Groth–Sahai Proofs Revisited

Essam Ghadafi, Nigel P. Smart, and Bogdan Warinschi

Dept. Computer Science,
University of Bristol,
Merchant Venturers Building,
Woodland Road,
Bristol, BS8 1UB,
United Kingdom
{ghadafi,nigel,bogdan}@cs.bris.ac.uk

Abstract. Since their introduction in 2008, the non-interactive zero-knowledge (NIZK) and non-interactive witness indistinguishable (NIWI) proofs designed by Groth and Sahai have been used in numerous applications. In this paper, we offer two contributions to the study of these proof systems. First, we identify and correct some errors, present in the oringal online manuscript, that occur in two of the three instantiations of the Groth-Sahai NIWI proofs for which the equation checked by the verifier is not valid for honest executions of the protocol. In particular, implementations of these proofs would not work correctly. We explain why, perhaps surprisingly, the NIZK proofs that are built from these NIWI proofs do not suffer from a similar problem. Secondly, we study the efficiency of existing instantiations and note that only one of the three instantiations has the potential of being practical. We therefore propose a natural extension of an existing assumption from symmetric pairings to asymmetric ones which in turn enables Groth-Sahai proofs based on new classes of efficient pairings.

1 Introduction

Background. Interactive proofs allow a prover who possesses some witness ω to convince a verifier that a certain statement $x \in L$ is true, where L is some language and ω is a witness that attests to this fact. A particularly fascinating class of interactive proofs are those where the interaction does not reveal information about the witness, even if the verifier behaves maliciously. Two popular flavors of witness privacy are witness-indistinguishability [14], when it is unfeasible for an adversary to decide which of the possible witnesses is used by the prover, and zero-knowledge[19,20], when it is possible to simulate the interaction between the prover and the verifier without access to a witness. The two notions share many commonalities, but are also different in important respects and suitable for different applications. For example, WI proofs can be executed in parallel while preserving the privacy of the witness, while ZK proofs may fail in this scenario.

P.Q. Nguyen and D. Pointcheval (Eds.): PKC 2010, LNCS 6056, pp. 177–192, 2010.

A variant of zero-knowledge proofs useful in multiple application scenarios are the non-interactive ones [6] (NIZK). In such proofs the interaction between the prover and the verifier is minimal: the prover simply sends the verifier a single message after which the latter verifies correctness of the proof without any further interaction with the prover. It is not difficult to see that NIZK proofs are impossible in the plain model [18], so some additional setup assumptions are required. Originally, such proofs were constructed in a setting where parties share a common random string (CRS) [15]. Later, non-interactive protocols were also constructed by eliminating interaction through the use of random oracles [5].

Unsurprisingly, both zero-knowledge and witness-indistinguishable proofs have found countless applications in cryptography. The power and versatility of such proofs is based on general results that show how to construct zero-knowledge proof systems for any language in NP [21]. For example, with zero-knowledge proofs, a party can prove that he/she is following a certain protocol, without revealing any information about its internal state, and thus can be used to compile protocols secure for honest-but-curious adversaries into protocols secure against arbitrary adversaries. Witness indistinguishable proofs can be used, for instance, in the Yao garbled-circuit protocol, to show that public commitments are commitments to elements in $\{0, 1\}$. The usability of proofs is tightly tied to the class of languages to which they apply, and to the efficiency of the associated proof systems. Clearly, these two requirements are contradictory. Indeed, the approach of [21] is quite general, but the combination of general NP-reductions to problems along with ZK protocols leads to highly impractical protocols even for the simplest languages.

A crucial step towards more efficient non-interactive zero-knowledge proofs was the breakthrough work of Groth and Sahai [25]. The authors show how to give NIWI and NIZK proofs for a large class of languages, without going through the use of a general NP reduction. Numerous cryptographic results use GS proofs to obtain efficient implementations for various primitives, see the related work section for a very partial list of such works. In this paper, we contribute to the understanding of these proofs in two different ways. We extend the range of implementations to new, potentially more efficient settings and we fix an inconspicuous flaw that affects an important part of the original online manuscript [26]. To explain our contributions, we recall some details of the settings used by [25].

In the original (conference) version of the Groth–Sahai paper [25], the authors give a general, abstract framework for the construction of NIWI/NIZK proofs based on cryptographic pairings. We note that none of the errors we identify occur in [25]. Proofs and details for three different instantiations are given in the e-print archive version of the paper [26]. The first instantiation uses pairings over groups of large composite order; the other two use pairings over prime order groups. The cryptographic assumptions on which the results rely are: the subgroup decisional problem [8] in the first case, the decisional linear assumption (DLIN) [7], and the symmetric external Diffie–Hellman assumption (SXDH) [1], for the remaining two instantiations, respectively. To obtain the later instantiations the authors essentially use a general procedure [16] of converting protocols

from the subgroup decision setting for composite order pairing groups, into protocols for the DLIN and SXDH assumptions in prime order pairing groups.

Efficient implementations based on a new assumption. From a practical perspective, pairings for groups of composite order are likely to have little practical impact, due to their inherent inefficiency. The same holds true for symmetric pairings, i.e. Type-1 pairings in the vocabulary of [17], which are the pairings used in the second instantiation. Therefore, the only practical instantiation proposed in [26] remains the one based on SXDH in Type-3 curves. In this paper, we propose new GS proofs which can be used with the most efficient curves for pairing based cryptography. Our proposals are based on a natural extension of the DLIN assumption from the symmetric setting to the asymmetric one. We thus give DLIN-based GS proofs that work for all of the asymmetric pairing types. In particular, our proofs are the first GS proofs that work for Type-2 pairings.

We wish to warn readers against judging the efficiency of the proof systems based on Type-1 curves versus those based on Type-2 and Type-3 solely based on the number of group elements needed. The efficiency of the former curves is only illusory since the key sizes for these curves grow faster, and the benefits are immediately lost. Also, we note that the relative merits of the SXDH assumption versus the DLIN assumption are a matter of debate in pairing based cryptography; some people prefer the DLIN assumption as it applies to both symmetric and asymmetric settings, although the latter is never formally stated and we need to formalise the underlying hard problem in this paper. On the other hand the SXDH assumption only applies to Type 3 pairings, which produce the most efficient pairings known. The SXDH assumption also usually results in cleaner and simpler protocol, with Groth–Sahai proofs being no exception. In addition the SXDH assumption is more closely related to a long standing natural number theoretic problem, i.e. decisional Diffie–Hellman, than the DLIN assumption.

Fixing the inconspicuous flaw. The construction of Groth–Sahai NIZK proofs in [25,26] is done in two stages. First, the authors show how to construct NIWI proofs, and then following a trick they turn these proofs into full zero-knowledge ones. Unfortunately, the NIWI proofs based on DLIN and SXDH presented in [26] are actually invalid: the verification equation is not always satisfied when the execution is between honest provers and verifiers. As such, these proofs do not apply for many rather simple but quite useful statements. The details are somewhat technical and we explain this point later in the paper. These errors were introduced during the translation from the construction based on the subgroup decisional problem to the DLIN and SXDH settings [27]. Interestingly, this problem does not affect the construction of NIZK proofs out of NIWI proofs, since in this case the verification equation is always satisfied! Again, we elaborate on this point later in the paper.

