic.)

As an example, our results show that the VRF of Dodis and Yampolskiy cannot be proven secure (at least not generically) from more traditional hardness assumptions. Our results also show that the (comparatively large) proofs in Lysyanskaya's VRF are inherent, at least when relying on standard hardness assumptions. Figure 1 lists more VRFs that fulfill our requirements (and whose proof sizes and/or assumptions can hence be justified with our results).

While our result (a) is a direct consequence of our requirement on consecutive verifiability, we in fact give two versions of statement (b) that differ in exact requirements and formalization. For instance, one version of (b) even excludes *algebraic* reductions (i.e., is formalized within the algebraic group model [21]) from non-interactive assumptions of any polynomial size, but only applies to VRFs whose verification keys only depend on a single variable or from non-interactive computational assumptions that depend on a single variable. This allows to model Dodis and Yampolskiy's VRF, but not Lysyanskaya's. The other version of (b) allows more general verification keys, but only excludes *generic* reductions (i.e., is formalized within the generic group model [37, 43, 48]). In the next section, we give a more technical overview over our results.

Discussion. While the formal requirements for our lower bounds seem restrictive, their preconditions are met by most existing VRFs (see Fig. 1). In that sense, they justify the limitations of existing constructions, resp. proofs. An obvious question is thus: How can

[2] A prominent verifiable *unpredictable* function (VUF, a weaker form of VRF) that does not fall into this class is the one by Brakerski *et al.* [12]. This VUF takes group elements *as input*, and hence does not quite fit into our framework. We will discuss this particular construction in Section 2.1, and argue that this approach is unlikely to yield purely group-based *VRFs*.

one circumvent our lower bounds (in order to construct VRFs with short proofs from standard assumptions)?

First of all, one could of course circumvent our results by not (or at least not completely) working over cyclic groups. However, while there are a few more generic VRF constructions (e.g., [6, 25]) that do not rely on groups, it seems that generic VRF constructions are less well-investigated than constructions based on cyclic groups.

Second, one could try to circumvent the more specific requirements of our lower bounds. In particular, our "consecutive verifiability" requirement seems like a very specific requirement. An "interesting" (as opposed to a purely mechanical) way to circumvent consecutive verifiability would be the following. Recall that consecutive verifiability implies that VRF images consist of rational functions, i.e., are of the form $\mathbf{y} = \mathbf{g}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})}$. Jumping ahead, we will be interested in small-degree polynomials σ_x, ρ_x. The following VRF candidate does not have this property:

$$\mathsf{vk} = e(\mathbf{g}, \mathbf{g})^s, \qquad \mathbf{y} = \mathbf{g}^{\sqrt[3]{s+x}} \qquad \pi = \mathbf{g}^{(\sqrt[3]{s+x})^2}.$$

Verification checks that $e(\mathbf{y}, \mathbf{y}) = e(\mathbf{g}, \pi)$ and $e(\pi, \mathbf{y}) = \mathsf{vk} \cdot e(\mathbf{g}, \mathbf{g})^x$. The security of this VRF candidate seems unclear, but observe that we require $3 \not| (\mathrm{ord}(\mathbf{g}) - 1)$ both for uniqueness, and to be able to compute $\sqrt[3]{s+x} \bmod \mathrm{ord}(\mathbf{g})$.

More generally, our results do not exclude VRFs in which the image is an active ingredient in intermediate verification computations, and not only considered in a final verification step (that involves previously computed and/or verified proof elements). Of course, for constructions that use, e.g., roots of exponents (like the above candidate), it may be challenging to prove their security from Diffie-Hellman-like assumptions.

1.1 High-level Technical Overview

The Evaluation Degree of a VRF. Our technical results rely on the "evaluation degree" of a VRF vrf as a helpful technical notion that connects vrf's proof sizes and vrf's underlying hardness assumption. Hence, let us first take a closer look at this notion of degree.

First, we recall one of our restrictions on the VRFs we consider. We assume that vk and π consist of group elements, and that verification operates in a "consecutive" way, in the following sense: Assume that verification wants to verify a proof π (which consists of, say, κ group elements $\pi_1, \ldots, \pi_\kappa$) for an alleged image $\pi_{\kappa+1} := \mathbf{y}$ (which is a single group element). Then, we require that verification proceeds in $\kappa+1$ steps, and in the i-th step checks an a priori fixed system of pairing product equations in variables $\pi_1, \ldots, \pi_i$ and vk. We also require that in the equations for the check for π_i, this element only occurs linearly (but not quadratically, i.e., in both arguments of a pairing).

Verification succeeds if all these systems of equations hold. In other words, proof elements (and eventually image $\mathbf{y}$) are verified one at a time, each time checking a quadratic equation in the corresponding exponents of this and all previous elements and vk.

This notion of consecutive verification sounds natural in a pairing setting, and indeed many existing vrf constructions (including the ones from [17, 36]) have a consecutive verification procedure in the above sense. Intuitively, consecutive verification

Reference	CV	degree	$\lvert\mathsf{vk}\rvert$	$\lvert\pi\rvert$	assumption	remark
MRV99 [38]	x	—	large	large	RSA	tree-based
Lys02 [36]	✓	λ	2λ	λ	q-type	
Dod03 [16]	✓	$O(\lambda)$	$O(\lambda)$	$O(\lambda)$	ad-hoc	
DY05 [17]	✓	1	2	1	q-type	small inputs
ACF09 [1]	✓	$\lambda + 2$	$2\lambda + 2$	$\lambda + 1$	q-type	
BCKL09 [4]	x	1	3	$O(1)$	q-type	small inputs
BGRV09 [12]	x	—	1	1	gap-CDH	weak security
BMR10 [9]	✓	$\lambda + 1$	$(\lambda + 2)$	λ	q-type	small inputs
HW10 [27]	✓	$\lambda + 1$	$\lambda + 3$	$\lambda + 1$	q-type	
Jag15 [28]	✓	$O(\lambda)$	$O(\lambda)$	$O(\lambda)$	q-type	
LLC15 [34]	✓	$\lambda + 1$	$2\lambda + 1$	1	q-type	multilinear maps
HJ16 [26]	✓	$O(\lambda)$	$O(\lambda)$	$O(\lambda)$	DLIN	
Bit17 [6]	x	—	depends	large	depends	generic/NIWI-based
GHKW17 [25]	x	—	depends	large	depends	generic/NIWI-based
Kat17 [30]	✓	$\omega(\log(\lambda)^2)$	$\omega(\sqrt{\lambda}\log(\lambda))$	$\omega(\sqrt{\lambda})$	q-type	
Yam17 [50]	✓	$O(\log(\lambda)^2)$	$O(\lambda\log(\lambda)^2)$	$O(\log(\lambda)^2)$	q-type	
Ros18 [46]	✓	$O(\lambda)$	$O(\lambda)$	$O(\lambda)$	DLIN	smaller π than [26]
Koh19 [32]	✓	κ	$\mathrm{poly}(\lambda)$	κ	DLIN	$\kappa \in \omega(1)$ parameter
Nie21 [44]	✓	$O(\lambda)$	$\omega(\log(\lambda))$	$O(\lambda)$	q-type	

Fig. 1. Existing VRF constructions. The "CV" column indicates whether the construction is consecutively verifiable in our sense. "Degree" denotes its evaluation degree (where applicable), and $|\mathsf{vk}|$ and $|\pi|$ denote its verification key size, resp. proof size in group elements. When possible, we have chosen parameters such that the input size is $\{0,1\}^\lambda$. For comparability, we classify assumptions with polynomially many challenge elements as "q-type", and other nonstandard assumptions as "ad-hoc". "Small inputs" (as a remark) means that the VRF only supports polynomially-small input spaces.

requires that "higher-degree" exponents in proof elements or image must be verified using intermediate group elements with intermediate degrees. Fortunately, as already outlined, consecutive verification also implies that images $\mathbf{y}$ are of the form

$$\mathsf{vrf}_{\mathsf{sk}}(x) = \mathbf{y} = \mathbf{g}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})}$$

for multivariate polynomials σ_x and ρ_x (which both are efficiently computable from x), and the component-wise discrete logarithm $\overrightarrow{v}$ of vk. Now we say that the *evaluation degree* of vrf (or $\mathbf{y}$) is simply the maximum of the polynomial degrees of σ_x and ρ_x. The evaluation degree of the VRF is then simply the maximal degree over all inputs x.

First Result: Proof Size Bounds Degree for VRFs with Consecutive Verification. Our first result ((a) above, described in more detail in Section 2.1, and in full detail in Section 4) shows that for VRFs vrf with consecutive verification (as above), the size of proofs π imposes a limit on the vrf's evaluation degree. Concretely, we show that the evaluation degree of vrf is at most exponential in the proof size κ. Hence, if its proof size is constant, then so is the evaluation degree of vrf.

This result is not too surprising, since intuitively, each additional proof element only raises the degree of computed exponents (as algebraic fractions in $\overrightarrow{v}$) by a factor of 2. In fact, our proof largely consists in keeping track of expressions of all intermediate proof elements (and finally of $\mathbf{y}$) as expressions in $\overrightarrow{v}$. The main technical work consists in maintaining a suitable canonical form of these (rational) expressions at all times.

Interlude: the Case of Trivial Denominators. If function images are of the form $\mathbf{y} = \mathbf{g}^{\sigma_x(\overrightarrow{v})}$ for a constant-degree (but multivariate) polynomial σ_x, already a very simple linear algebra attack breaks the pseudorandomness of the given VRF. In fact, for sufficiently many preimages x_i, the polynomials σ_{x_i} must eventually become linearly dependent (because the set of their monomials is polynomially small). Hence, it is possible to linearly combine sufficiently many given images to form the image of a fresh preimage. This breaks pseudorandomness, and we detail this attack in Appendix A for completeness. The case of rational function images $\mathbf{y} = \mathbf{g}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})}$ (with $\deg(\rho_x) \geq 1$) is hence not only more general (and covers, e.g., the Dodis-Yampolskiy VRF), but also technically much more interesting.

Second Result: Security of Polynomial-Degree VRFs Requires Complex Assumptions (for Univariate Verification Keys and in the Algebraic Group Model). Our second result (first variant of (b) above, described in Section 2.2 more extensively, and in Section 5 in full detail) shows that for any polynomial-degree VRF vrf, we can rule out the existence of an "algebraic black-box" reduction to a class of non-interactive group-based computational assumptions. Here, an "algebraic black-box" reduction $\mathcal{B}$ fulfills the following requirements:

- It is algebraic (in the sense of [21]): That means that whenever $\mathcal{B}$ outputs a group element $\mathbf{g}^*$, it also outputs (on a special channel) an explanation as to how $\mathbf{g}^*$ is computed from previously seen group elements.
- It uses the VRF adversary $\mathcal{A}$ only in a black-box way (i.e., it gets oracle access to polynomially many instances of $\mathcal{A}$).

Most existing reductions (in particular for VRFs) are simple in the above sense.

A non-interactive (group-based) computational assumption (NICA) states that it is hard for any efficient adversary $\mathcal{B}$ to win the following game: $\mathcal{B}$ gets a challenge (that is a vector of s group elements), and is then supposed to output a solution to that challenge (which can be of any form).

We are now ready to state our result a bit more formally: Assume we are given a polynomial-degree VRF vrf with verification key $\mathsf{vk} = \mathbf{g}^v$. Furthermore, assume that vrf enjoys a simple reduction $\mathcal{B}$ to a NICA. Then, we construct a meta-reduction [13] that wins the NICA game without any external help. Our meta-reduction $\mathcal{M}$ interacts with $\mathcal{B}$ (which gets a NICA challenge), and then attempts to take the role of a successful VRF adversary $\mathcal{A}$. In order to do this, $\mathcal{M}$ can query many VRF images $\mathbf{y}_i$, and use the algebraicity of $\mathcal{B}$ to obtain representations of these $\mathbf{y}_i$ in terms of the NICA challenge elements. Hence, eventually $\mathcal{B}$ will find linear dependencies between the queried VRF images by making sufficiently (but still polynomially) many queries. These linear dependencies can then be used to compute the verification key's exponent v. Using v, the meta-reduction can predict *any* challenge image as $\mathbf{g}^{\sigma_x(v)/\rho_x(v)}$. This allows $\mathcal{A}$ to win the VRF security game, and hence $\mathcal{M}$ can use $\mathcal{B}$ to solve the NICA.

This intuition neglects a number of technical obstacles: For instance, the linear dependencies among the algebraic representations of VRF images linearly connect the algebraic fractions $\sigma_{x_i}(v)/\rho_{x_i}(v)$ of the corresponding images. To construct a new image $\mathbf{g}^{\sigma_{x^*}(v)/\rho_{x^*}(v)}$ from these, we need to distinguish the cases when the polynomial fraction $\sigma_{x^*}(X)/\rho_{x^*}(X)$ of the challenge can be expressed as a linear combination of the polynomial fractions $\sigma_{x_i}(X)/\rho_{x_i}(X)$ of the queries, and when this is not the case. In the first case, the corresponding linear dependence immediately allows to compute $\mathbf{g}^{\sigma_{x^*}(v)/\rho_{x^*}(v)}$. Note that this is also possible for an adversary that does *not* get to see the algebraic representations because the linear dependence holds for the fractions, not only for the representations.

In the second case, we have to develop a linear dependence among the algebraic representations (in the NICA challenge elements) of the $\sigma_{x_i}(v)/\rho_{x_i}(v)$. In this case, in fact the linear *independence* of the fractions $\sigma_{x_i}(X)/\rho_{x_i}(X)$ guarantees that these linearly dependent algebraic representations allow to extract the secret v.

In these observations, we crucially use that we deal with univariate polynomials σ_{x_i} and ρ_{x_i} of small degree (which can be represented by short coefficient vectors). In a separate result, we generalize this approach to multivariate σ_{x_i} and ρ_{x_i} where the underlying assumption only depends on a single variable with polynomial degree.

Third Result: Security of Low-Degree VRFs Requires Complex Assumptions (in the Generic Group Model). Our last result (second variant of (b) above, explained more extensively in Section 2.3, and in full detail in Section 6.1 for the case of the Dodis-Yampolskiy VRF, and in Section 6.3 for a more general case) is similar in spirit to our second result, but features different requirements on the considered VRFs and reductions. Specifically, we prove that no generic reduction (i.e., that treats the underlying group as generic in the sense of [48]) that is algebraic black-box (as outlined above) is able to show security of a constant-degree VRF based on any "Uber-assumption" [8, 11] of arbitrary polynomial degree but constant challenge size.

An "Uber-assumption" is a special class of a NICA in which an adversary $\mathcal{B}$ is given a number of group elements $\mathbf{g}^{f_i(\overrightarrow{z})}$, where the f_i are multivariate polynomials specific to the concrete assumption, and $\overrightarrow{z}$ is a vector of secret (and uniformly randomly chosen) exponents. Typically, the task of $\mathcal{B}$ is then to compute a group element not in the linear span of the given group elements (or to distinguish such an element from random). Here, we restrict ourselves to Uber-assumptions in which the degree of the f_i is at most polynomial in the security parameter.

We again give a meta-reduction $\mathcal{M}$ that shows the following: Any simple generic reduction $\mathcal{B}$ that shows the security of a constant-degree VRF under such an Uber-assumption can be transformed into a successful Uber-solver. Again, $\mathcal{M}$ takes the role of a VRF adversary that interacts with $\mathcal{B}$. In the following, we outline our technique for the specific case of the Dodis-Yampolskiy VRF, in which $\mathsf{vk} = (\mathsf{vk}_1, \mathsf{vk}_2) = (\mathbf{h}, \mathbf{h}^s)$, $\mathbf{y} = e(\mathbf{h}, \mathbf{h})^{1/(s+x)}$ (for a pairing e), and $\pi = \mathbf{h}^{1/(s+x)}$.

Our meta-reduction $\mathcal{M}$, when interacting with a reduction $\mathcal{B}$ in the role of a VRF adversary $\mathcal{A}$, first of all gets to see vk and an algebraic representation of vk in terms of the NICA challenge. (In this work, we call an algorithm generic iff it is generic in the sense of Shoup's GGM *and* algebraic, cf. Definition 5.) This representation of

$\mathsf{vk} = (\mathsf{vk}_1, \mathsf{vk}_2)$ allows $\mathcal{M}$ to write $\mathsf{vk}_i = \mathbf{g}^{g_i(\overrightarrow{z})}$ for polynomials g_i in the Uber-assumption secrets $\overrightarrow{z}$.

Now we distinguish two cases: First, if the polynomial g_2 is a scalar multiple of g_1, (i.e., if $g_2 = s' \cdot g_1$ for a scalar s'), then we have found the VRF secret key $s = s'$. This s can directly be used to break VRF security and allows $\mathcal{M}$ to imitate a successful adversary for $\mathcal{B}$ (which in turn breaks the underlying Uber-assumption). But in case g_2 is not a scalar multiple of g_1, such a simple extraction of s is not possible.

The main technical work in our proof consists in showing that this second case cannot, in fact, occur with non-negligible probability. Essentially, we do so by observing that the representations of VRF proofs $\pi_i = \mathbf{h}^{1/(s+x_i)}$ (i.e., of $(s + x_i)$-th roots of $\mathbf{h} = \mathsf{vk}_1$) imply polynomial factors of g_1. We prove that if g_2 is not a scalar multiple of g_1, then these factors are coprime for different x_i. Hence, querying sufficiently many VRF proofs (for different x_i) yields many non-trivial coprime factors of g_1. Since we assumed that the degree of g_1 is polynomial (since the Uber-assumption polynomials f_i are of polynomial degree), this eventually yields a contradiction. Hence, g_2 must be a scalar multiple of g_1, and our meta-reduction $\mathcal{M}$ can proceed as described above.

In Section 6.3, we also show how to generalize this argument to a broader class of constant-degree VRFs which we call *parameterized*.

Omitted Details. All the above explanations have omitted or simplified a few details. For instance, we did not discuss the role of *group parameters* (that fix the concrete group and pairing setting). For VRFs, such group parameters should be *certified* [26] (i.e., reliably defining an actual group), and they can be an additional part of vk or any public parameters. Since generic groups can be viewed as "implicitly trusted", we omit this certification in the generic group model.

Furthermore, we have treated the VRF image always as a target group element. However, since we are in a pairing setting, this image can also (and in fact without loss of generality) be an element of the *source group* of the pairing. (This does not change any of the arguments above.) Finally, we mostly consider verifiable *unpredictable* functions (VUFs), a relaxation of verifiable *random* functions. Since we present lower bounds, this only makes our results stronger.

2 Detailed Technical Overview

2.1 First Result: Connecting the Proof Size with the Evaluation Degree

Consecutively Verifiable VUFs/VRFs. To make the connection between the number of group elements in the proof and the evaluation degree, we first define a class of VUFs/VRFs that have a very straightforward verification algorithm. We assume that the VUFs/VRFs in question operate over a symmetric[3] pairing group with pairing $e\colon \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$:

- The verification key vk consists of group elements $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{G} \cup \mathbb{G}_T$

[3] We note that our results can easily be transferred to asymmetric pairings, but for simplicity we restrict ourselves to symmetric pairings.

- For each input x, the proof consist of group elements $\pi_1, \ldots, \pi_\kappa \in \mathbb{G} \cup \mathbb{G}_T$
- For each input x, the evaluation value is a group element $\mathbf{y} \in \mathbb{G}_T$

Each possible input element x of the VUF/VRF defines a set of pairing equations E_x that can be efficiently derived[4] from the input x. By pairing equations we mean a set of polynomial equations of degree 2 in the input variables. We make the additional restriction that variables that represent elements from the target group may appear only in monomials of degree 1. We require that the pairing equations can be verified *consecutively*, that is, there is an ordering of the group elements in the proof and subsets $E_{i,x}$ of the sets of pairing equations such that the following hold:

- in the pairing equation set $E_{i,x}$ for the i-th proof element, only the verification key elements and proof elements up to the i-th occur, i.e., $E_{i,x} \subset \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots P_i]$
- in the pairing equation set $E_{i,x}$ for the i-th proof element, there is at least one equation where the i-th proof element occurs only linearly, i.e., there exist polynomials $a_i \in \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots, P_{i-1}], b_i \in \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots P_i]$ such that $a_i \cdot P_i + b_i = 0$ is an equation that occurs in $E_{i,x}$.

We further make a more technical requirement that the coefficient a_i of the i-th proof element in the equation where it occurs linearly cannot become zero. Let the proof have κ many elements, then we consider the evaluation value to be the $\kappa + 1$st proof element, i.e., it is the last group element to be "verified" in this way.

This consecutive verification property on the one hand yields an efficient pairing-based verification algorithm (for input x, first efficiently derive the pairing equation sets $E_{i,x}$, then consecutively check them). On the other hand, the linearity requirement actually implies that given the verification key and the previous proof elements, each proof element is uniquely defined. As the evaluation value is the last element to be verified, i.e., the $\kappa + 1$st "proof element", it is therefore also uniquely provable.

We note that this consecutive verification property applies to many known VRFs, see Fig. 1 for a detailed overview.

We briefly sketch how the pairing equations look for the VRF of Dodis & Yampolskiy [17]: Recall that the evaluation key is $\mathsf{sk} = s \in \mathbb{Z}_p$ and the verification key is $\mathsf{vk} = \mathbf{h}^s$ for a publicly known group generator $\mathbf{h}$ of $\mathbb{G}$. Evaluation at value x computes $\mathbf{y} = e(\mathbf{h}, \mathbf{h})^{\frac{1}{s+x}}$ as well as the proof $\pi = \mathbf{h}^{\frac{1}{s+x}}$. We can consecutively verify this as follows: First verify the proof via $E_{1,x} = \{(V + x) \cdot P = 1\}$ where V represents the verification key, and P represents the group element. That is, the verification algorithm checks $e(\mathsf{vk} \cdot \mathbf{h}^x, \pi) = e(\mathbf{h}, \mathbf{h})$. Then, we verify $E_{2,x} = \{P \cdot 1 = Y\}$ where P is as before and Y represents the evaluation value, that is the verification algorithm checks $e(\pi, \mathbf{h}) = \mathbf{y}$.

Remark 1 (Consecutive Verifiability of the VUF of Brakerski et al. *[12]).* As we pointed out above, the weak VUF of Brakerski *et al.* [12], where evaluation works by $\mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}, \mathbf{h}) = \mathbf{h}^{\mathsf{sk}}$ for $\mathsf{sk} \in \mathbb{Z}_p$ and $\mathsf{vk} = \mathbf{g}^{\mathsf{sk}}$ and an input $\mathbf{h} \in \mathbb{G}$, and verification accepts if $e(\mathbf{h}, \mathsf{vk}) = e(\mathbf{y}, \mathbf{g})$, is not consecutively verifiable in the sense of this work. In fact,

[4] We note that the weak VRF by Brakerski *et al.* [12] does not have this efficiency property, as the inputs are group elements and the pairing equations can only be derived from the discrete logarithm of the inputs.

we would need to know the discrete logarithm of the input $\mathbf{h}$ to efficiently compute a pairing equation for it. Therefore, the results of this paper are not applicable to this VUF.

However, while this might seem to limit the class of VUFs we consider in this work, we claim that weak VUFs that have group elements as inputs are – for the pursuit of strong VRFs – not relevant, anyway. In fact, images of the weak VUF of Brakerski *et al.* [12] can easily be predicted for adversially chosen inputs. This observation can be extended to other weak VRF/VUF candidates that operate in a similar algebraic manner, i.e., that take group elements as inputs and interpret them *as group elements* only and use the group operations and pairing operations on them to compute the output. These VRFs/VUFs become insecure by Theorem 10 as their evaluation degree is at most 2 in the inputs if the discrete logarithms of the input group elements are known to the adversary.

Rational VUFs/VRFs. We want to show that the formerly mentioned class of consecutively verifiable VUFs/VRFs has a particularly straightforward way to describe their evaluation algorithm. To this end, we define rational VUFs. These are VUFs whose evaluation value consists of a (publicly known) group generator raised to a rational function evaluated on the exponents of the verification key. More formally, for each input value x, there are polynomials ρ_x and σ_x such that the output $\mathbf{y}$ evaluated at x is

$$\mathbf{y} = \mathbf{g}_{\mathbf{T}}^{\frac{\sigma_x(v_1,\ldots,v_n)}{\rho_x(v_1,\ldots,v_n)}}$$

where $v_1, \ldots, v_n$ are the exponents of the group elements in the verification key vk. We say that the total degree of the polynomials σ_x and ρ_x is the *evaluation degree* of the VUF/VRF.

From Consecutive Verifiability to Rationality with Bounded Degree. We show, using an inductive argument, that (a) consecutively verifiable pairing based VUFs/VRFs are also rational VUFs/VRFs, and (b) that the evaluation degree is at most exponential in the proof size – this implies that the proof size needs to be at least logarithmic in the evaluation degree for consecutively verifiable VUFs/VRFs. The proof uses induction to show that in fact all proof elements can be expressed through rational functions in the exponent, i.e., there exist σ_{x,π_i} and ρ_{x,π_i}, and that the degree of the i-th proof element is at most 4^i. The base case is easy to see: To obtain σ_{x,π_1} and ρ_{x,π_1} from the first set of pairing equations, we use the pairing equation that contains P_1 as a linear factor. This equation can be expressed as $a \cdot P_1 + b = 0$ where a, b are polynomials (a has degree at most 1 and b degree at most 2). We can therefore express $P_1 = b / - a$.

For the inductive step it is again crucial that the i-th proof element occurs only linearly in at least one pairing equation, as it can then be viewed as a zero of a linear equation and expressed as a rational function of the previous proof elements and the verification key. We replace the previous proof element P_{i-1} by its rational expression $\frac{\sigma_{x,\pi_{i-1}}}{\rho_{x,\pi_{i-1}}}$ in the pairing equation set $E_{i,x}$ to obtain $P_i \cdot a'_i + b'_i = 0$ where the a'_i and b'_i are rational functions in the verification key elements. We then derive the rational expression for $P_i = b'_i / - a'_i = \sigma_{x,\pi_i} / \rho_{x,\pi_i}$ where σ_{x,π_i} and ρ_{x,π_i} are polynomials.

It remains to show that the resulting polynomials have the degrees required by our statement which can be done using some simple arguments.

Inductively replacing all proof elements by such rational expressions in the verification key elements yields the result for the last element to be verified – the evaluation value.

2.2 Second Result: Security of Univariate Polynomial-Degree VRFs Requires Complex Assumptions

In current pairing-based constructions of VRFs there seems to be a tradeoff between the size/complexity of the underlying assumption and the size of the proofs. Some constructions, like [17], achieve constant-sized proofs but require a q-type assumption, while others [32] achieve proofs of any superconstant size under a constant-sized assumption. Here, we consider VRF constructions based on non-interactive (group-based) computational assumptions (NICA), i.e., search problems as opposed to a decisional assumptions. These NICAs state that any "efficient" algorithm only has a negligible probability of solving the corresponding computational problem, e.g. finding some "secret" exponent. In particular, we consider NICAs where the challenge elements' exponents only depend on a single variable with polynomial degree (these include for example the q-DLog-assumption and the q-DHI-assumption. There the challenge is $\mathbf{g}, \mathbf{g}^{\alpha}, \mathbf{g}^{\alpha^2}, \ldots, \mathbf{g}^{\alpha^q}$ and the secret exponent is α. We give two meta-reductions [14] (for slightly different settings) that break the resp. underlying assumption if there is an algebraic reduction from the assumption to the unpredictability (resp. pseudorandomness) of the VUF (resp. VRF).

Theorem 1 (Informal Lower Bound for Univariate VUFs). *Let* vuf *be a rational VUF whose verification key exponents depend—with polynomial degree—on a single common variable. Let* NICA *be any NICA of polynomial size. If there exists an algebraic reduction that transforms an adversary for the weak selective unpredictability of* vuf *into a solver for* NICA*, then* NICA *can be solved in polynomial time with some noticable advantage.*

Theorem 2 (Informal Lower Bound for Univariate NICAs). *Let* vrf *be a rational VRF. Let* NICA *be any NICA of polynomial size whose exponents depend—with polynomial degree—on a single common variable (e.g.* q-DLog *or* q-DHI*). If there exists an algebraic reduction that transforms an adversary for the adaptive pseudorandomness into a solver for* NICA*, then* NICA *can be solved in polynomial time with some noticable advantage.*

Remark 2 (Separation between Decisional and Computational Assumptions). As a theoretical sidenote, we observe that on the one hand non-interactive *decisional* assumptions, like q-DDH, suffice for constructing VRFs [50], while on the other hand (univariate) non-interactive *computational* assumptions, like the q-DLog or q-DHI assumption, do not suffice via algebraic reductions. This yields in particular an algebraic separation between the q-DDH and the q-DLog assumption.

Remark 3 (No Algebraic GL Construction). One can transform a VUF (e.g. the VUF of Dodis & Yampolskiy [17] based on the q-DHI assumption) into a VRF via the construction of Goldreich & Levin [23]. While this seems like a contradiction (because it gives a VRF based on the q-DHI assumption), it is actually consistent with our results because the GL hardcore bit is not an algebraic technique[5], hence the reduction from the q-DHI assumption to the pseudorandomness of the resulting VRF is not an algebraic reduction. By contraposition, our results show that there cannot be an algebraic analogue of the GL construction.

Our Technique. Both meta-reductions share the same core idea. In a nutshell, the meta-reduction—when simulating an adversary towards the reduction—uses the representation vectors[6] of the received group elements to either (a) predict the challenge image, e.g. as a linear combination of received representations, or (b) construct a polynomial function over the exponent field $\mathbb{Z}_p$ which has the NICA's secret exponent as a zero. Thus, in case (a) the meta-reduction could successfully answer the reduction's challenge while in case (b) the meta-reduction can leverage the fact that polynomials over some finite field can be efficiently factorized and solve its own challenge directly using the NICA's secret exponent.

