Somewhere Statistically Binding Commitment Schemes with Applications

No Author Given

Abstract. We define a new primitive that we call a *somewhere statistically binding* (SSB) commitment scheme, which is a generalization of dual-mode commitments but has similarities with SSB hash functions (Hubacek and Wichs, ITCS 2015) without local opening. In (existing) SSB hash functions, one can compute a hash of a vector v that is statistically binding in one coordinate of v. Meanwhile, in SSB commitment schemes, a commitment of a vector v is statistically binding in some coordinates of v and is statistically hiding in the other coordinates. The set of indices where binding holds is predetermined but known only to the commitment key generator. We show that the primitive can be instantiated by generalizing the succinct Extended Multi-Pedersen commitment scheme (González et al., Asiacrypt 2015). We further introduce the notion of functional SSB commitment schemes and, importantly, use it to get an efficient quasi-adaptive NIZK for arithmetic circuits and efficient oblivious database queries.

Keywords: Commitment scheme, oblivious transfer, QA-NIZK, SSB

1 Introduction

Commitment schemes are one of the most useful primitives in cryptography. In essence, a commitment to a value binds the value to the commitment, but hides the value from other parties. Commitment schemes are naturally used in zero-knowledge proofs, where one often proves statements about a committed value while keeping the value hidden. For instance, to complete a digital transaction a party may need to prove he has available funds in his account without actually revealing his exact balance. Such proofs on committed values are very efficient due to Bulletproofs [7], and are used in many privacy-preserving cryptocurrency designs such as Mimblewimble [21, 41] and Quisquis [20].

Dual-mode commitment schemes [9, 12, 13] are an interesting variant where the commitment key can be set up in one of two modes: binding or hiding. In the binding mode, the commitment can only be opened to one valid value. Meanwhile, in the hiding mode, a commitment hides the committed value even to unbounded adversaries. For this definition to make sense, one should not be able to guess which mode is being used based on the commitment key, i.e., the commitment key hides the mode. Dual-mode commitments are an essential tool in Groth-Sahai proofs [31] which is a framework for constructing non-interactive zero-knowledge (NIZK) proofs for algebraic relations.

In the case of committing to a vector, the two modes of a dual-mode commitment can be seen to be two extremes: the commitment is either binding in all

positions in the vector, or in none of them. A natural way to generalize the notion would be to have multiple modes of commitment, specifying that the commitment is binding in some positions in the vector of values. A similar generalization for hash functions is known as somewhere statistically binding hash [32, 39], in which one can compute a hash of a vector v such that the computed hash is statistically binding in one coordinate of v.

A generalization of dual-mode commitments would lead to interesting applications in NIZK arguments. In a typical zero-knowledge succinct argument of knowledge (zk-SNARK) for Circuit-SAT [14, 22, 29, 37], the prover commits to the witness (i.e., all the inputs to a circuit), and the proof of (knowledge) soundness involves using a non-falsifiable assumption to extract the whole committed vector which is then used to check each gate to establish where exactly the prover cheated; based on the knowledge of the witness one then breaks a computational assumption. One can get a more efficient extraction under falsifiable assumptions if the commitment was binding only on the values corresponding to the inputs and outputs of a specific gate: one then only needs to check the extracted values against a randomly chosen gate. As a caveat, the technique will lead to a security loss linear in the number of gates.

In fact, the above extraction technique has been done before [15, 28] using a generalization of the Pedersen commitment scheme called *Extended Multi-Pedersen* [26, 27] and resulting in efficient NIZK arguments under falsifiable assumptions. However, the above results are not zk-SNARKs: they are *quasi-adaptive* NIZK (QA-NIZK) arguments which means the CRS may depend on the relation, and while the argument is succinct, the commitment is not[1]. Moreover, previous work did not formalize which properties of a commitment scheme would be required to enable efficient NIZK arguments.

