This is the full version of the extended abstract which appears in Proceedings of the Cryptographers' Track at the RSA Conference (CT-RSA 2016).

Short Structure-Preserving Signatures

Essam Ghadafi*

University College London, London, UK
e.ghadafi@ucl.ac.uk

Abstract. We construct a new structure-preserving signature scheme in the efficient Type-III asymmetric bilinear group setting with signatures shorter than all existing schemes. Our signatures consist of 3 group elements from the first source group and therefore have shorter size than all existing schemes as existing ones have at least one component of the signature in the second source group whose elements bit size is at least double their first group counterparts.
Besides enjoying short signatures, our scheme is fully re-randomizable which is a useful property for many applications. Our result also constitutes a proof that the impossibility of unilateral structure-preserving signatures in the Type-III setting result of Abe et al. (Crypto 2011) does not apply to constructions in which the message space is dual in both source groups. Besides checking the well-formedness of the message, verifying a signature in our scheme requires checking 2 Pairing Product Equations (PPE) and require the evaluation of only 5 pairings in total which matches the best existing scheme and outperforms many other existing ones. Reducing the number of pairings in the verification equations is very important when combining structure-preserving signature schemes with Groth-Sahai proofs as the number of pairings required for verifying Groth-Sahai proofs for PPE equations grows linearly with the number of pairing monomials in the source equations. We give some examples of how using our new scheme instead of existing ones improves the efficiency of some existing cryptographic protocols such as direct anonymous attestation and group signature related constructions.

Keywords. Structure-Preserving, Digital Signatures, Bilinear Groups.

1 Introduction

Structure-Preserving Signatures (SPS) [3] are digital signature schemes defined over bilinear groups ($e : \mathbb{G} \times \tilde{\mathbb{G}} \rightarrow \mathbb{T}$). Their messages, verification key and signatures are all group elements and signature verification involves evaluating Pairing Product Equations (PPE). They are a useful

*The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 307937 and EPSRC grant EP/J009520/1.

tool for the design of modular cryptographic protocols since they compose nicely with existing popular tools such as Groth-Sahai proofs [30] and El-Gamal encryption scheme [20]. They are prominently used in combination with Groth-Sahai proofs and other tools to design cryptographic protocols that do not rely on heuristic assumptions such as random oracles [21]. They have numerous applications which include group signatures, e.g. [3, 33, 34], blind signatures, e.g. [3, 23], tightly secure encryption schemes, e.g. [31, 2], malleable signatures, e.g. [9], anonymous credentials, e.g. [23], network coding, e.g. [9], oblivious transfer, e.g. [27].

Related Work. The notion was formally defined by Abe et al. [3] but earlier schemes conforming to the definition were given by Groth [28] and Green and Hohenberger [27]. Because of its importance, the notion has received a significant amount of attention from the cryptographic community and many results relating to proving lower bounds for the design of such schemes as well as new schemes meeting those lower bounds have been published in the literature. Abe et al. [3] gave two constructions of structure-preserving signatures both relying on non-interactive intractability assumptions. Abe et al. [4] proved that any structure-preserving signature scheme in the most efficient Type-III bilinear group setting (cf. Section 2.1) must have at least 3 group elements and 2 pairing product verification equations. They also ruled out the existence of unilateral signatures and argued that the signature must contain elements from both source groups. They also gave constructions meeting the lower bound and proved them secure in the generic group model [39]. Abe et al. [5] proved the impossibility of the existence of a 3 group element structure-preserving signature in the Type-III setting that is based on non-interactive intractability assumptions. In essence, their result implies that in the Type-III setting, the only way to meet the 3 group element lower bound is to either employ interactive intractability assumptions or resort to direct proofs in the generic group model. Ghadafi [25] gave a structure-preserving variant of the Camenisch-Lysyanskaya signature scheme [15] that is secure under an interactive assumption in the Type-III setting. Abe et al. [7] constructed a scheme in the Type-II setting (where there is an efficiently computable isomorphism from the second source group to the first) which contains only 2 group elements. Chatterjee and Menezes [17] revisited the work of [7] and showed that Type-III constructions outperform their Type-II counterparts. [17] also gave constructions in Type-III setting meeting the 3 group element lower bound. Barthe et al. [10] also gave optimal constructions of structure-preserving signatures in Type-II setting. Constructions relying on standard assumptions (such as DLIN and DDH) were given by [16, 1, 14, 2, 32, 34]. We remark here that constructions relying on standard assumptions are much less efficient than their counterparts relying on non-standard assumptions or those proven directly in the generic group model. More recently, Abe et al. [8] and Groth [29] gave fully structure-preserving constructions where even the secret key consists of only group elements.

While by now there exist a number of schemes, e.g. [4, 17, 6, 10, 29], with signatures meeting the 3 group element lower bound in the Type-III setting proved by Abe et al. [4], all those schemes have at least one component of the signature in group $\tilde{\mathbb{G}}$ whose elements bit size is at least double that of those in $\mathbb{G}$. To the best of our knowledge, the only existing structure-preserving signature scheme in the Type-III setting whose all signature components are in $\mathbb{G}$ is that of Ghadafi [25]. However, signatures of latter consist of 4 group elements and require 3 pairing-product verification equations.

Our Contribution. We construct a (unilateral) structure-preserving signature scheme with signatures shorter than all existing structure-preserving signatures. Our scheme yields fully re-randomizable signatures consisting of 3 group elements from the first short source group.

Our results also serve as a proof that the impossibility of unilateral structure-preserving signature schemes in the Type-III setting result of Abe et al. [4] does not apply when the message space is dual in both source groups. As is the tradition with most existing structure-preserving schemes, we prove the security of our scheme directly in the generic group model. Our scheme can be viewed as an extension of the recent non-structure-preserving signature scheme of Pointcheval and Sanders [37].

Finally, we show that replacing some existing schemes used as building blocks in some protocols with ours improves the efficiency of those protocols which include direct anonymous attestation and group signature related constructions.

Paper Organization. In Section 2, we give some preliminary definitions. In Section 3, we present our signature scheme and prove its security. We give some applications of our signature scheme in Section 4.

Notation. We write $y = A(x; r)$ when the algorithm A on input x and randomness r outputs y. We write $y \leftarrow A(x)$ for the process of setting $y = A(x; r)$ where r is sampled at random. We also write $y \leftarrow S$ for sampling y uniformly at random from a set S. A function $\nu(.) : \mathbb{N} \rightarrow \mathbb{R}^+$ is negligible (in n) if for every polynomial $p(.)$ and all sufficiently large values of n, it holds that $\nu(n) < \frac{1}{p(n)}$. By PPT we mean running in probabilistic polynomial time in the relevant security parameter. By $[k]$, we denote the set $\{1, \ldots, k\}$.

2 Preliminaries

In this section we provide some preliminary definitions.

2.1 Bilinear Groups

A bilinear group is a tuple $\mathcal{P} := (\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{T}, p, G, \tilde{G}, e)$ where $\mathbb{G}$, $\tilde{\mathbb{G}}$ and $\mathbb{T}$ are groups of a prime order p, and G and $\tilde{G}$ generate $\mathbb{G}$ and $\tilde{\mathbb{G}}$, respectively. The function e is a non-degenerate bilinear map $e : \mathbb{G} \times \tilde{\mathbb{G}} \longrightarrow \mathbb{T}$.

For clarity, elements of $\tilde{\mathbb{G}}$ will be accented with $\tilde{}$. We use multiplicative notation for all the groups. We let $\mathbb{G}^{\times} := \mathbb{G} \setminus \{1_{\mathbb{G}}\}$ and $\tilde{\mathbb{G}}^{\times} := \tilde{\mathbb{G}} \setminus \{1_{\tilde{\mathbb{G}}}\}$. In this paper, we work in the efficient Type-III setting [24], where $\mathbb{G} \neq \tilde{\mathbb{G}}$ and there is no efficiently computable isomorphism between the source groups in either direction. We assume there is an algorithm $\mathsf{BGSetup}$ that on input a security parameter λ, outputs a description of bilinear groups.