In both cases the meta-reduction relies on the facts that the VUF (resp. VRF) has correctness and unique provability, and that the VUF (resp. VRF) is of *rational* form, i.e., $\mathsf{vrf}_{\mathsf{sk}}(x) = \mathbf{g_T}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})}$ where σ_x, ρ_x are of polynomial degree and $\overrightarrow{v}$ is the vector of verification key exponents. Because the reduction is algebraic, whenever it outputs a group element $\mathbf{y} \in \mathbb{G}_T$ it must also provide a representation $\overrightarrow{z} \in \mathbb{Z}_p^L$ w.r.t. the NICA challenge elements s.t.

$$\mathbf{g_T}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})} = \mathbf{y} = \mathbf{g_T}^{f_1(s)z_1+\ldots+f_L(s)z_L} \tag{1}$$

$$\iff \sigma_x(\overrightarrow{v}) - (f_1(s)z_1 + \ldots + f_L(s)z_L)\rho_x(\overrightarrow{v}) = 0 \tag{2}$$

where $\mathbf{g}^{(f_1(s),\ldots,f_L(s))} \in \mathbb{G}^L$ is the NICA challenge and $s \overset{\$}{\leftarrow} \mathbb{Z}_p$ is the secret exponent. Equation (2) is the basis for both meta-reductions. For Theorem 1 the meta-reduction queries many preimages $x_1, \ldots, x_Q$ and challenge x_0 uniformly at random. We consider two cases (for simple exposition we assume that the verification key only has one group element $\mathbf{g}^v$):

In the first case (a) the rational functions $\sigma_{x_i}(V)/\rho_{x_i}(V)$ are linearly dependent. With this linear dependence the meta-reduction can predict the challenge image by combining the representations of the queried images.[7]

In the second case (b) although the rational functions $\sigma_{x_i}(V)/\rho_{x_i}(V)$ are linearly independent, by a counting argument there must exist a linear dependence $\alpha \in \mathbb{Z}_p^Q$ among the representations of the queried preimages. The meta-reduction computes the polynomial $\psi(V) := \rho_{x_1}(V) \cdots \rho_{x_Q}(V) \cdot \sum_{\ell=1}^{Q} \alpha_\ell \sigma_\ell(V)/\rho_\ell(V)$. Because $\sigma_{x_i}(V)/\rho_{x_i}(V)$

[5] The GL construction uses the bits of the representation of the group elements.

[6] Recall that we consider algebraic reductions here, so they have to output a vector of representations with each group element.

[7] If all $\sigma_{x_i}(V)/\rho_{x_i}(V)$ are $q+1$-wise linearly dependent, then with high probability the challenge's function $\sigma_{x_0}(V)/\rho_{x_0}(V)$ will be linearly dependent on the other rational functions because all x_i are independent and identically distributed.

are linearly independent, the polynomial is non-zero yet it contains the vk's exponent v as a zero (due to $\sum_{\ell=1}^{Q} \alpha_\ell \sigma_\ell(v)/\rho_\ell(v) = 0$). Thus the meta-reduction can factor the polynomial ψ to obtain the secret exponent and predict the challenge image as $\mathbf{g_T}^{\sigma_{x_0}(v)/\rho_{x_0}(v)}$.

For Theorem 2 we consider adaptive pseudorandomness, hence the meta-reduction obtains a representation for each verification key element and a representation $\overrightarrow{z}^*$ for the challenge image $\mathbf{y}^*$. That is, the meta-reduction knows a function[8] $t : \mathbb{Z}_p \to \mathbb{Z}_p^L$ that maps the NICA challenge's secret key to the verification key exponents $\overrightarrow{v} = t(s)$. Plugging t into Eq. (2) gives

$$\sigma_x(t(s)) - (f_1(s)z_1 + \ldots + f_L(s)z_L)\rho_x(t(s)) = 0 \,. \tag{3}$$

Now, for any representation $\overrightarrow{z}$ of the real challenge image the univariate polynomial $\psi_{\overrightarrow{z}}(S) := \sigma_x(t(S)) - (f_1(S)z_1 + \ldots + f_L(S)z_L)\rho_x(t(S))$ must vanish on the secret exponent s due to Eq. (3).
If $\psi_{\overrightarrow{z}}(S) \not\equiv 0$ is non-zero for all $\overrightarrow{z}$, then the meta-reduction can factorize $\psi_{\overrightarrow{z}^*}(S)$ and find a list of polynomially many candidates for the NICA's secret exponent. If no candidate matches the NICA's secret exponent, then the challenge image $\mathbf{y}^*$ must be random, otherwise the meta-reduction has trivially found the NICA's secret exponent.
On the other hand, if $\psi_{\overrightarrow{z}}(S) \equiv 0$ is zero for some $\overrightarrow{z}$, then the meta-reduction can efficiently find such a representation $\overrightarrow{z}$. Due to Eq. (3) such a $\overrightarrow{z}$ must correspond to the correct challenge image, hence the meta-reduction can distinguish the given element from random.

2.3 Third Result: Security of Low-Degree VRFs Requires Complex Assumptions

As explained before, Theorem 1 states that there is no algebraic reduction that transforms an adversary for the unpredictability of a rational VUF with polynomial evaluation degree to a solver for a hard polynomial size assumption. However, this result has the caveat that the VUF in question needs to have univariate verification keys, i.e., the verification key needs to be fully determined by one secret variable.

In the remaining part of this work, we will circumvent this problem and show lower bounds for another class of VUFs – the class of *rational parametrized VUFs* (Definition 18) – which imposes no restrictions on the verification keys of its VUFs. This class contains the candidates of Dodis & Yampolskiy [17] and of Belenkiy *et al.* [4] and all other DY-inspired candidates.

However, this result comes at a cost: It only shows the impossibility of *generic* reductions that transform adversaries for the unpredictability of parametrized VUFs into solvers of *extremely small* – yet superconstant – Uber-assumptions.

Informally, our result states the following:

Theorem 3 (Informal Lower Bound for Rational Parametrized VUFs). *Let* vuf *be a parametrized rational VUF of constant evaluation degree, i.e., it is rational and the numerators and denominators for evaluation depend polynomially on the input* $x \in \mathbb{Z}_p$*. Let* NICA *be an Uber-assumption of size* $\sqrt{\log\log \mathsf{poly}(\lambda)}$.

[8] For simplicity assume that all f_i and hence t are polynomials.

Then, there is no generic reduction that transforms an adversary for the weak selective unpredictability of vuf *to a* NICA *solver.*

We want to emphasize the significance of Theorem 3 for the pursuit of pairing-based VRFs with proofs of constant size. Theorem 3 shows that the security of each VUF in the style of [17] with constant proofs cannot be generically based on a constant-size Uber-assumption.

Now, we want to explain some details that appear in the statement of Theorem 3 before we jump to a proof:

Uber-Assumptions. We demand that NICA is an Uber-assumption [11], i.e., its challenges consist of group elements $\mathbf{g}, \mathbf{g}^{f_1(\overrightarrow{z})}, \ldots, \mathbf{g}^{f_{q_1}(\overrightarrow{z})}, \mathbf{g_T}^{g_1(\overrightarrow{z})}, \ldots, \mathbf{g_T}^{g_{q_2}(\overrightarrow{z})}$ where $\overrightarrow{z} \overset{\$}{\leftarrow} \mathbb{Z}_p^t$ has been sampled secretly and uniformly at random by the challenger and $f_1, \ldots, f_{q_1}, g_1, \ldots, g_{q_2} \in \mathbb{Z}_p[Z_1, \ldots, Z_t]$ are publicly known polynomials.

Parametrized Rational VUFs. It is required that vuf is parametrized rational of constant evaluation degree. Formally, this means there are constant-degree polynomials $\sigma, \rho \in \mathbb{Z}_p[V_1, \ldots, V_n, X]$ s.t. we have for each input $x \in \mathbb{Z}_p$ and each verification key vk and corresponding secret key sk

$$\mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}, x) = \mathbf{g_T}^{\frac{\sigma(x, \overrightarrow{v})}{\rho(x, \overrightarrow{v})}}$$

where $\overrightarrow{v}$ denotes the vector of exponents of the group elements of vk.

We are now able to sketch a proof for Theorem 3:

Sketch of Proof, Part 1. Assume that Theorem 3 is false for some parametrized VUF vuf and let $\mathcal{R}$ be a reduction that solves instances of some Uber-assumption NICA when given access to an adversary for the unpredictability of vuf. To show a contradiction we construct a meta-reduction $\mathcal{M}$ that takes the role of a successful adversary in the weak selective unpredictability game with $\mathcal{R}$.

$\mathcal{R}$ is given a challenge $\mathbf{g}, \mathbf{g}^{f_1(\overrightarrow{z})}, \ldots, \mathbf{g}^{f_{q_1}(\overrightarrow{z})}, \mathbf{g_T}^{g_1(\overrightarrow{z})}, \ldots, \mathbf{g_T}^{g_{q_2}(\overrightarrow{z})}$ by the NICA challenger and has to compute some solution from this tuple of group elements while having oracle access to $\mathcal{M}$. Since $\mathcal{R}$ is a generic algorithm, we can apply a hybrid step and change the groups $\mathbb{G}, \mathbb{G}_T$ which encode elements of $\mathbb{Z}_p$ to groups $\mathbb{G}^Z, \mathbb{G}_T^Z$ that encode polynomials of $\mathbb{Z}_p[Z_1, \ldots, Z_t]$ without $\mathcal{R}$ noticing the internal change of groups. Additionally, the NICA challenger will now give the group elements $\mathbf{g}, \mathbf{g}^{f_1(\overrightarrow{Z})}, \ldots, \mathbf{g}^{f_{q_1}(\overrightarrow{Z})}, \mathbf{g_T}^{g_1(\overrightarrow{Z})}, \ldots, \mathbf{g_T}^{g_{q_2}(\overrightarrow{Z})}$ as challenge to $\mathcal{R}$. Further, because of the genericness of $\mathcal{R}$, the exponent of each target group element it outputs must be a polynomial of the form

$$\alpha + \sum_{i=1}^{q_1} \beta_i \cdot f_i(\overrightarrow{Z}) + \sum_{i,j=1} \gamma_{i,j} \cdot f_i(\overrightarrow{Z}) \cdot f_j(\overrightarrow{Z}) + \sum_{i=1}^{q_2} \delta_i \cdot g_i(\overrightarrow{Z}) \tag{4}$$

for scalars $\alpha, \beta_i, \gamma_{i,j}, \delta_i \in \mathbb{Z}_p$. Let W denote the vector space of all polynomials that can be expressed in the above way, i.e., $W = \mathrm{span}_{\mathbb{Z}_p}\{1, (f_i)_i, (f_i \cdot f_j)_{i,j}, (g_i)_i\} \subset \mathbb{Z}_p[Z]$. The space W contains the exponents of all target group elements that can be constructed by generic group operations and pairings from the elements of the NICA challenge. In particular, the exponent of each group element outputted by $\mathcal{R}$ must lie in W.

Now, when $\mathcal{R}$ accesses $\mathcal{M}$ it sends a verification key vk, random inputs $x_0, \ldots, x_Q$, image values $\mathbf{y}_1, \ldots, \mathbf{y}_Q$ and proofs $\pi_1, \ldots, \pi_Q$ to $\mathcal{M}$. To win the unpredictability game, $\mathcal{M}$ needs to return the evaluation $\mathbf{y}_0$ of vuf at x_0 to $\mathcal{R}$. As stated above, the exponents of each group element of vk and of the image values $\mathbf{y}_1, \ldots, \mathbf{y}_Q$ must lie in W. Let $v_1(\overrightarrow{Z}), \ldots, v_n(\overrightarrow{Z}), y_1(\overrightarrow{Z}), \ldots, y_Q(\overrightarrow{Z}) \in W$ be exponents of these group elements. Since $\mathcal{R}$ is generic, $\mathcal{M}$ can extract those polynomials from $\mathcal{R}$ while playing the unpredictability game with $\mathcal{R}$ (we assume in this work that genericness always implies algebraicity, cf. Definition 5). With the help of $\pi_1, \ldots, \pi_Q$ the meta-reduction $\mathcal{M}$ can ensure that for each $i \in [Q]$ the equation

$$\frac{\sigma(x_i, v_1(\overrightarrow{Z}), \ldots, v_n(\overrightarrow{Z}))}{\rho(x_i, v_1(\overrightarrow{Z}), \ldots, v_n(\overrightarrow{Z}))} = y_i(\overrightarrow{Z}) \tag{5}$$

holds.

Sketch of Proof, Part 2. In the first part of the proof, we showed that the fractions $\frac{\sigma(x_i, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x_i, \overrightarrow{v}(\overrightarrow{Z}))}$, $i \in [Q]$, are not only polynomials, but additionally lie in W. This is the point where we can spring our mathematical trap: we can show if all fractions $\frac{\sigma(x_1, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x_1, \overrightarrow{v}(\overrightarrow{Z}))}, \ldots, \frac{\sigma(x_Q, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x_Q, \overrightarrow{v}(\overrightarrow{Z}))}$ lie in W for a large enough number Q then, in fact, the fraction $\frac{\sigma(x, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x, \overrightarrow{v}(\overrightarrow{Z}))}$ must be an element of W for *each* $x \in \mathbb{Z}_p$. In particular, the exponent $\frac{\sigma(x_0, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x_0, \overrightarrow{v}(\overrightarrow{Z}))}$ of $\mathbf{y}_0$ must be of this form and therefore $\mathcal{M}$ can compute the element $\mathbf{y}_0 = \mathbf{g_T}^{\frac{\sigma(x_0, \overrightarrow{v}(\overrightarrow{Z}))}{\rho(x_0, \overrightarrow{v}(\overrightarrow{Z}))}}$ from the group elements of the NICA challenge on its own. Ergo, $\mathcal{M}$ can successfully answer the queries of $\mathcal{R}$ for a large enough number of queries Q which gives rise to a generic PPT NICA solver. A contradiction to the hardness of NICA!

Overview of Section 6. The crucial point in the proof of Theorem 3 is that the fraction $\sigma(x, \overrightarrow{v}(\overrightarrow{Z}))/\rho(x, \overrightarrow{v}(\overrightarrow{Z}))$ lying in W for a large enough number of inputs x implies that this fraction must lie in W for all $x \in \mathbb{Z}_p$. However, proving this statement in general requires a great deal of mathematical tools for Groebner bases and projective algebraic geometry (e.g. the Projective Extension Theorem 7). Therefore, in Section 6.1 we illustrate some of our ideas by first proving the unpredictability of the VUF of Dodis & Yampolskiy [17] and then give the full proof of Theorem 3 in Section 6.3. At the beginning of Section 6, we introduce a technical framework for both proofs and, in Section 6.2, we summarize results in the field of Groebner bases and projective algebraic geometry. We should note that in the actual proofs we avoid using hybrid steps directly. Instead, we introduce a technical tool, which we call *verification equations*, that ensure that Eq. (5) holds for each $i \in [Q]$ without changing the actual groups $\mathbb{G}, \mathbb{G}_T$ in the unpredictability game between $\mathcal{R}$ and $\mathcal{M}$ (Lemma 3).

2.4 Organization of this Work

In Section 3, we introduce notations and preliminaries. In Section 4, we define consecutive verifiable and rational VUFs and show our first result: a consecutive verifiable VUF is rational and its evaluation degree is exponentially bounded by the size of its proofs. In Section 5, we show our second result: Theorem 1 and Theorem 2, which state that the security of rational VUFs cannot be based by an algebraic reduction on the hardness of a NICA, if either the verification key of the VUF or the NICA is univariate. Finally, in Section 6, we introduce the notion of parametrized rational VUFs and prove Theorem 3, which states that the security parametrized rational VUFs cannot be based by a generic reduction on an Uber-assumption.

3 Preliminaries

3.1 Notation

We denote the security parameter by λ. We denote vectors by $\overrightarrow{x}$ and group elements by $\mathbf{a}$. For a matrix M we denote by $m_{i,j}$ the entry in the i-th row and the j-th column. For a finite set X we denote by $x \overset{\$}{\leftarrow} X$ that x is sampled uniformly at random from X.

For a probabilistic algorithm Alg we denote by $y \overset{\$}{\leftarrow} \mathsf{Alg}(x)$ that y is computed by Alg on input x with a uniform random tape. Set further $\mathsf{poly}(\lambda) := \{f : \mathbb{N} \to \mathbb{N} \mid \exists a, b \in \mathbb{N}, \forall n \in \mathbb{N} : \ f(n) \leq a + n^b\}$ and $\mathsf{negl}(\lambda) := \{\varepsilon : \mathbb{N} \to \mathbb{R} \mid \forall c \in \mathbb{N} : \lim_{n\to\infty} n^c \cdot \varepsilon(n) = 0\}$. For any $n \in \mathbb{N}$ we set $[n] := \{1, \ldots, n\}$. We call an algorithm PPT iff it is probabilistic, and its time complexity lies in $\mathsf{poly}(\lambda)$.

3.2 Mathematical Foundations

Definition 1 (Rational Functions). *For a prime p we define the field of **rational functions** over $\mathbb{Z}_p$ in variables $X_1, \ldots, X_n$ by*

$$\mathbb{Z}_p(X_1, \ldots, X_n) := \left\{ \frac{\sigma(X_1, \ldots, X_n)}{\rho(X_1, \ldots, X_n)} \middle| \sigma, \rho \in \mathbb{Z}_p[X_1, \ldots, X_n],\ \rho \neq 0 \right\}.$$

*Given a rational function $f \in \mathbb{Z}_p(X_1, \ldots, X_n)$, the **degree** of f is defined as*

$$\deg(f) := \min\{\max(\deg(\sigma), \deg(\rho)) \mid \sigma, \rho \in \mathbb{Z}_p[X_1, \ldots, X_n], \rho \neq 0, \rho \cdot f = \sigma\}$$

where $\deg(\sigma), \deg(\rho)$ denote the total degrees of the polynomials σ, ρ.

We recall the following helpful lemma:

Lemma 1 (Schwartz-Zippel-Lemma, [47]). *Let $f \in \mathbb{Z}_p[X_1, \ldots, X_n]$ be a non-zero polynomial over $\mathbb{Z}_p$. Denote by $\deg(f)$ the total degree of f. Then*

$$\Pr_{r_1,\ldots,r_n \overset{\$}{\leftarrow} \mathbb{Z}_p} [f(r_1, \ldots, r_n) = 0] \leq \frac{\deg(f)}{p}.$$

3.3 Cryptographic Groups

Definition 2 (Bilinear Group Generator, [26]). *A **bilinear group generator** is a probabilistic polynomial-time algorithm* GrpGen *that takes as input a security parameter λ (in unary) and outputs $\Pi = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \phi(1)) \xleftarrow{\$} \mathsf{GrpGen}(1^\lambda)$ such that the following requirements are satisfied.*

1. *The parameter p is prime and $\log(p) \in \Omega(\lambda)$.*
2. *$\mathbb{G}$ and $\mathbb{G}_T$ as described by $\mathsf{pp}_{\mathbb{G}}$ and $\mathsf{pp}_{\mathbb{G}_T}$ are subsets of $\{0,1\}^*$, defined by algorithmic descriptions of maps $\phi : \mathbb{Z}_p \to \mathbb{G}$ and $\phi_{\mathsf{T}} : \mathbb{Z}_p \to \mathbb{G}_T$.*
3. *$\circ$ and $\circ_{\mathsf{T}}$ are algorithmic descriptions of efficiently computable (in λ) maps $\circ : \mathbb{G} \times \mathbb{G} \to \mathbb{G}$ and $\circ_{\mathsf{T}} : \mathbb{G}_T \times \mathbb{G}_T \to \mathbb{G}_T$, such that*
   (a) $(\mathbb{G}, \circ)$ and $(\mathbb{G}_T, \circ_{\mathsf{T}})$ form abstract groups and
   (b) ϕ is a group isomorphism from $(\mathbb{Z}_p, +)$ to $(\mathbb{G}, \circ)$ and
   (c) ϕ_{T} is a group isomorphism from $(\mathbb{Z}_p, +)$ to $(\mathbb{G}_T, \circ_{\mathsf{T}})$.
4. *e is an algorithmic description of an efficiently computable (in λ) bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$. We require that e is non-degenerate, i.e., $x \neq 0 \implies e(\phi(x), \phi(x)) \neq \phi_{\mathsf{T}}(0)$.*

Remark 4. For simplicity, we only consider symmetric pairings. However, while our upcoming formulation of "consecutive verifiability" is easier to state with symmetric pairings, our results do not depend on symmetry of the pairing.

Definition 3 (Certified Generator, [26]). *We say a bilinear group generator* GrpGen *is **certified**, if there exists a deterministic polynomial-time algorithm* GrpVfy *with the following properties:*

Parameter Validation. *Given a string Π (which may not necessarily be generated by* GrpGen*), algorithm $\mathsf{GrpVfy}(\Pi)$ outputs 1 if and only if Π has the form $\Pi = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \phi(1))$ and all requirements from Definition 2 are satisfied.*

Recognition and Unique Representation of Elements of $\mathbb{G}$ ($\mathbb{G}_T$). *Furthermore, we require that each element in $\mathbb{G}$ ($\mathbb{G}_T$) has a unique representation, which can be efficiently recognized. That is, on input two strings Π and s, $\mathsf{GrpVfy}(\Pi, s)$ outputs 1 if and only if $\mathsf{GrpVfy}(\Pi) = 1$ and it holds that $s = \phi(x)$ ($s = \phi_{\mathsf{T}}(x)$) for some $x \in \mathbb{Z}_p$. Here $\phi : \mathbb{Z}_p \to \mathbb{G}$ ($\phi_{\mathsf{T}} : \mathbb{Z}_p \to \mathbb{G}_T$) denotes the fixed group isomorphism contained in Π to specify the representation of elements of $\mathbb{G}$ (of $\mathbb{G}_T$) (see Definition 2).*

We recall the definition of algebraic algorithms which was first used by [10, 45] in the context of meta-reductions. Our definition of algebraic algorithms is closer to that of [3, 21].

Definition 4 (Algebraic Algorithms [3, 21]). *Let $\mathsf{pp}_{\mathcal{G}} = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ_{\mathbb{G}}, \circ_{\mathbb{G}_T}, e, \phi_{\mathbb{G}}, \phi_{\mathbb{G}_T})$ be as in Definition 2. Let $\mathcal{A}$ be an algorithm that receives as input source group elements $\mathbf{g}_1, \ldots, \mathbf{g}_s \in \mathbb{G}$, target group elements $\mathbf{h}_1, \ldots, \mathbf{h}_t \in \mathbb{G}_T$ and some non-group-element input x.*

*We say that $\mathcal{A}$ is **algebraic** if, whenever $\mathcal{A}$ outputs a group element $\mathbf{y}$, it also outputs one of the following representations: If $\mathbf{y} \in \mathbb{G}$, a vector*

$$\overrightarrow{z} \in \mathbb{Z}_p^s \ \ \textit{s.t.} \ \ \mathbf{y} = \prod_{i=1}^{s} \mathbf{g}_i^{z_i}$$

and if $\mathbf{y} \in \mathbb{G}_T$, a vector and a matrix

$$\overrightarrow{z} \in \mathbb{Z}_p^t, M = (m_{ij})_{i,j=1}^s \in \mathbb{Z}_p^{s \times s} \ \ \textit{s.t.} \ \ \mathbf{y} = \prod_{i=1}^{t} \mathbf{h}_i^{z_i} \cdot \left(\prod_{i,j=1}^{s} e(\mathbf{g}_i, \mathbf{g}_j)^{m_{ij}} \right)$$

Definition 5 (The Generic Group Model [43, 48]). *An algorithm interacting with a group (or pairing group) is called **generic** if it is algebraic in the sense of Definition 4* and *it suffices that the algorithm accesses the group only through an oracle. More concretely, all group elements $\mathbf{g}_i$ that the algorithm receives as input are represented by random strings $\sigma(\mathbf{g}_i)$, called **handles**, and whenever the algorithm wants to compute the product $\mathbf{g}_i \cdot \mathbf{g}_j$ resp. the exponentiation $\mathbf{g}^x$, it passes $(\sigma(\mathbf{g}_i), \sigma(\mathbf{g}_j))$ resp. $(\sigma(\mathbf{g}_i), x)$ to the corresponding group operation oracle, and the oracle returns $\sigma(\mathbf{g}_i \cdot \mathbf{g}_j)$ resp. $\sigma(\mathbf{g}_i^x)$. In a pairing setting the algorithm is given access to a second such group oracle for the target group, as well as a pairing oracle that takes as input two handles $\sigma(\mathbf{g}_i), \sigma(\mathbf{g}_j)$ and outputs $\sigma(e(\mathbf{g}_i, \mathbf{g}_j))$ if both elements $\mathbf{g}_i, \mathbf{g}_j$ are elements of the source group.*

Remark 5. It has been shown recently – despite popular belief – that an algorithm that only interacts with a group by oracles in Shoup's GGM does not need to be algebraic [31, 52]. To circumvent this problem, we require in the definition of generic algorithms explicitly that a generic algorithm is algebraic.

Remark 6. It is not clear how to adapt the notion of a certified group generator (Definition 3) to generic groups. Indeed, in the generic group model, there are no group descriptions as in Definition 2, and instead all algorithms have access to a group via group operation oracles. However, these oracles can be viewed as "implicitly trusted", in the sense that the properties from Definition 2 are always guaranteed. Hence, we will not consider certified (bilinear) group generators in the context of generic groups.

Definition 6 (Non-Interactive Computational Assumptions, NICAs [20]). *A **non-interactive computational assumption** NICA is defined by the following two oracles available to the adversary:*

Setup *Generates a challenge $c \xleftarrow{\$} \mathcal{D}(1^\lambda)$ from a challenge distribution $\mathcal{D}(1^\lambda)$ parameterized over the security parameter λ. Saves an internal state* st*.*

Finalize *On input of a candidate solution s and the internal state* st*, the Finalize subroutine outputs either 1 (indicating that s is a correct solution) or 0 (indicating that s is not a correct solution)*

*We say that an adversary $\mathcal{A}$ (t, ϵ)-**breaks** the assumption if the adversary outputs a correct solution with probability at least $\epsilon(\lambda)$ in time at most $t(\lambda)$. We further say the* *assumption is (t, ϵ)-**hard** if there exists no adversary $\mathcal{A}$ that (t, ϵ)-breaks the assumption. If* NICA *is $(t, \frac{1}{r})$-hard for all $t, r \in \mathsf{poly}(\lambda)$, $r > 0$, we call* NICA ***hard**.*

*For a NICA in a group where the challenge consists of m group elements, we call m the **size** of the NICA. If m is linear in a parameter q, we call* NICA *a q**-type assumption**. If m is constant we call* NICA *a **constant-size assumption**.*

Definition 7 (Univariate Polynomial-Degree Assumptions). *Let $p = p(\lambda)$ be a superpolynomial group order. Let $l_1, l_2, d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$, let $r_1, \ldots, r_{l_1}, t_1, \ldots, t_{l_2} \in \mathbb{Z}_p[S]$ be non-zero polynomials of degree at most d_{NICA}. We say* NICA *is a univariate polynomial-degree assumption, iff it is an $(l_1 + l_2)$-type NICA according to Definition 6 and if its challenge distribution*[9] *is $\mathcal{D}(1^\lambda) \to c = (\mathit{\Pi}, \mathbf{g}^{r_1(s)}, \ldots, \mathbf{g}^{r_{l_1}(s)}, \mathbf{g}^{1/t_1(s)}, \ldots, \mathbf{g}^{1/t_{l_2}(s)})$ where $s \overset{\$}{\leftarrow} \mathbb{Z}_p$ is the secret exponent and $\mathit{\Pi} = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \phi(1)) \overset{\$}{\leftarrow}$ $\mathsf{GrpGen}(1^\lambda)$ is a certified group description.*

Definition 8 (DLog-Hard Assumptions). *Let $l_1, l_2, d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$, let $r_1, \ldots, r_{l_1}$, $t_1, \ldots, t_{l_2} \in \mathbb{Z}_p[S]$ be non-zero polynomials of degree at most d_{NICA}. We say* NICA *is a DLog-hard assumption, iff it is an $(l_1 + l_2)$-type assumption according to Definition 7 and if no polynomial-time algorithm has noticable probability of solving the corresponding DLog problem, i.e., outputting the secret exponent $s \in \mathbb{Z}_p$.*

Remark 7. In particular the *computational* q-DHI assumption (Diffie-Hellman inversion assumption) is a univariate polynomial-degree assumption for $q \in \mathsf{poly}(\lambda)$. The decisional variant is *not* univariate because of the last challenge element.

3.4 Verifiable Unpredictable Functions

Definition 9 (Verifiable Unpredictable Functions, VUFs [38]). *Let* $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ *be a tuple of algorithms of the following form:*

- $\mathsf{Gen}_{\mathsf{vuf}}(1^\lambda)$ *outputs a secret key* sk *and a verification key* vk.
- $\mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}, x)$ *on input a secret key* sk *and $x \in \mathcal{X} = (\mathcal{X}_\lambda)_\lambda$ outputs an image $y \in \mathcal{Y} = (\mathcal{Y}_\lambda)_\lambda$ and a proof π. We assume that the input space $\mathcal{X}_\lambda$ has a superpolynomial cardinality in the security parameter λ.*
- $\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, y, \pi)$ *on input a verification key* vk*, a preimage x, an image y and a proof π outputs a bit $b \in \{0, 1\}$.*

We say that vuf *is a (t, Q, ϵ)-**verifiable unpredictable function** (VUF) if the following holds:*

***Statistical Correctness**. There exists a negligible function $\mu \in \mathsf{negl}(\lambda)$ s.t. for all $\lambda \in \mathbb{N}$ and for all inputs $x \in \mathcal{X}_\lambda$ it holds that*

$$\Pr_{(\mathsf{sk},\mathsf{vk}) \overset{\$}{\leftarrow} \mathsf{Gen}_{\mathsf{vuf}}(1^\lambda)} [\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, y, \pi) = 1 \mid (y, \pi) \leftarrow \mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}, x)] \geq 1 - \mu(\lambda) .$$

[9] For exposition, we assume all group element to be in the source group. Our technique applies as well for assumptions with target group elements.

Unique Provability. *For all $\lambda \in \mathbb{N}$ and all possible* vk *(not necessarily generated by* $\mathsf{Gen}_{\mathsf{vuf}}$*), all $x \in \mathcal{X}_\lambda$, all $y_1, y_2 \in \mathcal{Y}_\lambda$ and all possible proofs π_1, π_2 it holds that*

$$\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, y_1, \pi_1) = 1 \wedge \mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, y_2, \pi_2) = 1 \implies y_1 = y_2$$

Weak Q-Selective Unpredictability [12]. *For any adversary $\mathcal{A}$ running in time at most $t(\lambda)$, we have*

$$\left| \Pr\left[\mathcal{A}(\mathsf{vk}, \overrightarrow{x}, \overrightarrow{\mathbf{y}}, \overrightarrow{\pi}) = \mathbf{y}_0 \;\middle|\; \begin{array}{l} \overrightarrow{x} = (x_0, \ldots, x_Q) \xleftarrow{\$} \mathcal{X}_\lambda^{Q+1} \\ (\mathsf{sk}, \mathsf{vk}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{vrf}}(1^\lambda) \\ (\mathbf{y}_i, \pi_i) \leftarrow \mathsf{Eval}_{\mathsf{vrf}}(\mathsf{sk}, x_i) \\ \overrightarrow{\mathbf{y}} = (\mathbf{y}_1, \ldots, \mathbf{y}_Q) \\ \overrightarrow{\pi} = (\pi_1, \ldots, \pi_Q) \end{array} \right] - \frac{1}{|\mathcal{Y}_\lambda|} \right| \le \epsilon(\lambda) .$$

Remark 8. Our notion of weak selective unpredicability is even weaker than the eponymous notion used by Niehues [44] with a loss of $1/Q$ by guessing the adversary's challenge index and reordering the preimages. However, our notion has the advantage that it is a non-interactive game, in particular, no state has to be transmitted between parts of the adversary $(\mathcal{A}_1, \mathcal{A}_2)$ as in [44].

