Efficient Non-interactive Proof Systems for Bilinear Groups*

Jens Groth[1]** and Amit Sahai[2]***

[1] University College London
j.groth@ucl.ac.uk
[2] University of California Los Angeles
sahai@cs.ucla.edu

Abstract. Non-interactive zero-knowledge proofs and non-interactive witness-indistinguishable proofs have played a significant role in the theory of cryptography. However, lack of efficiency has prevented them from being used in practice. One of the roots of this inefficiency is that non-interactive zero-knowledge proofs have been constructed for general NP-complete languages such as Circuit Satisfiability, causing an expensive blowup in the size of the statement when reducing it to a circuit. The contribution of this paper is a general methodology for constructing very simple and efficient non-interactive zero-knowledge proofs and non-interactive witness-indistinguishable proofs that work directly for groups with a bilinear map, without needing a reduction to Circuit Satisfiability.

Groups with bilinear maps have enjoyed tremendous success in the field of cryptography in recent years and have been used to construct a plethora of protocols. This paper provides non-interactive witness-indistinguishable proofs and non-interactive zero-knowledge proofs that can be used in connection with these protocols. Our goal is to spread the use of non-interactive cryptographic proofs from mainly theoretical purposes to the large class of practical cryptographic protocols based on bilinear groups.

Keywords: Non-interactive witness-indistinguishability, non-interactive zero-knowledge, common reference string, bilinear groups.

1 Introduction

Non-interactive zero-knowledge proofs and non-interactive witness-indistinguishable proofs have played a significant role in the theory of cryptography. However, lack of

* Work presented and part of work done while participating in Securing Cyberspace: Applications and Foundations of Cryptography and Computer Security, Institute of Pure and Applied Mathematics, UCLA, 2006.

** Part of work done while at UCLA supported by NSF ITR/Cybertrust grant 0456717.

*** This research was supported in part by NSF ITR and Cybertrust programs (including grants 0627781, 0456717, 0716389, and 0205594), a subgrant from SRI as part of the Army Cyber-TA program, an equipment grant from Intel, an Okawa Research Award, and an Alfred P. Sloan Foundation Research Fellowship.

efficiency has prevented them from being used in practice. Our goal is to construct efficient and practical non-interactive zero-knowledge (NIZK) proofs and non-interactive witness-indistinguishable (NIWI) proofs.

Blum, Feldman and Micali [4] introduced NIZK proofs. Their paper and subsequent work, e.g. [19, 16, 29, 17], demonstrates that NIZK proofs exist for all of NP. Unfortunately, these NIZK proofs are all very inefficient. While leading to interesting theoretical results, such as the construction of public-key encryption secure against chosen-ciphertext attack by Dolev, Dwork and Naor [18], they have therefore not had any impact in practice.

Since we want to construct NIZK proofs that can be used in practice, it is worthwhile to identify the roots of the inefficiency in the above mentioned NIZK proofs. One drawback is that they were designed with a general NP-complete language in mind, e.g. Circuit Satisfiability. In practice, we want to prove statements such as "the ciphertext c encrypts a signature on the message m" or "the three commitments c_a, c_b, c_c contain messages a, b, c so $c = ab$". An NP-reduction of even very simple statements like these gives us big circuits containing thousands of gates and the corresponding NIZK proofs become very large.

While we want to avoid an expensive NP-reduction, it is still desirable to have a general way to express statements that arise in practice instead of having to construct non-interactive proofs on an ad hoc basis. A useful observation in this context is that many public-key cryptography protocols are based on finite abelian groups. If we can capture statements that express relations between group elements, then we can express statements that come up in practice such as "the commitments c_a, c_b, c_c contain messages so $c = ab$" or "the plaintext of c is a signature on m", as long as those commitment, encryption, and signature schemes work over the same finite group. In the paper, we will therefore construct NIWI and NIZK proofs for *group-dependent* languages.

The next issue to address is where to find suitable group-dependent languages. We will look at statements related to groups with a bilinear map, which have become widely used in the design of cryptographic protocols. Not only have bilinear groups been used to give new constructions of such cryptographic staples as public-key encryption, digital signatures, and key agreement (see [31] and the references therein), but bilinear groups have enabled the first constructions achieving goals that had never been attained before. The most notable of these is the Identity-Based Encryption scheme of Boneh and Franklin [10] (see also [6, 7, 35]), and there are many others, such as Attribute-Based Encryption [32, 22], Searchable Public-Key Encryption [9, 12, 13], and One-time Double-Homomorphic Encryption [11]. For an incomplete list of papers (currently over 200) on the application of bilinear groups in cryptography, see [2].

1.1 Our Contribution

For completeness, let us recap the definition of a bilinear group. *Please note that for notational convenience we will follow the tradition of mathematics and use additive notation*[3] *for the binary operations in G_1 and G_2.* We have a probabilistic

[3] We remark that in the cryptographic literature it is more common to use multiplicative notation for these groups, since the "discrete log problem" is believed to be hard in these groups,

polynomial time algorithm $\mathcal{G}$ that takes a security parameter as input and outputs $(\mathbf{n}, G_1, G_2, G_T, e, \mathcal{P}_1, \mathcal{P}_2)$ where

– G_1, G_2, G_T are descriptions of cyclic groups of order $\mathbf{n}$.
– The elements $\mathcal{P}_1, \mathcal{P}_2$ generate G_1 and G_2 respectively.
– $e : G_1 \times G_2$ is a non-degenerate bilinear map so $e(\mathcal{P}_1, \mathcal{P}_2)$ generates G_T and for all $a, b \in \mathbb{Z}_{\mathbf{n}}$ we have $e(a\mathcal{P}_1, b\mathcal{P}_2) = e(\mathcal{P}_1, \mathcal{P}_2)^{ab}$.
– We can efficiently compute group operations, compute the bilinear map and decide membership.

In this work, we develop a general set of highly efficient techniques for proving statements involving bilinear groups. The generality of our work extends in two directions. First, we formulate our constructions in terms of modules over commutative rings with an associated bilinear map. This framework captures all known bilinear groups with cryptographic significance – for both supersingular and ordinary elliptic curves, for groups of both prime and composite order. Second, we consider all mathematical operations that can take place in the context of a bilinear group - addition in G_1 and G_2, scalar point-multiplication, addition or multiplication of scalars, and use of the bilinear map. We also allow both group elements and exponents to be "unknowns" in the statements to be proven.

With our level of generality, it would for example be easy to write down a short statement, using the operations above, that encodes "c is an encryption of the value committed to in d under the product of the two keys committed to in a and b" where the encryptions and commitments being referred to are existing cryptographic constructions based on bilinear groups. Logical operations like AND and OR are also easy to encode into our framework using standard techniques in arithmetization.

The proof systems we build are *non-interactive*. This allows them to be used in contexts where interaction is undesirable or impossible. We first build highly efficient witness-indistinguishable proof systems, which are of independent interest. We then show how to transform these into zero-knowledge proof systems. We also provide a detailed examination of the efficiency of our constructions in various settings (depending on what type of bilinear group is used).

The security of constructions arising from our framework can be based on *any* of a variety of computational assumptions about bilinear groups (3 of which we discuss in detail here). Thus, our techniques do not rely on any one assumption in particular.

Informal statement of our results. We consider equations over variables from G_1, G_2 and $\mathbb{Z}_{\mathbf{n}}$ as described in Figure 1. We construct efficient witness-indistinguishable proofs for the simultaneous satisfiability of a set of such equations. The witness-indistinguishable proofs have perfect completeness and there are two computationally indistinguishable types of common reference strings giving respectively perfect soundness and perfect witness indistinguishability. Due to lack of space we have to refer to the full paper [28] for precise definitions.

We also consider the question of non-interactive zero-knowledge. We show that we can give zero-knowledge proofs for multi-scalar multiplication in G_1 or G_2 and for

which is also important to us. In our setting, however, it will be much more convenient to use multiplicative notation to refer to the action of the bilinear map.

quadratic equations in $\mathbb{Z}_{\mathbf{n}}$. We can also give zero-knowledge proofs for pairing product equations with $t_T = 1$. When $t_T \neq 1$ we can still give zero-knowledge proofs if we can find $\mathcal{P}_1, \mathcal{Q}_1, \ldots, \mathcal{P}_n, \mathcal{Q}_n$ such that $t_T = \prod_{i=1}^{n} e(\mathcal{P}_i, \mathcal{Q}_i)$.