In the above construction, we need a succinct *somewhere statistically binding* property that guarantees that the chosen coordinate is statistically binding while the remaining coordinates can be computationally binding. On the other hand, to get zero-knowledge, the commitment needs to be *almost-everywhere statistically hiding*, that is, computationally hiding at the chosen coordinate, and statistically hiding at any other coordinates. We also need *index-set hiding*, which means an adversary that is given the commitment key does not know which particular coordinate is statistically binding.

Our Contributions. Formalizing the properties of the *Extended Multi-Pedersen* (EMP) commitment scheme [26, 27], we define a *somewhere statistically binding (SSB) commitment scheme* to n-dimensional vectors. In the commitment key generation phase of an SSB commitment scheme one chooses an index-set $\mathcal{S} \subseteq [1\,..\,n]$ of size at most $q \leq n$ and defines a commitment key $\mathtt{ck}$ that depends on n, q and $\mathcal{S}$. A commitment to an n-dimensional vector $\boldsymbol{x}$ will be statistically binding and extractable at coordinates indexed by $\mathcal{S}$ and perfectly hiding at all other coordinates. Moreover, commitment keys corresponding to any two index-

[1] One cannot construct zk-SNARKs in a black-box way from falsifiable assumptions [24], hence any black-box construction from falsifiable assumptions will not be fully succinct.

sets $\mathcal{S}_1$ and $\mathcal{S}_2$ of size at most q must be computationally indistinguishable. Thus, an *SSB commitment scheme* is required to be SSB, *somewhere statistically extractable* (SSE), *almost everywhere statistically hiding* (AESH), and *index-set hiding* (ISH). An SSB commitment scheme generalizes dual-mode commitment schemes (where $n = 1$ and $q \in \{0, 1\}$ determines the mode) and the EMP commitment scheme (where $q = 1$ and n is arbitrary).

In Section 4, we define algebraic commitment schemes (ACS), where the commitments keys are matrices. We prove that the distribution of key matrices defines which properties of SSB commitments hold in each coordinate and show that these commitments are suitable for working with QA-NIZK arguments. This is because they behave like linear maps and the properties of SSB commitments can be expressed in terms of membership to linear subspaces. Next, we generalize the EMP commitment scheme to work with arbitrary values of q. Importantly, a single EMP commitment consists of $q + 1$ group elements and is thus succinct given small q. We prove that EMP satisfies the mentioned security requirements under a standard Matrix DDH assumption [19].

In Section 5, we define *functional SSB* commitments, which are statistically binding on some components that are outputs of some functions $\mathcal{S} = \{f_i\}_i$ where $|\mathcal{S}| \leq q$. It is a generalization of SSB commitments, where the extracted values are the result of some linear functions of the committed values, instead of the values themselves. We show that results which hold for SSB commitments also naturally hold for functional SSB commitments. The notion of functional SSB commitments for families of linear functions was already used indirectly in prior work [15]; however, they were not formally defined and their security properties were not analyzed. We also see that a minor modification of EMP works as a functional SSB commitment if we consider only linear functions.

We provide some applications of functional SSB commitments. In Section 6.1 we propose a novel (but natural) application that we call oblivious database queries (ODQ), where a sender has a private database $\boldsymbol{x}$ and a receiver wants to query the database to learn $f_1(\boldsymbol{x}), \ldots, f_q(\boldsymbol{x})$ without revealing the functions f_i. In Section 6.2 we present a QA-NIZK for Square Arithmetic Programs (SAP, [30]) that follows a similar strategy to prior work [15] but can be used for arithmetic circuit satisfiability instead of Boolean circuit satisfiability. Our QA-NIZK has comparable efficiency and also under falsifiable assumptions.

Relation to other primitives. The SSB requirement makes the EMP commitment scheme look similar to SSB hash functions [32, 39], but there are obvious differences. SSB hash has the local opening property, where the committer can efficiently open just one coordinate of the committed vector, but SSB commitments do not[2]. Meanwhile, we need hiding while SSB hash does not. This is, intuitively, a natural distinction and corresponds to the difference between collision-resistant hash families and statistically hiding commitment schemes. Also, we allow `ck` to be long, but require commitments to be succinct.