Remark 9. We note that we do not require *perfect* correctness as for some of the VUFs we consider in this work this property does not hold perfectly (e.g. in the case where $\mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}, x)$ is undefined for a small number of $x \in \mathcal{X}$ for some secret key sk).

Remark 10. We consider pairing-based VUFs where $y \in (\mathbb{G} \cup \mathbb{G}_T)$ and $\pi \in (\mathbb{G} \cup \mathbb{G}_T)^*$. W.l.o.g. we assume that a VUF's image is an element of the target group, i.e., $\mathcal{Y} = \mathbb{G}_T$. Otherwise, we can modify the VUF by appending the original (source group) image $\mathbf{y}_{\mathsf{S}} \in \mathbb{G}$ to the proof elements, and set the new image as $\mathbf{y}_{\mathsf{T}} := e(\mathbf{g}_{\mathsf{S}}, \mathbf{y}_{\mathsf{S}})$ where $\mathbf{g}_{\mathsf{S}}$ is a designated generator of the source group in the verification key. Obviously, the unpredictability of the former VUF can be reduced to the unpredictability of latter, without any loss.

Definition 10 (Verifiable Random Functions, VRFs [38]). *Let* $\mathsf{vrf} = (\mathsf{Gen}_{\mathsf{vrf}}, \mathsf{Eval}_{\mathsf{vrf}}, \mathsf{Verify}_{\mathsf{vrf}})$ *be a VUF according to Definition 9. We say that* vrf *is a (t, ϵ)-**verifiable random function** (VRF) if the following*[10] *holds:*

0-***Adaptive Pseudorandomness***. *For any adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ running in time at most $t(\lambda)$, we have*

$$\left| \Pr\left[\mathcal{A}_2(\mathsf{st}, \mathbf{y}_b) = b \;\middle|\; \begin{array}{l} (\mathsf{sk}, \mathsf{vk}) \xleftarrow{\$} \mathsf{Gen}_{\mathsf{vrf}}(1^\lambda) \\ (x, \mathsf{st}) \xleftarrow{\$} \mathcal{A}_1(1^\lambda, \mathsf{vk}) \\ b \xleftarrow{\$} \{0, 1\} \\ (\mathbf{y}_0, \pi_0) \leftarrow \mathsf{Eval}_{\mathsf{vrf}}(\mathsf{sk}, x) \\ \mathbf{y}_1 \xleftarrow{\$} \mathbb{G}_T \end{array} \right] - \frac{1}{2} \right| \le \epsilon(\lambda) .$$

[10] To keep the definitions minimal, we choose to only present the 0-adaptive pseudorandomness property since it is the security notion considered in our results.

3.5 Reductions

Definition 11. *For a VUF* vuf *and a NICA* NICA*, we say a Turing machine* $\mathcal{B}$ *is a* $(t_{\mathcal{B}}, \epsilon_{\mathcal{B}}, r, Q, \epsilon_{\mathcal{A}})$*-**reduction** from breaking* NICA *to breaking the weak selective unpredictability of* vuf*, if for any* $\mathcal{A}$ *that* $(t_{\mathcal{A}}, Q, \epsilon_{\mathcal{A}})$*-breaks the weak selective unpredicability of* vuf*, the TM* $\mathcal{B}^{\mathcal{A}}$ $(t_{\mathcal{B}} + rt_{\mathcal{A}}, \epsilon_{\mathcal{B}})$*-breaks* NICA *making at most r oracle queries*[11] *to* $\mathcal{A}$*.*

4 Proof Size

4.1 Classes of VUFs over Pairing-Friendly Groups

In the following, we introduce the class of VUFs that we want to discuss. Informally speaking, we consider VUFs whose verification algorithm only verifies group membership and pairing equations over the proof, evaluation value, and verification key. We further require that the verification algorithm is *consecutive*, i.e., it first verifies the first element of the proof, then the second, then the third, and so on and at the end of its execution it verifies that the evaluation value is correct. This class of VUFs covers many existing VUFs, we refer to Fig. 1 for an overview of which VUFs are consecutively verifiable.

In this section, we want to show that the evaluation function of VUFs that have such a natural verification algorithm can be expressed as a target group element where the exponent is a rational function in the discrete logarithms of the verification key element and that, informally speaking, the degree of the rational function can be bounded as exponential in the size of the proof. We begin by giving a formal definition of what we consider a set of pairing equations.

Definition 12 (Pairing Equations). *Let* $E \subset \mathbb{Z}_p[X_1, \ldots, X_m]$*. We call* E *a set of **pairing equations** for a pairing group* $\mathcal{G}$ *with public parameters* $\Pi = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \phi(1)) \xleftarrow{\$} \mathsf{GrpGen}(1^\lambda)$ *over variables* $\overrightarrow{X} = X_1, \ldots, X_m$ *with target indicator*[12] *set* $T \subset \{1, \ldots, m\}$ *if the following hold:*

1. $\max_{f \in E}(\deg f) \leq 2$*,*
2. *for all* $i \in T$ *and* $f \in E$ *it holds that if* X_i *appears in a monomial* m *of* f*, then* $m = c \cdot X_i$ *for some* $c \in \mathbb{Z}_p$*.*

We describe the evaluation *of a finite set of pairing equations* E *on input* $\mathbf{x}_1, \ldots, \mathbf{x}_m$ *as follows:*

- *We check that the input is a set of group elements* $(\mathbf{x}_1, \ldots \mathbf{x}_m)$*, i.e.,* $\mathbf{x}_i \in \mathbb{G}$ *or* $\mathbf{x}_i \in \mathbb{G}_T$ *for all i, and output* $\perp$ *if otherwise.*
- *For each* $i \in [m]$*, we check if* $i \in T \iff \mathbf{x}_i \in \mathbb{G}_T$ *and output* $\perp$ *if otherwise.*
- *For* $f = \left(\sum_{m \in M_f} m\right) \in E$ *where* M_f *is the set of monomials of* f*, we compute* $\mathbf{f}(\overrightarrow{\mathbf{x}}) := \prod_{m \in M_f} \mathbf{m}(\overrightarrow{\mathbf{x}})$ *where* $\mathbf{m}(\overrightarrow{\mathbf{x}})$ *are computed as follows:*

[11] Because our weak selective unpredictability is a non-interactive game, there are no concurrency issues.

[12] This set indicates which verification key elements are in the target group. Hence, their exponents should only occur linearly, while source group exponents can occur quadratically.

  - *if* $m = c \cdot X_i \cdot X_j$ *for some* $i, j \notin T$ *and* $c \in \mathbb{Z}_p$*, compute* $\mathbf{m}(\overrightarrow{\mathbf{x}}) := e(\mathbf{x}_i, \mathbf{x}_j)^c$*,*
  - *if* $m = c \cdot X_i$ *for some* $i \notin T$ *and some* $c \in \mathbb{Z}_p$ *and if* $\mathbf{x}_i \in \mathbb{G}$*, compute* $\mathbf{m}(\overrightarrow{\mathbf{x}}) := e(\mathbf{x}_i, \mathbf{g})^c$ *where* $\mathbf{g} = \phi(1)$ *is the fixed generator of* $\mathbb{G}$ *as given in the group parameters* Π*. If* $i \in T$ *and* $\mathbf{x}_i \in \mathbb{G}_T$ *compute* $\mathbf{m}(\overrightarrow{\mathbf{x}}) := \mathbf{x}_i^c$*,*
  - *if* $m = c$ *for* $c \in \mathbb{Z}_p$*, compute* $\mathbf{m}(\overrightarrow{\mathbf{x}}) := e(\mathbf{g}, \mathbf{g})^c$*.*
- *We denote by* $E(\overrightarrow{\mathbf{x}})$ *the function that outputs* 1 *if for all* $f \in E$ *it holds that* $\mathbf{f}(\overrightarrow{\mathbf{x}}) = e(\mathbf{g}, \mathbf{g})^0$ *(if* $E = \emptyset$ *this always holds) and otherwise outputs* 0*.*

In the following we describe our class of VUFs that have a consecutive verification algorithm.

Definition 13 (Consecutively Verifiable Pairing-Based VUFs). *We say a VUF* $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ *with input space* $\mathcal{X}$ *is a* ***consecutively verifiable pairing-based VUF*** *if the following hold:*

1. $\mathsf{Gen}_{\mathsf{vuf}}$ *takes as input* 1^λ*. It samples group parameters* $\Pi = (p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \mathbf{g} := \phi(1)) \xleftarrow{\$} \mathsf{GrpGen}(1^\lambda)$ *and outputs a verification key* $\mathsf{vk} = (\Pi, \overrightarrow{\mathbf{v}})$ *such that* $\overrightarrow{\mathbf{v}}$ *consists of elements of* $\mathbb{G}$ *and* $\mathbb{G}_T$ *(plus a secret key* sk *for which we make no further constraints).*
2. *All function values* $\mathbf{y}$ *consist of values in* $\mathbb{G}_T$*.*
3. *All proofs consist of* κ *values in* $\mathbb{G} \cup \mathbb{G}_T$*.*
4. *For all* $x \in \mathcal{X}$ *and all* $i \in [\kappa + 1]$*, there exists a set* $E_{i,x}$ *of pairing equations that can be efficiently derived from* x *and the description of* vuf*. We require that* $E_{i,x} \subset \mathbb{Z}_p[V_1, \dots, V_n, P_1, \dots, P_i]$ *such that there is at least one polynomial of the form* $a_{i,x} \cdot P_i + b_{i,x} \in E_{i,x}$ *where* $a_{i,x}, b_{i,x} \in \mathbb{Z}_p[V_1, \dots, V_n, P_1, \dots, P_{i-1}]$*. (We note that since the set* $E_{i,x}$ *consists of pairing equations it holds that* $a_{i,x}$ *has degree at most* 1 *and* $b_{i,x}$ *has degree at most* 2*.)*
5. *We require that* $\mathsf{Verify}_{\mathsf{vuf}}$ *on input* $(\mathsf{vk} = (\Pi, \overrightarrow{\mathbf{v}}), x, \mathbf{y} =: \pi_{\kappa+1}, \overrightarrow{\pi})$ *outputs* 1 *if and only if the following hold:* $\mathsf{GrpVfy}(\Pi) = 1$*, all* $\mathbf{v}_i$*, for* $i \in [n]$*, and all* π_i*, for* $i \in [\kappa + 1]$*, are valid group elements w.r.t.* Π*, and for all* $i \in [\kappa + 1]$ *we have* $E_{i,x}(\overrightarrow{\mathbf{v}}, \pi_1, \dots, \pi_i) = 1$*.*
6. *We further require that the ideal* $(E_{1,x}, \dots, E_{\kappa+1,x}, a_{1,x} \cdot \dots \cdot a_{\kappa+1,x})$ *(which is generated by the elements of* $E_{1,x}, \dots, E_{\kappa+1,x}$ *and the polynomial* $a_{1,x} \cdot \dots \cdot a_{\kappa+1,x}$*) contains the constant polynomial 1 (i.e.,* $(E_{1,x}, \dots, E_{\kappa+1,x}, a_{1,x} \cdot \dots \cdot a_{\kappa+1,x}) = \mathbb{Z}_p[V_1, \dots, V_k, P_1, \dots, P_{\kappa+1}]$*).*

Requirement 4 will be useful in Lemma 2, as it basically means there needs to be at least one equation that contains the current proof element as a linear factor only. This yields in particular that the proof element in question is not a (non-unique) square root of other elements. The last requirement on a consecutively verifiable pairing-based VUF might seem odd, however, as we will see later, it makes sure that there is no tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ s.t. any of the a_i can evaluate to zero on the exponents of $(\mathsf{vk}, x, \mathbf{y}, \pi)$.

We now define the class of VUFs that evaluate a rational function in the exponent. We will show later that a VUF that fulfills Definition 13 and where the number of group elements in the proof is in $O(\log(\lambda))$ also fulfills Definition 14.

Definition 14 (Rational VUFs). *Let* $d, n \in \mathsf{poly}(\lambda)$*. We say that a VUF* $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ *is* ***rational*** *of* ***evaluation degree*** d *with* $n = n_{\mathsf{S}} + n_{\mathsf{T}}$ *verification key elements, if the verification key is of the form* $\mathsf{vk} = (\Pi, \overrightarrow{\mathbf{v}})$ *where* $\Pi :=$

$(p, \mathsf{pp}_{\mathbb{G}}, \mathsf{pp}_{\mathbb{G}_T}, \circ, \circ_{\mathsf{T}}, e, \mathbf{g} = \phi(1)) \xleftarrow{\$} \mathsf{GrpGen}(1^\lambda)$ is a certified group description according to Definition 3, and $\overrightarrow{\mathbf{v}} := (\mathbf{g}^{v_{\mathsf{S},1}}, \ldots, \mathbf{g}^{v_{\mathsf{S},n_\mathsf{S}}}, e(\mathbf{g},\mathbf{g})^{v_{\mathsf{T},1}}, \ldots, e(\mathbf{g},\mathbf{g})^{v_{\mathsf{T},n_\mathsf{T}}}) \in \mathbb{G}^{n_\mathsf{S}} \times \mathbb{G}_T^{n_\mathsf{T}}$.

Further, we require for a rational VUF of evaluation degree d that for each $x \in \mathcal{X}$ there are coprime polynomials $\sigma_x, \rho_x \in \mathbb{Z}_p[V_1, \ldots, V_n]$ of total degree at most d s.t. we have for all vk, all π and all $\mathbf{y} \in \mathbb{G}_T$

$$\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, \mathbf{y}, \pi) = 1 \implies \rho_x(v_1, \ldots, v_n) \neq 0 \textit{ and } \mathbf{y} = e(\mathbf{g},\mathbf{g})^{\frac{\sigma_x(v_1,\ldots,v_n)}{\rho_x(v_1,\ldots,v_n)}} \quad (6)$$

where $(v_1, \ldots, v_n) = (v_{\mathsf{S},1}, \ldots, v_{\mathsf{S},n_\mathsf{S}}, v_{\mathsf{T},1}, \ldots, v_{\mathsf{T},n_\mathsf{T}})$ are the exponents of vk.

We require that – given x and a description of vuf – one can efficiently compute descriptions of σ_x and ρ_x, e.g. as coefficient vectors.

Definition 15 (Rational Univariate VUFs). *Let $d, n, d_f \in \mathsf{poly}(\lambda)$ and let $f_1, \ldots, f_n : \mathbb{Z}_p \to \mathbb{Z}_p$ be n efficiently computable polynomials of degree at most d_f. Let $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ be a rational VUF **evaluation degree** d with $n = n_\mathsf{S} + n_\mathsf{T}$ verification key elements as in Definition 14. We say vuf is a rational **univariate** VUF of **internal degree** d_f relative to $f_1, \ldots, f_n$, iff for all vk, all $x \in \mathcal{X}$, all π and all $\mathbf{y} \in \mathbb{G}_T$ a successful verification $\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x, \mathbf{y}, \pi) = 1$ implies the existence of an "effective secret key" s, i.e.,*

$$\exists s \in \mathbb{Z}_p \textit{ s.t. } \overrightarrow{\mathbf{v}} = (\mathbf{g}^{f_1(s)}, \ldots, \mathbf{g}^{f_{n_\mathsf{S}}(s)}, e(\mathbf{g},\mathbf{g})^{f_{n_\mathsf{S}+1}(s)}, \ldots, e(\mathbf{g},\mathbf{g})^{f_n(s)})\,, \quad (7)$$

thus $\mathbf{y} = e(\mathbf{g},\mathbf{g})^{\frac{\sigma_x(f_1(s),\ldots,f_n(s))}{\rho_x(f_1(s),\ldots,f_n(s))}} = \mathbf{g}^{\widetilde{\sigma}_x(s)/\widetilde{\rho}_x(s)}$ where σ_x and ρ_x are defined in Definition 14, and $\widetilde{\sigma}_x(s) = \sigma_x(f_1(s), \ldots, f_n(s))$ and $\widetilde{\rho}_x(s) = \rho_x(f_1(s), \ldots, f_n(s))$. Note that $\deg(\widetilde{\sigma}_x), \deg(\widetilde{\rho}_x) \leq d \cdot d_f$.

Remark 11. In particular, the popular VRF of Dodis & Yampolskiy [17] is a rational univariate VUF with $n = d = d_f = 1$ (if extended by a certified group description).

4.2 From Consecutively Verifiable Pairing-Based VUFs to Rational VUFs

We now turn to proving that the evaluation outputs of consecutively verifiable pairing-based VUFs can be expressed through rational functions in the exponents.

Lemma 2. *Let $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ be a pairing-based consecutively verifiable VUF with proofs of size κ and a verification key of size n.*

Then, vuf is a rational VUF of evaluation degree at most $4^{\kappa+1}$ over n variables.

Proof. Fix $x \in \mathcal{X}$. We want to show inductively that for each $k \in [\kappa + 1]$ there exist polynomials $\sigma_{x,\pi_k}, \rho_{x,\pi_k} \in \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots, P_{k-1}]$ such that $\deg(\sigma_{x,\pi_k}) \leq 1 + \sum_{i=0}^{k-1} 4^i$, $\deg(\rho_{x,\pi_k}) \leq 4^{k-1}$ and such that for each tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ accepted by $\mathsf{Verify}_{\mathsf{vuf}}$ we have:

$$\pi_k = \mathbf{g}^{\frac{\sigma_{x,\pi_k}(\overrightarrow{\vartheta})}{\rho_{x,\pi_k}(\overrightarrow{\vartheta})}} \text{ or } \pi_k = e(\mathbf{g},\mathbf{g})^{\frac{\sigma_{x,\pi_k}(\overrightarrow{\vartheta})}{\rho_{x,\pi_k}(\overrightarrow{\vartheta})}}$$

and $\rho_{x,\pi_k}(\overrightarrow{\vartheta}) \neq 0$. (Recall that we denote $\mathbf{y}$ by $\pi_{\kappa+1}$).

We first prove a useful statement.

Claim 1. As we required that $1 \in (\bigcup_{i=1}^{\kappa+1} E_{i,x}, a_{1,x} \cdots a_{\kappa+1,x})$ (i.e., $(\bigcup_{i=1}^{\kappa+1} E_{i,x}, a_{1,x} \cdot \ldots \cdot a_{\kappa+1,x}) = \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots, P_\kappa, Y]$), no $a_{i,x}$ has a root that is common between all $E_{1,x}, \ldots, E_{\kappa+1,x}$.

Proof. In fact, if there were some point $(\overrightarrow{v}, \overrightarrow{p}, y) \in \mathbb{Z}_p^{n+\kappa+1}$ and one $j \in [\kappa+1]$ such that $a_{j,x}$ and each $f \in \bigcup_{i=1}^{\kappa+1} E_{i,x}$ were to vanish on $(\overrightarrow{v}, \overrightarrow{p}, y)$, then the constant polynomial $1 \in (\bigcup_{i=1}^{\kappa+1} E_{i,x}, a_{1,x} \cdot \ldots \cdot a_{\kappa+1,x})$ which can be written as $1 = \sum_{i=1}^{\kappa+1} \sum_{f \in E_{i,x}} h_f \cdot f + h' \cdot a_{j,x}$ (for fitting $h_f, h' \in \mathbb{Z}_p[V_1, \ldots V_n, P_1, \ldots, P_\kappa, Y]$) would need to vanish on $(\overrightarrow{v}, \overrightarrow{p}, y)$. Clearly, a contradiction!

We now turn to the inductive proof. In the following we denote by $\overrightarrow{v} := \mathsf{dlog}(\overrightarrow{\mathbf{v}})$ and $\overrightarrow{p} := \mathsf{dlog}(\pi)$ where the discrete logarithms are component-wise. First, we show that our induction hypothesis holds for $k = 1$.

By Item 4 in Definition 13, we can find $h_{1,x} = a_{1,x} \cdot P_1 + b_{1,x} \in E_{1,x}$ where P_1 is the formal variable representing the exponent p_1 of π_1. By Claim 1 it holds that $a_{1,x}(\overrightarrow{v}) \neq 0$ if $(\mathsf{vk}, x, \mathbf{y}, \pi)$ is accepted by $\mathsf{Verify}_{\mathsf{vuf}}$. We can therefore express p_1 as a fraction $p_1 = -\frac{b_{1,x}(\overrightarrow{v})}{a_{1,x}(\overrightarrow{v})}$. We define $\sigma_{x,\pi_1} := -b_{1,x}$ and $\rho_{x,\pi_1} := a_{1,x}$. We further note that $\deg(\sigma_{x,\pi_1}) \leq 2 = 1+1$ and $\deg(\rho_{x,\pi_1}) \leq 1 = 4^0 = 4^{1-1}$ Setting the polynomials $\sigma_{x,\pi_1} := -b_{1,x}$ and $\rho_{x,\pi_1} := a_{1,x}$ yields the hypothesis for π_1, where $\rho_{x,\pi_1}(\overrightarrow{v})$ never vanishes on accepted verification keys vk because of Claim 1.

We now turn to the inductive step. Fix $k \in \{2, \ldots, \kappa\}$ and assume that for all $i < k$ it holds that there exist polynomials $\sigma_{x,\pi_i}, \rho_{x,\pi_i} \in \mathbb{Z}_p[V_1, \ldots, V_n, P_1, \ldots, P_{i-1}]$ of degrees $\deg(\sigma_{x,\pi_i}) \leq 1 + \sum_{j=0}^{i-1} 4^j$, $\deg(\rho_{x,\pi_i}) \leq 4^{i-1}$ s.t. $p_i = \frac{\sigma_{x,\pi_i}(\overrightarrow{v})}{\rho_{x,\pi_i}(\overrightarrow{v})}$ and $\rho_{x,\pi_i}(\overrightarrow{v}) \neq 0$ for the exponents of tuples accepted by $\mathsf{Verify}_{\mathsf{vuf}}$.

We again consider the polynomial $h_{k,x} = a_{k,x} \cdot P_k + b_{k,x}$ from $E_{x,k}$ (which exists due to Item 4 in Definition 13). We note that as $h_{k,x}$ has degree at most two, $a_{k,x}$ has degree at most 1 in the variables $\overrightarrow{V}, P_1, \ldots P_{k-1}$. If $(\mathsf{vk}, x, \mathbf{y}, \pi)$ is accepted by $\mathsf{Verify}_{\mathsf{vuf}}$, we know that $a_{k,x}(\overrightarrow{v}, p_1, \ldots, p_{k-1}) \neq 0$ by Claim 1.

We now want to plug in the induction hypothesis into $a_{k,x}$ and $b_{k,x}$. Knowing from the induction hypothesis that the exponents of π_i with $i = 1, \ldots, k-1$ can all be expressed as $\frac{\sigma_{x,\pi_i}(\overrightarrow{V})}{\rho_{x,\pi_i}(\overrightarrow{V})}$ where σ_{x,π_i} and ρ_{x,π_i} have degrees at most $1+\sum_{j=0}^{j-1} 4^i$ and 4^{i-1}, respectively, we obtain that there exist alternative rational functions $\overline{a_{k,x}}, \overline{b_{k,x}} \in \mathbb{Z}_p(\overrightarrow{V})$

$$\overline{a_{k,x}}(\overrightarrow{V}) := a_{k,x}\left(\overrightarrow{V}, \frac{\sigma_{x,\pi_1}(\overrightarrow{V})}{\rho_{x,\pi_1}(\overrightarrow{V})}, \ldots, \frac{\sigma_{x,\pi_{k-1}}(\overrightarrow{V})}{\rho_{x,\pi_{k-1}}(\overrightarrow{V})}\right),$$

$$\overline{b_{k,x}}(\overrightarrow{V}) := b_{k,x}\left(\overrightarrow{V}, \frac{\sigma_{x,\pi_1}(\overrightarrow{V})}{\rho_{x,\pi_1}(\overrightarrow{V})}, \ldots, \frac{\sigma_{x,\pi_{k-1}}(\overrightarrow{V})}{\rho_{x,\pi_{k-1}}(\overrightarrow{V})}\right)$$

which coincide with $a_{k,x}, b_{k,x}$ on the exponents of each tuple $(\mathsf{vk}, x, \pi, \mathbf{y})$ accepted by $\mathsf{Verify}_{\mathsf{vuf}}$. In fact, $\overline{a_{k,x}}$ and $\overline{b_{k,x}}$ are well-defined for input $\overrightarrow{v}$, if there exist values $\overrightarrow{\pi}, \mathbf{y}$ for which $\mathsf{Verify}_{\mathsf{vuf}}(\overrightarrow{\mathsf{vk}}, x, \mathbf{y}, \overrightarrow{\pi}) = 1$ as our induction hypothesis states that

$\rho_{x,\pi_1}(\overrightarrow{v}), \dots, \rho_{x,\pi_{k-1}}(\overrightarrow{v})$ are non-zero on that input. Further, we have for those inputs, $\overline{a_{k,x}}(\overrightarrow{v}) = a_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1})$ and $\overline{b_{k,x}}(\overrightarrow{v}) = b_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1})$ as our induction hypothesis states $p_i = \frac{\sigma_{x,\pi_i}(\overrightarrow{v})}{\rho_{x,\pi_i}(\overrightarrow{v})}$ for $i < k$.

Since $h_{k,x}(\overrightarrow{v}, p_1, \dots, p_k) = a_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1}) \cdot p_k + b_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1})$ $= 0$ holds for all valid $(\overrightarrow{v}, p_1, \dots, p_k)$, we have the equation

$$p_k = \frac{-b_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1})}{a_{k,x}(\overrightarrow{v}, p_1, \dots, p_{k-1})} = \frac{-\overline{b_{k,x}}(\overrightarrow{v})}{\overline{a_{k,x}}(\overrightarrow{v})} = \frac{-\prod_{j=1}^{k-1} \rho_{x,\pi_j}(\overrightarrow{v})^2 \cdot \overline{b_{k,x}}(\overrightarrow{v})}{\prod_{j=1}^{k-1} \rho_{x,\pi_j}(\overrightarrow{v})^2 \cdot \overline{a_{k,x}}(\overrightarrow{v})}.$$

Since each fraction $\frac{\sigma_{x,\pi_1}(\overrightarrow{V})}{\rho_{x,\pi_1}(\overrightarrow{V})}, \dots, \frac{\sigma_{x,\pi_{k-1}}(\overrightarrow{V})}{\rho_{x,\pi_{k-1}}(\overrightarrow{V})}$ appears at most quadratically in $\overline{a_{k,x}}$ and $\overline{b_{k,x}}$, the functions

$$\sigma_{x,\pi_k}(\overrightarrow{V}) \coloneqq -\prod_{j=1}^{k-1} \rho_{x,\pi_j}(\overrightarrow{V})^2 \cdot \overline{b_{k,x}}(\overrightarrow{V}),$$
$$\rho_{x,\pi_k}(\overrightarrow{V}) \coloneqq \prod_{j=1}^{k-1} \rho_{x,\pi_j}(\overrightarrow{V})^2 \cdot \overline{a_{k,x}}(\overrightarrow{V})$$

are indeed polynomials in $\mathbb{Z}_p[\overrightarrow{V}]$ such that we have for the exponents of each tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ accepted by $\mathsf{Verify}_{\mathsf{vuf}}$

$$p_k = \frac{\sigma_{x,\pi_k}(\overrightarrow{v})}{\rho_{x,\pi_k}(\overrightarrow{v})}.$$

Further, $\rho_{x,\pi_k}(\overrightarrow{v}) \neq 0$ since $\overline{a_{k,x}}(\overrightarrow{v}) \neq 0$ and our induction hypothesis stated that each $\rho_{x,\pi_j}(\overrightarrow{v})$ does not vanish.

We now want to argue that $\deg(\sigma_{x,\pi_k}) \leq 1 + \sum_{i=0}^{k-1} 4^i$ and $\deg(\rho_{x,\pi_k}) \leq 4^{k-1}$. Let σ^* be the polynomial σ_{x,π_j} of maximal degree for some $j \in \{1, \dots, k-1\}$. Then, it holds that

$$\deg(\rho_{x,\pi_k}) \overset{(*)}{\leq} \deg(\sigma^*) + 2\deg\left(\prod_{i=1}^{k-1} \rho_{x,\pi_i}\right) \overset{I.H.}{\leq} 1 + \sum_{i=0}^{k-2} 4^i + 2 \cdot \sum_{i=1}^{k-1} 4^{i-1}$$
$$= 1 + 3 \cdot \sum_{i=0}^{k-2} 4^i = 4^{k-1}$$

where the last equality is easily verified by induction and in $(*)$ we used that $\rho_{x,\pi_k} = \overline{a_{k,x}} \cdot \prod_{i=1}^{k} (\rho_{x,\pi_i})^2$ and σ^* appears at most linearly in the numerator of $\overline{a_{x,\pi_k}}$ and thus the highest degree that can appear in the numerators of $\overline{a_{x,\pi_k}}$ can be $\deg(\sigma^*)$.

For σ_{x,π_k} we compute the following

$$\deg(\sigma_{x,\pi_j}) \overset{(*)}{\leq} 2 \cdot \deg(\sigma^*) + 2\deg\left(\prod_{i=1}^{k-1} \rho_{x,\pi_i}\right) \overset{I.H.}{\leq} 2 + 2 \cdot \sum_{i=0}^{k-2} 4^i + 2 \cdot \sum_{i=0}^{k-2} 4^{i-1}$$

$$= 2 + 4 \cdot \sum_{i=0}^{k-2} 4^i = 1 + \sum_{i=0}^{k-1} 4^i$$

where in $(*)$ we used that $\sigma_{x,\pi_k} = \overline{b_{k,x}} \cdot \prod_{i=1}^{k} (\rho_{x,\pi_i})^2$ and σ^* appears at most squared in the numerators $\overline{b_{k,x}}$ and thus the highest degree of the numerators in $\overline{b_{k,x}}$ is $2 \cdot \deg(\sigma^*)$.

We note here that since the sets $E_{i,x}$ can be efficiently derived from x and the description of vuf, and the proof is bounded in size, and the replacement steps used in this proof, the description of σ_x and ρ_x can also be efficiently derived from x and the description of vuf.

5 Algebraic Attacks on Rational VUFs

In this section we prove that the unpredictability of rational univariate VUFs cannot be based algebraically on some non-interactive computational assumptions. To this end, for any algebraic reduction from the NICA to the unpredictability of the VUF, we give a meta-reduction that internally runs the reduction and supplies it with an adversary for the unpredictability of the VUF. This meta-reduction finds a non-zero, low-degree, univariate *target polynomial* that contains the reduction's *effective* secret key as a root. Because the target polynomial has low (polynomial) degree and is non-zero, the meta-reduction can simply factor it and test each of its polynomially many roots against the reduction's verification key. Using the previously obtained secret key the meta-reduction can predict the reduction's challenge image.