Variables: $\mathcal{X}_1, \ldots, \mathcal{X}_m \in G_1$, $\mathcal{Y}_1, \ldots, \mathcal{Y}_n \in G_2$, $x_1, \ldots, x_{m'}, y_1, \ldots, y_{n'} \in \mathbb{Z}_{\mathbf{n}}$. [a]

Pairing product equation:

$$\prod_{i=1}^{n} e(\mathcal{A}_i, \mathcal{Y}_i) \cdot \prod_{i=1}^{m} e(\mathcal{X}_i, \mathcal{B}_i) \cdot \prod_{i=1}^{m} \prod_{j=1}^{n} e(\mathcal{X}_i, \mathcal{Y}_j)^{\gamma_{ij}} = t_T \ ,$$

for constants $\mathcal{A}_i \in G_1, \mathcal{B}_i \in G_2, t_T \in G_T, \gamma_{ij} \in \mathbb{Z}_{\mathbf{n}}$.

Multi-scalar multiplication equation in G_1:

$$\sum_{i=1}^{n'} y_i \mathcal{A}_i + \sum_{i=1}^{m} b_i \mathcal{X}_i + \sum_{i=1}^{m} \sum_{j=1}^{n'} \gamma_{ij} y_j \mathcal{X}_i \ = \mathcal{T}_1,$$

for constants $\mathcal{A}_i, \mathcal{T}_1 \in G_1$ and $b_i, \gamma_{ij} \in \mathbb{Z}_{\mathbf{n}}$. [b]

Multi-scalar multiplication equation in G_2:

$$\sum_{i=1}^{n} a_i \mathcal{Y}_i + \sum_{i=1}^{m'} x_i \mathcal{B}_i + \sum_{i=1}^{m'} \sum_{j=1}^{n} \gamma_{ij} x_i \mathcal{Y}_j \ = \ \mathcal{T}_2,$$

for constants $\mathcal{B}_i, \mathcal{T}_2 \in G_2$ and $a_i, \gamma_{ij} \in \mathbb{Z}_{\mathbf{n}}$.

Quadratic equation in $\mathbb{Z}_{\mathbf{n}}$:

$$\sum_{i=1}^{n'} a_i y_i + \sum_{i=1}^{m'} x_i b_i + \sum_{i=1}^{m'} \sum_{j=1}^{n'} \gamma_{ij} x_i y_j = t,$$

for constants $a_i, \gamma_{ij}, t \in \mathbb{Z}_{\mathbf{n}}$.

[a] We list variables in $\mathbb{Z}_{\mathbf{n}}$ in two separate groups because we will treat them differently in the NIWI proofs. If we wish to deal with only one group of variables in $\mathbb{Z}_{\mathbf{n}}$ we can add equations in $\mathbb{Z}_{\mathbf{n}}$ of the form $x_1 = y_1, x_2 = y_2$, etc.

[b] With multiplicative notation, these equations would be multi-exponentiation equations. We use additive notation for G_1 and G_2, since this will be notationally convenient in the paper, but stress that the discrete logarithm problem will typically be hard in these groups.

Fig. 1. Equations over groups with bilinear map.

Instantiations. In the full paper we give three possible instantiations of the bilinear groups; there are many more. The first instantiation is based on the composite order groups introduced by Boneh, Goh and Nissim [11]. We work over a composite order bilinear group $(\mathbf{n}, G, G_T, e, \mathcal{P})$ where $\mathbf{n} = \mathbf{pq}$. The security of this instantiation is based on the subgroup decision assumption that says it is hard to distinguish random elements of order $\mathbf{n}$ from random elements of order $\mathbf{q}$.

The second instantiation is based on prime order groups $(\mathbf{p}, G_1, G_2, G_T, e, \mathcal{P}_1, \mathcal{P}_2)$. Security depends on the symmetric external Diffie-Hellman (SXDH) assumption [33, 8, 1, 20, 34] that says the DDH problem is hard in both G_1 and G_2.

The third instantiation is based on prime order groups $(\mathbf{p}, G, G_T, e, \mathcal{P})$ where the decisional linear (DLIN) problem is hard. The DLIN problem introduced by Boneh, Boyen and Shacham [8] states that given $(\alpha\mathcal{P}, \beta\mathcal{P}, r\alpha\mathcal{P}, s\beta\mathcal{P}, t\mathcal{P})$ for random $\alpha, \beta, r, s \in \mathbb{Z}_\mathbf{p}$ it is hard to tell whether $t = r + s$ or t is random.

The instantiations illustrate the variety of ways bilinear groups can be constructed. We can choose prime order or composite order groups, we can use $G_1 = G_2$ and $G_1 \neq G_2$, and we can make various cryptographic assumptions. All three security assumptions have been used in the cryptographic literature to build interesting protocols.

For all three instantiations, the techniques presented here give us short NIWI proofs. In particular, the cost in proof size of each extra equation is constant and independent of the number of variables in the equation. The size of the proofs, can be computed by adding the cost, measured in group elements from G_1 or G_2, of each variable and each equation listed in Figure 2. We refer to the full paper [28] for more detailed tables.

	Subgroup decision	SXDH	DLIN
Variable in G_1 or G_2	1	2	3
Variable in $\mathbb{Z}_\mathbf{n}$ or $\mathbb{Z}_\mathbf{p}$	1	2	3
Paring product equation	1	8	9
Multi-scalar multiplication in G_1 or G_2	1	6	9
Quadratic equation in $\mathbb{Z}_\mathbf{n}$ or $\mathbb{Z}_\mathbf{p}$	1	4	6

Fig. 2. Number of group elements each variable or equation adds to the size of a NIWI proof.

1.2 Related Work

As we mentioned before, early work on NIZK proofs demonstrated that all NP-languages have non-interactive proofs, however, did not yield efficient proofs. One cause for these proofs being inefficient in practice was the need for an expensive NP-reduction to e.g. Circuit Satisfiability. Another cause of inefficiency was the reliance on the so-called hidden bits model, which even for small circuits is inefficient.

Groth, Ostrovsky, and Sahai [27, 26] investigated NIZK proofs for Circuit Satisfiability using bilinear groups. This addressed the second cause of inefficiency since their techniques give efficient proofs for Circuit Satisfiability, but to use their proofs one must still make an NP-reduction to Circuit Satisfiability thus limiting the applications. We stress that while [27, 26] used bilinear groups, their application was to build proof systems for Circuit Satisfiability. Here, we devise entirely new techniques to deal with general statements about equations in bilinear groups, *without* having to reduce to an NP-complete language.

Addressing the issue of avoiding an expensive NP-reduction we have works by Boyen and Waters [13, 14] that suggest efficient NIWI proofs for statements related to group signatures. These proofs are based on bilinear groups of composite order and rely on the subgroup decision assumption.

Groth [23] was the first to suggest a general group-dependent language and NIZK proofs for statements in this language. He investigated satisfiability of pairing product equations and only allowed group elements to be variables. He also looked only at the special case of prime order groups G, G_T with a bilinear map $e : G \times G \to G_T$ and, based on the decisional linear assumption [8], constructed NIZK proofs for such pairing product equations. However, even for very small statements, the very different and much more complicated techniques of Groth yield proofs consisting of thousands of group elements (whereas ours would be in the tens). Our techniques are much easier to understand, significantly more general, and vastly more efficient.

We summarize our comparison with other works on NIZK proofs in Figure 3.

	Inefficient	Efficient
Circuit Satisfiability	E.g. [29]	[27, 26]
Group-dependent language	[23] (restricted case)	This work

Fig. 3. Classification of NIZK proofs according to usefulness.

We note that there have been many earlier works (starting with [21]) dealing with efficient *interactive* zero-knowledge protocols for a number of algebraic relations. Here, we focus on *non-interactive* proofs. We also note that even for interactive zero-knowledge proofs, no set of techniques was known for dealing with general algebraic assertions arising in bilinear groups, as we do here.

1.3 New Techniques

[27, 26, 23] start by constructing non-interactive proofs for simple statements and then combine many of them to get more powerful proofs. The main building block in [27], for instance, is a proof that a given commitment contains either 0 or 1, which has little expressive power on its own. Our approach is the opposite: we directly construct proofs for very expressive languages; as such, our techniques are very different from previous work.