The message space of our signature scheme are elements of the subgroup $\hat{G}$ of $\mathbb{G} \times \tilde{\mathbb{G}}$ defined as the image of the map

$$\psi : \begin{cases} \mathbb{Z}_p \longrightarrow & \mathbb{G} \times \tilde{\mathbb{G}} \\ \quad x \longmapsto & (G^x, \tilde{G}^x) \end{cases}$$

Given an element $(M, \tilde{N}) \in \mathbb{G} \times \tilde{\mathbb{G}}$, one can efficiently test whether $(M, \tilde{N}) \in \hat{G}$ by checking $e(M, \tilde{G}) = e(G, \tilde{N})$. [1]

2.2 Complexity Assumptions

Definition 1 (Decisional Diffie-Hellman (DDH) Assumption). *The DDH assumption holds relative to a group setup $\mathcal{G}$ if for all PPT adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{l}(\mathbb{G}, G, p) \leftarrow \mathcal{G}(1^\lambda);\ r, s, t \leftarrow \mathbb{Z}_p;\ b \leftarrow \{0,1\}; \\ R := G^r;\ S := G^s;\ T := G^{brs+(1-b)t}\ : \mathcal{A}(G, R, S, T) = b\end{array}\right] \leq \frac{1}{2} + \nu(\lambda)\ .$$

Definition 2 (Symmetric External Diffie-Hellman (SXDH) Assumption). *Given a bilinear group $\mathcal{P} := (\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{T}, p, G, \tilde{G}, e)$, the SXDH assumption requires that the DDH assumption holds in both groups $\mathbb{G}$ and $\tilde{\mathbb{G}}$.*

[1] The elements of this group are called Diffie-Hellman pairs in [22, 3].

2.3 Digital Signatures

A digital signature scheme over a bilinear group $\mathcal{P}$ generated by $\mathsf{BGSetup}$ is a tuple $\mathcal{DS} := (\mathsf{KeyGen}, \mathsf{Sign}, \mathsf{Verify})$ whose definitions are:

- $\mathsf{KeyGen}(\mathcal{P})$ this probabilistic algorithm takes as input a bilinear group $\mathcal{P}$ and outputs a pair of secret/verification keys $(\mathsf{sk}, \mathsf{vk})$.
- $\mathsf{Sign}(\mathsf{sk}, m)$ this probabilistic algorithm takes as input a secret key sk and a message $m \in \mathcal{M}$, and outputs a signature σ.
- $\mathsf{Verify}(\mathsf{vk}, m, \sigma)$ this deterministic algorithm outputs 1 if σ is a vlaid signature on m w.r.t. the verification key vk.

Definition 3 (Correctness). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* (perfectly) correct *if for all $\lambda \in \mathbb{N}$*

$$\Pr\left[\begin{array}{c}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P}); \\ m \leftarrow \mathcal{M}; \sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m) : \mathsf{Verify}(\mathsf{vk}, m, \sigma) = 1\end{array}\right] = 1.$$

Definition 4 (Existential Unforgeability). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* existentially unforgeable *against adaptive chosen-message attack if for all $\lambda \in \mathbb{N}$ for all PPT adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{l}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P}); \\ (\sigma^*, m^*) \leftarrow \mathcal{A}^{\mathsf{Sign}(\mathsf{sk}, \cdot)}(\mathcal{P}, \mathsf{vk}) \\ \quad : \mathsf{Verify}(\mathsf{vk}, m^*, \sigma^*) = 1 \ \wedge \ m^* \notin Q_{\mathsf{Sign}}\end{array}\right] \leq \nu(\lambda),$$

where Q_{Sign} is the set of messages queried to Sign.

We consider schemes which are re-randomizable (i.e. weakly unforgeable) in the sense that given a signature on a message m, anyone without knowledge of the signing key, can compute a fresh signature on the same message. A desirable property for such class of schemes is that randomized signatures are indistinguishable from fresh signatures on the same message. Thus, we define an algorithm $\mathsf{Randomize}$ which on input (vk, m, σ), where σ being a valid signature on m, outputs a new signature σ' on m.

Definition 5 (Randomizability). *A signature scheme $\mathcal{DS}$ over a bilinear group generator* $\mathsf{BGSetup}$ *is* randomizable *if for all $\lambda \in \mathbb{N}$ for all stateful adversaries $\mathcal{A}$*

$$\Pr\left[\begin{array}{l}\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda); (\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KeyGen}(\mathcal{P}); \\ (\sigma^*, m^*) \leftarrow \mathcal{A}(\mathcal{P}, \mathsf{sk}, \mathsf{vk}); b \leftarrow \{0,1\}; \\ \sigma_0 \leftarrow \mathsf{Sign}(\mathsf{sk}, m^*); \sigma_1 \leftarrow \mathsf{Randomize}(\mathsf{vk}, m^*, \sigma^*); \\ \quad : \mathsf{Verify}(\mathsf{vk}, m^*, \sigma^*) = 1 \ \wedge \ \mathcal{A}(\sigma_b) = b\end{array}\right] \leq \frac{1}{2} + \nu(\lambda).$$

We say the scheme has *Perfect Randomizability* when $\nu(\lambda) = 0$. Note that the above definition of randomizability is stronger than the variant where the signature σ^* is generated by the challenger rather than the adversary herself.

Structure-Preserving Signatures. Structure-preserving signatures [3] are signature schemes defined over bilinear groups where the messages, the verification key and signatures are all group elements and verifying signatures only involves deciding group membership of the signature components and evaluating Pairing Product Equations (PPE) of the form of equation 1.

$$\prod_i \prod_j e(A_i, \tilde{B}_j)^{c_{i,j}} = 1_{\mathbb{T}}, \tag{1}$$

where $A_i \in \mathbb{G}$ and $\tilde{B}_j \in \tilde{\mathbb{G}}$ are group elements appearing in $\mathcal{P}, m, \mathsf{vk}, \sigma$, whereas $c_{i,j} \in \mathbb{Z}_p$ are constants.

2.4 Randomizable Weakly Blind Signatures

A randomizable weakly blind signature scheme, as defined by Bernhard et al. [12], is similar to a standard blind signature scheme [18] but unlike the latter, in the former, the signer never gets to see the signed message. More precisely, in the blindness game of the former (referred to as *weak blindness*), the challenge messages are chosen by the challenger rather than the adversary and are never revealed to the adversary.

Formally, a randomizable weakly blind signature scheme BS (with a two-move signature request phase) consists of the following polynomial-time algorithms $\mathsf{BS} := (\mathsf{Setup}_{\mathsf{BS}}, \mathsf{KeyGen}_{\mathsf{BS}}, \mathsf{Request}_{\mathsf{BS}}, \mathsf{Issue}_{\mathsf{BS}}, \mathsf{Verify}_{\mathsf{BS}}, \mathsf{Randomize}_{\mathsf{BS}})$. All algorithms (bar $\mathsf{Setup}_{\mathsf{BS}}$) are assumed to take as (implicit) input a parameter set $\mathsf{param}_{\mathsf{BS}}$ output by $\mathsf{Setup}_{\mathsf{BS}}$.

- $\mathsf{Setup}_{\mathsf{BS}}(1^\lambda)$ outputs public parameters $\mathsf{param}_{\mathsf{BS}}$.
- $\mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$ outputs a public/secret key pair $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}})$ for the signer.
- $(\mathsf{Request}^0_{\mathsf{BS}}, \mathsf{Issue}^1_{\mathsf{BS}}, \mathsf{Request}^1_{\mathsf{BS}})$ is an interactive protocol run between a user and a signer. The protocol is initiated by the user by calling $\mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m)$ to obtain a value ρ_0 and some state information st^0_R (which is assumed to contain the message m). Then the signer and user execute, respectively,

$$(\beta_1, \mathsf{st}^1_I) \leftarrow \mathsf{Issue}^1_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}, \rho_0) \quad \text{and} \quad \sigma \leftarrow \mathsf{Request}^1_{\mathsf{BS}}(\beta_1, \mathsf{st}^0_R),$$

where σ is a signature on the message m (or the reject symbol $\perp$). We write $\sigma \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}})\rangle$ for the output of correct running of this protocol on the given inputs.

- $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma)$ outputs 1 if σ is a valid signature on m and 0 otherwise.
- $\mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma)$ given a signature σ on an unknown message m, produces another valid signature σ' on the same message.

