Improved Structure Preserving Signatures under Standard Bilinear Assumptions *

Charanjit S. Jutla[1] and Arnab Roy[2]

[1] IBM T. J. Watson Research Center, Yorktown Heights, NY, USA
csjutla@us.ibm.com
[2] Fujitsu Laboratories of America, Sunnyvale, CA, USA
aroy@us.fujitsu.com

Abstract. We show that the recent structure-preserving signature (SPS) scheme of Kiltz, Pan and Wee [CRYPTO 2015], provably secure under the standard bilinear pairings group assumption SXDH, can be improved to have one less group element and one less pairing product equation in the signature verification step. Our improved SPS scheme only requires six group elements (five in one group, and one in the other), and two pairing product equations for verification. The number of pairing product equations is optimal, as it matches a known lower bound of Abe et al [CRYPTO 2011]. The number of group elements in the signature also approaches the known lower bound of four for SXDH assumption. Further, while the earlier scheme had a security reduction which incurred a security loss that is quadratic in number of queries Q, our novel security reduction incurs only a $Q \log Q$ factor loss in security.

Structure-preserving signatures are used pervasively in group signatures, group encryptions, blind signatures, proxy signatures and many other anonymous credential applications. Our work directly leads to improvements in these schemes. Moreover, the improvements are usually of a higher multiplicative factor order, as these constructions use Groth-Sahai NIZK proofs for zero-knowledge verification of pairing-product equations.

We also give our construction under the more general and standard $\mathcal{D}_k$-MDDH (Matrix-DDH) assumption. The signature size in our scheme is $3k+2$ elements in one group, and one element in the other. The number of pairing product equations required for verification is only $2k$, whereas the earlier schemes required at least $2k+1$ equations.

Keywords: Structure preserving signatures, bilinear pairings, SXDH, Matrix-DDH, Groth-Sahai, Cramer-Shoup, QA-NIZK

1 Introduction

The notion of *structure-preserving signatures* (SPS) was introduced in [AFG+10] so that such signatures are compatible with the bilinear-pairings based efficient

non-interactive zero-knowledge (NIZK) proofs of Groth and Sahai [GS08]. The messages, signatures, and verification keys are required to be elements of groups that support efficient bilinear-pairings (*bilinear groups*), and the signature verification consists of just evaluating one or more bilinear-pairing product equations. With the structure of the signature preserved, one can then build many interesting cryptographic primitives and protocols that require (hiding) commitments to such messages and signatures and yet retain the ability to prove properties about these using Groth-Sahai NIZK proofs (GS-NIZK proofs). To list a few, SPS have been used to build blind signatures [AO09,AFG$^+$10], group signatures [AHO10], traceable signatures [ACHO11], group encryption [CLY09], and delegatable credential systems [Fuc11].

The first SPS was introduced by Groth in 2006 even before GS-NIZK proofs were introduced [Gro06]. In the same work Groth also introduced NIZK proofs for algebraic equations over bilinear groups, but since this construction was rather inefficient, it was best viewed as a feasibility study. A variation of the Camenisch-Lysyanskaya signature scheme [CL04] was shown to be an SPS secure against random message attacks [GH08]. Cathalo, Libert and Yung [CLY09] and Fuchsbauer [Fuc09] gave schemes which are efficient when signing a single group element, but their signature size increases linearly in the size of the message. In [AHO10], the authors presented the first constant-size SPS consisting of seven group elements, provable under a non-interactive but dynamic q-type assumption. In [AGHO11], the authors show a three group element SPS scheme provable in the generic asymmetric pairings group model. Interestingly, they also showed that any SPS scheme in asymmetric bilinear groups must require at least three group elements and two pairing product verification equations. They also gave a four group element SPS scheme under a non-interactive but dynamic q-type assumption. In [AGO11], the authors show that any SPS scheme proven secure by a black-box reduction of the standard SXDH assumption in asymmetric bilinear groups must have four group elements.

Recently, Kiltz, Pan and Wee [KPW15] and Libert, Peters and Yung [LPY15] gave efficient SPS schemes under standard bilinear assumptions such as SXDH (Symmetric eXternal Diffie-Hellman assumption) or MDDH (Matrix-DDH assumption). While the latter scheme required ten group elements, the former was even shorter requiring only seven group elements (under SXDH). However, both schemes required three pairing product equations for signature verification, which is sub-optimal. Moreover, the security proofs given for both schemes incurred a quadratic (in the number of signature queries) loss in security.

1.1 Our Contributions

In this work, we show that the scheme of Kiltz, Pan and Wee [KPW15] can be modified to have a signature size of only six group elements. More importantly, the number of pairing product equations required for signature verification is reduced to two, which is optimal by the lower bound of [AGHO11]. Further, we give a security proof that only has a $Q \log Q$ security loss in reduction from standard SXDH or MDDH assumptions.

The ramifications of these improvements are many-fold. First, note that since SPS are used along with commitments, encryptions and GS-NIZK proofs, this can lead to a multiplicative factor improvement in the final cryptographic application. For example, every group element in the SPS that needs a Groth-Sahai commitment leads to a factor two blowup. A CCA2-encryption such as the Cramer-Shoup encryption [CS02] could lead to a factor four or five blowup. Each pairing product equation can lead to up to eight extra group elements in GS-NIZK proofs (under SXDH assumption), and indeed the type of extra pairing product equation in [KPW15] does take eight extra group elements (four in each of the two asymmetric bilinear groups).

Using the methodology of [AHO10,AFG$^+$10], [LPY15] build a dynamic group signature scheme with signature size of 30 group elements in $\mathbb{G}_1$, 14 group elements in $\mathbb{G}_2$ and an integer tag. The improvements presented in this work are directly applicable and should lead to a reduction of at least ten group elements in the size of the signature. Similar improvements are expected in blind signature schemes and other anonymous credentials based schemes.

We also give constructions and security proofs under the more general k-MDDH (matrix-DDH) assumption. Our results and comparison with previous work is summarized in Fig. 1.

As for the improved security reduction, [KPW15] show that if an adaptive chosen-message attack adversary makes at most Q signature queries, then its success probability of forging a signature on a new message is bounded from above by (roughly)

$$Q^2 \cdot \mathrm{ADV}_{\mathrm{DDH}} + Q^2/q$$

where q is the order of the cyclic groups, and $\mathrm{ADV}_{\mathrm{DDH}}$ is the maximum advantage an efficient adversary has in a (decisional Diffie-Hellman) DDH-challenge game in either of the asymmetric bilinear groups. In this work, we show that the success probability of forging a signature is at most (roughly)

$$Q \cdot \log Q \cdot \mathrm{ADV}_{\mathrm{DDH}} + Q^2/q$$

Since, by Pollard's Rho method [Pol78], $\mathrm{ADV}_{\mathrm{DDH}}$ is at least $1/\sqrt{q}$, the first term in both of the above success probabilities is dominant. Thus, for the same security guarantee, and for large number of signatures (which should be expected for group signatures and other such anonymous credential applications), the earlier schemes would require almost twice the number of bits in representation of the group elements.

1.2 Our Techniques

The underlying idea in the SPS schemes of both [KPW15] and [LPY15], and our scheme is to hide a secret using a CCA2 encryption scheme, and in particular the Cramer-Shoup encryption [CS02], and prove in zero-knowledge that the signer knows the secret encrypted in the ciphertext. This methodology of building signature schemes was already described in [CCS09] (also, see a refinement

Table 1. Comparison with existing unbounded security SPS schemes with table adapted from [KPW15]. (n_1, n_2) denotes n_1 $\mathbb{G}_1$ elements and n_2 $\mathbb{G}_2$ elements. The table gives message, signature and public key sizes and finally the number of pairing product equations needed for verification. $RE(\mathcal{D}_k)$ is the number of group elements needed for representing a sample from $\mathcal{D}_k$; $\overline{RE}(\mathcal{D}_k)$ is the same for all but the last row of a sample. For k-Linear assumption these are $k+1$ and k respectively.

	Assumption	$\lvert m\rvert$	$\lvert\sigma\rvert$	$\lvert pk\rvert$	#PPEs
[AGHO11]	Interactive (Generic)	(n_1, n_2)	$(2, 1)$	$n_1 + n_2 + 2$	2
[AGHO11]	Non-interactive (Generic)	(n_1, n_2)	$(3, 3)$	$n_1 + n_2 + 2$	2
[AGHO11]	Non-Interactive (Generic)	$(n_1, 0)$	$(3, 1)$	$n_1 + 2$	2
[ACD+12]	SXDH, XDLIN	$(n_1, 0)$	$(7, 4)$	$20 + n_1$	4
[ACD+12]	SXDH, XDLIN	(n_1, n_2)	$(8, 6)$	$22 + n_1 + n_2$	5
[ADK+13]	2-Lin ($\mathbb{G}_1 = \mathbb{G}_2$)	n	14	$22 + n$	7
[AFG+10]	q-SFP	$(n_1, 0)$	$(5, 2)$	$13 + n_1$	2
[LPY15]	SXDH, XDLIN	$(n_1, 0)$	$(9, 1)$	$2n_1 + 21$	5
[KPW15]	D_k – MDDH	(n_1, n_2)	$(4k + 3, k + 2)$	$(n_1 + n_2 + 3k + 3)k + 2RE(\mathcal{D}_k)$	$3k + 1$
[KPW15]	$\mathcal{D}_k$ – MDDH	$(n_1, 0)$	$(3k + 3, 1)$	$(n_1 + 2k + 3)k + RE(\mathcal{D}_k)$	$2k + 1$
This paper	$\mathcal{D}_k$ – MDDH	$(n_1, 0)$	$(3k + 2, 1)$	$(n_1 + 2k + 3)k + \overline{RE}(\mathcal{D}_k)$	$2k$

of this method in [JR13]). However, as is well-known, the Cramer-Shoup encryption scheme requires exponentiation with a tag which is computed from other elements in the ciphertext in a 1-1 fashion. This enforces the tag to be different if the ciphertext is changed in any way. However, this clearly is not structure-preserving, as the 1-1 mapping is required to map from the group elements to another group $\mathbb{Z}_q$, where q is the order of the bilinear groups.

In [KPW15] and [LPY15], the tag is instead chosen afresh at random (i.e., independent of other elements in the ciphertext), and its representation in the bilinear group is given as part of the signature. The tag is also used in the aforementioned exponentiation (in fact, more than one), and simple bilinear tests can check that these values are consistent. To get a better understanding, we now give some specific details. Let k be the secret of the signer. To create the signature, it generates a Cramer-Shoup encryption, by picking r at random, and setting

$$\rho = g_1^r, \hat{\rho} = (g_1^b)^r, \gamma = g_1^k \cdot (g_1^d)^r \cdot (g_1^e)^{t \cdot r}$$

where t is the tag, and g_1^b, g_1^d, g_1^e are part of the public key. In SPS, since t is chosen afresh, the signer also gives $\psi = g_1^{t \cdot r}$ and $\tau = g_2^t$. Note that τ is in group $\mathbb{G}_2$, whereas all other elements are in group $\mathbb{G}_2$. The consistency of ρ, ψ and τ is easily checked by a bilinear pairing product equation, i.e., $\mathsf{e}(\rho, \tau) = \mathsf{e}(\psi, g_2)$.

If one were to follow the methodology of [CCS09], the signer also gives a NIZK proof π that $\rho, \hat{\rho}, \psi$ and γ are consistent with the public key, and some public information about k. However, with the quasi-adaptive computationally-sound NIZK proofs (QA-NIZK) of [JR13], one can give a QA-NIZK proof that these elements are in an affine span of the underlying linear subspace language, with the verifier CRS independent of the affine component (i.e. g_1^k).

The scheme in [KPW15] (also [LPY15]) also gives an additional element $\hat{\psi} = (g_1^b)^{t \cdot r}$, and the signature verification requires another consistency check, i.e. $\mathsf{e}(\hat{\rho}, \tau) = \mathsf{e}(\hat{\psi}, g_2)$. The main reason for this additional verification is that [KPW15] does not follow the above methodology for the security proof, and instead uses

a core computational lemma which was used to give an unbounded-simulation sound QA-NIZK scheme [KW15]. As mentioned earlier, it suffices to use a (non simulation-sound) NIZK as long as one uses a CCA2 encryption like Cramer-Shoup (which in itself is just a one-time simulation-sound method). Now, readers familiar with Cramer-Shoup encryption will recall that the main idea there is the ability for the simulator to use an alternate decryption. However, in signature schemes, as opposed to Cramer-Shoup encryption, there is no real decryption, but just a verification of the signature using private trapdoor keys. This can also be done efficiently using the bilinear pairing available, and this is the reason why a single additional test of the relationship between ψ, ρ and τ suffices. More details can be found in Section 3.1.

1.3 Recursive Complexity-Leveraging for Improved Security Reduction

For improving the security reduction, we first note that [KPW15] requires a complexity-leveraging technique, because the simulator of the challenger in the SPS security game must guess a query index (the one for which the adversary may use the same tag), and then try to simulate signatures only for indices other than this guess. However, since the adversary is adaptive, this guess is only correct with probability $1/Q$, where Q is the maximum number of queries the adversary makes.