We believe that the reason why this error had not been discovered so far is two-fold. On the one hand, as explained above, GS NIZK proofs are actually correct. On the other hand, when used in applications, GS (NIWI) proofs are usually

treated in a black-box way: the actual proofs are never spelled out, and the associated equations are never verified. Clearly, the problem would immediately show up in an implementation. We fix these problems by giving the correct versions of the proofs.

Finally, we note that in an effort to encourage further study of the Groth-Sahai proofs we depart from the notation in the original paper and use some notation that we believe is more expressive and easier to follow.

Related work. Despite their recent introduction, Groth–Sahai proofs have been widely used. Since Groth–Sahai proofs apply to bilinear groups, they are mainly used to design cryptographic primitives that do not rely on the random oracle assumption. The proofs are used to prove a knowledge of some secret witnesses or as a proof of membership. The scenarios in which the Groth–Sahai proofs are used in the literature include: proving the possession of some signature without actually revealing the signature, proving that two ciphertexts encrypt the same message, etc. For instance, they were used by Camenisch et al. [10] to build an encryption scheme that is KDM-CCA2 secure. Also, the NIWI and NIZK proofs were used by Belenkiy et al.[2,3] to design p-signatures and anonymous credentials. Groth and Lu[24] used the NIZK proofs to prove the correctness of a shuffle. Huang et al. [28] used Groth–Sahai NIWI and NIZK proofs to construct optimistic fair exchange protocol. In [31], Phong et al. used the NIZK proofs to construct undeniable signatures. Belenkiy et al. in[4] have extensively used both the NIWI and NIZK proofs to construct many cryptographic primitives such as p-signatures, verifiable random functions and compact e-cash system. Groth–Sahai proofs have also been used to construct group-signatures [23,30]. In [13,22] the proofs are used to design universally composable oblivious transfer protocols. The first of these is particularly interesting from our perspective; in [13] the authors use a NIWI proof to prove that set of linear equations holds. When this protocol is instantiated with the DLIN or SXDH protocols from [26] one would not obtain a proof which verifies. This is an example of an instance where the verification equations of the GS NIWI proofs are not valid.

Many of the previous applications of Groth–Sahai proofs for prime order groups, are assumed to be in the (inefficient) symmetric pairing setting, as they wish to use protocols based on the DLIN assumption; or they are in the asymmetric setting and need to make a DLIN assumption related to their scheme and then an additional SXDH assumption to apply Groth–Sahai proofs. By extending the DLIN setting to both Type-2 and Type-3 pairings we hope to simplify future applications of Groth–Sahai proofs, in addition by providing a mechanism for implementing Groth–Sahai proofs in the Type-2 setting other applications may open up.

2 Bilinear Groups

Bilinear groups are a set of three groups $\mathbb{G}_1$,$\mathbb{G}_2$ and G_T of prime order q along with a bilinear map (deterministic function) $\hat{t}$ which takes as input an element in $\mathbb{G}_1$ and an element in $\mathbb{G}_2$ and outputs an element in $\mathbb{G}_T$. We shall write $\mathbb{G}_1$

and $\mathbb{G}_2$ additively, and $\mathbb{G}_T$ multiplicatively, and write $\mathbb{G}_1 = \langle P_1 \rangle, \mathbb{G}_2 = \langle P_2 \rangle$, for two explicitly given generators P_1 and P_2.

The function $\hat{t}$ must have the following three properties:

1. Bilinearity: $\forall Q_1 \in \mathbb{G}_1$, $Q_2 \in \mathbb{G}_2$ $x, y \in \mathbb{Z}$, we have
$$\hat{t}([x]Q_1, [y]Q_2) = \hat{t}(Q_1, Q_2)^{xy}.$$
2. Non-Degeneracy: The value $\hat{t}(P_1, P_2)$ generates $\mathbb{G}_T$.
3. The function $\hat{t}$ is efficiently computable.

In [17], pairings were categorized into three Types:

- **Type-1**: This is the symmetric pairing setting in which $\mathbb{G}_1 = \mathbb{G}_2$.
- **Type-2**: Here we have $\mathbb{G}_1 \neq \mathbb{G}_2$, but there is an efficiently computable isomorphism $\psi : \mathbb{G}_2 \longrightarrow \mathbb{G}_1$ where $\psi(P_2) = P_1$.
- **Type-3**: Again $\mathbb{G}_1 \neq \mathbb{G}_2$, but now there is no known efficiently computable isomorphism.

In the Type-1 setting the decision Diffie–Hellman problem is easy in $\mathbb{G}_1$, and hence in $\mathbb{G}_2$. In the Type-2 setting the decision Diffie–Hellman problem is easy in $\mathbb{G}_2$, but believed to be hard in $\mathbb{G}_1$. In the Type-3 setting the decision Diffie–Hellman problem is believed to be hard in both $\mathbb{G}_1$ and $\mathbb{G}_2$. This last belief is often formalised as the symmetric external Diffie–Hellman assumption:

Definition 1. Symmetric External Diffie-Hellman (SXDH) Assumption: *In Type-3 pairings the Decisional Diffie-Hellman (DDH) problem is hard in both groups* $\mathbb{G}_1$ *and* $\mathbb{G}_2$.

As a note on naming, the "external" part relates to the fact we are talking about DDH in $\mathbb{G}_1$ and $\mathbb{G}_2$, as opposed to the pairing based BDDH problem. The "symmetric" part is related to the fact that we are talking about DDH being hard in both $\mathbb{G}_1$ and $\mathbb{G}_2$. It is perhaps unfortunate terminology that this symmetry only applies in the asymmetric pairing setting!

As the SXDH problem only applies to Type-3 pairings, it is common to make the following assumption for Type-1 pairings, as a natural strengthening of the normal DDH assumption, which no longer applies in Type-1 pairings:

Definition 2. Decisional Linear Problem (DLIN) Assumption: *For Type-1 pairings with* $\mathbb{G}_1 = \mathbb{G}_2 = \mathbb{G}$ *and* $P = P_1 = P_2$*, given the tuple* $([a]P, [b]P, [ra]P, [sb]P, [t]P)$ *where* $a, b, r, s, t \in \mathbb{F}_q$ *are unknowns, it is hard to tell whether* $t = r + s$ *or* t *is random.*

To extend this definition to the Type-2 or Type-3 setting one could insist that DLIN is hard in either $\mathbb{G}_1$ or $\mathbb{G}_2$, however we will require that it is hard in *both* $\mathbb{G}_1$ and $\mathbb{G}_2$. We call this latter notion, in following the naming of the SXDH assumption, as the symmetric DLIN (SDLIN) assumption.

Definition 3. Symmetric Decisional Linear Problem (SDLIN) Assumption: *SDLIN is said to hold if DLIN is hard in both* $\mathbb{G}_1$ *and* $\mathbb{G}_2$.