Theorem 1. *Let p be a superpolynomial group order. Let NICA be a non-interactive computational assumption of size $q \in \mathsf{poly}(\lambda)$. Let $n, d, d_f \in \mathsf{poly}(\lambda)$ and let $f_1, \ldots, f_n \in \mathbb{Z}_p[S]$ be some polynomials of degree at most d_f. Let vuf be a rational univariate VUF of evaluation degree d and internal degree d_f over n variables relative to the polynomials $f_1, \ldots, f_n$.*

If there exists an algebraic $(t_{\mathcal{B}}, \epsilon_{\mathcal{B}}, r, Q, 1/(Q+1))$-reduction $\mathcal{B}$ from NICA to the weak Q-selective unpredictability of vuf s.t. $Q \geq q^2 + 1$ and $r \in \mathsf{poly}(\lambda)$, then there exists an adversary $\mathcal{M}$ that $(t_{\mathcal{M}}, \epsilon_{\mathcal{M}})$-breaks NICA with $\epsilon_{\mathcal{M}} \geq \epsilon_{\mathcal{B}} - 2^{-\lambda}$ and $t_{\mathcal{M}} \leq t_{\mathcal{B}} + \mathsf{poly}(\lambda)$.

Proof. We describe the adversary $\mathcal{M}$ as a meta-reduction [14] that takes inputs $\overrightarrow{\mathbf{c}} = (\overrightarrow{\mathbf{c}}_{\mathsf{S}}, \overrightarrow{\mathbf{c}}_{\mathsf{T}}) \in \mathbb{G}^{q_{\mathsf{S}}} \times \mathbb{G}_T^{q_{\mathsf{T}}}$ from NICA, forwards it to the reduction $\mathcal{B}$ and simulates an adversary for the weak selective unpredictability that is indistinguishable from a computationally unbounded adversary that does not get to see the algebraic representation of group elements. Let us first describe the unbounded adversary with success probability at least $1/(Q+1)$. The adversary obtains as input the verification key $\mathsf{vk} = (\mathit{\Pi}, \overrightarrow{\mathbf{v}})$, the challenge preimage $x_0 \in \mathcal{X}_\lambda$, the evaluation preimages $x_1, \ldots, x_Q \in \mathcal{X}_\lambda$, its corresponding images $\mathbf{y}_1, \ldots, \mathbf{y}_Q \in \mathbb{G}_T$ and proofs $\pi_1, \ldots, \pi_Q$. If the group validation fails or $\mathsf{Verify}_{\mathsf{vrf}}(\mathsf{vk}, x_\ell, \mathbf{y}_\ell, \pi_\ell) = 0$ for any $\ell \in \{1, \ldots, Q\}$ the adversary aborts. For each $\ell \in \{0, \ldots, Q\}$ the adversary computes the rational function $\zeta_\ell := \frac{\widetilde{\sigma}_{x_\ell}}{\widetilde{\rho}_{x_\ell}} \in \mathbb{Z}_p(V)$ where $\widetilde{\sigma}_{x_\ell}$ and $\widetilde{\rho}_{x_\ell}$ are as in Definition 15.

If the rational functions $\zeta_0, \dots, \zeta_Q$ are linearly dependent but $\zeta_0 \notin \mathsf{span}(\zeta_1, \dots, \zeta_Q) \subset \mathbb{Z}_p(V)$, then the adversary aborts. If $x_0, \dots, x_Q$ are uniform, i.e., honestly generated, then this happens with probability[13] at most $1 - 1/(Q+1)$. Otherwise, the adversary bruteforces the correct image for the challenge x_0.

Now, we describe the meta-reduction. It forwards its input to the reduction; the NICA challenge has the form $\overrightarrow{\mathbf{c}} = (\mathbf{c}_{\mathsf{S},1}, \dots, \mathbf{c}_{\mathsf{S},q_\mathsf{S}}, \mathbf{c}_{\mathsf{T},1}, \dots, \mathbf{c}_{\mathsf{T},q_\mathsf{T}})$ where $q = q_\mathsf{S} + q_\mathsf{T}$. Because the reduction is algebraic, whenever it outputs a group element $\mathbf{y} \in \mathbb{G}_T$ it must also provide a representation $(\overrightarrow{z}, M = (m_{i,j})_{i,j \in [q_\mathsf{S}]}) \in \mathbb{Z}_p^{q_\mathsf{T}} \times \mathbb{Z}_p^{q_\mathsf{S} \times q_\mathsf{S}}$ w.r.t. the NICA challenge elements s.t.

$$\mathbf{y} = \prod_{i=1}^{q_\mathsf{T}} \mathbf{c}_{\mathsf{T},i}^{z_i} \cdot \prod_{i,j=1}^{q_\mathsf{S}} e(\mathbf{c}_{\mathsf{S},i}, \mathbf{c}_{\mathsf{S},j})^{m_{i,j}} \tag{8}$$

As such, the representation $(\overrightarrow{z}, M)$ gives a concise way of writing an element relative to the NICA challenge.

The meta-reduction simulates an unbounded adversary as follows: It obtains from the reduction as input the verification key vk, the challenge preimage $x_0 \in \mathcal{X}_\lambda$, the evaluation preimages $x_1, \dots, x_Q \in \mathcal{X}_\lambda$, its corresponding images $\mathbf{y}_1, \dots, \mathbf{y}_Q \in \mathbb{G}_T$ and proofs $\pi_1, \dots, \pi_Q$, and the algebraic representations of the images $(\overrightarrow{z}_1, M_1), \dots, (\overrightarrow{z}_Q, M_Q) \in \mathbb{Z}_p^{q_\mathsf{T}} \times \mathbb{Z}_p^{q_\mathsf{S} \times q_\mathsf{S}}$. Let $\mathbf{g}$ be the generator designated in the public parameters of the group's description. Denote by $s \in \mathbb{Z}_p$ the "effective secret key" s.t. $\overrightarrow{\mathbf{v}} = (\mathbf{g}^{f_1(s)}, \dots, \mathbf{g}^{f_{n_\mathsf{S}}(s)}, e(\mathbf{g},\mathbf{g})^{f_{n_\mathsf{S}+1}(s)}, \dots, e(\mathbf{g},\mathbf{g})^{f_n(s)})$ as guaranteed by Definition 15. If the group validation fails or $\mathsf{Verify}_\mathsf{vrf}(\mathsf{vk}, x_\ell, \mathbf{y}_\ell, \pi_\ell) = 0$ for any $\ell \in \{1, \dots, Q\}$ the meta-reduction aborts.

We now distinguish two cases:

Case 1: The rational functions $\zeta_0, \dots, \zeta_Q$ are linearly dependent, i.e.,

$$\exists \overrightarrow{\alpha} \in \mathbb{Z}_p^{Q+1} \setminus \{0\} : \sum_{\ell=0}^{Q} \alpha_\ell \zeta_\ell(V) \equiv 0 \in \mathbb{Z}_p(V) \; . \tag{9}$$

If there exists such a vector $\overrightarrow{\alpha}$ with $\alpha_0 \neq 0$, the meta-reduction scales $\overrightarrow{\alpha}$ s.t. $\alpha_0 = -1$. Otherwise, if $\zeta_0 \notin \mathsf{span}(\zeta_1, \dots, \zeta_Q) \subset \mathbb{Z}_p(V)$, then the meta-reduction aborts.[14]

Now, due to the correctness of the scheme it holds for each $\ell \in \{0, \dots, Q\}$

$$\mathbf{y}_\ell = e(\mathbf{g},\mathbf{g})^{\widetilde{\sigma}_{x_\ell}(s)/\widetilde{\rho}_{x_\ell}(s)} = e(\mathbf{g},\mathbf{g})^{\zeta_\ell(s)} \; . \tag{10}$$

Consequently, the meta-reduction can predict the image value $\mathbf{y}_0$ of the challenge preimage x_0 as

$$\mathbf{y}_0 \overset{\underline{Eq.}~(10)}{=} e(\mathbf{g},\mathbf{g})^{\zeta_0(s)} \overset{\underline{Eq.}~(9)}{=} e(\mathbf{g},\mathbf{g})^{\sum_{\ell=1}^{Q} \alpha_\ell \zeta_\ell(s)} \overset{\underline{Eq.}~(10)}{=} \prod_{\ell=1}^{Q} \mathbf{y}_\ell^{\alpha_\ell} \tag{11}$$

because $\mathbf{y}_1, \dots, \mathbf{y}_Q$ and $\overrightarrow{\alpha}$ are known by the meta-reduction.

[13] This follows from $\exists \overrightarrow{\alpha} \in \mathbb{Z}_p^{Q+1} \setminus \{0\} : \sum_{\ell=0}^{Q} \alpha_\ell \zeta_\ell(V) = 0 \in \mathbb{Z}_p(V)$. Hence $\exists \hat{\imath} \in \{0, \dots, Q\} : \alpha_{\hat{\imath}} \neq 0$. Because x_i are uniform, it holds that $\Pr[\hat{\imath} \neq 0] \leq 1 - 1/(Q+1)$.

[14] Note that the unbounded adversary also aborts in this case.

Case 2: Now, we consider the case where all $\zeta_0, \ldots, \zeta_Q$ are linearly independent. Because the meta-reduction obtains images and representations for $Q = q^2 + 1 > q_\mathsf{S}^2 + q_\mathsf{T}$ many[15] queries, it can compute some linear dependence between the representations, i.e.,

$$\exists \overrightarrow{\alpha} \in \mathbb{Z}_p^Q \setminus \{0\} : \sum_{\ell=1}^{Q} \alpha_\ell \overrightarrow{z}_\ell = 0 \in \mathbb{Z}_p^{q_\mathsf{T}} \ \wedge \ \sum_{\ell=1}^{Q} \alpha_\ell M_\ell = 0 \in \mathbb{Z}_p^{q_\mathsf{S} \times q_\mathsf{S}} \, . \tag{12}$$

In particular, we have for such a vector $\overrightarrow{\alpha} \in \mathbb{Z}_p^Q$

$$\prod_{\ell=1}^{Q} \mathbf{y}_\ell^{\alpha_\ell} \overset{Eq.~(8)}{=} \prod_{\ell=1}^{Q} \left(\prod_{i=1}^{q_\mathsf{T}} \mathbf{c}_{\mathsf{T},i}^{z_{\ell,i}} \cdot \prod_{i,j=1}^{q_\mathsf{S}} e(\mathbf{c}_{\mathsf{S},i}, \mathbf{c}_{\mathsf{S},j})^{m_{\ell,i,j}} \right)^{\alpha_\ell} \tag{13}$$

$$= \prod_{i=1}^{q_\mathsf{T}} \left(\prod_{\ell=1}^{Q} \mathbf{c}_{\mathsf{T},i}^{\alpha_\ell \cdot z_{\ell,i}} \right) \cdot \prod_{i,j=1}^{q_\mathsf{S}} \left(\prod_{\ell=1}^{Q} e(\mathbf{c}_{\mathsf{S},i}, \mathbf{c}_{\mathsf{S},j})^{\alpha_\ell \cdot m_{\ell,i,j}} \right) \tag{14}$$

$$\overset{Eq.~(12)}{=} \prod_{i=1}^{q_\mathsf{T}} \mathbf{c}_{\mathsf{T},i}^{0} \cdot \prod_{i,j=1}^{q_\mathsf{S}} e(\mathbf{c}_{\mathsf{S},i}, \mathbf{c}_{\mathsf{S},j})^{0} = e(\mathbf{g}, \mathbf{g})^0 \tag{15}$$

and hence $\sum_{\ell=1}^{Q} \alpha_\ell \zeta_\ell(s) = 0$ is implied by

$$e(\mathbf{g}, \mathbf{g})^{\sum_{\ell=1}^{Q} \alpha_\ell \zeta_\ell(s)} = \prod_{\ell=1}^{Q} e(\mathbf{g}, \mathbf{g})^{\alpha_\ell \zeta_\ell(s)} = \prod_{\ell=1}^{Q} \mathbf{y}_\ell^{\alpha_\ell} = e(\mathbf{g}, \mathbf{g})^0 \, . \tag{16}$$

The meta-reduction uses this dependence $\overrightarrow{\alpha}$ to compute the univariate target polynomial

$$\psi(V) := \widetilde{\rho}_{x_1}(V) \cdots \widetilde{\rho}_{x_Q}(V) \cdot \sum_{\ell=1}^{Q} \alpha_\ell \zeta_\ell(V) \tag{17}$$

$$= \sum_{\ell=1}^{Q} \alpha_\ell \widetilde{\sigma}_{x_\ell}(V) \prod_{\ell' \neq \ell} \widetilde{\rho}_{x_{\ell'}}(V) \in \mathbb{Z}_p[V] \tag{18}$$

which fulfills the following properties:

1. ψ is non-zero, since otherwise the rational functions $\zeta_1(V), \ldots, \zeta_Q(V)$ would be linearly dependent (because none of the $\widetilde{\rho}_{x_\ell}(V)$ can be zero).
2. ψ is a polynomial whose degree is bounded by $Q \cdot d \cdot d_f \in \mathsf{poly}(\lambda)$.
3. ψ must vanish at the effective secret key s. This follows from Eq. (16).

In the next step the meta-reduction factors the polynomial ψ to obtain its set of roots $Z(\psi) := \{r \in \mathbb{Z}_p \mid \psi(r) = 0\}$. Factoring a univariate polynomial ψ over a field of size p is possible in time $\mathsf{poly}(\deg(\psi), \log p) = \mathsf{poly}(\lambda)$ through a PPT algorithm [5]; see [49] for a survey on factoring univariate polynomials. Via standard success

[15] Q is larger than the sum of the dimensions of the vector spaces of the algebraic representations for the source and target group elements, i.e., $q_\mathsf{S}^2 + q_\mathsf{T} = \dim(\mathbb{Z}_p^{q_\mathsf{S} \times q_\mathsf{S}} \times \mathbb{Z}_p^{q_\mathsf{T}})$.

boosting, i.e., repeating the factoring algorithm polynomially many times, the meta-reduction can find *all* irreducible factors of ψ with overwhelming probability, e.g. $1 - 2^{-\lambda}$. The meta-reduction then takes all linear factors[16] $(V - r_1), \dots, (V - r_t)$ to obtain the set $Z(\psi) = \{r_1, \dots, r_t\}$ with $|Z(\psi)| \leq \deg(\psi) \leq Qdd_f$.

Since the exponent of the reduction's verification key is $s \in Z(\psi)$ the meta-reduction can find $s = r_i$ by trying out each candidate r_i, i.e., checking if $\overrightarrow{\mathbf{v}} \overset{?}{=} (\mathbf{g}^{f_1(r_i)}, \dots, \mathbf{g}^{f_{n_\mathsf{S}}(r_i)}, e(\mathbf{g}, \mathbf{g})^{f_{n_\mathsf{S}+1}(r_i)}, \dots, e(\mathbf{g}, \mathbf{g})^{f_n(r_i)})$. The meta-reduction can finally compute the challenge image

$$\mathbf{y}_0 = \mathbf{g}^{\widetilde{\sigma}_{x_0}(s)/\widetilde{\rho}_{x_0}(s)} \tag{19}$$

and return it to the reduction.

Note that in both cases, the meta-reduction behaves indistinguishably from the unbounded adversary, which does not get to see the algebraic representations of group elements. Therefore, the meta-reduction has the same winning probability as the reduction $\epsilon_\mathcal{M} \geq \epsilon_\mathcal{B} - 2^{-\lambda}$ except for when the factorization of the target polynomial fails. The meta-reduction runs the reduction once and has to simulate the adversary r times. Hence, the meta-reduction's runtime is $t_\mathcal{M} = t_\mathcal{B} + r\mathsf{poly}(\lambda) = t_\mathcal{B} + \mathsf{poly}(\lambda)$.

Remark 12. Indeed, Theorem 1 can be applied if the input space $\mathcal{X}$ is only of polynomial size for a suitable definition of weak selective unpredictiability. Here, one has to make sure that the challenge preimage is not contained in the Q many query preimages, otherwise the adversary could predict trivially.

Corollary 1. *If the reduction in Theorem 1 is efficient, then* NICA *is efficiently solvable. In other words,* $t_\mathcal{B}/\epsilon_\mathcal{B} \in \mathsf{poly}(\lambda) \implies t_\mathcal{M}/\epsilon_\mathcal{M} \in \mathsf{poly}(\lambda)$.

We move on to our next result.

Theorem 2. *Let* $p = p(\lambda)$ *be a superpolynomial group order. Let* NICA *be some univariate DLog-hard assumption according to Definition 7 with* $l_1, l_2, d_\mathsf{NICA} \in \mathsf{poly}(\lambda)$, *and polynomials* $r_1, \dots, r_{l_1}, t_1, \dots, t_{l_2} \in \mathbb{Z}_p[S]$ *of degree at most* d_NICA. *Let* $n, d, r \in \mathsf{poly}(\lambda)$. *Let* vrf *be a rational VRF of evaluation degree* d *with* n *verification key elements s.t.* $\forall x \in \mathcal{X} : \sigma_x(\overrightarrow{V}) = V_1$.[17]

If there exists an algebraic $(t_\mathcal{B}, \epsilon_\mathcal{B}, r, 0, 1)$*-reduction* $\mathcal{B}$ *(that forwards its group description as part of the verification key) from* NICA *to the* 0*-adaptive pseudorandomness of* vrf, *then there exists an adversary* $\mathcal{M}$ *that* $(t_\mathcal{M}, \epsilon_\mathcal{M})$*-breaks* NICA *with* $\epsilon_\mathcal{M} \geq \epsilon_\mathcal{B} - 2^{-\lambda}$ *and* $t_\mathcal{M} \leq t_\mathcal{B} + \mathsf{poly}(l_2, d_\mathsf{NICA}, d, \log p, r) = t_\mathcal{B} + \mathsf{poly}(\lambda)$.

Proof. For short notation, let $L := l_1 + l_2$ and let $\overrightarrow{c}(S) := (1, r_1(S), \dots, r_{l_1}(S), 1/t_1(S), \dots, 1/t_{l_2}(S)) \in \mathbb{Z}_p^{1+L}$ denote the function that maps the NICA's secret exponent to its elements actual exponents (plus the generator as the first element). We

[16] Note that for irreducible non-linear factors, i.e., of degree more than 1, all roots must be in the algebraic closure $\overline{\mathbb{Z}}_p \setminus \mathbb{Z}_p$, otherwise the factor would not be irreducible.

[17] Essentially, the first verification key element $\mathbf{h} := \mathbf{v}_1$ is the new generator relative to which the VRF is evaluated.

describe the adversary $\mathcal{M}$ as a meta-reduction [14] that takes input a group description Π and group elements $\overrightarrow{\mathbf{c}}$ from NICA (where $\mathbf{g}$ is the generator designated in the public parameters of the group's description), forwards it to the reduction $\mathcal{B}$ and simulates an adversary for the 0-adaptive pseudorandomness that is indistinguishable from a computationally unbounded adversary that does not get to see the algebraic representation of group elements. Because the reduction is algebraic, whenever it outputs a group element $\mathbf{y} \in \mathbb{G}_T$ it must also provide a representation[18] $M \in \mathbb{Z}_p^{(1+L)\times(1+L)}$ w.r.t. the NICA challenge elements s.t.

$$\mathbf{y} = \mathbf{g_T}^{\overrightarrow{c}(s)^\intercal M \overrightarrow{c}(s)} \tag{20}$$

As such, the representation M gives a concise way of writing an element relative to the elements of the NICA challenge.

Let us first describe the unbounded adversary with success probability 1. The adversary obtains as input the verification key $\mathsf{vk} = (\Pi, \overrightarrow{\mathbf{v}})$. Then the adversary samples and submits $x^* \overset{\$}{\leftarrow} \mathcal{X}$ as the challenge (in fact, arbitrary x^* works as well) and obtains the image $\mathbf{y}^*$ and its representation M^*. If the group validation fails, the adversary aborts. Otherwise, the (unbounded) adversary computes the secret evaluation key from vk and uses it to compute $\mathbf{y}' = \mathsf{Eval}_{\mathsf{vrf}}(\mathsf{sk}, x^*)$. It outputs REAL if $\mathbf{y}' = \mathbf{y}^*$, RANDOM otherwise.

Now, we describe the meta-reduction. It forwards its input to the reduction; the NICA challenge has the form $\overrightarrow{\mathbf{c}} = (\mathbf{g}^{r_1(s)}, \ldots, \mathbf{g}^{r_{l_1}(s)}, \mathbf{g}^{1/t_1(s)}, \ldots, \mathbf{g}^{1/t_{l_2}(s)})$. The meta-reduction simulates an unbounded adversary as follows: It obtains from the reduction as input the verification key vk and its algebraic representations $M_{\mathsf{vk},1}, \ldots, M_{\mathsf{vk},n}$. The meta-reduction proceeds analogously to the unbounded adversary in that it submits $x^* \overset{\$}{\leftarrow} \mathcal{X}$ as the challenge and obtains the image $\mathbf{y}^*$ and its representation M^*. If the group validation fails, the meta-reduction aborts.

Denote by $s \in \mathbb{Z}_p$ the concrete exponent of the NICA challenge. Let $M_{\mathbf{h}} := M_{\mathsf{vk},1} \neq 0 \in \mathbb{Z}_p^{(1+L)\times(1+L)}$ and let $\mathbf{h} := \mathbf{v}_1 = \mathbf{g}^{v_1} = \mathbf{g}^{\sigma_x(\overrightarrow{v})} = \mathbf{g}^{\overrightarrow{c}(s)^\intercal M \overrightarrow{c}(s)}$. Let

$$t : \mathbb{Z}_p \to \mathbb{Z}_p^n : S \mapsto \begin{pmatrix} \overrightarrow{c}(S)^\intercal M_{\mathsf{vk},1} \overrightarrow{c}(S) \\ \vdots \\ \overrightarrow{c}(S)^\intercal M_{\mathsf{vk},n} \overrightarrow{c}(S) \end{pmatrix} \tag{21}$$

be the function that maps the exponent of the NICA challenge to the verification key exponents, i.e., $\overrightarrow{V} = t(S)$. Furthermore, for any preimage $x \in \mathcal{X}$ let $\rho'_x(S) := (\rho_x \circ t)(S) = \rho_x(\overrightarrow{V})$.

By the correctness and unique provability of the VRF, for each correct (and only for correct) representation M of the image $\mathbf{y}$ of x it holds that

$$\mathbf{g_T}^{\overrightarrow{c}(s)^\intercal M \overrightarrow{c}(s)} = \mathbf{y} \tag{22}$$

$$= \mathbf{g_T}^{\sigma_x(\overrightarrow{v})/\rho_x(\overrightarrow{v})} \tag{23}$$

[18] Since Definition 7 considers NICAs with only source group elements (for simply exposition) we have no $\overrightarrow{z}$ term here. Considering target group NICAs does not affect our result.

$$= e(\mathbf{g}, \mathbf{g})^{\sigma_x(\vec{v})/\rho_x(\vec{v})} \tag{24}$$

$$= e(\mathbf{h}, \mathbf{g})^{1/\rho_x(\vec{v})} \tag{25}$$

$$= e(\mathbf{g}^{\vec{c}(s)^\intercal M \vec{c}(s)}, \mathbf{g})^{1/\rho_x(\vec{v})} \tag{26}$$

$$= \mathbf{g_T}^{\vec{c}(s)^\intercal M \vec{c}(s)/\rho_x(\vec{v})} \tag{27}$$

$$= \mathbf{g_T}^{\vec{c}(s)^\intercal M \vec{c}(s)/\rho'_x(s)} \tag{28}$$

where $\vec{v}$ are the exponents of the verification key, and thus

$$0 = \vec{c}(s)^\intercal \left(M \cdot \rho'_x(s) - M_\mathbf{h}\right) \vec{c}(s) \, . \tag{29}$$

Hence, the polynomial

$$\psi'_{x,M}(S) = \left(\prod_{i,j=1}^{l_2} t_i t_j \right)^{d+1} \cdot \vec{c}(S)^\intercal \left(M \cdot \rho'_x(S) - M_\mathbf{h}\right) \vec{c}(S) \tag{30}$$

contains the exponent of the NICA challenge s as a zero. Note that $\psi'_{x,M}(S)$ is a polynomial of degree at most $d_\psi := 8d \cdot d^2_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$. Now, consider the two cases:

- $\exists M : \psi'_{x^*,M}(S) \equiv 0$, then the meta-reduction finds such a representation[19] M' s.t. $\psi'_{x^*,M'}(S) \equiv 0$ is zero, and outputs REAL iff $\mathbf{y}^* = \prod_{i,j=0}^{L} e(\mathbf{c}_i, \mathbf{c}_j)^{M'_{i+1,j+1}}$ where $\mathbf{c}_0 := \mathbf{g}$, and RANDOM otherwise.
- $\forall M : \psi'_{x^*,M}(S) \not\equiv 0$, then the meta-reduction computes the target polynomial $\psi'_{x^*,M^*}(S) \not\equiv 0$, and factorizes it to obtain a set of at most d_ψ zeros $Z(\psi'_{x^*,M^*}) = \{s_1, \ldots, s_d\}$. We assume that the factorization is repeated sufficiently often (polynomial many times) so that it fails with probability at most $2^{-\lambda}$. If any candidate zero s_i is the NICA exponent, the meta-reduction solves its NICA challenge directly. If no candidate zero s_i is the NICA exponent, then the meta-reduction answers RANDOM.

Analysis. In the first case, we know that any M' s.t. $\psi'_{x^*,M'}(S) \equiv 0$ is a valid representation for the image of x^* by Eq. (29). Thus, if M^* and M' describe the same group element, then the image $\mathbf{y}^*$ is the real image of x^*, otherwise it is a random element.

In the second case, suppose that the image $\mathbf{y}^*$ is the real image, then the polynomial $\psi'_{x^*,M^*}(S)$ must contain the NICA exponent s as a zero. However, we know that $\psi'_{x^*,M}(S) \not\equiv 0$ for any M, and it is of degree at most d_ψ. Therefore, factoring it takes time at most $\mathsf{poly}(l, d_\psi, \log p)$ and yields a set of at most d_ψ zeros. If any zero is the NICA exponent, the meta-reduction has solved the corresponding DLog-challenge which—by Definition 8—can only happen with negligible probability. Consequently, if no zero is the NICA exponent, then the representation M^* must describe a random group element; the meta-reduction outputs RANDOM.

[19] This amounts to solving a system of polynomially many linear equations.

6 Generic Attacks on Parametrized Rational VUFs

In the previous section, we showed two results which rule out VUFs with logarithmic proofs whose security can be algebraically based on a non-interactive computational hardness assumption. However, both results have caveats: Theorem 1 requires that the exponent of each verification key element can be expressed as a univariate polynomial in one secret variable, i.e., the secret key consists of basically just one element, while Theorem 2 requires a similar restriction for the exponents of the assumption on which the security of the VUF is based.

To circumvent the restrictions of the previous results, we want to show a lower bound for a different class of VUFs where neither the verification key nor the NICA needs to be univariate. The class of VUFs that we consider here will be the class of *parametrized rational* VUFs: If vuf is rational of evaluation degree d_{vuf}, we know that there are families of polynomials $(\sigma_x)_x, (\rho_x)_x$ s.t. we have for each tuple $(x, \mathsf{vk}, \mathbf{y}, \pi)$ accepted by $\mathsf{Verify}_{\mathsf{vuf}}$

$$\mathbf{y} = e(\mathbf{h}, \mathbf{h})^{\sigma_x(\mathsf{dlog}_{\mathbf{h}}(\mathsf{vk}))/\rho_x(\mathsf{dlog}_{\mathbf{h}}(\mathsf{vk}))}$$

for some dedicated generator $\mathbf{h}$ in vk. We call vuf **parametrized rational** of evaluation degree d_{vuf}, if the set of inputs $\mathcal{X}$ equals $\mathbb{Z}_p$ and if there are polynomials $\sigma, \rho \in \mathbb{Z}_p[V, X]$ of total degree d_{vuf} s.t. we have for each $x \in \mathcal{X}$

$$\sigma_x(V) = \sigma(V, x) \quad \text{and} \quad \rho_x(V) = \rho(V, x).$$

I.e., the members of the family $(\sigma_x)_x$ resp. $(\rho_x)_x$ are derived from a universal polynomial $\sigma(\cdot, X)$ resp. $\rho(\cdot, X)$ of total degree d_{vuf}. We give a formal definition of parametrized rational VUFs later (Definition 18).

For the class of parametrized VUFs we can show that there can be no reductions which base their security on hard NICAs. However, now there are two new restrictions that we need to impose on the reduction. First, we can only exclude *generic* reductions now, while our previous bounds did hold for algebraic reductions. Second, we can only exclude reductions that solve *extremely small* Uber-assumptions [11]. Under those two restrictions, we get the following result:

Theorem 3. *Let* vuf *be a parametrized rational VUF of evaluation degree* $d_{\mathsf{vuf}} \in O(1)$. *Let* NICA *be an Uber-assumption of degree* $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$ *and of size* $q \leq \sqrt{\log\log(w)}$ *for some* $w \in \mathsf{poly}(\lambda)$.

If NICA *is hard and* $Q > 2 \cdot (1 + \log\log w) \cdot w^{2\log(d_{\mathsf{vuf}}+1)}$, *then there is no generic reduction that can transform an adversary for the weak* Q*-selective unpredictability of* vuf *to a* NICA *solver.*

Unfortunately, the proof of Theorem 3 is very technical and requires, among other things, notions and results from the field of Groebner bases and projective algebraic geometry. We will therefore not directly prove Theorem 3. Instead, we will first introduce a new technicality which we call *verification equations*. Then, in Section 6.1, we will illustrate the use of verification equations and some key ideas in the proof of Theorem 3 by proving a lower bound for a toy example (the VUF of Dodis & Yampolskiy [17]).

In Section 6.2, we will summarize necessary results of the field of Groebner bases and algebraic geometry and finally, in Section 6.3, we will give a proof of Theorem 3 using verification equations and algebraic geometry.

Before we start, we explain the setting in which we prove our results. Since we are only considering generic algorithms here, we will assume in this section that the groups $\mathbb{G}, \mathbb{G}_T$ and the pairing $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ form a bilinear *generic group* [8, 48]. Note that the generic bilinear group $(\mathbb{G}, \mathbb{G}_T, e)$ does not need an explicit verification algorithm in the sense of Definition 3 anymore, since generic groups are certified by default. Indeed, the group operation oracles given to the reduction reject each handle that does not point to a valid group element. Further, in this section, we will no longer demand that there is a publicly known group generator of $\mathbb{G}$. Instead, we expect the verification key of each VUF vuf to contain a dedicated group generator of $\mathbb{G}$. Further, note that in this work we call an algorithm only generic iff it is generic in the sense of Shoup's GGM and algebraic (cf. Definition 5). Therefore, whenever we require that a reduction $\mathcal{R}$ is generic, we additionally require, implicitly, that $\mathcal{R}$ is algebraic.

Let us start with our technical framework. We first introduce a notion of *Uber-assumptions* that is similar to the definition of Boyen [11], but differs in two subtle points: first, we do not make any requirements about the solution that a challenger has to compute when given a sample of NICA, while Boyen expects the adversary to compute a concrete polynomial in the exponent. Second, we require that the polynomials $f_{A_1}, \dots, f_{A_{q_1}}, f_{B_1}, \dots, f_{B_{q_2}}$ that we use to compute the exponents of the challenge elements are *sparse*, i.e., only have a polynomial number of non-zero coefficients and can efficiently be written in normal form, while Boyen makes no special restriction on the polynomials $f_{A_1}, \dots, f_{A_{q_1}}, f_{B_1}, \dots, f_{B_{q_2}}$.