We follow a recursive approach, where the simulator goes through Q hybrid games. In the first $Q/2$ hybrid games, the simulator guesses a set Z of size $Q/2$, and then simulates queries outside this set. Now, the simulator's correct guess probability that the adversary's tag will match a tag in query from set Z is much higher, i.e., $1/2$. From the $Q/2$-th hybrid onwards, we show that the simulator can switch to another sequence of hybrid games, where now the simulator guesses a set Z of size $Q/4$, and so forth inductively. The penalty in the security reduction in this switch is only a factor of two. Note that we are paying a penalty of factor 2^m for only the last $Q/2^{m-1}$ hybrids, and this leads to a reduction with only a $Q \log Q$ security loss. We expect our novel complexity-leveraging technique to be more widely applicable, and of independent interest.

2 Preliminaries

We will consider cyclic groups $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ of prime order q, with an efficient bilinear map $\mathsf{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$. Group elements $\mathbf{g}_1$ and $\mathbf{g}_2$ will typically denote generators of the group $\mathbb{G}_1$ and $\mathbb{G}_2$ respectively. Following [EHK+13], we will use the notations $[a]_1, [a]_2$ and $[a]_T$ to denote $a\mathbf{g}_1, a\mathbf{g}_2$, and $a \cdot \mathsf{e}(\mathbf{g}_1, \mathbf{g}_2)$ respectively and use additive notations for group operations. When talking about a general group $\mathbb{G}$ with generator $\mathbf{g}$, we will just use the notation $[a]$ to denote $a\mathbf{g}$. The notation generalizes to vectors and matrices in a natural component-wise way.

For two vector or matrices A and B, we will denote the product $A^\top B$ as $A \cdot B$. The pairing product $\mathsf{e}([A]_1, [B]_2)$ evaluates to the matrix product $[AB]_T$

in the target group with pairing as multiplication and target group operation as addition.

We recall the *Matrix Decisional Diffie-Hellman* or MDDH assumptions from [EHK+13]. A matrix distribution $\mathcal{D}_{l,k}$, where $l > k$, is defined to be an efficiently samplable distribution on $\mathbb{Z}_q^{l\times k}$ which is full-ranked with overwhelming probability. The *$\mathcal{D}_{l,k}$-MDDH assumption* in group $\mathbb{G}$ states that with samples $\mathbf{A} \leftarrow \mathcal{D}_{l,k}, \mathbf{s} \leftarrow \mathbb{Z}_q^k$ and $\mathbf{s}' \leftarrow \mathbb{Z}_q^l$, the tuple $([\mathbf{A}], [\mathbf{As}])$ is computationally indistinguishable from $([\mathbf{A}], [\mathbf{s}'])$. A matrix distribution $\mathcal{D}_{k+1,k}$ is simply denoted by $\mathcal{D}_k$.

2.1 Quasi-Adaptive NIZK Proofs

A witness relation is a binary relation on pairs of inputs, the first called a word and the second called a witness. Each witness relation R defines a corresponding language L which is the set of all words x for which there exists a witness w, such that $R(x, w)$ holds.

We will consider Quasi-Adaptive NIZK proofs [JR13] for a probability distribution $\mathcal{D}$ on a collection of (witness-) relations $\mathcal{R} = \{R_\rho\}$ (with corresponding languages L_ρ). Recall that in a quasi-adaptive NIZK, the CRS can be set after the language parameter has been chosen according to $\mathcal{D}$. Please refer to [JR13] for detailed definitions.

For our SPS construction we will also need a property called true-simulation-soundness and an extension of QA-NIZKs called strong split-CRS QA-NIZK. We also recall the definitions of these concepts below.

Definition 1 (QA-NIZK [JR13]). *We call a tuple of efficient algorithms* $(\mathsf{pargen}, \mathsf{crsgen}, \mathsf{prover}, \mathsf{ver})$ *a* quasi-adaptive non-interactive zero- knowledge *(QA-NIZK) proof system for witness-relations $\mathcal{R}_\lambda = \{R_\rho\}$ with parameters sampled from a distribution $\mathcal{D}$ over associated parameter language $\mathcal{L}_{par}$, if there exist simulators* crssim *and* sim *such that for all non-uniform PPT adversaries $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$, we have (in all of the following probabilistic experiments, the experiment starts by setting λ as $\lambda \leftarrow \mathsf{pargen}(1^m)$, and choosing ρ as $\rho \leftarrow \mathcal{D}_\lambda$):*

Quasi-Adaptive Completeness:

$$\Pr\left[\begin{array}{l}\mathsf{CRS} \leftarrow \mathsf{crsgen}(\lambda, \rho)\\ (x, w) \leftarrow \mathcal{A}_1(\mathsf{CRS}, \rho)\\ \pi \leftarrow \mathsf{prover}(\mathsf{CRS}, x, w)\end{array} : \begin{array}{c}\mathsf{ver}(\mathsf{CRS}, x, \pi) = 1 \textbf{ if}\\ R_\rho(x, w)\end{array}\right] = 1$$

Quasi-Adaptive Soundness:

$$\Pr\left[\begin{array}{l}\mathsf{CRS} \leftarrow \mathsf{crsgen}(\lambda, \rho)\\ (x, \pi) \leftarrow \mathcal{A}_2(\mathsf{CRS}, \rho)\end{array} : \begin{array}{c}x \notin L_\rho \textbf{ and}\\ \mathsf{ver}(\mathsf{CRS}, x, \pi) = 1]\end{array}\right] \approx 0$$

Quasi-Adaptive Zero-Knowledge:

$$\Pr\left[\mathsf{CRS} \leftarrow \mathsf{crsgen}(\lambda, \rho) \ : \ \mathcal{A}_3^{\mathsf{prover}(\mathsf{CRS},\cdot,\cdot)}(\mathsf{CRS}, \rho) = 1\right]$$
$$\approx$$
$$\Pr\left[(\mathsf{CRS}, \mathsf{trap}) \leftarrow \mathsf{crssim}(\lambda, \rho) \ : \ \mathcal{A}_3^{\mathsf{sim}^*(\mathsf{CRS},\mathsf{trap},\cdot,\cdot)}(\mathsf{CRS}, \rho) = 1\right],$$

where $\mathsf{sim}^*(\mathsf{CRS}, \mathsf{trap}, x, w) = \mathsf{sim}(\mathsf{CRS}, \mathsf{trap}, x)$ *for* $(x, w) \in R_\rho$ *and both oracles (i.e.* prover *and* sim^**) output failure if* $(x, w) \notin R_\rho$*.*

Definition 2 (True-Simulation-Sound [Har11]). *A QA-NIZK is called* **true -simulation-sound** *if the verifier is sound even when an adaptive adversary has access to simulated proofs on language members. More precisely, for all PPT* $\mathcal{A}$*,*

$$\Pr\left[\begin{array}{l|c}(\mathsf{CRS}, \mathsf{trap}) \leftarrow \mathsf{crssim}(\lambda, \rho) & x \notin L_\rho \textbf{ and} \\ (x, \pi) \leftarrow \mathcal{A}^{\mathsf{sim}(\mathsf{CRS}, \mathsf{trap}, \cdot, \cdot)}(\mathsf{CRS}, \rho) & \mathsf{ver}(\mathsf{CRS}, x, \pi) = 1\end{array}\right] \approx 0,$$

where the experiment aborts if the oracle is called with some $x \notin L_\rho$*.*

Definition 3 (Strong Split-CRS QA-NIZK [JR13]). *We call a tuple of efficient algorithms* $(\mathsf{pargen}, \mathsf{crsgen}_v, \mathsf{crsgen}_p, \mathsf{prover}, \mathsf{ver})$ *a* **strong split-CRS QA-NIZK** *proof system for an ensemble of distributions* $\{\mathcal{D}_\lambda\}$ *on collection of witness-relations* $\mathcal{R}_\lambda = \{R_\rho\}$ *with associated parameter language* $\mathcal{L}_{par}$ *if there exists probabilistic polynomial time simulators* $(\mathsf{crssim}_v, \mathsf{crssim}_p, \mathsf{sim})$*, such that for all non-uniform PPT adversaries* $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$*, and* $\lambda \leftarrow \mathsf{pargen}(1^m)$*, we have:*

Quasi-Adaptive Completeness:

$$\Pr\left[\begin{array}{l|c}(\mathsf{CRS}_v, st) \leftarrow \mathsf{crsgen}_v(\lambda),\ \rho \leftarrow \mathcal{D}_\lambda & \\ \mathsf{CRS}_p \leftarrow \mathsf{crsgen}_p(\lambda, \rho, st) & \mathsf{ver}(\mathsf{CRS}_v, x, \pi) = 1 \textbf{ if} \\ (x, w) \leftarrow \mathcal{A}_1(\lambda, \mathsf{CRS}_v, \mathsf{CRS}_p, \rho) & R_\rho(x, w) \\ \pi \leftarrow \mathsf{prover}(\mathsf{CRS}_p, x, w) & \end{array}\right] = 1$$

Quasi-Adaptive Soundness:

$$\Pr\left[\begin{array}{l|c}(\mathsf{CRS}_v, st) \leftarrow \mathsf{crsgen}_v(\lambda),\ \rho \leftarrow \mathcal{D}_\lambda & \mathsf{ver}(\mathsf{CRS}_v, x, \pi) = 1 \textbf{ and} \\ \mathsf{CRS}_p \leftarrow \mathsf{crsgen}_p(\lambda, \rho, st) & \textbf{not } (\exists w : R_\rho(x, w)) \\ (x, \pi) \leftarrow \mathcal{A}_2(\lambda, \mathsf{CRS}_v, \mathsf{CRS}_p, \rho) & \end{array}\right] \approx 0$$

Quasi-Adaptive Zero-Knowledge:

$$\Pr\left[\begin{array}{l|c}(\mathsf{CRS}_v, st) \leftarrow \mathsf{crsgen}_v(\lambda) & \\ \rho \leftarrow \mathcal{D}_\lambda & \mathcal{A}_3^{\mathsf{prover}(\mathsf{CRS}_p, \cdot, \cdot)}(\lambda, \mathsf{CRS}_v, \mathsf{CRS}_p, \rho) = 1 \\ \mathsf{CRS}_p \leftarrow \mathsf{crsgen}_p(\lambda, \rho, st) & \end{array}\right]$$

$$\approx$$

$$\Pr\left[\begin{array}{l|c}(\mathsf{CRS}_v, \mathsf{trap}, st) \leftarrow \mathsf{crssim}_v(\lambda) & \\ \rho \leftarrow \mathcal{D}_\lambda & \mathcal{A}_3^{\mathsf{sim}^*(\mathsf{trap}, \cdot, \cdot)}(\lambda, \mathsf{CRS}_v, \mathsf{CRS}_p, \rho) = 1 \\ \mathsf{CRS}_p \leftarrow \mathsf{crssim}_p(\lambda, \rho, st) & \end{array}\right],$$

where $\mathsf{sim}^*(\mathsf{trap}, x, w) = \mathsf{sim}(\mathsf{trap}, x)$ *for* $(x, w) \in R_\rho$ *and both oracles (i.e.* prover *and* sim^**) output failure if* $(x, w) \notin R_\rho$*.*

2.2 Strong Split-CRS QA-NIZK for Affine Languages

We now describe a strong split-CRS QA-NIZK $(\mathsf{pargen}, \mathsf{crsgen}_v, \mathsf{crsgen}_p, \mathsf{prover}, \mathsf{ver})$ for affine linear subspace languages $\{L_{[\mathbf{M}]_1, [\mathbf{a}]_1}\}$, consisting of words of the

form $[\mathbf{M}\mathbf{x}+\mathbf{a}]_1$, with parameters sampled from a robust and efficiently witness-samplable distribution $\mathcal{D}$ over the associated parameter language $\mathcal{L}_{\mathrm{par}}$ and with soundness under a $\mathcal{D}_k$-MDDH assumption. Robustness means that the top square matrix of $\mathbf{M}$ is full-ranked with overwhelming probability. The construction is essentially the one of [JR13] adapted to the framework of [KW15].