This is a stronger form of a version of the asymmetric DLIN problem considered in other works such as [22], where a single problem with some variable instances in $\mathbb{G}_1$ and some in $\mathbb{G}_2$ is considered.

We end this section by noting that in [9], Boneh et al. showed that the existence of the isomorphism in the Type-2 setting can affect the security of some cryptographic primitives. On the other hand, Chatterjee and Menezes [11] show that a protocol which is secure in Type-2 setting can almost always be transfered to one which is secure in Type-3 setting.

3 Groth–Sahai Proofs

In [25,26] Groth and Sahai presented a way to construct efficient non-interactive witness-indistinguishable and zero-knowledge proofs for a wide variety of statements in the common reference string model. In this section, we recap on their notation, and point out the problems with their presentation.

The NIZK proof systems allow the same methodology to be applied to four distinct types of equations, or three distinct types in the case of Type-1 pairings. In this section the four different types are presented in one go using the abstraction of Groth-Sahai. Later we present the specialisations to the different settings.

Let q be the order of $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ as above. We first create $\mathbb{F}_q$-vector spaces $\mathbb{A}_1, \mathbb{A}_2, \mathbb{A}_T, \mathbb{B}_1, \mathbb{B}_2$ and $\mathbb{B}_T$. In [26] these are $\mathbb{Z}_n$-modules and not $\mathbb{F}_q$-vector spaces since n may be composite, in our situations we always have $n = q$, a prime. We assume these vector spaces are equiped with bilinear maps $f : \mathbb{A}_1 \times \mathbb{A}_2 \to \mathbb{A}_T$ and $F : \mathbb{B}_1 \times \mathbb{B}_2 \to \mathbb{B}_T$. In addition, there are inclusion and projection maps for each pair, i.e. we have maps $\iota_1 : \mathbb{A}_1 \to \mathbb{B}_1$, $\iota_2 : \mathbb{A}_2 \to \mathbb{B}_2$, $\iota_T : \mathbb{A}_T \to \mathbb{B}_T$, and $p_1 : \mathbb{B}_1 \to \mathbb{A}_1$, $p_2 : \mathbb{B}_2 \to \mathbb{A}_2$, $p_T : \mathbb{B}_T \to \mathbb{A}_T$. Note, that the ι maps are required to be computable, but that the p maps will not be computable in general. The maps are extended to vectors of elements in a componentwise fashion.

All these maps need to satisfy the following commutative properties:

$$\forall x \in \mathbb{A}_1, \forall y \in \mathbb{A}_2 : F(\iota_1(x), \iota_2(y)) = \iota_T(f(x, y)),$$
$$\forall \mathcal{X} \in \mathbb{B}_1, \forall \mathcal{Y} \in \mathbb{B}_2 : f(p_1(\mathcal{X}), p_2(\mathcal{Y})) = p_T(F(\mathcal{X}, \mathcal{Y})).$$

The essential problem in the DLIN and SXDH settings from [26] is that the specific values of these maps, for three of the four equation types, do not result in the first of these commutative properties holding. In particular the given presentation of ι_T is incorrect. This leads to the resulting verification of the NIWI proofs being invalid.

The CRS we use in our proofs is a set of $\hat{m}_1$ and $\hat{m}_2$ elements of $\mathbb{B}_1$ and $\mathbb{B}_2$, which we will denote by $\mathcal{U}_1^{(1)}, \ldots, \mathcal{U}_1^{(\hat{m}_1)} \in \mathbb{B}_1$ and $\mathcal{U}_2^{(1)}, \ldots, \mathcal{U}_2^{(\hat{m}_2)} \in \mathbb{B}_2$. To commit to an element $x \in \mathbb{A}_i$ one picks $\underline{r} = (r_1, \ldots, r_{\hat{m}_i}) \in \mathbb{F}_q^{\hat{m}_i}$ and computes

$$\begin{aligned} \mathrm{comm_i}(x) &= \iota_i(x) + \sum_{j=1}^{\hat{m}_i} [r_j] \mathcal{U}_i^{(j)} \\ &= \iota_i(x) + \underline{r} \cdot \underline{\mathcal{U}_i}. \end{aligned}$$

Now suppose we wish to produce a NIWI proof for the equation,

$$\underline{a} \otimes \underline{y} + \underline{x} \otimes \underline{b} + \underline{x} \otimes \Gamma \underline{y} = t, \tag{1}$$

where we use the shorthand $x \otimes y$ for $f(x, y)$, with an obvious extension to vectors. In the above equation; $\underline{x} \in \mathbb{A}_1^n$, $\underline{y} \in \mathbb{A}_2^m$ are the secret witnesses, with $\underline{a} \in \mathbb{A}_1^m$, $\underline{b} \in \mathbb{A}_2^n$, $\Gamma \in \mathrm{Mat}_{n \times m}(\mathbb{F}_q)$, and $t \in \mathbb{A}_T$ the known constants.

We commit to $\underline{x}$ and $\underline{y}$ using the random values given by $R \in \mathrm{Mat}_{n \times \hat{m}_1}(\mathbb{F}_q)$ and $S \in \mathrm{Mat}_{m \times \hat{m}_2}(\mathbb{F}_q)$ via

$$\underline{c} = \iota_1(\underline{x}) + R\, \underline{\mathcal{U}_1} \text{ and } \underline{d} = \iota_2(\underline{y}) + S\, \underline{\mathcal{U}_2}.$$

The NIWI proof is then given by the following two vector values; one picks $T \in \mathrm{Mat}_{\hat{m}_2 \times \hat{m}_1}(\mathbb{F}_q)$ uniformly at random and computes

$$\underline{\pi} = R^{\mathrm{T}} \iota_2(\underline{b}) + R^{\mathrm{T}} \Gamma \iota_2(\underline{y}) + R^{\mathrm{T}} \Gamma S\, \underline{\mathcal{U}_2} - T^{\mathrm{T}} \underline{\mathcal{U}_2} \in \mathbb{B}_2^{\hat{m}_1},$$
$$\underline{\theta} = S^{\mathrm{T}} \iota_1(\underline{a}) + S^{\mathrm{T}} \Gamma^{\mathrm{T}} \iota_1(\underline{x}) + T\, \underline{\mathcal{U}_1} \in \mathbb{B}_1^{\hat{m}_2}.$$

Verification of the proof $(\underline{\pi}, \underline{\theta})$ is performed by checking whether

$$\iota_1(\underline{a}) \bullet \underline{d} + \underline{c} \bullet \iota_2(\underline{b}) + \underline{c} \bullet \Gamma \underline{d} = \iota_T(t) + \underline{\mathcal{U}_1} \bullet \underline{\pi} + \underline{\theta} \bullet \underline{\mathcal{U}_2}$$

holds. Here we use $\mathcal{X} \bullet \mathcal{Y}$ as a shorthand for $F(\mathcal{X}, \mathcal{Y})$, again with an obvious extension for vectors.