The way we achieve our generality is by viewing the groups G_1, G_2, G_T as modules over the ring $\mathbb{Z}_{\mathbf{n}}$. The ring $\mathbb{Z}_{\mathbf{n}}$ itself can also be viewed as a $\mathbb{Z}_{\mathbf{n}}$-module. We therefore look at the more general question of satisfiability of quadratic equations over $\mathbb{Z}_{\mathbf{n}}$-modules A_1, A_2, A_T with a bilinear map, see Section 2 for details. Since many bilinear groups with various cryptographic assumptions and various mathematical properties can be viewed as modules we are not bound to any particular bilinear group or any particular assumption. We remark that while bilinear groups can be interpreted as modules with a bilinear map, it is possible that there exist other interesting modules with a bilinear map that are not based on bilinear groups. We leave the existence of such modules as an interesting open problem.

Given modules A_1, A_2, A_T with a bilinear map, we construct new modules B_1, B_2, B_T, also equipped with a bilinear map, and we map the elements in A_1, A_2, A_T into B_1, B_2, B_T. More precisely, we devise commitment schemes that map variables from A_1, A_2 to the modules B_1, B_2. The commitment schemes are homomorphic with respect to the module operations but also with respect to the bilinear map.

Our techniques for constructing witness-indistinguishable proofs are fairly involved mathematically, but we will try to present some high level intuition here. (We give more detailed intuition later in Section 5, where we present our main proof system). The main idea is the following: because our commitment schemes are homomorphic *and* we equip them with a bilinear map, we can take the equation that we are trying to prove, and just replace the variables in the equation with commitments to those variables. Of course, because the commitment schemes are hiding, the equations will no longer be valid. Intuitively, however, we can extract out the additional terms introduced by the randomness of the commitments: if we give away these terms in the proof, then this would be a *convincing* proof of the equation's validity (again, because of the homomorphic properties). But, giving away these terms might destroy witness indistinguishability. Suppose, however, that there is only one "additional term" introduced by substituting the commitments. Then, because it would be the unique value which makes the equation true, giving it away would preserve witness indistinguishability! In general, we are not so lucky. But if there are many terms, that means that these terms are not unique, and because of the nice algebraic environment that we work in, we can randomize these terms so that the equation is still true, but so that we effectively reduce to the case of there being a single term being given away with a unique value.

1.4 Applications

Independently of our work, Boyen and Waters [14] have constructed non-interactive proofs that they use for group signatures (see also their earlier paper [13]). These proofs can be seen as examples of the NIWI proofs in instantiation 1. Subsequent to the announcement of our work, several papers have built upon it: Chandran, Groth and Sahai [15] have constructed ring-signatures of sub-linear size using the NIWI proofs in the first instantiation, which is based on the subgroup decision problem. Groth and Lu [25] have used the NIWI and NIZK proofs from instantiation 3 to construct a NIZK proof for the correctness of a shuffle. Groth [24] has used the NIWI and NIZK proofs from instantiation 3 to construct a fully anonymous group signature scheme. Belenkiy, Chase, Kohlweiss and Lysyanskaya [3] have used instantiations 2 and 3 to construct non-interactive anonymous credentials. Also, by attaching NIZK proofs to semantically secure public-key encryption in any instantiation we get an efficient non-interactive verifiable cryptosystem. Boneh [5] has suggested using this for optimistic fair exchange [30], where two parties use a trusted but lazy third party to guarantee fairness.

2 Modules with Bilinear Maps

Let $(\mathcal{R}, +, \cdot, 0, 1)$ be a finite commutative ring. Recall that an $\mathcal{R}$-module A is an abelian group $(A, +, 0)$ where the ring acts on the group such that $\forall r, s \in \mathcal{R}\ \forall x, y \in A$:

$$(r+s)x = rx + sx \ \wedge\ r(x+y) = rx + ry \ \wedge\ r(sx) = (rs)x \ \wedge\ 1x = x.$$

A cyclic group G of order $\mathbf{n}$ can in a natural way be viewed as a $\mathbb{Z}_{\mathbf{n}}$-module. We will observe that all the equations in Figure 1 can be viewed as equations over $\mathbb{Z}_{\mathbf{n}}$-modules with a bilinear map. To generalize completely, let $\mathcal{R}$ be a finite commutative ring and let A_1, A_2, A_T be finite $\mathcal{R}$-modules with a bilinear map $f : A_1 \times A_2 \to A_T$. We consider quadratic equations over variables $x_1, \ldots, x_m \in A_1, y_1, \ldots, y_n \in A_2$ of the form

$$\sum_{j=1}^{n} f(a_j, y_j) + \sum_{i=1}^{m} f(x_i, b_i) + \sum_{i=1}^{m}\sum_{j=1}^{n} \gamma_{ij} f(x_i, y_j) = t.$$

In order to simplify notation, let us for $x_1, \ldots, x_n \in A_1, y_1, \ldots, y_n \in A_2$ define

$$\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{i=1}^{n} f(x_i, y_i).$$

The equations can now be written as

$$\boldsymbol{a} \cdot \boldsymbol{y} + \boldsymbol{x} \cdot \boldsymbol{b} + \boldsymbol{x} \cdot \Gamma \boldsymbol{y} = t.$$

We note for future use that due to the bilinear properties of f, we have for any matrix $\Gamma \in \mathrm{Mat}_{m \times n}(\mathcal{R})$ and for any $x_1, \ldots, x_m, y_1, \ldots, y_n$ that $\boldsymbol{x} \cdot \Gamma \boldsymbol{y} = \Gamma^{\top} \boldsymbol{x} \cdot \boldsymbol{y}$.

Let us now return to the equations in Figure 1 and see how they can be recast as quadratic equations over $\mathbb{Z}_{\mathbf{n}}$-modules with a bilinear map.

Pairing product equations: Define $\mathcal{R} = \mathbb{Z}_{\mathbf{n}}, A_1 = G_1, A_2 = G_2, A_T = G_T, f(x, y) = e(x, y)$ and we can rewrite[4] the pairing product equation as $(\boldsymbol{\mathcal{A}} \cdot \boldsymbol{\mathcal{Y}})(\boldsymbol{\mathcal{X}} \cdot \boldsymbol{\mathcal{B}})(\boldsymbol{\mathcal{X}} \cdot \Gamma \boldsymbol{\mathcal{Y}}) = t_T$.

Multi-scalar multiplication in G_1: Define $\mathcal{R} = \mathbb{Z}_{\mathbf{n}}, A_1 = G_1, A_2 = \mathbb{Z}_{\mathbf{n}}, A_T = G_1, f(\mathcal{X}, y) = y\mathcal{X}$ and we can rewrite the multi-scalar multiplication equation as $\boldsymbol{\mathcal{A}} \cdot \boldsymbol{y} + \boldsymbol{\mathcal{X}} \cdot \boldsymbol{b} + \boldsymbol{\mathcal{X}} \cdot \Gamma \boldsymbol{y} = \mathcal{T}_1$.

Multi-scalar multiplication in G_2: Define $\mathcal{R} = \mathbb{Z}_{\mathbf{n}}, A_1 = \mathbb{Z}_{\mathbf{n}}, A_2 = G_2, A_T = G_2, f(x, \mathcal{Y}) = x\mathcal{Y}$ and we can rewrite the multi-scalar multiplication equation as $\boldsymbol{a} \cdot \boldsymbol{\mathcal{Y}} + \boldsymbol{x} \cdot \boldsymbol{\mathcal{B}} + \boldsymbol{x} \cdot \Gamma \boldsymbol{\mathcal{Y}} = \mathcal{T}_2$.

Quadratic equation in $\mathbb{Z}_{\mathbf{n}}$: Define $\mathcal{R} = \mathbb{Z}_{\mathbf{n}}, A_1 = \mathbb{Z}_{\mathbf{n}}, A_2 = \mathbb{Z}_{\mathbf{n}}, A_T = \mathbb{Z}_{\mathbf{n}}, f(x, y) = xy \bmod \mathbf{n}$ and we can rewrite the quadratic equation in $\mathbb{Z}_{\mathbf{n}}$ as $\boldsymbol{a} \cdot \boldsymbol{y} + \boldsymbol{x} \cdot \boldsymbol{b} + \boldsymbol{x} \cdot \Gamma \boldsymbol{y} = t$.