Algorithm crsgen_v: The algorithm crsgen_v samples a matrix $\mathbf{K} \leftarrow \mathbb{Z}_q^{n\times k}$, a vector $\mathbf{k} \leftarrow \mathbb{Z}_q^k$ and a matrix $\mathbf{A}^{(k+1)\times k}$ from the MDDH distribution $\mathcal{D}_k$. Let $\bar{\mathbf{A}}$ be the top $k \times k$ square matrix of $\mathbf{A}$. Then it computes:

$$\mathsf{CRS}_v := \left([\mathbf{C}_0]_2^{n\times k} = [\mathbf{K}\bar{\mathbf{A}}]_2, \quad [\mathbf{C}_1]_2^{1\times k} = [\mathbf{k}\cdot\bar{\mathbf{A}}]_2, \quad [\bar{\mathbf{A}}]_2^{k\times k}\right)$$

and state $st = (\mathbf{K},\ \mathbf{k})$.
Algorithm crsgen_p: Let $\rho = ([\mathbf{M}]_1^{n\times t},\ [\mathbf{a}]_1^{n\times 1})$ be the language parameter supplied to crsgen_p and $st = (\mathbf{K},\ \mathbf{k})$ be the state transmitted by crsgen_v. Then it computes:

$$\mathsf{CRS}_p := \left([\mathbf{P}_0]_1^{t\times k} = [\mathbf{M}^\top\mathbf{K}]_1, \quad [\mathbf{P}_1]_1^{1\times k} = [\mathbf{a}\cdot\mathbf{K}+\mathbf{k}^\top]_1\right)$$

Prover prover**:** Given candidate $\mathbf{y} = [\mathbf{M}\mathbf{x}+\mathbf{a}]_1$ with witness vector $\mathbf{x}^{t\times 1}$, the prover generates the following proof consisting of k elements in $\mathbb{G}_1$:

$$\boldsymbol{\pi} := \mathbf{x}\cdot[\mathbf{P}_0]_1 + [\mathbf{P}_1]_1$$

Verifier ver**:** Given candidate $\mathbf{y}$, and proof $\boldsymbol{\pi}$, compute:

$$\mathsf{e}(\mathbf{y}^\top, [\mathbf{C}_0]_2) + \mathsf{e}([1]_1, [\mathbf{C}_1]_2) \stackrel{?}{=} \mathsf{e}(\boldsymbol{\pi}, [\bar{\mathbf{A}}]_2)$$

Simulators $\mathsf{crssim}_v, \mathsf{crssim}_p$ **and** sim: The algorithms crssim_v and crssim_p are identical to crsgen_v and crsgen_p respectively, except that crsgen_v also outputs $\mathsf{trap} := (\mathbf{K},\ [\mathbf{k}]_1)$. The proof simulator sim takes candidate $\mathbf{y}$ and trapdoor $(\mathbf{K},\ [\mathbf{k}]_1)$ and outputs:

$$\boldsymbol{\pi} := \mathbf{y}\cdot\mathbf{K} + [\mathbf{k}^\top]_1$$

Theorem 1. *The above algorithms* $(\mathsf{pargen}, \mathsf{crsgen}_v, \mathsf{crsgen}_p, \mathsf{prover}, \mathsf{ver})$ *constitute a true-simulation-sound strong split-CRS QA-NIZK proof system for affine languages* $\{L_{[\mathbf{M}]_1,[\mathbf{a}]_1}\}$ *with parameters* $([\mathbf{M}]_1,\ [\mathbf{a}]_1)$ *sampled from a robust and efficiently witness-samplable distribution* $\mathcal{D}$ *over the associated parameter language* $\mathcal{L}_{par}$*, given any group generation algorithm for which the* $\mathcal{D}_k$-MDDH *assumption holds for group* $\mathbb{G}_2$*.*

2.3 Projective Hash Proof System.

For a language L, let X be a superset of L and let $H = (H_k)_{k\in K}$ be a collection of (hash) functions indexed by K with domain X and range another set Π. The hash function family is generalized to a notion of *projective hash function family* if there is a set S of projection keys, and a projection map $\alpha :\ K \to S$,

and further the action of H_k on subset L of X is completely determined by the projection key $\alpha(k)$. Finally, the projective hash function family is defined to be ϵ-**universal$_2$** if for all $s \in S$, $x, x^* \in X$, and $\pi, \pi^* \in \Pi$ with $x \notin L \cup \{x^*\}$, the following holds:

$$\Pr[H_k(x) = \pi \mid H_k(x^*) = \pi^* \wedge \alpha(k) = s] \leq \epsilon.$$

A projective hash function family is called ϵ-**smooth** if for all $x \in X \setminus L$, the statistical difference between the following two distributions is ϵ: sample k uniformly from K and π' uniformly from Π; the first distribution is given by the pair $(\alpha(k), H_k(x))$ and the second by the pair $(\alpha(k), \pi')$. For languages defined by a witness-relation R, the projective hash proof family constitutes a *projective hash proof system* (PHPS) if α, H_k, and another *public evaluation function* $\hat{H}$ that computes H_k on $x \in L$, given a witness of x and *only* the projection key $\alpha(k)$, are all efficiently computable. An efficient algorithm for sampling the key $k \in K$ is also assumed.

The above notions can also incorporate labels. In an *extended PHPS*, the hash functions take an additional input called *label*. The public evaluation algorithm also takes this label. All the above notions are now required to hold for each possible value of label. The extended PHPS is now defined to be ϵ-**universal$_2$** is for all $s \in S$, $x, x^* \in X$, all labels $\mathtt{l}$ and $\mathtt{l}^*$, and $\pi, \pi^* \in \Pi$ with $x \notin L$ and $(x, \mathtt{l}) \neq (x^*, \mathtt{l}^*)$, the following holds:

$$\Pr[H_k(x, \mathtt{l}) = \pi \mid H_k(x^*, \mathtt{l}^*) = \pi^* \wedge \alpha(k) = s] \leq \epsilon.$$

Since we are interested in distributions of languages, we extend the above definition to distribution of languages. So consider a parametrized class of languages $\{L_\rho\}_\rho$ with the parameters coming from an associated parameter language $\mathcal{L}_{\text{par}}$. Assume that all the languages in this collection are subsets of X. Let H as above be a collection of hash functions from X to Π. We say that the hash family is a projective hash family if for all L_ρ, the action of H_k on L_ρ is determined by $\alpha(k)$. Similarly, the hash family is ϵ-universal$_2$ (ϵ-smooth) for $\{L_\rho\}_\rho$ if for all languages L_ρ the ϵ-universal$_2$ (resp. ϵ-smooth) property holds.

2.4 Structure-Preserving Signatures

Definition 4 (Structure-preserving signature). *A structure-preserving signature scheme SPS is defined as a triple of probabilistic polynomial time (PPT) algorithms* $SPS = (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Verify})$*:*

- *The probabilistic key generation algorithm* $\mathsf{Gen}(par)$ *returns the public/secret key* (pk, sk)*, where* $pk \in \mathbb{G}^{n_{pk}}$ *for some* $n_{pk} \in poly(\lambda)$*. We assume that pk implicitly defines a message space* $M := \mathbb{G}^n$ *for some* $n \in poly(\lambda)$*.*
- *The probabilistic signing algorithm* $\mathsf{Sign}(sk, [m])$ *returns a signature* $\sigma \in \mathbb{G}^{n_\sigma}$ *for* $n_\sigma \in poly(\lambda)$*.*
- *The deterministic verification algorithm* $\mathsf{Verify}(pk, [m], \sigma)$ *only consists of pairing product equations and returns 1 (accept) or 0 (reject).*

Perfect correctness holds if for all $(pk, sk) \leftarrow \mathsf{Gen}(par)$ *and all messages* $[m] \in M$ *and all* $\sigma \leftarrow \mathsf{Sign}(sk, [m])$ *we have* $\mathsf{Verify}(pk, [m], \sigma) = 1$.

Definition 5 (Unforgeability against chosen message attack). *To an adversary A and scheme SPS we associate the advantage function:*

$$\mathrm{ADV}^{CMA}_{SPS}(A) := \Pr\left[\begin{matrix}(pk, sk) \leftarrow \mathsf{Gen}(par) \\ ([m^*], \sigma^*) \leftarrow A^{SignO(\cdot)}(pk)\end{matrix} \;:\; \begin{matrix}[m^*] \notin Q_{msg} \textbf{ and} \\ \mathsf{Verify}(pk, [m^*], \sigma^*) = 1\end{matrix}\right]$$

where SignO([m]) runs $\sigma \leftarrow \mathsf{Sign}(sk, [m])$*, adds the vector* $[m]$ *to* Q_{msg} *(initialized with* $\emptyset$*) and returns* σ *to A. An SPS is said to be (unbounded) CMA-secure if for all PPT adversaries A,* $\mathrm{ADV}^{CMA}_{SPS}(A)$ *is negligible.*

3 SPS Construction

Our SPS construction for a general $\mathcal{D}_k$-MDDH assumption is given in Figure 1. We also give the instantiation of this SPS for the Symmetric eXternal Diffie-Hellman Assumption (SXDH) assumption in Figure 2. The construction assumes groups $\mathbb{G}_1$ and $\mathbb{G}_2$ and a target group $\mathbb{G}_T$ with an efficient bilinear pairing e from $\mathbb{G}_1 \times \mathbb{G}_2$ to $\mathbb{G}_T$.

3.1 Security of the SPS Scheme

In this section we state and prove the security of the scheme SPS_{MDDH} described in Figure 1. The proof is similar to the proof of CCA2 secure encryption scheme of Cramer and Shoup [CS02], where tag-based universal$_2$ projective hash proofs were introduced. The main difference is that the tag in structure preserving signatures (SPS) cannot be generated by hashing some of the group elements. The tag is therefore generated randomly and independently in SPS. The adversary may then try to forge a signature by setting the tag to be the same as the tag in one of the signatures it obtained earlier, and choosing other elements in the forged signature by modifying and combining elements of various signatures it obtained. In contrast, in Cramer-Shoup encryption, any change in other group elements of a ciphertext forces the tag to be different from all earlier ciphertext tags. To circumvent this problem in SPS, the tag t is provided as both $[t]_2$ and $[t\mathbf{r}]_1$, where $[\mathbf{r}]_1$ is randomness introduced as part of the signature. The validity of this relation can be checked publicly and efficiently using asymmetric bilinear pairing. Intuitively, this disallows the adversary to modify and combine elements from various signatures. It is now forced to modify at most one signature, while keeping the tag the same as in that signature. However, an affine secret component $[k_0]_1$ in the SPS signature, which is issued encrypted under an CCA2 encryption scheme and verified using a publicly verifiable QA-NIZK for affine languages, then disallows even this kind of forgery.

Gen $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathsf{e}, [1]_1, [1]_2, n, \mathcal{D}_k)$:

Let $\mathcal{D}$ be a distribution on $(\tilde{\mathbf{M}}, \tilde{\mathbf{a}})$ defined as follows :

Sample $\mathbf{B}^{(k+1)\times k} \leftarrow \mathcal{D}_k$ and $(k_0,\ \mathbf{k},\ \mathbf{d},\ \mathbf{e}) \leftarrow \mathbb{Z}_q \times \mathbb{Z}_q^n \times \mathbb{Z}_q^k \times \mathbb{Z}_q^k$.

Let $\tilde{\mathbf{M}} := \left(\begin{array}{c|c|c} \mathrm{I}^{n\times n} & \mathbf{0}^{n\times k} & \mathbf{0}^{n\times k} \\ \hline \mathbf{0}^{(k+1)\times n} & \mathbf{B} & \mathbf{0}^{(k+1)\times k} \\ \hline \mathbf{0}^{k\times n} & \mathbf{0}^{k\times k} & \overline{\mathbf{B}} \\ \hline \mathbf{k}^\top & \mathbf{d}\cdot\overline{\mathbf{B}} & \mathbf{e}\cdot\overline{\mathbf{B}} \end{array}\right) \in \mathbb{Z}_q^{(n+2k+2)\times(n+2k)}$

and $\tilde{\mathbf{a}} := \begin{pmatrix} \mathbf{0}^{(n+2k+1)\times 1} \\ k_0 \end{pmatrix} \in \mathbb{Z}_q^{n+2k+2}$.

Let Π be a strong split-CRS QA-NIZK for

$L_{\tilde{\mathbf{M}},\tilde{\mathbf{a}}} = \{[\tilde{\mathbf{M}}\mathbf{x} + \tilde{\mathbf{a}}]_1 \mid \mathbf{x} \in \mathbb{Z}_q^{n+2k}\},$ with $(\tilde{\mathbf{M}}, \tilde{\mathbf{a}}) \leftarrow \mathcal{D}$

which is true-simulation-sound under the $\mathcal{D}_k$-MDDH assumption in $\mathbb{G}_2$.