Notes. There are four possible instantiations of the equations:

- $\mathbb{A}_1 = \mathbb{G}_1$, $\mathbb{A}_2 = \mathbb{G}_2$, $f(P, Q) = \hat{t}(P, Q)$: This case is called *pairing product equations*.
- $\mathbb{A}_1 = \mathbb{G}_1$, $\mathbb{A}_2 = \mathbb{F}_q$, $f(P, y) = [y]P$: This case is called *multi-scalar multiplication in* $\mathbb{G}_1$.
- $\mathbb{A}_1 = \mathbb{F}_q$, $\mathbb{A}_2 = \mathbb{G}_2$, $f(x, Q) = [x]Q$: This case is called *multi-scalar multiplication in* $\mathbb{G}_2$.
- $\mathbb{A}_1 = \mathbb{F}_q$, $\mathbb{A}_2 = \mathbb{F}_q$, $f(x, y) = x \cdot y$: This case is called *quadratic equation in* $\mathbb{F}_q$.

In the DLIN and SXDH cases, the formulaes for ι_T for the last three types of equations are given incorrectly in [26]. ¿From examining the above methods for NIWI proofs, we see that the NIWI proofs would not verify, unless the value t was the trivial element.

We note that in the simpler, yet very common, setting of having $\Gamma = 0$ and either $\underline{a} = \underline{0}$ or $\underline{b} = \underline{0}$ in equation (1), the proofs can be simplified further by setting the random matrix T to be zero.

The CRS, and hence the commitment scheme used to commit to elements in $\mathbb{A}_1$ and $\mathbb{A}_2$, comes in two flavours: either we have a *binding key*, or a *hiding key*.

- **Binding key:** This setting requires that for $i = 1$ and $i = 2$, $p_i(\iota_i(x)) = x$ and $p_i(\mathcal{U}_i^{(j)}) = 0$ for all j. Hence we have $p_i(\mathrm{comm_i}(x)) = x$ which gives us a perfectly binding, computationally hiding commitment scheme. When used in the proof, this results in perfectly-sound proofs with computational witness indistinguishablity.

– **Hiding key:** This setting requires that $\{\mathcal{U}_i^{(1)}, \ldots, \mathcal{U}_i^{(\hat{m}_i)}\}$, i.e. the set of commitment keys, generate the entire space $\mathbb{B}_i$, and hence we have $\iota_i(\mathbb{A}_i) \subseteq \langle \mathcal{U}_i^{(1)}, \ldots, \mathcal{U}_i^{(\hat{m}_i)} \rangle$. Therefore, if the randomness vector, $\underline{r}$, is uniformly chosen, the commitment scheme is computationally binding and perfectly hiding. If this setting is used, the resulting proofs are computationally sound and perfectly witness-indistinguishable.

The security of the whole system is ensured as long as the adversary is unable to distinguish between a hiding and a binding key. The security proofs can be found in [26]. When producing a real system, one relies on a trusted third party to produce a binding key, however when producing a simulated proof etc. one relies on a hiding key, which essentially provides a trapdoor for the simulator in the CRS model.

For the DLIN assumption in the Type-1 setting in [26], a method is given to make the map F symmetric, in the sense that $F(\mathcal{X}, \mathcal{Y}) = F(\mathcal{Y}, \mathcal{X})$. We shall see when F is instantiated below, that such a symmetry is not possible for Type-2 and Type-3 pairings. When F is symmetric the associated proofs can be made much simpler, we leave the reader to consult [26] for details.

To convert the above method for NIWI proofs into a method for NIZK proofs, we first reorganize the above equation as

$$\underline{a} \otimes \underline{y} + (-1 \otimes t) + \underline{x} \otimes \underline{b} + \underline{x} \otimes \Gamma \underline{y} = \begin{cases} 0 & \text{If } \mathbb{A}_T = \mathbb{F}_q, \\ \mathcal{O} & \text{If } \mathbb{A}_T = \mathbb{G}_1 \text{ or } \mathbb{G}_2, \\ 1 & \text{If } \mathbb{A}_T = \mathbb{G}_T. \end{cases}$$

The vector of commitments $\underline{c}$ is extended to include a commitment to the element one, this is done to deal with the extra term in the left hand side of the above equation. Then the above NIWI method is applied. This results in the NIZK proofs in the pairing product equation subcase only applying when either $t = 1$ in equation (1), or one knows $P_1, \ldots, P_n$ and $Q_1, \ldots, Q_n$ such that $t = \hat{t}(P_1, Q_1) \cdots \hat{t}(P_n, Q_n)$, since only then can the above transform be applied. This is the only restriction in the method for obtaining NIZK proofs.

In all cases, to obtain NIZK proofs we apply the method for NIWI proofs in the case where the equation is homogeneous, i.e. has a trivial right hand side. This latter point is crucial in understanding why the NIZK proofs from [26] work but the NIWI proofs do not. Hence, even though ι_T was presented incorrectly in [26], since the method to produce NIZK proofs will result in a trivial value of ι_T, the method for NIZK is sound.

4 Equations for ι and p

From the last section, it is seen that the whole system depends on the choice of the ι and p maps, plus the CRS. The maps must be chosen so that they have the required commutativity property over f and F. In this section, we give such maps and the relevant CRS for the SXDH and SDLIN examples in the asymmetric pairing setting.

We present the data in the following way, for each setting we first present the hiding and binding CRS, along with the map F and the groups $\mathbb{B}_i$ and $\mathbb{B}_T$. Then we present the maps ι_i and p_i for the cases $\mathbb{A}_i = \mathbb{F}_q$ and $\mathbb{A}_i = \mathbb{G}_i$. At this point we overload the symbols ι_i and p_i, with the precise maps being obtained by type-checking. This helps simplify our notation somewhat.

Once the maps are defined we can proceed to produce the commitment schemes, and the NIWI and NIZK proofs. Then for the four types of equation being proved, we present the maps ι_T and p_T, which result in the maps being commutative. With these maps one can then verify the resulting NIWI proofs. Again we overload ι_T and p_T, with the precise map being determined by type checking.