Definition 16 (Computational Uber-Assumptions). *Let* NICA *be a non-interactive computational assumption.*

We call NICA *an* ***Uber-assumption*** *if there is a polynomial bound* $t = t(\lambda)$ *and a set of sparse polynomials* $f_{A_1}, \dots, f_{A_{q_1}}, f_{B_1}, \dots, f_{B_{q_2}} \in \mathbb{Z}_p[Z_1, \dots, Z_t]$ *that can be computed efficiently and uniformly s.t. the distributions of challenge samples of* NICA *is identical to the output of the following algorithm:*

```
$\mathcal{D}(1^\lambda) :=$   {
        draw $\mathbf{g}$ by any distribution s.t. $\mathbf{g}$ is a generator of $\mathbb{G}$;
        draw $(z_1, \dots, z_t) \xleftarrow{\$} \mathbb{Z}_p^t$ uniformly and indpendently at random;
        set $a_1 := f_{A_1}(z_1, \dots, z_t), \dots, a_{q_1} := f_{A_{q_1}}(z_1, \dots, z_t)$;
        set $b_1 := f_{B_1}(z_1, \dots, z_t), \dots, b_{q_2} := f_{B_{q_2}}(z_1, \dots, z_t)$;
        return $(\mathbf{g}, \mathbf{g}^{a_1}, \dots, \mathbf{g}^{a_{q_1}}, e(\mathbf{g}, \mathbf{g})^{b_1}, \dots, e(\mathbf{g}, \mathbf{g})^{b_{q_2}})$;
    }.
```

Let $d_{\mathsf{NICA}} = \max\{\deg f_{A_1}, \dots, \deg f_{A_{q_1}}, \deg f_{B_1}, \dots, \deg f_{B_{q_2}}\}$. *We call* d_{NICA} *the* ***degree*** *of* NICA *and* $q = 1 + q_1 + q_2$ *the* ***size*** *of* NICA.

Now, we introduce the notion of *verification equations*. To this end, let $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ be a VUF. In this section, we denote by $\overrightarrow{\mathsf{vk}_S} \in \mathbb{G}^{n_1}$ resp.

$\overrightarrow{\mathsf{vk}_T} \in \mathbb{G}_T^{n_2}$ the vectors of source resp. target group elements of vk and by $\overrightarrow{\pi_S} \in \mathbb{G}^{u_1}$ resp. $\overrightarrow{\pi_T} \in \mathbb{G}_T^{u_2}$ the vectors of source resp. target group elements of π. Let $n := n_1 + n_2$ denote the size of vk and $u := u_1 + u_2$ the size of π. Remember that we denoted by $\mathcal{X} = (\mathcal{X}_\lambda)_\lambda$ the domain of $\mathsf{Eval}_{\mathsf{vuf}}(\mathsf{sk}_{\mathsf{vuf}}, \_)$.

Definition 17. *Consider the polynomial ring in $n + u + 1$ variables*

$$\mathbb{Z}_p[V, P, Y] := \\ \mathbb{Z}_p[V_{S,1}, \ldots, V_{S,n_1}, V_{T,1}, \ldots, V_{T,n_2}, P_{S,1}, \ldots, P_{S,u_1}, P_{T,1}, \ldots, P_{T,u_2}, Y].$$

We want to evaluate elements of $\mathbb{Z}_p[V, P, Y]$ on exponents of tuples $(\mathsf{vk}, \pi, \mathbf{y})$. The formal variables $V_{S,i}$ resp. $V_{T,i}$ are placeholders for the exponents of the source resp. target group elements of vk. Analogously, the variables $P_{S,i}$ resp. $P_{T,i}$ evaluate the exponents of π and the variable Y evaluates the exponent of $\mathbf{y}$.

Let $\phi = (\phi_x)_{x \in \mathcal{X}}$ be a family of polynomials, $\phi_x \in \mathbb{Z}_p[V, P, Y]$. ϕ is called a ***(family of) verification equation(s)*** *of vuf if for each tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ accepted by the verification algorithm $\mathsf{Verify}_{\mathsf{vrf}}$ and for* ***each*** *generator $\mathbf{h} \in \mathbb{G}$ and corresponding generator $\mathbf{h}_T := e(\mathbf{h}, \mathbf{h}) \in \mathbb{G}_T$ it holds that*

$$\phi_x(v_{S,1}, \ldots, v_{S,n_1}, v_{T,1}, \ldots, v_{T,n_2}, p_{S,1}, \ldots, p_{S,u_1}, p_{T,1}, \ldots, p_{T,u_2}, y) = 0$$

where $v_{S,i} := \mathsf{dlog}_{\mathbf{h}}(\mathsf{vk}_{S,i})$, $v_{T,j} := \mathsf{dlog}_{\mathbf{h}_T}(\mathsf{vk}_{T,j})$, $p_{S,l} := \mathsf{dlog}_{\mathbf{h}}(\pi_{S,l})$, $p_{T,m} := \mathsf{dlog}_{\mathbf{h}_T}(\pi_{T,m})$ and $y := \mathsf{dlog}_{\mathbf{h}_T}(\mathbf{y})$ for $i \in [n_1], j \in [n_2], l \in [u_1]$ and $m \in [u_2]$. We define the ***degree*** *of ϕ as $d_\phi = \max_{x \in \mathcal{X}} \deg \phi_x$.*

A verification equation is a necessary implication of the exponents of a correct tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$. I.e., if $(\mathsf{vk}, x, \mathbf{y}, \pi)$ is accepted by $\mathsf{Verify}_{\mathsf{vuf}}$, then ϕ_x must vanish on the exponents of $\mathsf{vk}, \pi, \mathbf{y}$. This means a certain equation holds over the discrete logarithms of $\mathsf{vk}, \pi, \mathbf{y}$. Note that the evaluation of a verification equation ϕ is independent of the chosen generator $\mathbf{h}$ of $\mathbb{G}$. This is important, since a reduction – like the one in [17] – may give the adversary a special group generator that allows the reduction to answer some evaluation queries.

In comparison with the notion of *pairing equations*, which we introduced in Definition 12, we note three differences: first, the formula $\phi_x(\overrightarrow{v}, \overrightarrow{p}, y) = 0$ is invariant under the group generator relative to which the discrete logarithms $\overrightarrow{v}, \overrightarrow{p}, y$ are computed to, while the evaluation of a pairing equation is dependent of a fixed group generator. Second, verification equations are allowed to be of arbitrary degree, while pairing equations can at most be quadratic. Third, verification equations are defined relative to a VUF and *must* vanish on the exponents of each tuple of group elements accepted by its verification algorithm.

Given the tool of verification equations, we can now formally define parametrized rational VUFs:

Definition 18. *A VUF $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ is called* ***parametrized rational*** *of* ***evaluation degree*** *$d_{\mathsf{vuf}} = d_{\mathsf{vuf}}(\lambda)$, if the following things hold:*

1. *The set of possible inputs of vuf is $\mathcal{X} = \mathbb{Z}_p$.*

2. *There are polynomials $\sigma, \rho \in \mathbb{Z}_p[V_{S,1}, \ldots, V_{S,n_1}, V_{T,1}, \ldots, V_{T,n_2}, X]$ of total degree d_{vuf} such that the family $(\phi_x)_{x \in \mathcal{X}}$ of polynomials*

$$\phi_x(V, P, Y) := \rho(V, x) \cdot Y - \sigma(V, x)$$

   is a family of verification equations of vuf.
3. *For each generator $\mathbf{h} \in \mathbb{G}$ and each tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ accepted by* $\mathsf{Verify}_{\mathsf{vuf}}$ *we have*

$$\rho(\overrightarrow{v_S}, \overrightarrow{v_T}, x) \neq 0$$

   where $\overrightarrow{v_S}$ resp. $\overrightarrow{v_T}$ denote the exponents of the elements $\mathsf{vk}_{S,1}, \ldots, \mathsf{vk}_{S,n_1}$ resp. $\mathsf{vk}_{T,1}, \ldots, \mathsf{vk}_{T,n_2}$ relative to the basis $\mathbf{h}$ resp. $e(\mathbf{h}, \mathbf{h})$.

The notion of a *parametrized* rational VUF is more restrictive than the notion of a rational VUF, which we introduced in Definition 14. It requires that there is a strong algebraic connection between the fractions that describe the exponents of different images, while the normal definition of a rational VUF does not demand any correlation between the fractions that are induced by different inputs. Obviously, the VUF of Dodis & Yampolskiy [17] is parametrized rational.

However, for now let vuf be any VUF and assume there is a generic reduction $\mathcal{R}$ that can solve instances of the Uber-assumption NICA of degree $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$ when given access to a successful adversary $\mathcal{A}$ against the weak selective Q-unpredictability of vuf.

$\mathcal{R}$ is given a tuple $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}})$ sampled from $\mathcal{D}$. Recall that a tuple $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \xleftarrow{\$} \mathcal{D}(1^\lambda)$ is of the form

$$(\mathbf{g}, \mathbf{g}^{f_{A_1}(\overrightarrow{z})}, \ldots, \mathbf{g}^{f_{A_{q_1}}(\overrightarrow{z})}, e(\mathbf{g}, \mathbf{g})^{f_{B_1}(\overrightarrow{z})}, \ldots, e(\mathbf{g}, \mathbf{g})^{f_{B_{q_2}}(\overrightarrow{z})})$$

where $z_1, \ldots, z_t \in \mathbb{Z}_p$ are drawn uniformly at random. Let $A_1, \ldots, A_{q_1}, B_1, \ldots, B_{q_2}$ be formal variables that represent the exponents $a_1, \ldots, a_{q_1}$, $b_1, \ldots, b_{q_2}$ of $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}})$. The polynomials $f_{A_1}, \ldots, f_{B_1}, \ldots$ induce a homomorphism of rings $\mathcal{F}_2 : \mathbb{Z}_p[A, B] \to \mathbb{Z}_p[Z]$ that maps each A_i to f_{A_i} and each B_j to f_{B_j}.

$\mathcal{R}$ has to solve $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}})$ while it may make black-box use of a weak selective adversary $\mathcal{A}$ that asks for Q evaluations of the VUF. We may model $\mathcal{R}$'s usage of $\mathcal{A}$ as an oracle access.

Whenever $\mathcal{R}$ accesses $\mathcal{A}$, it sends $\mathcal{A}$ a verification key vk, a list of inputs $x_0, \ldots, x_Q$, a list of proofs $\pi_1, \ldots, \pi_Q$ and a list of image group elements $\mathbf{y}_1, \ldots, \mathbf{y}_Q$. Since $\mathcal{R}$ is algebraic, the exponents of the source group elements $(\mathsf{vk}_{S,i})_{i \in [n_1]}, (\pi_{S,i,j})_{i \in [Q], j \in [u_1]}$ and the target group elements $(\mathsf{vk}_{T,i})_{i \in [n_2]}, (\pi_{T,i,j})_{i \in [Q], j \in [u_2]}, (\mathbf{y}_i)_{i \in [Q]}$ must be representable by polynomials

$$(v_{S,i})_i, (p_{S,i,j})_{i,j}(v_{T,i})_i, (p_{T,i,j})_{i,j}, (y_i)_i \subset \mathbb{Z}_p[A, B]$$

of degree 1 resp. 2 in the variables $A_1, \ldots, A_{q_1}, B_1, \ldots, B_{q_2}$, since those variables represent the exponents of $\mathbf{g}^{\overrightarrow{a}}$ and $\mathbf{g}^{\overrightarrow{b}}$. Let $(V_{S,i})_i, (V_{T,i})_i, (P_{S,i,j})_{i,j}, (P_{T,i,j})_{i,j}$ and $(Y_i)_i$ be the formal variables that represent the exponents of $(\mathsf{vk}_{S,i})_i$, $(\mathsf{vk}_{T,i})_i$,

$(\pi_{S,i,j})_{i,j}$, $(\pi_{T,i,j})_{i,j}$ and $(\mathbf{y}_i)_i$. Then, the algebraic representations of those group elements induce a morphism of rings $\mathcal{F}_1 : \mathbb{Z}_p[V_S, V_T, P_S, P_T, Y] \to \mathbb{Z}_p[A, B]$ that maps each variable $V_{S,i}$, $V_{T,i}$, $P_{S,i,j}$, $P_{T,i,j}$ resp. Y_i to the corresponding polynomial $v_{S,i}$, $v_{T,i}$, $p_{S,i,j}$, $p_{T,i,j}$ resp. y_i of degree ≤ 2.

By composing $\mathcal{F}_1$ and $\mathcal{F}_2$, we get a ring homomorphism

$$\mathcal{F} := \mathcal{F}_2 \circ \mathcal{F}_1 : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z], \tag{31}$$

which maps each variable to a sparse polynomial of degree $\leq 2d_{\mathsf{NICA}}$ that explains the corresponding group element outputted by $\mathcal{R}$ in terms of the values $z_1, \ldots, z_t$ drawn by $\mathcal{D}$.

Now, let $(\phi_x)_x$ be a family of verification equations of constant degree d_ϕ. For $i \in [Q]$, the polynomial ϕ_{x_i} lies in $\mathbb{Z}_p[V, P_i, Y_i]$. Assume that for each $i \in [Q]$, $\mathsf{Verify}_{\mathsf{vuf}}$ accepts the tuple $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ (otherwise, we assume that the adversary $\mathcal{A}$ aborts). Then, the polynomial ϕ_{x_i} must vanish over the exponents of $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$. This translates to

$$\mathcal{F}(\phi_{x_i})(z_1, \ldots, z_t) = 0$$

where $\mathcal{F}(\phi_{x_i}) \in \mathbb{Z}_p[Z]$ is the polynomial yielded by replacing each variable $V_{S,j}$, $V_{T,j}$, $P_{S,i,j}$, $P_{T,i,j}$ resp. Y_i by the polynomial $\mathcal{F}(V_{S,j})$, $\mathcal{F}(V_{T,j})$, $\mathcal{F}(P_{S,i,j})$, $\mathcal{F}(P_{T,i,j})$ resp. $\mathcal{F}(Y_i) \in \mathbb{Z}_p[Z]$.

Now, note that $\mathcal{F}(\phi_{x_i})$ is a polynomial of degree $d_\phi \cdot 2d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$. A Schwartz-Zippel-like argument (see Lemma 1) guarantees that – since the values $z_1, \ldots, z_t \in \mathbb{Z}_p$ have been drawn uniformly at random – $\mathcal{F}(\phi_{x_i})$ can only vanish with non-negligible probability at $\overrightarrow{z}$ if it is the zero polynomial in $\mathbb{Z}_p[Z]$. The next lemma makes this observation formal.

Lemma 3. *Let $d_{\mathsf{NICA}} = d_{\mathsf{NICA}}(\lambda) > 0$ and $d_\phi \in O(1)$ be such that $\frac{p}{d_{\mathsf{NICA}}}$ grows faster than any polynomial. Let $Q \in \mathsf{poly}(\lambda)$.*

Let $\mathcal{R}$ be a generic PPT reduction as above. Let $(\phi_x)_x$ be a family of verification equations for vuf *of degree d_ϕ. Let $\mathcal{A}$ be an adversary on the weak Q-selective unpredictability of* vuf. *Assume that $\mathcal{A}$ aborts (i.e., returns $\perp$) if it is given at least one tuple* $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ *that is rejected by* $\mathsf{Verify}_{\mathsf{vuf}}$.

Draw $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \xleftarrow{\$} \mathcal{D}(1^\lambda)$ and let

$$\mathsf{vk}, x_0, \ldots, x_Q, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$$

be the information sent by $\mathcal{R}$ to $\mathcal{A}$ in their first interchange of a run of $\mathcal{R}(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}$, $e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}})$ where $\mathcal{A}$ does not abort. Let $\mathcal{F} : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$ be the corresponding morphism from Eq. (31)*, which is determined by the algebraic explanations of the group elements* $\mathsf{vk}, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$.

With overwhelming probability over the randomness of $\mathcal{R}$ and $\mathcal{D}$, we have for each $i \in [Q]$

$$\mathcal{F}(\phi_{x_i})(Z) = 0 \in \mathbb{Z}_p[Z].$$

We give a proof of this lemma in Appendix B.

We will illustrate the use of Lemma 3 by giving a lower bound for the security of the VUF of Dodis & Yampolskiy [17] in Section 6.1. In Section 6.3, we will finally use Lemma 3 to prove Theorem 3, i.e., a lower bound for parametrized rational VUFs.

6.1 Toy Example: the VUF of Dodis and Yampolskiy

In this subsection, we will show lower bounds for the VRF resp. VUF of Dodis & Yampolskiy [17]. To do so, we use Lemma 3 together with a mathematical trick to show that the weak selective unpredictability of the VUF of Dodis & Yampolskiy [17] cannot be based on the hardness of an Uber-assumption NICA of polynomial degree d_{NICA} if the number Q of evaluation queries is higher than $d_{\mathsf{NICA}} + 1$.

We do so by describing an efficient meta-reduction $\mathcal{M}$ that plays the weak selective unpredictability game of the VUF against an efficient and *generic* reduction $\mathcal{R}$. We show that – with overwhelming probability – $\mathcal{M}$ is able to simulate a perfect unbounded adversary for the weak selective unpredictability of the VUF. It then follows that there is a generic PPT algorithm that can solve NICA. This is a contradiction to the hardness of NICA.

Similar to the proof idea of Theorem 1, $\mathcal{M}$ uses the fact that $\mathcal{R}$ must supply for each random input x_i a proof π_i and an image value $\mathbf{y}_i \in \mathbb{G}_T$. Since the exponent of each $\mathbf{y}_i$ is mathematically uniquely determined by the verification key, to which $\mathcal{R}$ committed, and the input x_i, $\mathcal{M}$ is able to extract a secret key corresponding to the reduction's verification key vk by querying enough image values $\mathbf{y}_1, \ldots, \mathbf{y}_Q$.

It can then in turn use this secret key to evaluate the function and thereby break unpredictability. However, with the techniques presented in this section, we obtain a bound on the number Q of queries that depends on the degree of NICA instead of its size like in the previous section. In fact, we show that $Q \geq d_{\mathsf{NICA}} + 2$ evaluation queries of $\mathcal{M}$ suffice to extract the secret key.

Now, let us recall the VUF $\mathsf{vuf}^{\mathsf{DY}} = (\mathsf{Gen}^{\mathsf{DY}}_{\mathsf{vuf}}, \mathsf{Eval}^{\mathsf{DY}}_{\mathsf{vuf}}, \mathsf{Verify}^{\mathsf{DY}}_{\mathsf{vuf}})$ by Dodis & Yampolskiy [17] over a pairing group $(\mathbb{G}, \mathbb{G}_T, e)$:

$\mathsf{Gen}^{\mathsf{DY}}_{\mathsf{vuf}}(1^\lambda)$: Given the parameter λ, compute a generator $\mathbf{h}$ of the cyclic source group $\mathbb{G}$ of prime order p, sample $s \xleftarrow{\$} \mathbb{Z}_p^\times$ and set $\mathsf{vk}_1 := \mathbf{h}, \mathsf{vk}_2 := \mathbf{h}^s$.
Output $\mathsf{vk} = (\mathsf{vk}_1, \mathsf{vk}_2)$ as verification key and $\mathsf{sk} = (\mathbf{h}, s)$ as secret key.

$\mathsf{Eval}^{\mathsf{DY}}_{\mathsf{vuf}}(\mathsf{sk}, x)$: Given the secret key $\mathsf{sk} = (\mathbf{h}, s)$ and an input[20] $x \in \mathbb{Z}_p$, we compute the source group element $\pi = \mathbf{h}^{\frac{1}{s+x}}$, the target group element $\mathbf{y} = e(\mathbf{h}, \mathbf{h})^{\frac{1}{s+x}}$ and output both. (We output $\perp$, if $s + x = 0$.)

$\mathsf{Verify}^{\mathsf{DY}}_{\mathsf{vuf}}(\mathsf{vk}, x, \mathbf{y}, \pi)$: Given the verification key $\mathsf{vk} = (\mathsf{vk}_1, \mathsf{vk}_2)$, an input $x \in \mathbb{Z}_p$, a proof $\pi \in \mathbb{G}$ and an output value $\mathbf{y} \in \mathbb{G}_T$, we first check[21] that vk_1 is indeed a generator of the group $\mathbb{G}$. After that, verification checks that the following equalities of target group elements hold

$$e(\mathsf{vk}_1^x \cdot \mathsf{vk}_2, \pi) = e(\mathsf{vk}_1, \mathsf{vk}_1) \quad \text{and} \quad e(\mathsf{vk}_1, \pi) = \mathbf{y}.$$

We accept if vk_1 is indeed a generator and both equations hold, and reject otherwise.

We want to analyze the verification algorithm of $\mathsf{vuf}^{\mathsf{DY}}$. To this end, let $x \in \mathbb{Z}_p$ and let V_1, V_2, Y, P be formal variables that represent the discrete logarithms of $\mathsf{vk}_1, \mathsf{vk}_2, \mathbf{y}, \pi$.

[20] In [17], x is a binary string and there exists an efficient, injective mapping into $\mathbb{Z}_p$. For simplicity, we skip this step and consider x to be an element of $\mathbb{Z}_p$.

[21] We can generically check that vk_1 is a non-trivial group element by making sure $\mathsf{vk}_1 \cdot \mathsf{vk}_1 \neq \mathsf{vk}_1$, and in a prime order group any non-trivial element is a generator.

The two checks from the verification algorithm can be formalized as verification equations in the sense of Definition 17 as follows:

$$\alpha_x = (x \cdot V_1 + V_2) \cdot P - V_1^2 \quad \text{and} \quad \beta_x = V_1 \cdot P - Y \in \mathbb{Z}_p[V_1, V_2, P, Y].$$

By eliminating P, we get the following polynomial

$$\phi_x := V_1 \cdot \alpha_x - (x \cdot V_1 + V_2) \cdot \beta_x = (x \cdot V_1 + V_2) \cdot Y - V_1^3.$$

ϕ_x vanishes on the discrete logarithms of $\mathsf{vk}_1, \mathsf{vk}_2, \mathbf{y}, \pi$ whenever α_x and β_x vanish. Therefore, the collection $(\phi_x)_x$ is a family of verification equations for $\mathsf{vuf}^{\mathsf{DY}}$.

To prove that the unpredictability of $\mathsf{vuf}^{\mathsf{DY}}$ cannot be generically based on the hardness of NICA for an unbounded number of evaluation queries, we recall some notions from algebra and prove a simple mathematical lemma.

Remark 13. Let R be a domain, i.e. a zero divisors free commutative ring. Remember that the group of **units** $R^\times$ is defined as the set of multiplicatively invertible elements of R, i.e.

$$R^\times = \{a \in R \mid \exists b \in R : ab = 1\}.$$

An element $r \in R$ is called **irreducible**, if $r \neq 0$ and for each decomposition $r = ab$ with $a, b \in R$, we have $a \in R^\times$ or $b \in R^\times$.

Now, let k be a field and consider $k[X] := k[X_1, \ldots, X_n]$. Note that the group of units of $k[X]$ is exactly $k^\times = k \setminus \{0\}$. An element $f \in k[X]$ is said to **divide** $g \in k[X]$, if there is an $h \in k[X]$ s.t. $fh = g$. We will write $f \mid g$ in this case. Two elements $f, g \in k[X]$ are called **coprime** if we have for each $h \in k[X]$

$$h|f \wedge h|g \implies h \in k^\times.$$

f, g are coprime iff they do not share an irreducible component. If f, g are irreducible, they are coprime iff they are not scalar multiples of each other. The **greatest common divisor** $\gcd(f, g)$ of two polynomials $f, g \in k[X]$ is defined as some polynomial h of maximum degree s.t. h divides f and g. h is not determined uniquely but up to multiplication with a unit. (The greatest common divisor of 0 and 0 is 0).

Lemma 4. *Let $H, S \in \mathbb{Z}_p[Z_1, \ldots, Z_t]$ be non-zero. Let $x_1, \ldots, x_Q \in \mathbb{Z}_p$ be Q distinct scalars. Assume that $S \notin \mathbb{Z}_p \cdot H$.*

Then, the polynomials $(S + x_1 H), \ldots, (S + x_Q H)$ contain $Q - 1$ irreducible components, which are coprime to each other.

Proof. Let a be the greatest common divisor of S and H. Then, $\hat{S} := \frac{S}{a}$ and $\hat{H} := \frac{H}{a}$ are coprime polynomials. Since $S \notin \mathbb{Z}_p \cdot H$, $\hat{H}$ and $\hat{S}$ cannot both be scalars. For $i \in [Q]$, set

$$g_i := \frac{S + x_i H}{a} = \hat{S} + x_i \hat{H}.$$

We claim g_i, g_j are coprime if $i \neq j$. In fact, let d be a common divisor of g_i and g_j. Write $\hat{S} + x_i \cdot \hat{H} = b \cdot d$ and $\hat{S} + x_j \cdot \hat{H} = c \cdot d$ for fitting $b, c \in \mathbb{Z}_p[Z]$. Then, $(x_i - x_j) \cdot \hat{H} = \hat{S} + x_i \cdot \hat{H} - (\hat{S} + x_j \cdot \hat{H}) = (b - c) \cdot d$. Since $b \neq c$ and $x_i \neq x_j$

it follows that $d|\hat{H}$. As we assumed $d|(\hat{S} + x_i \cdot \hat{H})$, it follows that $d|\hat{S}$. Since $\hat{S}, \hat{H}$ are coprime, it follows $d \in \mathbb{Z}_p^\times$. Ergo, g_i, g_j are coprime whenever $i \neq j$.

We further claim that there is at most one $i \in [Q]$ s.t. $g_i \in \mathbb{Z}_p$. Assume – for the sake of contradiction – that $g_i, g_j \in \mathbb{Z}_p$ for two different i, j. Since $x_j \neq x_j$, it follows that $\hat{S}$ and $\hat{H}$ lie in $\mathbb{Z}_p$. However, we ruled this out in the beginning. Therefore, there is at most one $i \in [Q]$ s.t. g_i is constant.

W.l.o.g., we can now assume that the polynomials $g_1, \ldots, g_{Q-1}$ have positive degree and are all coprime to each other. In particular, each one of them contains an irreducible component that is coprime to each other g_i. Therefore, our claim follows.

We can now use Lemma 3 and Lemma 4 to prove that adversaries for the weak selective unpredictability of the VUF of Dodis & Yampolskiy [17], for an unbounded number of evaluation queries, cannot be generically transformed to NICA solvers.

Theorem 4. *Let* NICA *be an Uber-assumption of polynomial degree* $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$.

If NICA *is hard, then there is no generic PPT reduction* $\mathcal{R}$ *that solves samples of* NICA *with non-negligible advantage when given access to an adversary against the weak* $(d_{\mathsf{NICA}} + 2)$*-selective unpredictability of the VUF* $\mathsf{vuf}^{\mathsf{DY}} = (\mathsf{Gen}^{\mathsf{DY}}_{\mathsf{vuf}}, \mathsf{Eval}^{\mathsf{DY}}_{\mathsf{vuf}}, \mathsf{Verify}^{\mathsf{DY}}_{\mathsf{vuf}})$ *of Dodis & Yampolskiy [17].*

Proof. Set $Q = d_{\mathsf{NICA}} + 2$. Let $\mathcal{A}$ be a computationally unbounded adversary against the weak selective Q-unpredictability of $\mathsf{vuf}^{\mathsf{DY}}$ that proceeds as follows:

1. $\mathcal{A}$ receives $\mathsf{vk}, x_0, \ldots, x_Q, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$ from $\mathcal{R}$. If there are $i, j \in \{0, \ldots, Q\}$ s.t. $i \neq j$ and $x_i = x_j$, $\mathcal{A}$ returns $\perp$.
2. $\mathcal{A}$ checks if $\mathsf{Verify}^{\mathsf{DY}}_{\mathsf{vuf}}$ accepts each tuple $(\mathsf{vk}, x_i, \pi_i, \mathbf{y}_i)$. If $\mathsf{Verify}^{\mathsf{DY}}_{\mathsf{vuf}}$ rejects at least one $(\mathsf{vk}, x_i, \pi_i, \mathbf{y}_i)$, then $\mathcal{A}$ returns $\perp$.
3. $\mathcal{A}$ computes the discrete logarithms of the verification key to extract the secret key $s \in \mathbb{Z}_p$. If $s + x_0 \neq 0$, $\mathcal{A}$ uses this secret key to compute and output $\mathbf{y}_0 = e(\mathsf{vk}_1, \mathsf{vk}_1)^{\frac{1}{s+x_0}}$, otherwise it returns $\perp$.

Since $\mathcal{X} = \mathbb{Z}_p$ is exponentially large, the probability that $\mathcal{A}$ aborts in its first step in the real unpredictability game is negligible.

Assume, for the sake of contradiction, there is a generic PPT reduction $\mathcal{R}$ that – when given access to $\mathcal{A}$ – is able to solve NICA challenges.

We will show that there is an efficient meta-reduction $\mathcal{M}$ that can break NICA by using $\mathcal{R}$ and imitating the ideal adversary $\mathcal{A}$ for $\mathcal{R}$.

Let $\phi_x(V_1, V_2, P, Y) = (x \cdot V_1 + V_2) \cdot Y - V_1^3$ be the verification equation of $\mathsf{vuf}^{\mathsf{DY}}$ that we described above.

In the first successful interaction between $\mathcal{R}$ and $\mathcal{A}$, let $\mathcal{F} : \mathbb{Z}_p[V_1, V_2, P_1, \ldots, P_Q, Y_1, \ldots, Y_Q] \to \mathbb{Z}_p[Z]$ be the morphism of Eq. (31) that is derived from the algebraic explanations of the group elements $\mathsf{vk}_1, \mathsf{vk}_2, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$. Then, $\mathcal{F}(V_1), \mathcal{F}(V_2)$ are polynomials of degree d_{NICA}, while $\mathcal{F}(Y_1), \ldots, \mathcal{F}(Y_Q)$ are polynomials of degree $2d_{\mathsf{NICA}}$.