Sample $(\mathsf{CRS}_v, \mathsf{trap}, st) \leftarrow \Pi.\mathsf{crssim}_v$ and $(\mathbf{M}, \mathbf{a}) \leftarrow \mathcal{D}$

Let $pk := \mathsf{CRS}_v$ and $sk := (\mathbf{M},\ \mathbf{a},\ \mathsf{trap})$

Return (pk, sk)

Sign $(sk = (\mathbf{M}, \mathbf{a}, \mathsf{trap}),\ \boldsymbol{\mu} \in \mathbb{G}_1^n)$:

Sample $\mathbf{r} \leftarrow \mathbb{Z}_q^k$ and $\mathrm{TAG} \leftarrow \mathbb{Z}_q$

Let $(\boldsymbol{\mu}, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma) := \mathbf{M}\begin{pmatrix} \boldsymbol{\mu} \\ [\mathbf{r}]_1 \\ [\mathrm{TAG}\cdot\mathbf{r}]_1 \end{pmatrix} + [\mathbf{a}]_1 \ \in \mathbb{G}_1^n \times \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_1^k \times \mathbb{G}_1$

Let $\boldsymbol{\pi} := \Pi.\mathsf{sim}(\mathsf{trap},\ (\boldsymbol{\mu}, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma))$ and $\tau := [\mathrm{TAG}]_2$

Return $(\boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma, \tau, \boldsymbol{\pi}) \in \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_2 \times \mathbb{G}_1^k$

Verify $(pk = \mathsf{CRS}_v,\ \boldsymbol{\mu},\ \sigma = (\boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma, \tau, \boldsymbol{\pi}))$:

Return $\Pi.\mathsf{ver}(\mathsf{CRS}_v,\ (\boldsymbol{\mu}, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma),\ \boldsymbol{\pi})$ **and** $\mathsf{e}(\boldsymbol{\rho}, \tau) \stackrel{?}{=} \mathsf{e}(\boldsymbol{\psi}, [1]_2)$

Fig. 1. Structure Preserving Signature SPS_{MDDH}

Gen $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathsf{e}, [1]_1, [1]_2, n)$: Sample $b,\ k_0,\ d$ and e uniformly from $\mathbb{Z}_q$ and $\mathbf{k}$ uniformly from $\mathbb{Z}_q^n$. Define the language L of tuples $(\boldsymbol{\mu}, \rho, \hat{\rho}, \psi, \gamma) \in \mathbb{G}^{n+4}$, such that there exists $(\mathbf{m}, r, r') \in \mathbb{Z}_q^{n+2}$, such that:

$$\boldsymbol{\mu} = [\mathbf{m}]_1,\ \rho = [r]_1,\ \hat{\rho} = [br]_1,\ \psi = [r']_1,\ \gamma = [k_0 + \mathbf{k} \cdot \mathbf{m} + dr + er']_1$$

Let Π be a strong split-CRS QA-NIZK for the affine language L, which is true-simulation-sound under the DDH assumption in $\mathbb{G}_2$. Let the simulation CRS generator $\Pi.\mathsf{crssim}_v$ output $(\mathsf{CRS}_v, \mathsf{trap}, st)$. Set $pk := \mathsf{CRS}_v$ and $sk := (b,\ k_0,\ \mathbf{k},\ d,\ e,\ \mathsf{trap})$, and return (pk, sk).

Sign $(sk = (b, k_0, \mathbf{k}, d, e, \mathsf{trap}),\ \boldsymbol{\mu} \in \mathbb{G}_1^n)$: Sample r and TAG uniformly from $\mathbb{Z}_q$. Let:

$$\rho = [r]_1,\ \hat{\rho} = [br]_1,\ \psi = [\text{TAG} \cdot r]_1,\ \gamma = \mathbf{k} \cdot \boldsymbol{\mu} + [k_0 + dr + \text{TAG} \cdot er]_1$$

Let $\pi := \Pi.\mathsf{sim}(\mathsf{trap},\ (\boldsymbol{\mu}, \rho, \hat{\rho}, \psi, \gamma))$ and $\tau := [\text{TAG}]_2$. Return:

$$\sigma := (\rho, \hat{\rho}, \psi, \gamma, \tau, \pi) \in \mathbb{G}_1^4 \times \mathbb{G}_2 \times \mathbb{G}_1.$$

Verify $(pk = \mathsf{CRS}_v,\ \boldsymbol{\mu},\ \sigma = (\rho, \hat{\rho}, \psi, \gamma, \tau, \pi))$: Return the boolean:

$$\Pi.\mathsf{ver}(\mathsf{CRS}_v, (\boldsymbol{\mu}, \rho, \hat{\rho}, \psi, \gamma), \pi) \textbf{ and } \mathsf{e}(\rho, \tau) \stackrel{?}{=} \mathsf{e}(\psi, [1]_2).$$

Fig. 2. Structure Preserving Signature SPS_{SXDH}

Theorem 2. *For any efficient adversary $\mathcal{A}$, which makes at most Q signature queries before attempting a forgery, its probability of success in the EUF-CMA game against the scheme SPS_{MDDH} is at most*

$$\text{ADV}^{TSS}_{\Pi} + Q^2 \cdot \left(\text{ADV}_{\mathcal{D}_k\text{-MDDH}} + \frac{3}{2q}\right) + \frac{Q}{q} + \frac{1}{q}$$

Proof. We go through a sequence of Games $\mathbf{G_0}$ to $\mathbf{G_6}$ which are described below and summarized in Figure 3. In the following, $\text{Prob}_i[X]$ will denote probability of predicate X holding in probability space defined in game $\mathbf{G_i}$.

Game $\mathbf{G_0}$: Given setup parameters $(q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \mathsf{e}, [1]_1, [1]_2, n, \mathcal{D}_k)$, the challenger $\mathcal{C}$ initializes a list $\mathcal{M}$ to empty, generates $(\mathsf{CRS}_v, \mathsf{trap}, st) \leftarrow \Pi.\mathsf{crssim}_v$, and then samples $\mathbf{B}^{(k+1)\times k} \leftarrow \mathcal{D}_k$ and $(k_0,\ \mathbf{k},\ \mathbf{d},\ \mathbf{e}) \leftarrow \mathbb{Z}_q \times \mathbb{Z}_q^n \times \mathbb{Z}_q^k \times \mathbb{Z}_q^k$.

Then it sends the setup parameters and CRS_v to adversary $\mathcal{A}$ as public key. For $i \in [1..Q]$, $\mathcal{A}$ adaptively requests signature on $\boldsymbol{\mu}_i$ $(\in \mathbb{G}_1^n)$. The challenger $\mathcal{C}$ generates signature σ_i by first sampling $(\mathbf{r}, \text{TAG}) \leftarrow \mathbb{Z}_q^k \times \mathbb{Z}_q$, and then setting:

$$\sigma_i := \left(\begin{array}{c} \boldsymbol{\rho} = [\overline{\mathbf{B}}\mathbf{r}]_1,\ \hat{\rho} = [\underline{\mathbf{B}}\mathbf{r}]_1,\ \boldsymbol{\psi} = \text{TAG}\ [\overline{\mathbf{B}}\mathbf{r}]_1, \\ \gamma = \mathbf{k}\cdot\boldsymbol{\mu}_i + [k_0]_1 + \mathbf{d}\cdot\boldsymbol{\rho} + \mathbf{e}\cdot\boldsymbol{\psi},\ \tau = [\text{TAG}]_2, \\ \boldsymbol{\pi} = \Pi.\mathsf{sim}(\mathsf{trap}, (\boldsymbol{\mu}_i, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma)) \end{array}\right)$$

It then sends σ_i to $\mathcal{A}$, and adds $\boldsymbol{\mu}_i$ to the list $\mathcal{M}$. After it obtains Q signatures, $\mathcal{A}$ responds with a message $\boldsymbol{\mu}^*$ and a claimed signature on it σ^*. Adversary wins if $\boldsymbol{\mu}^* \notin \mathcal{M}$ and $(\boldsymbol{\mu}^*, \sigma^*)$ passes verify. Define:

$$\boxed{\mathsf{WIN}_0 \stackrel{\triangle}{=} (\boldsymbol{\mu}^* \notin \mathcal{M})\ \textbf{and}\ (\mathsf{verify}(\mathsf{CRS}_v, \boldsymbol{\mu}^*, \sigma^*) = 1)}$$

This game exactly replicates the real construction to the adversary. So the adversary's advantage in $\mathbf{G_0}$ is the EUF-CMA advantage we seek to bound.

Game $\mathbf{G_1}$: The challenge-response in this game is the same as Game $\mathbf{G_0}$ except that in each signature the value TAG is chosen randomly but distinctly from all the earlier TAG's. The winning condition remains the same, i.e. WIN_0.

The statistical difference between the view of the adversary in $\mathbf{G_0}$ and $\mathbf{G_1}$ is the probability of collision in the choice of TAG for the Q signature queries in $\mathbf{G_0}$, which is at most $Q^2/(2\cdot q)$.

Game $\mathbf{G_2}$: The challenge-response in this game is the same as $\mathbf{G_1}$. The winning condition is now defined as

$$\begin{array}{r} \mathsf{WIN}_2 \stackrel{\triangle}{=} \mathsf{WIN}_0\ \textbf{and}\ (\sigma^* = (\boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*)\ \text{s.t.} \\ (\gamma^* = \mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}\cdot\boldsymbol{\rho}^* + \mathbf{e}\cdot\boldsymbol{\psi}^*) \\ \textbf{and}\ ((\boldsymbol{\rho}^*, \hat{\rho}^*) \in \mathit{Span}([\mathbf{B}]_1)) \end{array}$$

The difference in advantages of the adversary is upper bounded by the unbounded true-simulation-soundness of Π:

$$|\text{Prob}_2[\mathsf{WIN}_2] - \text{Prob}_1[\mathsf{WIN}_1]| \leq \text{ADV}^{TSS}_{\Pi} \tag{1}$$

$\mathsf{Gen}() : \cdots$ Sample $\mathbf{B}^{(k+1)\times k} \leftarrow \mathcal{D}_k$

Let $\mathbf{t} = (\underline{\mathbf{B}}\ \overline{\mathbf{B}}^{-1})^\top \in \mathbb{Z}_q^k$

Games 0-3 Sample $(\mathbf{d},\ \mathbf{e}) \leftarrow \mathbb{Z}_q^k \times \mathbb{Z}_q^k$

Games 4-6 Sample $(\mathbf{d}_1,\ d_2,\ \mathbf{e}_1,\ e_2) \leftarrow \mathbb{Z}_q^k \times \mathbb{Z}_q \times \mathbb{Z}_q^k \times \mathbb{Z}_q\ \cdots$

$\mathsf{Sign}(sk,\ \boldsymbol{\mu}_j \in \mathbb{G}_1^n)$:

Sample $(\boldsymbol{\rho},\ \theta,\ \phi) \leftarrow \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_1$

Game 0 Sample $\text{TAG} \leftarrow \mathbb{Z}_q$

Games 1-6 Sample $\text{TAG} \leftarrow \mathbb{Z}_q \setminus \{\text{TAG}_l\}_{l<j}$

Let $\boldsymbol{\psi} := \text{TAG}\ \boldsymbol{\rho}$

Let $(\hat{\rho}, \gamma) :=$

Game 0-3 $\big(\mathbf{t}\cdot\boldsymbol{\rho},\ \mathbf{k}\cdot\boldsymbol{\mu}_y + [k_0]_1 + \mathbf{d}\cdot\boldsymbol{\rho} + \text{TAG}\ \mathbf{e}\cdot\boldsymbol{\rho}\big)$

Game 4-5 $\big(\mathbf{t}\cdot\boldsymbol{\rho},\ \mathbf{k}\cdot\boldsymbol{\mu}_y + [k_0]_1 + (\mathbf{d}_1 + d_2\mathbf{t})\cdot\boldsymbol{\rho} + \text{TAG}\ (\mathbf{e}_1 + e_2\mathbf{t})\cdot\boldsymbol{\rho}\big)$

Game 6 $(\theta,\ \phi)$

Let $\boldsymbol{\pi} := \Pi.\mathsf{sim}(\mathsf{trap},\ (\boldsymbol{\mu}_j, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma))$ and $\tau := [\text{TAG}]_2$

Return $(\boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma, \tau, \boldsymbol{\pi})$

$\mathsf{WIN} \triangleq$ $(\boldsymbol{\mu}^* \notin \mathcal{M})$ **and** $\Pi.\mathsf{ver}(\mathsf{CRS}_v, (\boldsymbol{\mu}^*, \boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*), \pi^*)$ **and** $\mathsf{e}(\boldsymbol{\rho}^*, \tau^*) \stackrel{?}{=} \mathsf{e}(\boldsymbol{\psi}^*, [1]_2)$

Games 2-6 **and** $\sigma^* = (\boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*)$:

Game 2 $\gamma^* \stackrel{?}{=} \mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}\cdot\boldsymbol{\rho}^* + \mathbf{e}\cdot\boldsymbol{\psi}^*$

Game 3 $\mathsf{e}(\gamma^*, [1]_2) \stackrel{?}{=} \mathsf{e}(\mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}\cdot\boldsymbol{\rho}^*, [1]_2) + \mathsf{e}(\mathbf{e}\cdot\boldsymbol{\rho}^*, \tau^*)$

Games 4-6 $\mathsf{e}(\gamma^*, [1]_2) \stackrel{?}{=} \mathsf{e}(\mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}_1\cdot\boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2) + \mathsf{e}(\mathbf{e}_1\cdot\boldsymbol{\rho}^* + e_2\hat{\rho}^*, \tau^*)$

Games 0-4 **and** $\hat{\rho}^* \stackrel{?}{=} \mathbf{t}\cdot\boldsymbol{\rho}^*$

Fig. 3. G Games and winning conditions

Game $\mathbf{G_3}$: The challenge-response in this game is the same as $\mathbf{G_2}$. The winning condition is now defined as

$$\begin{aligned}\mathsf{WIN}_3 \triangleq \mathsf{WIN}_0 \textbf{ and } (\sigma^* = (\boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*) \text{ s.t.} \\ (\mathsf{e}(\gamma^*, [1]_2) = \mathsf{e}(\mathbf{k} \cdot \boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d} \cdot \boldsymbol{\rho}^*, [1]_2) + \mathsf{e}(\mathbf{e} \cdot \boldsymbol{\rho}^*, \tau^*)) \\ \textbf{and } ((\boldsymbol{\rho}^*, \hat{\rho}^*) \in Span([\mathbf{B}]_1))\end{aligned}$$

Note that the predicate WIN_3 is efficiently computable by the challenger $\mathcal{C}$ as it generated $\mathbf{B}$ as part of the language parameters $(\mathbf{M}, \mathbf{a})$. As WIN_0 implies $\mathsf{e}(\boldsymbol{\psi}^*, [1]_2) = \mathsf{e}(\boldsymbol{\rho}^*, \tau^*)$, the winning condition is unchanged from the previous game and thus, $\mathrm{Prob}_2[\mathsf{WIN}_2]$ is the same as $\mathrm{Prob}_3[\mathsf{WIN}_3]$.