4.1 SXDH-Based Proofs

Setup. We set $\mathbb{B}_1 = \mathbb{G}_1^2$, $\mathbb{B}_2 = \mathbb{G}_2^2$ and $\mathbb{B}_T = \mathbb{G}_T^4$, all with operations performed componentwise. We let

$$F : \begin{cases} \mathbb{B}_1 \times \mathbb{B}_2 & \longrightarrow \mathbb{B}_T \\ (X_1, Y_1), (X_2, Y_2) & \longmapsto \left(\hat{t}(X_1, X_2),\ \hat{t}(X_1, Y_2),\ \hat{t}(Y_1, X_2),\ \hat{t}(Y_1, Y_2) \right) \end{cases}$$

Since the underlying pairing $\hat{t}$ is bilinear, it follows that the map F is also bilinear. To generate the CRS, the trusted party generates, for $i = 1, 2$, $a_i, t_i \in \mathbb{F}_q^*$ at random and defines

$$Q_i = [a_i]P_i, \quad U_i = [t_i]P_i, \quad V_i = [t_i]Q_i.$$

We now set

$$\mathcal{U}_i^{(1)} = (P_i, Q_i) \in \mathbb{B}_i,$$
$$\mathcal{U}_i^{(2)} = \begin{cases} [t_i]\mathcal{U}_i^{(1)} = (U_i, V_i) & \text{Binding Case} \\ [t_i]\mathcal{U}_i^{(1)} - (\mathcal{O}, P_i) = (U_i, V_i - P_i) & \text{Hiding Case} \end{cases} \in \mathbb{B}_i.$$

The CRS is then the set $\{\mathcal{U}_1, \mathcal{U}_2\}$ where $\mathcal{U}_1 = \{\mathcal{U}_1^{(1)}, \mathcal{U}_1^{(2)}\}$, and $\mathcal{U}_2 = \{\mathcal{U}_2^{(1)}, \mathcal{U}_2^{(2)}\}$. Under the SXDH assumption one cannot tell a binding key from a hiding key. To aid what follows, we first set $\mathcal{W}_i = \mathcal{U}_i^{(2)} + (\mathcal{O}, P_i) = (W_{i,1}, W_{i,2}) \in \mathbb{B}_i$.

ι_i, p_i and comm$_i$. We now define the maps $\iota_i : \mathbb{A}_i \to \mathbb{B}_i$, $p_i : \mathbb{B}_i \to \mathbb{A}_i$ and the commitment scheme comm$_i$. There are two cases we need to consider; $\mathbb{A}_i = \mathbb{F}_q$ and $\mathbb{A}_i = \mathbb{G}_i$.

$\underline{A_i = \mathbb{F}_q}$: We define, in this case, the maps via

$$\iota_i : \begin{cases} \mathbb{F}_q & \longrightarrow \mathbb{B}_i \\ x & \longmapsto [x]\mathcal{W}_i \end{cases} \qquad p_i : \begin{cases} \mathbb{B}_i & \longrightarrow \mathbb{F}_q \\ \mathcal{X} = ([c_1]P_i, [c_2]P_i) & \longmapsto c_2 - a_i c_1 \end{cases}$$

Note, that computing p_i requires one to solve discrete logarithms. This is not an issue since we at no point will compute p_i, we simply need to know it exists and it has the correct properties.

The commitment scheme comm_i is obtained as before, except we select $\hat{m}_i = 1$, as opposed to $\hat{m}_i = 2$, this simplifies the equations somewhat. Hence we have

$$\text{comm}_\text{i} : \begin{cases} \mathbb{F}_q \times \mathbb{F}_q \longrightarrow \mathbb{B}_i \\ (x, r) \longmapsto \iota_i(x) + [r]\mathcal{U}_i^{(1)} \end{cases}$$

$\underline{A_i = \mathbb{G}_i}$: In this case we define

$$\iota_i : \begin{cases} \mathbb{G}_i \longrightarrow \mathbb{B}_i \\ X \longmapsto (\mathcal{O}, X) \end{cases} \qquad p_i : \begin{cases} \mathbb{B}_i \longrightarrow \mathbb{G}_i \\ \mathcal{X} = (C_1, C_2) \longmapsto C_2 - [a_i]C_1 \end{cases}$$

The commitment scheme comm_i is obtained as in our main discussion, i.e. with $\hat{m}_i = 2$. Hence we have

$$\text{comm}_\text{i} : \begin{cases} \mathbb{G}_i \times \mathbb{F}_q \times \mathbb{F}_q \longrightarrow \mathbb{B}_i \\ (X, r_1, r_2) \longmapsto \iota_i(X) + [r_1]\mathcal{U}_i^{(1)} + [r_2]\mathcal{U}_i^{(2)} \end{cases}$$

***ι_T* and *p_T*.** Here we have four cases, depending on which of the four types of equation we are dealing with

Pairing Product Equations.

$$\iota_T : \begin{cases} \mathbb{G}_T \longrightarrow \mathbb{B}_T \\ \zeta \longmapsto (1, 1, 1, \zeta) \end{cases} \quad p_T : \begin{cases} \mathbb{B}_T \longrightarrow \mathbb{G}_T \\ (\zeta_{1,1}, \zeta_{1,2}, \zeta_{2,1}, \zeta_{2,2}) \longmapsto \zeta_{2,2}\zeta_{1,2}^{-a_1}(\zeta_{2,1}\zeta_{1,1}^{-a_1})^{-a_2} \end{cases}$$

Multi-Scalar Multiplication in $\mathbb{G}_1$ and $\mathbb{G}_2$.

In both of these cases we have

$$p_T : \begin{cases} \mathbb{B}_T \longrightarrow \mathbb{G}_i \\ (\zeta^{s_1}, \zeta^{s_2}, \zeta^{s_3}, \zeta^{s_4}) \longmapsto [s_4 - a_1 s_2 - a_2 s_3 + a_1 a_2 s_1]P_i \end{cases}$$

where $\zeta = \hat{t}(P_1, P_2)$. For multi-scalar multiplication in $\mathbb{G}_1$ the map ι_T is defined by

$$\iota_T : \begin{cases} \mathbb{G}_1 \longrightarrow \mathbb{B}_T \\ X \longmapsto (1, 1, \hat{t}(X, W_{2,1}), \hat{t}(X, W_{2,2})) \end{cases}$$

Whilst for multi-scalar multiplication in $\mathbb{G}_2$ the map ι_T is defined by

$$\iota_T : \begin{cases} \mathbb{G}_2 \longrightarrow \mathbb{B}_T \\ X \longmapsto (1, \hat{t}(W_{1,1}, X), 1, \hat{t}(W_{1,2}, X)). \end{cases}$$

Note, these are different definitions from those given in [26]. The above definitions produce the required commutative properties.

Quadratic Equations in $\mathbb{F}_q$.

In this case we have

$$p_T : \begin{cases} \mathbb{B}_T \longrightarrow \mathbb{F}_q \\ (\zeta^{s_1}, \zeta^{s_2}, \zeta^{s_3}, \zeta^{s_4}) \longmapsto s_4 - a_1 s_2 - a_2 s_3 + a_1 a_2 s_1 \end{cases}$$

where $\zeta = \hat{t}(P_1, P_2)$. The function ι_T is given by

$$\iota_T(z) : \begin{cases} \mathbb{F}_q \longrightarrow \mathbb{B}_T \\ z \longmapsto F(\mathcal{W}_1, \mathcal{W}_2)^z. \end{cases}$$

Again this is different from the map given in [26].

4.2 SDLIN-Based Proofs

We now perform a similar analysis when we wish to base security on the SDLIN problem. Recall in [26] this situation is only described for the Type-1 pairing situation. What we describe below can be used in both the Type-2 and Type-3 situations. In addition by specialising it to the Type-1 situation, and applying the optimization of [26], to produce a symmetric version of $F(\mathcal{X}, \mathcal{Y})$, one obtains more efficient NIZK proofs for Type-1 pairings as well.

