Shorter Quadratic QA-NIZK Proofs

Vanesa Daza[1], Alonso González[2⋆], Zaira Pindado[1], Carla Ràfols[1⋆⋆], Javier Silva[1⋆⋆⋆]

[1] Cybercat and Universitat Pompeu Fabra, Barcelona, Spain.
[2] ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), France.
{vanesa.daza,zaira.pindado,carla.rafols,javier.silva}@upf.edu, alonso.gonzalez@ens-lyon.fr

Abstract. Despite recent advances in the area of pairing-friendly Non-Interactive Zero-Knowledge proofs, there have not been many efficiency improvements in constructing arguments of satisfiability of quadratic (and larger degree) equations since the publication of the Groth-Sahai proof system (JoC'12). In this work, we address the problem of aggregating such proofs using techniques derived from the interactive setting and recent constructions of SNARKs. For certain types of quadratic equations, this problem was investigated before by González et al. (ASIACRYPT'15). Compared to their result, we reduce the proof size by approximately 50% and the common reference string from quadratic to linear, at the price of using less standard computational assumptions. A theoretical motivation for our work is to investigate how efficient NIZK proofs based on falsifiable assumptions can be. On the practical side, quadratic equations appear naturally in several cryptographic schemes like shuffle and range arguments. This is the full version of the paper of the same title published at PKC 2019.

1 Introduction

NIZK in Bilinear Groups. Non-Interactive Zero-Knowledge Proofs allow to convince any party of the truth of a statement without revealing any other information. They are a very useful building block in the construction of cryptographic protocols. Since the first pairing-friendly NIZK proof system of Groth, Ostrovsky and Sahai [17] many different constructions have emerged in different models and under different assumptions, for various types of statements. Compared to a plain discrete logarithm setting, bilinear groups have a rich structure which is specially amenable to construct NIZK proofs.

Among this variety of results, there are three particularly interesting families with different advantages in terms of generality, efficiency or strength of the assumptions. On the one hand, there is a line of research initiated by Groth, Ostrovsky and Sahai [17] and which culminated in the Groth-Sahai proof system [19]. The latter result provides relatively efficient proofs for proving satisfiability of several types of quadratic equations in bilinear groups based on standard assumptions. Although several works have tried to improve the efficiency of Groth-Sahai proofs [7, 28], for many equation types they still remain the best alternative based on falsifiable assumptions.

Another family of results are the constructions of quasi-adaptive NIZK (QA-NIZK) arguments, initiated by Jutla and Roy [20] and leading to very efficient proofs of very concrete statements. Most notably, given a bilinear group $gk := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2)$, proving membership in linear spaces in $\mathbb{G}_1^m$ or $\mathbb{G}_2^m$, for some $m \in \mathbb{N}$, requires only one group element [22, 21]. The power of the quasi-adaptive notion of zero-knowledge allows to specialize the common reference string to the language one is proving membership in, trading generality for efficiency under very weak computational assumptions. Other works have constructed proofs for different languages in the QA-NIZK setting, like the proof for bilateral spaces (linear spaces in $\mathbb{G}_1^m \times \mathbb{G}_2^n$) [12], or, beyond linear spaces, the language of vector commitments to integers opening to a boolean vector [12] or shuffles and range proofs [13].

⋆ This author was supported in part by the French ANR ALAMBIC project (ANR-16-CE39-0006).
⋆⋆ This author was supported by a Marie Curie "UPF Fellows" Postdoctoral Grant and by the Catalan Agency for Management of University and Research Grant 2017 SGR 1739.
⋆⋆⋆ This author was supported by a PhD grant from the Spanish government, co-financed by the ESF (Ayudas para contratos predoctorales para la formación de doctores 2016).

Finally, in the last few years, an extremely successful line of research has constructed succinct non-interactive arguments of knowledge (zk-SNARKs) [14, 25, 10, 6, 15] for NP complete languages, which are not only constant-size (independent of the witness size) but which are also very efficient in a concrete sense. One of the main downsides of SNARKs is that their security relies on knowledge of exponent assumptions, a very strong type of assumptions classified as non-falsifiable [27]. However, one cannot achieve succinctness (proofs essentially independent of the size of the statement being proved and its witness) and security based on falsifiable assumptions at the same time, as per the impossibility result by Gentry and Wichs [11].

Commit-and-Prove. In a broad sense, we can think of many of the results in these three families as commit-and-prove schemes [5]. This is very clear for the Groth-Sahai proof system, which has even been recasted in the commit-and-prove formalism by Escala and Groth [7]. This is probably less obvious for some results in the QA-NIZK setting. However, as noted already in the first QA-NIZK construction of membership in linear spaces [20], in these cases one can often think of the statement as a commitment to the witness. For instance, in the case of proving that a vector $\boldsymbol{y}$ in the exponent is in the linear span of the columns of some matrix $\mathbf{A}$, this means that $\boldsymbol{y} = \mathbf{A}\boldsymbol{w}$ and we can think of $\boldsymbol{y}$ as a commitment to $\boldsymbol{w}$. Finally, in the case of many SNARK constructions, e.g. [6] the commitment is usually a "knowledge commitment" — from which the witness is extracted in the soundness proof using knowledge assumptions — while the rest can be considered the "proof".

With this idea in mind, it is interesting to compare these three approaches for constructing proofs of satisfiability of d equations in n variables in bilinear groups in terms of proof size. We observe that for linear equations, while the original Groth-Sahai proof system required $O(n)$ group elements for the commit step and $O(d)$ for the "prove" one, recent works have shown how to aggregate the proof in the quasi-adaptive setting [21, 12], reducing the "prove" step to $O(1)$ in many cases. For quadratic equations in the other hand, we summarize the three different approaches in Table 1.

Construction	Assumption	Commitment size	Proof size	CRS size
Groth-Sahai [17]	Falsifiable	$O(n)$	$O(n)$	$O(1)$
QA-NIZK [12]	Falsifiable	$O(n)$	$10\|\mathbb{G}_1\| + 10\|\mathbb{G}_2\|$	$O(n^2)$
SNARKs [6]	Non-falsifiable	$\|\mathbb{G}_1\| + \|\mathbb{G}_2\|$	$2\|\mathbb{G}_1\|$	$O(n)$

Table 1. Three different approaches for proving quadratic equations in bilinear groups. For concreteness, assume that one wants to prove that a set of values $x_1, \ldots, x_n$ form a bitstring, that is, satisfiability of $x_i(x_i - 1) = 0$.

Motivation. Quadratic equations are much more powerful than linear ones. In particular, they allow to prove boolean Circuit Sat, but they are also important to prove other statements like range, shuffle proofs or validity of an encrypted vote. While for proving statements about large circuits non-falsifiable assumptions are necessary to get around impossibility results, it would be desirable to eliminate them in less demanding settings, to understand better what the security claims mean in a concrete sense. As in the QA-NIZK arguments for linear spaces, there are even natural situations in which the statement is already "an encrypted witness", and it seems unnatural to use the full power of knowledge of exponent assumptions in these cases (for instance, in the case of vote validity).

In summary, it is worth investigating efficiency improvements for quadratic equations under falsifiable assumptions. In particular, aggregating the "prove" step would be an important step towards this goal. The techniques for the linear case do not apply to the quadratic one, and we are only aware of one result in aggregating the proof of quadratic equations, namely the bitstring argument of González et al. [12] for proving that a set of commitments to integers opens to boolean values. There is a large concrete gap between this result and the others in the non-falsifiable setting both in terms of the size of the proof and the common reference string. Thus, it is natural to ask if it is possible to reduce the gap and improve on this result importing techniques from SNARKs in the falsifiable setting.

1.1 Our results

We introduce new techniques to aggregate proofs of quadratic equations. First, in Sect. 3.1, we construct a proof system for proving that d equations of the type $X_i(X_i - 2) = 0$ are satisfied, where X_i is an affine

combination of some $a_1, \ldots, a_n$. The size of the proof is constant and the set of commitments to the variables is of size linear in n, and the size of the CRS is linear in d. The prover computes a number of exponentiations linear in $n + d$, while the verifier computes a number of pairings linear in d. Our proof system is perfect zero-knowledge and computationally sound under a variant of the so-called target strong Diffie-Hellman assumption. These assumptions belong to the broader class of q-assumptions, where each instance of the problem is of size proportional to some integer q, which in our case is the number of equations. In particular, the bitstring language of [12] can be formulated as such a system of equations. In Sect. 3.2 we discusss as a particular case an argument for unit vector, and argue how to modify our general proof system so that it can be proven sound under static assumptions. A typical application of membership in these languages is for computing disjunctions of statements such as "the committed verification key is equal to $\mathcal{V}_1$, or $\mathcal{V}_2$, ..., or $\mathcal{V}_m$", which might be expressed as $vk = \sum_{i=1}^{m} b_i \mathcal{V}_i, b_i \in \{0, 1\}$ and $(b_1, \ldots, b_m)$ is a unit vector.

Next, in Sect. 4, we generalize the previous argument to prove that d equations of the type $(X_i - z_1)(X_i - z_2) \ldots (X_i - z_m) = 0$ are satisfied, where X_i is an affine combination of the variables $a_1, \ldots, a_n$. For this we combine techniques from the interactive setting of [4] for proving set membership in a set of size m of $\mathbb{Z}_p$ with ideas from Sect. 3.1 and from quasi-adaptive aggregation [21]. In Appendix A, we illustrate how to use this for improved range proofs in bilinear groups under falsifiable assumptions.

Finally, in Sect. 5 we discuss two approaches to construct shuffle arguments. They are the most efficient in terms of proof size in the common reference string model under falsifiable assumptions in bilinear groups (comparing favorably even to the best constructions in the generic bilinear group model [9]), but they have large public parameters (quadratic in the shuffle size).

	Language	Proof size	CRS size	Assumption
Sect. 3.1	Quadratic equations	$4\|\mathbb{G}_1\| + 6\|\mathbb{G}_2\|$	$(d + O(1))\|\mathbb{G}_1\|$ $+(d + 3n + O(1))\|\mathbb{G}_2\|$	q-STSDH (7)
Sect. 3.2	Unit vector	$6\|\mathbb{G}_1\| + 6\|\mathbb{G}_2\|$	$(4(n + 1) + O(1))\|\mathbb{G}_1\|$ $+(5(n + 1) + O(1))\|\mathbb{G}_2\|$	1-STSDH (7)
Sect. 4.2	Set Membership	$6\|\mathbb{G}_1\| + 6\|\mathbb{G}_2\|$	$(mn + 2n + 3m + O(1))\|\mathbb{G}_1\|$ $+(5mn + O(1))\|\mathbb{G}_2\|$	$\mathcal{Z}$-GSDH (6), q-QTSDH (8)

Table 2. The table shows the proof sizes (not including commitments) and CRS sizes of our constructions. We consider d variables and n equations, and m is the size of the set from the set membership proof. The assumptions 6, 7 and 8 are new.

	Proof size	CRS size	Assumption
[16]	$15n + 246$	$2n + 8$	PPA, SPA, DLIN
[9]	$(4n - 1)\|\mathbb{G}_1\| + (3n + 1)\|\mathbb{G}_2\|$	$O(n)(\|\mathbb{G}_1\| + \|\mathbb{G}_2\|)$	Bilinear generic group model
[13]	$(4n + 17)\|\mathbb{G}_1\| + 14\|\mathbb{G}_2\|$	$O(n^2)\|\mathbb{G}_1\| + O(n)\|\mathbb{G}_2\|$	SXDH, SSDP [13]
Sect. 5.1	$(4n + 11)\|\mathbb{G}_1\| + 8\|\mathbb{G}_2\|$	$O(n^2)\|\mathbb{G}_1\| + O(n)\|\mathbb{G}_2\|$	SXDH, 1-STSDH (7)
Sect. 5.2	$(2n + 11)\|\mathbb{G}_1\| + 8\|\mathbb{G}_2\|$	$O(n^2)(\|\mathbb{G}_1\| + \|\mathbb{G}_2\|)$	SXDH, n-QTSDH (7)

Table 3. Comparison of our shuffle arguments with state-of-the-art arguments. Note that PPA stands for the Pairing Permutation Assumption and SPA for the Simultaneous Pairing Assumption.

1.2 Our techniques

Let $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ be groups of prime order p and let $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a bilinear map. Both SNARKs and our schemes can be seen as "commit-and-prove" schemes [7]: in the first step we commit to the solution of the equations. In the case of SNARKs, the knowledge assumption allows to extract the solutions from a constant-size commitment during the soundness proof, but we are trying to avoid using these assumptions, so we require perfectly binding commitments for each element of the solution. The second step is a proof of the opening of the commitments verifying the equations.

Let $r_1, \ldots, r_d \in \mathbb{Z}_p$. The "prove" part is handled with a polynomial aggregation technique in which satisfiability of a set of d equations is encoded into a polynomial $p(X)$ such that $p(r_j) = 0$ if and only if the jth equation is satisfied. To prove that d equations are satisfied, one needs to prove that $p(X)$ is divisible by $\prod_{j=1}^{d}(X - r_j)$. The key to succinctness is that the divisibility condition is only checked at a secret point s chosen by the trusted party who generates the CRS. This preserves soundness as long as the prover only knows s (or powers thereof) in $\mathbb{G}_1$ or $\mathbb{G}_2$, but not its discrete logarithm.

In the soundness proof, the witness is extracted from the knowledge commitment, and then used to find some r_j such that $p(r_j) \neq 0$ and compute auxiliary information which, together with the proof, allows to break a hard problem, e.g. the q-Target Strong Diffie-Hellman Assumption in [6]. Under non-falsifiable assumptions the commitments, even if perfectly binding, can be only opened in the source groups, instead of in $\mathbb{Z}_p$. This has an impact on the soundness proof, as it is not possible to eliminate some terms in the proof to find a solution to the q-TSDH assumption, so we need to consider a more flexible assumption. Furthermore, since the solutions define the coefficients of polynomial $p(X)$, our access to this polynomial is much more limited.

For our set-membership proof we start from the following insight: the satisfiability of equation $b(b-1) = 0$ can be proven showing knowledge of a signature for b if only signatures for 0 or 1 are known. This approach can be easily extended for larger sets of solutions as done by Camenisch et al. [4]. To express the validity of many signature and message pairs, we again encode the signature verification equations as a problem of divisibility of polynomials.

This requires the signature verification to be expressible as a set of quadratic equations. While structure preserving signatures clearly solve this problem, it is overkill, since we only need unforgeability against static queries. Further, even the generic group construction of [15] requires at least 3 group elements. We choose basic Boneh-Boyen signatures since each signature consists of only one group element. Our argument needs to solve other technical difficulties which are explained in more detail in Sect. 4.

1.3 Related Works

The recent line of research in SNARKs started with [14], in which the first sub-linear arguments without random oracles were presented, but with CRS of quadratic size. Subsequent works have defined alternative models for the encoding of the circuit [24, 10, 6, 15], reducing the CRS size to linear and obtaining smaller proofs, going as small as 3 group elements in the case of [15]. In particular, our encodings are based on those of [10, 6].

When considering falsifiable assumptions, one classic way to prove quadratic equations in the non-interactive setting makes use of Groth-Sahai proofs [18], which are quite efficient and can be aggregated to obtain a constant-size proof of many equations.

In this work, we also use techniques from QA-NIZK proofs. This model was introduced in [20] to build proofs of membership in linear subspaces over $\mathbb{G}_1$ or $\mathbb{G}_2$. It was later improved to make proofs constant-size (independent of the size of the witness) [21, 22, 23] and adapted to the asymmetric setting [12]. Although introduced initially to build proofs of linear equations, the QA-NIZK setting has also been used to build the first constant-size aggregated proofs of some quadratic equations under standard assumptions [12], in particular the proof that a set of commitments open to bits.

The usage of signatures for proving membership in a set dates back to the work of Camenisch et al. [4] in the interactive setting, and in the non-interactive setting by Rial et al. [29]. Both works achieve constant-size proofs but without aggregation (i.e. proving n instances requires $O(n)$ communication). Set membership proofs were also recently investigated by Bootle and Groth [3] in the interactive setting. They construct proofs logarithmic in the size of the set and aggregate n instances with a multiplicative overhead of $O(\sqrt{n})$. In the non-interactive setting, González et al. constructed set membership proofs of size linear in the size of the set and aggregated many instances without any overhead [13].

1.4 Organization

In Sect. 2 we establish the assumptions required for our proofs, present the relevant security definitions and recall the subschemes that we will make use of, namely ElGamal encryption, Boneh-Boyen signatures, Groth-Sahai proofs and proofs of membership in linear spaces. Sect. 2.3 recalls some standard definitions in the QA-NIZK setting. In Sect. 3, we present our proof system for satisfiability of quadratic equations. In Sect. 3.2, we give an argument to prove that a commitment opens to a unit vector which can be proven secure based on a static assumption. In Sect. 4 we present an aggregated argument to prove membership in a set of $\mathbb{Z}_p$. In Sect. 5 we discuss new approaches to construct shuffle arguments. In the appendix, Sect. A discusses the application of the set membership argument in $\mathbb{Z}_p$ to range proofs, and Sect. B contains the proofs of hardness in generic groups of our assumptions.

2 Preliminaries

2.1 Bilinear Groups and Implicit Notation

Let $\mathcal{G}$ be some probabilistic polynomial time algorithm which on input 1^λ, where λ is the security parameter, returns the *group key* which is the description of an asymmetric bilinear group $gk := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2)$, where $\mathbb{G}_1, \mathbb{G}_2$ and $\mathbb{G}_T$ are additive groups of prime order p, the elements $\mathcal{P}_1, \mathcal{P}_2$ are generators of $\mathbb{G}_1, \mathbb{G}_2$ respectively, $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficiently computable, non-degenerate bilinear map, and there is no efficiently computable isomorphism between $\mathbb{G}_1$ and $\mathbb{G}_2$.

Elements in $\mathbb{G}_\gamma$ are denoted implicitly as $[a]_\gamma := a\mathcal{P}_\gamma$, where $\gamma \in \{1, 2, T\}$ and $\mathcal{P}_T := e(\mathcal{P}_1, \mathcal{P}_2)$. For simplicity, we often write $[a]_{1,2}$ for the pair $[a]_1, [a]_2$. The pairing operation will be written as a product $\cdot$, that is $[a]_1 \cdot [b]_2 = [a]_1[b]_2 = e([a]_1, [b]_2) = [ab]_T$. Vectors and matrices are denoted in boldface. Given a matrix $\mathbf{T} = (t_{i,j})$, $[\mathbf{T}]_\gamma$ is the natural embedding of $\mathbf{T}$ in $\mathbb{G}_\gamma$, that is, the matrix whose (i, j)th entry is $t_{i,j}\mathcal{P}_\gamma$. We denote by $|\mathbb{G}_\gamma|$ the bit-size of the elements of $\mathbb{G}_\gamma$.

$\mathbf{I}_n$ refers to the identity matrix in $\mathbb{Z}_p^{n\times n}$, $\mathbf{0}_{m\times n}$ refers to the all-zero matrix in $\mathbb{Z}_p^{m\times n}$, and $\boldsymbol{e}_i^n$ the ith element of the canonical basis of $\mathbb{Z}_p^n$ (simply $\mathbf{I}$, $\mathbf{0}$, and $\boldsymbol{e}_i$, respectively, if n, m are clear from the context).

Given a set $\mathcal{R} = \{r_1, \ldots, r_d\} \subset \mathbb{Z}_p$, we denote by $\ell_i(X) = \prod_{j\neq i} \frac{(X - r_i)}{(r_j - r_i)}$ the ith Lagrange interpolation polynomial associated to $\mathcal{R}$.

2.2 Hardness Assumptions

Definition 1. *Let $\ell, k \in \mathbb{N}$. We call $\mathcal{D}_{\ell,k}$ a matrix distribution if it outputs (in PPT time, with overwhelming probability) matrices in $\mathbb{Z}_p^{\ell\times k}$. We define $\mathcal{D}_k := \mathcal{D}_{k+1,k}$.*

The following applies for $\mathbb{G}_\gamma$, where $\gamma \in \{1, 2\}$.

Assumption 1 (Matrix Decisional Diffie-Hellman Assumption in $\mathbb{G}_\gamma$ [8]) *For all non-uniform PPT adversaries $\mathcal{A}$,*

$$|\Pr[\mathcal{A}(gk, [\mathbf{A}]_\gamma, [\mathbf{A}\boldsymbol{w}]_\gamma) = 1] - \Pr[\mathcal{A}(gk, [\mathbf{A}]_\gamma, [\boldsymbol{z}]_\gamma) = 1]| \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_p^k, [\boldsymbol{z}]_\gamma \leftarrow \mathbb{G}_\gamma^\ell$ and the coin tosses of adversary $\mathcal{A}$.