Due to Lemma 3, we know that each $\mathcal{F}(\phi_{x_i}) = (x_i \cdot \mathcal{F}(V_1) + \mathcal{F}(V_2)) \cdot \mathcal{F}(Y) - \mathcal{F}(V_1)^3$ must be – with overwhelming probability – the zero polynomial in $\mathbb{Z}_p[Z]$. That is, for each $i \in [Q]$ it holds that

$$(x_i \cdot \mathcal{F}(V_1) + \mathcal{F}(V_2)) \cdot \mathcal{F}(Y_i) = \mathcal{F}(V_1)^3$$

as polynomials in $\mathbb{Z}_p[Z]$. If $\mathcal{F}(V_2)$ is not a scalar multiple of $\mathcal{F}(V_1)$, i.e., there exists no scalar $s \in \mathbb{Z}_p$ such that $s \cdot \mathcal{F}(V_1) = \mathcal{F}(V_2)$, then – by Lemma 4 – the polynomials $(x_1 \cdot \mathcal{F}(V_1) + \mathcal{F}(V_2)), \ldots, (x_Q \cdot \mathcal{F}(V_1) + \mathcal{F}(V_2))$ contain at least $(d_{\mathsf{NICA}} + 1)$ irreducible coprime components that all divide $\mathcal{F}(V_1)^3$. Since those components are irreducible, they all divide $\mathcal{F}(V_1)$. However, $\mathcal{F}(V_1)$ is of degree d_{NICA} and cannot be divided by $Q - 1 = d_{\mathsf{NICA}} + 1$ coprime irreducible components, which are no scalars. Therefore, $\mathcal{F}(V_2)$ must be a scalar multiple of $\mathcal{F}(V_1)$, i.e., there is one $s \in \mathbb{Z}_p$ s.t. we have the equality of formal polynomials

$$\mathcal{F}(V_2) = s \cdot \mathcal{F}(V_1).$$

The scalar s is the secret key of $\mathsf{vuf}^{\mathsf{DY}}$. Since $\mathcal{M}$ can compute $\mathcal{F}$ (recall that $\mathcal{M}$ has access to the algebraic explanations of the group elements issued by $\mathcal{R}$), it can compute s on its own.

Therefore, the meta-reduction $\mathcal{M}$ – which uses $\mathcal{R}$ to solve NICA – can efficiently simulate the ideal adversary $\mathcal{A}$ in its first successful interaction with $\mathcal{R}$. Subsequently, $\mathcal{M}$ can simulate the ideal adversary $\mathcal{A}$ in each call of $\mathcal{R}$, and therefore, $\mathcal{M}$'s advantage on solving NICA equals $\mathcal{R}$'s advantage when given oracle access to $\mathcal{A}$.

Since we assumed that NICA is hard and since $\mathcal{M}$ is a PPT algorithm, it must follow that $\mathcal{R}$'s advantage on solving NICA is negligible.

6.2 Preliminaries on Groebner Bases and Projective Algebraic Geometry

To prove Theorem 3, we need to introduce the tool of so called *Groebner Bases* and the *Projective Extension Theorem* [15]. Therefore, we will give here an overview of some notions and results regarding Groebner bases and algebraic projective geometry, which we will be using in Section 6.3.

We assume that the reader is familiar with the notion of *ideals* of rings. An ideal I of a ring R is a subset of R that is nonempty, closed under addition of elements of the ideal and closed under multiplication with arbitrary elements of R. The ideal *generated* by a set $S \subset R$ is the smallest ideal of R that contains S. We will denote this ideal by (S).

Definition 19 (Monomial Orders). *Let $R = k[X_1, \ldots, X_n]$ be the polynomial ring of the variables $X_1, \ldots, X_n$ with coefficients in a field k.*

*A **monomial order** $\leq$ on R is a linear order on the set of monomials $\mathrm{Mon} := \{X_1^{\alpha_1} \cdots X_n^{\alpha_n} \mid \alpha_1, \ldots, \alpha_n \in \mathbb{N}_0\}$. We call $\leq$ **admissible**, if the following two properties hold:*

1. $\forall m \in \mathrm{Mon} : 1 \leq m$.
2. $\forall a, b, c \in \mathrm{Mon} : a \leq b \implies a \cdot c \leq b \cdot c$.

Definition 20 (Leading Terms). *Let $\leq$ be an admissible monomial order on $R = k[X_1, \ldots, X_n]$.*

Each $f \in R$ can be written as

$$f = \sum_{\alpha \in \mathbb{N}_0^n} c_\alpha \cdot X^\alpha$$

where only finitely many c_α are non-zero. Let X^β be the maximum of the set $\{X^\alpha \mid c_\alpha \neq 0\}$ according to $\leq$.

*Then, the **leading term** $\mathrm{lt}(f)$ of f is defined as $c_\beta X^\beta$.*

Definition 21 (Groebner Bases). *Let $\leq$ be an admissible monomial order on $R = k[X_1, \ldots, X_n]$ and let $I \subset R$ be an ideal.*

*A set $G \subset I$ is called a **Groebner basis** of I, if for each $f \in I$ there is a $g \in G$ s.t. $\mathrm{lt}(g) | \mathrm{lt}(f)$.*

Note, that each Groebner basis of I is, in particular, a generating set of I.

It is well known that, given a Groebner basis, the ideal membership problem can be efficiently solved. However, it is usually infeasible to compute a Groebner basis for a given generator set. Dubé [18] showed that the degrees of the elements a Groebner basis can be bounded from above by a function that is double exponential in the number of variables.

Theorem 5 (Dubé [18]). *Let $I \subset k[X_1, \ldots, X_n]$ be an ideal generated by degree d polynomials $f_1, \ldots, f_m \in k[X_1, \ldots, X_n]$.*

Then, there is a finite Groebner basis G of I such that we have for each $g \in G$

$$\deg(g) \leq 2\left(\frac{d^2}{2} + d\right)^{2^{n-1}} \in O(d^{2^n}).$$

A monomial order of special interest is the so-called *lexicographical order* $\leq_{\mathsf{lex}}$, which is given by

$$\begin{aligned} & X_1^{\alpha_1} \cdots X_n^{\alpha_n} \leq_{\mathsf{lex}} X_1^{\beta_1} \cdots X_n^{\beta_n} \\ :\iff & \exists i \in [n+1] : (\forall j < i : \alpha_j = \beta_j) \wedge (i = n+1 \vee \alpha_i > \beta_i). \end{aligned}$$

The lexicographical order is useful for eliminating variables. For $i \in \{0, \ldots, n\}$, set $I_i := I \cap k[X_1, \ldots, X_i]$. Then I_i is an ideal of the polynomial ring $k[X_1, \ldots, X_i]$. The elimination theorem states that each Groebner basis of I with respect to $\leq_{\mathsf{lex}}$ contains a Groebner basis of I_i.

Theorem 6 (Elimination Theorem). *Let $I \subset k[X_1, \ldots, X_n]$, let $\leq_{\mathsf{lex}}$ be the lexicographical order as explained above (with $X_1 <_{\mathsf{lex}} X_2 <_{\mathsf{lex}} \ldots <_{\mathsf{lex}} X_n$).*

For $i \in \{0, \ldots, n\}$, set $I_i := I \cap k[X_1, \ldots, X_i]$. Let G be a Groebner basis of I with respect to $\leq_{\mathsf{lex}}$.

Then, $G \cap k[X_1, \ldots, X_i]$ is a Groebner basis of I_i with with respect to $\leq_{\mathsf{lex}}$ restricted to $k[X_1, \ldots, X_i]$.

A proof of the Elimination Theorem can be found in [15] (note, that we reversed the order of the variables $X_1, \ldots, X_n$ in $<_{\mathsf{lex}}$ here).

We can now combine the above theorems to get the following corollary:

Corollary 2. *Let $I \subset k[X_1, \ldots, X_n]$ be an ideal generated by elements $f_1, \ldots, f_m$ of degree d.*

Then, if the principal ideal $I \cap k[X_1]$ is not zero, it is generated by an element of degree $\leq 2\left(\frac{d^2}{2} + d\right)^{2^{n-1}}$.

We assume that the reader is familiar with the notion of *projective spaces*.If k is a field, we can define a relation $\sim$ on $k^{n+1} \setminus \{0\}$ by

$$a \sim b :\iff \exists \lambda \in k^{\times} : \lambda a = b$$

for $a, b \in k^{n+1} \setminus \{0\}$. Equivalence classes of $\sim$ are denoted as $[x_0 : \ldots : x_n]$ for $(x_0, \ldots, x_n) \in k^{n+1} \setminus \{0\}$. The relation $\sim$ is an equivalence relation on $k^{n+1} \setminus \{0\}$ and its quotient set $\mathbb{P}^n(k) := (k^{n+1} \setminus \{0\}) / \sim$ is called the **projective space** of dimension n.

One can imagine $\mathbb{P}^n(k)$ to be the affine space k^n together with a set of additional points that describe k^n's geometry at *infinity*. E.g., $\mathbb{P}^1(\mathbb{R}) \approx \mathbb{R} \cup \{\infty\}$ can be imagined as the line $\mathbb{R}$ together with a point ∞ that connects both ends of $\mathbb{R}$ and turns the line into a sphere. $\mathbb{P}^2(\mathbb{R}) \approx \mathbb{R}^2 \cup \mathbb{P}^1(\mathbb{R})$ can be somewhat imagined as the Euclidean plane together with a sphere at its infinity boundary that turns the plane into a compact manifold.

Definition 22. *A polynomial $f \in k[X_0, X_1, \ldots, X_n, Y_1, \ldots, Y_m]$ is called $(X_0, X_1, \ldots, X_n)$-**homogenous**, if there is a $d \in \mathbb{N}_0$ s.t. f can be written as*

$$f = \sum_{\alpha \in \mathbb{N}_0^{n+1} : |\alpha| = d} X^{\alpha} \cdot g_{\alpha}$$

for $g_{\alpha} \in k[Y_1, \ldots, Y_m]$.

*An ideal $I \subseteq k[X_0, X_1, \ldots, X_n, Y_1, \ldots, Y_m]$ is called $(X_0, X_1, \ldots, X_n)$-**homogenous** if it is generated by $(X_0, X_1, \ldots, X_n)$-homogenous polynomials.*

A polynomial $f \in k[X_0, X_1, \ldots, X_n, Y_1, \ldots, Y_m]$ that is X-homogenous cannot be evaluated on points of $\mathbb{P}^n(k) \times k^m$. However, for a point

$$([x_0 : \ldots : x_n], (y_1, \ldots, y_m)) \in \mathbb{P}^n(k) \times k^m,$$

the statement

$$f([x_0 : \ldots : x_n], (y_1, \ldots, y_m)) = f(x_0, \ldots, x_n, y_1, \ldots, y_m) = 0$$

is independent of the concrete representation $(x_0, \ldots, x_n)$ of $[x_0 : \ldots : x_n]$, and therefore well-defined.

Definition 23. *Let $\overline{k}$ be the algebraic closure of k. If $S \subset k[X_0, X_1, \ldots, X_n, Y_1, \ldots, Y_m]$ is a set of X-homogenous polynomials, we define the variety $\mathcal{V}(S) \subset \mathbb{P}^n(\overline{k}) \times \overline{k}^m$ by*

$$\mathcal{V}(S) := \{([x], y) \in \mathbb{P}^n(\overline{k}) \times \overline{k}^m \mid \forall f \in S : \; f([x], y) = 0\}.$$

We will now cite a result, presented in [15], which will be – for the sake of simplicity – the combination of two different results of [15] (Theorem 6 and Proposition 8 of the chapter *Projective Algebraic Geometry*):

Theorem 7 (Projective Extension Theorem).
Let $f_1, \ldots, f_s \in k[X_0, \ldots, X_n, Y_1, \ldots, Y_m]$ be X-homogenous polynomials. Let

$$\begin{aligned} \pi : \mathbb{P}^n(\overline{k}) \times \overline{k}^m &\longrightarrow \overline{k}^m \\ ([x], y) &\longmapsto y \end{aligned}$$

be the projection to the last m coordinates. For each $i \in \{0, \ldots, n\}$, define the following morphism of k-algebras

$$\begin{aligned} d_i : k[X_0, \ldots, X_n, Y_1, \ldots, Y_m] &\longrightarrow k[X_0, \ldots, X_{i-1}, X_{i+1}, \ldots X_n, Y_1, \ldots, Y_m] \\ f(X_0, \ldots, X_n, Y_1, \ldots, Y_m) &\longmapsto f(X_0, \ldots, X_{i-1}, 1, X_{i+1}, \ldots X_n, Y_1, \ldots, Y_m). \end{aligned}$$

Then, we have

$$\pi(\mathcal{V}(f_1, \ldots, f_s)) = \mathcal{V}(\hat{I})$$

where the ideal $\hat{I}$ is given by

$$\hat{I} = \bigcap_{i=0}^{n} \left((d_i(f_1), \ldots, d_i(f_s)) \cap k[Y_1, \ldots, Y_m] \right).$$

Now, one can combine the projective extension theorem with our knowledge of Groebner bases to get the following corollary:

Corollary 3. *Let $f_1, \ldots, f_s \in k[X_0, \ldots, X_n, Y]$ be X-homogenous polynomials of total degree d.*

Then, either $\pi(\mathcal{V}(f_1, \ldots, f_s)) = \overline{k}$ or there is a non-zero polynomial $g \in k[Y]$ of degree $\leq (n+1) \cdot 2 \left(\frac{d^2}{2} + d \right)^{2^n}$ s.t.

$$\pi(\mathcal{V}(f_1, \ldots, f_s)) = \mathcal{V}(g).$$

Proof. Set $A_i := (d_i(f_1), \ldots, d_i(f_s)) \cap k[Y]$. The projective extension theorem states that

$$\pi(\mathcal{V}(f_1, \ldots, f_s)) = \mathcal{V}(\hat{I})$$

for

$$\hat{I} = \bigcap_{i=0}^{n} A_i.$$

According to Corollary 2, A_i is either zero or generated by an element g_i of degree $\leq 2 \left(\frac{d^2}{2} + d \right)^{2^n}$.

If one of the A_i is the zero ideal, then $\hat{I} = 0$ and $\mathcal{V}(\hat{I}) = \overline{k}$. Otherwise, since $k[Y]$ is a principal ideal domain, the intersection

$$\hat{I} = A_0 \cap \ldots \cap A_n = (g_0) \cap \ldots \cap (g_n)$$

is generated by the smallest common multiple g of $g_0, \ldots, g_n$. The degree of g is at most $\deg(g_0) + \ldots + \deg(g_n) \leq (n+1) \cdot 2 \left(\frac{d^2}{2} + d \right)^{2^n}$.

6.3 Proof of Theorem 3

We are finally going to prove Theorem 3.

For this end, let NICA be an Uber-assumption of degree $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$ and size $q = 1 + q_1 + q_2 = \sqrt{\log\log w}$ for some $w \in \mathsf{poly}(\lambda)$.

Let $f_{A_1}, \ldots, f_{A_{q_1}}, f_{B_1}, \ldots, f_{B_{q_2}} \in \mathbb{Z}_p[Z]$ be the polynomials that are used by $\mathcal{D}$ to compute the exponents of the group elements of a sample of NICA. We will now define when we call a polynomial a *constructible target exponent*.

Definition 24. *Let $W_S \subset \mathbb{Z}_p[Z]$ be the $\mathbb{Z}_p$-vector space generated by the elements $1, f_{A_1}, \ldots, f_{A_{q_1}}$, i.e.*

$$W_S := \operatorname{span}_{\mathbb{Z}_p}\{1, f_{A_1}, \ldots, f_{A_{q_1}}\}.$$

Set further

$$W_T := W_S^2 + \operatorname{span}_{\mathbb{Z}_p}\{f_{B_1}, \ldots, f_{B_{q_2}}\}.$$

Polynomials in W_T are called ***constructible target exponents****.*

The intuition behind this notion is that, if we were to apply a hybrid step and replace the groups $\mathbb{G}, \mathbb{G}_T$ by groups $\mathbb{G}^Z, \mathbb{G}_T^Z$ that encode elements of $\mathbb{Z}_p[Z]$, then W_S contains the exponents of all group elements in $\mathbb{G}^Z$ that can be computed by a generic algorithm that is given the group elements $\mathbf{g}, \mathbf{g}^{f_{A_1}(Z)}, \ldots, \mathbf{g}^{f_{A_{q_1}}(Z)}$, while W_T contains the exponents of all group elements in $\mathbb{G}_T^Z$ that can be computed by a generic algorithm that is given the group elements $\mathbf{g}, \mathbf{g}^{f_{A_1}(Z)}, \ldots, \mathbf{g}^{f_{A_{q_1}}(Z)}, e(\mathbf{g}, \mathbf{g})^{f_{B_1}(Z)}, \ldots, e(\mathbf{g}, \mathbf{g})^{f_{B_{q_2}}(Z)}$. Note, that W_T is a vector space of dimension $\leq (1 + q_1)^2 + q_2 \leq q^2 = \log\log w$.

In Theorem 3, we stated that there is no generic reduction that – when given access to an adversary for the weak selective unpredictability of a parametrized rational VUF of constant evaluation degree – can efficiently solve NICA if NICA is hard.

Now, let $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ be a parametrized rational VUF of constant evaluation degree d_{vuf}. Remember that, since vuf is parametrized rational, there must be two universal polynomials $\sigma, \rho \in \mathbb{Z}_p[V_S, V_T, X]$ of degree d_{vuf} s.t. the family $(\phi_x)_{x \in \mathbb{Z}_P}$ of polynomials

$$\phi_x(V, P, Y) := \rho(V, x) \cdot Y - \sigma(V, x)$$

is a family of verification equations for vuf.

Let $\mathcal{R}$ be a generic reduction for the weak selective unpredictability of vuf. $\mathcal{R}$ attempts to solve a NICA challenge $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \xleftarrow{\$} \mathcal{D}(1^\lambda)$ by making oracle queries to an adversary $\mathcal{A}$ on the weak Q-selective unpredictability of vuf. To prove Theorem 3, we will describe here a meta-reduction $\mathcal{M}$ that will use $\mathcal{R}$ and efficiently simulate an ideal adversary $\mathcal{A}$ in the unpredictability game with $\mathcal{R}$. The ideal adversary $\mathcal{A}$ is described as follows:

1. $\mathcal{A}$ receives as input $\mathsf{vk}, x_0, \ldots, x_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q, \pi_1, \ldots, \pi_Q$ by $\mathcal{R}$.
2. It checks that $\mathsf{Verify}_{\mathsf{vuf}}$ accepts each tuple $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$. If one of the tuples is not accepted, $\mathcal{A}$ returns $\perp$.
3. If there are $i, j \in \{0, \ldots, Q\}$ s.t. $i \neq j$ and $x_i = x_j$, then $\mathcal{A}$ returns $\perp$.
4. Otherwise, it extracts the exponents $\overrightarrow{v}$ of the group elements of vk relative to some group generator $\mathbf{h} \in \mathbb{G}$ of vk.

5. $\mathcal{A}$ computes $\rho(\overrightarrow{v}, x_0)$. If $\rho(\overrightarrow{v}, x_0) = 0$, then $\mathcal{A}$ returns $\perp$.
6. Otherwise, $\mathcal{A}$ computes $\sigma(\overrightarrow{v}, x_0)$ and outputs $\mathbf{y}_0 = e(\mathbf{h}, \mathbf{h})^{\frac{\sigma(\overrightarrow{v}, x_0)}{\rho(\overrightarrow{v}, x_0)}}$ where $\mathbf{h}$ is the group generator it used in step 4.

Note, that the probability of $\mathcal{A}$ returning $\perp$ in the steps 1 - 4 is negligible in the real weak Q-selective unpredictability game of vuf, since we assume that vuf is correct and since the probability of two x's drawn uniformly from $\mathbb{Z}_p$ to be equal is negligible.

If $\mathcal{A}$ does not return $\perp$ in the steps 1 - 4, it will always win, since $\mathbf{y}_0$ is the only image value in $\mathbb{G}_T$ for that there can exist a proof π_0 s.t. $\mathsf{Verify}_{\mathsf{vuf}}$ accepts the tuple $(\mathsf{vk}, x_0, \mathbf{y}_0, \pi_0)$. In fact, if $\mathsf{Verify}_{\mathsf{vuf}}$ accepts $(\mathsf{vk}, x_0, \mathbf{y}_0, \pi_0)$, then ϕ_{x_0} must vanish on the exponents $\overrightarrow{v}, y_0$ of $\mathsf{vk}, \mathbf{y}_0$ relative to $\mathbf{h}, e(\mathbf{h}, \mathbf{h})$, which implies

$$\rho(\overrightarrow{v}, x_0) \cdot y_0 = \sigma(\overrightarrow{v}, x_0)$$

and in particular $\mathsf{dlog}_{e(\mathbf{h},\mathbf{h})}(\mathbf{y}_0) = y_0 = \sigma(\overrightarrow{v}, x_0)/\rho(\overrightarrow{v}, x_0)$. Therefore, $\mathcal{A}$ has an overwhelming win probability in the real weak Q-selective unpredictability game of vuf.

Again, in each interaction between $\mathcal{R}$ and $\mathcal{M}$, we let $\mathcal{F}$ be a ring morphism of type

$$\mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$$

where

$$\begin{aligned} &\mathbb{Z}_p[V, P, Y] := \\ &\quad \mathbb{Z}_p[(V_{S,j})_{j \in [n_1]}, (V_{T,j})_{j \in [n_2]}, (P_{S,i,j})_{i \in [Q], j \in [u_1]}, (P_{T,i,j})_{i \in [Q], j \in [u_2]}, (Y_i)_{i \in [Q]}]. \end{aligned}$$

$\mathcal{F}$ is induced by the algebraic representations of the exponents of $\mathsf{vk}, \pi_1, \dots, \pi_Q$, $\mathbf{y}_1, \dots, \mathbf{y}_Q$ and by the polynomials $f_{A_1}, \dots, f_{B_1}, \dots$, which specify the Uber-assumption NICA. Again, $\mathcal{F}$ maps each single variable to a polynomial in $\mathbb{Z}_p[Z]$ of degree $\leq 2d_{\mathsf{NICA}}$. Since $\mathcal{M}$ knows the polynomials $f_{A_1}, \dots, f_{B_1}$ and since $\mathcal{M}$ can extract algebraic representations of $\mathsf{vk}, \pi_1, \dots, \pi_Q, \mathbf{y}_1, \dots, \mathbf{y}_Q$ out of $\mathcal{R}$, $\mathcal{M}$ can efficiently evaluate $\mathcal{F}$ on elements of $\mathbb{Z}_p[V, P, Y]$ of constant degree.

Now, our meta-reduction $\mathcal{M}$ simulates the ideal adversary $\mathcal{A}$ as follows:

1. When receiving the verification key vk, the inputs $x_0, \dots, x_Q \in \mathbb{Z}_p$, the proofs $\pi_1, \dots, \pi_Q$ and the image values $\mathbf{y}_1, \dots, \mathbf{y}_Q$, $\mathcal{M}$ checks that $\mathsf{Verify}_{\mathsf{vuf}}$ accepts each tuple $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ for $i \in [Q]$. If one of the tuples is not accepted, $\mathcal{M}$ returns $\perp$.
2. If there are $i, j \in \{0, \dots, Q\}$ s.t. $i \neq j$ and $x_i = x_j$, then $\mathcal{M}$ returns $\perp$.
3. $\mathcal{M}$ extracts the map $\mathcal{F} : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$ that explains algebraically each group element $\mathcal{M}$ received by $\mathcal{R}$.
4. $\mathcal{M}$ computes $\rho(\mathcal{F}(V), x_0) \in \mathbb{Z}_p[Z]$. $\mathcal{M}$ can do so, since each $\mathcal{F}(V_{S,i})$ resp. $\mathcal{F}(V_{T,i})$ is sparse and ρ is a polynomial of constant degree d_{vuf}. If $\rho(\mathcal{F}(V), x_0) = 0$, then $\mathcal{M}$ returns $\perp$.
5. Now, $\mathcal{M}$ computes $\sigma(\mathcal{F}(V), x_0)$. By mere linear algebra, $\mathcal{M}$ checks if

   $$\sigma(\mathcal{F}(V), x_0) \in \rho(\mathcal{F}(V), x_0) \cdot W_T.$$

   It can do so, since $\rho(\mathcal{F}(V), x_0) \cdot W_T$ is a vector space of dimension $O(\log\log(w))$ that is generated by sparse polynomials.

6. If $\sigma(\mathcal{F}(V), x_0) \in \rho(\mathcal{F}(V), x_0) \cdot W_T$, $\mathcal{M}$ computes

$$\mathbf{y}_0 = e(\mathbf{g}, \mathbf{g})^{\sigma(\mathcal{F}(V)(\overrightarrow{z}), x_0) / \rho(\mathcal{F}(V)(\overrightarrow{z}), x_0)}$$

   as follows:
   By mere linear algebra, $\mathcal{M}$ computes scalars $(\alpha_{i,j})_{i,j=0,\dots,q_1} \subset \mathbb{Z}_p$, $\beta_1, \dots, \beta_{q_2} \in \mathbb{Z}_p$, s.t.

$$\sigma(\mathcal{F}(V), x_0) = \rho(\mathcal{F}(V), x_0) \cdot \left(\sum_{i,j=0}^{q_1} \alpha_{i,j} \cdot f_{A_i} \cdot f_{A_j} + \sum_{i=1}^{q_2} \beta_i \cdot f_{B_i} \right)$$

   (where we set $f_{A_0}(Z) := 1$ for simplicity).
   We now have for the rational functions in $\mathbb{Z}_p(Z)$

$$\frac{\sigma(\mathcal{F}(V), x_0)}{\rho(\mathcal{F}(V), x_0)} = \sum_{i,j=0}^{q_1} \alpha_{i,j} \cdot f_{A_i} \cdot f_{A_j} + \sum_{i=1}^{q_2} \beta_i \cdot f_{B_i}.$$

   Let $\overrightarrow{z} \in \mathbb{Z}_p^t$ be the concrete values drawn by $\mathcal{D}$ and let $\overrightarrow{v}$ be the exponents of vk relative to the basis $\mathbf{g}, e(\mathbf{g}, \mathbf{g})$. If $\rho(\mathcal{F}(V), x_0)(\overrightarrow{z}) = \rho(\overrightarrow{v}, x_0) \neq 0$, then we have

$$\begin{aligned} \frac{\sigma(\mathcal{F}(V), x_0)}{\rho(\mathcal{F}(V), x_0)}(\overrightarrow{z}) &= \frac{\sigma(\overrightarrow{v}, x_0)}{\rho(\overrightarrow{v}, x_0)} \\ &= \sum_{i,j=0}^{q_1} \alpha_{i,j} \cdot f_{A_i}(\overrightarrow{z}) \cdot f_{A_j}(\overrightarrow{z}) + \sum_{i=1}^{q_2} \beta_i \cdot f_{B_i}(\overrightarrow{z}) \\ &= \sum_{i,j=0}^{q_1} \alpha_{i,j} a_i a_j + \sum_{i=1}^{q_2} \beta_i b_i \end{aligned}$$

   where $a_0 := 1, a_1, \dots, a_{q_1}, b_1, \dots, b_{q_2}$ are exponents of the elements of the challenge $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}})$ relative to the basis $\mathbf{g}$ resp. $e(\mathbf{g}, \mathbf{g})$.
   Therefore, $\mathcal{M}$ computes and outputs

$$e(\mathbf{g}, \mathbf{g})^{\frac{\sigma(\overrightarrow{v}, x_0)}{\rho(\overrightarrow{v}, x_0)}} = \prod_{i,j=0}^{q_1} e(\mathbf{g}^{a_i}, \mathbf{g}^{a_j}) \cdot \prod_{i=1}^{q_2} e(\mathbf{g}, \mathbf{g})^{b_i}.$$

7. Otherwise, $\mathcal{M}$ returns $\perp$.

It is easy to show that – if $\sigma(\mathcal{F}(V), x_0) \in \rho(\mathcal{F}(V), x_0) \cdot W_T$ – the meta-reduction $\mathcal{M}$ will succeed in simulating $\mathcal{A}$:

Lemma 5. *Assume that $d_{\mathsf{vuf}} \in O(1)$ and $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$.*

If $\sigma(\mathcal{F}(V), x_0) \in \rho(\mathcal{F}(V), x_0) \cdot W_T$, then – with overwhelming probability – $\mathcal{M}$ simulates the behaviour of $\mathcal{A}$ in its interaction with $\mathcal{R}$.

Proof. Let $z_1, \ldots, z_t$ be the real values for the variables $Z_1, \ldots, Z_t$ that have been drawn by $\mathcal{D}$. We then have for the exponents of vk relative to the generator $\mathbf{g}$

$$\begin{aligned} v_{S,1} &= \mathcal{F}(V_{S,1})(\overrightarrow{z}), \ldots, v_{S,n_1} = \mathcal{F}(V_{S,n_1})(\overrightarrow{z}), \\ v_{T,1} &= \mathcal{F}(V_{T,1})(\overrightarrow{z}), \ldots, v_{T,n_2} = \mathcal{F}(V_{T,n_2})(\overrightarrow{z}). \end{aligned}$$

It is easy to see that – if $\rho(\overrightarrow{v}, x_0) \neq 0$ – then $\mathcal{M}$ and $\mathcal{A}$ will output the same element $\mathbf{y}_0$.

However, if $\rho(\overrightarrow{v}, x_0) = 0$ and $\rho(\mathcal{F}(V), x_0) \neq 0$, then $\mathcal{A}$ will return $\perp$ while $\mathcal{M}$ may not return $\perp$. Since $\overrightarrow{z} \in \mathbb{Z}_p^t$ is drawn uniformly at random by $\mathcal{D}$ and since $\rho(\mathcal{F}(V), x_0)$ is a polynomial of degree $\leq 2 \cdot d_{\mathsf{vuf}} \cdot d_{\mathsf{NICA}}$ in Z, the probability that

$$\rho(v, x_0) = \rho(\mathcal{F}(V), x_0)(\overrightarrow{z}) = 0$$

is bounded by $\frac{2 \cdot d_{\mathsf{vuf}} \cdot d_{\mathsf{NICA}}}{p} \in \mathsf{negl}(\lambda)$.

So, it suffices to show that, by setting Q high enough, we can force $\frac{\sigma(\mathcal{F}(V), x_0)}{\rho(\mathcal{F}(V), x_0)}$ to lie in W_T. Therefore, we will prove the following theorem.

Theorem 8. *Assume that* $d_{\mathsf{vuf}} \in O(1)$ *and* $d_{\mathsf{NICA}} \in \mathsf{poly}(\lambda)$. *Let there be a* $w \in \mathsf{poly}(\lambda)$*, s.t.* NICA *is of size* $q = \sqrt{\log \log w}$.

Then, if $Q > 2 \cdot (q^2 + 1) \cdot \left(\frac{(d_{\mathsf{vuf}}+1)^2}{2} + d_{\mathsf{vuf}} + 1 \right)^{2^{q^2}}$*, one has in the game between* $\mathcal{R}$ *and* $\mathcal{M}$ *either*

$$\frac{\sigma(\mathcal{F}(V), x_0)}{\rho(\mathcal{F}(V), x_0)} \in W_T$$

or

$$\rho(\mathcal{F}(V), x_0) = 0.$$

To prove this theorem, let $\omega := \dim_{\mathbb{Z}_p} W_T \leq q^2 = \log \log w$ be the vector space dimension of W_T and let $u_1, \ldots, u_\omega \in \mathbb{Z}_p[Z]$ be a basis of W_T. Since W_T is generated by sparse polynomials of degree $\leq 2d_{\mathsf{NICA}}$, $u_1, \ldots, u_\omega$ are sparse of degree $\leq 2d_{\mathsf{NICA}}$, too.