Definition 6 (Correctness). *A randomizable weakly blind signature scheme is* (perfectly) correct *if for all* $\lambda \in \mathbb{N}$

$$\Pr\left[\begin{array}{l} \mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^{\lambda}); (\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}}); \\ m \leftarrow \mathcal{M}_{\mathsf{BS}}; \sigma \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}})\rangle; \\ \sigma' \leftarrow \mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma) \\ \quad : \mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma) = 1 \ \wedge \ \mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m, \sigma') = 1 \end{array}\right] = 1.$$

Definition 7 (Unforgeability). *A randomizable weakly blind signature scheme is* unforgeable *if for all* $\lambda \in \mathbb{N}$*, all PPT adversaries* $\mathcal{A}$ *have a negligible advantage in the game in Fig. 1.*

Experiment: $\mathsf{Exp}^{\text{Unforge}}_{\mathsf{BS},\mathcal{A}}(\lambda)$:

– $\mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^{\lambda})$.

– $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$.

– $\Big((m_1, \sigma_1), \ldots, (m_{n+1}, \sigma_{n+1})\Big) \leftarrow \mathcal{A}^{\mathsf{Issue}_{\mathsf{BS}}(\cdot,\cdot)}(\mathsf{vk}_{\mathsf{BS}}, \mathsf{param}_{\mathsf{BS}})$.

– Return 0 if any of the following holds. Otherwise, Return 1:

  ∘ $\mathcal{A}$ called its oracle more than n times.

  ∘ $\exists i, j \in \{1, \ldots, n+1\}$ s.t. $i \neq j$, but $m_i = m_j$.

  ∘ $\exists i \in \{1, \ldots, n+1\}$ s.t. $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_i, \sigma_i) = 0$.

Fig. 1. The Unforgeability game for randomizable weakly blind signatures

Definition 8 (Weak Blindness). *A randomizable weakly blind signature scheme is* weakly blind *if for all* $\lambda \in \mathbb{N}$*, all PPT adversaries* $\mathcal{A}$ *have a negligible advantage in the game in Fig. 2.*

2.5 Groth-Sahai Proofs

Groth-Sahai (GS) proofs [30] are non-interactive proofs in the CRS model. We will use GS proofs that are secure under the SXDH assumption, which

```
Experiment: $\mathsf{Exp}^{\mathrm{wBlind}}_{\mathsf{BS},\mathcal{A}}(\lambda)$:
– $\mathsf{param}_{\mathsf{BS}} \leftarrow \mathsf{Setup}_{\mathsf{BS}}(1^{\lambda})$.
– $(\mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}) \leftarrow \mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$.
– $m_0, m_1 \leftarrow \mathcal{M}_{\mathsf{BS}}$.
– $(\rho_0, \mathsf{st}^0_R) \leftarrow \mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_0)$.
– $(\beta_1, \mathsf{st}_{\mathcal{A}}) \leftarrow \mathcal{A}(\mathsf{param}_{\mathsf{BS}}, \mathsf{vk}_{\mathsf{BS}}, \mathsf{sk}_{\mathsf{BS}}, \rho_0)$.
– $\sigma_0 \leftarrow \mathsf{Request}^1_{\mathsf{BS}}(\beta_1, \mathsf{st}^0_R)$.
– If $\sigma_0 =\perp$ or $\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_0, \sigma_0) = 0$ Then Return 0.
– $b \leftarrow \{0,1\}$.
– If $b = 0$ Then $\sigma_1 \leftarrow \mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma_0)$.
– Else $\sigma_1 \leftarrow \langle \mathsf{Request}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, m_1), \mathsf{Issue}_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}) \rangle$.
– $b^* \leftarrow \mathcal{A}(\mathsf{st}_{\mathcal{A}}, \sigma_0, \sigma_1)$.
– Return 1 If $b = b^*$ Else Return 0.
```

Fig. 2. The Weak Blindness game for randomizable weakly blind signatures

is the most efficient instantiation of the proof system [26], and that prove knowledge of witnesses to pairing-product equations of the form

$$\prod_{j=1}^{n} e(A_j, \underline{\tilde{Y}_j}) \prod_{i=1}^{m} e(\underline{X_i}, \tilde{B}_i) \prod_{i=1}^{m}\prod_{j=1}^{n} e(\underline{X_i}, \underline{\tilde{Y}_j})^{\gamma_{i,j}} = \prod_{\ell=1}^{k} e(G_\ell, \tilde{H}_\ell), \tag{2}$$

where all underlined variables are part of the witness whereas the rest of the values are public constants. The language for these proofs is of the form

$$\mathcal{L} := \{\mathsf{statement} \mid \exists\, \mathsf{witness} : E(\mathsf{statement}, \mathsf{witness}) \text{ holds}\},$$

where $E(\mathsf{statement}, \cdot)$ is a set of pairing-product equations.

The system is defined by a tuple of algorithms $\big(\mathsf{GSSetup}, \mathsf{GSProve}, \mathsf{GSVerify}, \mathsf{GSExtract}, \mathsf{GSSimSetup}, \mathsf{GSSimProve}\big)$. $\mathsf{GSSetup}$ takes as input the description of a bilinear group $\mathcal{P}$ and outputs a *binding* reference string crs and an extraction key xk. $\mathsf{GSProve}$ takes as input the string crs, a set of equations $\mathsf{statement}$ and a witness, and outputs a proof Ω for the satisfiability of the equations. $\mathsf{GSVerify}$ takes as input a set of equations, a string crs and a proof Ω and outputs 1 if the proof is valid, and 0 otherwise. $\mathsf{GSExtract}$ takes as input a binding crs, the extraction key xk and a valid proof Ω, and outputs the witness used for the proof. $\mathsf{GSSimSetup}$, on input a bilinear group $\mathcal{P}$, outputs a *hiding* string $\mathsf{crs}_{\mathrm{Sim}}$ and a trapdoor key tr that allows to simulate proofs. $\mathsf{GSSimProve}$ takes as input $\mathsf{crs}_{\mathrm{Sim}}$, a statement and the trapdoor tr and produces a simulated proof Ω_{Sim} without a witness. The distributions of strings crs and $\mathsf{crs}_{\text{Sim}}$ are computationally indistinguishable and simulated proofs are indistinguishable from proofs generated by an honest prover.

The proof system has perfect completeness, perfect soundness, composable witness-indistinguishability/composable zero-knowledge. We refer to [30] for the formal definitions and the details of the instantiation.

3 Our Structure-Preserving Signature Scheme

Given the description of Type-III bilinear groups $\mathcal{P}$ output by $\mathsf{BGSetup}(1^\lambda)$, our scheme is given by the following four algorithms.

- $\mathsf{KeyGen}(\mathcal{P})$: Select $x, y \leftarrow \mathbb{Z}_p^\times$. Set $\mathsf{sk} := (x, y)$ and $\mathsf{vk} := (\tilde{X}, \tilde{Y}) := (\tilde{G}^x, \tilde{G}^y)$.
- $\mathsf{Sign}(\mathsf{sk}, (M, \tilde{N}))$: To sign a message $(M, \tilde{N}) \in \hat{G}$, (i.e. $(M, \tilde{N}) \in \mathbb{G} \times \tilde{\mathbb{G}}$ and $e(M, \tilde{G}) = e(G, \tilde{N})$), select $a \leftarrow \mathbb{Z}_p^\times$, and set $A := G^a$, $B := M^a$, $C := A^x \cdot B^y$. Return $\sigma := (A, B, C) \in \mathbb{G}^3$.
- $\mathsf{Verify}(\mathsf{vk}, (M, \tilde{N}), \sigma = (A, B, C))$: Return 1 iff $A \in \mathbb{G}^\times$ (i.e. $A \neq 1_{\mathbb{G}}$), $B, C \in \mathbb{G}$, $(M, \tilde{N}) \in \hat{G}$ (i.e. $e(G, \tilde{N}) = e(M, \tilde{G})$), and all of the following hold:

$$e(A, \tilde{N}) = e(B, \tilde{G})$$
$$e(C, \tilde{G}) = e(A, \tilde{X}) e(B, \tilde{Y})$$

- $\mathsf{Randomize}(\mathsf{vk}, (M, \tilde{N}), \sigma = (A, B, C))$: Select $r \leftarrow \mathbb{Z}_p^\times$, and set $A' := A^r$, $B' := B^r$, $C' := C^r$. Return $\sigma' := (A', B', C')$.