Intuitively, the $\mathcal{D}_{\ell,k}$-MDDH assumption means that it is hard to decide whether a vector is in the image space of a matrix or it is a random vector, where the matrix is drawn from $\mathcal{D}_{\ell,k}$. In this paper we will refer to the following matrix distributions:

$$\mathcal{L}_k : \mathbf{A} = \begin{pmatrix} a_1 & 0 & \ldots & 0 \\ 0 & a_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & a_k \\ 1 & 1 & \ldots & 1 \end{pmatrix}, \qquad \mathcal{U}_{\ell,k} : \mathbf{A} = \begin{pmatrix} a_{1,1} & \ldots & a_{1,k} \\ \vdots & \ddots & \vdots \\ a_{\ell,1} & \ldots & a_{\ell,k} \end{pmatrix},$$

where $a_i, a_{i,j} \leftarrow \mathbb{Z}_p$. The $\mathcal{L}_k$-MDDH Assumption is the k-linear family of Decisional Assumptions and corresponds to the Decisional Diffie-Hellman (DDH) Assumption in $\mathbb{G}_\gamma$ when $k = 1$. The SXDH Assumption states that DDH holds in $\mathbb{G}_\gamma$ for all $\gamma \in \{1, 2\}$. The $\mathcal{U}_{\ell,k}$-MDDH assumption is the *Uniform* Assumption and is the weakest of all matrix assumptions of size $\ell \times k$.

Additionally, we will be using the following family of computational assumptions:

Assumption 2 (Kernel Diffie-Hellman Assumption in $\mathbb{G}_\gamma$ [26]) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[[\boldsymbol{x}]_{3-\gamma} \leftarrow \mathcal{A}(gk, [\mathbf{A}]_\gamma) : \boldsymbol{x} \neq 0 \wedge \boldsymbol{x}^\top \mathbf{A} = \mathbf{0}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ and the coin tosses of adversary $\mathcal{A}$.

The $\mathcal{D}_{\ell,k}$-$\mathsf{KerMDH}_{\mathbb{G}_\gamma}$ Assumption is not stronger than the $\mathcal{D}_{\ell,k}$-$\mathsf{MDDH}_{\mathbb{G}_\gamma}$ Assumption, since a solution to the former allows to decide membership in $\mathbf{Im}([\mathbf{A}]_\gamma)$. In asymmetric bilinear groups, there is a natural variant of this assumption.

Assumption 3 (Split Kernel Diffie-Hellman Assumption [12]) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[[\boldsymbol{r}]_1, [\boldsymbol{s}]_2 \leftarrow \mathcal{A}(gk, [\mathbf{A}]_{1,2}) : \boldsymbol{r} \neq \boldsymbol{s} \wedge \boldsymbol{r}^\top \mathbf{A} = \boldsymbol{s}^\top \mathbf{A}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ and the coin tosses of adversary $\mathcal{A}$.

While the Kernel Diffie-Hellman Assumption says one cannot find a non-zero vector in one of the groups which is in the co-kernel of $\mathbf{A}$, the split assumption says one cannot find different vectors in $\mathbb{G}_1^\ell \times \mathbb{G}_2^\ell$ such that the difference of the vector of their discrete logarithms is in the co-kernel of $\mathbf{A}$. As a particular case, [12] considers the *Split Simultaneous Double Pairing Assumption in* $\mathbb{G}_1, \mathbb{G}_2$ (SSDP) which is the $\mathcal{RL}_2$-$\mathsf{SKerMDH}$ Assumption, where $\mathcal{RL}_2$ is the distribution which results of sampling a matrix from $\mathcal{L}_2$ and replacing the last row by random elements.

***q*-Assumptions.** We first recall the q-Strong Diffie-Hellman and q-Target Strong Diffie-Hellman assumptions, which essentially tell us that inversion is hard in the exponent, even when given q powers of the element to invert.

Assumption 4 (q-Strong Diffie Hellman Assumption in $\mathbb{G}_\gamma$, q-SDH [2]) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[\left(r, [\nu]_\gamma\right) \leftarrow \mathcal{A}(gk, \{[s^i]_\gamma\}_{i=1}^q) : \nu = \frac{1}{s-r}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $s \leftarrow \mathbb{Z}_p$ and the coin tosses of adversary $\mathcal{A}$.

Assumption 5 (q-Target Strong Diffie-Hellman Assumption, q-TSDH [1]) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[(r, [\nu]_T) \leftarrow \mathcal{A}(gk, \{[s^i]_{1,2}\}_{i=1}^q) : \nu = \frac{1}{s-r}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $s \leftarrow \mathbb{Z}_p$ and the coin tosses of adversary $\mathcal{A}$.

The soundness proofs of our schemes will rely on the following variations of the two assumptions above.

Assumption 6 ($\mathcal{Z}$-Group Strong DH Assumption in $\mathbb{G}_\gamma$, $\mathcal{Z}$-GSDH) *Let $\mathcal{Z} \subset \mathbb{Z}_p$ such that $\#\mathcal{Z} = q$. For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[([z_1]_1, [z_2]_\gamma, [\nu]_2) \leftarrow \mathcal{A}\left(gk, \mathcal{Z}, [\varepsilon]_{1,2}, \{[s^i]_{1,2}\}_{i=1}^q\right) : \begin{array}{c} z_1 \notin \mathcal{Z} \wedge z_2 = \varepsilon z_1 \\ \nu = \frac{\prod_{z \in \mathcal{Z}}(s-z)}{s-z_1} \end{array}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $s, \varepsilon \leftarrow \mathbb{Z}_p$ and the coin tosses of adversary $\mathcal{A}$.

The name is motivated by the fact that it is a variant of the q-SDH Assumption in which the adversary must only give $[z_1]_1$ in the group $\mathbb{G}_1$, instead of giving it in $\mathbb{Z}_p$ as in the q-SDH Assumption.

Assumption 7 (q-Square TSDH Assumption, q-STSDH) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[(r,[\beta_1]_1,[\beta_2]_2,[\nu]_T) \leftarrow \mathcal{A}\left(gk,[\varepsilon]_2,\{[s^i]_{1,2}\}_{i=1}^q\right) : \begin{array}{c}\beta_1 \neq \pm 1\\ \beta_2 = \varepsilon\beta_1 \wedge \nu = \frac{\beta_1^2-1}{s-r}\end{array}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $s,\varepsilon \leftarrow \mathbb{Z}_p$ and the coin tosses of adversary $\mathcal{A}$.

Note that the challenger knows ε, s, so this assumption is falsifiable. Indeed, upon receiving $(r,[\beta_1]_1,[\beta_2]_2,[\nu]_T)$, the challenger verifies that $[\beta_1]_1 \neq [\pm 1]_1, e([1]_1,[\beta_2]_2) = e(\varepsilon[\beta_1]_1,[1]_2)$, and $\varepsilon(s-r)[\nu]_T = e([\beta_1]_1,[\beta_2]_2) - e([\varepsilon]_1,[1]_2)$. A similar argument can be made for the other assumptions in this section.

Assumption 8 (q-Quadratic TSDH Assumption, q-QTSDH) *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[\begin{array}{c}\left(r,[\beta_1]_1,[\beta_2]_1,[\tilde{\beta}_1]_2,[\tilde{\beta}_2]_2,[\nu]_T\right) \leftarrow \mathcal{A}\left(gk,[\varepsilon]_{1,2},\{[s^i]_{1,2}\}_{i=1}^q\right) :\\ \beta_1\tilde{\beta}_1 \neq 1\\ \beta_2 = \varepsilon\beta_1 \wedge \tilde{\beta}_2 = \varepsilon\tilde{\beta}_1 \wedge \nu = \frac{\beta_1\tilde{\beta}_1-1}{s-r}\end{array}\right] \approx 0,$$

where the probability is taken over $gk \leftarrow \mathcal{G}(1^\lambda)$, $s,\varepsilon \leftarrow \mathbb{Z}_p$ and the coin tosses of adversary $\mathcal{A}$.

2.3 Quasi-Adaptive Non-Interactive Zero-Knowledge Arguments

In this section we recall the formal definition of Quasi-Adaptive non-interactive zero-knowledge proofs. A Quasi-Adaptive NIZK proof system [20] enables to prove membership in a language defined by a relation $\mathcal{R}_\rho$, which in turn is completely determined by some parameter ρ sampled from a distribution $\mathcal{D}_{gk}$. We say that $\mathcal{D}_{gk}$ is *witness sampleable* if there exists an efficient algorithm that samples (ρ,ω) from a distribution $\mathcal{D}_{gk}^{\mathsf{par}}$ such that ρ is distributed according to $\mathcal{D}_{gk}$, and membership of ρ in the *parameter language* $\mathcal{L}_{\mathsf{par}}$ can be efficiently verified with ω. While the Common Reference String (CRS) can be set based on ρ, the zero-knowledge simulator is required to be a single probabilistic polynomial time algorithm that works for the whole collection of relations $\mathcal{R}_{gk}$.

A tuple of algorithms $(\mathsf{K}_0,\mathsf{K}_1,\mathsf{P},\mathsf{V})$ is called a QA-NIZK proof system for witness-relations $\mathcal{R}_{gk} = \{\mathcal{R}_\rho\}_{\rho\in\sup(\mathcal{D}_{gk})}$ with parameters sampled from a distribution $\mathcal{D}_{gk}$ over associated parameter language $\mathcal{L}_{\mathsf{par}}$, if there exists a probabilistic polynomial time simulator $(\mathsf{S}_1,\mathsf{S}_2)$, such that for all non-uniform PPT adversaries $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$ we have:

Quasi-Adaptive Completeness:

$$\Pr\left[\begin{array}{l}gk \leftarrow \mathsf{K}_0(1^\lambda); \rho \leftarrow \mathcal{D}_{gk}; \mathsf{CRS} \leftarrow \mathsf{K}_1(gk,\rho);\\ (x,w) \leftarrow \mathcal{A}_1(gk,\mathsf{CRS}); \pi \leftarrow \mathsf{P}(\mathsf{CRS},x,w)\end{array} : \mathsf{V}(\mathsf{CRS},x,\pi) = 1 \text{ if } \mathcal{R}_\rho(x,w)\right] = 1.$$

Computational Quasi-Adaptive Soundness:

$$\Pr\left[\begin{array}{l}gk \leftarrow \mathsf{K}_0(1^\lambda); \rho \leftarrow \mathcal{D}_{gk};\\ \mathsf{CRS} \leftarrow \mathsf{K}_1(gk,\rho); (x,\pi) \leftarrow \mathcal{A}_2(gk,\mathsf{CRS})\end{array} : \begin{array}{c}\mathsf{V}(\mathsf{CRS},x,\pi) = 1 \text{ and}\\ \neg(\exists w : \mathcal{R}_\rho(x,w))\end{array}\right] \approx 0.$$

Perfect Quasi-Adaptive Zero-Knowledge:

$$\begin{aligned}&\Pr[gk \leftarrow \mathsf{K}_0(1^\lambda); \rho \leftarrow \mathcal{D}_{gk}; \mathsf{CRS} \leftarrow \mathsf{K}_1(gk,\rho) : \mathcal{A}_3^{\mathsf{P}(\mathsf{CRS},\cdot,\cdot)}(gk,\mathsf{CRS}) = 1] =\\ &\Pr[gk \leftarrow \mathsf{K}_0(1^\lambda); \rho \leftarrow \mathcal{D}_{gk}; (\mathsf{CRS},\tau) \leftarrow \mathsf{S}_1(gk,\rho) : \mathcal{A}_3^{\mathsf{S}(\mathsf{CRS},\tau,\cdot,\cdot)}(gk,\mathsf{CRS}) = 1]\end{aligned}$$

where

– $\mathsf{P}(\mathsf{CRS}, \cdot, \cdot)$ emulates the actual prover. It takes input (x, w) and outputs a proof π if $(x, w) \in \mathcal{R}_\rho$. Otherwise, it outputs $\perp$.
– $\mathsf{S}(\mathsf{CRS}, \tau, \cdot, \cdot)$ is an oracle that takes input (x, w). It outputs a simulated proof $\mathsf{S}_2(\mathsf{CRS}, \tau, x)$ if $(x, w) \in \mathcal{R}_\rho$ and $\perp$ if $(x, w) \notin \mathcal{R}_\rho$.

We assume that CRS contains an encoding of ρ, which is thus available to V.

For witness sampleable distributions, a stronger notion of soundness, where the adversary has also access to a witness of the parameter ρ, is defined in the full version of [12].

Computational Quasi-Adaptive Strong Soundness:

$$\Pr\left[\begin{matrix} gk \leftarrow \mathsf{K}_0(1^\lambda); (\rho, \omega) \leftarrow \mathcal{D}_{gk}^{\mathsf{par}}; \\ \mathsf{CRS} \leftarrow \mathsf{K}_1(gk, \rho); (x, \pi) \leftarrow \mathcal{A}_2(gk, \omega, \mathsf{CRS}) \end{matrix} : \begin{matrix} \mathsf{V}(\mathsf{CRS}, x, \pi) = 1 \text{ and} \\ \neg(\exists w : \mathcal{R}_\rho(x, w)) \end{matrix}\right] \approx 0.$$

2.4 Building Blocks

ElGamal encryption. We denote by $\mathsf{Enc}_{[\mathsf{sk}]}(\mathsf{m}, r)$ the lifted ElGamal encryption of message m with randomness r and public key $[\mathsf{sk}]$. Using implicit group notation, ElGamal encryption is as follows:

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \mathsf{Enc}_{[\mathsf{sk}]}(\mathsf{m}, r) = \mathsf{m}[\boldsymbol{e}_2] + r \begin{bmatrix} 1 \\ \mathsf{sk} \end{bmatrix},$$

where if one knows the secret key sk in $\mathbb{Z}_p$, then one can recover the message in $\mathbb{G}$ by computing $[c_2] - \mathsf{sk}[c_1] = [\mathsf{m}]$. ElGamal encryption is semantically secure under the DDH assumption. It can be seen as a commitment scheme, in which case it is perfectly binding and computationally hiding under the DDH assumption, and in fact this is how we will use it in our schemes.

Boneh-Boyen signatures [2]. We briefly recall Boneh-Boyen signatures. Let $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ be a bilinear group. Messages are elements of $\mathbb{Z}_p$, and signatures are elements of $\mathbb{G}_2$. The secret key is $\mathsf{sk} \in \mathbb{Z}_p$, and the public key (verification key) is $[\mathsf{sk}]_1 \in \mathbb{G}_1$. To sign a message $x \in \mathbb{Z}_p$, the signer computes

$$[\sigma]_2 = \left[\frac{1}{\mathsf{sk} - x}\right]_2$$

The verifier accepts the signature if the equation $e([\mathsf{sk}]_1 - [x]_1, [\sigma]_2) = [1]_T$ holds. Boneh-Boyen signatures are existentially unforgeable under the q-SDH assumption.

Dual-mode commitments and Groth-Sahai proofs [18]. Groth-Sahai proofs allow to prove satisfiability of quadratic equations in bilinear groups in the non-interactive setting. More precisely, Groth-Sahai proofs deal with equations of the form

$$\sum_{j=1}^{m_y} a_j \mathsf{y}_j + \sum_{i=1}^{m_x} b_i \mathsf{x}_i + \sum_{i,j=1}^{m_x, m_y} \gamma_{i,j} \mathsf{x}_i \mathsf{y}_j = t,$$

in which the set of variables is divided into two disjoint subsets $\mathsf{X} = \{\mathsf{x}_1, \ldots, \mathsf{x}_{m_x}\}$ and $\mathsf{Y} = \{\mathsf{y}_1, \ldots, \mathsf{y}_{m_y}\}$, and depending on the type of equation $\mathsf{X}, \mathsf{Y} \subset \mathbb{Z}_p$ (quadratic equations in $\mathbb{Z}_p$), $\mathsf{X} \subset \mathbb{Z}_p, \mathsf{Y} \subset \mathbb{G}_\gamma$ (multi-exponentiation equations in $\mathbb{G}_\gamma$) for $\gamma \in \{1, 2\}$ or $\mathsf{X} \subset \mathbb{G}_1$ and $\mathsf{Y} \subset \mathbb{G}_2$ (pairing product equations). Here the product means a bilinear operation which is multiplication in $\mathbb{Z}_p$, exponentiation or the pairing operation.

The scheme can be seen as a commit-and-prove scheme [7], where in the first step the prover gives commitments to the solutions, and in the second provides a proof that these commitments verify the corresponding equation. In particular, the commitments used are *dual-mode commitments*, that is, commitments that can be either perfectly binding or perfectly hiding, and we can switch from one to the other with an

indistinguishable change of security game. More precisely, Groth-Sahai commitments to field elements $z \in \mathbb{Z}_p$ and group elements $[z] \in \mathbb{G}$ are, respectively:

$$\mathsf{Com}(z; w) = z\,[\boldsymbol{u}] + w[\boldsymbol{u}_1], \qquad \mathsf{Com}([z]; w_1, w_2) = \begin{bmatrix} 0 \\ z \end{bmatrix} + w_1[\boldsymbol{u}_1] + w_2[\boldsymbol{u}_2],$$

where $[\boldsymbol{u}], [\boldsymbol{u}_1], [\boldsymbol{u}_2]$ are vectors in $\mathbb{G}^2$ given in the commitment key, and their definitions depend on whether we want the commitments to be perfectly binding or perfectly hiding.

Groth-Sahai proofs are sound, witness-indistinguishable and, in many cases, zero-knowledge. More precisely, the proof is always zero-knowledge for quadratic equations in $\mathbb{Z}_p$ and multi-exponentiation equations, and also for pairing product equations provided that $t = 1$.

QA-NIZK Arguments of Membership in Linear Spaces [20]. We describe some languages for which there exist constant-size QA-NIZK arguments of membership which will be used as building blocks in our constructions. These languages are (i) linear subspaces of $\mathbb{G}_\gamma^m$, $\gamma \in \{1, 2\}$ [21, 22], and (ii) bilateral linear subspaces, that is, linear subspaces of $\mathbb{G}_1^m \times \mathbb{G}_2^n$ [12]. For $\gamma \in \{1, 2\}$,

$$\mathcal{L}_{[\mathbf{M}]_\gamma} := \{[\boldsymbol{x}]_\gamma \in \mathbb{G}_\gamma^n : \exists \boldsymbol{w} \in \mathbb{Z}_q^t,\ \boldsymbol{x} = \mathbf{M}\boldsymbol{w}\}, \tag{i}$$

$$\mathcal{L}_{[\mathbf{M}]_1,[\mathbf{N}]_2} := \{([\boldsymbol{x}]_1, [\boldsymbol{y}]_2) \in \mathbb{G}_1^m \times \mathbb{G}_2^n : \exists \boldsymbol{w} \in \mathbb{Z}_q^t,\ \boldsymbol{x} = \mathbf{M}\boldsymbol{w},\ \boldsymbol{y} = \mathbf{N}\boldsymbol{w}\}, \text{(ii)}$$

We use LS (BLS) to designate (bilateral) linear subspace proof systems for the languages $\mathcal{L}_{[\mathbf{M}]_\gamma}$ ($\mathcal{L}_{[\mathbf{M}]_1,[\mathbf{N}]_2}$). These proof systems verify strong soundness, which essentially means that they are sound even when the discrete logarithm of the matrices is given. This property is formally defined in González et al. [12].

Case (i) can be instantiated based on the Kernel Diffie-Hellman assumption 2, and the proof has size $|\mathbb{G}_\gamma|$, whereas (ii) can be based on the Split Kernel Diffie-Hellman assumption 3, and the proof has size $2|\mathbb{G}_1| + 2|\mathbb{G}_2|$.

3 Proving Satisfiability of Quadratic Equations

In this section we present a scheme in which soundness is based on the q-STSDH Assumption.