Now, we introduce new formal variables $H_0, \ldots, H_\omega$ and let

$$g \in \mathbb{Z}_p[H, X, Z] := \mathbb{Z}_p[H_0, \ldots, H_\omega, X, Z_1, \ldots, Z_t].$$

We can consider g as a polynomial in Z whose coefficients are polynomials in H and X and write it as

$$g = \sum_{\alpha \in \mathbb{N}_0^t} c_\alpha \cdot Z^\alpha$$

with $c_\alpha \in \mathbb{Z}_p[H, X]$ where only finitely many c_α are non-zero. If we set

$$\mathcal{T}(g) := \{c_\alpha \mid \alpha \in \mathbb{N}_0^{\omega+1}\}$$

we get a map

$$\mathcal{T} : \mathbb{Z}_p[H, X, Z] \longrightarrow \mathcal{P}(\mathbb{Z}_p[H, X]).$$

If g is sparse, then $\mathcal{T}(g)$ only contains a polynomial number of elements.

Note, that we have for each $(h_0, \ldots, h_\omega, x) \in \overline{\mathbb{Z}_p}^{\omega+2}$

$$\begin{aligned} g(\overrightarrow{h}, x, Z) = 0 &\iff \forall c \in \mathcal{T}(g): \ c(\overrightarrow{h}, x) = 0 \\ &\iff (\overrightarrow{h}, x) \in \mathcal{V}(\mathcal{T}(g)) \end{aligned}$$

where $\mathcal{V}(\mathcal{T}(g))$ is the affine zero locus of the elements of $\mathcal{T}(g)$.

Now, let $x_0, \ldots, x_Q \in \mathbb{Z}_p$, $\mathsf{vk}, \mathbf{y}_1, \ldots, \mathbf{y}_Q, \pi_1, \ldots, \pi_Q$ and $\mathcal{F} : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$ be the products of the interaction between $\mathcal{R}$ and $\mathcal{M}$. We can assume that $x_0, \ldots, x_Q$ are pairwise distinct.

Set

$$g(H, X, Z) := \rho(\mathcal{F}(V)(Z), X) \cdot \left(\sum_{j=1}^{\omega} H_j \cdot u_j \right) - \sigma(\phi(V)(Z), X) \cdot H_0$$

where $u_1, \ldots, u_\omega$ is the basis of W_T. The polynomial $g \in \mathbb{Z}_p[H, X, Z]$ is homogenous in H. It is easy to see that each element of $\mathcal{T}(g) \subset \mathbb{Z}_p[H, X]$ is homogenous in H, too. Therefore, we can consider the semi-projective variety

$$\mathcal{V}(\mathcal{T}(g)) = \{([h], x) \in \mathbb{P}^\omega(\overline{\mathbb{Z}_p}) \times \overline{\mathbb{Z}_p} \mid \forall c \in \mathcal{T}(g): \ c(\overrightarrow{h}, x) = 0\}$$

where $\overline{\mathbb{Z}_p}$ is the algebraic closure of $\mathbb{Z}_p$.

Since ρ and σ are of degree d_{vuf} in X, the total degree of each element of $\mathcal{T}(g)$ is bounded by $d_{\mathsf{vuf}} + 1$. Corollary 3 now states for the projection $\pi_{\overline{\mathbb{Z}_p}} : \mathbb{P}^\omega(\overline{\mathbb{Z}_p}) \times \overline{\mathbb{Z}_p} \to \overline{\mathbb{Z}_p}$ onto the last coordinate one of the following two cases occur:

- either $\pi_{\overline{\mathbb{Z}_p}}(\mathcal{V}(\mathcal{T}(g))) = \overline{\mathbb{Z}_p}$,
- or there is a non-zero polynomial $g' \in \mathbb{Z}_p[X]$ of degree

$$\deg g' \leq (\omega + 1) \cdot 2 \cdot \left(\frac{(d_{\mathsf{vuf}} + 1)^2}{2} + d_{\mathsf{vuf}} + 1 \right)^{2^\omega} < Q$$

  such that we have

$$\pi_{\overline{\mathbb{Z}_p}}(\mathcal{V}(\mathcal{T}(g))) = \mathcal{V}(g').$$

Assume – for the sake of contradiction – that the second case would occur. Then, the polynomial system $\mathcal{T}(g)$ would only be solvable for at most $\deg g'$ different x values. However, we asked the reduction for evaluation queries for at least $Q > \deg g'$ different x values.

In fact, let $i \in [Q]$. Since $\mathcal{F}(Y_i)$ must lie in W_T, there must be scalars $h_1, \ldots, h_\omega \in \mathbb{Z}_p$, s.t. we have $\mathcal{F}(Y_i) = \sum_{i=1}^{\omega} h_i \cdot u_i(Z)$ and therefore

$$g(1, h_1, \ldots, h_\omega, x_i, Z) = \rho(\mathcal{F}(V), x_i) \cdot \mathcal{F}(Y_i) - \sigma(\mathcal{F}(V), x_i) = 0,$$

since $(\rho(V, x) \cdot Y - \sigma(V, x))_x$ is a family of verification equations for vuf.

In particular, for each $i \in [Q]$, there are $h_1, \ldots, h_\omega \in \mathbb{Z}_p$ s.t. the point $([1 : h_1 : \ldots : h_Q], x_i)$ lies in $\mathcal{V}(\mathcal{T}(g))$. Therefore, $\pi_{\overline{\mathbb{Z}_p}}(\mathcal{V}(\mathcal{T}(g)))$ must contain the Q

different points $x_1, \dots, x_Q \in \mathbb{Z}_p$. Therefore, $\pi_{\overline{\mathbb{Z}_p}}(\mathcal{V}(\mathcal{T}(g)))$ cannot be the zero locus of a polynomial of degree $< Q$.

It follows that the first case must hold, i.e., $\pi_{\overline{\mathbb{Z}_p}}(V(\mathcal{T}(g))) = \overline{\mathbb{Z}_p}$. This implies, that for $x_0 \in \overline{\mathbb{Z}_p}$, there must exist $h_0, \dots, h_\omega \in \overline{\mathbb{Z}_p}$, not all zero, such that we have

$$\rho(\mathcal{F}(V), x_i) \cdot y_0(Z) - \sigma(\mathcal{F}(V), x_i) \cdot h_0 = 0$$

for $y_0(Z) = \sum_{i=1}^{Q} h_i \cdot u_i(Z)$. Again, we distinguish two cases:

- if $h_0 = 0$, then we have $\rho(\mathcal{F}(V), x_i) = 0$, since $y_0(Z) = \sum_{i=1}^{Q} h_i \cdot u_i(Z) \neq 0$, since at least one h_i must be non-zero and the polynomials $u_1, \dots, u_\omega$ are linearly independent.
  In this case, the claim of Theorem 8 holds.
- Otherwise, $h_0 \neq 0$. Without loss of generality, we can assume that $h_0 = 1$ and $\rho(\mathcal{F}(V), x_i) \neq 0$. We then have

$$y_0(Z) = \frac{\sigma(\mathcal{F}(V), x_0)}{\rho(\mathcal{F}(V), x_0)} = \sum_{i=1}^{Q} h_i \cdot u_i(Z) \in \mathsf{span}_{\overline{\mathbb{Z}_p}}(W_T).$$

In the second case, it remains to show that the scalars $h_1, \dots, h_\omega \in \overline{\mathbb{Z}_p}$ can be chosen such that they lie in $\mathbb{Z}_p$. In fact, the following lemma finishes the proof of Theorem 8:

Lemma 6. *Let $u_0, \dots, u_\omega$ be vectors in $\mathbb{Z}_p^n$. If there are scalars $h_1, \dots, h_\omega \in \overline{\mathbb{Z}_p}$ s.t.*

$$u_0 = \sum_{i=1}^{\omega} h_i \cdot u_i,$$

then there are $b_1, \dots, b_\omega \in \mathbb{Z}_p$ s.t.

$$u_0 = \sum_{i=1}^{\omega} b_i \cdot u_i.$$

Proof. Choose an embedding

$$\iota : \mathbb{Z}_p \longrightarrow \overline{\mathbb{Z}_p}$$

of fields. Since ι is a $\mathbb{Z}_p$-linear map and $\overline{\mathbb{Z}_p}$ is a $\mathbb{Z}_p$-vector space, there is a $\mathbb{Z}_p$-linear map $r : \overline{\mathbb{Z}_p} \to \mathbb{Z}_p$ such that

$$r \circ \iota = \mathrm{id}_{\mathbb{Z}_p}.$$

Now, we set $b_i := r(h_i)$ for $i \in [\omega]$ and have

$$u_0 = \sum_{i=1}^{\omega} h_i \cdot u_i = r\left(\sum_{i=1}^{\omega} h_i \cdot u_i\right) = \sum_{i=1}^{\omega} r\left(h_i\right) \cdot u_i = \sum_{i=1}^{\omega} b_i \cdot u_i$$

by applying r component-wise.

Remark 14. In this section, we only presented and proved results for reductions that transform adversaries against the weak selective unpredictability of a VUF to solvers for *computational* Uber-assumptions. However, our results and techniques can – by simple methods – be extended to *decisional* Uber-assumptions. We detail this in Appendix C.

References

1. Abdalla, M., Catalano, D. & Fiore, D. *Verifiable Random Functions from Identity-Based Key Encapsulation* in *EUROCRYPT 2009* (2009).
2. Au, M. H., Susilo, W. & Mu, Y. *Practical Compact E-Cash* in *ACISP 07* (2007).
3. Bauer, B., Fuchsbauer, G. & Loss, J. *A Classification of Computational Assumptions in the Algebraic Group Model* in *CRYPTO 2020, Part II* (2020).
4. Belenkiy, M., Chase, M., Kohlweiss, M. & Lysyanskaya, A. *Compact E-Cash and Simulatable VRFs Revisited* in *PAIRING 2009* (2009).
5. Berlekamp, E. R. Factoring Polynomials Over Large Finite Fields. *Mathematics of Computation* (1970).
6. Bitansky, N. *Verifiable Random Functions from Non-interactive Witness-Indistinguishable Proofs* in *TCC 2017, Part II* (2017).
7. Blum, M. & Micali, S. *How to Generate Cryptographically Strong Sequences of Pseudo Random Bits* in *23rd FOCS* (1982).
8. Boneh, D., Boyen, X. & Goh, E.-J. *Hierarchical Identity Based Encryption with Constant Size Ciphertext* in *EUROCRYPT 2005* (2005).
9. Boneh, D., Montgomery, H. W. & Raghunathan, A. *Algebraic pseudorandom functions with improved efficiency from the augmented cascade* in *ACM CCS 2010* (2010).
10. Boneh, D. & Venkatesan, R. *Breaking RSA May Not Be Equivalent to Factoring* in *EUROCRYPT'98* (1998).
11. Boyen, X. *The Uber-Assumption Family (Invited Talk)* in *PAIRING 2008* (2008).
12. Brakerski, Z., Goldwasser, S., Rothblum, G. N. & Vaikuntanathan, V. *Weak Verifiable Random Functions* in *TCC 2009* (2009).
13. Coron, J.-S. *On the Exact Security of Full Domain Hash* in *CRYPTO 2000* (2000).
14. Coron, J.-S. *Optimal Security Proofs for PSS and Other Signature Schemes* in *EUROCRYPT 2002* (2002).
15. Cox, D., Little, J. & O'Shea, D. Ideals, Varieties, and Algorithms. An Introduction to Computational Algebraic Geometry and Commutative Algebra (2007).
16. Dodis, Y. *Efficient Construction of (Distributed) Verifiable Random Functions* in *PKC 2003* (2003).
17. Dodis, Y. & Yampolskiy, A. *A Verifiable Random Function with Short Proofs and Keys* in *PKC 2005* (2005).
18. Dubé, T. W. The Structure of Polynomial Ideals and Gröbner Bases. *SIAM Journal on Computing* (1990).
19. Fiore, D. & Schröder, D. *Uniqueness Is a Different Story: Impossibility of Verifiable Random Functions from Trapdoor Permutations* in *TCC 2012* (2012).
20. Fleischhacker, N., Jager, T. & Schröder, D. On Tight Security Proofs for Schnorr Signatures. *Journal of Cryptology* (2019).
21. Fuchsbauer, G., Kiltz, E. & Loss, J. *The Algebraic Group Model and its Applications* in *CRYPTO 2018, Part II* (2018).
22. Goldreich, O., Goldwasser, S. & Micali, S. *How to Construct Random Functions (Extended Abstract)* in *25th FOCS* (1984).
23. Goldreich, O. & Levin, L. A. *A Hard-Core Predicate for all One-Way Functions* in *21st ACM STOC* (1989).
24. Goldwasser, S. & Ostrovsky, R. *Invariant Signatures and Non-Interactive Zero-Knowledge Proofs are Equivalent (Extended Abstract)* in *CRYPTO'92* (1993).
25. Goyal, R., Hohenberger, S., Koppula, V. & Waters, B. *A Generic Approach to Constructing and Proving Verifiable Random Functions* in *TCC 2017, Part II* (2017).

26. Hofheinz, D. & Jager, T. *Verifiable Random Functions from Standard Assumptions* in *TCC 2016-A, Part I* (2016).
27. Hohenberger, S. & Waters, B. *Constructing Verifiable Random Functions with Large Input Spaces* in *EUROCRYPT 2010* (2010).
28. Jager, T. *Verifiable Random Functions from Weaker Assumptions* in *TCC 2015, Part II* (2015).
29. Jarecki, S. & Shmatikov, V. *Handcuffing Big Brother: an Abuse-Resilient Transaction Escrow Scheme* in *EUROCRYPT 2004* (2004).
30. Katsumata, S. *On the Untapped Potential of Encoding Predicates by Arithmetic Circuits and Their Applications* in *ASIACRYPT 2017, Part III* (2017).
31. Katz, J., Zhang, C. & Zhou, H.-S. *An Analysis of the Algebraic Group Model* Cryptology ePrint Archive, Report 2022/210. 2022.
32. Kohl, L. *Hunting and Gathering - Verifiable Random Functions from Standard Assumptions with Short Proofs* in *PKC 2019, Part II* (2019).
33. Kurosawa, K., Nojima, R. & Phong, L. T. *Relation between Verifiable Random Functions and Convertible Undeniable Signatures, and New Constructions* in *ACISP 12* (2012).
34. Liang, B., Li, H. & Chang, J. *Verifiable Random Functions from (Leveled) Multilinear Maps* in *CANS 15* (2015).
35. Liskov, M. *Updatable Zero-Knowledge Databases* in *ASIACRYPT 2005* (2005).
36. Lysyanskaya, A. *Unique Signatures and Verifiable Random Functions from the DH-DDH Separation* in *CRYPTO 2002* (2002).
37. Maurer, U. M. *Abstract Models of Computation in Cryptography (Invited Paper)* in *10th IMA International Conference on Cryptography and Coding* (2005).
38. Micali, S., Rabin, M. O. & Vadhan, S. P. *Verifiable Random Functions* in *40th FOCS* (1999).
39. Micali, S. & Reyzin, L. *Soundness in the Public-Key Model* in *CRYPTO 2001* (2001).
40. Micali, S. & Rivest, R. L. *Micropayments Revisited* in *CT-RSA 2002* (2002).
41. Naor, M. & Reingold, O. *Number-theoretic Constructions of Efficient Pseudo-random Functions* in *38th FOCS* (1997).
42. Naor, M. & Reingold, O. *From Unpredictability to Indistinguishability: A Simple Construction of Pseudo-Random Functions from MACs (Extended Abstract)* in *CRYPTO'98* (1998).
43. Nechaev, V. I. Complexity of a Determinate Algorithm for the Discrete Logarithm. *Mathematical Notes* (1994).
44. Niehues, D. *Verifiable Random Functions with Optimal Tightness* in *PKC 2021, Part II* (2021).
45. Paillier, P. & Vergnaud, D. *Discrete-Log-Based Signatures May Not Be Equivalent to Discrete Log* in *ASIACRYPT 2005* (2005).
46. Rosie, R. *Adaptive-Secure VRFs with Shorter Keys from Static Assumptions* in *CANS 18* (2018).
47. Schwartz, J. T. Fast Probabilistic Algorithms for Verification of Polynomial Identities. *J. ACM* (1980).
48. Shoup, V. *Lower Bounds for Discrete Logarithms and Related Problems* in *EUROCRYPT'97* (1997).
49. von zur Gathen, J. & Panario, D. Factoring Polynomials Over Finite Fields: A Survey. *Journal of Symbolic Computation* (2001).
50. Yamada, S. *Asymptotically Compact Adaptively Secure Lattice IBEs and Verifiable Random Functions via Generalized Partitioning Techniques* in *CRYPTO 2017, Part III* (2017).
51. Yao, A. C.-C. *Theory and Applications of Trapdoor Functions (Extended Abstract)* in *23rd FOCS* (1982).
52. Zhandry, M. *To Label, or Not To Label (in Generic Groups)* Cryptology ePrint Archive, Report 2022/226. 2022.

Supplementary Material

A Linear Attacks on Constant-Degree UFs

An unpredictable function (UF) is a cryptographic primitive that – given a secret key sk and a preimage x – outputs an image $\mathbf{y}$ that looks somewhat random to any party that does not know the secret key sk.

In this section, we will discuss two attacks on unpredictable functions whose evaluation function can be represented as a polynomial of constant degree on either its secret key or the user input.

Let us first give a formal definition of UFs.

Definition 25 (Unpredictable Functions, UFs [42]). *Let* $\mathsf{uf} = (\mathsf{Gen}_{\mathsf{uf}}, \mathsf{Eval}_{\mathsf{uf}})$ *be a pair of algorithms of the following form:*

- $\mathsf{Gen}_{\mathsf{uf}}(1^\lambda)$ *outputs a secret key* sk.
- $\mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x)$ *is a* deterministic *algorithm that – on input a secret key* sk *and a preimage* $x \in \mathcal{X} = (\mathcal{X}_\lambda)_\lambda$ *– outputs an image* $\mathbf{y} \in \mathcal{Y} = (\mathcal{Y}_\lambda)_\lambda$.

Definition 26 (Weak Selective Unpredictability). *Let* $Q \in \mathsf{poly}(\lambda)$. *We call a UF* $\mathsf{uf} = (\mathsf{Gen}_{\mathsf{uf}}, \mathsf{Eval}_{\mathsf{uf}})$ ***weakly** Q**-selectively unpredictable** if for each PPT adversary* $\mathcal{A}$ *there is a negligible function* $\epsilon(\lambda)$ *s.t.:*

$$\left|\Pr\left[\mathcal{A}(1^\lambda, \overrightarrow{x}, \overrightarrow{\mathbf{y}}) = \mathbf{y}_0 \;\middle|\; \begin{array}{l} \overrightarrow{x} = (x_0, \dots, x_Q) \xleftarrow{\$} \mathcal{X}_\lambda^{Q+1} \\ \mathsf{sk} \xleftarrow{\$} \mathsf{Gen}_{\mathsf{uf}}(1^\lambda) \\ \mathbf{y}_i \leftarrow \mathsf{Eval}_{\mathsf{vrf}}(\mathsf{sk}, x_i) \\ \overrightarrow{\mathbf{y}} = (y_1, \dots, y_Q) \end{array}\right] - \frac{1}{|\mathcal{Y}_\lambda|}\right| \leq \epsilon(\lambda)$$

Now, let $\mathsf{uf} = (\mathsf{Gen}_{\mathsf{uf}}, \mathsf{Eval}_{\mathsf{uf}})$ be a candidate unpredictable function. We denote the space of possible outputs of $\mathsf{Gen}_{\mathsf{uf}}$ by $\mathcal{K}_{\mathsf{uf}} = (\mathcal{K}_{\mathsf{uf},\lambda})_\lambda$. Assume further that $\mathcal{X} = \{0,1\}^n$ for $n = n(\lambda) \in \mathsf{poly}(\lambda)$ and that $\mathsf{Eval}_{\mathsf{uf}}$ is of the following type for an arbitrary pairing group $(\mathbb{G}, \mathbb{G}_T, e)$:

$$\mathsf{Eval}_{\mathsf{uf}} \colon \mathcal{K}_{\mathsf{uf}} \times \{0,1\}^n \longrightarrow \mathbb{G}_T$$

for a poly-bit prime $p = p(\lambda) = |\mathbb{G}_T|$.

Furthermore, let $d = d(\lambda) \in \mathbb{N}$ be arbitrary and assume that for each $\mathsf{sk} \in \mathcal{K}_{\mathsf{uf}}$ there is an element $\mathbf{h} \in \mathbb{G}_T$ and a polynomial $f_{\mathsf{sk}} \in \mathbb{Z}_p[X_1, \dots, X_n]$ of total degree $\deg f_{\mathsf{sk}} \leq d(\lambda)$ s.t.

$$\mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_1, \dots, x_n) = \mathbf{h}^{f_{\mathsf{sk}}(x_1, \dots, x_n)}.$$

Theorem 9. *Let* uf *be as described above and let* $n \in \Omega(\lambda), d \in O(1)$. *Then,* uf *is **not** weakly Q-selectively unpredictable for* $Q \geq n^d + 1$.

Proof. For $n \geq d \geq 1$, set

$$N_d = \binom{n}{d} + \binom{n}{d-1} + \ldots + \binom{n}{1} + \binom{n}{0}.$$

We claim

$$N_d \leq n^d + 1.$$

In fact, we have by induction on d

$$N_d = \binom{n}{d} + N_{d-1} \leq \binom{n}{d} + n^{d-1} + 1 \leq \frac{n^d}{d!} + n^{d-1} + 1.$$

For $n \geq d \geq 2$, it follows

$$\frac{n^d}{d!} + n^{d-1} \leq \frac{n^d}{2} + n^{d-1} = n^{d-1}(\frac{n}{2} + 1) \leq n^d.$$

Let $M = \{X^\alpha = X_1^{\alpha_1} \cdots X_n^{\alpha_n} \mid \forall i = 1, \ldots, n : \alpha_i \in \{0,1\}, \sum_{i=1}^n \alpha_i \leq d\}$ be the set of all monomials of degree at most d that contain each variable at most once. Then $|M| = N_d$.

Let $m_1, \ldots, m_{N_d}$ be any ordering of the elements of M and consider the following mapping:

$$\begin{aligned} v_d : \mathbb{Z}_p^n &\longrightarrow \mathbb{Z}_p^{N_d} \\ x &\longmapsto (m_1(x), \ldots, m_{N_d}(x)). \end{aligned}$$

While $\mathsf{img}(v_d)$ is not a vector space, we claim that its elements span the whole space $\mathbb{Z}_p^{N_d}$. Set

$$S_d \coloneqq \{x \in \{0,1\}^n \mid ||x||_1 \leq d\}.$$

That is, S_d is the set of bitstrings of length n where at most d entries are non-zero. Then, $|S_d| = N_d$. We claim that $v_d(S_d)$ is a basis of $\mathbb{Z}_p^{N_d}$. It suffices to show that the elements of $v_d(S_d)$ are linearly independent as we already know that $|S|_d = N_d$. We will show this statement by induction on d:

Induction Start: Let $d = 1$. Then, $N_d = n + 1$ and v_1 is given by

$$\begin{aligned} v_1 : \mathbb{Z}_p^n &\longrightarrow \mathbb{Z}_p^{N_d} \\ (x_1, \ldots, x_n) &\longmapsto (1, x_1, \ldots, x_n) \end{aligned}$$

(depending on the ordering of the monomials $m_1(x), \ldots, m_{N_d}$). S_1 is the set of all binary vectors that contain at most one non-zero entry and $v_1(S_1)$ is given by

$$v_1(S_1) = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \ldots, \begin{pmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} \right\}.$$

Therefore, it is easy to see that $v_1(S_1)$ does not contain any linear dependencies.

Induction Step: Let $d > 2$ and let us assume that the statement holds for $d-1$. I.e., we know that the vectors in $v_{d-1}(S_{d-1})$ are linearly independent. Assume – for the sake of contradiction – there exists a non-zero $\overrightarrow{\rho} \in \mathbb{Z}_p^{N_d}$ s.t. we have

$$\sum_{i=1}^{N_d} \rho_i v_d(s_i) = 0$$

where we denote $S_d = \{s_1, \ldots, s_{N_d}\}$.
Since $v_{d-1}(S_{d-1})$ is linearly independent, there must be at least one $\iota \in [N_d]$ s.t. $\rho_\iota \neq 0$ and $||s_\iota||_1 = d$. That is, s_ι is 1 exactly on d distinct indices $j_1, \ldots, j_d \in \{1, \ldots, n\}$. Choose $\alpha \in [N_d]$ s.t.

$$m_\alpha(X) = X_{j_1} \cdots X_{j_d} \in M.$$

m_α is one at s_ι and zero at each other point of S_d. We now have

$$0 = \sum_{i=1}^{N_d} \rho_i \cdot (v_d(s_i))_\alpha = \sum_{i=1}^{N_d} \rho_i m_\alpha(s_i) = \rho_\iota m_\alpha(s_\iota) = \rho_\iota.$$

This contradicts our previous assumption that $\rho_\iota \neq 0$.
Ergo, $v_d(S_d)$ is linearly independent and therefore a basis of $\mathbb{Z}_p^{N_d}$.

We now describe our adversary $\mathcal{A}$ against the unpredictability of the uf. $\mathcal{A}$ gets $1^\lambda, x_0, \ldots, x_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$ as input and has to output $\mathbf{y}_0 = \mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_0)$.

Since $Q \geq N_d$, the vectors $v_d(x_0), \ldots, v_d(x_Q)$ must be linearly dependent. If $v_d(x_0) \notin \mathrm{span}_{\mathbb{Z}_p}\{v_d(x_1), \ldots, v_d(x_Q)\}$, then $\mathcal{A}$ returns $\perp$. Since the preimages $x_0, \ldots, x_Q$ have been drawn uniformly and independently at random, this will happen with probability at most $\frac{Q}{Q+1}$.

If $\mathcal{A}$ does not return $\perp$, then $v_d(x_0) \in \mathrm{span}_{\mathbb{Z}_p}\{v_d(x_1), \ldots, v_d(x_Q)\}$ and there exists a vector $\rho \in \mathbb{Z}_p^Q$ s.t.

$$v_d(x_0) = \sum_{i=1}^{Q} \rho_i \cdot v_d(x_i).$$

This allows the adversary to compute the value $\mathbf{y}_0$ such that $\mathbf{y}_0 = \mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x^*)$ by

$$\mathbf{y}_0 := \mathbf{y}_1^{\rho_1} \cdots \mathbf{y}_Q^{\rho_Q}.$$

The adversary then returns $\mathbf{y}_0$ as its candidate solution to the game.

We claim that $\mathcal{A}$ can always correctly compute $\mathbf{y}_0 \leftarrow \mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_0)$. Fix sk and let $\mathbf{h} \in \mathbb{G}_T$ and $f_{\mathsf{sk}} \in \mathbb{Z}_p[X_1, \ldots, X_n]$ be a polynomial of total degree d s.t.

$$\mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, X_1, \ldots, X_n) = \mathbf{h}^{f_{\mathsf{sk}}(X_1, \ldots, X_n)}.$$

Note, that there is a reduced polynomial

$$\overline{f_{\mathsf{sk}}} \in \mathbb{Z}_p[X_1, \ldots, X_n]/(X_1^2 - X_1, \ldots, X_n^2 - X_n)$$

s.t. f_{sk} and $\overline{f_{\mathsf{sk}}}$ coincide on $\{0,1\}^n$.

Since each monomial of $\overline{f_{\mathsf{sk}}}$ contains each variable at most once, $\overline{f_{\mathsf{sk}}} \in \operatorname{span}_{\mathbb{Z}_p}(M)$. In fact, choose $c_1, \ldots, c_{N_d} \in \mathbb{Z}_p$ s.t.

$$\overline{f_{\mathsf{sk}}}(X) = \sum_{i=1}^{N_d} c_i m_i(X).$$

Then, we have

$$\begin{aligned}\overline{f_{\mathsf{sk}}}(x_0) &= \sum_{i=1}^{N_d} c_i m_i(x_0) = \sum_{i=1}^{N_d} c_i \left(\sum_{j=1}^{Q} \rho_j m_i(x_j) \right) \\ &= \sum_{j=1}^{Q} \rho_j \sum_{i=1}^{N_d} c_i m_i(x_j) = \sum_{j=1}^{Q} \rho_j \overline{f_{\mathsf{sk}}}(x_j)\end{aligned}$$

Thus, $\mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_0) = \mathbf{y}_1^{\rho_1} \cdots \mathbf{y}_Q^{\rho_Q}$ as claimed. Therefore, $\mathcal{A}$ can compute $\mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_0)$ on its own with probability at least $\frac{1}{N_d+1}$.

While $Q \geq N_d$, note that $\mathcal{A}$ only needs N_d preimages and values of $\mathsf{Eval}_{\mathsf{uf}}$, store $N_d(N_d+1)$ values of $\mathbb{Z}_p$, solve an $N_d \times N_d$ linear equation system over $\mathbb{Z}_p$ and compute N_d exponentiations and $N_d - 1$ multiplications in the target group. Therefore, $\mathcal{A}$ has a time complexity of $O(N_d^3) = O(n^{3d})$ and a space complexity of $O(N_d^2) = O(n^{2d})$.

We will now consider a different kind of UF where the evaluation function is polynomial in the secret key.

Let $m = m(\lambda) \in \mathsf{poly}(\lambda)$. We consider the following evaluation function of uf that is given by an efficiently computable function

$$\mathsf{Eval}_{\mathsf{uf}} : \mathbb{Z}_p^m \times \mathcal{X} \longrightarrow \mathbb{G}_T.$$

I.e., the secret key lies now in $\mathbb{Z}_p^m$ while we do not impose any restrictions on the input space $\mathcal{X}$.

Let $d = d(\lambda)$ be arbitrary and assume, that there is a group element $\mathbf{h} \in \mathbb{G}_T$ s.t. for each $x \in \mathcal{X}$ there is a polynomial $f_x \in \mathbb{Z}_p[S_1, \ldots, S_m]$ of total degree $\leq d(\lambda)$ s.t.

$$\mathsf{Eval}_{\mathsf{uf}}((s_1, \ldots, s_m), x) = \mathbf{h}^{f_x(s_1, \ldots, s_m)}.$$

We assume that – by knowing x and a description of uf – one can efficiently compute f_x.

We then have the following attack on uf.