Game $\mathbf{G_4}$: Define $\mathbf{t}^{k\times 1} \triangleq (\underline{\mathbf{B}}\ \overline{\mathbf{B}}^{-1})^\top$. Since $\mathbf{B}$ is overwhelmingly a full ranked matrix, we observe that $\boldsymbol{\rho}$ can be just sampled uniformly randomly from $\mathbb{Z}_q^k$ and $\hat{\rho}$ can be set to $\mathbf{t} \cdot \boldsymbol{\rho}$ in the signature generation algorithm. Also in the winning condition $(\boldsymbol{\rho}^*, \hat{\rho}^*) \in Span([\mathbf{B}]_1)$ can be equivalently written as $\hat{\rho}^* \stackrel{?}{=} \mathbf{t} \cdot \boldsymbol{\rho}^*$, with no other constraints on $\boldsymbol{\rho}^*$.

In Game $\mathbf{G_4}$, the challenger $\mathcal{C}$ picks $(\mathbf{d}_1,\ d_2,\ \mathbf{e}_1,\ e_2)$ at random from $\mathbb{Z}_q^{2k+2}$, and sets $\mathbf{d} = \mathbf{d}_1 + d_2\mathbf{t}$ and $\mathbf{e} = \mathbf{e}_1 + e_2\mathbf{t}$ (i.e., instead of directly picking $\mathbf{d}$ and $\mathbf{e}$ at random while defining $\mathcal{L}_{\mathrm{par}}$). This has no statistical change in the view of the adversary.

The winning condition is now defined and computed as:

$$\begin{aligned}\mathsf{WIN}_4 \triangleq \mathsf{WIN}_0 \textbf{ and } (\sigma^* = (\boldsymbol{\rho}^*, \hat{\boldsymbol{\rho}}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*) \text{ s.t.} \\ (\mathsf{e}(\gamma^*, [1]_2) = \mathsf{e}(\mathbf{k} \cdot \boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}_1 \cdot \boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2) \\ + \mathsf{e}(\mathbf{e}_1 \cdot \boldsymbol{\rho}^* + e_2\hat{\rho}^*, \tau^*)) \\ \textbf{and } (\hat{\rho}^* \stackrel{?}{=} \mathbf{t} \cdot \boldsymbol{\rho}^*)\end{aligned}$$

Since $\hat{\rho}^* = \mathbf{t} \cdot \boldsymbol{\rho}^*$, it directly follows that $(\mathbf{d}_1 + d_2\mathbf{t}) \cdot \boldsymbol{\rho}^*$ is the same as $(\mathbf{d}_1 \cdot \boldsymbol{\rho}^* + d_2\hat{\rho}^*)$, and $(\mathbf{e}_1 + e_2\mathbf{t}) \cdot \boldsymbol{\rho}^*$ is the same as $(\mathbf{e}_1 \cdot \boldsymbol{\rho}^* + e_2\hat{\rho}^*)$. Therefore $\mathsf{WIN}_4 \equiv \mathsf{WIN}_3$.

Game $\mathbf{G_5}$: In this game, we define WIN_5 to be the same as WIN_4, except that it does not have the conjunct $\hat{\rho}^* \stackrel{?}{=} \mathbf{t} \cdot \boldsymbol{\rho}$.

$$\begin{aligned}\mathsf{WIN}_5 \triangleq \mathsf{WIN}_0 \textbf{ and } (\sigma^* = (\boldsymbol{\rho}^*, \hat{\boldsymbol{\rho}}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*) \text{ s.t.} \\ (\mathsf{e}(\gamma^*, [1]_2) = \mathsf{e}(\mathbf{k} \cdot \boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}_1 \cdot \boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2) \\ + \mathsf{e}(\mathbf{e}_1 \cdot \boldsymbol{\rho}^* + e_2\hat{\rho}^*, \tau^*))\end{aligned}$$

We now prove that:

$$|\mathrm{Prob}_5[\mathsf{WIN}_5] - \mathrm{Prob}_4[\mathsf{WIN}_4]| \leq 1/q \tag{2}$$

Firstly, note that the probability spaces in $\mathbf{G_4}$ and $\mathbf{G_5}$ are identical. We will now show that an adversary $\mathcal{A}$ in Game $\mathbf{G_4}$ has probability at most $1/q$ of forcing WIN_5 while not satisfying WIN_4, i.e., forcing WIN_5 **and** $\hat{\rho}^* \neq \mathbf{t} \cdot \boldsymbol{\rho}^*$.

The claim is an easy consequence of private hash on a non-$Span([\mathbf{B}]_1)$ word being random and independent of the public (projection) hash key [CS02]. Here, the public hash key is $[\mathbf{d}_1 + d_2\mathbf{t}]_1$, with private hash key $(\mathbf{d}_1, d_2)$ (see Section 2.3). The public hash key is given to the adversary as part of all the signatures issued to the adversary. In particular it is used in computing γ component of the signature. The QA-NIZK proof is simulated, and the QA-NIZK simulator trapdoors do not use $(\mathbf{d}_1, d_2)$. Further, $(\mathbf{d}_1, d_2)$ are not used anywhere else, including CRS_v.

If $(\boldsymbol{\rho}^*, \hat{\rho}^*) \notin Span([\mathbf{B}]_1)$, then the right side of the pairing equation in WIN_5 includes an additive component $\mathsf{e}(\mathbf{d}_1 \cdot \boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2)$, which is the same as $\mathsf{e}(P, [1]_2)$ where P is the private hash of $(\boldsymbol{\rho}^*, \hat{\rho}^*)$ using keys $(\mathbf{d}_1, d_2)$. Since, all other additive terms on the right hand side of the pairing equation are independent of this hash proof system, and the adversary $\mathcal{A}$ also supplies γ^*, the probability of $\mathsf{e}(\gamma^*, [1]_2)$ equaling the right hand side is at most $1/q$. This finishes the proof of the claim.

Game $\mathbf{G_6}$: In this game the challenger generates all signatures σ_i with $\hat{\rho}_i$ and γ_i set to uniformly and independently chosen random values. The computation of $\boldsymbol{\rho}, \boldsymbol{\psi}, \tau$ and $\boldsymbol{\pi}$ and the winning condition remain the same as in $\mathbf{G_5}$.

We now claim that the difference between the advantage of the adversary in Game $\mathbf{G_6}$ and Q times the advantage of the adversary in Game $\mathbf{G_5}$ is negligible in Lemma 1 below, which is proved later:

Lemma 1.

$$|\mathrm{Prob}_5[\mathsf{WIN}_5] - Q \cdot \mathrm{Prob}_6[\mathsf{WIN}_6]| \leq Q^2 \left(\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + \frac{1}{q} \right)$$

Now, in Game $\mathbf{G_6}$, all the signatures on the Q adversarial queries are generated without using k_0. Since k_0 is also not part of the public key (which includes CRS_v), the probability of adversary satisfying WIN_6 is $1/q$. Thus, probability of WIN_6 holding in Game $\mathbf{G_6}$ is at most $1/q$:

$$\mathrm{Prob}_6[\mathsf{WIN}_6] \leq 1/q$$

Thus the proof of Lemma 1 will conclude the proof, which we proceed to do next.

Proof (of Lemma 1). To prove this lemma we consider several hybrid Games $\mathbf{G_{5,i}}$, for $i \in [0..Q]$, where $\mathbf{G_{5,0}}$ will turn out to be the same as $\mathbf{G_5}$, and $\mathbf{G_{5,Q}}$ will turn out to be the same as $\mathbf{G_6}$. The hybrid Games $\mathbf{G_{5,i}}$ for $i \in [0..Q]$ are defined as follows.

Game $\mathbf{G_{5,i}}$: The game differs from $\mathbf{G_5}$ as follows: After it has generated the public key and sent it to $\mathcal{A}$ just as in $\mathbf{G_5}$, the challenger now picks a random index z from $[1..Q]$. If $i < Q$, it picks i distinct indices randomly from $[1..Q] \setminus \{z\}$. Call this set of indices as S (note S is empty in Game $\mathbf{G_{5,0}}$). If $i = Q$, let S be the full set $[1..Q]$. While generating a signature on a query with index $j \in S$, the challenger generates the signature as in Game $\mathbf{G_6}$ (i.e. random γ_i and $\hat{\rho}_i$ terms),

and for a query with index outside S it generates the signature as in Game $\mathbf{G_5}$. The winning predicate for the adversary remains the same, i.e., WIN_5. As the winning condition will remain the same till the end of proof, we just define $\mathsf{WIN} \equiv \mathsf{WIN}_5$. The game is described in Figure 4.

$\mathsf{Gen}()$: $\cdots$

Sample $z \leftarrow [1..Q]$ and $S \leftarrow 2^{[1..Q]\setminus\{z\}}$, such that $|S| = i$

$\mathsf{Sign}(sk,\ \boldsymbol{\mu}_j \in \mathbb{G}_1^n)$:

Sample $(\boldsymbol{\rho},\ \theta,\ \phi,\ \text{TAG}) \leftarrow \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_1 \times (\mathbb{Z}_q \setminus \{\text{TAG}_l\}_{l<j})$

Let $\boldsymbol{\psi} := \text{TAG}\ \boldsymbol{\rho}$

If $(j \notin S)$ let $(\hat{\rho}, \gamma) :=$

$\big(\mathbf{t}\cdot\boldsymbol{\rho},\ \mathbf{k}\cdot\boldsymbol{\mu}_j + [k_0]_1 + (\mathbf{d}_1 + d_2\mathbf{t})\cdot\boldsymbol{\rho} + \text{TAG}\ (\mathbf{e}_1 + e_2\mathbf{t})\cdot\boldsymbol{\rho}\big)$

Else if $(j \in S)$ let $(\hat{\rho}, \gamma) :=$

$(\theta,\ \phi)$

Let $\boldsymbol{\pi} := \Pi.\mathsf{sim}(\mathsf{trap},\ (\boldsymbol{\mu}_j, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma))$ and $\tau := [\text{TAG}]_2$

Return $(\boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma, \tau, \boldsymbol{\pi})$

$\mathsf{WIN} \triangleq \mathsf{WIN}_0$ **and** $\sigma^* = (\boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*)$:

$\mathsf{e}(\gamma^*, [1]_2) \stackrel{?}{=} \mathsf{e}(\mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}_1\cdot\boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2) + \mathsf{e}(\mathbf{e}_1\cdot\boldsymbol{\rho}^* + e_2\hat{\rho}^*, \tau^*)$

Fig. 4. Games $\mathbf{G_{5,i}}$

Note that in Game $\mathbf{G_{5,0}}$, the probability of adversary winning, i.e. WIN holding is the same as in Game $\mathbf{G_5}$, since the set S is empty, and hence z might as well not be chosen.

To prove the requisite probability relations between the different games, consider the following predicate GOOD, defined at the end of each game. We will denote the components of the j-th signature σ_j by using subscript j.

$$\boxed{\mathsf{GOOD} \triangleq \forall j \in [1..Q] \setminus \{z\} : (\text{TAG}^* \neq \text{TAG}_j)}$$

Lemma 2.

$$\text{Prob}_{5,0}[\mathsf{WIN}] \leq Q \cdot \text{Prob}_{5,0}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]$$

Lemma 3. *For* $i \in [1..Q-1]$,

$$\left| \begin{matrix} \text{Prob}_{5,i-1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\text{Prob}_{5,i}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \end{matrix} \right| \leq \text{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

Lemma 4.

$$|\text{Prob}_{5,Q-1}[\mathsf{WIN}] - \text{Prob}_{5,Q}[\mathsf{WIN}]| \leq \text{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

Fig. 5. Lemmas

Given the definitions of Games $\mathbf{G_{5,i}}$ and GOOD above, we now prove the lemma via the three lemmas given in Figure 5. Chaining Lemma 3 sequentially $(Q-1)$ times, it follows that

$$\left| \begin{matrix} \text{Prob}_{5,0}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] - \\ \text{Prob}_{5,Q-1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,] \end{matrix} \right| \leq (Q-1) \cdot (\text{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q)$$

Now noting that $\text{Prob}_{5,Q-1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,] \leq \text{Prob}_{5,Q-1}[\mathsf{WIN}]$ and using Lemma 4, we get:

$$\left| \begin{matrix} \text{Prob}_{5,0}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\ \text{Prob}_{5,Q}[\mathsf{WIN}] \end{matrix} \right| \leq Q \cdot (\text{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q)$$

Now, using Lemma 2, we finally establish Lemma 1:

$$|\text{Prob}_{5,0}[\mathsf{WIN}] - Q \cdot \text{Prob}_{5,Q}[\mathsf{WIN}]| \leq Q^2 \left(\text{ADV}_{\mathcal{D}_k\text{-MDDH}} + \frac{1}{q} \right)$$

We proceed to prove Lemmas 2, 3 and 4 now.