From now on, we will therefore focus on the more general problem of constructing non-interactive composable witness-indistinguishable proofs for satisfiability of quadratic equations over $\mathcal{R}$-modules A_1, A_2, A_T (using additive notation for all modules) with a bilinear map f.

3 Commitment from Modules

In our NIWI proofs we will commit to the variables $x_1, \ldots, x_m \in A_1, y_1, \ldots, y_n \in A_2$. We do this by mapping them into other $\mathcal{R}$-modules B_1, B_2 and making the commitments in those modules.

[4] We use multiplicative notation here, because, usually G_T is written multiplicatively in the literature. When we work with the abstract modules, however, we will use additive notation.

Let us for now just consider how to commit to elements from one $\mathcal{R}$-module A. The public key for the commitment scheme will describe another $\mathcal{R}$-module B and $\mathcal{R}$-linear maps $\iota : A \to B$ and $p : B \to A$. It will also contain elements $u_1, \ldots, u_n \in B$. To commit to $x \in A$ we pick $r_1, \ldots, r_n \leftarrow \mathcal{R}$ at random and compute the commitment

$$c := \iota(x) + \sum_{i=1}^{n} r_i u_i.$$

Our commitment scheme will have two types of commitment keys.

Hiding key: A hiding key contains $(B, \iota, p, u_1, \ldots, u_n)$ such that $\iota(G) \subseteq \langle u_1, \ldots, u_n \rangle$. The commitment $c := \iota(x) + \sum_{i=1}^{n} r_i u_i$ is perfectly hiding when $r_1, \ldots, r_n$ are chosen at random from $\mathcal{R}$.

Binding key: A binding key contains $(B, \iota, p, u_1, \ldots, u_n)$ such that $\forall i : p(u_i) = 0$ and $\iota \circ p$ is the identity.[5] The commitment $c := \iota(x) + \sum_{i=1}^{n} r_i u_i$ is perfectly binding, since it determines x as $p(c) = p(\iota(x)) = x$.[6]

Computational indistinguishability: The main assumption that we will be making throughout this paper is that the distribution of hiding keys and the distribution of binding keys are computationally indistinguishable. Witness-indistinguishability of our NIWI proofs and later the zero-knowledge property of our NIZK proofs will rely on this property.

Often we will commit to many elements at a time so let us define some convenient notation. Given elements $x_1, \ldots, x_m$ we write $\boldsymbol{c} := \iota(\boldsymbol{x}) + R\boldsymbol{u}$ with $R \in \mathrm{Mat}_{m \times n}(\mathcal{R})$ for making commitments $c_1, \ldots, c_m$ computed as $c_i := \iota(x_i) + \sum_{j=1}^{n} r_{ij} u_j$.

The treatment of commitments using the language of modules generalizes several previous works dealing with commitments over bilinear groups, including [11, 27, 26, 23, 36]. We refer to the full paper [28] for a demonstration of how the commitment scheme can be instantiated with respectively the subgroup decision, the SXDH and the DLIN assumptions.

4 Setup

In our NIWI proofs the common reference string will contain commitment keys to commit to elements in respectively A_1 and A_2. These commitment keys specify $B_1, \iota_1, p_1, u_1, \ldots, u_{\hat{m}}$ so $\iota_1 \circ p_1$ is the identity map and $B_2, \iota_2, p_2, v_1, \ldots, v_{\hat{n}}$ so $\iota_2 \circ p_2$ is the identity map. In addition, the common reference string will also specify a third $\mathcal{R}$-module B_T together with $\mathcal{R}$-linear maps $\iota_T : A_T \to B_T$ and $p_T : B_T \to A_T$ so $\iota_T \circ p_T$ is the identity map. There will be a bilinear map $F : B_1 \times B_2 \to B_T$ as well. We require that the maps are commutative. We refer to Figure 4 for an overview of the modules and the maps.

[5] In the full paper [28], we also consider the case where $\iota \circ p$ is not the identity. In particular, in the instantiation based on the subgroup decision problem, $\iota \circ p$ is the projection on the order $\mathbf{p}$ subgroup of G.

[6] The map p is not efficiently computable. However, one can imagine scenarios where a secret key will make p efficiently computable making the commitment scheme a cryptosystem with p being the decryption operation.

$$\begin{array}{ccccc} A_1 & \times & A_2 & \xrightarrow{f} & A_T \\ \iota_1 \downarrow\uparrow p_1 & & \iota_2 \downarrow\uparrow p_2 & & \iota_T \downarrow\uparrow p_T \\ B_1 & \times & B_2 & \xrightarrow{F} & B_T \end{array}$$

$$\forall x \in A_1 \, \forall y \in A_2 : \; F(\iota_1(x), \iota_2(y)) = \iota_T(f(x,y))$$
$$\forall x \in B_1 \, \forall y \in B_2 : \; f(p_1(x), p_2(x)) = p_T(F(x,y))$$

Fig. 4. Modules and maps between them.

For notational convenience, let us define for $\boldsymbol{x} \in B_1^n, \boldsymbol{y} \in B_2^n$ that

$$\boldsymbol{x} \bullet \boldsymbol{y} = \sum_{i=1}^{n} F(x_i, y_i).$$

The final part of the common reference string is a set of matrices $H_1, \ldots, H_\eta \in \mathrm{Mat}_{\hat{m} \times \hat{n}}(\mathcal{R})$ that all satisfy $\boldsymbol{u} \bullet H_i \boldsymbol{v} = 0$.[7]

There will be two different types of settings of interest to us.

Soundness setting: In the soundness setting, we require that the commitment keys are binding so we have $p_1(\boldsymbol{u}) = \boldsymbol{0}$ and $p_2(\boldsymbol{v}) = \boldsymbol{0}$.

Witness-indistinguishability setting: In the witness-indistinguishability setting we have hiding commitment keys, so $\iota_1(G_1) \subseteq \langle u_1, \ldots, u_{\widehat{m}} \rangle$ and $\iota_2(G_2) \subseteq \langle v_1, \ldots, v_{\widehat{n}} \rangle$. We also require that $H_1, \ldots, H_\eta$ generate the R-module of all matrices H so $\boldsymbol{u} \bullet H \boldsymbol{v} = 0$. As we will see in the next section, these matrices play a role as randomizers in the witness-indistinguishability proof.

Computational indistinguishability: The (only) computational assumption this paper is based on is that the two settings can be set up in a computationally indistinguishable way. The instantiations show that there are many ways to get such computationally indistinguishable soundness and witness-indistinguishability setups.

All three instantiations based on the subgroup decision, the SXDH and the DLIN assumptions enable us to make this kind of setup, see the full paper [28] for details.

5 Proving that Committed Values Satisfy a Quadratic Equation

Recall that in our setting, a quadratic equation looks like the following:

$$\boldsymbol{a} \cdot \boldsymbol{y} + \boldsymbol{x} \cdot \boldsymbol{b} + \boldsymbol{x} \cdot \Gamma \boldsymbol{y} = t, \tag{1}$$

[7] The number of matrices $H_1, \ldots, H_\eta$ depends on the concrete setting. In many cases, we need no matrices at all and we have $\eta = 0$, but there are also cases where they are needed.

with constants $\boldsymbol{a} \in A_1^n, \boldsymbol{b} \in A_2^m, \Gamma \in \mathrm{Mat}_{m\times n}(\mathcal{R}), t \in A_T$. We will first consider the case of a single quadratic equation of the above form. The first step in our NIWI proof will be to commit to all the variables $\boldsymbol{x}, \boldsymbol{y}$. The commitments are of the form

$$\boldsymbol{c} = \iota_1(\boldsymbol{x}) + R\boldsymbol{u} \quad , \quad \boldsymbol{d} = \iota_2(\boldsymbol{y}) + S\boldsymbol{v}, \tag{2}$$

with $R \in \mathrm{Mat}_{m\times\widehat{m}}(\mathcal{R}), S \in \mathrm{Mat}_{n\times\widehat{n}}(\mathcal{R})$. The prover's task is to convince the verifier that the commitments contain $\boldsymbol{x} \in A_1^m, \boldsymbol{y} \in A_2^n$ that satisfy the quadratic equation. (Note that for all equations we will use these same commitments.)