Remark 1. Note that verifying the well-formedness of the message pair, i.e. that $(M, \tilde{N}) \in \hat{G}$, need only be done once when verifying multiple signatures on the same message. A similar argument applies to signature schemes with the same message space, e.g. [22, 3, 25].

Also, since the component A of the signature is information-theoretically independent of the message, even when proving knowledge of a signature, one can reveal A after re-randomizing it which allows for verifying that $A \neq 1_{\mathbb{G}}$ for free. We end by noting that Ghadafi [25] gave efficient Groth-Sahai proofs that a committed Groth-Sahai value is not the identity element.

Correctness of the scheme follows by inspection and is straightforward to verify. Also, that the signature is perfectly randomizable is straightforward. The distributions of valid signatures returned by the $\mathsf{Randomize}$ algorithm are identical to those returned by the Sign algorithm on the same message. Also, note that assuming the signature to be re-randomized is valid, one only needs the old signature to be able to produce a new one.

The following theorem proves that the scheme is unforgeable in the generic group model [39, 36]. We note here that the unforgeability of the scheme could also be based on an interactive assumption.

Theorem 1. *The structure-preserving signature scheme is (weakly) existentially unforgeable against adaptive chosen-message attack in the generic group model.*

Proof. The proof follows from the proof of the following theorem:

Theorem 2. *Let $\mathcal{A}$ be an adversary in the generic group model against our scheme. Assume $\mathcal{A}$ makes q_G group operation queries, q_P pairing queries, and q_S sign queries. The probability ϵ of adversary $\mathcal{A}$ winning the game is bounded by $\epsilon \leq \frac{(q_G+q_P+3q_S+4)^2 \cdot 3}{p}$, where p is the (prime) order of the generic groups.*

Proof. We start by re-stating the following Schwartz Zippel lemma [38]:

Lemma 1. *Let p be a prime and $P(x_1, \ldots, x_n) \in \mathbb{F}_p[x_1, \ldots, x_n]$ be a non-zero polynomial with a total degree $\leq d$. Then the probability that $P(x_1, \ldots, x_n) = 0$ is $\leq \frac{d}{p}$.*

Adversary $\mathcal{A}$ interacts with those oracles via group handles. We define three random encoding functions $\xi_1 : \mathbb{G} \longrightarrow \{0,1\}^*$, $\xi_2 : \tilde{\mathbb{G}} \longrightarrow \{0,1\}^*$ and $\xi_3 : \mathbb{T} \longrightarrow \{0,1\}^*$ where ξ_i maps elements from the corresponding group into random strings. The challenger keeps three lists $\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_T$ which contain pairs of the form (τ, P) where τ is a "random" encoding of the group element (i.e. τ is an output of the map ξ_i) and P is some polynomial in $\mathbb{F}_p[X, Y, A_1, \ldots, A_{q_S}]$.

To each list we associate an Update algorithm, that takes as input the specific list $\mathcal{L}_i$ and a polynomial P. The algorithm $\mathsf{Update}(\mathcal{L}_i, P)$ searches the list in question for a pair whose second component is equal to P, if such a pair is found , the algorithm returns its first component as a result. Otherwise, a new random encoding τ, different from all other elements used so far, is chosen and the pair (τ, P) is added to the list $\mathcal{L}_i$. The value τ is then returned. Note that at no point $\mathcal{A}$ gets access to the second element in the pairs.

The challenger starts the game by calling: $\mathsf{Update}(\mathcal{L}_1, 1)$, $\mathsf{Update}(\mathcal{L}_2, 1)$, $\mathsf{Update}(\mathcal{L}_2, X)$ and $\mathsf{Update}(\mathcal{L}_2, Y)$. Those correspond to the group elements $G \in \mathbb{G}$ and $\tilde{G}, \tilde{X}, \tilde{Y} \in \tilde{\mathbb{G}}$ of the verification key and public elements the adversary gets in the scheme.

The oracles used in the game are defined as follows:

- Group Oracles: Oracles $\mathcal{O}_1$, $\mathcal{O}_2$ and $\mathcal{O}_T$ allow $\mathcal{A}$ access to the group operations in groups $\mathbb{G}, \tilde{\mathbb{G}}$ and $\mathbb{T}$, respectively, via subtraction/addition operations. On a call to $\mathcal{O}_i(\tau_1, \tau_2)$ $\mathcal{B}$ searches list $\mathcal{L}_i$ for pairs of the form (τ_1, P_1) and (τ_2, P_2). If both pairs exist, $\mathcal{B}$ returns the output of $\mathsf{Update}(\mathcal{L}_i, P_1 \pm P_2)$. Otherwise, it returns $\perp$. Note that exponentiation operations can be performed by calls to the group operation oracles.
- Pairing Oracle: Oracle $\mathcal{O}_P$ allows $\mathcal{A}$ to perform pairing operations. On a call to $\mathcal{O}_P(\tau_1, \tau_2)$, $\mathcal{B}$ searches the list $\mathcal{L}_1$ for the pair (τ_1, P_1), and the list $\mathcal{L}_2$ for the pair (τ_2, P_2). If both pairs exist, $\mathcal{B}$ returns the output of $\mathsf{Update}(\mathcal{L}_T, P_1 \cdot P_2)$. Otherwise, it returns $\perp$.
- Sign Oracle: The adversary may make up to q_S queries $\mathcal{O}_S(\tau_1, \tau_2)$. The challenger searches list $\mathcal{L}_1$ for a pair (τ_1, P_1) and list $\mathcal{L}_2$ for a pair (τ_2, P_2). If they do not exist or $P_1 \neq P_2$, $\mathcal{B}$ returns $\perp$. Otherwise, it executes the following operations, where A_i, X and Y are indeterminants:

$$\begin{aligned}
\tau_{A_i} &\leftarrow \mathsf{Update}(\mathcal{L}_1, A_i), \\
\tau_{B_i} &\leftarrow \mathsf{Update}(\mathcal{L}_1, A_i \cdot P_1), \\
\tau_{C_i} &\leftarrow \mathsf{Update}(\mathcal{L}_1, A_i \cdot (X + P_1 \cdot Y)).
\end{aligned}$$

  Returning the tuple $(\tau_{A_i}, \tau_{B_i}, \tau_{C_i})$ to $\mathcal{A}$.

At the end of the game, the total number of non-constant polynomials contained in the three lists $\mathcal{L}_1, \mathcal{L}_2$ and $\mathcal{L}_T$ is bounded from above by $t = q_G + q_P + 3q_S + 4$.

By using the above oracles, we can simulate the entire run of the adversary where the adversary may make no decision which depends on the particular encoding of group elements used.

The Adversary Output. Eventually, $\mathcal{A}$ outputs a tuple $\big(\tau_{A^*}, \tau_{B^*}, \tau_{C^*}, \tau_{M^*}, \tau_{\tilde{N}^*}\big)$, where τ_{A^*}, τ_{B^*}, τ_{C^*}, and τ_{M^*} are on list $\mathcal{L}_1$ while $\tau_{\tilde{N}^*}$ is on list $\mathcal{L}_2$. We let $P_{A^*}, P_{B^*}, P_{C^*}, P_{M^*}, P_{\tilde{N}^*}$ denote the polynomials associated with these encodings. For $\mathcal{A}$'s output to be valid, those polynomials can be assumed to satisfy, for some assignment $(x, y, a_1, \ldots, a_{q_S}) \in \mathbb{F}_p^{2+q_S}$ to the variables $(X, Y, A_1, \ldots, A_{q_S})$, the equations:

$$P_{B^*} = P_{A^*} \cdot P_{\tilde{N}^*} \tag{3}$$

$$P_{C^*} = P_{A^*} \cdot X + P_{B^*} \cdot Y \tag{4}$$

$$P_{M^*} = P_{\tilde{N}^*} \tag{5}$$

From this we derive a contradiction, i.e. conclude that the adversary cannot win the game. To achieve this, we need to first ensure that these polynomial identities cannot hold identically, i.e. regardless of any particular assignment $(x, y, a_1, \ldots, a_{q_S}) \in \mathbb{F}_p^{2+q_S}$ to the variables $(X, Y, A_1, \ldots, A_{q_S})$.