3.1 Arguments for Quadratic Equations from q-Assumptions

Intuition. Given $n, d \in \mathbb{N}$, the number of variables and equations, respectively, we build a proof system for the family of languages

$$\mathcal{L}_{\mathrm{quad},ck} = \left\{ ([\boldsymbol{c}]_1, \mathbf{V}, \boldsymbol{b}) \in \mathbb{G}_1^{2n} \times \mathbb{Z}_p^{n\times d} \times \mathbb{Z}_p^d \;\middle|\; \begin{array}{l} \exists \boldsymbol{a}, \boldsymbol{w} \in \mathbb{Z}_p^n \text{ s.t} \\ [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{w}) \text{ and} \\ \left\{\boldsymbol{a}^\top \boldsymbol{v}_j + b_j\right\}_{j=1}^d \in \{0, 2\} \end{array} \right\}$$

where $[\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{w})$ is a vector of ElGamal encryptions. This generalizes to any other perfectly binding commitment of the form $[\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}; \boldsymbol{w}) = [\mathbf{U}_1\boldsymbol{a} + \mathbf{U}_2\boldsymbol{w}]_1$ for $ck = ([\mathbf{U}_1]_1, [\mathbf{U}_2]_1)$, and $[\mathbf{U}_1]_1, [\mathbf{U}_2]_1$ are from a witness sampleable distribution.

We follow the approach of Danezis et al. [6] and encode the equations

$$\boldsymbol{a}^\top \boldsymbol{v}_j + b_j \in \{0, 2\}$$

into a *Square Span Program (SSP)*: we construct $n+1$ polynomials $v_0(X), \ldots, v_n(X)$ and a target polynomial $t(X)$, where $\deg(v_i) < \deg(t) = d$ for all $i \in \{0, \ldots, n\}$. This codification asserts that a witness $\boldsymbol{a}$ satisfies the set of equations if and only if $t(X)$ divides $p(X)$, where

$$p(X) = \left(v_0(X) + \sum_{i=1}^{n} a_i v_i(X)\right)^2 - 1.$$

The polynomials $v_i(X)$, $i \in \{1,\ldots,n\}$, are defined as the interpolation polynomials of the coefficients v_{ij} of $\mathbf{V}$ at $r_1,\ldots,r_d$, which are fixed, arbitrary, pairwise different points of $\mathbb{Z}_p$. Similarly, $v_0(X)$ is the interpolation polynomial of $b_j - 1$ at the same points. That is, if $\boldsymbol{v}_j$ is the jth column of $\mathbf{V}$,

$$\boldsymbol{a}^\top \boldsymbol{v}_j + b_j - 1 = \sum_{i=1}^{n} a_i v_{ij} + b_j - 1 = \sum_{i=1}^{n} a_i v_i(r_j) + v_0(r_j).$$

Note that the statement $Z \in \{0,2\}$ is equivalent to $(Z-1)^2 - 1 = 0$ and hence, the polynomial $p(X)$ interpolates the left side of this equation in $r_1,\ldots,r_d$ when Z is replaced by $\boldsymbol{a}^\top \boldsymbol{v}_j + b_j - 1$ for each $j \in \{1,\ldots,d\}$. The target polynomial $t(X) = \prod_{i=1}^{d}(X - r_i)$ is 0 at $r_1,\ldots,r_d$ and therefore encodes the right sides. This codification gives us the equivalence: the equations hold if and only if $t(X)$ divides $p(X)$.

Danezis et al. constructed a SNARK for this statement, "$t(X)$ divides $p(X)$", which is very efficient because it just checks that the divisibility relation holds at a single secret point $s \in \mathbb{Z}_p$ whose powers $[s]_1, [s]_2, \ldots, [s^d]_1, [s^d]_2$ are published in the CRS. That is, the proof essentially shows "in the exponent" that

$$p(s) = h(s)t(s),$$

where $h(X) = p(X)/t(X)$. When all the equations hold, $h(X)$ is a polynomial and the evaluation at s can be constructed as a linear combination of the powers of s in the CRS. When some equation does not hold, $h(X)$ is a rational function, and its evaluation at s is no longer efficiently computable from the CRS. The actual proof system has some additional randomization elements to achieve Zero-Knowledge, but its soundness follows from this argument.

In the scheme of Danezis et al., the prover outputs a perfectly hiding commitment to the witness. In the soundness proof, one uses a knowledge of exponent assumption to extract the witness in $\mathbb{Z}_p^n$ from the commitment. The witness is used to derive a reduction from breaking soundness to the d-TSDH Assumption. More precisely, it follows from the SSP characterization that if the equation with index j^* does not hold, then $p(X) = q(X)(X - r_{j^*}) + b$, for some $b \neq 0$. From the extracted value of the witness $\boldsymbol{a}$ one can identify at least one such j^* and also recover the coefficients of $q(X)$ and the value b in $\mathbb{Z}_p$. From the verification equation, the reduction can obtain

$$\left[\frac{p(s)}{s - r_{j^*}}\right]_T = \left[q(s) + \frac{b}{s - r_{j^*}}\right]_T \tag{1}$$

and using $b, q(s)$ derive $\left[\dfrac{1}{s - r_{j^*}}\right]_T$.

In other words, there are two ways in which the Danezis et al.'s scheme (as well as most other SNARKs) use knowledge assumptions: (a) extracting vectors of committed values from one single group element (beyond what is information-theoretically possible), and (b) extract in the base field, so computing discrete logarithms. Our goal is to avoid knowledge of exponent assumptions, so to circumvent (a) we change the scheme to include perfectly binding commitments to the witness. However, we still have to deal with (b), as our commitments to $\boldsymbol{a}$ can only be opened to $[\boldsymbol{a}]_\gamma \in \mathbb{G}_\gamma$. Therefore, we are no longer able to compute $[q(s)]_T$ since it requires to compute terms of the form $[a_i a_j s^k]_T$ from $[a_i]_1, [a_j]_2$ and powers of s in one of the groups, in any case it would be a multiplication of three group elements.

At this point, we would like to be able to include in the proof a commitment that allows the reduction to extract $q(s)$, but the fact that $q(s)$ is "quadratic" in the witness makes this difficult. For this reason, we factor $q(X)$ into two polynomials $q_1(X)$ and $q_2(X)$. In the soundness game we will program the CRS[3] to depend on an index j^* and let the prover compute binding commitment to $[q_2(s)]_2$, while $[q_1(s)]_1$ can be directly computed from the proof. From these factors we are able to compute $[q(s)]_T$. However, extracting b in $\mathbb{Z}_p$ to obtain a reduction to the q-TSDH problem seems difficult, so we will rely on a more flexible security

[3] This is why we lose a factor $1/d$ in the soundness reduction.

assumption where we do not need to remove b. The idea of the new assumption is to give the adversary powers of s in the source groups and ask the adversary to output

$$\left(r_{j^*}, [\beta]_1, \left[\frac{b}{s-r_{j^*}}\right]_T\right), \text{ where } \beta^2 - 1 = b.$$

However, this is not a hard problem, as the adversary can set b as a combination of $s - r_{j^*}$ to achieve elimination of the denominator in $\frac{b}{s-r_{j^*}}$. For example, if an adversary sets $\beta = s - r_{j^*} + 1$, it can compute a valid solution as $(r_{j^*}, [\beta]_1, [s - r_{j^*} + 2]_T)$. To prevent this type of attacks from happening, we add an element $[\varepsilon]_2 \in \mathbb{G}_2$ to the challenge, and ask the adversary to output $[\varepsilon\beta]_2$ too, so that β cannot be set as a function of s (since the adversary will not be able to compute εs in $\mathbb{G}_2$). We call the modified assumption the q-STSDH, which is proven to be generically secure (see Sect. B). Further, it can be easily checked that the assumption is falsifiable as we note in Sect. 2.2. To make sure that we can extract $[\varepsilon\beta]_2$ from the prover's output and also that the rest of the elements of the proof are of the right form, we will require the prover to show that its output is in a given linear space.

Scheme description. Given $n, d \in \mathbb{N}$ we construct a QA-NIZK argument for the language $\mathcal{L}_{\text{quad},ck}$.

Setup.

- Algorithm $\mathsf{K}_0(gk, n, d)$ samples $ck = [\boldsymbol{u}]_1 \leftarrow \mathcal{L}_1$. A commitment $\mathsf{Com}_{ck}(\boldsymbol{a}; \boldsymbol{w})$ is the concatenation of $\mathsf{Enc}_{ck}(a_i; w_i) = [a_i \boldsymbol{e}_2 + w_i \boldsymbol{u}]_1$. That is, $\mathsf{Com}_{ck}(\boldsymbol{a}; \boldsymbol{w}) = [\mathbf{U}_1 \boldsymbol{a} + \mathbf{U}_2 \boldsymbol{w}]_1$, where $\mathbf{U}_1, \mathbf{U}_2$ are $2n \times n$ matrices such that $\mathbf{U}_1$ has $\boldsymbol{e}_2$ in the diagonal and $[\mathbf{U}_2]_1$ has $\boldsymbol{u}$ in the diagonal.
- Algorithm $\mathsf{K}_1(gk, ck, n, d)$ picks $s \leftarrow \mathbb{Z}_p$, $\left\{\hat{\boldsymbol{\phi}}_i\right\}_{i\in\{1,\ldots,n+1\}} \leftarrow \mathbb{Z}_p^3$, $\mathbf{Q}_2 \leftarrow \mathcal{U}_{3,3}$ and generates also the CRS for proving membership in bilateral linear spaces of Sect. 2, $\mathsf{BLS.CRS}$, for the linear spaces generated by the matrices:

$$[\mathbf{M}]_1 = \left[\begin{array}{ccc|ccc|c} \boldsymbol{e}_2 & & & \boldsymbol{u} & & & \\ & \ddots & & & \ddots & & \mathbf{0} \\ & & \boldsymbol{e}_2 & & & \boldsymbol{u} & \\ \hline v_1(s) & \ldots & v_n(s) & \mathbf{0} & & & t(s) \quad \mathbf{0} \end{array}\right]_1 \in \mathbb{G}_1^{(2n+1)\times(2n+4)},$$

$$[\mathbf{N}]_2 = \left[\begin{array}{ccc|c|cc} v_1(s) & \ldots & v_n(s) & \mathbf{0} & t(s) & \mathbf{0} \\ \hat{\boldsymbol{\phi}}_1 & \ldots & \hat{\boldsymbol{\phi}}_n & & \hat{\boldsymbol{\phi}}_{n+1} & \mathbf{Q}_2 \end{array}\right]_2 \in \mathbb{G}_2^{4\times(2n+4)}.$$

  The CRS includes the elements

$$\left(gk, ck, \left\{\left[s^i\right]_{1,2}\right\}_{i\in\{1,\ldots,d\}}, \left\{\left[\hat{\boldsymbol{\phi}}_i\right]_2\right\}_{i\in\{1,\ldots,n+1\}}, [\mathbf{Q}_2]_2, \mathsf{BLS.CRS}\right).$$

Prover. The prover P with input $(\text{CRS}, [\boldsymbol{c}]_1, \mathbf{V}, \boldsymbol{b}, \boldsymbol{a})$ picks $\delta \leftarrow \mathbb{Z}_p, \boldsymbol{r}_{q.2} \leftarrow \mathbb{Z}_p^3$ and defines the polynomial

$$p(X) = \left(v_0(X) + \sum_{i=1}^{n} a_i v_i(X) + \delta t(X)\right)^2 - 1 \in \mathbb{Z}_p[X],$$

where each $v_i(X)$, for $i \in \{1, \ldots, n\}$, is the interpolation polynomial of the components v_{ij} of $\mathbf{V}$ at points r_j, for $j \in \{1, \ldots, d\}$, and $v_0(X)$ is the interpolation polynomial of $b_j - 1$ at the same points. It then computes $h(X) = \frac{p(X)}{t(X)}$, which is a polynomial in $\mathbb{Z}_p[X]$ because $\boldsymbol{a}$ satisfies the equations, and the following elements:

$$[V]_1 = \left[\textstyle\sum_{i=1}^n a_i v_i(s) + \delta t(s)\right]_1 \qquad [V]_2 = \left[\textstyle\sum_{i=1}^n a_i v_i(s) + \delta t(s)\right]_2$$
$$[H]_1 = [h(s)]_1 \qquad [\boldsymbol{q}_2]_2 = \left[\textstyle\sum_{i=1}^n a_i \hat{\boldsymbol{\phi}}_i + \delta \hat{\boldsymbol{\phi}}_{n+1} + \mathbf{Q}_2 \boldsymbol{r}_{q.2}\right]_2.$$

The prover can compute all these elements as linear combinations of the powers of s in the CRS. The prover also computes a BLS proof ψ of

$$([\boldsymbol{c}]_1, [V]_1, [V]_2, [\boldsymbol{q}_2]_2)^\top \in \mathbf{Im}\begin{pmatrix}[\mathbf{M}]_1\\ [\mathbf{N}]_2\end{pmatrix}$$

with witness $(\boldsymbol{a}, \boldsymbol{w}, \delta, \boldsymbol{r}_{q.2})^\top \in \mathbb{Z}_p^{2n+4}$.

Finally, it sends the proof π to the verifier, where $\pi := \Big([H]_1, [V]_{1,2}, [\boldsymbol{q}_2]_2, \psi\Big)$.

Verifier. The verifier V with input $(\mathrm{CRS}, [\boldsymbol{c}]_1, \mathbf{V}, \boldsymbol{b}, \pi)$ checks whether the equation

$$e([v_0(s)+V]_1, [v_0(s)+V]_2) - [1]_T = e([H]_1, [t(s)]_2) \tag{2}$$

holds and $\mathsf{BLS.verify}(\psi) = 1$. If both conditions hold, it returns 1, else it returns 0.

Completeness. This property is based on the perfect completeness of membership in bilateral spaces, and the observation that the left hand side of the verification equation is $e([v_0(s)+V]_1, [v_0(s)+V]_2) - [1]_T = \left[(v_0(s)+V)^2 - 1\right]_T = [p(s)]_T$, and the right hand side is $e([H]_1, [t(s)]_2) = e([h(s)]_1, [t(s)]_2) = [p(s)]_T$.

Soundness. We introduce two technical lemmas that we will use in the following to prove the soundness of the scheme. Recall that $\mathcal{U}_{\ell,k}$ is the uniform distribution on $\ell \times k$ matrices over $\mathbb{Z}_p$. We define $\mathcal{U}_{\ell,k,r}$ to be the uniform distribution on $\ell \times k$ matrices over $\mathbb{Z}_p$ with rank r.

Lemma 1. *For any $k, r \in \mathbb{N}, r < k$, there exists an $\mathcal{L}_1$-$\mathsf{MDDH}_{\mathbb{G}_1}$ PPT adversary $\mathcal{B}_0$ such that for any PPT adversary $\mathcal{A}$*

$$\Pr[\mathbf{M} \leftarrow \mathcal{U}_{k,k,r+1} : \mathcal{A}([\mathbf{M}]_1) = 1] - \Pr[\mathbf{M} \leftarrow \mathcal{U}_{k,k,r} : \mathcal{A}([\mathbf{M}]_1) = 1]| \leq \mathrm{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_1}(\mathcal{B}_0).$$

Proof. Direct application of Theorem 1 of [30].

Lemma 2. *Let $v(X)$ be a polynomial in $\mathbb{Z}_p[X]$. For any $r \in \mathbb{Z}_p$, we define $q_2(X)$ and β as the quotient and remainder, respectively, of the polynomial division of $v(X)$ by $X - r$, i.e. $v(X) = q_2(X)(X-r) + \beta$. If $p(X) = v(X)^2 - 1$, then*

$$p(X) = (v(X) + \beta)\, q_2(X)(X-r) + \beta^2 - 1.$$

Proof. By definition, $p(X) = v(X)^2 - 1$, if we expand this expression using the definition of $q_2(X)$ we have:

$$\begin{aligned} p(X) =& v(X)\,(q_2(X)(X-r)+\beta) - 1 = v(X)q_2(X)(X-r) + v(X)\beta - 1\\ =& v(X)q_2(X)(X-r) + q_2(X)(X-r)\beta + \beta^2 - 1\\ =& (v(X)+\beta)q_2(X)(X-r) + \beta^2 - 1. \ \square \end{aligned}$$

Theorem 1. *Let $\mathsf{Adv}_{\mathrm{Sound}}(\mathcal{A})$ be the advantage of any PPT adversary $\mathcal{A}$ against the soundness of the scheme. There exist PPT adversaries $\mathcal{B}_1, \mathcal{B}_3$ against the $\mathcal{L}_1$-$\mathsf{MDDH}_{\mathbb{G}_2}$ and d-STSDH Assumptions, respectively, and an adversary $\mathcal{B}_2$ against strong soundness of the BLS proof such that*

$$\mathsf{Adv}_{\mathrm{Sound}}(\mathcal{A}) \leq d\Big(2\mathsf{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_2}(\mathcal{B}_1) + \mathsf{Adv}_{\mathsf{BLS}}(\mathcal{B}_2) + \mathsf{Adv}_{d\text{-}\mathsf{STSDH}}(\mathcal{B}_3)\Big).$$

Proof. In order to prove soundness we will prove indistinguishability of the following games.

- Real: This is the real soundness game. The output is 1 if the adversary produces a false accepting proof, i.e. if there is some equation $\boldsymbol{a}^\top \boldsymbol{v}_i + b_i \notin \{0, 2\}$ and the verifier accepts the proof.
- Game_0: This game is identical to the previous one, except that the commitment key $\boldsymbol{u}$ is chosen by the game.

– Game_1: This game is identical to the previous one, except that some $j^* \leftarrow \{1,\ldots,d\}$ is chosen and the game aborts if $\boldsymbol{a}$ satisfies the j^*-th equation, i.e. $[\boldsymbol{a}]_1^\top \boldsymbol{v}_{j^*} + [b_{j^*}]_1 \in \{[0]_1, [2]_1\}$.
– Game_2: For $r = r_{j^*}$ and $i \in \{1,\ldots,n+1\}$ let $\alpha_i(X)$ and β_i be the quotient and the reminder of the polynomial division of $v_i(X)$ by $X - r_{j^*}$ if $i \in \{1,\ldots,n\}$, and of $t(X)$ by $X - r_{j^*}$ if $i = n+1$. This game is identical to the previous one, except that $\mathbf{Q}_2$ is now a uniformly random matrix conditioned on having rank 1, and each $\left[\hat{\boldsymbol{\phi}}_i\right]_2$ is changed to

$$\left[\hat{\boldsymbol{\phi}}_i\right]_2 = [\alpha_i(s)]_2 \boldsymbol{e}_2 + \beta_i [\varepsilon]_2 \boldsymbol{e}_3 + [\mathbf{Q}_2]_2 \boldsymbol{r}_i,$$

where $\varepsilon \leftarrow \mathbb{Z}_p, \boldsymbol{r}_i \leftarrow \mathbb{Z}_p^3$ and $\boldsymbol{e}_i$ is the ith vector of the canonical basis of $\mathbb{Z}_p^3$.

Obviously, the games Real and Game_0 are indistinguishable.

Lemma 3. $\Pr[\mathsf{Game}_0(\mathcal{A}) = 1] \leq d \cdot \Pr[\mathsf{Game}_1(\mathcal{A}) = 1]$.

Proof. If $\mathcal{A}$ breaks soundness, at least one equation does not hold. Thus the challenger has at least a probability of $\frac{1}{d}$ of guessing this equation. □

Lemma 4. *There exists a $\mathcal{L}_1$-$\mathsf{MDDH}_{\mathbb{G}_2}$ adversary $\mathcal{B}_1$ such that*

$$|\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] - \Pr[\mathsf{Game}_2(\mathcal{A}) = 1]| \leq 2\mathrm{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_2}(\mathcal{B}_1).$$

Proof. We construct an adversary $\mathcal{B}_1$ against the $\mathcal{U}_{3,3}$-rank problem in which it receives $[\mathbf{Q}_2]_2 \in \mathbb{G}_2^{3\times 3}$ as input and must decide if the matrix has rank 1 or rank 3. $\mathcal{B}_1$ constructs the elements of CRS as in the previous game, but it uses $[\mathbf{Q}_2]_2$ as commitment key and defines $[\hat{\boldsymbol{\phi}}_i]_2$ as:

$$\left[\hat{\boldsymbol{\phi}}_i\right]_2 = [\alpha_i(s)]_2 \boldsymbol{e}_2 + [\mathbf{Q}_2]_2 \boldsymbol{r}_i, \qquad \text{where } \boldsymbol{r}_i \leftarrow \mathbb{Z}_p^3.$$

If $[\mathbf{Q}_2]_2$ has full rank, then $[\mathbf{Q}_2]_2 \boldsymbol{r}_i$ is a uniformly distributed element of $\mathbb{G}_2^3$, so adversary perfectly simulates Game_1, else it perfectly simulates Game_2.