[2] The properties of SSB and local opening are orthogonal: it is possible to construct efficient SSB hashes without local opening [39] and efficient vector commitments [8, 35] (which have a local opening) without the SSB property

SSB commitments are directly related to two-message oblivious transfer (OT) protocols as defined in [2]. Essentially, SSB commitments are non-interactive analogs of such protocols: the commitment key corresponds to the first OT message ot_1 and the commitment corresponds to the second OT message ot_2. Importantly, while in OT, the ot_1 generator is always untrusted, in our applications, it is sufficient to consider a trusted $\mathtt{ck}$ generator. This allows for more efficient constructions.

We discuss the relation to existing primitives in more detail in Appendix B.

2 Preliminaries

For a set S, let $\mathbb{P}(S)$ denote the power set (i.e., the set of subsets) of S, and let $\mathbb{P}(S,q)$ denote the set of q-size subsets of S. For an n-dimensional vector $\boldsymbol{\alpha}$ and $i \in [1\,..\,n]$, let α_i be its ith coefficient. Let $\boldsymbol{e}_i$ be the ith unit vector of implicitly understood dimension. For a tuple $\mathcal{S} = (\sigma_1, \ldots, \sigma_q)$ with $\sigma_i < \sigma_{i+1}$, let $\boldsymbol{\alpha}_{\mathcal{S}} = (\alpha_{\sigma_1}, \ldots, \alpha_{\sigma_q})$. Let $\boldsymbol{\alpha}_{\emptyset}$ be the empty string.

Let PPT denote probabilistic polynomial-time and let $\lambda \in \mathbb{N}$ be the security parameter. All adversaries will be stateful. Let $\mathsf{RND}_\lambda(\mathcal{A})$ denote the random tape of the algorithm $\mathcal{A}$ for a fixed λ. We denote by $\mathsf{negl}(\lambda)$ an arbitrary negligible function, and by $\mathsf{poly}(\lambda)$ an arbitrary polynomial function. Functions f, g are negligibly close, denoted $f \approx_\lambda g$, if $|f - g| = \mathsf{negl}(\lambda)$.

2.1 Bilinear Groups

In the case of groups, we will use additive notation together with the bracket notation [19], that is, for $\iota \in \{1, 2, T\}$ we define $[a]_\iota := a[1]_\iota$, where $[1]_\iota$ is a fixed generator of the group $\mathbb{G}_\iota$. A *bilinear group generator* $\mathsf{Pgen}(1^\lambda)$ returns $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e}, [1]_1, [1]_2)$, where p (a large prime) is the order of cyclic Abelian groups $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$. Moreover, $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficient non-degenerate bilinear pairing, such that $\hat{e}([a]_1, [b]_2) = [ab]_T$. Denote $[a]_1[b]_2 := \hat{e}([a]_1, [b]_2)$, and $[1]_T := [1]_1[1]_2$. We use matrix-vector notation freely, writing say $[\boldsymbol{M}_1]_1[\boldsymbol{M}_2]_2 = [\boldsymbol{M}_1\boldsymbol{M}_2]_T$ for any compatible matrices $\boldsymbol{M}_1$ and $\boldsymbol{M}_2$.

We use F-extraction notation to mean extraction of the function F. E.g., if F is exponentiation then we have $[\cdot]_\iota$-extraction, where we extract elements in the group $\mathbb{G}_\iota$. Several of our cryptographic primitives have their own parameter generator Pgen. In all concrete instantiations of the primitives, we instantiate Pgen with the bilinear group generator, which is then denoted Pgen_{bg}. Distribution families $\mathcal{D}^0 = \{\mathcal{D}^0_\lambda\}_\lambda$ and $\mathcal{D}^1 = \{\mathcal{D}^1_\lambda\}_\lambda$ are *computationally indistinguishable*, if $\forall$ PPT $\mathcal{A}$, $\Pr[x \leftarrow_\$ \mathcal{D}^0_\lambda : \mathcal{A}(x) = 1] \approx_\lambda \Pr[x \leftarrow_\$ \mathcal{D}^1_\lambda : \mathcal{A}(x) = 1]$.