Intuition. Before giving the proof let us give some intuition. In the previous sections, we have carefully set up our commitments so that the commitments themselves also "behave" like the values being committed to: they also belong to modules (the B modules) equipped with a bilinear map (the map F, also implicitly used in the $\bullet$ operation). Given that we have done this, a natural idea is to take the quadratic equation (1), and "plug in" the commitments (2) in place of the variables; let us evaluate:

$$\iota_1(\boldsymbol{a}) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}) + \boldsymbol{c} \bullet \Gamma\boldsymbol{d}.$$

After some computations, where we expand the commitments (2), make use of the bilinearity of $\bullet$, and rearrange terms (the details can be found in the proof of Theorem 1 in the full paper [28]) we get

$$\begin{aligned}&\Big(\iota_1(\boldsymbol{a}) \bullet \iota_2(\boldsymbol{y}) + \iota_1(\boldsymbol{x}) \bullet \iota_2(\boldsymbol{b}) + \iota_1(\boldsymbol{x}) \bullet \Gamma\iota_2(\boldsymbol{y})\Big)\\ &+\iota_1(\boldsymbol{a}) \bullet S\boldsymbol{v} + R\boldsymbol{u} \bullet \iota_2(\boldsymbol{b}) + \iota_1(\boldsymbol{x}) \bullet \Gamma S\boldsymbol{v} + R\boldsymbol{u} \bullet \Gamma\iota_2(\boldsymbol{y}) + R\boldsymbol{u} \bullet \Gamma S\boldsymbol{v}.\end{aligned}$$

By the commutative properties of the maps, the first group of three terms is equal to $\iota_T(t)$, if Equation 1 holds. Looking at the remaining terms, note that the verifier knows $\boldsymbol{u}$ and $\boldsymbol{v}$. Using the fact that bilinearity implies that for any $\boldsymbol{x}, \boldsymbol{y}$ we have $\boldsymbol{x} \bullet \Gamma\boldsymbol{y} = \Gamma^\top\boldsymbol{x} \bullet \boldsymbol{y}$, we can sort the remaining terms so that they match either $\boldsymbol{u}$ or $\boldsymbol{v}$ to get (again see the proof of Theorem 1 in the full paper for details)

$$\iota_T(t) + \boldsymbol{u} \bullet \Big(R^\top\iota_2(\boldsymbol{b}) + R^\top\Gamma\iota_2(\boldsymbol{y}) + R^\top\Gamma S\boldsymbol{v}\Big) + \Big(S^\top\iota_1(\boldsymbol{a}) + S^\top\Gamma^\top\iota_1(\boldsymbol{x})\Big) \bullet \boldsymbol{v}. \tag{3}$$

Now, for sake of intuition, let us make some simplifying assumptions: Let's assume that we're working in a symmetric case where $A_1 = A_2$, and $B_1 = B_2$, and therefore $\boldsymbol{u} = \boldsymbol{v}$ and, so, the above equation can be simplified further to get:

$$\iota_T(t) + \boldsymbol{u} \bullet \Big(R^\top\iota_2(\boldsymbol{b}) + R^\top\Gamma\iota_2(\boldsymbol{y}) + R^\top\Gamma S\boldsymbol{u} + S^\top\iota_1(\boldsymbol{a}) + S^\top\Gamma^\top\iota_1(\boldsymbol{x})\Big).$$

Now, suppose the prover gives to the verifier as his proof $\boldsymbol{\pi} = \Big(R^\top\iota_2(\boldsymbol{b}) + R^\top\Gamma\iota_2(\boldsymbol{y}) + S^\top\iota_1(\boldsymbol{a}) + S^\top\Gamma^\top\iota_1(\boldsymbol{x})\Big)$. The verifier would then check that the following *verification equation* holds:

$$\iota_1(\boldsymbol{a}) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}) + \boldsymbol{c} \bullet \Gamma\boldsymbol{d} = \iota_T(t) + \boldsymbol{u} \bullet \boldsymbol{\pi}.$$

It is easy to see that this proof would be convincing in the soundness setting, because we have that $p_1(\boldsymbol{u}) = \mathbf{0}$. Then the verifier would know (but not be able to compute) that by applying the maps p_1, p_2, p_T we get

$$\boldsymbol{a} \bullet p_2(\boldsymbol{d}) + p_1(\boldsymbol{c}) \bullet \boldsymbol{b} + p_1(\boldsymbol{c}) \bullet \Gamma p_2(\boldsymbol{d}) = t + p_1(\boldsymbol{u}) \bullet p_2(\boldsymbol{\pi}) = t.$$

This gives us soundness, since $\boldsymbol{x} := p_1(\boldsymbol{c})$ and $\boldsymbol{y} := p_2(\boldsymbol{d})$ satisfy the equations.

The remaining problem is to get witness-indistinguishability. Recall that in the witness-indistinguishability setting, the commitments are perfectly hiding. Therefore, in the verification equation, nothing except for $\boldsymbol{\pi}$ has any information about $\boldsymbol{x}$ and $\boldsymbol{y}$ except for the information that can be inferred from the quadratic equation itself. So, let's consider two cases:

1. Suppose that $\boldsymbol{\pi}$ is the unique value so that the verification equation is valid. In this case, we trivially have witness indistinguishability, since this means that all witnesses would lead to the same value for $\boldsymbol{\pi}$.
2. The simple case above might seem too good to be true, but let's see what it means if it isn't true. If two values $\boldsymbol{\pi}$ and $\boldsymbol{\pi}'$ both satisfy the verification equation, then just subtracting the equations shows that $\boldsymbol{u} \bullet (\boldsymbol{\pi} - \boldsymbol{\pi}') = 0$. On the other hand, recall that in the witness indistinguishability setting, the $\boldsymbol{u}$ vectors generate the entire space where $\boldsymbol{\pi}$ or $\boldsymbol{\pi}'$ live, and furthermore we know that the matrices $H_1, \ldots, H_\eta$ generate all H such that $\boldsymbol{u} \bullet H\boldsymbol{u} = 0$. Therefore, let's choose $r_1, \ldots, r_\eta$ at random, and consider the distribution $\boldsymbol{\pi}'' = \boldsymbol{\pi} + \sum_{i=1}^{\eta} r_i H_i \boldsymbol{u}$. We thus obtain the same distribution on $\boldsymbol{\pi}''$ regardless of what $\boldsymbol{\pi}$ we started from, and such that $\boldsymbol{\pi}''$ always satisfies the verification equation.

Thus, for the symmetric case we obtain a witness indistinguishable proof system. For the general non-symmetric case, instead of having just $\boldsymbol{\pi}$ for the $\boldsymbol{u}$ part of Equation 3, we would also have a proof $\boldsymbol{\theta}$ for the $\boldsymbol{v}$ part. In this case, we would also have to make sure that this split does not reveal any information about the witness. What we will do is to randomize the proofs such that they get a uniform distribution on all $\boldsymbol{\pi}, \boldsymbol{\theta}$ that satisfy the verification equation. If we pick $T \leftarrow \mathrm{Mat}_{\widehat{n} \times \widehat{m}}(\mathcal{R})$ at random we have that $\boldsymbol{\theta} + T\boldsymbol{u}$ completely randomizes $\boldsymbol{\theta}$. The part we add in $\boldsymbol{\theta}$ can be "subtracted" from $\boldsymbol{\pi}$ by observing that

$$\iota_T(t) + \boldsymbol{u} \bullet \boldsymbol{\pi} + \boldsymbol{\theta} \bullet \boldsymbol{v} = \iota_T(t) + \boldsymbol{u} \bullet \Big(\boldsymbol{\pi} - T^\top \boldsymbol{v}\Big) + \Big(\boldsymbol{\theta} + T\boldsymbol{u}\Big) \bullet \boldsymbol{v}.$$

This leads to a unique distribution of proofs for the general non-symmetric case as well.

Having explained the intuition behind the proof system, we proceed to a formal description and proof of security properties.

Proof: Pick $T \leftarrow \mathrm{Mat}_{\widehat{n} \times \widehat{m}}(\mathcal{R}), r_1, \ldots, r_\eta \leftarrow \mathcal{R}$ at random. Compute

$$\begin{aligned} \boldsymbol{\pi} &:= R^\top \iota_2(\boldsymbol{b}) + R^\top \Gamma \iota_2(\boldsymbol{y}) + R^\top \Gamma S \boldsymbol{v} - T^\top \boldsymbol{v} + \sum_{i=1}^{\eta} r_i H_i \boldsymbol{v} \\ \boldsymbol{\theta} &:= S^\top \iota_1(\boldsymbol{a}) + S^\top \Gamma^\top \iota_1(\boldsymbol{x}) + T\boldsymbol{u} \end{aligned}$$

and return the proof $(\boldsymbol{\theta}, \boldsymbol{\pi})$.