We conclude by using the reduction between the rank problem and the $\mathcal{L}_1$-$\mathsf{MDDH}_{\mathbb{G}_2}$ problem, as established in Lemma 1. □

Lemma 5. *There exists an adversary $\mathcal{B}_2$ against the strong soundness of the* BLS *proof and a d-*STSDH *adversary $\mathcal{B}_3$ such that*

$$\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \leq \mathsf{Adv}_{\mathsf{BLS}}(\mathcal{B}_2) + \mathsf{Adv}_{d\text{-}\mathsf{STSDH}}(\mathcal{B}_3).$$

Proof. For any adversary which breaks soundness $\mathcal{A}$, let E be the event that $([\boldsymbol{c}]_1, [V]_1, [V]_2, [\boldsymbol{q}_2]_2)^\top \in \mathbf{Im}\begin{pmatrix}[\mathbf{M}]_1\\ [\mathbf{N}]_2\end{pmatrix}$ of Sect. 2 and $\overline{E}$ be the complementary event. Obviously,

$$\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \leq \Pr[\mathsf{Game}_2(\mathcal{A}) = 1 | E] + \Pr[\mathsf{Game}_2(\mathcal{A}) = 1 | \overline{E}]. \tag{3}$$

We can bound the second summand by the advantage of an adversary $\mathcal{B}_2$ against the strong soundness of BLS. Such an adversary receives $[\mathbf{M}]_1, [\mathbf{N}]_2$ sampled according to the distribution specified by Game_3 and the witness that proves that $\mathbf{M}, \mathbf{N}$ are sampled according to this distribution, which is s (see strong soundness, defined in Sect. 2.3). It also generates the $\mathsf{BLS.CRS}$, and the rest of the CRS is chosen in the usual way. Adversary $\mathcal{B}_2$ can use the output of $\mathcal{A}$ to break the soundness of BLS in a straightforward way.

In the following, we bound the first term of the sum in equation (3) by constructing an adversary $\mathcal{B}_3$ which breaks the d-STSDH Assumption in the case that E happens. Note that in this case there exists a witness $(\boldsymbol{a}, \boldsymbol{w}, \delta, \boldsymbol{r}_{q.2})^\top$ of membership in $\mathbf{Im}\begin{pmatrix}[\mathbf{M}]_1\\ [\mathbf{N}]_2\end{pmatrix}$. Further, this witness is partially unique, because $[\boldsymbol{c}]_1$ is a perfectly binding commitment, so $\boldsymbol{a}, \boldsymbol{w}, \delta$ are uniquely determined, and in particular this uniquely determines the polynomial $p(X)$.

We now describe the full reduction. Adversary $\mathcal{B}_3$ receives a challenge of the d-STSDH Assumption and plugs it in the CRS. The rest of the elements are chosen by adversary $\mathcal{B}_3$ with the distribution specified by the game. The CRS is then sent to the soundness adversary $\mathcal{A}$, who eventually outputs π for the corresponding $[\boldsymbol{c}]_1$.

Adversary $\mathcal{B}_3$ extracts $[\boldsymbol{a}]_1 \in \mathbb{G}_1$ from the knowledge of $\boldsymbol{u} \in \mathbb{Z}_p^2$ and aborts if the j^*-th equation is satisfied. By definition $e([v_0(s)+V]_1, [v_0(s)+V]_2) - [1]_T = [p(s)]_T$. If we divide both sides of the verification equation (2) by $s - r_{j^*}$,

$$\left[\frac{p(s)}{s-r_{j^*}}\right]_T = e\left([H]_1, \left[\frac{t(s)}{s-r_{j^*}}\right]_2\right) = e\left([H]_1, \left[\prod_{i\neq j^*}(s-r_i)\right]_2\right), \tag{4}$$

so the adversary $\mathcal{B}_3$ can compute $\left[\frac{p(s)}{s-r_{j^*}}\right]_T$ from $[H]_1$ and the powers of $[s]_{1,2}$ in the CRS. On the other hand, if we apply Lemma 2 to $p(X)$, we have

$$\left[\frac{p(s)}{s-r_{j^*}}\right]_T = \left[(v(s)+\beta)q_2(s) + \frac{\beta^2-1}{s-r_{j^*}}\right]_T, \tag{5}$$

and we have $\beta^2 - 1 \neq 0$ (otherwise the j^*-th equation is satisfied, in which case the game aborts). We describe in the following how $\mathcal{B}_3$ can compute right side of (5) and the elements to break the d-STSDH Assumption.

$\mathcal{B}_3$ can compute $[\beta]_1 = \sum_{i=0}^{n}[a_i]_1\beta_i$ and also $[v(s)+\beta]_1 = [V]_1 + [\beta]_1$, because it knows $[V]_1$ from the proof π and the extracted values $[a_i]_1$, and β_i are the reminders of dividing $v_i(X)$ by $X - r_{j^*}$.

Since $\mathcal{B}_3$ sampled $\mathbf{Q}_2$ itself, it can recover $[q_2(s)]_2$ and $[\varepsilon\beta]_2$ from $[\boldsymbol{q}_2]_2$ because it can compute two vectors $\boldsymbol{v}_2, \boldsymbol{v}_3 \in \mathbb{Z}_p^3$ such that $\boldsymbol{v}_i^\top[\mathbf{Q}_2]_2 = \mathbf{0}$, $\boldsymbol{v}_i^\top \boldsymbol{e}_j = 0$ if $i \neq j$ and $\boldsymbol{v}_i^\top \boldsymbol{e}_j = 1$ if $i = j$. $\mathcal{B}_3$ multiplies these vectors by $\boldsymbol{q}_2$ (which is correctly computed, because E holds), resulting in:

$$\boldsymbol{v}_2^\top[\boldsymbol{q}_2]_2 = \left[\boldsymbol{v}_2^\top \sum_{i=1}^{n+1} a_i\left(\alpha_i(s)\boldsymbol{e}_2 + \beta_i\varepsilon\boldsymbol{e}_3 + \mathbf{Q}_2\boldsymbol{r}_i\right) + \boldsymbol{v}_2^\top\mathbf{Q}_2\boldsymbol{r}_{q.2}\right]_2 = \left[\sum_{i=1}^{n+1} a_i\alpha_i(s)\right]_2,$$
$$\boldsymbol{v}_3^\top[\boldsymbol{q}_2]_2 = \left[\sum_{i=1}^{n+1} a_i\beta_i\varepsilon\right]_2.$$

From these values, $\mathcal{B}_3$ can compute $[q_2(s)]_2$ and $[\varepsilon\beta]_2$ by adding $[\alpha_0(s)]_2$ and $\beta_0[\varepsilon]_2$ to the above extracted elements, respectively:

$$\left[\alpha_0(s) + \sum_{i=1}^{n+1} a_i\alpha_i(s)\right]_2 = [q_2(s)]_2, \qquad \beta_0[\varepsilon]_2 + \left[\varepsilon\sum_{i=1}^{n+1} a_i\beta_i\right]_2 = [\varepsilon\beta]_2.$$

From these values and $[v(s)+\beta]_2$, computed above, $\mathcal{B}$ can derive $[(v(s)+\beta)q_2(s)]_T$ as $e([v(s)+\beta]_1, [q_2(s)]_2)$, and from equation (5) recover $\left[\frac{\beta^2-1}{s-r_{j^*}}\right]_T$.

Finally, $\mathcal{B}_3$ returns $\left(r_{j^*}, [\beta]_1, [\varepsilon\beta]_2, \left[\frac{\beta^2-1}{s-r_{j^*}}\right]_T\right)$, breaking the d-STSDH Assumption. □

Zero-Knowledge. We describe the simulation algorithm $(\mathsf{S}_1, \mathsf{S}_2)$. $\mathsf{S}_1(gk)$ outputs $(\mathrm{CRS}, \tau = \{s\}, \tau_{\mathsf{BLS}})$, the common reference string computed in the usual way plus the simulation trapdoor $s \in \mathbb{Z}_p$ and the simulation trapdoor of the bilateral spaces membership proof.

Simulator $\mathsf{S}_2(\mathrm{CRS}, [\boldsymbol{c}]_1, \tau, \tau_{\mathsf{BLS}})$*:* This algorithm samples $V^S \in \mathbb{Z}_p$, $\left[\boldsymbol{q}_2^S\right]_2 \leftarrow \mathbb{G}_2^3$, and defines:

$$\left[H^S\right]_1 = \left[\frac{(V^S)^2 - 1}{t(s)}\right]_1.$$

S also constructs $\psi^S \leftarrow \mathsf{BLS.simulator}(\mathrm{CRS}, [\boldsymbol{c}]_1, [V^S]_1, [V^S]_2, [\boldsymbol{q}_2^S]_2, \tau_{\mathsf{BLS}})$. The algorithm outputs $\pi := ([\boldsymbol{c}]_1, [V^S]_1, [V^S]_2, [\boldsymbol{q}_2^S]_2, \psi^S)$.

Theorem 2. *The scheme above is Perfect Zero-Knowledge.*

Proof. The key idea behind the proof is that all its the elements can be seen as perfectly hiding commitments to $\boldsymbol{a}$, where $\boldsymbol{a}$ is the opening of $[\boldsymbol{c}]_1$. For any V^S and any $\boldsymbol{a}$, there always exists a compatible δ. Further, since $\mathbf{Q}_2$ has full rank, $[\boldsymbol{q}_2^S]_2$ is compatible with any values $\boldsymbol{a}$, δ. $[H^S]_1$ is uniquely determined by V^S and the rest of the elements of the CRS. Finally, perfect zero-knowledge follows from the perfect zero-knowledge property of the bilateral space membership proof. □

3.2 Unit Vector from Static Assumptions

Given n, we build a proof system for the family of languages

$$\mathcal{L}_{\mathrm{uv},ck} = \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^{2n} \middle| \begin{array}{l} \exists \boldsymbol{a}, \boldsymbol{w} \in \mathbb{Z}_p^n \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{w}), \\ \boldsymbol{a} \in \{0,1\}^n \text{ and } \sum_{j=1}^{n} a_j = 1 \end{array} \right\},$$

where Com_{ck} is a perfectly binding commitment scheme, with ck chosen from a witness samplable distribution $\mathcal{D}_\rho$. For simplicity, we assume that $[\boldsymbol{c}]_1$ is a vector of ElGamal encryptions as in the previous schemes.

Alternatively, to better match the description of the language $\mathcal{L}_{\mathrm{quad},ck}$ given in Sect. 3.1, we can also define this language as:

$$\mathcal{L}_{\mathrm{uv},ck} = \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^{2n} \middle| \begin{array}{l} \exists \boldsymbol{a}, \boldsymbol{w} \in \mathbb{Z}_p^n \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{w}), \\ \boldsymbol{a}^\top \mathbf{V} + \boldsymbol{b}^\top \in \{0,2\}^{n+1} \end{array} \right\},$$

where

$$\mathbf{V} = \begin{pmatrix} 2 & & 1 \\ & \ddots & \vdots \\ & & 2\ 1 \end{pmatrix}, \qquad \boldsymbol{b} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{pmatrix}.$$

That is, $\mathbf{V} = (2\mathbf{I}_n | \mathbf{1})$ and $\boldsymbol{b} = (\mathbf{0}_n | 1)^\top$. In particular, this is a special case of the language $\mathcal{L}_{\mathrm{quad},ck}$, with $\mathbf{V} = (v_{ij})$ and $\boldsymbol{b}$ fixed.

Our argument for this language is almost identical to the argument in Sect. 3.1, except that we use a dual point of view and now the points $\mathcal{R} = \{r_1, \ldots, r_{n+1}\}$ are published only in the exponent, while $s \in \mathbb{Z}_p$ can be public[4]. We remark that this change affects crucially the information that must be included in the CRS to allow the prover to compute $[H]_1$, the quotient of dividing $p(s)$ by $t(s)$. In the general case (for any $\mathbf{V}, \boldsymbol{b}$), this information would be quadratic in n after this change. On the other hand, the advantage of this approach is that soundness is based on a static assumption. The intuition behind it is that if the points $r_1, \ldots, r_{n+1}$ are random and unrelated, one can reduce satisfiability of the jth equation to a computational problem which is only related to r_j and independent of the rest.

The fact that the CRS is quadratic makes the scheme less interesting in the general case. For this reason, we restrict this dual approach to the unit vector argument. A similar situation is found in the paper of González *et al.* [12], in which they provided a constant proof that a set of perfectly binding commitments to integers open to bits. In the general case, the common reference string was quadratic, while in the unit vector case it was linear.

[4] Actually it is not necessary for completeness, but it can be published without compromising security.

Intuition. Apart from the change of basis for computing $[H]_1$, another important but very technical difference with respect to the previous scheme is that we use a special form of interpolation. Given some points $\{r_1, \ldots, r_{n+1}\}$ we can define the polynomial $v_i(x) = \sum_{j=1}^{n+1} v_{ij}\tilde{\ell}_j(X)$ where $\tilde{\ell}_j(X) = \prod_{k\neq j} \frac{X - r_k}{r_j - r_k}$ is the (normalized) Lagrange interpolation polynomial for which $v_i(X)$ is the polynomial that at the point r_j goes through v_{ij}, that is the ijth matrix entry of $\mathbf{V}$ of Sect. 3.1. We want to prove security under static assumptions. So, we just want one point challenge instead of d as in the previous constructions, where the assumptions were not static. In our construction we need to compute the interpolation polynomials knowing all the interpolation points in $\mathbb{Z}_p$ but one, say r_{j^*}, that we know in the group $\mathbb{G}_\gamma$. The polynomials $\tilde{\ell}_j(x)$ are rational functions in terms of r_{j^*} and they are infeasible to compute in this situation. Our approach allows us to compute the interpolation polynomials as degree 1 polynomials in terms of r_{j^*}. We achieve that using the non-normalized Lagrange interpolation with polynomials $\ell_j(X) = \prod_{k\neq j}(X - r_k)$ for which $v_i(x)$ in point r_j goes through $\mu_j v_{ij}$, where $\mu_j = \prod_{k\neq j}(r_j - r_k)$.

As in Sect. 3.1 if we consider $\boldsymbol{Z} = \boldsymbol{a}^\top\mathbf{V} + \boldsymbol{b}^\top$, $\boldsymbol{Z}$ satisfies equations $\boldsymbol{Z} \in \{0, 2\}^{n+1}$ if and only if $(\boldsymbol{Z} - \mathbf{1})^2 = \mathbf{1}$. Given a set of points $\mathcal{R} = \{r_1, \ldots, r_{n+1}\}$, the non-normalized interpolation polynomials, $v_i(X)$ such that $v_i(r_j) = \mu_j v_{ij}$ for $i \in \{1, \ldots, n\}$ and $v_0(r_j) = \mu_j(b_j - 1)$ have a very specific form, namely

$$v_0(X) = -\sum_{i=1}^{n} \ell_i(X), \qquad v_i(X) = 2\ell_i(X) + \ell_{n+1}(X), \qquad \text{for } i \in \{1, \ldots, n\}.$$

With the definition of $\ell_i(X)$ that we are using, by a similar argument as in previous sections now the polynomial $p(X)$ is of the form:

$$p(X) = \left(-\sum_{i=1}^{n} \ell_i(X) + \sum_{i=1}^{n} a_i \left(2\ell_i(X) + \ell_{n+1}(X)\right)\right)^2 - \left(\sum_{i=1}^{n+1} \ell_i(X)\right)^2, \tag{6}$$

where $\sum_{i=1}^{n+1} \ell_i(X)$ is the interpolation polynomial that has value μ_i in each point r_i. The equation (6) is 0 in $\{r_1, \ldots, r_{n+1}\}$ if and only if all the equations are satisfied.

If $[\boldsymbol{c}]_1$ is in the language and $\boldsymbol{a}$ is its opening, there exists an index i^* such that $a_{i^*} = 1$ and $a_j = 0$ if $j \neq i^*$. Thus, substituting these values in the equation (6),

$$p(X) = \left(-\sum_{i=1, i\neq i^*}^{n} \ell_i(X) + \ell_{i^*}(X) + \ell_{n+1}(X)\right)^2 - \left(\sum_{i=1}^{n+1} \ell_i(X)\right)^2.$$

Consequently, in order to compute the polynomial $h(X) = \frac{p(X)}{t(X)}$, the prover would need products like $\ell_{j,i} := \frac{\ell_j(X)\ell_i(X)}{t(X)} = \prod_{k\neq i,j}(X - r_k)$. The trivial solution is to provide $\{\ell_{j,i}\}_{i,j=1}^{n+1}$ in the CRS, but this implies a quadratic CRS. Our approach allows to give just $n + 1$ combinations of ℓ_i as we can see in the following, which results in a linear CRS.

Again, the key difference with the scheme of Sect. 3.1 is that here we want the prover to know s in $\mathbb{Z}_p$ but not the interpolation points, so the way we compute H changes. We decompose $p(X)$ in a product of polynomials as follows. Let $\overline{v}(X) = v_0(X) + \sum_{i=1}^{n} a_i v_i(X) = -\sum_{i=1, i\neq i^*}^{n} \ell_i(X) + \ell_{i^*}(X) + \ell_{n+1}(X)$ and $k(X) = \sum_{i=1}^{n+1} \ell_i(X)$. Then,

$$p(X) = (\overline{v}(X) + k(X))(\overline{v}(X) - k(X)). \tag{7}$$

Note that

$$\begin{aligned}
\overline{v}(X) + k(X) &= -\sum_{i=1, i\neq i^*}^{n} \ell_i(X) + \ell_{i^*}(X) + \ell_{n+1}(X) + \sum_{i=1}^{n+1} \ell_i(X) = 2\left(\ell_{i^*}(X) + \ell_{n+1}(X)\right) \\
\overline{v}(X) - k(X) &= -\sum_{i=1, i\neq i^*}^{n} \ell_i(X) + \ell_{i^*}(X) + \ell_{n+1}(X) - \sum_{i=1}^{n+1} \ell_i(X) = -2\sum_{i=1, i\neq i^*}^{n} \ell_i(X).
\end{aligned} \tag{8}$$

Now we can use this decomposition to compute $h(X)$:

$$h(X) = \frac{(\overline{v}(X)+k(X))(\overline{v}(X)-k(X))}{t(X)} = \frac{-4\left(\sum_{i=1,i\neq i^*}^{n}\ell_i(X)\right)(\ell_{i^*}(X)+\ell_{n+1}(X))}{t(X)} \\ = -4\sum_{i=1,i\neq i^*}^{n}\ell_{i,i^*}(X)+\ell_{i,n+1}(X). \tag{9}$$

This $h(X)$ can be computed evaluated in s for any i^* using equation (9) by an honest prover who is given

$$\left\{\sum_{j=1,j\neq i}^{n}\ell_{j,i}(s)+\ell_{j,n+1}(s)\right\}_{i\in\{1,\ldots,n\}}$$

in the CRS.

Note that in the scheme, $h(X)$ is randomized with an additional term $\delta t(X)$ in $v(X)$, where $\delta \leftarrow \mathbb{Z}_p$, in order to get zero-knowledge.

Scheme description. Given $n \in \mathbb{Z}_p$ we construct a QA-NIZK argument for the language $\mathcal{L}_{\mathrm{uv},ck}$.

Setup.