Let $(M_i, \tilde{N}_i)$ denote the i-th query to the sign oracle, where we discount queries which return $\perp$. Note that the only polynomials on the list $\mathcal{L}_2$ are linear combinations of the terms $1, X$ and Y we see that we must have $P_{\tilde{N}_i} = r_i + s_i \cdot X + t_i \cdot Y$. Since we must have $P_{M_i} = P_{\tilde{N}_i}$, this implies that the above polynomials must also appear on the list $\mathcal{L}_1$. However, it is also clear that there is no operation in $\mathbb{G}$ which creates a polynomial with a monomial term of X, nor one of Y. Thus, we conclude that all queries to the sign oracle correspond to elements whose polynomials are a constant term of the form $P_{M_i} = P_{\tilde{N}_i} = r_i$.

By a similar argument, we can also deduce that the output of the adversary corresponds to polynomials with $P_{M^*} = P_{\tilde{N}^*} = r^*$. This is precisely where we use the property that the oracle will return $\perp$ unless the input query lies in $\hat{G}$.

Since the queries are for constant polynomials only, we see that the only polynomials which can appear on the list $\mathcal{L}_1$ are of the form

$$P_{A^*} = w_1 + \sum_{i=1}^{q} u_{1,i} \cdot A_i + \sum_{i=1}^{q} v_{1,i} \cdot A_i \cdot (X + r_i \cdot Y) \tag{6}$$

$$P_{B^*} = w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i + \sum_{i=1}^{q} v_{2,i} \cdot A_i \cdot (X + r_i \cdot Y) \tag{7}$$

$$P_{C^*} = w_3 + \sum_{i=1}^{q} u_{3,i} \cdot A_i + \sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot (X + r_i \cdot Y) \tag{8}$$

where $w_j, u_{j,i}, v_{j,i} \in \mathbb{F}_p$.

First note that P_{C^*}, i.e. equation 8, there is no monomial with a power > 1 of Y. Also, there is no monomial in $X \cdot Y$. This implies we must have $v_{1,i} = v_{2,i} = 0$ for all i, i.e. neither P_{A^*} nor P_{B^*} should contain any monomials in X or Y. This means we have

$$P_{A^*} = w_1 + \sum_{i=1}^{q} u_{1,i} \cdot A_i$$

$$P_{B^*} = w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i$$

Now by Equation (3) we must have that

$$w_2 + \sum_{i=1}^{q} u_{2,i} \cdot A_i = r^* \cdot w_1 + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i$$

For the above to hold, we must have $w_2 = r^* \cdot w_1$ and $r^* \cdot u_{1,i} = u_{2,i}$ for all i.

By Equation (4) we must have

$$\begin{aligned} w_3 + \sum_{i=1}^{q} u_{3,i} \cdot A_i + \sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot (X + r_i \cdot Y) \\ = w_1 \cdot X + \sum_{i=1}^{q} u_{1,i} \cdot A_i \cdot X + r^* \cdot w_1 \cdot Y + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i \cdot Y \end{aligned}$$

There is no term in X on the left so we must have $w_1 = 0$. Also, no constant terms or terms in A_i on the right so we must have $w_3 = 0$ and $u_{3,i} = 0$ for all i. Thus, we must have

$$\sum_{i=1}^{q} v_{3,i} \cdot A_i \cdot X + \sum_{i=1}^{q} v_{3,i} \cdot r_i \cdot A_i \cdot Y = \sum_{i=1}^{q} u_{1,i} \cdot A_i \cdot X + \sum_{i=1}^{q} r^* \cdot u_{1,i} \cdot A_i \cdot Y$$

By the term $A_i \cdot X$, we must have $u_{1,i} = v_{3,i}$ for all i. Since we must have $A^* \neq 1_{\mathbb{G}}$, we must have at least one pair $u_{1,i} = v_{3,i} \neq 0$ for some i. Also, by the term $A_i \cdot Y$, we must have $v_{3,i} \cdot r_i = r^* \cdot u_{1,i}$. Since as we have seen we must have $u_{1,i} = v_{3,i}$, we have $r_i = r^*$ which contradicts the unforgeability requirement as the forgery is on a message pair that was queried to the sign oracle.

Thus, the adversary must win, or tell it is in a simulation, via a specific (random) assignment to the variables. We now turn to bounding the probability that the adversary wins (or detects the simulation) in this case.

The Simulation. Now $\mathcal{B}$ chooses random values $x, y, a_i \in \mathbb{F}_p$ and evaluates the polynomials. We need to show that the challenger's simulation is sound. If $\mathcal{A}$ learned it was interacting in a simulated game, there would be two different polynomials $P_{i,j}(x, y, a_i) = P_{i,j'}(x, y, a_i)$ in list $\mathcal{L}_i$ where $P_{i,j} \neq P_{i,j'}$. The simulation will fail if any of the following is correct:

$$P_{1,j}(x, y, a_i) = P_{1,j'}(x, y, a_i) \tag{9}$$

$$P_{2,j}(x, y, a_i) = P_{2,j'}(x, y, a_i) \tag{10}$$

$$P_{T,j}(x, y, a_i) = P_{T,j'}(x, y, a_i) \tag{11}$$

Since the maximum degree of any polynomial in list $\mathcal{L}_1 \leq 2$, by applying [39][Lemma 1], we have that the probability of Equation (9) holding is $\leq \frac{2}{p}$. Similarly, since the maximum degree of any polynomial in list $\mathcal{L}_2$ ≤ 1, we have that the probability of Equation (10) holding is $\leq \frac{1}{p}$. Finally, the probability of Equation (11) holding is $\leq \frac{3}{p}$.

Summing over all possible values of j in each case, we have

$$\epsilon \leq \binom{|\mathcal{L}_1|}{2}\frac{2}{p} + \binom{|\mathcal{L}_2|}{2}\frac{1}{p} + \binom{|\mathcal{L}_T|}{2}\frac{3}{p},$$

where $|\mathcal{L}_i|$ denotes the size of list $\mathcal{L}_i$.

In conclusion, the probability that an adversary wins the unforgeability game is bounded by $\epsilon \leq \frac{(q_G+q_P+3q_S+4)^2 \cdot 3}{p}$.

3.1 Useful Properties of Our Scheme

As mentioned above, having all 3 components of the signature in group $\mathbb{G}$ means our signatures are shorter than those of all existing schemes. Also, our scheme is fully re-randomizable and re-randomized signatures are indistinguishable from fresh signatures on the same message. Thus, this might provide efficiency gains when, for example, combined with Groth-Sahai proofs [30] as one does not need to hide all components of the signature if one needs to prove knowledge of a signature. In addition, even though one has to know both sides of the message when requesting signatures on a message, only the $\mathbb{G}$ component of the message is actually needed for signing, whereas verification only requires the $\tilde{\mathbb{G}}$ component. For instance, as we will show, our scheme is ideal for instantiating randomizable weakly blind signatures [12, 11] used for constructing direct anonymous attestation protocols.

3.2 Efficiency Comparison

We compare in Table 1 the efficiency of our scheme with that of existing schemes for a single a message in the Type-III setting. For concrete comparison, for instance, at 128-bit security, elements of $\mathbb{G}$ and $\tilde{\mathbb{G}}$ in Type-III are 256 and 512 bits long, respectively. Therefore, our signatures at this security level are at least 256 bits shorter than the best existing scheme. The efficiency gain is even better as the security level increases. Also, as can be seen from the table, our scheme compares favorably to existing ones in terms of the efficiency of the verification equation. Since it is well-known that Type-III constructions outperform their Type-I and Type-II counterparts, we limit our comparison to those efficient constructions in the Type-III setting.

For the schemes in [22, 25] as well as ours (i.e. those whose message space is $\hat{G}$), the cost does not include checking membership of the message in the relevant group. As discussed earlier, such a check only needs to be performed once when verifying multiple signatures on the same message.