Setup. We set $\mathbb{B}_1 = \mathbb{G}_1^3$, $\mathbb{B}_2 = \mathbb{G}_2^3$ and $\mathbb{B}_T = \mathbb{G}_T^9$, all with operations performed componentwise. We let

$$F : \begin{cases} \mathbb{B}_1 \times \mathbb{B}_2 & \longrightarrow & \mathbb{B}_T \\ (X_1, Y_1, Z_1), (X_2, Y_2, Z_2) & \longmapsto & \begin{pmatrix} \hat{t}(X_1, X_2) & \hat{t}(X_1, Y_2) & \hat{t}(X_1, Z_2) \\ \hat{t}(Y_1, X_2) & \hat{t}(Y_1, Y_2) & \hat{t}(Y_1, Z_2) \\ \hat{t}(Z_1, X_2) & \hat{t}(Z_1, Y_2) & \hat{t}(Z_1, Z_2) \end{pmatrix} \end{cases}$$

Since the underlying pairing $\hat{t}$ is bilinear, it follows that the map F is also bilinear. To generate the CRS the trusted party generates, for $i = 1, 2$ $a_i, r_i, s_i, t_i \in \mathbb{F}_q^*$ at random and defines

$$U_i = [a_i]P_i, \quad V_i = [t_i]P_i.$$

We now set

$$\begin{aligned} \mathcal{U}_i^{(1)} &= (U_i, \mathcal{O}, P_i) \in \mathbb{B}_i, \\ \mathcal{U}_i^{(2)} &= (\mathcal{O}, V_i, P_i) \in \mathbb{B}_i, \\ \mathcal{U}_i^{(3)} &= \begin{cases} [r_i]\mathcal{U}_i^{(1)} + [s_i]\mathcal{U}_i^{(2)} & \\ \quad = ([r_i]U_i, [s_i]V_i, [r_i + s_i]P_i) & \text{Binding Case} \\ [r_i]\mathcal{U}_i^{(1)} + [s_i]\mathcal{U}_i^{(2)} - (\mathcal{O}, \mathcal{O}, P_i) & \\ \quad = ([r_i]U_i, [s_i]V_i, [r_i + s_i - 1]P_i) & \text{Hiding Case} \end{cases} \end{aligned}$$

The CRS is then the set $\{\mathcal{U}_1, \mathcal{U}_2\}$ where $\mathcal{U}_1 = \{\mathcal{U}_1^{(1)}, \mathcal{U}_1^{(2)}, \mathcal{U}_1^{(3)}\}$, and $\mathcal{U}_2 = \{\mathcal{U}_2^{(1)}, \mathcal{U}_2^{(2)}, \mathcal{U}_2^{(3)}\}$. Under the SDLIN assumption one cannot tell a binding key from a hiding key. To aid notation in what follows, we first set $\mathcal{W}_i = \mathcal{U}_i^{(3)} + (\mathcal{O}, \mathcal{O}, P_i) = (W_{i,1}, W_{i,2}, W_{i,3}) \in \mathbb{B}_i$.

ι_i, p_i and $\mathrm{comm_i}$. We now define the maps $\iota_i : \mathbb{A}_i \to \mathbb{B}_i$, $p_i : \mathbb{B}_i \to \mathbb{A}_i$ and the commitment scheme $\mathrm{comm_i}$. There are two cases; $\mathbb{A}_i = \mathbb{F}_q$ and $\mathbb{A}_i = \mathbb{G}_i$.

$\underline{A_i = \mathbb{F}_q}$: We define the maps via

$$\iota_i : \begin{cases} \mathbb{F}_q & \longrightarrow & \mathbb{B}_i \\ x & \longmapsto & [x]\mathcal{W}_i \end{cases} \qquad p_i : \begin{cases} \mathbb{B}_i & \longrightarrow & \mathbb{F}_q \\ \mathcal{X} = ([c_1]P_i, [c_2]P_i, [c_3]P_i) & \longmapsto & c_3 - \frac{1}{a_i}c_1 - \frac{1}{t_i}c_2 \end{cases}$$

The commitment scheme $\mathrm{comm_i}$ is obtained as before, except we select $\hat{m}_i = 2$, as opposed to $\hat{m}_i = 3$, this again simplifies the equations. Hence we have

$$\mathrm{comm_i} : \begin{cases} \mathbb{F}_q \times \mathbb{F}_q \times \mathbb{F}_q & \longrightarrow & \mathbb{B}_i \\ (x, r_1, r_2) & \longmapsto & \iota_i(x) + [r_1]\mathcal{U}_i^{(1)} + [r_2]\mathcal{U}_i^{(2)} \end{cases}$$

$\underline{A_i = \mathbb{G}_i}$: We define

$$\iota_i : \begin{cases} \mathbb{G}_i \longrightarrow \mathbb{B}_i \\ X \longmapsto (\mathcal{O}, \mathcal{O}, X) \end{cases} \qquad p_i : \begin{cases} \mathbb{B}_i \longrightarrow \mathbb{G}_i \\ \mathcal{X} = (C_1, C_2, C_3) \longmapsto C_3 - [\frac{1}{a_i}]C_1 - [\frac{1}{t_i}]C_2 \end{cases}$$

The commitment scheme $\mathrm{comm_i}$ is obtained as in our main discussion, i.e. with $\hat{m}_i = 3$. Hence we have

$$\mathrm{comm_i} : \begin{cases} \mathbb{G}_i \times \mathbb{F}_q \times \mathbb{F}_q \times \mathbb{F}_q \longrightarrow \mathbb{B}_i \\ (X, r_1, r_2, r_3) \longmapsto \iota_i(X) + [r_1]\mathcal{U}_i^{(1)} + [r_2]\mathcal{U}_i^{(2)} + [r_3]\mathcal{U}_i^{(3)} \end{cases}$$

***ι_T* and *p_T*.** Here we have four cases, depending on which of the four types of equation we are dealing with

Pairing Product Equations.

$$\iota_T : \begin{cases} \mathbb{G}_T \longrightarrow \mathbb{B}_T \\ \zeta \longmapsto \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & \zeta \end{pmatrix} \end{cases} \qquad p_T : \begin{cases} \mathbb{B}_T \longrightarrow \mathbb{G}_T \\ \begin{pmatrix} \zeta_{1,1} & \zeta_{1,2} & \zeta_{1,3} \\ \zeta_{2,1} & \zeta_{2,2} & \zeta_{2,3} \\ \zeta_{3,1} & \zeta_{3,2} & \zeta_{3,3} \end{pmatrix} \longmapsto \gamma_1^{-1/a_2}\gamma_2^{-1/t_2}\gamma_3 \end{cases}$$

where $\gamma_i = \zeta_{1,i}^{-1/a_1}\zeta_{2,i}^{-1/t_1}\zeta_{3,i}$.