The Matrix DDH (MDDH) assumption. Let $\ell, k \in \mathbb{N}$, with $\ell \geq k$, be small constants. Let p be a large prime. Following [19], we call $\mathcal{D}_{\ell k}$ a *matrix distribution* if it outputs, in polynomial time, matrices $\boldsymbol{A}$ in $\mathbb{Z}_p^{\ell \times k}$ of full rank k. We denote $\mathcal{D}_{k+1,k}$ by $\mathcal{D}_k$. Let $\mathcal{U}_{\ell k}$ denote the uniform distribution over $\mathbb{Z}_p^{\ell \times k}$.

Let Pgen be as before, and let $\iota \in \{1,2\}$. $\mathcal{D}_{\ell k}$-*MDDH*$_{\mathbb{G}_\iota}$ [19] holds relative to Pgen, if $\forall$ PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{A},\mathcal{D}_{\ell k},\iota,\mathsf{Pgen}}(\lambda) := |\varepsilon^0_{\mathcal{A}}(\lambda) - \varepsilon^1_{\mathcal{A}}(\lambda)| \approx_\lambda 0$, where

$$\varepsilon^\beta_{\mathcal{A}}(\lambda) := \Pr\begin{bmatrix} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathbf{A} \leftarrow_\$ \mathcal{D}_{\ell k}; \mathtt{w} \leftarrow_\$ \mathbb{Z}_p^k; \\ \boldsymbol{y}_0 \leftarrow_\$ \mathbb{Z}_p^\ell; \boldsymbol{y}_1 \leftarrow \mathbf{A}\mathtt{w} : \mathcal{A}(\mathsf{p}, [\mathbf{A}, \boldsymbol{y}_\beta]_\iota) = 1 \end{bmatrix} .$$

Common distributions for the MDDH assumption are $\mathcal{U}_k := \mathcal{U}_{k+1,k}$ and the linear distribution $\mathcal{L}_k$ over $\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A'} \\ 1 \ \cdots \ 1 \end{pmatrix}$, where $\boldsymbol{A'} \in \mathbb{Z}_p^{k\times k}$ is a diagonal matrix with $a'_{ii} \leftarrow_\$ \mathbb{Z}_p$.

2.2 Quasi-adaptive NIZK

A quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proof [33] enables one to prove membership in a language defined by a relation $\mathcal{R}_\rho$, which is determined by some parameter ρ sampled from a distribution $\mathcal{D}_{\mathsf{gk}}$.

A tuple of algorithms $(\mathsf{K}_0, \mathsf{K}_1, \mathsf{P}, \mathsf{V})$, where $\mathsf{gk} \leftarrow \mathsf{K}_0(1^\lambda), \mathtt{crs} \leftarrow \mathsf{K}_1(\mathsf{gk}, \rho), \pi \leftarrow \mathsf{P}(\mathtt{crs}, x, w), 0/1 \leftarrow \mathsf{V}(\mathtt{crs}, x, \pi)$, is a *QA-NIZK proof system* for witness-relations $\mathcal{R}_{\mathsf{gk}} = \{\mathcal{R}_\rho\}_{\rho\in\mathrm{sup}(\mathcal{D}_{\mathsf{gk}})}$, if it satisfies the following properties (see Appendix A for formal definitions): (i) Quasi-Adaptive Completeness: if $(x,w) \in \mathcal{R}_\rho$ then V accepts P's proof. (ii) Computational Quasi-Adaptive Soundness: if $\neg(\exists w : \mathcal{R}_\rho(x,w))$ then V accepts P's proof only with negligible probability. (iii) Perfect Quasi-Adaptive Zero-Knowledge: there exists a trapdoor τ and PPT simulator S such that for $(x,w) \leftarrow_\$ \mathcal{R}_\rho$, the distributions $\mathsf{P}(\mathtt{crs}, x, w)$ and $\mathsf{P}(\mathtt{crs}, \tau, x)$ are identical. We assume that $\mathtt{crs}$ contains an encoding of ρ, which is thus available to V. See Appendix A for more details.