Note that even though some of the existing schemes have the property that a small number of the pairings used in the verification equations can be pre-computed, this provides no advantage in the context of structure-preserving signatures as according to the definition, verification equations should not contain any elements from the target group other than the identity element $1_{\mathbb{T}}$. Therefore, one still has to evaluate all pairings for each signature. This is, for instance, required when producing Groth-Sahai zero-knowledge proofs of knowledge of a signature.

It is obvious that structure-preserving signatures (on unilateral messages) in the Type-III setting have shorter messages than schemes, including ours, whose message space is $\hat{G}$. However, we stress that this is a small price to pay to get shorter signatures and more efficient verification while remaining in the most efficient Type-III bilinear group setting.

Scheme	Size				Randomize?	Assumptions	Verification	
	σ	vk	Param	m			#PPE	#Pairings
[27] [a]	$\mathbb{G}^4 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}^2$	-	$\mathbb{G}$	Yes	q-HLRSW	4	8
[22]	$\mathbb{G}^3 \times \tilde{\mathbb{G}}^2$	$\mathbb{G} \times \tilde{\mathbb{G}}$	$\mathbb{G}^3$	$\hat{G}$	No	q-ADHSDH + AWFCDH	3	7
[3] I	$\mathbb{G}^5 \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^{10} \times \tilde{\mathbb{G}}^4$	-	$\mathbb{G}$	Partially	q-SFP	2	12
[3] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}^5$	$\mathbb{G}^{10} \times \tilde{\mathbb{G}}^4$	-	$\tilde{\mathbb{G}}$	Partially	q-SFP	2	12
[4] I	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\mathbb{G} \times \tilde{\mathbb{G}}^3$	-	$\mathbb{G} \times \tilde{\mathbb{G}}$	No	GGM	2	7
[4] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\mathbb{G} \times \tilde{\mathbb{G}}$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	5
[25]	$\mathbb{G}^4$	$\tilde{\mathbb{G}}^2$	-	$\hat{G}$	Yes	DH-LRSW	3	6
[17] I	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	No	GGM	2	5
[17] II	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	6
[17] III	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}^2$	-	$\mathbb{G}$	Yes	GGM	2	6
[6] I	$\mathbb{G}^3 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	$\mathbb{G}$	$\mathbb{G}$	Yes	GGM	2	6
[6] II	$\mathbb{G}^2 \times \tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	$\mathbb{G}$	$\mathbb{G}$	No	GGM	2	6
[10]	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}^2$	-	$\tilde{\mathbb{G}}$	Yes	GGM	2	5
[29] I	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}$	$\tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	Yes	GGM	2	6
[29] II	$\mathbb{G} \times \tilde{\mathbb{G}}^2$	$\mathbb{G}$	$\tilde{\mathbb{G}}$	$\tilde{\mathbb{G}}$	No	GGM	2	7
Ours	$\mathbb{G}^3$	$\tilde{\mathbb{G}}^2$	-	$\hat{G}$	Yes	GGM	2	5

[a] This scheme is only secure against a random message attack.

Table 1. Efficiency comparison between our scheme and other schemes

4 Applications of Our Scheme

In this section we show how using our signature scheme instead of some existing schemes improves the efficiency of some existing cryptographic protocols. We list here two examples which include direct anonymous attestation [13] and group signature related primitives.

4.1 Direct Anonymous Attestation

Bernhard et al. [11] gave the first instantiations of Direct Anonymous Attestation (DAA) [13] which do not rely on random oracles. Their constructions are instantiations of Bernhard et al. [12] generic construction. Among other things, the generic construction of the latter requires a randomizable weakly blind signature. The weakly blind signature is used in the join protocol to issue a credential to the user without learning her secret key. Note that unlike in group signatures [19], in DAA users do not have public keys matching their secret keys.

To get an efficient instantiation of the notion and hence an efficient instantiation of DAA (without relying on random oracles), the efficient instantiation of Bernhard et al. [11] combined Ghadafi's structure-preserving signature scheme [25] with Groth-Sahai proofs [30] to construct an efficient weakly blind signature scheme. Their weakly blind signature instantiation yields signatures of size $\mathbb{G}^4$ and require 3 PPE equations (7 pairings or 6 pairings and 1 elliptic curve point addition in total) to verify. Exploiting the fact that our signature scheme has a similar structure to Ghadafi's scheme but yet has shorter signatures and the verification algorithm is more efficient, we get a more efficient instantiation of weakly blind signatures and hence DAA by using our scheme instead. The weakly blind signature (see Fig. 3) obtained by combining our signature scheme with Groth-Sahai proofs yields signatures of size $\mathbb{G}^3$ and require only 2 PPE equations (5 pairings in total) to verify. Also, the communication complexity of both the user and the signer in the signing protocol is the same as that in the instantiation in [11]. Thus, using our scheme one gets more efficient instantiations of DAA without relying on random oracles.

In the construction detailed in Fig. 3, we use the following languages for the zero-knowledge proofs for the user and signer respectively [2]:

$$\mathcal{L}_1 : \Big\{ \big(M, (\tilde{N}, \tilde{G}')\big) : \quad e(G, \underline{\tilde{N}}) = e(M, \underline{\tilde{G}'}) \quad \wedge \quad \underline{\tilde{G}'} \cdot \tilde{G}^{-1} = 1_{\tilde{\mathbb{G}}} \Big\}$$

[2] The purpose of the two multi-scalar multiplication equations is to make the equations simulatable so that the proofs are zero-knowledge [30].

```
$\mathsf{Setup}_{\mathsf{BS}}(1^\lambda)$
    $\mathcal{P} \leftarrow \mathsf{BGSetup}(1^\lambda)$. $(\mathsf{crs}_1, \mathsf{xk}_1) \leftarrow \mathsf{GSSetup}(\mathcal{P})$.
    $(\mathsf{crs}_2, \mathsf{xk}_2) \leftarrow \mathsf{GSSetup}(\mathcal{P})$.
    Return $\mathsf{param}_{\mathsf{BS}} := (\mathcal{P}, \mathsf{crs}_1, \mathsf{crs}_2)$.

$\mathsf{KeyGen}_{\mathsf{BS}}(\mathsf{param}_{\mathsf{BS}})$
    $x, y \leftarrow \mathbb{Z}_p$. $\tilde{X} := \tilde{G}^x; \tilde{Y} := \tilde{G}^y$.
    Return $\big(\mathsf{sk}_{\mathsf{BS}} := (x, y), \mathsf{vk}_{\mathsf{BS}} := (\tilde{X}, \tilde{Y})\big)$.

$\mathsf{Request}^0_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, (M, \tilde{N}))$
    $\pi \leftarrow \mathsf{GSProve}\left(\mathsf{crs}_1, \{\tilde{N}, \tilde{G}'\} : M \in \mathcal{L}_1\right)$.
    Return $(\rho_0 := (M, \pi), \mathsf{st}^0_R := (M, \tilde{N}))$.

$\mathsf{Issue}^1_{\mathsf{BS}}(\mathsf{sk}_{\mathsf{BS}}, \rho_0)$
    Parse $\rho_0$ as $(M, \pi)$.
    If $\mathsf{GSVerify}(\mathsf{crs}_1, M \in \mathcal{L}_1, \pi) = 0$, Return $\perp$.
    $a \leftarrow \mathbb{Z}_p^\times$; $A := G^a$; $B := M^a$; $C := A^x \cdot B^y$.
    $\Omega \leftarrow \mathsf{GSProve}(\mathsf{crs}_2, \{\tilde{A}, \tilde{G}'\} : (A, B, M) \in \mathcal{L}_2)$.
    Return $\beta_1 := \big((A, B, C), \Omega\big)$.
```

```
$\mathsf{Request}^1_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \beta_1, \mathsf{st}^0_R)$
    Parse $\beta_1$ as $((A, B, C), \Omega)$.
    Parse $\mathsf{st}^0_R$ as $(M, \tilde{N})$.
    Return $\perp$ if any of the following hold:
        ○ $A = 1_{\mathbb{G}}$.
        ○ $e(C, \tilde{G}) \neq e(A, \tilde{X}) e(B, \tilde{Y})$.
        ○ $\mathsf{GSVerify}(\mathsf{crs}_2, (A, B, M) \in \mathcal{L}_2, \Omega) = 0$.
    Return $\sigma \leftarrow \mathsf{Randomize}_{\mathsf{BS}}\big(\mathsf{vk}_{\mathsf{BS}}, (A, B, C)\big)$.

$\mathsf{Verify}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, (M, \tilde{N}), (A, B, C))$
    If $A = 1_{\mathbb{G}}$ or $e(A, \tilde{N}) \neq e(B, \tilde{G})$
        or $e(C, \tilde{G}) \neq e(A, \tilde{X}) e(B, \tilde{Y})$
        Then Return 0.
    Else Return 1.

$\mathsf{Randomize}_{\mathsf{BS}}(\mathsf{vk}_{\mathsf{BS}}, \sigma)$
    Parse $\sigma$ as $(A, B, C)$.
    $r \leftarrow \mathbb{Z}_p^\times$; $A' := A^r$; $B' := B^r$; $C' := C^r$.
    Return $(A', B', C')$.
```