Proof (of Lemma 2). We equivalently show that:

$$\text{Prob}_{5,0}[\overline{\mathsf{GOOD}} \mid \mathsf{WIN}] \leq (1 - 1/Q)$$

First note that in Game $\mathbf{G_{5,0}}$, the value z can be chosen after the adversary has supplied its forged signature. Now, observe that:

$$\text{Prob}_{5,0}[\overline{\mathsf{GOOD}} \mid \mathsf{WIN}] \leq \text{Prob}_{5,0}[\ \text{TAG}^* \neq \text{TAG}_z \mid \mathsf{WIN}\ \textbf{and}\ \exists j : \text{TAG}^* = \text{TAG}_j\ ]$$

Since z is chosen after the adversary has replied with the forgery and given TAG^* equals some TAG_j, the probability of $z = j$ is at least $1/Q$ (regardless of WIN holding or not), and thus the probability of TAG^* equaling TAG_z is at least $1/Q$.

Discussion of Lemmas 3 and 4. From a formal proof perspective, one goes through many hybrid games, where in each subsequent hybrid Game $\mathbf{G_{5,i}}$, the signature of one more element is simulated without using the affine component $[k_0]_1$. However, as is well known from proofs of Cramer-Shoup encryption, this can only be done as long as the forgery uses a different tag from the signature being simulated. Thus, the simulator instead guesses an index z, and picks the additional signature to be simulated from a query index different from z. This is always possible, as long as the simulator is in hybrid game $\mathbf{G_{5,i}}$, with $i < Q-1$. If the simulator's guess turns out to be wrong, the adversary is declared outright winner. However, this gives the adversary only a Q factor advantage over its success in an MDDH challenge game.

The other main difference from Cramer-Shoup encryption is that there is no real decryption, but just a verification of the signature using private trapdoor keys. This can also be done efficiently using the bilinear pairing available, and this is the reason why a single additional test of the relationship between $[t]_2$, $[t\mathbf{r}]_1$ and $[\mathbf{r}]_1$ suffices.

The proof of Lemma 4, which handles the case $i = Q-1$ is similar to (and easier than) proof of Lemma 3 except that in game $\mathbf{G_{5,Q-1}}$, all but one signatures are simulated without keys k_0 and $\mathbf{k}$. This makes the analysis similar to that of a one-time signature scheme.

Proof (of Lemma 3). We will consider three hybrid games which are summarized in Figure 6. Game $\mathbf{H_0}$ will be the same as game $\mathbf{G_{5,i-1}}$, and $\mathbf{H_2}$ the same as $\mathbf{G_{5,i}}$.

Game $\mathbf{H_0}$: The challenger picks yet another index y at random from $[1..Q] \setminus (\{z\} \cup S)$, and issues the signature on the y-th query in the same way as for other indices *not in* S. The idea is that in these sequence of games we will convert the signature generation on the y-th index to be same as for those indices *in* S. This will effectively expand the set S by one element and thus enable us to transition from Game $\mathbf{G_{5,i-1}}$ to $\mathbf{G_{5,i}}$, as long as $i \leq Q-1$. Games $\mathbf{H_0}$ and $\mathbf{G_{5,i-1}}$ are semantically equivalent.

Game $\mathbf{H_1}$: In Game $\mathbf{H_1}$, the challenger issues the signature on the y-th query as follows: it picks $\boldsymbol{\rho}_y$, θ and TAG_y at random. It sets $\hat{\rho}_y = \theta$, $\boldsymbol{\psi}_y = \text{TAG}_y\ \boldsymbol{\rho}_y$, $\tau_y = [\text{TAG}_y]_2$ and $\gamma_y = \mathbf{k} \cdot \boldsymbol{\mu}_y + [k_0]_1 + (\mathbf{d}_1 \cdot \boldsymbol{\rho}_y + d_2\hat{\rho}_y) + \text{TAG}_y\ (\mathbf{e}_1 \cdot \boldsymbol{\rho}_y + e_2\hat{\rho}_y)$. It computes a QA-NIZK $\boldsymbol{\pi}_y$, on the tuple $(\boldsymbol{\mu}_j, \boldsymbol{\rho}_y, \hat{\rho}_y, \boldsymbol{\psi}_y, \gamma_y)$ using the QA-NIZK simulator crssim, just as in all previous games. It outputs as signature σ_y the tuple $(\boldsymbol{\rho}_y, \hat{\rho}_y, \boldsymbol{\psi}_y, \gamma_y, \tau_y, \boldsymbol{\pi}_y)$. Rest of the game and the winning condition is the same as $\mathbf{H_0}$. We now prove that:

$$\left| \begin{array}{l} \text{Prob}_{H_0}[\mathsf{WIN\ \textbf{and}\ GOOD}] - \\ \text{Prob}_{H_1}[\mathsf{WIN\ \textbf{and}\ GOOD}] \end{array} \right| < \text{ADV}_{\mathcal{D}_k\text{-MDDH}} \tag{3}$$

Let $\mathcal{A}$ be any efficient adversary playing against $\mathcal{C}$ in either game $\mathbf{H_0}$ or $\mathbf{H_1}$. Using $\mathcal{A}$ and the challenger $\mathcal{C}$ we will build another adversary $\mathcal{A}'$ that plays against a $\mathcal{D}_k$-MDDH challenger. So, suppose the MDDH challenger issues either

$\mathsf{Gen}()$: $\cdots$

Sample $z \leftarrow [1..Q]$ and $S \leftarrow 2^{[1..Q]\setminus\{z\}}$, such that $|S| = i$

Sample $y \leftarrow [1..Q] \setminus (\{z\} \cup S)$

$\mathsf{Sign}(sk,\ \boldsymbol{\mu}_j \in \mathbb{G}_1^n)$:

Sample $(\boldsymbol{\rho},\ \theta,\ \phi,\ \text{TAG}) \leftarrow \mathbb{G}_1^k \times \mathbb{G}_1 \times \mathbb{G}_1 \times (\mathbb{Z}_q \setminus \{\text{TAG}_l\}_{l<j})$

Let $\boldsymbol{\psi} := \text{TAG}\ \boldsymbol{\rho}$

If $(j \notin S \cup \{y\})$ let $(\hat{\rho}, \gamma) :=$

$\big(\mathbf{t}\cdot\boldsymbol{\rho},\ \mathbf{k}\cdot\boldsymbol{\mu}_j + [k_0]_1 + (\mathbf{d}_1 + d_2\mathbf{t})\cdot\boldsymbol{\rho} + \text{TAG}\ (\mathbf{e}_1 + e_2\mathbf{t})\cdot\boldsymbol{\rho}\big)$

Else if $(j \in S)$ let $(\hat{\rho}, \gamma) :=$

$(\theta,\ \phi)$

Else if $(j = y)$ let $(\hat{\rho}, \gamma) :=$

Game 0 $\big(\mathbf{t}\cdot\boldsymbol{\rho},\ \mathbf{k}\cdot\boldsymbol{\mu}_y + [k_0]_1 + (\mathbf{d}_1 + d_2\mathbf{t})\cdot\boldsymbol{\rho} + \text{TAG}\ (\mathbf{e}_1 + e_2\mathbf{t})\cdot\boldsymbol{\rho}\big)$

Game 1 $\big(\theta,\ \mathbf{k}\cdot\boldsymbol{\mu}_y + [k_0]_1 + (\mathbf{d}_1\cdot\boldsymbol{\rho} + d_2\theta) + \text{TAG}\ (\mathbf{e}_1\cdot\boldsymbol{\rho} + e_2\theta)\big)$

Game 2 $(\theta,\ \phi)$

Let $\boldsymbol{\pi} := \Pi.\mathsf{sim}(\mathsf{trap},\ (\boldsymbol{\mu}_j, \boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma))$ and $\tau := [\text{TAG}]_2$

Return $(\boldsymbol{\rho}, \hat{\rho}, \boldsymbol{\psi}, \gamma, \tau, \boldsymbol{\pi})$

$\mathsf{WIN} \triangleq \mathsf{WIN}_0$ **and** $\sigma^* = (\boldsymbol{\rho}^*, \hat{\rho}^*, \boldsymbol{\psi}^*, \gamma^*, \tau^*, \boldsymbol{\pi}^*)$:

$\mathsf{e}(\gamma^*, [1]_2) \stackrel{?}{=} \mathsf{e}(\mathbf{k}\cdot\boldsymbol{\mu}^* + [k_0]_1 + \mathbf{d}_1\cdot\boldsymbol{\rho}^* + d_2\hat{\rho}^*, [1]_2) + \mathsf{e}(\mathbf{e}_1\cdot\boldsymbol{\rho}^* + e_2\hat{\rho}^*, \tau^*)$

Fig. 6. H Games and winning condition

a real tuple ($[\mathbf{B}]_1,\ \boldsymbol{\zeta} = [\mathbf{Br}]_1$) or a fake tuple ($[\mathbf{B}]_1,\ \boldsymbol{\zeta} = [\mathbf{r}']_1 \in \mathbb{G}_1^{k+1}$), with $\mathbf{B} \leftarrow \mathcal{D}_k$ and $(\mathbf{r},\ \mathbf{r}') \leftarrow \mathbb{Z}_q^k \times \mathbb{Z}_q^{k+1}$. In the first case, we will say that $\mathcal{A}'$ is in the MDDHREAL game and in the latter case, we will say that $\mathcal{A}'$ is in the MDDHFAKE game. $\mathcal{A}'$ uses $[\mathbf{B}]_1$ to simulate $\mathcal{C}$ in building the language parameters $\mathcal{L}_{\text{par}}$ by choosing all other random values on its own. It then simulates $\mathcal{C}$ for the rest of the game $\mathbf{H_0}/\mathbf{H_1}$, including interaction with $\mathcal{A}$, till the point of issuing the y-th signature. For the y-th signature, $\mathcal{A}'$ sets $(\boldsymbol{\rho}_y,\ \hat{\rho}_y) := \boldsymbol{\zeta}$, and picks TAG_y at random, and sets $\boldsymbol{\psi}_y = \text{TAG}_y\ \boldsymbol{\rho}_y$. The values τ_y and γ_y and $\boldsymbol{\pi}_y$ can then be computed from values already obtained.

After $\mathcal{A}'$ issues this signature to $\mathcal{A}$, adversary $\mathcal{A}'$ continues the simulation of $\mathcal{C}$, along with its interaction with $\mathcal{A}$ till the computation and output of winning condition. $\mathcal{A}'$ outputs 1 iff WIN **and** GOOD. Now, note that if $\mathcal{A}'$ is in the MDDHREAL game, then the view of the adversary $\mathcal{A}$ is identical to its view in $\mathbf{H_0}$. And, if $\mathcal{A}'$ is in the MDDHFAKE game, then the view of the adversary $\mathcal{A}$ is identical to its view in $\mathbf{H_1}$. Thus:

$$\text{Prob}[\mathcal{A}'(\text{MDDHREAL}) = 1] = \text{Prob}_{H_0}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]$$

$$\text{Prob}[\mathcal{A}'(\text{MDDHFAKE}) = 1] = \text{Prob}_{H_1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}].$$

That completes the proof of the claim, as the maximum advantage any efficient adversary has in winning an MDDH-challenge game is $\text{ADV}_{\mathcal{D}_k\text{-MDDH}}$.

Game $\mathbf{H_2}$: In Game $\mathbf{H_2}$, in the computation of the signature on y-th query, the value γ_y is just sampled independently randomly from $\mathbb{Z}_q$. The winning condition remains WIN. We now prove that the view of the adversary in Games $\mathbf{H_2}$ and $\mathbf{H_1}$ is statistically indistinguishable. More precisely,

$$|\text{Prob}_{H_2}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] - \text{Prob}_{H_1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]| \leq 1/q$$

The claim is a consequence of private hash on a non-$Span([\mathbf{B}]_1)$ word being random and independent of the public universal$_2$ projection hash key [CS02]. Here, the public universal$_2$ projection hash key is the pair $[\mathbf{d}_1 + d_2\mathbf{t}]_1$ and $[\mathbf{e}_1 + e_2\mathbf{t}]_1$, with private universal$_2$ hash key $(\mathbf{d}_1, d_2, \mathbf{e}_1, e_2)$. The public hash key is given to the adversary as part of all the signatures issued to the adversary, with the exception of the signature issued by $\mathcal{C}$ on query index y. In the y-th query, the challenger discloses to the adversary one private hash on a non-$Span([\mathbf{B}]_1)$ word. In particular $\boldsymbol{\gamma}_y$ includes as an additive term $(\mathbf{d}_1 \cdot \boldsymbol{\rho}_y + d_2\hat{\rho}_y) + \text{TAG}_y\ (\mathbf{e}_1 \cdot \boldsymbol{\rho}_y + e_2\hat{\rho}_y)$, which is exactly the private universal$_2$ hash on $(\boldsymbol{\rho}_y, \hat{\rho}_y)$ using tag t_y. Now note that GOOD and $z \neq y$ implies $\text{TAG}^* \neq \text{TAG}_y$, as y was chosen distinct from z. Thus, TAG^* is different from TAG_y used in the one private hash given to the adversary on a non-$Span([\mathbf{B}]_1)$ word.