- Algorithm $\mathsf{K}_0(gk,n)$ samples $ck=([\boldsymbol{u}]_1)$ from the 1-Lin distribution $\mathcal{L}_1$. A commitment $\mathsf{Com}_{ck}(\boldsymbol{a};\boldsymbol{w})$ is the concatenation of $\mathsf{Enc}_{ck}(a_i;w_i)=[a_i\boldsymbol{e}_2+w_i\boldsymbol{u}]_1$ of ElGamal encryptions.
- Algorithm $\mathsf{K}_1(gk,ck,n)$ picks $s,\{r_j\}_{j\in\{1,\ldots,n+1\}}\leftarrow\mathbb{Z}_p$, computes the non-normalized Lagrange interpolation polynomials $\ell_i(X)=\prod_{k\neq i}(X-r_k)$ using the points r_j as interpolation points, and evaluates $\ell_i(s)$, for $i\in\{1,\ldots,n+1\}$. It also defines $t(X):=\prod_{i=1}^{n+1}(X-r_i)$ and

$$\left\{L_i(s):=\sum_{j=1,j\neq i}^{n}\ell_{j,i}(s)+\ell_{j,n+1}(s)\right\}_{i\in\{1,\ldots,n\}}.$$

  It picks $\left\{\boldsymbol{\phi}_i,\hat{\boldsymbol{\phi}}_i\right\}_{i\in\{1,\ldots,n+1\}}\leftarrow\mathbb{Z}_p^2\times\mathbb{Z}_p^3$, $\mathbf{Q}_1\leftarrow\mathcal{U}_{2,2}$, $\mathbf{Q}_2\leftarrow\mathcal{U}_{3,3}$ and generates also the CRS for proving membership in bilateral linear spaces $\mathsf{BLS.CRS}$, for the linear space generated by the matrices:

$$[\mathbf{M}]_1=\left[\begin{array}{ccc|c|c} \boldsymbol{e}_2 & & & \boldsymbol{u} & \\ & \ddots & & \ddots & \mathbf{0} \\ & & \boldsymbol{e}_2 & & \boldsymbol{u} \\ \hline 2\ell_1(s)+\ell_{n+1}(s) & \ldots & 2\ell_n(s)+\ell_{n+1}(s) & \mathbf{0} & t(s)\ \mathbf{0}\ \mathbf{0} \\ \boldsymbol{\phi}_1 & \ldots & \boldsymbol{\phi}_n & & \boldsymbol{\phi}_{n+1}\ \mathbf{Q}_1\ \mathbf{0} \end{array}\right]_1 \in \mathbb{G}_1^{(2n+3)\times(2n+6)},$$

$$[\mathbf{N}]_2=\left[\begin{array}{ccc|c|c} 2\ell_1(s)+\ell_{n+1}(s) & \ldots & 2\ell_n(s)+\ell_{n+1}(s) & \mathbf{0} & t(s)\ \mathbf{0}\ \mathbf{0} \\ \hat{\boldsymbol{\phi}}_1 & \ldots & \hat{\boldsymbol{\phi}}_n & & \hat{\boldsymbol{\phi}}_{n+1}\ \mathbf{0}\ \mathbf{Q}_2 \end{array}\right]_2 \in \mathbb{G}_2^{4\times(2n+6)}.$$

  The CRS includes the elements

$$\Bigg(gk, ck, \left\{[\ell_i(s)]_{1,2}, [L_i(s)]_{1,2}, [\boldsymbol{\phi}_i]_1, \left[\hat{\boldsymbol{\phi}}_i\right]_2\right\}_{i\in\{1,\ldots,n+1\}}, [t(s)]_{1,2}, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \\ \mathsf{BLS.CRS}\Bigg).$$

Prover.

– The prover $\mathsf{P}(\mathrm{CRS}, [\boldsymbol{c}]_1, \mathbf{V}, \boldsymbol{b}, \boldsymbol{a})$ picks $\delta \leftarrow \mathbb{Z}_p, \boldsymbol{r}_{q.1}, \boldsymbol{r}_{q.2} \leftarrow \mathbb{Z}_p^2 \times \mathbb{Z}_p^3$ and computes

$$[V]_{1,2} = [2\ell_{i^*}(s) + \ell_{n+1}(s) + \delta t(s)]_{1,2}\,,$$

defines $p(s) = (v_0(s) + V + \sum_{i=1}^{n+1} \ell_i(s))(v_0(s) + V - \sum_{i=1}^{n+1} \ell_i(s))$ and $H = h(s) = \frac{p(s)}{t(s)}$. The prover can compute

$$[H]_1 = \left[-4L_{i^*}(s) + 2\delta(v_0(s) + V) - \delta^2 t(s)\right]_1 \text{ (see the intuition above)}$$

and the following elements:

$$[\boldsymbol{q}_1]_1 = [\textstyle\sum_{i=1}^{n} a_i \boldsymbol{\phi}_i + \delta\boldsymbol{\phi}_{n+1} + \mathbf{Q}_1 \boldsymbol{r}_{q.1}]_1\,, \qquad [\boldsymbol{q}_2]_2 = \left[\textstyle\sum_{i=1}^{n} a_i \hat{\boldsymbol{\phi}}_i + \delta\hat{\boldsymbol{\phi}}_{n+1} + \mathbf{Q}_2 \boldsymbol{r}_{q.2}\right]_2.$$

The prover also computes a BLS proof ψ that

$$([\boldsymbol{c}]_1, [V]_1\,, [\boldsymbol{q}_1]_1, [V]_2\,, [\boldsymbol{q}_2]_2)^\top \in \mathbf{Im}\begin{pmatrix} [\mathbf{M}]_1 \\ [\mathbf{N}]_2 \end{pmatrix},$$

with witness $(\boldsymbol{a}, \boldsymbol{w}, \delta, \boldsymbol{r}_{q.1}, \boldsymbol{r}_{q.2})^\top \in \mathbb{Z}_p^{2n+6}$. Finally, it sends the proof π to the verifier, where

$$\pi := ([H]_1\,, [V]_1\,, [V]_2\,, [\boldsymbol{q}_1]_1\,, [\boldsymbol{q}_2]_2\,, \psi)\,.$$

Verifier.

– The verifier $\mathsf{V}(\mathrm{CRS}, [\boldsymbol{c}]_1, \mathbf{V}, \boldsymbol{b}, \pi)$ checks whether the equation

$$e([v_0(s) + V]_1\,, [v_0(s) + V]_2) - e\left(\left[\sum_{i=1}^{n+1} \ell_i(s)\right]_1, \left[\sum_{i=1}^{n+1} \ell_i(s)\right]_2\right) = e([H]_1\,, [t(s)]_2) \tag{10}$$

holds, where $[v_0(s)]_1 = -\sum_{i=1}^{n} [\ell_i(s)]_1$, and $\mathsf{BLS.verify}(\psi) = 1$. If both conditions hold, it returns 1, else it returns 0.

Completeness. The reason why the prover can compute H is explained in the intuition. On the other hand, membership in bilateral spaces is perfectly complete. Further, the right hand side of the verification equation is $e\left([H]_1\,, [t(s)]_2\right) = e\left(\left[\frac{p(s)}{t(s)}\right]_1, [t(s)]_2\right) = [p(s)]_T$, while the left hand is $\left[(v_0(s) + V)^2 - \left(\sum_{i=1}^{n+1} \ell_i(s)\right)^2\right]_T = [p(s)]_T$.

Soundness.

Theorem 3. *Let $\mathsf{Adv}_{\mathrm{PS}}(\mathcal{A})$ be the advantage of any PPT adversary $\mathcal{A}$ against the soundness of the scheme. There exist PPT adversaries $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3, \mathcal{B}_4$ such that*

$$\mathsf{Adv}_{\mathrm{PS}}(\mathcal{A}) \leq (n+1)\Big(\mathsf{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_1}(\mathcal{B}_1) + 2\mathsf{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_2}(\mathcal{B}_2) + \mathsf{Adv}_{\mathsf{BLS}}(\mathcal{B}_3) + \mathsf{Adv}_{1\text{-}\mathsf{STSDH}}(\mathcal{B}_4)\Big).$$

Proof. In order to prove soundness we will prove indistinguishability of the following games.

- Real: This is the real soundness game. The output is 1 if the adversary produces a false accepting proof, i.e. if there is some equation $\boldsymbol{a}^\top \boldsymbol{v}_i + b_i \notin \{0, 2\}$ and the verifier accepts the proof.
- Game_0: This game is identical to the previous one, except that the commitment key $\boldsymbol{u}$ is chosen by the game.
- Game_1: This game is identical to the previous one, except that some $j^* \leftarrow \{1, \ldots, n+1\}$ is chosen and the game aborts if $\boldsymbol{a}$ satisfies the j^*-th equation. This can be checked by opening $\boldsymbol{c}$ thanks to knowledge of $\boldsymbol{u}$ and checking whether $[\boldsymbol{a}]_1^\top \boldsymbol{v}_{j^*} + [b_{j^*}]_1 \in \{[0]_1, [2]_1\}$.
- Game_2: This game is identical to the previous one, except that $\mathbf{Q}_1$ is now a uniformly random matrix conditioned on having rank 1. If $j^* \neq n+1$, the elements $[\boldsymbol{\phi}_i]_1$ are changed to

$$[\boldsymbol{\phi}_i]_1 = \begin{cases} [\mathbf{Q}_1]_1 \boldsymbol{r}_i, & i = 1, \ldots, n+1, i \neq j^* \\ [2\ell_{j^*}]_1 \boldsymbol{e}_1^2 + [\mathbf{Q}_1]_1 \boldsymbol{r}_{j^*}, & i = j^* \end{cases}$$

  where $\boldsymbol{r}_i, \boldsymbol{r}_{j^*}, \leftarrow \mathbb{Z}_p^2$ and $\boldsymbol{e}_1^2$ is the first element in the canonical basis of $\mathbb{Z}_p^2$, while if $j^* = n+1$, each $[\boldsymbol{\phi}_i]_1$ is changed to

$$[\boldsymbol{\phi}_i]_1 = \begin{cases} [\ell_{n+1}(s)]_1 \, \boldsymbol{e}_1^2 + [\mathbf{Q}_1]_1 \boldsymbol{r}_i, & i = 1, \ldots, n \\ [\mathbf{Q}_1]_1 \, \boldsymbol{r}_{n+1}, & i = n+1 \end{cases}$$

  where $\boldsymbol{r}_i, \boldsymbol{r}_{n+1} \leftarrow \mathbb{Z}_p^2$.
- Game_3: This game is identical to the previous one, except that $\mathbf{Q}_2$ is now a uniformly random matrix conditioned on having rank 1. If $j^* \neq n+1$, the elements $\left[\hat{\boldsymbol{\phi}}_i\right]_2$ are changed to

$$\left[\hat{\boldsymbol{\phi}}_i\right]_2 = \begin{cases} [2\ell_{i,j^*}(s) + \ell_{n+1,j^*}(s)]_2 \, \boldsymbol{e}_1^3 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_i, & i = 1, \ldots, n, i \neq j^* \\ [\varepsilon]_2 \, \boldsymbol{e}_3^3 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_{j^*} & i = j^* \\ [\ell_{j^*}(s)]_2 \, \boldsymbol{e}_1^3 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_{n+1} & i = n+1 \end{cases}$$

  where $\widetilde{\boldsymbol{r}}_i, \widetilde{\boldsymbol{r}}_{j^*}, \widetilde{\boldsymbol{r}}_{n+1} \leftarrow \mathbb{Z}_p^3$, $\varepsilon \leftarrow \mathbb{Z}_p$ and $\boldsymbol{e}_i^3$ is the ith element in the canonical basis of $\mathbb{Z}_p^3$, while if $j^* = n+1$, the elements $\left[\hat{\boldsymbol{\phi}}_i\right]_2$ are changed to

$$\left[\hat{\boldsymbol{\phi}}_i\right]_2 = \begin{cases} [2\ell_{i,n+1}(s)]_2 \, \boldsymbol{e}_1^3 + [\varepsilon]_2 \, \boldsymbol{e}_3^3 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_i, & i = 1, \ldots, n, i \neq j^* \\ [\ell_{n+1}(s)]_2 \, \boldsymbol{e}_1^3 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_{n+1}, & i = n+1 \end{cases}$$

  where $\widetilde{\boldsymbol{r}}_i, \widetilde{\boldsymbol{r}}_{n+1} \leftarrow \mathbb{Z}_p^3$, $\varepsilon \leftarrow \mathbb{Z}_p$.

Obviously, the games Real and Game_0 are indistinguishable. The indistinguishability of the other games is based on the rank problem as in the soundness proof in Sect. 3.1.

Lemma 6. *There exists an adversary $\mathcal{B}_3$ against the strong soundness of the* BLS *argument and an adversary $\mathcal{B}_4$ against the* 1-STSDH *assumption such that*

$$\Pr[\mathsf{Game}_3(\mathcal{A}) = 1] \leq \mathsf{Adv}_{\mathsf{BLS}}(\mathcal{B}_3) + \mathsf{Adv}_{\mathsf{1\text{-}STSDH}}(\mathcal{B}_4).$$

Proof. As in the proof of Lemma 5, we distinguish two events, the event that the adversary succeeds in giving a false proof of membership in bilinear spaces (event $\overline{E}$), and the complementary event E, which is the interesting part of the proof. It can be proved easily following an analogous argument that such an adversary $\mathcal{B}_3$ could break soundness of BLS. We describe the reduction for adversary $\mathcal{B}_4$ that receives a challenge $(gk, [\varepsilon]_2, [\alpha]_{1,2})$ of the 1-STSDH Assumption and plugs $(gk, [\varepsilon]_2)$ in the CRS. Adversary $\mathcal{B}_4$ chooses $s \leftarrow \mathbb{Z}_p$, and all the points r_i, $i \neq j^*$ as random elements in $\mathbb{Z}_p$. Then it implicitly sets r_{j^*} to be $s - r_{j^*} = \alpha$, so that r_{j^*} can be computed in $\mathbb{G}_1$ and $\mathbb{G}_2$ but not in $\mathbb{Z}_p$. Then it sets:

$$[\ell_j(s)]_{1,2} = \textstyle\prod_{i \neq j, j^*} (s - r_i) \, [\alpha]_{1,2} \text{ for } j \neq j^*, \qquad [\ell_{j^*}(s)]_{1,2} = \left[\textstyle\prod_{i \neq j^*} (s - r_i)\right]_{1,2},$$
$$[L_j(s)]_{1,2} = \textstyle\sum_{i=1, i \neq j}^{n} \left([\ell_{j,i}(s)]_{1,2} + [\ell_{n+1,i}(s)]_{1,2}\right), \quad [t(s)]_{1,2} = \textstyle\prod_{i \neq j^*} (s - r_i) \, [\alpha]_{1,2}$$

for $j \in \{1, \ldots, n+1\}$. The rest of the elements of the CRS are computed as sampled as specified by the game. All these elements are added to the CRS and sent to the soundness adversary $\mathcal{A}$, who eventually outputs π for the corresponding $[\boldsymbol{c}]_1$.

Adversary $\mathcal{B}_4$ extracts $[\boldsymbol{a}]_1 \in \mathbb{G}_1$ from $[\boldsymbol{c}]_1$ and the knowledge of $\boldsymbol{u} \in \mathbb{Z}_p^2$ and aborts if the j^*th equation is satisfied.

By definition $e([v_0(s)+V]_1, [v_0(s)+V]_2) - e\left(\left[\sum_{i=1}^{n+1} \ell_i(s)\right]_1, \left[\sum_{i=1}^{n+1} \ell_i(s)\right]_2\right) = [p(s)]_T$. We note that it can compute $\left[\frac{p(s)}{\alpha}\right]_T$ from $[H]_1$:

$$\left[\frac{p(s)}{\alpha}\right]_T = e\left([H]_1, \left[\frac{t(s)}{\alpha}\right]_2\right) = e([H]_1, [\ell_{j^*}(s)]_2) \tag{11}$$

by the verification equation (10) and $[\ell_{j^*}(s)]_2$ is efficiently computable by the adversary.

Moreover, $p(s)$ can be factored as $p(s) = (\overline{v}(s) + k(s))(\overline{v}(s) - k(s))$ for equation (9). We can write $\overline{v}(s) = (V_0 + \alpha V_1)$ and $k(s) = (K_0 + \alpha K_1)$, where V_0, K_0 are the terms which the adversary does not know how to divide by α. More specifically,

$$p(s) = (V_0 + \alpha V_1)^2 - (K_0 + \alpha K_1)^2, \tag{12}$$

and therefore

$$\begin{aligned}
\frac{p(s)}{\alpha} =& \left(\frac{V_0}{\alpha} + V_1\right)(V_0 + \alpha V_1) - \left(\frac{K_0}{\alpha} + K_1\right)(K_0 + \alpha K_1) \\
=& \frac{V_0^2 - K_0^2}{\alpha} + 2V_0V_1 + \alpha V_1^2 - 2K_0K_1 - \alpha K_1^2 \\
=& \frac{V_0^2 - K_0^2}{\alpha} + (2V_0 + \alpha V_1)\, V_1 - 2K_0K_1 - \alpha K_1^2 \\
=& \frac{V_0^2 - K_0^2}{\alpha} + (V_0 + V)\, V_1 - 2K_0K_1 - \alpha K_1^2
\end{aligned} \tag{13}$$

The BLS proof guarantees the existence of values $\boldsymbol{a}, \delta, \boldsymbol{r}_{q.1}, \boldsymbol{r}_{q.2}$, binding property of commitments c_i assure a_i are unique. So, the elements V_0, V_1, K_0, K_1 are uniquely determined. We remember here the polynomials $\overline{v}(X), k(X)$ that we have defined above in (8) but adding the randomization term in $\overline{v}(X)$:

$$\begin{aligned}
\overline{v}(X) =& \sum_{i=1}^{n} (2a_i - 1)\ell_i(X) + \sum_{i=1}^{n} a_i \ell_{n+1}(X) + \delta t(X), \\
k(X) =& \sum_{i=1}^{n+1} \ell_i(X),
\end{aligned}$$

for which the equation (6) holds. Assuming $j^* \neq n+1$, if we divide by $X - r_{j^*}$, we obtain

$$\begin{aligned}
\frac{\overline{v}(X)}{X - r_{j^*}} =& \frac{(2a_{j^*} - 1)\ell_{j^*}(X)}{X - r_{j^*}} + \sum_{i=1, i\neq j^*}^{n} (2a_i - 1)\ell_{i,j^*}(X) + \sum_{i=1}^{n} a_i \ell_{n+1,j^*}(X) + \delta \ell_{j^*}(X), \\
\frac{k(X)}{X - r_{j^*}} =& \frac{\ell_{j^*}(X)}{X - r_{j^*}} + \sum_{i=1, i\neq j^*}^{n+1} \ell_{i,j^*}(X),
\end{aligned} \tag{14}$$

where the first term that is not divisible by $X - r_{j^*}$ corresponds to V_0, K_0 in each equation, respectively when the polynomials are evaluated on s. The other terms of the equations correspond to V_1, K_1 respectively.

So, if $j^* \neq n+1$:

$$\begin{aligned}
V_0 &= (2a_{j^*} - 1)\ell_{j^*}(s) \\
V_1 &= \sum_{i=1, i\neq j^*}^{n} (2a_i - 1)\ell_{i,j^*}(s) + \left(\sum_{i=1}^{n} a_i\right)\ell_{n+1,j^*}(s) + \delta\ell_{j^*}(s) \\
K_0 &= \ell_{j^*}(s) \\
K_1 &= \sum_{i=1, i\neq j^*}^{n+1} \ell_{i,j^*}(s),
\end{aligned}$$

otherwise, if $j^* = n+1$:

$$\begin{aligned}
V_0 &= \sum_{i=1}^{n} a_i \ell_{n+1}(s) \\
V_1 &= \sum_{i=1}^{n} (2a_i - 1)\ell_{i,n+1}(s) + \delta\ell_{n+1}(s) \\
K_0 &= \ell_{n+1}(s) \\
K_1 &= \sum_{i=1}^{n} \ell_{i,n+1}(s).
\end{aligned}$$

In either case, $\mathcal{B}_4$ knows $[V]_{1,2}$ from the proof, $K_0, K_1 \in \mathbb{Z}_p$, we will now argue that V_0 can be computed in $\mathbb{G}_1$ from one of the extracted values of $[\boldsymbol{q}_1]_1$ and V_1 can be computed in $\mathbb{G}_2$ from the extracted values of $[\boldsymbol{q}_2]_2$. More specifically, remember that in this game if $j^* \neq n+1$

$$\begin{aligned}
[\boldsymbol{q}_1]_1 &= \left[\sum_{i=1, i\neq j^*}^{n} a_i \mathbf{Q}_1 \boldsymbol{r}_i + a_{j^*}\left(2\ell_{j^*}(s)\boldsymbol{e}_1^2 + \mathbf{Q}_1 \boldsymbol{r}_{j^*}\right) + \delta \mathbf{Q}_1 \boldsymbol{r}_{n+1} + \mathbf{Q}_1 \boldsymbol{r}_{q.1}\right]_1 \\
[\boldsymbol{q}_2]_2 &= \left[\sum_{i=1, i\neq j^*}^{n} a_i \left((2\ell_{i,j^*}(s) + \ell_{n+1,j^*}(s))\boldsymbol{e}_1^3 + \mathbf{Q}_2 \widetilde{\boldsymbol{r}}_i\right) + a_{j^*}\left(\varepsilon \boldsymbol{e}_3^3 + \mathbf{Q}_2 \widetilde{\boldsymbol{r}}_{j^*}\right)\right]_2 \\
&\quad + \left[\delta\left(\ell_{j^*}(s)\boldsymbol{e}_1^3 + \mathbf{Q}_2 \widetilde{\boldsymbol{r}}_{n+1}\right) + \mathbf{Q}_2 \boldsymbol{r}_{q.2}\right]_2,
\end{aligned}$$

if $j^* = n+1$

$$\begin{aligned}
[\boldsymbol{q}_1]_1 &= \left[\sum_{i=1}^{n} a_i \left(\ell_{n+1}(s)\boldsymbol{e}_1^2 + \mathbf{Q}_1 \boldsymbol{r}_i\right) + \delta \mathbf{Q}_1 \boldsymbol{r}_{n+1} + \mathbf{Q}_1 \boldsymbol{r}_{q.1}\right]_1 \\
[\boldsymbol{q}_2]_2 &= \left[\sum_{i=1}^{n} a_i \left(2\ell_{i,n+1}(s)\boldsymbol{e}_1^3 + \varepsilon \boldsymbol{e}_2^3 + \mathbf{Q}_2 \widetilde{\boldsymbol{r}}_i\right) + \delta\left(\ell_{n+1}(s)\boldsymbol{e}_1^3 + \mathbf{Q}_2 \widetilde{\boldsymbol{r}}_{n+1}\right) + \mathbf{Q}_2 \boldsymbol{r}_{q.2}\right]_2.
\end{aligned}$$

Since $\mathcal{B}_4$ sampled $\mathbf{Q}_1, \mathbf{Q}_2$ itself, it can extract the following values from $[\boldsymbol{q}_1]_1$ and $[\boldsymbol{q}_2]_2$ defining appropriate orthogonal vectors to theses matrices, similarly to the extraction explained in Lemma 5:

- if $j^* \neq n+1$, it extracts $[a_{j^*} 2\ell_{j^*}(s)]_1$, $\left[\sum_{i=1, i\neq j^*}^{n} a_i(2\ell_{i,j^*}(s) + \ell_{n+1,j^*}(s)) + \delta\ell_{j^*}(s)\right]_2$ and $[\varepsilon a_{j^*}]_2$.
- if $j^* = n+1$, it extracts $[\sum_{i=1}^{n} a_i \ell_{n+1}(s)]_1$, $[\sum_{i=1}^{n} 2a_i \ell_{i,n+1} + \delta\ell_{n+1}(s)]_2$ and $[\varepsilon \sum_{i=1}^{n} a_i]_2$.