Fig. 3. Our Weakly Blind Signature Scheme

$$\mathcal{L}_2 : \Big\{\big((A, B, M), (\tilde{A}, \tilde{G}')\big) : \quad e(G, \underline{\tilde{A}}) = e(A, \underline{\tilde{G}'}) \quad \wedge \quad e(M, \underline{\tilde{A}}) = e(B, \underline{\tilde{G}'}) \\ \wedge \ \underline{\tilde{G}'} \cdot \tilde{G}^{-1} = 1_{\tilde{\mathbb{G}}}\Big\}$$

We prove following theorem in Appendix A.

Theorem 3. *If the SXDH assumption holds and the signature scheme is existentially unforgeable, the weakly blind signature scheme in Fig. 3 is secure.*

4.2 Group Signatures and Similar Primitives

In all constructions of group signatures [19], the issuer (the group manager) issues membership certificates by certifying users' verification keys. The message space of our scheme being the set of Diffie-Hellman pairs makes our scheme ideal to be combined with the automorphic structure-preserving signature scheme of Fuchsbauer [22, 3]. For instance, combining our signature scheme with Fuchsbauer's blind signature scheme [22, 3], we get more efficient instantiations of group blind signatures [35, 25] (without relying on random oracles) than those in [25]. An instantiation using our signature scheme yields group blind signatures of size $36 \cdot |\mathbb{G}| + 34 \cdot |\tilde{\mathbb{G}}|$ compared to $38 \cdot |\mathbb{G}| + 36 \cdot |\tilde{\mathbb{G}}|$ and $42 \cdot |\mathbb{G}| + 38 \cdot |\tilde{\mathbb{G}}|$ for the original constructions given in [25]. Also, since the final signature involves less

Groth-Sahai proofs, the verification algorithm is much more efficient as each Groth-Sahai proof requires a few pairings to verify.

References

1. M. Abe, M. Chase, B. David, M. Kohlweiss, R. Nishimaki and M. Ohkubo. Constant-Size Structure-Preserving Signatures: Generic Constructions and Simple Assumptions. In *ASIACRYPT 2012*, Springer LNCS 7658, 4–24, 2012.
2. M. Abe, B. David, M. Kohlweiss, R. Nishimaki and M. Ohkubo. Tagged One-Time Signatures: Tight Security and Optimal Tag Size. In *PKC 2013*, Springer LNCS 7778, 312–331, 2013.
3. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev and M. Ohkubo. Structure-preserving signatures and commitments to group elements. In *CRYPTO 2010*, Springer LNCS 6223, 209–236, 2010.
4. M. Abe, J. Groth, K. Haralambiev and M. Ohkubo. Optimal Structure-Preserving Signatures in Asymmetric Bilinear Groups. In *CRYPTO 2011*, Springer LNCS 6841, 649–666, 2011.
5. M. Abe, J. Groth and M. Ohkubo. Separating Short Structure-Preserving Signatures from Non-interactive Assumptions. In *ASIACRYPT 2011*, Springer LNCS 7073, 628–646, 2011.
6. M. Abe, J. Groth, M. Ohkubo and M. Tibouchi. Unified, Minimal and Selectively Randomizable Structure-Preserving Signatures. In TCC 2014, Springer LNCS 8349, 688–712, 2014.
7. M. Abe, J. Groth, M. Ohkubo and M. Tibouchi. Structure-Preserving Signatures from Type II Pairings. In CRYPTO 2014, Springer LNCS 8616, 390–407, 2014.
8. M. Abe, M. Kohlweiss, M. Ohkubo and M. Tibouchi. Fully Structure-Preserving Signatures and Shrinking Commitments. In *EUROCRYPT 2015*, Springer LNCS 9057, 35–65, 2015.
9. N. Attrapadung, B. Libert and T. Peters. Computing on authenticated data: new privacy definitions and constructions. In *ASIACRYPT 2012*, Springer LNCS 7658, 367–385, 2012.
10. G. Barthe, E. Fagerholm, D. Fiore, A. Scedrov, B. Schmidt and M. Tibouchi. Strongly-Optimal Structure Preserving Signatures from Type II Pairings: Synthesis and Lower Bounds. In *PKC 2015*, Springer LNCS 9020, 355–376, 2015.
11. D. Bernhard, G. Fuchsbauer and E. Ghadafi. Efficient Signatures of Knowledge and DAA in the Standard Model. In *ACNS 2013*, Springer LNCS 7954, 518–533, 2013.
12. D. Bernhard, G. Fuchsbauer, E. Ghadafi, N.P. Smart and B. Warinschi. Anonymous attestation with user-controlled linkability. In International Journal of Information Security, volume 12(3), 219–249, 2013.
13. E. Brickell, J. Camenisch and L. Chen. Direct anonymous attestation. *CCS 2004*, ACM, 132–145, 2004.
14. J. Camenisch, M. Dubovitskaya and K. Haralambiev. Efficient Structure-Preserving Signature Scheme from Standard Assumptions. In *SCN 2012*, Springer LNCS 7485, 76–94, 2012.
15. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *CRYPTO 2004*, Springer LNCS 3152, 56–72, 2004.
16. M. Chase and M. Kohlweiss. A New Hash-and-Sign Approach and Structure-Preserving Signatures from DLIN. In *SCN 2012*, Springer LNCS 7485, 131–148, 2012.