Verification: Return 1 if and only if

$$\iota_1(\boldsymbol{a}) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}) + \boldsymbol{c} \bullet \Gamma \boldsymbol{d} = \iota_T(t) + \boldsymbol{u} \bullet \boldsymbol{\pi} + \boldsymbol{\theta} \bullet \boldsymbol{v}.$$

Perfect completeness of our NIWI proof will follow from the following theorem no matter whether we are in the soundness setting or the witness-indistinguishability setting. We refer to the full paper [28] for the proof.

Theorem 1. *Given $\boldsymbol{x} \in A_1^m, \boldsymbol{y} \in A_2^n, R \in \mathrm{Mat}_{m \times \widehat{m}}(\mathcal{R}), S \in \mathrm{Mat}_{n \times \widehat{n}}(\mathcal{R})$ satisfying*

$$\boldsymbol{c} = \iota_1(\boldsymbol{x}) + R\boldsymbol{u} \quad , \quad \boldsymbol{d} = \iota_2(\boldsymbol{y}) + S\boldsymbol{v} \quad , \quad \boldsymbol{a} \cdot \boldsymbol{y} + \boldsymbol{x} \cdot \boldsymbol{b} + \boldsymbol{x} \cdot \Gamma \boldsymbol{y} = t,$$

we have for all choices of $T, r_1, \ldots, r_\eta$ that the proofs $\boldsymbol{\pi}, \boldsymbol{\theta}$ constructed as above will be accepted.

Perfect soundness of our NIWI proof follows from the following theorem.

Theorem 2. *In the soundness setting, where we have $p_1(\boldsymbol{u}) = \boldsymbol{0}$ and $p_2(\boldsymbol{v}) = \boldsymbol{0}$, a valid proof implies $\boldsymbol{a} \cdot p_2(\boldsymbol{d}) + p_1(\boldsymbol{c}) \cdot \boldsymbol{b} + p_1(\boldsymbol{c}) \cdot \Gamma p_2(\boldsymbol{d}) = t$.*

Proof. An acceptable proof $\boldsymbol{\pi}, \boldsymbol{\theta}$ satisfies $\iota(a) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}) + \boldsymbol{c} \bullet \Gamma \boldsymbol{d} = \iota_T(t) + \boldsymbol{u} \bullet \boldsymbol{\pi} + \boldsymbol{\theta} \bullet \boldsymbol{v}$. The commutative property of the linear and bilinear maps gives us

$$\begin{aligned} & p_1(\iota_1(\boldsymbol{a})) \cdot p_2(\boldsymbol{d}) + p_1(\boldsymbol{c}) \cdot p_2(\iota_2(\boldsymbol{b})) + p_1(\boldsymbol{c}) \cdot \Gamma p_2(\boldsymbol{d}) \\ = \; & p_T(\iota_T(t)) + p_1(\boldsymbol{u}) \cdot p_2(\boldsymbol{\pi}) + p_1(\boldsymbol{\theta}) \cdot p_2(\boldsymbol{v}) = p_T(\iota_T(t)). \end{aligned}$$

□

Composable witness-indistinguishability follows from the following theorem, which we prove in the full paper [28].

Theorem 3. *In the witness-indistinguishable setting where $\iota_1(G_1) \subseteq \langle u_1, \ldots, u_{\widehat{m}} \rangle$, $\iota_2(G_2) \subseteq \langle v_1, \ldots, v_{\widehat{n}} \rangle$ and $H_1, \ldots, H_\eta$ generate all matrices H so $\boldsymbol{u} \bullet H\boldsymbol{v} = 0$, all satisfying witnesses $\boldsymbol{x}, \boldsymbol{y}, R, S$ yield proofs $\boldsymbol{\pi} \in \langle v_1, \ldots, v_{\widehat{n}} \rangle^{\widehat{m}}$ and $\boldsymbol{\theta} \in \langle u_1, \ldots, u_{\widehat{m}} \rangle^{\widehat{n}}$ that are uniformly distributed conditioned on the verification equation $\iota_1(\boldsymbol{a}) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}) + \boldsymbol{c} \bullet \Gamma \boldsymbol{d} = \iota_T(t) + \boldsymbol{u} \bullet \boldsymbol{\pi} + \boldsymbol{\theta} \bullet \boldsymbol{v}$.*

6 NIWI Proof for Satisfiability of a Set of Quadratic Equations

We will now give the full composable NIWI proof for satisfiability of a set of quadratic equations in a module with a bilinear map. The cryptographic assumption we make is that the common reference string is created by one of two algorithms K or S and that their outputs are computationally indistinguishable. The first algorithm outputs a common reference string that specifies a soundness setting, whereas the second algorithm outputs a common reference string that specifies a witness-indistinguishability setting.

Setup: $gk := (\mathcal{R}, A_1, A_2, A_T, f) \leftarrow \mathcal{G}(1^k)$.
Soundness string:
$\sigma := (B_1, B_2, B_T, F, \iota_1, p_1, \iota_2, p_2, \iota_T, p_T, \boldsymbol{u}, \boldsymbol{v}, H_1, \ldots, H_\eta) \leftarrow K(gk)$.

Witness-indistinguishability string:
$\sigma := (B_1, B_2, B_T, F, \iota_1, p_1, \iota_2, p_2, \iota_T, p_T, \boldsymbol{u}, \boldsymbol{v}, H_1, \ldots, H_\eta) \leftarrow S(gk)$.

Proof: The input consists of gk, σ, a list of quadratic equations $\{(\boldsymbol{a}_i, \boldsymbol{b}_i, \Gamma_i, t_i)\}_{i=1}^N$ and a satisfying witness $\boldsymbol{x} \in A_1^m, \boldsymbol{y} \in A_2^n$.

Pick at random $R \leftarrow \mathrm{Mat}_{m \times \widehat{m}}(\mathcal{R})$ and $S \leftarrow \mathrm{Mat}_{n \times \widehat{n}}(\mathcal{R})$ and commit to all the variables as $\boldsymbol{c} := \boldsymbol{x} + R\boldsymbol{u}$ and $\boldsymbol{d} := \boldsymbol{y} + S\boldsymbol{v}$.

For each equation $(\boldsymbol{a}_i, \boldsymbol{b}_i, \Gamma_i, t_i)$ make a proof as described in Section 5. In other words, pick $T_i \leftarrow \mathrm{Mat}_{\widehat{n} \times \widehat{m}}(\mathcal{R})$ and $r_{i1}, \ldots, r_{i\eta} \leftarrow \mathcal{R}$ compute

$$\boldsymbol{\pi}_i := R^\top \iota_2(\boldsymbol{b}_i) + R^\top \Gamma \iota_2(\boldsymbol{y}) + R^\top \Gamma S \boldsymbol{v} - T_i^\top \boldsymbol{v} + \sum_{j=1}^{\eta} r_{ij} H_j \boldsymbol{v}$$

$$\boldsymbol{\theta}_i := S^\top \iota_1(\boldsymbol{a}_i) + S^\top \Gamma^\top \iota_1(\boldsymbol{x}) + T_i \boldsymbol{u}.$$

Output the proof $(\boldsymbol{c}, \boldsymbol{d}, \{(\boldsymbol{\pi}_i, \boldsymbol{\theta}_i)\}_{i=1}^N)$.

Verification: The input is gk, σ, $\{(\boldsymbol{a}_i, \boldsymbol{b}_i, \Gamma_i, t_i)\}_{i=1}^N$ and the proof $(\boldsymbol{c}, \boldsymbol{d}, \{(\boldsymbol{\pi}_i, \boldsymbol{\theta}_i)\})$.

For each equation check

$$\iota_1(\boldsymbol{a}_i) \bullet \boldsymbol{d} + \boldsymbol{c} \bullet \iota_2(\boldsymbol{b}_i) + \boldsymbol{c} \bullet \Gamma_i \boldsymbol{d} = \iota_T(t_i) + \boldsymbol{u} \bullet \boldsymbol{\pi}_i + \boldsymbol{\theta}_i \bullet \boldsymbol{v}.$$

Output 1 if all the checks pass, else output 0.