3 SSB Commitment Schemes

In an SSB commitment scheme, the commitment key (i.e., the CRS) depends on n, q, and an index-set $\mathcal{S} \subseteq [1\,..\,n]$ of cardinality $\leq q$ (in the case of Groth-Sahai commitments [31], $n = q = 1$ while in the current paper $n = \mathsf{poly}(\lambda)$ and $q \geq 1$ is a small constant). At coordinates described by $\mathcal{S}$, an SSB commitment scheme must be *statistically binding* and *F-extractable* [5] for a well-chosen function F, while at all other coordinates it must be *statistically hiding* and *trapdoor*. Moreover, it must be index-set hiding, i.e., commitment keys corresponding to any two index-sets $\mathcal{S}_1$ and $\mathcal{S}_2$ of size $\leq q$ must be computationally indistinguishable.

The Groth-Sahai commitments correspond to a *bimodal* setting where either all coefficients are statistically hiding or statistically binding, and these two extremes are indistinguishable. SSB commitments correspond to a more fine-grained *multimodal* setting where some $\leq q$ coefficients are statistically binding and other coefficients are statistically hiding, and all possible selections of statistically binding coefficients are mutually indistinguishable. Our terminology is inspired by [32, 39] who defined SSB hashing; however, the consideration of the hiding property makes the case of SSB commitments sufficiently different.

Abbreviation	Property	Definition
ISH	Index-set hiding	The commitment key reveals nothing about the index-set $\mathcal{S}$
SSB	Somewhere statistically binding	A commitment to $\boldsymbol{x}$ statistically binds the values $\boldsymbol{x}_{\mathcal{S}}$
AESH	Almost everywhere statistically hiding	The commitment is statistically hiding in the indices outside the set $\mathcal{S}$
F-SSE	Somewhere statistical F-extractability	Given a commitment to $\boldsymbol{x}$ and the extraction key, one can extract the values $F(\boldsymbol{x}_S)$

Table 1. Properties of an SSB commitment scheme

3.1 Formalization and Definitions

An *F-extractable SSB commitment scheme* $\mathsf{COM} = (\mathsf{Pgen}, \mathsf{KC}, \mathsf{Com}, \mathsf{tdOpen}, \mathsf{Ext}_F)$ consists of the following polynomial-time algorithms:

Parameter generation: $\mathsf{Pgen}(1^\lambda)$ returns parameters p (e.g., description of a bilinear group).

Commitment key generation: for parameters p, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, and a tuple $\mathcal{S} \subseteq [1\,..\,n]$ with $|\mathcal{S}| \leq q$, $\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$ outputs a commitment key $\mathtt{ck}$ and a trapdoor $\mathtt{td} = (\mathtt{ek}, \mathtt{tk})$ consisting of an *extraction key* $\mathtt{ek}$, and a *trapdoor key* $\mathtt{tk}$. Also, $\mathtt{ck}$ implicitly specifies p, n, q, the message space $\mathtt{MSP}$, the randomizer space $\mathtt{RSP}$, and the commitment space $\mathtt{CSP}$, such that $F(\mathtt{MSP}) \subseteq \mathtt{ESP}$. For invalid input, KC outputs $(\mathtt{ck}, \mathtt{td}) = (\bot, \bot)$.

Commitment: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathtt{ck} \neq \bot$, a message $\boldsymbol{x} \in \mathtt{MSP}^n$, and a randomizer $r \in \mathtt{RSP}$, $\mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; r)$ outputs a commitment $c \in \mathtt{CSP}$.

Trapdoor opening: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathcal{S} \subseteq [1\,..\,n]$ with $|\mathcal{S}| \leq q$, $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$, two messages $\boldsymbol{x}, \boldsymbol{x}^* \in \mathtt{MSP}^n$, and a randomizer $r \in \mathtt{RSP}$, $\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}, r, \boldsymbol{x}^*)$ returns a randomizer $r^* \in \mathtt{RSP}$.