From these values it can compute $[V_0]_1$, $[V_1]_2$ in both cases, and also defining $\beta := 2a_{j^*} - 1$ for $j^* \neq n+1$ and $\beta := \sum_{i=1}^{n} a_i$ for $j^* = n+1$, it can compute $[\varepsilon\beta]_2$ in both cases (if $j^* \neq n+1$, computes $2[\varepsilon a_{j^*}]_2 - [\varepsilon]_2$, otherwise it has extracted $[\varepsilon \sum_{i=1}^{n} a_i]_2$).

Combining $[V_0]_1$, $[V_1]_2$, $[\alpha]_{1,2}$ with K_0, K_1 it can substract from equation (11) the terms $[(V+V_0)V_1 + 2K_0K_1 + \alpha K_1^2]_T$ in equation (13), so the adversary can compute in $\mathbb{G}_T$:

$$\left[\frac{V_0^2 - K_0^2}{\alpha}\right]_T = \begin{cases} \left[\frac{(2a_{j^*}-1)^2-1}{\alpha}\ell_{j^*}^2(s)\right]_T, & j^* \neq n+1 \\ \left[\frac{(\sum_{i=1}^n a_i)^2-1}{\alpha}\ell_{n+1}^2(s)\right]_T, & j^* = n+1. \end{cases}$$

Since the adversary knows $\ell_{j^*}^2(s) \in \mathbb{Z}_p$ in both cases, it can compute:

$$\left[\frac{V_0^2 - K_0^2}{\alpha\ell_{j^*}^2(s)}\right]_T = \begin{cases} \left[\frac{(2a_{j^*}-1)^2-1}{\alpha}\right]_T, & j^* \neq n+1 \\ \left[\frac{(\sum_{i=1}^n a_i)^2-1}{\alpha}\right]_T, & j^* = n+1 \end{cases}$$

which is $\left[\frac{\beta^2-1}{\alpha}\right]_T$ in both cases. Finally, the adversary can return $\left(r_{j^*}, [\beta]_1, [\varepsilon\beta]_2, \left[\frac{\beta^2-1}{\alpha}\right]_T\right)$, which breaks the 1-STSDH Assumption.

□

Theorem 4. *The scheme above is Perfect Zero-Knowledge, i.e. there exists a simulator algorithm* S *who has access to the trapdoor* $\tau = \{s, r_1, \ldots, r_{n+1}\}$*, that constructs a simulated proof* π^S *such that it is statistically indistinguishable from the real proof* π*.*

The proof is analogous to the one of Theorem 2 of Sect. 3.1.

3.3 Detailed Efficiency Comparison

Below we give more detailed figures for a better comparison of our result of Sect. 3.1 and our bitstring argument with the results of [12].

	Language	Proof size	CRS size	Assumption
Sect. 3.1	Bitstring	$4\lvert\mathbb{G}_1\rvert + 6\lvert\mathbb{G}_2\rvert$	$(n+O(1))\lvert\mathbb{G}_1\rvert +$ $+(4n+O(1))\lvert\mathbb{G}_2\rvert$	q-STSDH 7
Sect. 5 of [12]		$10\lvert\mathbb{G}_1\rvert + 10\lvert\mathbb{G}_2\rvert$	$O(n^2)\lvert\mathbb{G}_1\rvert +$ $+O(n^2)\lvert\mathbb{G}_2\rvert$	SSDP
Sect. 3.2	Unit vector	$6\lvert\mathbb{G}_1\rvert + 6\lvert\mathbb{G}_2\rvert$	$(4(n+1)+O(1))\lvert\mathbb{G}_1\rvert +$ $+(5(n+1)+O(1))\lvert\mathbb{G}_2\rvert$	1-STSDH 7
Sect. 5 of [12]		$10\lvert\mathbb{G}_1\rvert + 10\lvert\mathbb{G}_2\rvert$	$(20n+O(1))\lvert\mathbb{G}_1\rvert +$ $+(18n+O(1))\lvert\mathbb{G}_2\rvert$	SSDP

Table 4. The table shows the proof sizes (not including commitments) for bitstrings and unit vectors of size n.

4 Aggregated Set Membership Arguments

In the construction of Sect. 3.1, if $\mathbf{V}$ is the identity matrix and $\boldsymbol{b} = \mathbf{0}$, the equations $\boldsymbol{a}\mathbf{V} + \boldsymbol{b} \in \{0,2\}^d$ just prove that each $a_i \in \{0,2\}$. In this section we consider a generalization and build a proof system which proves that some perfectly binding commitments open to $a_i \in \mathcal{Z} = \{z_1, \ldots, z_m\} \subset \mathbb{Z}_p$. The proof is constant-size and uses the Boneh-Boyen signature scheme (the basic scheme from [2, Sect. 4.3]) together with a technique

to aggregate quadratic equations similar to the one of Sect. 3.1 and inspired by the quadratic span programs of Gennaro et al. [10].

First, in Sect. 4.1, we describe how to construct an argument of membership for a single $a \in \mathcal{Z}$ and then in Sect. 4.2 we show how to aggregate the argument. Sect. A in the appendix shows how to apply these ideas to construct a range proof.

4.1 Non-Aggregated Set Membership Argument

Intuition. We build a constant-size proof of membership for polynomially-large sets in $\mathbb{Z}_p$ with linear CRS. The idea is to give in the common reference string Boneh-Boyen signatures to each element of the set. The proof of membership is just a proof of knowledge of a valid signature. Recall that $[\sigma]_2$ is a valid signature for x if and only if

$$e([\mathsf{sk} - x]_1, [\sigma]_2) - [1]_T = [0]_T.$$

The statement $x \in \mathcal{Z}$ is proven committing to x and to $[\sigma]_2 = \left[\frac{1}{\mathsf{sk}-x}\right]_2$, and giving a Groth-Sahai proof for the satisfiability of the verification equation.

The problem with this approach is that it is not possible to extract $x \in \mathbb{Z}_p$ from its Groth-Sahai commitment, but only $[x]_1 \in \mathbb{G}_1$. Therefore, it is not clear how to reduce soundness to the EUF-CMA security of Boneh-Boyen, as the reduction can only output a "relaxed form" of forgery $([x]_1, [\sigma]_2)$, for some $x \notin \mathcal{Z}$, instead of $(x, [\sigma]_2)$.[5].

It turns out that Boneh-Boyen signatures are not unforgeable when purported forgeries are pairs of the form $([x]_1, [\sigma]_2)$. The problem is that $[x]_1$ may be dependent of sk, whereas this is impossible when $x \in \mathbb{Z}_p$ must be given. Indeed, for any message of the form $[\mathsf{sk} - x]_1$ one might compute a forgery as $[1/x]_2$.

To solve this issue, we force the prover to commit to $[\varepsilon x]_1$, where the discrete logarithm of $[\varepsilon]_1$ remains hidden. Since $[\mathsf{sk} \cdot \varepsilon]_1$ is not given, the adversary cannot choose x to be a function of sk.

Scheme description. We give a proof of membership in $\mathcal{Z} = \{z_1, \ldots, z_m\} \subset \mathbb{Z}_p$. More precisely, we build a proof for the family of languages:

$$\mathcal{L}_{\mathsf{memb},\mathcal{Z},ck} := \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^2 \,\middle|\, \exists \boldsymbol{w} \in \mathbb{Z}_p \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(x; w) \text{ and } x \in \mathcal{Z} \right\}.$$

Setup. Parameters for the Boneh-Boyen signatures are generated. Choose $\varepsilon \leftarrow \mathbb{Z}_p$. The CRS contains $[\varepsilon]_2$, signatures $[\sigma_j]_2 = \left[\frac{1}{\mathsf{sk}-z_j}\right]_2$ of each $z_j \in \mathcal{Z}$, and the Groth-Sahai CRS. The simulation trapdoor is ε and the GS simulation trapdoor for equations which are right-simulatable [6].

Prover. If $x \in \mathcal{Z}$, then there is some pair $([y]_2, [\sigma]_2)$, where $[\sigma]_2$ is in the CRS, such that

$$e([\mathsf{sk}]_1 - [x]_1, [\sigma]_2) = [1]_T \quad \text{and} \quad [y]_2 = x[\varepsilon]_2.$$

The prover produces a Groth-Sahai proof of the equations:

$$e\left([\mathsf{sk}]_1 - [X]_1, [\Sigma]_2\right) = [1]_T \quad \text{and} \quad [Y]_2 = X[\varepsilon]_2$$

where X, Y, Σ are the variables.

[5] An alternative is of course to commit to x bit-by-bit to make it extractable, but it is completely impractical.

[6] See Ràfols [28]. These are statements for which only the commitments in $\mathbb{G}_2$ need to be perfectly hiding and where it is sufficient to get the simulation trapdoor to equivocate commitments in $\mathbb{G}_2$.

Verifier. Accept if and only if both proofs are valid.

Theorem 5. *The argument above is computationally quasi-adaptively sound under the $\mathcal{Z}$-GSDH Assumption in $\mathbb{G}_2$ and the soundness of Groth-Sahai proofs.*

Proof. We construct an adversary $\mathcal{B}$ against the $\mathcal{Z}$-GSDH assumption, which receives $gk := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2)$ together with $[\varepsilon]_{1,2}$ and $\{[s^i]_{1,2}\}_{i=1}^m$ from the challenger. The adversary defines a new generator for $\mathbb{G}_2$, $\overline{\mathcal{P}}_2 = [\prod_{i=1}^m (s - z_i)]_2$, defines a new group key $\overline{gk} := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \overline{\mathcal{P}}_2)$, and defines $[\mathsf{sk}]_1 = [s]_1$. Note that we use implicit notation with respect to $\mathcal{P}_1, \mathcal{P}_2$ and not with respect to the new generators.

The adversary can now build the signatures

$$\left(z_j[\varepsilon]_2, \left[\prod_{\substack{i=1 \\ i \neq j}}^{m} (s - z_i) \right]_2 \right) = \left(z_j[\varepsilon]_2, \frac{1}{\mathsf{sk} - z_j} \overline{\mathcal{P}}_2 \right)$$

which are valid with respect to the group key $\overline{gk}$.

Let $\mathcal{A}$ be an adversary against our set membership proof. Adversary $\mathcal{B}$ runs $\mathcal{A}$ with the new group key $\overline{gk}$, Groth-Sahai commitment keys for which it knows the discrete logarithm (in order to open commitments), and signatures $([\sigma_1]_2, \ldots, [\sigma_m]_2)$. Suppose that $\mathcal{A}$ wins by producing an accepting proof for some $x \notin \mathcal{Z}$. From the adversary's proof and committed values one can extract $[x]_1$ and $([y^*]_2, [\sigma^*]_2)$ and, from perfect soundness of Groth-Sahai proofs, it follows that

$$e([\mathsf{sk}]_1 - [x]_1, [\sigma^*]_2) = e(\mathcal{P}_1, \overline{\mathcal{P}}_2) \quad \text{and} \quad [y^*]_2 = x[\varepsilon]_2.$$

This implies that $[\sigma^*]_2 = \left[\frac{\prod_{j=1}^m (\mathsf{sk} - z_j)}{\mathsf{sk} - x} \right]_2$, and hence $([x]_1, [y^*]_2, [\sigma^*]_2)$ is a solution to the $\mathcal{Z}$-GSDH problem. □

Theorem 6. *The argument above is composable zero-knowledge under the composable zero-knowledge property of Groth-Sahai proofs.*

Proof. The proof simulator uses the Groth-Sahai trapdoor and ε to simulate the Groth-Sahai proof of both equations (note that even though the commitment $[\boldsymbol{c}]_1$ is part of the statement, both equations are right-simulatable when ε is known). □

4.2 Aggregated Set Membership Argument

Let $\mathcal{Z} \subset \mathbb{Z}_p$, $m = |\mathcal{Z}|$, and $n \in \mathbb{N}$. We construct a QA-NIZK argument for the following language

$$\mathcal{L}_{\mathsf{memb},\mathcal{Z},ck} := \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^{2n} \;\middle|\; \begin{array}{l} \exists \boldsymbol{w} \in \mathbb{Z}_p^n \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{x}; \boldsymbol{w}) \\ \text{and } x_1, \ldots, x_n \in \mathcal{Z} \end{array} \right\},$$

where $[\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{x}; \boldsymbol{w})$ is a vector of ElGamal encryptions. The generalization to other perfectly binding commitments is straightforward.

Intuition. To express the validity of n signature and message pairs, we construct polynomials $v(X), y(X)$, which encode the set of n verification equations for the Boneh-Boyen signatures. Given the set $\mathcal{R} = \{r_1, \ldots, r_n\} \subset \mathbb{Z}_p$, recall that we denote as $\ell_i(X)$ the ith Lagrange interpolation polynomial associated to $\mathcal{R}$.

We define $v_0(X)$ as the constant polynomial $v_0(X) = \mathsf{sk}$, and $t(X) = \prod_{r_j \in \mathcal{R}} (X - r_j)$. The set of polynomials $v_0(X), \{\ell_i(X)\}_{i=0}^n, t(X)$ accepts $x_1, \ldots, x_n$ if and only if $t(X)$ divides $(v_0(X) - v(X))y(X) - 1$, where

$$v(X) = \sum_{j=1}^{n} x_i \ell_i(X), \qquad y(X) = \sum_{i=1}^{m} \sigma_{k(i)} \ell_i(X),$$

and $\sigma_{k(i)}$ is the signature of some $z_{k(i)}$ such that $x_i = z_{k(i)}$.

That is, at any point $r_j \in \mathcal{R}$, if $x_j = v(r_j)$, then $y(r_j)$ is a a valid signature of x_j. This follows from

$$\begin{aligned}&(v_0(X) - v(X))y(X) - 1 = h(X)t(X) \text{ for some polynomial } h(X)\\ \implies &(v_0(r_j) - v(r_j))y(r_j) - 1 = 0 \iff (\mathsf{sk} - x_j)y(r_j) - 1 = 0.\end{aligned}$$

In particular, if $j \in [n]$ is such that $x_j \notin \mathcal{Z}$, then $y(r_j)$ is a forgery for x_j. For simplicity, in this exposition we ignore the issue mentioned in previous section about commitment extractability, but this is taken into account in the argument.

Note that to compute $y(X)$ given $\ell_i(X)$ in some source group, the prover would need to know the discrete logarithm of the signatures. To render the interpolation polynomials efficiently computable, we include in the CRS the terms $[\sigma_i s^j]_2$, where $\sigma_i = \frac{1}{\mathsf{sk} - z_i}$, for all $i \in \{1, \ldots, m\}, j \in \{1, \ldots, n\}$, and all other values which require the signature's discrete logarithm. Consequently, our CRS is of size $O(nm)$.

A direct instantiation of techniques from Sect. 3.1 requires perfectly binding commitments to each of the signatures and hence, a proof of size linear in the number of statements. But it turns out that perfectly binding commitments to signatures are not necessary for proving membership in $\mathcal{Z}$. To achieve this, we use a trick similar to Sect. 3.1. We program the CRS in order to extract a valid signature for x_{j^*}, for a random $j^* \in \{1, \ldots, n\}$, in such a way that the adversary might only detect the change in the CRS with negligible probability.

Scheme description. Given $m, n \in \mathbb{N}$ and a set $\mathcal{Z} \subset \mathbb{Z}_p$, $|\mathcal{Z}| = m$, we construct a QA-NIZK argument for the language $\mathcal{L}_{\mathsf{memb},\mathcal{Z},ck}$.

Setup.

- Algorithm $\mathsf{K}_0(gk)$ sets $ck = [\boldsymbol{u}]_1 \leftarrow \mathcal{L}_1$.
- Algorithm $\mathsf{K}_1(gk, ck)$ picks $s \leftarrow \mathbb{Z}_p$, $\left\{\boldsymbol{\phi}_i, \hat{\boldsymbol{\phi}}_i\right\}_{i\in\{1,\ldots,n+1\}} \leftarrow \mathbb{Z}_p^3 \times \mathbb{Z}_p^4$, $\mathbf{Q}_1 \leftarrow \mathcal{U}_{3,3}, \mathbf{Q}_2 \leftarrow \mathcal{U}_{4,4}$, picks a Boneh-Boyen secret key $\mathsf{sk} \leftarrow \mathbb{Z}_p$, generates signatures $[\sigma_1]_2, \ldots, [\sigma_m]_2$ for each element in $\mathcal{Z}$ and generates also crs_{Π_1} and crs_{Π_2} for proving membership in the linear spaces generated, respectively, by the matrices $\mathbf{M}, \mathbf{N}$, where:

$$[\mathbf{M}]_1 = \left[\begin{array}{ccc|ccc|cc} \boldsymbol{e}_2 & & & \boldsymbol{u} & & & & \\ & \ddots & & & \ddots & & \multicolumn{2}{c}{\mathbf{0}} \\ & & \boldsymbol{e}_2 & & & \boldsymbol{u} & & \\ \hline \ell_1(s) & \ldots & \ell_n(s) & \multicolumn{3}{c|}{\multirow{2}{*}{$\mathbf{0}$}} & t(s) & \mathbf{0} \\ \boldsymbol{\phi}_1 & \ldots & \boldsymbol{\phi}_n & \multicolumn{3}{c|}{} & \boldsymbol{\phi}_{n+1} & \mathbf{Q}_1 \end{array}\right]_1 \in \mathbb{G}_1^{(2n+4)\times(2n+4)},$$

$$[\mathbf{N}]_2 = \left[\begin{array}{ccccc|cc} \sigma_1\ell_1(s) & \sigma_1\ell_2(s) & \ldots & \sigma_m\ell_n(s) & & t(s) & \mathbf{0} \\ \sigma_1\hat{\boldsymbol{\phi}}_1 & \sigma_1\hat{\boldsymbol{\phi}}_2 & \ldots & \sigma_m\hat{\boldsymbol{\phi}}_n & & \hat{\boldsymbol{\phi}}_{n+1} & \mathbf{Q}_2 \end{array}\right]_2 \in \mathbb{G}_2^{5\times(nm+5)}.$$

The CRS includes the elements

$$\left(gk, ck, \left\{[s^j]_1, [\mathsf{sk}s^j]_1, [\sigma_i s^j]_{1,2}, [\boldsymbol{\phi}_i]_1, [\sigma_i\hat{\boldsymbol{\phi}}_j]_2\right\}_{i\in\{1,\ldots,m\}, j\in\{1,\ldots,n\}}, [\boldsymbol{\phi}_{n+1}]_1, [\hat{\boldsymbol{\phi}}_{n+1}]_2, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathsf{crs}_{\Pi_1}, \mathsf{crs}_{\Pi_2}\right).$$

Prover. The prover $\mathsf{P}(\mathrm{CRS}, [\boldsymbol{c}]_1, \boldsymbol{x}, \boldsymbol{w})$ picks $\delta_v, \delta_y \leftarrow \mathbb{Z}_p, \boldsymbol{r}_{q.1} \leftarrow \mathbb{Z}_p^3, \boldsymbol{r}_{q.2} \leftarrow \mathbb{Z}_p^4$ and defines the polynomials

$$v(X) = \sum_{i=1}^{n} x_i\ell_i(X) + \delta_v t(X), \qquad y(X) = \sum_{i=1}^{n} \sigma_{k(i)}\ell_i(X) + \delta_y t(X)$$

$$h(X) = \frac{(v_0(X) - v(X))y(X) - 1}{t(X)}$$

where $v_0(r_j) = \mathsf{sk}$, for all $j \in \{1, \ldots, n\}$, $t(X) = \prod_{r \in \mathcal{R}}(X - r)$ and $\ell_i(X)$ is the ith Lagrangian interpolation polynomial associated to $\mathcal{R}$. By definition of the language, each x_i is equal to $z_{k(i)}$, for some $k(i) \in \{1, \ldots, m\}$.