The construction gives us a NIWI proof. We prove the following theorem in the full paper [28].

Theorem 4. *The protocol given above is a NIWI proof for satisfiability of a set of quadratic equations with perfect completeness, perfect soundness and composable witness-indistinguishability.*

Proof of knowledge. We observe that if K outputs an additional secret piece of information ξ that makes it possible to efficiently compute p_1 and p_2, then it is straightforward to compute the witness $\boldsymbol{x} = p_1(\boldsymbol{c})$ and $\boldsymbol{y} = p_2(\boldsymbol{d})$, so the proof is a perfect proof of knowledge.

Proof size. The size of the common reference string is $\widehat{m}$ elements in B_1 and $\widehat{n}$ elements in B_2 in addition to the description of the modules, the maps and $H_1, \ldots, H_\eta$. The size of the proof is $m + N\widehat{n}$ elements in B_1 and $n + N\widehat{m}$ elements in B_2.

Typically, $\widehat{m}$ and $\widehat{n}$ will be small, giving us a proof size that is $O(m + n + N)$ elements in B_1 and B_2. The proof size may thus be smaller than the description of the statement, which can be of size up to Nn elements in A_1, Nm elements in A_2, Nmn elements in $\mathcal{R}$ and N elements in A_T.

6.1 NIWI Proofs for Bilinear Groups

We will now outline the strategy for making NIWI proofs for satisfiability of a set of quadratic equations over bilinear groups. As we described in Section 2, there are four different types of equations, corresponding to the following four combinations of $\mathbb{Z}_{\mathbf{n}}$-modules:

Pairing product equations: $A_1 = G_1, A_2 = G_2, A_T = G_T, f(\mathcal{X}, \mathcal{Y}) = e(\mathcal{X}, \mathcal{Y})$.
Multi-scalar multiplication in G_1: $A_1 = G_1, A_2 = \mathbb{Z}_{\mathbf{n}}, A_T = G_1, f(\mathcal{X}, y) = y\mathcal{X}$.
Multi-scalar multiplication in G_2: $A_1 = \mathbb{Z}_{\mathbf{n}}, A_2 = G_2, A_T = G_T, f(x, \mathcal{Y}) = x\mathcal{Y}$.
Quadratic equations in $\mathbb{Z}_{\mathbf{n}}$: $A_1 = \mathbb{Z}_{\mathbf{n}}, A_2 = \mathbb{Z}_{\mathbf{n}}, A_T = \mathbb{Z}_{\mathbf{n}}, f(x, y) = xy \bmod \mathbf{n}$.

The common reference string will specify commitment schemes to respectively scalars and group elements. We first commit to all the variables and then make the NIWI proofs that correspond to the types of equations that we are looking at. It is important that we use the same commitment schemes and commitments for all equations, i.e., for instance we only commit to a scalar x once and we use the same commitment in the proof whether the equation x is involved in is a multi-scalar multiplication in G_2 or a quadratic equations in $\mathbb{Z}_{\mathbf{n}}$. The use of the same commitment in all the equations is necessary to ensure a consistent choice of x throughout the proof. As a consequence of this we use the same module B_1' to commit to x in both multi-scalar multiplication in G_2 and quadratic equations in $\mathbb{Z}_{\mathbf{n}}$. We therefore end up with at most four different modules B_1, B_1', B_2, B_2' to commit to respectively $\mathcal{X}, x, \mathcal{Y}, y$ variables. We give the full construction of efficient NIWI proofs for the three instantiations based on subgroup decision, SXDH and DLIN respectively in the full paper [28].

7 Zero-Knowledge

We will show that in many cases it is possible to make zero-knowledge proofs for satisfiability of quadratic equations. An obvious strategy is to use our NIWI proofs directly, however, such proofs may not be zero-knowledge because the zero-knowledge simulator may not be able to compute any witness for satisfiability of the equations. It turns out that the strategy is better than it seems at first sight, because we will often be able to modify the set of quadratic equations into an equivalent set of quadratic equations where a witness can be found.

We consider first the case where $A_1 = \mathcal{R}, A_2 = A_T, f(r, y) = ry$ and where S outputs an extra piece of information τ that makes it possible to trapdoor open the commitments in B_1. More precisely, τ permits the computation of $\boldsymbol{s} \in \mathcal{R}^{\widehat{m}}$ so $\iota_1(1) = \iota_1(0) + \boldsymbol{s}^\top \boldsymbol{u}$. We remark that this is a common case; in bilinear groups both multi-scalar multiplication equations in G_1, G_2 and quadratic equations in $\mathbb{Z}_{\mathbf{n}}$ have this structure.

Define $c = \iota_1(1)$ to be a commitment to $\phi = 1$. Let us rewrite the equations in the statement as

$$\boldsymbol{a}_i \cdot y + f(-\phi, t_i) + \boldsymbol{x} \cdot \boldsymbol{b}_i + \boldsymbol{x} \cdot \Gamma \boldsymbol{y} = 0.$$

We have introduced a new variable ϕ and if we choose all of our variables in these modified equations to be 0 then we have a satisfying witness. In the simulation, we give the simulator trapdoor information that permits it to open c to 0 and we can now use the NIWI proof from Section 6.

We will now describe the NIZK proof. The setup, common reference string generation, proof and verification work as a standard NIWI proof. Here we describe the simulator.

Simulation string: Using $\iota_1(1) = \iota_1(0) + \sum_{i=1}^{\widehat{m}} s_i u_i$ the simulation string is
$(\sigma, \tau) := ((B_1, B_2, B_T, F, \iota_1, p_1, \iota_2, p_2, \iota_T, p_T, \boldsymbol{u}, \boldsymbol{v}), \boldsymbol{s}, H_1, \ldots, H_\eta) \leftarrow S_1(gk)$.

Simulated proof: The input consists of gk, σ, a list of quadratic equations $\{(\boldsymbol{a}_i, \boldsymbol{b}_i, \Gamma_i, t_i)\}_{i=1}^N$ and a satisfying witness $\boldsymbol{x}, \boldsymbol{y}$.
Rewrite the equations as $\boldsymbol{a}_i \cdot \boldsymbol{y} + \boldsymbol{x} \cdot \boldsymbol{b}_i + f(\phi, -t_i) + \boldsymbol{x} \cdot \Gamma_i \boldsymbol{y} = 0$. Define $\boldsymbol{x} := \boldsymbol{0}, \boldsymbol{y} := \boldsymbol{0}$ and $\phi = 0$ to get a witness that satisfies all equations.
Pick at random $R \leftarrow \mathrm{Mat}_{m \times \widehat{m}}(\mathcal{R})$ and $S \leftarrow \mathrm{Mat}_{n \times \widehat{n}}(\mathcal{R})$ and commit to all the variables as $\boldsymbol{c} := \boldsymbol{0} + R\boldsymbol{u}$ and $\boldsymbol{d} := \boldsymbol{0} + S\boldsymbol{v}$. We also use $c := \iota_1(1) = \iota_1(0) + \sum_{i=1}^{\widehat{m}} s_i u_i$ and append it to $\boldsymbol{c}$.
For each modified equation $(\boldsymbol{a}_i, \boldsymbol{b}_i, -t_i, \Gamma_i, 0)$ make a proof as described in Section 5. Return the simulated proof $\{(\boldsymbol{c}, \boldsymbol{d}, \boldsymbol{\pi}_i, \boldsymbol{\theta}_i)\}_{i=1}^N$.

We prove in the full paper [28] that this construction gives us a perfect NIZK proof.

Theorem 5. *The NIWI proof from Section 6 with the simulator described above is a composable NIZK proof for satisfiability of pairing product equations with perfect completeness, soundness and composable zero-knowledge, when $A_1 = \mathcal{R}$ and the commitment in B_1 can be trapdoor opened.*

7.1 NIZK Proofs for Bilinear Groups

Let us return to the four types of quadratic equations given in Figure 1. If we set up the common reference string such that we can trapdoor open respectively $\iota_1'(1)$ and $\iota_2'(1)$ to $0 \in \mathbb{Z}_n$ then multi-scalar multiplication equations and quadratic equations in $\mathbb{Z}_{\mathbf{n}}$ are of the form for which we can give zero-knowledge proofs (at no additional cost).