Multi-Scalar Multiplication in $\mathbb{G}_1$ and $\mathbb{G}_2$.
In both of these cases we have

$$p_T : \begin{cases} \mathbb{B}_T \longrightarrow \mathbb{G}_i \\ \begin{pmatrix} \zeta^{s_{1,1}} & \zeta^{s_{1,2}} & \zeta^{s_{1,3}} \\ \zeta^{s_{2,1}} & \zeta^{s_{2,2}} & \zeta^{s_{2,3}} \\ \zeta^{s_{3,1}} & \zeta^{s_{3,2}} & \zeta^{s_{3,3}} \end{pmatrix} \longmapsto [s_3 - \frac{1}{a_2}s_1 - \frac{1}{t_2}s_2]P_i \end{cases}$$

where $\zeta = \hat{t}(P_1, P_2)$ and

$$s_i = s_{3,i} - \frac{1}{a_1}s_{1,i} - \frac{1}{t_1}s_{2,i}.$$

For multi-scalar multiplication in $\mathbb{G}_1$ the map ι_T is defined by

$$\iota_T : \begin{cases} \mathbb{G}_1 \longrightarrow \mathbb{B}_T \\ X \longmapsto \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ \hat{t}(X, W_{2,1}) & \hat{t}(X, W_{2,2}) & \hat{t}(X, W_{2,3}) \end{pmatrix} \end{cases}$$

Whilst for multi-scalar multiplication in $\mathbb{G}_2$ the map ι_T is defined by

$$\iota_T : \begin{cases} \mathbb{G}_2 \longrightarrow \mathbb{B}_T \\ X \longmapsto \begin{pmatrix} 1 & 1 & \hat{t}(W_{1,1}, X) \\ 1 & 1 & \hat{t}(W_{1,2}, X) \\ 1 & 1 & \hat{t}(W_{1,3}, X) \end{pmatrix} \end{cases}$$

When specialised to the symmetric case these are different definitions of ι_T to those given in [26]. The above definitions produce the required commutative properties.

Quadratic Equations in $\mathbb{F}_q$.
In this case we have

$$p_T : \begin{cases} \mathbb{B}_T & \longrightarrow & \mathbb{F}_q \\ \begin{pmatrix} \zeta^{s_{1,1}} & \zeta^{s_{1,2}} & \zeta^{s_{1,3}} \\ \zeta^{s_{2,1}} & \zeta^{s_{2,2}} & \zeta^{s_{2,3}} \\ \zeta^{s_{3,1}} & \zeta^{s_{3,2}} & \zeta^{s_{3,3}} \end{pmatrix} & \longmapsto & s_3 - \frac{1}{a_2}s_1 - \frac{1}{t_2}s_2 \end{cases}$$

where again we have $\zeta = \hat{t}(P_1, P_2)$ and

$$s_i = s_{3,i} - \frac{1}{a_1}s_{1,i} - \frac{1}{t_1}s_{2,i}.$$

The function ι_T is given by

$$\iota_T(z) : \begin{cases} \mathbb{F}_q \longrightarrow \mathbb{B}_T \\ z \longmapsto F(\mathcal{W}_1, \mathcal{W}_2)^z. \end{cases}$$

Again this is different from the mapping given in [26].

4.3 Combining SXDH and SDLIN

We end this section by noting an extension which was pointed out to us by J. Groth [27]. If one wanted to work in Type-2 pairings and one wanted a more efficient instantiation one could implement a system using DDH in $\mathbb{G}_1$ and DLIN in $\mathbb{G}_2$. We do not expand on the details of this construction, but remark that this would imply that elements in $\mathbb{B}_1$ would consist of two elements in $\mathbb{G}_1$ and elements in $\mathbb{B}_2$ would consist of three elements in $\mathbb{G}_2$, with $\mathbb{B}_T$ consisting of six elements in $\mathbb{G}_T$. This added efficiency is at the expense of having to assume DDH in $\mathbb{G}_1$, which defeats the benefit which some people (although not the current authors) see behind the DLIN assumption based constructions in pairing based cryptography; namely that a single protocol description can apply in all main three pairing types.

5 Performance Comparison

In this section we compare the relative commitment sizes of the different instantiations, the resulting proof sizes can be deduced from these and show a similar relative comparison. ¿From the size of the elements in the groups $\mathbb{B}_1$ and $\mathbb{B}_2$ one can also easily estimate the relative computational performance figures, as group operations are essentially a quadratic function of the bit length. Before proceeding we also note that Groth–Sahai proofs will usually be used in the context of another protocol or scheme which is likely to dictate the exact pairing type one is using, hence the following comparison is only for illustrative purposes.

To provide concrete numbers we assume a security level equivalent to 128-bits of symmetric key security. Using "standard" comparisons of different key sizes this equates to a minimum size of $\mathbb{G}_T$ of 3072-bits and a minimum size of

elements in $\mathbb{G}_1$ of 256-bits. We let k denote the pairing embedding degree. For Type-1 pairings the value of k is bounded by two for elliptic curves defined over fields of large prime characteristic and by six for curves which are defined over fields of characteristic three. For Type-2 and Type-3 curves the "optimal" value of k at this security level is $k = 12$. A crucial observation is that for Type-3 curves we have the ability to compress the elements in $\mathbb{G}_2$ by a factor of six at this security level by using BN curves.

We summarize the commitment sizes (in bits), i.e. the size of elements in $\mathbb{B}_1$ and $\mathbb{B}_2$, as well as the proof sizes (also in bits), in Table 1. From the table it would appear that using the SDLIN setting as introduced in this paper gives no advantage. However, this overlooks the fact that the point of Groth–Sahai proofs is to use them in other protocols and schemes. These protocols and schemes may require one to work in the Type-2 setting, or to base ones security on the SDLIN assumption. Thus in these situations it makes more sense to use Groth–Sahai proofs suited to the particular protocol. In addition some researchers prefer the SDLIN setting to the SXDH setting as they prefer not to use the "special" pairing setting of Type-3, where there is no computable isomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$.

For the Type-1 setting we give two figures to represent the case of large prime characteristic and characteristic three. Note in all cases the size of a proof is equal to $\hat{m}_1$ elements of $\mathbb{B}_2$ and $\hat{m}_2$ elements of $\mathbb{B}_1$, except in the case of Type-1 pairings where due to the symmetric nature of the map $F(\mathcal{X}, \mathcal{Y})$ one can simplify this to $\max(\hat{m}_1, \hat{m}_2)$ elements of $\mathbb{B}_1 = \mathbb{B}_2$.