17. S. Chatterjee and A. Menezes. Type 2 Structure-Preserving Signature Schemes Revisited. In *Cryptology ePrint Archive, Report 2014/635*. `http://eprint.iacr.org/2014/635.pdf`.
18. D. Chaum. Blind signatures for untraceable payments. In *CRYPTO 1982*, Springer LNCS, 199–203, 1983.
19. D. Chaum and E. van Heyst. Group signatures. In *EUROCRYPT 1991*, Springer LNCS 547, 257–265, 1991.
20. T. ElGamal. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. In IEEE Transactions on Information Theory, volume 31(4), 469–472, 1985.
21. A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO 1986*, Springer LNCS 263, 186–194, 1986.
22. G. Fuchsbauer. Automorphic Signatures in Bilinear Groups and an Application to Round-Optimal Blind Signatures. In *Cryptology ePrint Archive, Report 2009/320*. `http://eprint.iacr.org/2009/320.pdf`.
23. G. Fuchsbauer. Commuting signatures and verifiable encryption. In *EUROCRYPT 2011*, Springer LNCS 6632, 224–245, 2011.
24. S. Galbraith, K. Paterson and N.P. Smart. Pairings for cryptographers. *Discrete Applied Mathematics*, **156**, 3113–3121, 2008.
25. E. Ghadafi. Formalizing Group Blind Signatures and Practical Constructions without Random Oracles. In *ACISP 2013*, Springer LNCS 7959, 330–346, 2013.
26. E. Ghadafi, N.P. Smart and B. Warinschi. Groth-Sahai proofs revisited. In *PKC 2010*, Springer LNCS 6056, 177–192, 2010.
27. M. Green and S. Hohenberger. Universally Composable Adaptive Oblivious Transfer. In *ASIACRYPT 2008*, Springer LNCS 5350, 179–197, 2008.
28. J. Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In *ASIACRYPT 2006*, Springer LNCS 4284, 444–459, 2006.
29. J. Groth. Efficient Fully Structure-Preserving Signatures for Large Messages . In *Cryptology ePrint Archive, Report 2015/824*. `http://eprint.iacr.org/2015/824.pdf`.
30. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In SIAM Journal on Computing, volume 41(5), 1193–1232, 2012.
31. D. Hofheinz and T. Jager. Tightly Secure Signatures and Public-Key Encryption. In *CRYPTO 2012*, Springer LNCS 7417, 590–607, 2012.
32. E. Kiltz, J. Pan and H. Wee. Structure-Preserving Signatures from Standard Assumptions, Revisited. In *CRYPTO 2015*, Springer LNCS 9216, 275–295, 2015.
33. B. Libert, T. Peters and M. Yung. Scalable Group Signatures with Revocation. In *EUROCRYPT 2012*, Springer LNCS 7237, 609–627, 2012.
34. B. Libert, T. Peters and M. Yung. Short Group Signatures via Structure-Preserving Signatures: Standard Model Security from Simple Assumptions. In *CRYPTO 2015*, Springer LNCS 9216, 296–316, 2015.
35. A. Lysyanskaya and R. Zulfikar. Group blind digital signatures: A scalable solution to electronic cash. In *FC 1998*, Springer LNCS 1465, 184–197, 1998.
36. U. Maurer. Abstract models of computation in cryptography. In *Cryptography and Coding 2005*, Springer LNCS 3796, 1–12, 2005.
37. D. Pointcheval and O. Sanders. Short Randomizable Signatures. In *Cryptology ePrint Archive, Report 2015/525*. `http://eprint.iacr.org/2015/525.pdf`.
38. J. T. Schwartz. Fast Probabilistic Algorithms for Verification of Polynomial Identities. In *J. ACM*, **27**, 701–717, 1980.
39. V. Shoup. Lower Bounds for Discrete Logarithms and Related Problems. In *EUROCRYPT 1997*, Springer LNCS 3152, 41–55, 1997.

A Proof of Theorem 3

Correctness of the construction follows from that of the signature scheme and the perfect completeness of Groth-Sahai proofs. Unforgeability and weak blindness are proven by the following two lemmata, respectively.

Lemma 2. *The weakly blind signature scheme in Fig. 3 is unforgeable if the structure-preserving signature scheme in Section 3 is existentially unforgeable, $\mathcal{NIZK}_1$ (used by the user to produce π) is sound and $\mathcal{NIZK}_2$ (used by the signer to produce Ω) is zero-knolwedge.*

Proof. We instantiate crs_1 used for $\mathcal{NIZK}_1$ as a binding crs and hence $\mathcal{NIZK}_1$ is perfectly sound, whereas crs_2 is instantiated as a hiding string and hence we can simulate proof Ω. By the security of $\mathcal{NIZK}_2$, we have that an adversary has a negligible advantage in distinguishing a binding string from a hiding one.

We now use an adversary $\mathcal{A}$ against the unforgeability of the blind signature scheme to construct an adversary $\mathcal{B}$ against the unforgeability of the structure-preserving signature scheme.

$\mathcal{B}$ gets the verification key $\mathsf{vk} = (\tilde{X}, \tilde{Y})$ from its game which it forwards to $\mathcal{A}$. $\mathcal{B}$ has access to a sign oracle in its game. To answer a signature query on a message, $\mathcal{B}$ uses the extraction key of $\mathcal{NIZK}_1$ to extract the witness $\tilde{N}$ and forwards $(M, \tilde{N})$ (which is by the soundness of $\mathcal{NIZK}_1$ is valid Diffie-Hellman pair) to its sign oracle to get a signature $\sigma = (A, B, C)$. Now, $\mathcal{B}$ simulates the proof Ω (since it does know the exponent a used in the signature and hence cannot produce the element $\tilde{A}$). $\mathcal{B}$ returns $\big(\sigma = (A, B, C), \Omega\big)$ to $\mathcal{A}$.

Eventually, when $\mathcal{A}$ outputs its $n+1$ message-signature pairs, $\mathcal{B}$ returns the extra pair that it did not query its oracle on as its forgery.

By the existential unforgeability of the signature scheme, we have that this only happens with a negligible probability.

This concludes the proof. □

Lemma 3. *The weakly blind signature scheme in Fig.3 is weakly blind if $\mathcal{NIZK}_1$ is zero-knowledge, $\mathcal{NIZK}_2$ is sound and the DDH assumption holds in group $\mathbb{G}$.*

Proof. We instantiate crs_1 (used for $\mathcal{NIZK}_1$) as a hiding crs and hence we can simulate proof π. By the security of $\mathcal{NIZK}_1$, we have that an adversary has a negligible advantage distinguishing a binding string from a hiding one.

The string crs_2 (used for $\mathcal{NIZK}_2$) is instantiated as a binding string and hence the proof system $\mathcal{NIZK}_2$ is perfectly sound. We now use an adversary $\mathcal{A}$ against the weak blindness of the blind signature to construct an adversary $\mathcal{B}$ that breaks the DDH assumption in group $\mathbb{G}$. $\mathcal{B}$ gets $R = G^r$, $S = G^s$ and $T = G^t$, where $r, s, t \in \mathbb{Z}_p$ are random exponents unknown to $\mathcal{B}$. $\mathcal{B}$ chooses $x, y \leftarrow \mathbb{Z}_p$ and computes $\tilde{X} := \tilde{G}^x$ and $\tilde{Y} := \tilde{G}^y$. It forwards $\mathsf{sk}_{\mathsf{BS}} := (x, y)$ and $\mathsf{vk}_{\mathsf{BS}} := (\tilde{X}, \tilde{Y})$ to $\mathcal{A}$.

$\mathcal{B}$ requests from $\mathcal{A}$ a blind signature on the message R. Note that since $\mathcal{B}$ does not know r, it simulates proof π. $\mathcal{A}$ responds with a signatures $(\sigma = (A, B, C), \Omega)$. Since $\mathcal{NIZK}_2$ is perfectly sound, (A, B, C) is a valid signature on the message $(G^r, \tilde{G}^r)$, i.e. for some $a \in \mathbb{Z}_p$, we have $\sigma = (G^a, G^{ra}, G^{ax} \cdot G^{ray})$. To produce the challenge signature σ_1, $\mathcal{B}$ computes $\sigma_1 = (A^*, B^*, C^*) := (S, T, S^x \cdot T^y)$. $\mathcal{B}$ returns $\Big(\sigma_0 = \big(A, B, C\big), \sigma_1 = \big(A^*, B^*, C^*\big)\Big)$ to $\mathcal{A}$. The advantage of $\mathcal{B}$ in breaking the DDH assumption in $\mathbb{G}$ is the same as that of $\mathcal{A}$ winning the weak blindness game.

We argue now that the challenge signatures given to $\mathcal{A}$ are distributed identically to those $\mathcal{A}$ would get in the blindness game. We have two cases:

- Case $t = rs$ (i.e. a valid DDH tuple) : In this case, we have

  $$\sigma_1 = (A^*, B^*, C^*) := (G^s, G^{rs}, G^{sx} \cdot G^{rsy})$$

  Thus, σ_1 is a valid randomized signature on the message $(G^r, \tilde{G}^r)$. In particular, $\sigma_1 = (A^*, B^*, C^*) := (G^\alpha, G^{\alpha r}, G^{\alpha x} \cdot G^{\alpha r y})$ is a valid signature on $(G^r, \tilde{G}^r)$ for randomness $\alpha = \frac{s}{a}$. Since s is random so is α. This case corresponds to the case $b = 0$ in the blindness game.
- Case t is a random element in $\mathbb{Z}_p$: In this case, we have

  $$\sigma_1 = (A^*, B^*, C^*) := (G^s, G^t, G^{sx} \cdot G^{ty})$$

  Thus, σ_1 is a valid signature on the message $(G^{\frac{t}{s}}, \tilde{G}^{\frac{t}{s}})$. Since t is random so is $\frac{t}{s}$. This case corresponds to the case $b = 1$ in the blindness game.

This concludes the proof. □