In the case of pairing product equations we do not know how to get zero-knowledge, since even with the trapdoors we may not be able to compute a satisfiability witness. We do observe though that in the special case, where all $t_T = 1$ the choice of $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{O}}, \boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{O}}$ is a satisfactory witness. Since we also use $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{O}}, \boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{O}}$ in the other zero-knowledge proofs, the simulator can use this witness and give a NIWI proof. In the special case where all $t_T = 1$ we can therefore make NIZK proofs for satisfiability of the set of pairing product equations.

Next, let us look at the case where we have a pairing product equation with $t_T = \prod_{i=1}^n e(\mathcal{P}_i, \mathcal{Q}_i)$ for some known $\mathcal{P}_i, \mathcal{Q}_i$. In this case, we can add linear equations $\mathcal{Z}_i = \mathcal{P}_i$ to the set of multi-scalar multiplication equations in G_1. We already know that such equations have zero-knowledge proofs. We can now rewrite the pairing product equation as $(\boldsymbol{\mathcal{A}} \cdot \boldsymbol{\mathcal{Y}})(\boldsymbol{\mathcal{X}} \cdot \boldsymbol{\mathcal{B}})(\boldsymbol{\mathcal{Z}} \cdot \boldsymbol{\mathcal{Q}})(\boldsymbol{\mathcal{X}} \cdot \Gamma \boldsymbol{\mathcal{Y}}) = 1$. We can therefore also make zero-knowledge proofs if all the pairing product equations have t_T of the form $t_T = \prod_{i=1}^n e(\mathcal{P}_i, \mathcal{Q}_i)$ for some known $\mathcal{P}_i, \mathcal{Q}_i$.

The case of pairing product equations points to a couple of differences between witness-indistinguishable proofs and zero-knowledge proofs using our techniques. NIWI proofs can handle any target t_T, whereas zero-knowledge proofs can only handle special types of target t_T. Furthermore, if $t_T \neq 1$ the size of the NIWI proof for this equation is constant, whereas the NIZK proof for the same equation may be larger.

Acknowledgements

We gratefully acknowledge Brent Waters for a number of helpful ideas, comments, and conversations related to this work. In particular, our module-based approach can be

seen as formalizing part of the intuition expressed by Waters that the Decisional Linear Assumption, Subgroup Decision Assumption in composite-order groups, and SXDH can typically be exchanged for one another. (We were inspired by previously such connections made by [26, 36].) It would be interesting to see if this intuition can be made formal in other settings, such as Traitor Tracing [12] or Searchable Encryption [13]. We also thank Dan Boneh for his encouragement and for suggesting using our techniques to get fair exchange.

References

1. Lucas Ballard, Matthew Green, Breno de Medeiros, and Fabian Monrose. Correlation-resistant storage via keyword-searchable encryption. Cryptology ePrint Archive, Report 2005/417, 2005. Available at `http://eprint.iacr.org/2005/417`.
2. Paulo Barreto. The pairing-based crypto lounge, 2006. Available at `http://paginas.terra.com.br/informatica/paulobarreto/pblounge.html`.
3. Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. Non-interactive anonymous credentials. In *TCC*, Lecture Notes in Computer Science, 2008. Full paper available at `http://eprint.iacr.org/2007/384`.
4. Manuel Blum, Paul Feldman, and Silvio Micali. Non-interactive zero-knowledge and its applications. In *STOC*, pages 103–112, 1988.
5. Dan Boneh. Personal communication, 2006.
6. Dan Boneh and Xavier Boyen. Efficient selective-id secure identity-based encryption without random oracles. In *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238, 2004.
7. Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In *CRYPTO*, volume 3152 of *Lecture Notes in Computer Science*, pages 443–459, 2004.
8. Dan Boneh, Xavier Boyen, and Hovav Shacham. Short group signatures. In *CRYPTO*, volume 3152 of *Lecture Notes in Computer Science*, pages 41–55, 2004.
9. Dan Boneh, Giovanni Di Crescenzo, Rafail Ostrovsky, and Giuseppe Persiano. Public key encryption with keyword search. In *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 506–522, 2004.
10. Dan Boneh and Matthew K. Franklin. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.
11. Dan Boneh, Eu-Jin Goh, and Kobbi Nissim. Evaluating 2-DNF formulas on ciphertexts. In *TCC*, volume 3378 of *Lecture Notes in Computer Science*, pages 325–341, 2005.
12. Dan Boneh, Amit Sahai, and Brent Waters. Fully collusion resistant traitor tracing with short ciphertexts and private keys. In *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 573–592, 2006.
13. Xavier Boyen and Brent Waters. Compact group signatures without random oracles. In *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 427–444, 2006.
14. Xavier Boyen and Brent Waters. Full-domain subgroup hiding and constant-size group signatures. In *PKC*, volume 4450 of *Lecture Notes in Computer Science*, pages 1–15, 2007. Available at `http://www.cs.stanford.edu/~xb/pkc07/`.
15. Nishanth Chandran, Jens Groth, and Amit Sahai. Ring signatures of sub-linear size without random oracles. In *ICALP*, volume 4596 of *Lecture Notes in Computer Science*, pages 423–434, 2007.
16. Ivan Damgård. Non-interactive circuit based proofs and non-interactive perfect zero-knowledge with proprocessing. In *EUROCRYPT*, volume 658 of *Lecture Notes in Computer Science*, pages 341–355, 1992.

17. Alfredo De Santis, Giovanni Di Crescenzo, and Giuseppe Persiano. Randomness-optimal characterization of two NP proof systems. In *RANDOM*, volume 2483 of *Lecture Notes in Computer Science*, pages 179–193, 2002.
18. Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography. *SIAM Journal of Computing*, 30(2):391–437, 2000.
19. Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs under general assumptions. *SIAM Journal of Computing*, 29(1):1–28, 1999.
20. Steven D. Galbraith and Victor Rotger. Easy decision Diffie-Hellman groups. *London Mathematical Society Journal of Computation and Mathematics*, 7:201–218, 2004.
21. Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proofs. *SIAM Journal of Computing*, 18(1):186–208, 1989.
22. Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. Attribute-based encryption for fine-grained access control of encrypted data. In *ACM CCS*, pages 89–98, 2006.
23. Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In *ASIACRYPT*, volume 4248 of *Lecture Notes in Computer Science*, pages 444–459, 2006. Full paper available at `http://www.brics.dk/~jg/NIZKGroupSignFull.pdf`.
24. Jens Groth. Fully anonymous group signatures without random oracles. In *ASIACRYPT*, volume 4833 of *Lecture Notes in Computer Science*, pages 164–180, 2007. Full paper available at `http://www.brics.dk/~jg/CertiSignFull.pdf`.
25. Jens Groth and Steve Lu. A non-interactive shuffle with pairing based verifiability. In *ASIACRYPT*, volume 4833 of *Lecture Notes in Computer Science*, pages 51–67, 2007.
26. Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for NIZK. In *CRYPTO*, volume 4117 of *Lecture Notes in Computer Science*, pages 97–111, 2006.
27. Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero-knowledge for NP. In *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 339–358, 2006.
28. Jens Groth and Amit Sahai. Efficient non-interactive proof systems for bilinear groups. Cryptology ePrint Archive, Report 2007/155, 2007. Available at `http://eprint.iacr.org/2007/155`.
29. Joe Kilian and Erez Petrank. An efficient noninteractive zero-knowledge proof system for NP with general assumptions. *Journal of Cryptology*, 11(1):1–27, 1998.
30. Silvio Micali. Simple and fast optimistic protocols for fair electronic exchange. In *PODC*, pages 12–19, 2003.
31. Kenneth G. Paterson. Cryptography from pairings. In I.F. Blake, G. Seroussi, and N.P. Smart, editors, *Advances in Elliptic Curve Cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*, pages 215–251. Cambridge University Press, 2005.
32. Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473, 2005.
33. Mike Scott. Authenticated ID-based key exchange and remote log-in with simple token and PIN number. Cryptology ePrint Archive, Report 2002/164, 2002. Available at `http://eprint.iacr.org/2002/164`.
34. Eric R. Verheul. Evidence that XTR is more secure than supersingular elliptic curve cryptosystems. *Journal of Cryptology*, 17(4):277–296, 2004.
35. Brent Waters. Efficient identity-based encryption without random oracles. In *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 114–127, 2005.
36. Brent Waters. New techniques for slightly 2-homomorphic encryption, 2006. Manuscript.