The prover computes the following elements:

$$\begin{aligned} [H]_1 &= [h(s)]_1 \\ [V]_1 &= [v(s)]_1 \qquad & [\boldsymbol{q}_1]_1 &= [\textstyle\sum_{i=1}^n x_i \boldsymbol{\phi}_i + \delta_v \boldsymbol{\phi}_{n+1} + \mathbf{Q}_1 \boldsymbol{r}_{q.1}]_1 \\ [Y]_2 &= [y(s)]_2 \qquad & [\boldsymbol{q}_2]_2 &= \left[\textstyle\sum_{i=1}^n \sigma_{k(i)} \hat{\boldsymbol{\phi}}_i + \delta_y \hat{\boldsymbol{\phi}}_{n+1} + \mathbf{Q}_2 \boldsymbol{r}_{q.2}\right]_2 . \end{aligned}$$

The prover also computes two LS proofs

$$\psi_1 \leftarrow \mathit{\Pi}_1.\mathsf{LS.prove}\left(\mathsf{crs}_{\mathit{\Pi}_1}, \begin{bmatrix} \boldsymbol{c} \\ V \\ \boldsymbol{q}_1 \end{bmatrix}_1, \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{w} \\ \delta_v \\ \boldsymbol{r}_{q.1} \end{pmatrix}\right), \psi_2 \leftarrow \mathit{\Pi}_2.\mathsf{LS.prove}\left(\mathsf{crs}_{\mathit{\Pi}_2}, \begin{bmatrix} Y \\ \boldsymbol{q}_2 \end{bmatrix}_2, \begin{pmatrix} \boldsymbol{y} \\ \delta_y \\ \boldsymbol{r}_{q.2} \end{pmatrix}\right),$$

where $\boldsymbol{y} = (y_{1,1}, y_{1,2}, \ldots, y_{n,m})$ and $y_{i,j}$ is equal to 1 if $i = k(j)$ and 0 otherwise. Finally, it sends the proof π to the verifier, where

$$\pi := ([H]_1, [V]_1, [Y]_2, [\boldsymbol{q}_1]_1, [\boldsymbol{q}_2]_2, \psi_1, \psi_2).$$

Verifier. The verifier $\mathsf{V}(\mathrm{CRS}, \pi)$ checks whether the equation

$$e([H]_1, [t(s)]_2) = e([v_0(s)]_1 - [V]_1, [Y]_2) - [1]_T \text{ holds, and}$$

$$\mathit{\Pi}_1.\mathsf{LS.verify}\left(\mathsf{crs}_{\mathit{\Pi}_1}, \begin{bmatrix} \boldsymbol{c} \\ V \\ \boldsymbol{q}_1 \end{bmatrix}_1, \psi_1\right) = 1, \qquad \mathit{\Pi}_2.\mathsf{LS.verify}\left(\mathsf{crs}_{\mathit{\Pi}_2}, \begin{bmatrix} Y \\ \boldsymbol{q}_2 \end{bmatrix}_2, \psi_2\right) = 1.$$

If all of these conditions hold, it returns 1, else 0.

Completeness. If $x_1, \ldots, x_n \in \mathcal{Z}$ then $(v_0(r_j) - v(r_j))y(r_j) - 1 = (x_{k(j)} + \mathsf{sk})\sigma_{k(j)} - 1 = 0$ for all j, and thus $(v_0(X) - v(X))y(X) = 1 \mod t(X)$. This implies that $h(X)$ is a well defined polynomial in $\mathbb{Z}_p[X]$ such that $e([h(s)]_1, [t(s)]_2) = e([v_0(s) - v(s)]_1, [y(s)]_2) - [1]_T$. It is easy to check that

$$\begin{pmatrix} \boldsymbol{c} \\ V \\ \boldsymbol{q}_1 \end{pmatrix} = \mathbf{M} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{w} \\ \delta_v \\ \boldsymbol{r}_{q.1} \end{pmatrix} \text{ and } \begin{pmatrix} Y \\ \boldsymbol{q}_2 \end{pmatrix} = \mathbf{N} \begin{pmatrix} \boldsymbol{y} \\ \delta_y \\ \boldsymbol{r}_{q.2} \end{pmatrix},$$

where $\boldsymbol{y} = (y_{1,1}, \ldots, y_{m,n})$, and therefore ψ_1, ψ_2 are valid proofs.

Soundness.

Theorem 7. *Let $\mathsf{Adv}_{\mathrm{PS}}(\mathcal{A})$ be the advantage of a PPT adversary $\mathcal{A}$ against the soundness of the scheme. There exist PPT adversaries $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_{3,1}, \mathcal{B}_{3,2}, \mathcal{B}_4, \mathcal{B}_5$ such that*

$$\begin{aligned} \mathsf{Adv}_{\mathrm{PS}}(\mathcal{A}) \leq n \,(& 2\mathsf{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_1}(\mathcal{B}_1) + 3\mathsf{Adv}_{\mathcal{L}_1\text{-}\mathsf{MDDH},\mathbb{G}_2}(\mathcal{B}_2) + \mathsf{Adv}_{\mathsf{LS},\mathit{\Pi}_1}(\mathcal{B}_{3,1}) \\ & + \mathsf{Adv}_{\mathsf{LS},\mathit{\Pi}_2}(\mathcal{B}_{3,2}) + \mathsf{Adv}_{\mathcal{Z}\text{-}\mathsf{GSDH},\mathbb{G}_1}(\mathcal{B}_4) + \mathsf{Adv}_{n\text{-}\mathsf{QTSDH}}(\mathcal{B}_5)) . \end{aligned}$$

Proof. In order to prove soundness we will prove indistinguishability of the following games.

– Real: This is the real soundness game. The output is 1 if the adversary produces a false accepting proof, i.e. if there is some $x_i \notin \mathcal{Z}$ and the verifier accepts the proof.

– Game_0: This game is identical to the previous one, except that the commitment key $\boldsymbol{u}$ is chosen by the game in order to extract $[\boldsymbol{x}]_1$ from $[\boldsymbol{c}]_1$.
– Game_1: This game is identical to the previous one, except that some $j^* \leftarrow \{1,\ldots,n\}$ is chosen and the game aborts if the extracted value $[\boldsymbol{x}]_1$ is such that $[x_{j^*}]_1 \in [\mathcal{Z}]_1$.
– Game_2: For $i = 1,\ldots,n$, let $\alpha_i(X)$ and β_i be the quotient and the reminder, respectively, of dividing $\ell_i(X)$ by $X - r_{j^*}$. Let $\alpha_{n+1}(X)$ and β_{n+1} be the quotient and the reminder of dividing $t(X)$ by $X - r_{j^*}$. This game is identical to the previous one, except that $\mathbf{Q}_1$ is now a uniformly random matrix conditioned on having rank 1, and for $i = 1,\ldots,n+1$, $[\boldsymbol{\phi}_i]_1$ is changed to

$$[\boldsymbol{\phi}_i]_1 = [\alpha_i(s)]_1 \boldsymbol{e}_2^3 + \beta_i [\varepsilon]_1 \boldsymbol{e}_3^3 + [\mathbf{Q}_1]_1 \boldsymbol{r}_i,$$

where $\boldsymbol{e}_j^3$ is the jth vector of the canonical basis of $\mathbb{Z}_p^3$, $\boldsymbol{r}_i \leftarrow \mathbb{Z}_p^3$, $\varepsilon \leftarrow \mathbb{Z}_p$.
– Game_3: Let $\alpha_i(X)$ and β_i be defined as above. This game is identical to the previous one, except that $\mathbf{Q}_2$ is now a uniformly random matrix conditioned on having rank 1, and each $\left[\hat{\boldsymbol{\phi}}_i\right]_2$ is now defined as

$$\left[\hat{\boldsymbol{\phi}}_i\right]_2 = [\alpha_i(s)]_2 \boldsymbol{e}_2^4 + [\beta_i]_2 \boldsymbol{e}_3^4 + \beta_i [\varepsilon]_2 \boldsymbol{e}_4^4 + [\mathbf{Q}_2]_2 \widetilde{\boldsymbol{r}}_i,$$

where $\boldsymbol{e}_j^4$ is the jth vector of the canonical basis of $\mathbb{Z}_p^4$, $\widetilde{\boldsymbol{r}}_i \leftarrow \mathbb{Z}_p^4$ and $\varepsilon \leftarrow \mathbb{Z}_p$ is the same value used in the definition of $[\boldsymbol{\phi}_i]_1$.

Obviously, the games Real and Game_0 are indistinguishable. The proofs of indistinguishablility of $\mathsf{Game}_1, \mathsf{Game}_2$ and $\mathsf{Game}_2, \mathsf{Game}_3$ are the same as their analogues in Sect. 3.1. We proceed to prove that in Game_3 the adversary wins only with negligible probability.

Lemma 7. *There exists adversaries $\mathcal{B}_{3,i}$ against the soundness of $\Pi_i.\mathsf{LS}$, an adversary $\mathcal{B}_4$ against $\mathcal{Z}$-GSDH in $\mathbb{G}_1$, and an adversary $\mathcal{B}_5$ against n-QTSDH such that*

$$\Pr[\mathsf{Game}_3(\mathcal{A}) = 1] \leq \mathsf{Adv}_{\mathsf{LS}}(\mathcal{B}_{3,1}) + \mathsf{Adv}_{\mathsf{LS}}(\mathcal{B}_{3,2}) + \mathsf{Adv}_{n\text{-}\mathsf{QTSDH}}(\mathcal{B}_4) + \mathsf{Adv}_{\mathcal{Z}\text{-}\mathsf{GSDH},\mathbb{G}_1}(\mathcal{B}_5).$$

Proof. Let E_1 be the event where $(\boldsymbol{c}, V, \boldsymbol{q}_1)$ is not in the image of $\mathbf{M}$, E_2 the event that $(Y, \boldsymbol{q}_2)$ is not in the image of $\mathbf{N}$, and $E_3 = \overline{E_1 \cup E_2}$. Then

$$\begin{aligned}\Pr[\mathsf{Game}_3(\mathcal{A}) = 1] \leq & \Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_1] + \Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_2] + \\ & + \Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_3],\end{aligned} \tag{15}$$

and, clearly,

$$\Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_1] + \Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_2] \leq \mathsf{Adv}_{\Pi_1.\mathsf{LS}}(\mathcal{B}_{3,1}) + \mathsf{Adv}_{\Pi_2.\mathsf{LS}}(\mathcal{B}_{3,2}).$$

We now proceed to bound $\Pr[\mathsf{Game}_3(\mathcal{A}) = 1 | E_3]$. Conditioned on E_3, there exist some $\boldsymbol{x}^\dagger, \boldsymbol{w}, \delta_v, \boldsymbol{r}_{q.1}$ and $\boldsymbol{y}^\dagger, \delta_y, \boldsymbol{r}_{q.2}$ such that $(\boldsymbol{c}, V, \boldsymbol{q}_1)^\top = \mathbf{M}(\boldsymbol{x}^\dagger, \boldsymbol{w}, \delta_v, \boldsymbol{r}_{q.1})^\top$ and $(Y, \boldsymbol{q}_2)^\top = \mathbf{N}(\boldsymbol{y}^\dagger, \delta_y, \boldsymbol{r}_{q.2})^\top$. Given that $\boldsymbol{c}$ is perfectly binding, it must be that $\boldsymbol{x} = \boldsymbol{x}^\dagger$. It follows that $V = \sum_{i=1}^n x_i \ell_i(s) + \delta_v t(s) = v(s)$ and $Y = y^\dagger(s)$ for some polynomial $y^\dagger(X) = \sum_{i=1}^n \sum_{j=1}^m y_{i,j}^\dagger \sigma_i \ell_i(X) + \delta_y t(X)$. Further, except with probability $1/q$, each $\boldsymbol{e}_j^i$ is linearly independent of the columns of $[\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2$, so one can extract from $[\boldsymbol{q}_1]_1$ (resp. $[\boldsymbol{q}_2]_2$) the coefficients of these vectors in its expression in terms of $[\mathbf{Q}_1]_1, \boldsymbol{e}_2^3, \boldsymbol{e}_3^3$ (resp. $[\mathbf{Q}_2]_2, \boldsymbol{e}_2^4, \boldsymbol{e}_3^4, \boldsymbol{e}_4^4$), which are:

$$\begin{bmatrix} \sum_{i=1}^{n+1} x_i \alpha_i(s) \\ \sum_{i=1}^{n+1} x_i \beta_i \varepsilon \end{bmatrix}_1 = \begin{bmatrix} \alpha(s) \\ \beta\varepsilon \end{bmatrix}_1 \text{ and } \begin{bmatrix} \sum_{i,j=1}^{m,n} y_{i,j}^\dagger \sigma_i \tilde{\alpha}_j(s) + \delta_y \tilde{\alpha}_{n+1}(s) \\ \sum_{i,j=1}^{m,n} y_{i,j}^\dagger \sigma_i \beta_j + \delta_y \tilde{\beta}_{n+1} \\ \sum_{i,j=1}^{m,n} y_{i,j}^\dagger \sigma_i \beta_j \varepsilon + \delta_y \tilde{\beta}_{n+1} \varepsilon \end{bmatrix}_2 = \begin{bmatrix} \tilde{\alpha}(s) \\ \tilde{\beta} \\ \tilde{\beta}\varepsilon \end{bmatrix}_2$$

where $x_{n+1} = \delta_v$ and $\alpha(X), \tilde{\alpha}(X)$ are the quotients and $\beta, \tilde{\beta}$ are the reminders of dividing, respectively, $v(X)$ and $y(X)$ by $X - r_{j^*}$.

If we divide both sides of the verification equation by $(s - r_{j^*})$, and we denote by $\alpha_0(s), \beta_0$ we get that

$$\begin{aligned} e\left([H]_1, \left[\frac{t(s)}{s-r_{j^*}}\right]_2\right) =& \frac{1}{s-r_{j^*}}(e([v_0(s)]_1 - [v(s)]_1, [y(s)]_2) - [1]_T) \\ =& \frac{1}{s-r_{j^*}}\left[(v_0(s)-v(s))(\tilde{\alpha}(s)(s-r_{j^*})+\tilde{\beta}) - 1\right]_T \\ =& [(v_0(s)-v(s))\tilde{\alpha}(s) + \alpha(s)\tilde{\beta}]_T + \left[\frac{(v_0(s)-\beta)\tilde{\beta}-1}{s-r_{j^*}}\right]_T \end{aligned}$$

Note that $\beta = v(r_{j^*}) = x_{j^*}, v_0(s) = \mathsf{sk}$ and thus if $(v_0(s)-\beta)\tilde{\beta} - 1 = 0$, then $\tilde{\beta}$ is a valid signature for x_{j^*}.

Let E_4 the event $(v_0(s)-\beta)\tilde{\beta} - 1 = 0$ and thus $\Pr[\mathsf{Game}_4(\mathcal{A}) = 1|E_3] \leq \Pr[\mathsf{Game}_4(\mathcal{A}) = 1|E_4 \cap E_3] + \Pr[\mathsf{Game}_4(\mathcal{A}) = 1|\overline{E}_4 \cap E_3]$.

We build an adversary $\mathcal{B}_4$ against Assumption 6 which receives $gk, \{[\mathsf{sk}^i]_1, [\mathsf{sk}^i]_2\}_{i\in[m]}, [\varepsilon]_{1,2}$. Essentially, the adversary works as the one described in Sect. 4.1 for the (non-aggregated) set membership argument. It simulates $\mathsf{Game}_4(\mathcal{A})$ computing all the discrete logarithms of the CRS itself, except for the Boneh-Boyen secret key, $[\varepsilon]_{1,2}$, and the signatures in the CRS are computed as in Sect. 4.1. When $\mathcal{A}$ outputs $[\boldsymbol{q}_1]_1, [\boldsymbol{q}_2]_2$, $\mathcal{B}_4$ extracts $[\beta\varepsilon]_1, [\tilde{\beta}]_2$ and returns $([x_{j^*}]_1, [\beta\varepsilon]_1, [\tilde{\beta}]_2)$. In the case E_4, we have already argued that $\tilde{\beta}$ is a valid signature for x_{j^*}, and in this game $x_{j^*} \notin S$. We conclude that $\Pr[\mathsf{Game}_4(\mathcal{A}) = 1|E_4 \cap E_3] \leq \mathsf{Adv}_{\mathcal{Z}\text{-}\mathbf{GSDH},\mathbb{G}_1}(\mathcal{B}_4)$.

We also construct $\mathcal{B}_5$ an adversary against Assumption 8. It receives as input $[\varepsilon]_1, [\varepsilon]_2, [s]_1, [s]_2, \ldots, [s^d]_1 [s^d]_2$ and it starts a simulation of $\mathsf{Game}_4(\mathcal{A})$, by sampling honestly the rest of the elements of the CRS. Finally, $\mathcal{A}$ outputs $[V]_1, [Y]_2, [\boldsymbol{q}_1]_1, [\boldsymbol{q}_2]_2$ as part of the purported proof for $[\boldsymbol{c}]_1$. We will see in the following how $\mathcal{B}_4$ computes $[\nu]_T := \left[\frac{(v_0(s)-\beta)\tilde{\beta}-1}{s-r_{j^*}}\right]_T$ and returns $([v_0(s)-\beta]_1, [(v_0(s)-\beta)\varepsilon]_1, [\tilde{\beta}]_2, [\tilde{\beta}\varepsilon]_2, [\nu]_T)$, with $(v_0(s)-\beta)\tilde{\beta} - 1 \neq 0$, breaking Assumption 8.

The values $[\tilde{\alpha}(s)]_2, [\tilde{\beta}]_2$ and $[\tilde{\beta}\varepsilon]_2$ are extracted from $[\boldsymbol{q}_2]_2$, while $[\alpha(s)]_1, [\beta\varepsilon]_1$ are extracted from $[\boldsymbol{q}_1]_1$, $[\beta]_1 = [x_{j^*}]_1$ is extracted from $[\boldsymbol{c}]_1$, $\beta_0 = \mathsf{sk}$, and $[v_0(s)\varepsilon]_1 = \mathsf{sk}[\varepsilon]_1$ can be computed by $\mathcal{B}_5$ because it sampled sk. The value $[\nu]_T$ is computed as

$$[\nu]_T := e\left([H]_1, \left[\frac{t(s)}{s-r_{j^*}}\right]_2\right) - e([v_0(s)]_1 - [V]_1, [\tilde{\alpha}(s)]_2) - e\left([\alpha(s)]_1, [\tilde{\beta}]_2\right).$$

Zero-Knowledge. The proof of perfect zero-knowledge is essentially the same as for Theorem 2. Note that $[V]_1, [Y]_2, [\boldsymbol{q}_1]_1, [\boldsymbol{q}_2]_2$ are independent of $\boldsymbol{x}$, while $[H]_1$ is the unique solution to the verification equation. Perfect zero-knowledge of the argument of membership in linear spaces implies that the proofs ψ_1, ψ_2 can be simulated with the same distribution as honest proofs.