Recall that the QA-NIZK proof is simulated, and the QA-NIZK simulator trapdoors do not use $(\mathbf{d}_1, d_2, \mathbf{e}_1, e_2)$. Further, $(\mathbf{d}_1, d_2, \mathbf{e}_1, e_2)$ are not used anywhere else, including CRS_v.

Thus the additive term $(\mathbf{d}_1 \cdot \boldsymbol{\rho}_y + d_2\hat{\rho}_y) + \text{TAG}_y\ (\mathbf{e}_1 \cdot \boldsymbol{\rho}_y + e_2\hat{\rho}_y)$ in γ_y (in Game $\mathbf{H_1}$) completely hides $([k_0]_1 + \mathbf{k} \cdot \boldsymbol{\mu}_y)$. Thus, γ_y can just as well be sampled

independently randomly. This is the same as Game $\mathbf{H_2}$, and that proves the claim.

Thus, collecting all the inequalities, between consecutive games from $\mathbf{H_0}$ to $\mathbf{H_2}$, it follows that:

$$\left|\begin{matrix}\mathrm{Prob}_{5,i-1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\mathrm{Prob}_{5,i}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,]\end{matrix}\right| \leq \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

Proof (of Lemma 4). The proof of this lemma is similar to proof of Lemma 3, except that the predicate GOOD here is just defined to be true. The proof of Lemma 3 goes through all the hybrid games with predicate GOOD defined as true, except for the proof of

$$|\mathrm{Prob}_{H_2}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] - \mathrm{Prob}_{H_1}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]| \leq 1/q.$$

This proof for Lemma 3 required the fact that GOOD implies that $\mathrm{TAG}^* \neq \mathrm{TAG}_y$, where y was the query index being simulated with a fake MDDH tuple. Since, here we have defined GOOD to be true, there is no such restriction on TAG^*.

In case $\mathrm{TAG}^* \neq \mathrm{TAG}_y$, the proof continues to hold as before. If $\mathrm{TAG}^* = \mathrm{TAG}_y$, we note that since we are in various hybrids of initial game $\mathbf{H_0} = \mathbf{G_{5,Q-1}}$, no signature generated by $\mathcal{C}$ (other than the y-th signature) uses k_0 or $\mathbf{k}$. The trapdoors k_0 and $\mathbf{k}$ are also not used in generation of public key. Thus, the only information available to $\mathcal{A}$ about k_0 and $\mathbf{k}$ is through the y-th signature simulation, which includes $\mathbf{k} \cdot \boldsymbol{\mu}_y + [k_0]_1$ as an additive term. Thus, for WIN to hold, $\mathcal{A}$ must produce $\gamma^* - (\mathbf{d}_1 \cdot \boldsymbol{\rho}^* + d_2\hat{\rho}^*) - \mathrm{TAG}^*\ (\mathbf{e}_1 \cdot \boldsymbol{\rho}^* + e_2\hat{\rho}^*)$ equal to $\mathbf{k} \cdot \boldsymbol{\mu}^* + [k_0]_1$. By simple linear algebra, this latter quantity is random, even given $\mathbf{k} \cdot \boldsymbol{\mu}_y + [k_0]_1$, for $\boldsymbol{\mu}^* \neq \boldsymbol{\mu}_y$.

This linear algebra fact is most conveniently seen by the following information-theoretic argument: Let $\alpha \stackrel{\triangle}{=} \mathbf{k} \cdot \boldsymbol{\mu}_y + [k_0]_1$ and $\beta \stackrel{\triangle}{=} \mathbf{k} \cdot \boldsymbol{\mu}^* + [k_0]_1$. Now sample $(\mathbf{k},\ k') \leftarrow \mathbb{Z}_q^n \times \mathbb{Z}_q$, and then set $[k_0]_1 := [k']_1 - \mathbf{k} \cdot \boldsymbol{\mu}_y$. Then we have $\alpha = [k']_1$ and $\beta = [k']_1 + \mathbf{k} \cdot (\boldsymbol{\mu}^* - \boldsymbol{\mu}_y)$. Thus α is uniformly random and independent of $\mathbf{k}$, while β has an independent uniformly random distribution due to the additional term $\mathbf{k} \cdot (\boldsymbol{\mu}^* - \boldsymbol{\mu}_y)$, where $\mathbf{k}$ is uniformly random and $(\boldsymbol{\mu}^* - \boldsymbol{\mu}_y)$ is non-zero.

3.2 Improved Security Reduction for the SPS Scheme

Theorem 3. *For any efficient adversary $\mathcal{A}$, which makes at most Q signature queries before attempting a forgery, its probability of success in the EUF-CMA game against the SPS scheme is at most*

$$\mathrm{ADV}_{\Pi}^{TSS} + Q \cdot (2 + \log Q) \cdot \left(\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + \frac{1}{q}\right) + \frac{Q^2}{2q} + \frac{1}{q}$$

Proof. In the proof of this theorem and related lemmas, without loss of generality, we will assume that the number of signature queries Q made by the

adversary is a power of two. This can cause at most a factor of two difference in the success probability of the adversary.

The Games $\mathbf{G_0}$ to $\mathbf{G_6}$ are same as in proof of Theorem 2. However, we now obtain a better upper bound on the probability of event WIN holding in Game $\mathbf{G_5}$, as opposed to the bound obtained in Lemma 1.

Lemma 5.

$$\mathrm{Prob}_5[\mathsf{WIN}] \leq Q \cdot (2 + \log Q) \cdot (\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q)$$

Proof. Again, to prove this lemma we consider several hybrid Games $\mathbf{G_{5,i}}$, for $i \in [0..Q]$, where $\mathbf{G_{5,0}}$ will turn out to be same as $\mathbf{G_5}$, and $\mathbf{G_{5,Q}}$ will turn out to be same as $\mathbf{G_6}$. The hybrid Games $\mathbf{G_{5,i}}$ are defined slightly differently in this proof as compared to the proof of Lemma 1. These are summarized in Figure 7 and explained below.

Game $\mathbf{G_{5,i}}$: For $0 \leq i < Q$, the game differs from $\mathbf{G_5}$ as follows: After it has generated the public key and sent it to $\mathcal{A}$ just as in $\mathbf{G_5}$, the challenger now picks a random set Z of size $2^{\lfloor \log (Q-i) \rfloor}$ of distinct indices from $[1..Q]$. It then picks i distinct indices randomly from $[1..Q] \setminus Z$. Call this set of indices as S (note that S is empty in Game $\mathbf{G_{5,0}}$). If $i = Q$, let S be the full set $[1..Q]$. While generating signatures on a query with index $j \in S$, the challenger generates the signature as in Game $\mathbf{G_6}$ (i.e., samples γ and $\hat{\rho}$ uniformly randomly), and for all other queries it generates the signature as in Game $\mathbf{G_5}$. The winning predicate for the adversary remains the same, i.e., WIN.

Note that for hybrid Game $\mathbf{G_{5,i}}$, such that $(Q - i)$ is a power of two, the union of disjoint sets S and Z is the complete set of indices $[1..Q]$. However, in the next hybrid Game $\mathbf{G_{5,i+1}}$, the set Z is cut by half in size, so that there is a choice to pick S from $[1..Q] \setminus Z$. Thus, to relate such a hybrid Game $\mathbf{G_{5,i}}$ (i.e. when $Q - i$ is a power of two) to the next hybrid Game $\mathbf{G_{5,i+1}}$, we introduce an intermediate Game $\mathbf{G'_{5,i}}$.

For i, define Game $\mathbf{G'_{5,i}}$ to be similar to Game $\mathbf{G_{5,i}}$ except that the set of random and distinct indices Z is chosen to be of size 2^{l-1}. For S, we choose i distinct indices from $[1..Q] \setminus Z$, as before. The rest of the game and the winning condition remains the same.

For each hybrid Game $\mathbf{G_{5,i}}$ or $\mathbf{G'_{5,i}}$, define the following predicate

$$\mathsf{GOOD} \triangleq \forall j \in [1..Q] \setminus Z : (\boldsymbol{\tau}^* \neq \boldsymbol{\tau}_j)$$

In Lemma 6 below, we show that for $i = Q - 2^l$, the probability of WIN and GOOD holding in Game $\mathbf{G_{5,i}}$ is at most two times the probability of WIN and GOOD holding in Game $\mathbf{G'_{5,i}}$. Note that, for $i = 0$ the predicate GOOD is equivalent to true, as Z is the complete set. Thus, this implies that the probability of WIN holding in Game $\mathbf{G_5}$ is at most two times the probability of WIN and GOOD holding in Game $\mathbf{G'_{5,0}}$.

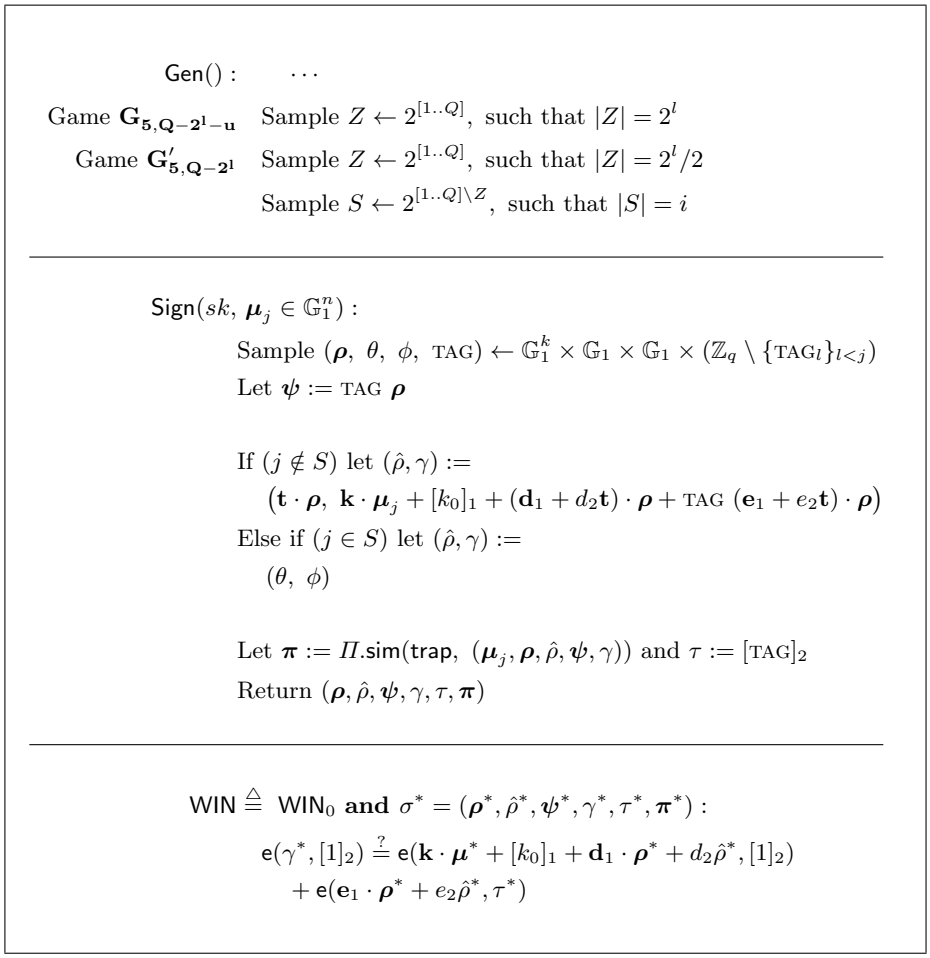

Fig. 7. Modified Games $\mathbf{G}_{\mathbf{5},\mathbf{i}}$. Above, $\log Q \le l \le 0$ and $0 \le u \le 2^l - 1$.