Table 1. Summary of the different instantiations

Pairing Type	1	2	3	3
Hard Problems	DLIN	SDLIN	SDLIN	SXDH
$\lvert\mathbb{G}_1\rvert$	1536/512	256	256	256
$\lvert\mathbb{G}_2\rvert$	1536/512	3072	512	512
$\lvert\mathbb{B}_1\rvert$	$3 \cdot \lvert\mathbb{G}_1\rvert = 4608/1536$	$3 \cdot \lvert\mathbb{G}_1\rvert = 768$	$3 \cdot \lvert\mathbb{G}_1\rvert = 768$	$2 \cdot \lvert\mathbb{G}_1\rvert = 512$
$\lvert\mathbb{B}_2\rvert$	$3 \cdot \lvert\mathbb{G}_2\rvert = 4608/1536$	$3 \cdot \lvert\mathbb{G}_2\rvert = 9216$	$3 \cdot \lvert\mathbb{G}_2\rvert = 1536$	$2 \cdot \lvert\mathbb{G}_2\rvert = 1024$
Pairing Product Equations				
$(\hat{m}_1, \hat{m}_2)$	(3,3)	(3,3)	(3,3)	(2,2)
Size	13824/4608	29952	6912	3072
Multi-scalar multiplication in $\mathbb{G}_1$				
$(\hat{m}_1, \hat{m}_2)$	(3,2)	(3,2)	(3,2)	(2,1)
Size	13824/4608	29184	6144	2560
Multi-scalar multiplication in $\mathbb{G}_2$				
$(\hat{m}_1, \hat{m}_2)$	(2,3)	(2,3)	(2,3)	(1,2)
Size	13824/4608	20736	5376	2048
Quadratic Equations in $\mathbb{F}_q$				
$(\hat{m}_1, \hat{m}_2)$	(2,2)	(2,2)	(2,2)	(1,1)
Size	9216/3072	19968	4608	1536

6 Summary

We have extended the Groth–Sahai techniques to pairings in the Type-2 setting, and to using the DLIN assumption in the Type-3 setting. This required us to introduce a minor extension to the DLIN hardness assumption. In doing so we corrected a number of mistakes in the formulae presented in [26]. Using our formulae all valid NIWI proofs in both the DLIN and SXDH settings will now verify.

Acknowledgements. The authors work was partially funded by the EU FP7 projects CACE and eCrypt-2. The work of the second author was supported by a Royal Society Wolfson Merit Award. The authors would like to thank Jens Groth and Amit Sahai for useful feedback on an earlier version of this manuscript.

References

1. Ateniese, G., Camenisch, J., de Medeiros, B., Hohenberger, S.: Practical group signatures without random oracles. Cryptology ePrint Archive, Report 2005/385 (2005)
2. Belenkiy, M., Camenisch, J., Chase, M., Kohlweiss, M., Lysyanskaya, A., Shacham, H.: Randomizable proofs and delegatable anonymous credentials. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 108–125. Springer, Heidelberg (2009)
3. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: P-signatures and noninteractive anonymous credentials. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 356–374. Springer, Heidelberg (2008)
4. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: Compact E-Cash and simulatable VRFs revisited. In: Shacham, H. (ed.) Pairing 2009. LNCS, vol. 5671, pp. 114–131. Springer, Heidelberg (2009)
5. Bellare, M., Rogaway, P.: Random oracles are practical: A Paradigm for Designing Efficient Protocols. In: Computer and Communications Security – CCS 1993, pp. 62–73. ACM, New York (1993)
6. Blum, M., Feldman, P., Micali, S.: Non-interactive zero-knowledge and its applications. In: Symposium on Theory of Computing – STOC 1988, pp. 103–112. ACM, New York (1988)
7. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)
8. Boneh, D., Goh, E., Nissim, K.: Evaluating 2-DNF Formulas on Ciphertexts. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 325–341. Springer, Heidelberg (2005)
9. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the Weil pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
10. Camenisch, J., Chandran, N., Shoup, V.: A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In: Joux, A. (ed.) EUROCRYPT 2009, vol. 5479, pp. 351–368. Springer, Heidelberg (2009)
11. Chatterjee, S., Menezes, A.: On cryptographic protocols employing asymmetric pairings – The role of Ψ revisited. Cryptology ePrint Archive, Report 2009/480 (2009)
12. Damgård, I.: Non-interactive circuit based proofs and non-interactive proofs of knowledge with preprocessing. In: Rueppel, R.A. (ed.) EUROCRYPT 1992. LNCS, vol. 658, pp. 341–355. Springer, Heidelberg (1993)

13. Damgård, I., Nielsen, J.B., Orlandi, C.: Essentially optimal universally composable oblivious transfer. In: Lee, P.J., Cheon, J.H. (eds.) ICISC 2008. LNCS, vol. 5461, pp. 318–335. Springer, Heidelberg (2009)
14. Feige, U., Shamir, A.: Witness indistinguishable and witness hidding protocols. In: Symposium on Theory of Computing, pp. 416–426. ACM, New York (1990)
15. Feige, U., Lapidot, D., Shamir, A.: Non-interactive zero-knowledge proofs based on a single random string. In: Foundations of Computer Science – FOCS 1990, pp. 308–317. ACM, New York (1990)
16. Freeman, D.M.: Converting pairing-based cryptosystems from composite-order groups to prime-order groups. Cryptology ePrint Archive, Report 2009/540 (2009)
17. Galbraith, S., Paterson, K., Smart, N.P.: Pairings for cryptographers. Discrete Applied Mathematics 156, 3113–3121 (2008)
18. Goldreich, O., Oren, Y.: Definitions and properties of zero-knowledge proof systems. Journal of Cryptology 7, 1–32 (1994)
19. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems (extended abstract). In: Symposium on Theory of Computing – STOC 1985, pp. 291–304. ACM, New York (1985)
20. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM Journal on Computing 18, 186–208 (1989)
21. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity. Journal of the ACM 38(3), 690–728 (1991)
22. Green, M., Hohenberger, S.: Universally composable adaptive oblivious transfer. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 179–197. Springer, Heidelberg (2008)
23. Groth, J.: Fully anonymous group signatures without random oracles. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 164–180. Springer, Heidelberg (2007)
24. Groth, J., Lu, S.: A non-interactive shuffle with pairing based verifiability. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 51–67. Springer, Heidelberg (2007)
25. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (2008)
26. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups (full version), `http://www.brics.dk/~jg/WImoduleFull.pdf`
27. Groth, J., Sahai, A.: Private Communication (December 2009)
28. Huang, Q., Yang, G., Wong, D.S., Susilo, W.: Ambiguous optimistic fair exchange. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp. 74–89. Springer, Heidelberg (2008)
29. Kilian, J., Petrank, E.: An efficient non-interactive proof system for NP with general assumptions. Journal of Cryptology 11, 1–27 (1998)
30. Liang, X., Cao, Z., Shao, J., Lin, H.: Short group signature without random oracles. In: Qing, S., Imai, H., Wang, G. (eds.) ICICS 2007. LNCS, vol. 4861, pp. 69–82. Springer, Heidelberg (2007)
31. Phong, L.T., Kurosawa, K., Ogata, W.: New DLOG-based convertible undeniable signature schemes in the standard model. Cryptology ePrint Archive, Report 2009/394
32. De Santis, A., Di Crescenzo, G., Persiano, G.: Randomness-optimal characterization of two NP proof systems. In: Rolim, J.D.P., Vadhan, S.P. (eds.) RANDOM 2002. LNCS, vol. 2483, pp. 179–193. Springer, Heidelberg (2002)