5 Shuffle Arguments

From our results, we can construct two different shuffle arguments in the CRS model under falsifiable assumptions. They both follow the basic template of the shuffle argument of [13]. Let $[\boldsymbol{c}_1]_2, [\boldsymbol{c}_2]_2$ be two vectors of n ciphertexts which open to vectors of plaintexts $[\boldsymbol{m}_1]_2, [\boldsymbol{m}_2]_2$, respectively, and we want to prove that $\boldsymbol{m}_2$ is a permutation of $\boldsymbol{m}_1$. The shuffle argument of [13] consists of the following steps. The CRS includes a vector of group elements $[\boldsymbol{z}]_1 = ([z_1]_1, \ldots, [z_n]_1)$ sampled uniformly and independently. The prover chooses a permutation $[\boldsymbol{x}]_1 = ([x_1]_1, \ldots, [x_n]_1)$ of $[\boldsymbol{z}]_1$ and proves: (1) $x_i \in \mathcal{Z} = \{z_1, \ldots, z_n\}$ for all $i \in \{1, \ldots, n\}$, (2) $\sum x_i = \sum z_i$ and (3) $\sum z_i m_{1,i} = \sum x_i m_{2,i}$.

The first two steps force $\boldsymbol{x}$ to be a permutation of $\boldsymbol{z}$: if all $x_i \in \mathcal{Z}$ and their sum equals the sum of all the elements in $\mathcal{Z}$ and $\boldsymbol{x}$ is not a permutation, the prover has found a non-trivial combination of elements of $\mathcal{Z}$ which is 0, which is a type of kernel problem. The last step links this fact with $\boldsymbol{m}_2$ being a permutation of $\boldsymbol{m}_1$.

In both our constructions and in the original argument of [13], Steps (2) and (3) are handled with the following Groth-Sahai equations, in which uppercase letters are variables for which the prover has provided commitments: (2) $\sum[X_i]_1 = \sum[z_i]_1$ and (3) $\sum e([z_i]_1, [M_{1,i}]_2) = \sum e([X_i]_1, [M_{2,i}]_2)$.

We next specify two different ways of proving Step 1, which results in two different constructions with different performance.

5.1 Unit Vector Argument

The first approach is the closest to the work of González et al. [13]. There, Step 1 is rewritten as proving that $\boldsymbol{x} = \boldsymbol{z}^\top \boldsymbol{B}$, for a matrix $\boldsymbol{B} = (\boldsymbol{b}_1 | \ldots | \boldsymbol{b}_n) \in \{0,1\}^{n^2}$, where the $\boldsymbol{b}_i$ are unitary vectors (not necessarily different, as this is handled by step 2). The approach of [13] is to adopt a commit-and-prove strategy using arguments for linear spaces and the bitstring argument of [12]. The 'prove' part is constant-size, but the 'commit' part is a priori quadratic, as we would need to commit to each entry of the matrix $\boldsymbol{B}$.

To overcome this and obtain linear complexity, they switch to shrinking commitments to each row $\boldsymbol{b}_i^*$ of $\boldsymbol{B}$, which take only two elements each. Obviously these commitments cannot be perfectly binding, and this fact interferes with the extraction step in soundness proof. However, a key step in their argument is that they set these commitments in a way that one single coordinate j^* (which remains unknown to the adversary) is perfectly binding. Thus the corresponding column is uniquely determined and can be extracted in the proof. From here, it is concluded that an adversary cannot cheat in the j^*-th ciphertext, and since j^* is unknown to the adversary, general soundness is reduced to this case with a tightness loss of $1/n$. Note that this is on top of the factor $1/n$ from the bitstring argument, resulting in a soundness loss of $1/n^2$.

We observe that we can plug our unit vector argument instead of the one from [12], modified to accept shrinking commitments to each of the rows of $\mathbf{B}$ as those in [13]. We include an additional game at the beginning of the soundness proof of the unit vector argument, in which we choose a random coordinate and abort if the corresponding commitment is not in the language. From here on the proof works as in Sect. 3.2. This proof inherits the disadvantages of [13], namely the quadratic CRS and the tightness loss in the security reduction, but we improve the proof size from $(4n+17)|\mathbb{G}_1| + 14|\mathbb{G}_2|$ to $(4n+11)|\mathbb{G}_1| + 8|\mathbb{G}_2|$ and our proof still uses falsifiable and static assumptions.

5.2 Argument of Membership in a Set of Group Elements

Another approach to Step 1, instead of the aggregated unit vector proofs, is to prove directly membership in a subset $\mathcal{Z} = \{[z_1]_1, \ldots, [z_n]_1\} \subset \mathbb{G}_1$. Note that the set is witness sampleable and in particular, the discrete logarithms might be known when generating the CRS. More precisely, we want to construct an argument for the language

$$\mathcal{L}_{\mathsf{memb\text{-}group},\mathcal{Z},ck} := \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^2 \,\middle|\, \exists \boldsymbol{w} \in \mathbb{Z}_p \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}([x]_1; w) \text{ and } [x]_1 \in \mathcal{Z} \right\},$$

and for efficiency, the proof should be aggregated. This can be achieved by modifying the aggregated membership proof in a subset of $\mathbb{Z}_p$ from Sect. 4.2. Note that there we had $x \in \mathbb{Z}_p$, and this was necessary to produce the proof, so to ensure completeness when the prover knows only $[x]_1 \in \mathcal{Z} \subset \mathbb{G}_1$, we provide additional elements in the CRS. This is possible because the set is witness sampleable. More precisely, x was involved in the definition of the terms

$$[V]_1 = [v(s)]_1, \qquad \text{where } v(X) = \sum_{i=1}^{n} x_i \ell_i(X) + \delta_v t(X),$$

$$[\boldsymbol{q}_1]_1 = \left[\sum_{i=1}^{n} x_i \boldsymbol{\phi}_i + \delta_v \boldsymbol{\phi}_{n+1} + \mathbf{Q}_1 \boldsymbol{r}_{q.1} \right]_1,$$

so we include the elements $\{[z_i \ell_j(s)]_1, [z_i \phi_j]_1\}_{i,j \in \{1,\ldots,n\}}$ in the CRS. The proof works exactly the same, as the reduction could only open the commitments in the group.

We can use this to prove Step 1 of the shuffle argument above. In this case, the CRS size is still quadratic in the number of ciphertexts, but we avoid losing the second factor $1/n$ in the reduction, and the proof consists only of the commitments to $[x_i]_1$ and a constant number of elements. More precisely, the proof size is $(2n + 11)|\mathbb{G}_1| + 8|\mathbb{G}_2|$.

References

1. D. Boneh and X. Boyen. Secure identity based encryption without random oracles. In M. Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 443–459. Springer, Heidelberg, Aug. 2004. 6
2. D. Boneh and X. Boyen. Short signatures without random oracles and the SDH assumption in bilinear groups. *Journal of Cryptology*, 21(2):149–177, Apr. 2008. 6, 8, 22
3. J. Bootle and J. Groth. Efficient batch zero-knowledge arguments for low degree polynomials. In *IACR International Workshop on Public Key Cryptography*, pages 561–588. Springer, 2018. 4
4. J. Camenisch, R. Chaabouni, and a. shelat. Efficient protocols for set membership and range proofs. In J. Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 234–252. Springer, Heidelberg, Dec. 2008. 3, 4, 32
5. R. Canetti, Y. Lindell, R. Ostrovsky, and A. Sahai. Universally composable two-party and multi-party secure computation. In *34th ACM STOC*, pages 494–503. ACM Press, May 2002. 2
6. G. Danezis, C. Fournet, J. Groth, and M. Kohlweiss. Square span programs with applications to succinct NIZK arguments. In P. Sarkar and T. Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 532–550. Springer, Heidelberg, Dec. 2014. 2, 4, 9
7. A. Escala and J. Groth. Fine-tuning Groth-Sahai proofs. In H. Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 630–649. Springer, Heidelberg, Mar. 2014. 1, 2, 3, 8
8. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar. An algebraic framework for Diffie-Hellman assumptions. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Heidelberg, Aug. 2013. 5
9. P. Fauzi, H. Lipmaa, J. Siim, and M. Zajac. An efficient pairing-based shuffle argument. In T. Takagi and T. Peyrin, editors, *ASIACRYPT 2017, Part II*, volume 10625 of *LNCS*, pages 97–127. Springer, Heidelberg, Dec. 2017. 3
10. R. Gennaro, C. Gentry, B. Parno, and M. Raykova. Quadratic span programs and succinct NIZKs without PCPs. In T. Johansson and P. Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 626–645. Springer, Heidelberg, May 2013. 2, 4, 23
11. C. Gentry and D. Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In L. Fortnow and S. P. Vadhan, editors, *43rd ACM STOC*, pages 99–108. ACM Press, June 2011. 2
12. A. González, A. Hevia, and C. Ràfols. QA-NIZK arguments in asymmetric groups: New tools and new constructions. In T. Iwata and J. H. Cheon, editors, *ASIACRYPT 2015, Part I*, volume 9452 of *LNCS*, pages 605–629. Springer, Heidelberg, Nov. / Dec. 2015. 1, 2, 3, 4, 6, 8, 9, 15, 22, 29, 32
13. A. González and C. Ràfols. New techniques for non-interactive shuffle and range arguments. In M. Manulis, A.-R. Sadeghi, and S. Schneider, editors, *ACNS 16*, volume 9696 of *LNCS*, pages 427–444. Springer, Heidelberg, June 2016. 1, 3, 4, 28, 29
14. J. Groth. Short pairing-based non-interactive zero-knowledge arguments. In M. Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 321–340. Springer, Heidelberg, Dec. 2010. 2, 4
15. J. Groth. On the size of pairing-based non-interactive arguments. In M. Fischlin and J.-S. Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Heidelberg, May 2016. 2, 4
16. J. Groth and S. Lu. A non-interactive shuffle with pairing based verifiability. In K. Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 51–67. Springer, Heidelberg, Dec. 2007. 3
17. J. Groth, R. Ostrovsky, and A. Sahai. New techniques for noninteractive zero-knowledge. *J. ACM*, 59(3):11, 2012. 1, 2
18. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In N. P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 415–432. Springer, Heidelberg, Apr. 2008. 4, 8
19. J. Groth and A. Sahai. Efficient noninteractive proof systems for bilinear groups. *SIAM J. Comput.*, 41(5):1193–1232, 2012. 1
20. C. S. Jutla and A. Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In K. Sako and P. Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 1–20. Springer, Heidelberg, Dec. 2013. 1, 2, 4, 7, 9

21. C. S. Jutla and A. Roy. Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 295–312. Springer, Heidelberg, Aug. 2014. 1, 2, 3, 4, 9
22. E. Kiltz and H. Wee. Quasi-adaptive NIZK for linear subspaces revisited. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 101–128. Springer, Heidelberg, Apr. 2015. 1, 4, 9
23. B. Libert, T. Peters, M. Joye, and M. Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 514–532. Springer, Heidelberg, May 2014. 4
24. H. Lipmaa. Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In R. Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 169–189. Springer, Heidelberg, Mar. 2012. 4
25. H. Lipmaa. Succinct non-interactive zero knowledge arguments from span programs and linear error-correcting codes. In K. Sako and P. Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 41–60. Springer, Heidelberg, Dec. 2013. 2
26. P. Morillo, C. Ràfols, and J. L. Villar. The kernel matrix Diffie-Hellman assumption. In J. H. Cheon and T. Takagi, editors, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 729–758. Springer, Heidelberg, Dec. 2016. 6
27. M. Naor. On cryptographic assumptions and challenges (invited talk). In D. Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 96–109. Springer, Heidelberg, Aug. 2003. 2
28. C. Ràfols. Stretching groth-sahai: NIZK proofs of partial satisfiability. In Y. Dodis and J. B. Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 247–276. Springer, Heidelberg, Mar. 2015. 1, 23
29. A. Rial, M. Kohlweiss, and B. Preneel. Universally composable adaptive priced oblivious transfer. In H. Shacham and B. Waters, editors, *PAIRING 2009*, volume 5671 of *LNCS*, pages 231–247. Springer, Heidelberg, Aug. 2009. 4, 32
30. J. L. Villar. Optimal reductions of some decisional problems to the rank problem. In X. Wang and K. Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 80–97. Springer, Heidelberg, Dec. 2012. 12

A Range Argument in the Interval $[0, 2^n - 1]$

We want to prove that a Groth-Sahai commitment $[\boldsymbol{c}]_1$ opens to some integer y in the range $[0, 2^n - 1]$. That is, we want to construct a NIZK proof system for the language

$$\mathcal{L}_{\mathsf{range},ck} := \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^2 : \begin{array}{c} \exists y, r \in \mathbb{Z}_p \text{ s.t. } [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(y; r) \\ \text{and } y \in [0, 2^n - 1] \end{array} \right\},$$

where $ck := ([\boldsymbol{u}_1]_1, [\boldsymbol{u}_2]_1) \leftarrow \mathsf{K}_0(1^\lambda)$. We follow a widely used approach (for example [29, 4] to name a few), which divides the statement $y \in [0, 2^n - 1]$ into ℓ range proofs in smaller intervals. That is,

1. commit to $y_1, \ldots, y_\ell$,
2. show that $y_i \in [0, d-1]$, for each $i \in [\ell]$,
3. show that $y = \sum_{i \in [\ell]} y_i d^{i-1}$.

We commit to $y_1, \ldots, y_\ell$ using only $\ell + 1$ group elements using a simple adaptation of ElGamal to vectors of size n. To prove point 3 we could use membership in linear spaces, as done in [12, Sect. 5.5], requiring only one element from $\mathbb{G}_1$. For point 2 we use our aggregated set-membership proof which requires 6 elements of $\mathbb{G}_1$ and 6 of $\mathbb{G}_2$. The total size of the proof is thus $\left(\frac{n}{\log d} + 7\right) \mathbb{G}_1 + 6\mathbb{G}_2$. Choosing $d = n^k$ we get that and $\ell = \frac{n}{\log n^k} = \frac{n}{k \log n}$, and thus the size of our Range Proof is $\left(\frac{n}{k \log n} + 7\right) \mathbb{G}_1 + 6\mathbb{G}_2$, for an arbitrarily chosen $k \in \mathbb{N}$. The size of the CRS is dominated by $5\ell \cdot d = 5\frac{n^{k+1}}{k \log n}$ (the size of matrix $\mathbf{N}$ in our set membership proof).

In practice, the size of the proof is bounded by the security parameter, i.e. $n < 128$ (one can't commit to a number bigger than the field size). Although for such a big n the size of the CRS is huge, ≈ 12000 and ≈ 730000 group elements for $k = 1, 2$ respectively, the size of the proof is just 26 and 18 group elements for $k = 1, 2$ respectively. For $n = 64$ and $k = 2$, the size of a proof is 13 group elements, it requires roughly 70000 group elements in the CRS. For $n = 64$ and $k = 1$, the size of a proof is bounded by ≈ 18 group elements and the CRS contains roughly 2000 group elements. For more conservative ranges, say $n \approx 10$, one gets proofs of size 10 group elements while the CRS contains roughly 500 group elements, for $k = 2$, or of size 12 with a CRS of size 50 for $k = 1$.

	Language	Proof size	CRS size	Assumption
Sect. A	Range Proof	$\left(\frac{n}{k \log n} + 7\right) \lvert\mathbb{G}_1\rvert + 6\lvert\mathbb{G}_2\rvert$	$\left(\frac{n^{k+1}}{k \log n} + O(1)\right) \lvert\mathbb{G}_1\rvert + \left(5\frac{n^{k+1}}{k \log n} + O(1)\right) \lvert\mathbb{G}_2\rvert$	$\mathcal{Z}$-GSDH 6, q-QTSDH 8
Sect. 4 of [29]		$\approx 15\frac{n}{\log n}(\lvert\mathbb{G}_1\rvert + \lvert\mathbb{G}_2\rvert)$	$O\left(\frac{n}{\log n}\right)$	q-HSDH

Table 5. The table shows the proof sizes (not including commitments for bitstring and unit vector) and CRS sizes of our results in range proofs. The range considered is $[0, 2^n - 1]$ and $k > 0$ is a free parameter (e.g. $k = 1/4, 1/2, 1, 2, \ldots$), and the constant of [29] is at least 4, for committing to signatures, plus $3 \cdot 4$ elements for Groth-Sahai proofs of the signature verification.

B Generic Hardness of Assumptions

Proposition 1. *The $\mathcal{Z}$-GSDH Assumption (6) in $\mathbb{G}_\gamma$ holds in the generic group model.*

Proof. A generic adversary receives $\mathcal{Z}$ in $\mathbb{Z}_p$, ε and the powers $1, s, \ldots, s^q$ in $\mathbb{G}_1$, and ε and $1, s, \ldots, s^q$ in $\mathbb{G}_2$. Then any z_1 output by the adversary must be of the form

$$z_1 = \sum_{i=0}^{q} b_i s^i,$$

for some coefficients $b_i, i \in \{0, \ldots, q\}$, and since $z_2 = \varepsilon z_1$, we have that necessarily

$$z_2 = \sum_{i=0}^{q} b_i \varepsilon s^i.$$

This forces $b_i = 0$ for $i \in \{1, \ldots, q\}$, since a generic adversary cannot compute εs^i in $\mathbb{G}_\gamma$. Thus $z_1 = b_0$.

Then the adversary cannot compute

$$\frac{\prod_{r \in \mathcal{Z}}(s - r)}{s - r_1}$$

in $\mathbb{G}_T$, since r_1 is not a root of $p(s) = \prod_{r \in \mathcal{Z}}(s - r)$, so the above is a rational function, which cannot be computed with group operations.

Proposition 2. *The q-*STSDH *Assumption(7) holds in the generic group model.*

Proof. A generic adversary receives the powers $1, s, \ldots, s^q$ in $\mathbb{G}_1$, and ε and $1, s, \ldots, s^q$ in $\mathbb{G}_2$. Then any β_1 output by the adversary must be of the form

$$\beta_1 = \sum_{i=0}^{q} b_i s^i,$$

for some coefficients $b_i, i \in \{0, \ldots, q\}$, and since $\beta_2 = \varepsilon \beta_1$, we have that necessarily

$$\beta_2 = \sum_{i=0}^{q} b_i \varepsilon s^i.$$

This forces $b_i = 0$ for $i \in \{1, \ldots, q\}$, since a generic adversary cannot compute εs^i in $\mathbb{G}_1$. Thus $\beta_1 = b_0$. Now, if a generic adversary is able to compute $\frac{\beta_1^2 - 1}{s - r}$ in $\mathbb{G}_T$, necessarily there exist polynomials p_1, p_2 such that

$$\frac{b_0^2 - 1}{s - r} = p_1(s, \varepsilon) \cdot p_2(s),$$

where $\deg p_1, \deg p_2 \leq q$ and p_1 does not have terms in εs^i for any i. However, since β_1 is a constant with respect to s, ε, and $\beta_1^2 - 1 \neq 0$, the above is a rational function, which cannot be computed with group operations.

Proposition 3. *The q-*QTSDH *Assumption (8) holds in the generic group model.*

Proof. A generic adversary receives ε and the powers $1, s, \ldots, s^q$ in $\mathbb{G}_1$, and $1, s, \ldots, s^q$ in $\mathbb{G}_2$. Then any β_1 output by the verifier must be of the form

$$\beta_1 = \sum_{i=0}^{q} b_i s^i + b_{q+1} \varepsilon,$$

for some coefficients $b_i, i \in \{0, \ldots, q+1\}$, and since $\beta_2 = \varepsilon \beta_1$, we have that necessarily

$$\beta_2 = \sum_{i=0}^{q} b_i \varepsilon s^i + b_{q+1} \varepsilon^2.$$

This forces $b_i = 0$ for $i \in \{1, \ldots, q\}$, since a generic adversary cannot compute εs^i in $\mathbb{G}_1$, and $b_{q+1} = 0$, since it cannot compute ε^2 either. Thus $\beta_1 = b_0$. Analogously, $\tilde{\beta}_1 = \tilde{b}_0$ for some constant $\tilde{b}_0$. Now, if a generic adversary is able to compute $\frac{\beta_1 \tilde{\beta}_1 - 1}{s - r}$ in $\mathbb{G}_T$, necessarily there exist polynomials p_1, p_2 such that

$$\frac{b_0 \tilde{b}_0 - 1}{s - r} = p_1(s, \varepsilon) \cdot p_2(s, \varepsilon),$$

where $\deg p_1, \deg p_2 \leq q$ and p_1 does not have terms in εs^i for any i. However, since $\beta_1 \tilde{\beta}_1$ is a constant with respect to s, ε, and $\beta_1 \tilde{\beta}_1 - 1 \neq 0$, the above is a rational function, which cannot be computed with group operations.