Using Lemmas 6 and 7 below, we now prove the recurrence relation that for $l \in [2..\log Q]$:

$$\begin{aligned}\mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{Q}-\mathbf{2^l}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq 2^{l-1} \cdot (\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q) + \\ 2 \cdot \mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{Q}-\mathbf{2^{l-1}}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]\end{aligned}$$

Also, as a base case we have (from Lemma 7),

$$\begin{aligned}\mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{Q}-\mathbf{2}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq 2 \cdot (\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q) + \\ \mathrm{Prob}_{\mathbf{G}_{\mathbf{5},\mathbf{Q}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,]\end{aligned}$$

However, in the proof of Lemma 1, we established that in the last hybrid Game $\mathbf{G}_{\mathbf{5},\mathbf{Q}}$, the probability of WIN is at most $1/q$. Thus,

$$\mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{Q}-\mathbf{2}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq 2 \cdot \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 3/q$$

Thus, by maintaining the induction hypothesis, for every $l \in [1..\log Q]$:

$$\mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{Q}-\mathbf{2^l}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq (2^{l-1}l + 2^l) \cdot (\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q)$$

we get by induction that

$$\mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{0}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq \left(\frac{Q}{2} \cdot \log Q + Q\right) \cdot (\mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q)$$

Lemma 6. *For $i \in [0..Q-1]$, and $i = Q - 2^l$,*

$$\mathrm{Prob}_{\mathbf{G}_{\mathbf{5},\mathbf{i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]\ \leq\ 2 \cdot \mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]$$

Proof. For i, $0 \leq i < Q$, such that $(Q-i)$ a power of two, note that the Game $\mathbf{G}_{\mathbf{5},\mathbf{i}}$ can be defined by first picking a set S of i distinct and random indices from $[1..Q]$, and then setting $Z = [1..Q] \setminus S$. Similarly, the Game $\mathbf{G}'_{\mathbf{5},\mathbf{i}}$ can be defined by first picking a set S of i distinct indices, and then picking a set Z' of $(Q-i)/2$ distinct and random indices from $Z = [1..Q] \setminus S$. This set can be picked after the adversary has replied with its claimed forgery. In other words, the probability of WIN and GOOD holding in $\mathbf{G}'_{\mathbf{5},\mathbf{i}}$ is same as probability of WIN and GOOD' holding in $\mathbf{G}_{\mathbf{5},\mathbf{i}}$ where GOOD' is defined as

$$\mathsf{GOOD}' \stackrel{\triangle}{=} \forall j \in [1..Q] \setminus Z' : (\boldsymbol{\tau}^* \neq \boldsymbol{\tau}_j)$$

Letting DIST stand for the predicate $\forall j \in [1..Q] : (\boldsymbol{\tau}^* \neq \boldsymbol{\tau}_j)$, it follows that GOOD' **and** GOOD **and** $\neg\mathsf{DIST}$ is equivalent to GOOD' **and** $\neg\mathsf{DIST}$. Thus,

$$\begin{aligned}\Pr[\mathsf{GOOD}' \mid \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}] = \Pr[\mathsf{GOOD}' \mid \mathsf{GOOD}\ \textbf{and}\ \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}] \\ \cdot \Pr[\mathsf{GOOD} \mid \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}]\end{aligned}$$

Now, $\Pr[\mathsf{GOOD}' \mid \mathsf{GOOD}\ \textbf{and}\ \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}]$ is exactly $1/2$. Thus, noting that GOOD is equivalent to $\mathsf{DIST} \vee (\mathsf{GOOD}\ \textbf{and}\ \neg\mathsf{DIST})$, and GOOD' is equivalent to $\mathsf{DIST} \vee (\mathsf{GOOD}'\ \textbf{and}\ \neg\mathsf{DIST})$, it follows that

$$\begin{aligned}\Pr[\mathsf{GOOD} \mid \mathsf{WIN}] = &\Pr[\mathsf{DIST} \mid \mathsf{WIN}] \\ &+ \Pr[\mathsf{GOOD} \mid \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}] \cdot \Pr[\neg\mathsf{DIST} \mid \mathsf{WIN}] \quad (4)\\ \Pr[\mathsf{GOOD}' \mid \mathsf{WIN}] = &\Pr[\mathsf{DIST} \mid \mathsf{WIN}] \\ &+ \frac{1}{2}\Pr[\mathsf{GOOD} \mid \neg\mathsf{DIST}\ \textbf{and}\ \mathsf{WIN}] \cdot \Pr[\neg\mathsf{DIST} \mid \mathsf{WIN}] \quad (5)\end{aligned}$$

Now, this implies $\Pr[\mathsf{GOOD} \mid \mathsf{WIN}] \leq 2 \cdot \Pr[\mathsf{GOOD}' \mid \mathsf{WIN}]$, because otherwise we obtain a contradiction that $\Pr[\mathsf{DIST} \mid \mathsf{WIN}] < 0$. Thus,

$$\Pr[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \leq 2 \cdot \Pr[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}']$$

Lemma 7. *For $i \in [1..Q]$, if $(Q - i + 1)$ is a power of two and $i \neq Q$, then*

$$\left|\begin{matrix}\mathrm{Prob}_{\mathbf{G'_{5,i-1}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\mathrm{Prob}_{\mathbf{G_{5,i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,]\end{matrix}\right| \leq \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

Otherwise (i.e., if $(Q - i + 1)$ is not a power of two or $i = Q$),

$$\left|\begin{matrix}\mathrm{Prob}_{\mathbf{G_{5,i-1}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\mathrm{Prob}_{\mathbf{G_{5,i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,]\end{matrix}\right| \leq \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

The proof of Lemma 7 is same as that for the proof of Lemma 3 (except for i equal to Q, when it is same as proof of Lemma 4). The only difference is in the proof of

$$|\mathrm{Prob}_{\mathbf{H_2}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] - \mathrm{Prob}_{\mathbf{H_1}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]| \leq 1/q$$

where we now argue that GOOD and $y \notin Z$ implies $t^* \neq t_y$.

Alternate Improved Reduction. The above reduction makes discrete 'big jumps' when $Q-i$ is a power of two and a series of smooth 'short jumps' in between these big jumps. Instead, we can smoothen the entire jump sequence by shortening the set Z by 1 at every i while going from a primed game to an unprimed game. In an unprimed game, Z and S will partition the set $[1..Q]$, while in a primed game there will be $Q - i$ choices for Z'. This will result in the following modifications of Lemmas 6 and 7 :

Lemma 8. *For $i \in [0..Q-2]$,*

$$\mathrm{Prob}_{\mathbf{G_{5,i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \ \leq\ \frac{Q-i}{Q-i-1} \cdot \mathrm{Prob}_{\mathbf{G'_{5,i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}]$$

Lemma 9. *For* $i \in [1..Q-1]$,

$$\left| \begin{matrix} \mathrm{Prob}_{\mathbf{G}'_{\mathbf{5},\mathbf{i-1}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\mathrm{Prob}_{\mathbf{G}_{\mathbf{5},\mathbf{i}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,] \end{matrix} \right| \leq \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

and

$$\left| \begin{matrix} \mathrm{Prob}_{\mathbf{G}_{\mathbf{5},\mathbf{Q-1}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}] \\ -\mathrm{Prob}_{\mathbf{G}_{\mathbf{5},\mathbf{Q}}}[\mathsf{WIN}\ \textbf{and}\ \mathsf{GOOD}\,] \end{matrix} \right| \leq \mathrm{ADV}_{\mathcal{D}_k\text{-MDDH}} + 1/q$$

However, this still results in a $Q \log Q$ loss in security.

Acknowledgments

The authors would like to thank the anonymous referees for helpful comments and filling a couple of gaps in the submission.

References

[ACD+12] Masayuki Abe, Melissa Chase, Bernardo David, Markulf Kohlweiss, Ryo Nishimaki, and Miyako Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology – ASIACRYPT 2012*, volume 7658 of *Lecture Notes in Computer Science*, pages 4–24, Beijing, China, December 2–6, 2012. Springer, Heidelberg, Germany.

[ACHO11] Masayuki Abe, Sherman S. M. Chow, Kristiyan Haralambiev, and Miyako Ohkubo. Double-trapdoor anonymous tags for traceable signatures. In Javier Lopez and Gene Tsudik, editors, *ACNS 11: 9th International Conference on Applied Cryptography and Network Security*, volume 6715 of *Lecture Notes in Computer Science*, pages 183–200, Nerja, Spain, June 7–10, 2011. Springer, Heidelberg, Germany.

[ADK+13] Masayuki Abe, Bernardo David, Markulf Kohlweiss, Ryo Nishimaki, and Miyako Ohkubo. Tagged one-time signatures: Tight security and optimal tag size. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *PKC 2013: 16th International Conference on Theory and Practice of Public Key Cryptography*, volume 7778 of *Lecture Notes in Computer Science*, pages 312–331, Nara, Japan, February 26 – March 1, 2013. Springer, Heidelberg, Germany.

[AFG+10] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-preserving signatures and commitments to group elements. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 209–236, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.

[AGHO11] Masayuki Abe, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Optimal structure-preserving signatures in asymmetric bilinear groups. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 649–666, Santa Barbara, CA, USA, August 14–18, 2011. Springer, Heidelberg, Germany.

[AGO11] Masayuki Abe, Jens Groth, and Miyako Ohkubo. Separating short structure-preserving signatures from non-interactive assumptions. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Computer Science*, pages 628–646, Seoul, South Korea, December 4–8, 2011. Springer, Heidelberg, Germany.

[AHO10] Masayuki Abe, Kristiyan Haralambiev, and Miyako Ohkubo. Signing on Elements in Bilinear Groups for Modular Protocol Design. IACR Cryptology ePrint Archive, 2010, 133.

[AO09] Masayuki Abe and Miyako Ohkubo. A framework for universally composable non-committing blind signatures. In Mitsuru Matsui, editor, *Advances in Cryptology – ASIACRYPT 2009*, volume 5912 of *Lecture Notes in Computer Science*, pages 435–450, Tokyo, Japan, December 6–10, 2009. Springer, Heidelberg, Germany.

[CCS09] Jan Camenisch, Nishanth Chandran, and Victor Shoup. A public key encryption scheme secure against key dependent chosen plaintext and adaptive chosen ciphertext attacks. In Antoine Joux, editor, *Advances in Cryptology – EUROCRYPT 2009*, volume 5479 of *Lecture Notes in Computer Science*, pages 351–368, Cologne, Germany, April 26–30, 2009. Springer, Heidelberg, Germany.

[CL04] Jan Camenisch and Anna Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In Matthew Franklin, editor, *Advances in Cryptology – CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 56–72, Santa Barbara, CA, USA, August 15–19, 2004. Springer, Heidelberg, Germany.

[CLY09] Julien Cathalo, Benoît Libert, and Moti Yung. Group encryption: Non-interactive realization in the standard model. In Mitsuru Matsui, editor, *Advances in Cryptology – ASIACRYPT 2009*, volume 5912 of *Lecture Notes in Computer Science*, pages 179–196, Tokyo, Japan, December 6–10, 2009. Springer, Heidelberg, Germany.

[CS02] Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *Advances in Cryptology – EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 45–64, Amsterdam, The Netherlands, April 28 – May 2, 2002. Springer, Heidelberg, Germany.

[EHK+13] Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 129–147, Santa Barbara, CA, USA, August 18–22, 2013. Springer, Heidelberg, Germany.

[Fuc11] Georg Fuchsbauer. Commuting signatures and verifiable encryption. In Kenneth G. Paterson, editor, *Advances in Cryptology – EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 224–245, Tallinn, Estonia, May 15–19, 2011. Springer, Heidelberg, Germany.

[Fuc09] Georg Fuchsbauer. Automorphic Signatures in Bilinear Groups and an Application to Round-Optimal Blind Signatures. IACR Cryptology ePrint Archive, 2009, 320.

[GH08] Matthew Green and Susan Hohenberger. Universally composable adaptive oblivious transfer. In Josef Pieprzyk, editor, *Advances in Cryptology – ASIACRYPT 2008*, volume 5350 of *Lecture Notes in Computer Science*, pages

179–197, Melbourne, Australia, December 7–11, 2008. Springer, Heidelberg, Germany.

[Gro06] Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In Xuejia Lai and Kefei Chen, editors, *Advances in Cryptology – ASIACRYPT 2006*, volume 4284 of *Lecture Notes in Computer Science*, pages 444–459, Shanghai, China, December 3–7, 2006. Springer, Heidelberg, Germany.

[GS08] Jens Groth and Amit Sahai. Efficient non-interactive proof systems for bilinear groups. In Nigel P. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 415–432, Istanbul, Turkey, April 13–17, 2008. Springer, Heidelberg, Germany.

[Har11] Kristiyan Haralambiev. Efficient cryptographic primitives for non-interactive zero-knowledge proofs and applications. *PhD Dissertation*, 2011.

[JR13] Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology – ASIACRYPT 2013, Part I*, volume 8269 of *Lecture Notes in Computer Science*, pages 1–20, Bengalore, India, December 1–5, 2013. Springer, Heidelberg, Germany.

[KPW15] Eike Kiltz, Jiaxin Pan, and Hoeteck Wee. Structure-preserving signatures from standard assumptions, revisited. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *Advances in Cryptology – CRYPTO 2015, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 275–295, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany.

[KW15] Eike Kiltz and Hoeteck Wee. Quasi-adaptive NIZK for linear subspaces revisited. In Elisabeth Oswald and Marc Fischlin, editors, *Advances in Cryptology – EUROCRYPT 2015, Part II*, volume 9057 of *Lecture Notes in Computer Science*, pages 101–128, Sofia, Bulgaria, April 26–30, 2015. Springer, Heidelberg, Germany.

[LPY15] Benoît Libert, Thomas Peters, and Moti Yung. Short group signatures via structure-preserving signatures: Standard model security from simple assumptions. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *Advances in Cryptology – CRYPTO 2015, Part II*, volume 9216 of *Lecture Notes in Computer Science*, pages 296–316, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany.

[Pol78] J. M. Pollard. Monte Carlo methods for index computation (mod p). Math. Comp. 32 (1978) 918-924.