Theorem 10. *If we have $|\mathcal{X}| > \binom{m+d}{d}$, then there is an adversary with time complexity $O\left(\binom{m+d}{d}^3\right)$ who can break the unpredictability of* uf *by asking $O\left(\binom{m+d}{d}\right)$ values of* uf *at arbitrary inputs.*

If $d \in O(1), m \in$ $\mathsf{poly}(\lambda)$, then uf *is **not** weakly Q-selectively unpredictable for $Q \geq \binom{m+d}{d}$.*

Proof. The number of monomials in m variables of total degree at most d is $\binom{m+d}{d}$. Therefore, the vector space of all polynomials in $\mathbb{Z}_p[Y_1, \dots, Y_m]$ of total degree $\leq d$ is at most $\binom{m+d}{d}$. For $\binom{m+d}{d} + 1$ different inputs $x_0, \dots, x_{\binom{m+d}{d}} \in \mathcal{X}$, there must be a non-trivial linear dependency of the polynomials $f_{x_0}(S), \dots, f_{x_{\binom{m+d}{d}}}(S)$.

This leads to the following adversary $\mathcal{A}$: $\mathcal{A}$ receives $1^\lambda, x_0, \dots, x_Q, \mathbf{y}_1, \dots, \mathbf{y}_q$ and has to compute $\mathbf{y}_0 = \mathsf{Eval}_{\mathsf{uf}}(\mathsf{sk}, x_0) = f_{x_0}(\mathsf{sk})$.

Since $Q \geq \binom{m+d}{d}$, the polynomials $f_{x_0}(S), \dots, f_{x_Q}(S)$ must be linearly dependent. If $f_{x_0} \notin \operatorname{span}_{\mathbb{Z}_p}(f_{x_1}, \dots, f_{x_m})$, $\mathcal{A}$ outputs $\perp$.

Otherwise, $\mathcal{A}$ can compute scalars $\kappa_1, \dots, \kappa_Q$ s.t.

$$f_{x_0}(S) = \sum_{i=1}^{Q} \kappa_i \cdot f_{x_i}(S)$$

by using Gauss-elimination. This allows $\mathcal{A}$ to predict $\mathbf{y}_0$ by computing

$$\mathbf{y}_0 = \mathbf{h}^{f_{x_0}(s)} = \mathbf{h}^{\sum_{i=1}^{Q} \kappa_i \cdot f_{x_i}(s)} = \prod_{i=1}^{Q} \mathbf{h}^{\kappa_i \cdot f_{x_i}(s)} = \prod_{i=1}^{Q} \mathbf{y}_i^{\kappa_i}.$$

Therefore, $\mathcal{A}$ can break the weak Q-selective unpredictability of uf with probability at least $\frac{1}{\binom{m+d}{d}+1}$.

B On the Lemma of Verification Equations

We want to give here a proof of Lemma 3.

For this end, let $(\mathbb{G}^Z, \mathbb{G}_T^Z, e^Z)$ be a generic group with a pairing that encodes elements of $\mathbb{Z}_p[Z_1, \dots, Z_t]$. I.e., the exponents of elements of $\mathbb{G}^Z$ and $\mathbb{G}_T^Z$ are formal polynomials.

Let NICA be an Uber-assumption as defined in Definition 16. I.e., there is a distribution procedure that has the form

```
$\mathcal{D}(1^\lambda) :=$ {
        draw $\mathbf{g}$ by any distribution s.t. $\mathbf{g}$ is a generator of $\mathbb{G}$;
        draw $(z_1, \dots, z_t) \xleftarrow{\$} \mathbb{Z}_p^t$ uniformly and independently at random;
        set $a_1 := f_{A_1}(z_1, \dots, z_t), \dots, a_{q_1} := f_{A_{q_1}}(z_1, \dots, z_t)$;
        set $b_1 := f_{B_1}(z_1, \dots, z_t), \dots, b_{q_2} := f_{B_{q_2}}(z_1, \dots, z_t)$;
        return $(\mathbf{g}, \mathbf{g}^{a_1}, \dots, \mathbf{g}^{a_{q_1}}, e(\mathbf{g}, \mathbf{g})^{b_1}, \dots, e(\mathbf{g}, \mathbf{g})^{b_{q_2}})$;
    }.
```

and samples challenges $(\mathbf{g}, \mathbf{g}^{\vec{a}}, \mathbf{g}^{\vec{b}})$ of NICA.

For a distribution $\mathcal{D}$ as above, we set $\mathcal{D}_Z$ to be the distribution over $(\mathbb{G}^Z)^{q_1+1} \times (\mathbb{G}_T^Z)^{q_2}$ given by

$$
\begin{aligned}
&\mathcal{D}_Z(1^\lambda) := \\
&\qquad \{ \\
&\qquad\quad \text{draw } \mathbf{g} \text{ by any distribution s.t. } \mathbf{g} \text{ is a generator of } \mathbb{G}^Z; \\
&\qquad\quad \text{set } a_1(Z) := f_{A_1}(Z_1, \ldots, Z_t), \ldots, a_{q_1}(Z) := f_{A_{q_1}}(Z_1, \ldots, Z_t) \in \mathbb{Z}_p[Z]; \\
&\qquad\quad \text{set } b_1(Z) := f_{B_1}(Z_1, \ldots, Z_t), \ldots, b_{q_2}(Z) := f_{B_{q_2}}(Z_1, \ldots, Z_t) \in \mathbb{Z}_p[Z]; \\
&\qquad\quad \text{return } (\mathbf{g}, \mathbf{g}^{a_1(Z)}, \ldots, \mathbf{g}^{a_{q_1}(Z)}, e(\mathbf{g}, \mathbf{g})^{b_1(Z)}, \ldots, e(\mathbf{g}, \mathbf{g})^{b_{q_2}(Z)}); \\
&\qquad \}.
\end{aligned}
$$

I.e., $\mathcal{D}_Z$ acts like $\mathcal{D}$ but does not replace the place-holder variables of $f_{A_1}, \ldots, f_{A_{q_1}}$, $f_{B_1}, \ldots, f_{B_{q_2}}$ with real values.

We claim that a generic distinguisher has a negligible advantage in distinguishing $\mathcal{D}$ from $\mathcal{D}_Z$:

Lemma 7. *Let d_{NICA} be the degree of the Uber-assumption NICA. The advantage of an adversary $\mathcal{A}$ running in time at most $t_{\mathcal{A}}$ at distinguishing between the following two games is at most*

$$\left| \Pr\left[\mathbf{G_0}^{\mathcal{A}} = 1\right] - \Pr\left[\mathbf{G_1}^{\mathcal{A}} = 1\right] \right| \le \frac{t_{\mathcal{A}}^2 \cdot 2d_{\mathsf{NICA}}}{p}$$

Game $\mathbf{G_0}$*:*

Setup: *Sample $z_1, \ldots, z_t \xleftarrow{\$} \mathbb{Z}_p$ uniformly at random. Evaluate $a_i = f_{A_1}(\overrightarrow{z}), \ldots, a_{q_1} = f_{A_{q_1}}(\overrightarrow{z})$, $b_1 = f_{B_1}(\overrightarrow{z}), \ldots, b_{q_2} = f_{B_{q_2}}(\overrightarrow{z})$. Add $a_1, \ldots, a_{q_1}, b_1, \ldots b_{q_2}$ along with group element handles to internal lists of elements of $\mathbb{G}$ and $\mathbb{G}_T$. Output the handles to $\mathcal{A}$.*

Online Phase: *Whenever the adversary makes a request to the group oracle with two handles, the game looks up the internal representation of the two group elements. If they exist, it adds the two group elements and looks up if there is already a handle. Otherwise it samples a new handle and adds it to the list. Same for pairing operations. It outputs the resulting handle to the adversary.*

Output Determination: *Once the adversary outputs a bit $\mathfrak{b}'$, the game $\mathbf{G_0}$ outputs 1 if $\mathfrak{b}' = 0$ and 0 otherwise.*

Game $\mathbf{G_1}$*:*

Setup: *The game internally generates a list of handles corresponding to polynomials over the variables $Z_1, \ldots Z_t$. It puts $f_{A_1}(Z), \ldots f_{A_{q_1}}(Z), f_{B_1}(Z), \ldots, f_{B_{q_2}}(Z)$ in the internal list. It then outputs the handles of $f_{A_1}, \ldots f_{A_{q_1}}$, $f_{B_1}, \ldots, f_{B_{q_2}}$ to the adversary.*

Online Phase: *Whenever the adversary makes a group oracle query or a pairing query, the game looks up the handles (if they are not in the list it returns $\perp$) and then either adds (in case of group operation) or multiplies (in case of pairing operation and* *both elements being from $\mathbb{G}_Z$) the polynomials. It then checks if the resulting polynomial already exists in the list and if yes returns the handle, and if not it samples a new handle, adds the polynomial and its handle to the list and returns the handle.*

Output Determination: *When the adversary outputs its final output bit $\mathfrak{b}'$, the game returns 1 if $\mathfrak{b}' = 1$ and 0 otherwise.*

Proof. We consider the following case. Take $\mathbf{G_1}$ as above. Let the adversary play $\mathbf{G_1}$. Once the game is finished, sample a vector $\overrightarrow{z} \xleftarrow{\$} \mathbb{Z}_p^t$. Then evaluate all polynomials in the internal representation lists at $\overrightarrow{z}$. We say a *collision* occurs if during the run of $\mathbf{G_1}$, two different group handles were given out for two group elements whose internal polynomials evaluated to the same value at $\overrightarrow{z}$. It is easy to see, that the games $\mathbf{G_0}$ and $\mathbf{G_1}$ are perfectly indistinguishable if no collisions occur. We therefore consider the probability that such a collision occurs. As the adversary runs in time $t_{\mathcal{A}}$, it can make at most $t_{\mathcal{A}}$ queries to the group oracles. Each pair of such queries yields a potential collision. Thus, there are $t_{\mathcal{A}}^2$ possible collision points. The game $\mathbf{G_1}$ only outputs different handles if the internal representations differ *as polynomials*, i.e., the difference of two colliding polynomials is not the zero polynomial. As the polynomials $f_{A_1}, \dots, f_{B_1}, \dots$ in NIDA have degree up to d_{NICA}, the resulting polynomials in the group representations from pairing operations can have degree up to $2d_{\mathsf{NICA}}$. Thus, the difference between two internal representations can also have degree up to $2d_{\mathsf{NICA}}$. Lemma 1 tells us that the probability for each single potential collision is therefore at most $\frac{2d_{\mathsf{NICA}}}{p}$. Thus, the overall difference that at least one collision occurs is $\frac{t_{\mathcal{A}}^2 \cdot 2d_{\mathsf{NICA}}}{p}$. This yields the statement.

Lemma 8. *Let NICA be an Uber-assumption of degree d_{NICA}. Let $d_h = d_h(\lambda)$ be any non-negative number.*

Let $\mathcal{A}$ be a generic algorithm of time complexity $t_{\mathcal{A}}$ that on input

$$(\mathbf{g}, \mathbf{g}^{f_{A_1}(\overrightarrow{z})}, \dots, g^{f_{A_{q_1}}(\overrightarrow{z})}, e(\mathbf{g}, \mathbf{g})^{f_{B_1}(\overrightarrow{z})}, \dots, e(\mathbf{g}, \mathbf{g})^{f_{B_{q_2}}(\overrightarrow{z})}) \xleftarrow{\$} \mathcal{D}(1^\lambda)$$

outputs a list of sparse polynomials $h_1, \dots, h_l \in \mathbb{Z}_p[Z_1, \dots, Z_t]$ of degree $\leq d_h$ s.t. we have for each $i \in [l]$

$$h_i(z_1, \dots, z_t) = 0.$$

Then, we have

$$\begin{aligned} &\Pr\left[\exists i \in [l] : \; h_i(Z) \neq 0 \;\middle|\; \begin{array}{l} (\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \xleftarrow{\$} \mathcal{D} \\ (h_1(Z), \dots, h_l(Z)) \xleftarrow{\$} \mathcal{A}(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \end{array} \right] \\ &\quad \leq l \cdot \frac{d_h}{p} + t_{\mathcal{A}}^2 \cdot \frac{2d_{\mathsf{NICA}}}{p}. \end{aligned}$$

Proof. The lemma basically states that the sample from $\mathcal{D}$ only gives a negligible advantage in "solving the Schwartz-Zippel-problem".

By Lemma 7, we know that altering $\mathcal{A}$'s source of inputs from $\mathcal{D}$ to $\mathcal{D}_Z$ affects $\mathcal{A}$'s outputs only with probability $\leq \frac{t_{\mathcal{A}}^2 \cdot 2d_{\mathsf{NICA}}}{p}$.

However, the outputs of $\mathcal{D}_Z$ are obviously devoid of any information about the values $z_1, \dots, z_t \in \mathbb{Z}_p$. Therefore, we can use Schwartz-Zippel again and gain for each polynomial h_i outputted by $\mathcal{A}(\mathcal{D}_Z)$

$$\Pr[h_i(z) = 0 \wedge h_i(Z) \neq 0] \leq \frac{d_h}{q}.$$

In total, the claim of the lemma follows by a union bound over all $i \in [l]$.

Equipped with Lemma 8, we can prove Lemma 3.

For this end, let $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$ be a verifiable unpredictable function that maps inputs $x \in \mathcal{X} = (\mathcal{X}_\lambda)_\lambda$ to target group elements $\mathbf{y} \in \mathbb{G}_T$. In Section 6, we made the following conventions about vuf:

1. We assumed that a verification key vk of vuf always has (besides other information) n_1 source group and n_2 target group elements. We denoted the source group elements of vk by $\mathsf{vk}_{S,1}, \dots, \mathsf{vk}_{S,n_1}$ and all target group elements of vk by $\mathsf{vk}_{T,1}, \dots, \mathsf{vk}_{T,n_2}$.
   Further, we assumed that a verification key vk of vrf contains – besides other information – always a declared generator of the group $\mathbb{G}$.
2. We assumed that a proof outputted by $\mathsf{Eval}_{\mathsf{vrf}}$ will always have $u_1 = u_1(\lambda)$ source and $u_2 = u_2(\lambda)$ target group elements (besides other information).
   For $(\pi, \mathbf{y}) \leftarrow \mathsf{Eval}_{\mathsf{vrf}}(\mathsf{sk}, x)$, we denote the source group elements of π by $\pi_{S,1}, \dots,$ π_{S,u_1} and the target group elements by $\pi_{T,1}, \dots, \pi_{T,u_2}$.
3. The verification algorithm always checks that the declared generator in vk is indeed a generator of $\mathbb{G}$.

Now, let $(\phi_x)_{x \in \mathcal{X}}$ be a family of verification equations for vuf of degree d_ϕ. I.e., each ϕ_x is a polynomial in $\mathbb{Z}_p[V_{S,1}, \dots, V_{S,n_1}, V_{T,1}, \dots, V_{T,n_2}, P_{S,1}, \dots, P_{S,u_1}, P_{T,1},$ $\dots, P_{T,u_2}, Y]$ such that we have for each tuple $(\mathsf{vk}, x, \mathbf{y}, \pi)$ accepted by $\mathsf{Verify}_{\mathsf{vuf}}$

$$\phi_x(\overrightarrow{v_S}, \overrightarrow{v_T}, \overrightarrow{p_S}, \overrightarrow{p_T}, y) = 0$$

where $\overrightarrow{v_S}, \overrightarrow{p_S}$ denote the exponents of the group elements $\mathsf{vk}_{S,1}, \dots, \mathsf{vk}_{S,n_1}, \pi_{S,1}, \dots,$ π_{S,u_1} relative to *any* generator $\mathbf{h} \in \mathbb{G}$ and $\overrightarrow{v_T}, \overrightarrow{p_T}, y$ denote the exponents of the group elements $\mathsf{vk}_{T,1}, \dots, \mathsf{vk}_{T,n_2}, \pi_{T,1}, \dots, \pi_{T,u_2}, \mathbf{y}$ relative to the corresponding generator $e(\mathbf{h}, \mathbf{h}) \in \mathbb{G}_T$.

Now, consider an interaction between a generic PPT reduction $\mathcal{R}$ and an adversary $\mathcal{A}$ on the weak selective unpredictability of vuf. When querying $\mathcal{A}$, $\mathcal{R}$ will compute some inputs $x_0, \dots, x_Q \in \mathcal{X}$,a verification key vk, proofs $\pi_1, \dots, \pi_Q$, image values $\mathbf{y}_1, \dots, \mathbf{y}_Q$ and send them to $\mathcal{A}$. If this happens, we will assume that $\mathcal{A}$ will return $\perp$ if there is one $i \in [Q]$ s.t. $\mathsf{Verify}_{\mathsf{vuf}}$ rejects $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$. If all tuples $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ are accepted by $\mathsf{Verify}_{\mathsf{vuf}}$, $\mathcal{A}$ will return $\perp$ or a target group element $\mathbf{y}_0$ to $\mathcal{R}$.

In the above interaction, the algebraic representations of the exponents of the group elements of $\mathsf{vk}, \pi_1, \dots, \pi_Q, \mathbf{y}_1, \dots, \mathbf{y}_Q$ induces a morphism of rings

$$\mathcal{F} : Z_p[V, P, Y] \to \mathbb{Z}_p[Z]$$

as explained in Section 6 where

$$\mathbb{Z}_p[V,P,Y] := \\ \mathbb{Z}_p[(V_{S,j})_{j\in[n_1]},(V_{T,j})_{j\in[n_2]},(P_{S,i,j})_{i\in[Q],j\in[u_1]},(P_{T,i,j})_{i\in[Q],j\in[u_2]},(Y_i)_{i\in[Q]}].$$

$\mathcal{F}$ maps each single variable to a polynomial in $\mathbb{Z}_p[Z]$ of degree $\leq 2d_{\mathsf{NICA}}$.

If $\mathcal{A}$ does not return $\perp$, $\mathsf{Verify}_{\mathsf{vuf}}$ must accept each tuple $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ in the above interaction. Therefore, each $\phi_{x_i} \in \mathbb{Z}_p[(V_{S,j})_{j\in[n_1]},(V_{T,j})_{j\in[n_2]},(P_{S,i,j})_{j\in[u_1]}, (P_{T,i,j})_{j\in[u_2]}, Y_i]$ must vanish on the exponents of $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ relative to the basis $\mathbf{g}$. However, if $\overrightarrow{z} \in \mathbb{Z}_p^t$ is the value drawn by $\mathcal{D}$, then the exponents of $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ are exactly the evaluations

$$(\mathcal{F}(V_{S,j})(\overrightarrow{z}))_{j\in[n_1]}, (\mathcal{F}(V_{T,j})(\overrightarrow{z}))_{j\in[n_2]}, \\ (\mathcal{F}(P_{S,i,j})(\overrightarrow{z}))_{j\in[u_1]}, (\mathcal{F}(P_{T,i,j})(\overrightarrow{z}))_{j\in[u_2]} \text{ and } \mathcal{F}(Y_i)(\overrightarrow{z}).$$

In particular, we have for each $i \in [Q]$

$$\mathcal{F}(\phi_{x_i})(\overrightarrow{z}) = 0.$$

Now, Lemma 3 states that with overwhelming probability we additionally have on the level of polynomials of $\mathbb{Z}_p[Z]$

$$\mathcal{F}(\phi_{x_i})(Z) = 0.$$

We will prove this now formally:

Proof (Lemma 3). To prove the claim we construct the following meta-algorithm $\mathcal{M}$:

1. On input $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g},\mathbf{g})^{\overrightarrow{b}})$, $\mathcal{M}$ starts running $\mathcal{R}(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g},\mathbf{g})^{\overrightarrow{b}})$.
2. Whenever $\mathcal{R}$ tries to access $\mathcal{A}$, $\mathcal{M}$ pauses $\mathcal{R}$ and reads the input $\mathsf{vk}, x_0, \ldots, x_Q, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q$ that $\mathcal{R}$ tries to send to $\mathcal{A}$.
3. If there is a $i \in [Q]$ s.t. $\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i) = 0$, $\mathcal{M}$ will answer $\mathcal{R}$'s query with $\perp$ and continue running $\mathcal{R}$.
4. Otherwise, $\mathcal{M}$ stops $\mathcal{R}$ and extracts polynomial representations of the exponents of the group elements in $(\mathsf{vk}, \pi_1, \ldots, \pi_Q, \mathbf{y}_1, \ldots, \mathbf{y}_Q))$ from $\mathcal{R}$.
5. $\mathcal{M}$ now outputs the sparse polynomials $\mathcal{F}(\phi_{x_1}), \ldots, \mathcal{F}(\phi_{x_Q})$ and stops.
6. If $\mathcal{R}$ stops before making a successful interaction with $\mathcal{A}$, then $\mathcal{M}$ outputs an empty list of polynomials and stops.

The time complexity of $\mathcal{M}$ is bounded by a polynomial, since $\mathcal{M}$ simulates $\mathcal{R}$ and since we assume that $\mathcal{M}$ can efficiently extract polynomials for the outputs of $\mathcal{R}$. Denote the time complexity of $\mathcal{M}$ by $t_{\mathcal{M}}$.

Now, draw $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g},\mathbf{g})^{\overrightarrow{b}}) \overset{\$}{\leftarrow} \mathcal{D}(1^\lambda)$. According to Lemma 8, the probability that at least one of the polynomials $\mathcal{F}(\phi_{x_1}), \ldots, \mathcal{F}(\phi_{x_Q}) \in \mathbb{Z}_p[Z]$ is not the zero-polynomial is upper-bounded by

$$Q \cdot \frac{2d_\phi d_{\mathsf{NICA}}}{p} + t_{\mathcal{M}} \cdot \frac{2d_{\mathsf{NICA}}}{p} \in \mathsf{poly}(\lambda) \cdot \frac{d_{\mathsf{NICA}}}{p} \subseteq \mathsf{negl}(\lambda).$$

This completes the proof of Lemma 3.

C A Note on Decisional Uber-Assumptions

We define non-interactive decisional assumptions as a natural analogue to non-interactive computational assumptions.

Definition 27. *A* non-interactive decisional assumption (NIDA)*, is defined by two distributions $\mathcal{D}_0(1^\lambda)$ and $\mathcal{D}_1(1^\lambda)$ (parameterized over the security parameter λ) and the following game where two oracles (each of which may be called only once) are available to the adversary:*

Setup. *Sample a bit $\mathfrak{b} \overset{\$}{\leftarrow} \{0,1\}$. Sample a challenge $c \overset{\$}{\leftarrow} \mathcal{D}_{\mathfrak{b}}$. Output c to the adversary.*

Finalize. *When the adversary outputs a bit $\mathfrak{b}'$ as a candidate solution, the game outputs 1 if $\mathfrak{b} = \mathfrak{b}'$. Otherwise, the game outputs 0.*

We say that an adversary $\mathcal{A}$ (t, ϵ)-breaks *the assumption if $\mathcal{A}$ runs in time at most $t(\lambda)$ and*

$$\left|\Pr\left[\mathsf{NIDA}^{\mathcal{A}}(\lambda) = 1\right] - \frac{1}{2}\right| \geq \epsilon(\lambda).$$

We say that the assumption is (t, ϵ)-hard *if no adversary (t, ϵ)-breaks the assumption. We say,* NIDA *is hard, if it is $(t, \frac{1}{r})$-hard for all $t, r \in \mathsf{poly}(\lambda)$.*

It is easy to see that Theorem 1 also holds for algebraic reductions that try to transform adversaries against the weak selective unpredictability of a rational univariate VUF to solvers of decisional assumptions.

To see that Theorem 4 and Theorem 3 also hold for decisional assumptions, we need to discuss decisional Uber-assumptions, which we will do in the following.

Boyen [11] introduced two different variants of *Decisional Uber-Assumptions*: a *strict* and a *general* variant. It is easy to see that both variants of decisional Uber-assumptions are – if all involved polynomials are sparse – subsumed by the following definition:

Definition 28 (Decisional Uber-Assumptions). *Let $t = t(\lambda)$ be polynomially bounded and let $f_{A_1}, \ldots, f_{A_{q_1}}, f_{B_1}, \ldots, f_{B_{q_2}} \in \mathbb{Z}_p[Z_1, \ldots, Z_t]$ be a set of sparse polynomials.*

Let NIDA *be a non-interactive decisional assumption whose challenges consist of $1 + q_1$ source group and q_2 target group elements.*

Further, let $\mathcal{E}_0, \mathcal{E}_1$ be two distributions over $\mathbb{Z}_p^t$ s.t. $\mathcal{E}_0$ is the uniform distribution over $\mathbb{Z}_p^t$.

We call NIDA *a* ***decisional Uber-Assumption****, if the distributions $\mathcal{D}_0, \mathcal{D}_1$, which are specified in Definition 27, work as follows (for $\mathfrak{b} \in \{0,1\}$):*

```
D_b(1^λ) := {
        draw $\mathbf{g}$ by any distribution s.t. $\mathbf{g}$ is a generator of $\mathbb{G}$;
        draw $(z_1, \ldots, z_t) \overset{\$}{\leftarrow} \mathcal{E}_{\mathfrak{b}}(1^\lambda)$;
        set $a_1 := f_{A_1}(z_1, \ldots, z_t), \ldots, a_{q_1} := f_{A_{q_1}}(z_1, \ldots, z_t)$;
        set $b_1 := f_{B_1}(z_1, \ldots, z_t), \ldots, b_{q_2} := f_{B_{q_2}}(z_1, \ldots, z_t)$;
        return $(\mathbf{g}, \mathbf{g}^{a_1}, \ldots, \mathbf{g}^{a_{q_1}}, e(\mathbf{g}, \mathbf{g})^{b_1}, \ldots, e(\mathbf{g}, \mathbf{g})^{b_{q_2}})$;
    }.
```

Let $d_{\mathsf{NIDA}} = \max\{\deg f_{A_1}, \dots, \deg f_{A_{q_1}}, \deg f_{B_1}, \dots, \deg f_{B_{q_2}}\}$. *We call* d_{NIDA} *the* ***degree*** *of* NIDA *and* $q = 1 + q_1 + q_2$ *the* ***size*** *of* NIDA.

It is easy to see that Theorem 4 and Theorem 3 hold for decisional Uber-assumption, if we can prove a decisional version of Lemma 3.

For this end, let NIDA be a decisional Uber-assumption of degree d_{NIDA} and let $\mathcal{R}$ be a generic reduction that tries to decide instances of NIDA while given black-box access to an adversary $\mathcal{A}$ for the weak selective unpredictability of a VUF $\mathsf{vuf} = (\mathsf{Gen}_{\mathsf{vuf}}, \mathsf{Eval}_{\mathsf{vuf}}, \mathsf{Verify}_{\mathsf{vuf}})$.

Then, we can formulate the decisional version of Lemma 3 as follows:

Lemma 9. *Let* $d_{\mathsf{NIDA}} = d_{\mathsf{NIDA}}(\lambda) > 0$ *and* $d_\phi \in O(1)$ *be such that* $\frac{p}{d_{\mathsf{NICA}}}$ *grows faster than any polynomial. Let* $Q \in \mathsf{poly}(\lambda)$.

Let $\mathcal{R}$ *be a generic PPT reduction as above. Let* $(\phi_x)_x$ *be a family of verification equations for* vuf *of degree* d_ϕ. *Let* $\mathcal{A}$ *be an adversary on the weak* Q*-selective unpredictability of* vuf. *Assume that* $\mathcal{A}$ *aborts (i.e., returns* $\perp$*) if it is given at least one tuple* $(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i)$ *that is rejected by* $\mathsf{Verify}_{\mathsf{vuf}}$.

Draw $\mathfrak{b} \xleftarrow{\$} \{0,1\}$ *and* $(\mathbf{g}, \mathbf{g}^{\vec{a}}, e(\mathbf{g},\mathbf{g})^{\vec{b}}) \xleftarrow{\$} \mathcal{D}(1^\lambda)$ *and let*

$$\mathsf{vk}, x_0, \dots, x_Q, \pi_1, \dots, \pi_Q, \mathbf{y}_1, \dots, \mathbf{y}_Q$$

be the information sent by $\mathcal{R}$ *to* $\mathcal{A}$ *in their first interchange of a run of* $\mathcal{R}(\mathbf{g}, \mathbf{g}^{\vec{a}}, e(\mathbf{g},\mathbf{g})^{\vec{b}})$ *where* $\mathcal{A}$ *does not abort. Let* $\mathcal{F} : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$ *be the corresponding morphism, which is determined by the algebraic explanations of the group elements* $\mathsf{vk}, \pi_1, \dots, \pi_Q, \mathbf{y}_1, \dots, \mathbf{y}_Q$.

If NIDA *is hard, then, with overwhelming probability, we will have for each* $i \in [Q]$

$$\mathcal{F}(\phi_{x_i})(Z) = 0 \in \mathbb{Z}_p[Z].$$

Proof. Lemma 3 already states that with overwhelming probability each polynomial $\mathcal{F}(\phi_{x_i})(Z)$ is zero, if $\mathfrak{b} = 0$.

It remains to show, that the same holds for $\mathfrak{b} = 1$, if NIDA is hard. For this end, we construct the following meta-reduction $\mathcal{M}$ that tries to decide NIDA:

1. On receiving $(\mathbf{g}, \mathbf{g}^{\vec{a}}, e(\mathbf{g},\mathbf{g})^{\vec{b}})$, $\mathcal{M}$ has to decide if $(\mathbf{g}, \mathbf{g}^{\vec{a}}, e(\mathbf{g},\mathbf{g})^{\vec{b}})$ has been sampled by $\mathcal{D}_0(1^\lambda)$ or $\mathcal{D}_1(1^\lambda)$.
2. To this end, $\mathcal{M}$ runs $\mathcal{R}$ with input $(\mathbf{g}, \mathbf{g}^{\vec{a}}, e(\mathbf{g},\mathbf{g})^{\vec{b}})$.
3. If $\mathcal{R}$ calls $\mathcal{A}$ with information

   $$\mathsf{vk}, x_0, \dots, x_Q, \pi_1, \dots, \pi_Q, \mathbf{y}_1, \dots, \mathbf{y}_Q,$$

   $\mathcal{M}$ returns $\perp$, if there is one $i \in [Q]$ s.t. $\mathsf{Verify}_{\mathsf{vuf}}(\mathsf{vk}, x_i, \mathbf{y}_i, \pi_i) = 0$, and continues running $\mathcal{R}$.
4. Otherwise, $\mathcal{M}$ stops $\mathcal{R}$ and extracts the morphism $\mathcal{F} : \mathbb{Z}_p[V, P, Y] \to \mathbb{Z}_p[Z]$.
5. If we have

   $$\mathcal{F}(\phi_{x_i})(Z) = 0$$

   for each $i \in [Q]$, then $\mathcal{M}$ outputs 0.

6. Otherwise, $\mathcal{M}$ outputs 1.

Assume, that the claim of the lemma would be false for $\mathfrak{b} = 1$. Then, with non-negligible probability, in the above run of $\mathcal{M}$ – when given $(\mathbf{g}, \mathbf{g}^{\overrightarrow{a}}, e(\mathbf{g}, \mathbf{g})^{\overrightarrow{b}}) \xleftarrow{\$} \mathcal{D}_1(1^\lambda)$ – there would be one $i \in [Q]$ s.t.

$$\mathcal{F}(\phi_{x_i})(Z) \neq 0$$

Therefore, $\mathcal{M}$ would have a non-negligible advantage on deciding NIDA if the claim of the lemma were to be false. Since we assumed the hardness of NIDA, the lemma must be true.