Extraction: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathcal{S} = (\sigma_1, \ldots, \sigma_{|\mathcal{S}|}) \subseteq [1\,..\,n]$ with $1 \leq |\mathcal{S}| \leq q$, $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$, $F : \mathtt{MSP} \to \mathtt{ESP}$ and $c \in \mathtt{CSP}$, $\mathsf{Ext}_F(\mathsf{p}, \mathtt{ek}; c)$ returns a tuple $(y_{\sigma_1}, \ldots, y_{\sigma_{|\mathcal{S}|}}) \in \mathtt{ESP}^{|\mathcal{S}|}$. We allow F to depend on p.

Note that SSB commitment schemes are non-interactive and work in the CRS model; the latter is needed to achieve trapdoor opening and extractability. With the current definition, *perfect completeness* is straightforward: to verify that C is a commitment of $\boldsymbol{x}$ with randomizer r, one just recomputes $C' \leftarrow \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; r)$ and checks whether $C = C'$.

An F-extractable SSB commitment scheme COM is *secure* if it satisfies the following security requirements. (See Table 1 for a brief summary.)

Index-Set Hiding (ISH): $\forall \lambda$, PPT $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| \approx_\lambda 0$, where $\varepsilon^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); (\mathcal{S}_0, \mathcal{S}_1) \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \forall i \in \{0,1\}, \mathcal{S}_i \subseteq [1\,..\,n] \wedge |\mathcal{S}_i| \leq q; \\ \beta \leftarrow_{\$} \{0,1\}; (\mathtt{ck}_\beta, \mathtt{td}_\beta) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_\beta) : \mathcal{A}(\mathtt{ck}_\beta) = \beta\end{array}\right].$$

Somewhere Statistically Binding (SSB): $\forall\lambda$, unbounded $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \approx_\lambda 0$, where $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p},n,q) \text{ s.t. } \mathcal{S} \subseteq [1\,..\,n] \wedge |\mathcal{S}| \le q;\\ (\mathtt{ck},\mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p},n,q,\mathcal{S}); (\boldsymbol{x}_0,\boldsymbol{x}_1,r_0,r_1) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} \neq \boldsymbol{x}_{1\mathcal{S}};\\ \mathsf{Com}(\mathtt{ck};\boldsymbol{x}_0;r_0) = \mathsf{Com}(\mathtt{ck};\boldsymbol{x}_1;r_1)\end{bmatrix} .$$

COM is *somewhere perfectly binding* (SPB) if $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) = 0$.

Almost Everywhere Statistically Hiding (AESH): $\forall\lambda$, unbounded adversary $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2\cdot|\varepsilon^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| \approx_\lambda 0$, where $\varepsilon^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p},n,q) \text{ s.t. } \mathcal{S} \subseteq [1\,..\,n] \wedge |\mathcal{S}| \le q;\\ (\mathtt{ck},\mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p},n,q,\mathcal{S}); (\boldsymbol{x}_0,\boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} = \boldsymbol{x}_{1\mathcal{S}};\\ \beta \leftarrow_\$ \{0,1\}; r \leftarrow_\$ \mathtt{RSP} : \mathcal{A}(\mathsf{Com}(\mathtt{ck};\boldsymbol{x}_\beta;r)) = \beta\end{bmatrix} .$$

COM is *almost everywhere perfectly hiding* (AEPH) if $\mathsf{Adv}^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) = 0$. If $\mathcal{A}$ is PPT, COM is *almost everywhere computationally hiding* (AECH).

Somewhere Statistical F-Extractability (F-SSE): $\forall\lambda$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathcal{S} = (\sigma_1,\ldots,\sigma_{|\mathcal{S}|})$ with $|\mathcal{S}| \le q$, $(\mathtt{ck},(\mathtt{ek},\mathtt{tk})) \leftarrow \mathsf{KC}(\mathsf{p},n,q,\mathcal{S})$, and PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\left[\boldsymbol{x}, r \leftarrow \mathcal{A}(\mathtt{ck}) : \mathsf{Ext}_F(\mathsf{p},\mathtt{ek};\mathsf{Com}(\mathtt{ck};\boldsymbol{x};r)) \neq (F(x_{\sigma_1}),\ldots,F(x_{\sigma_{|\mathcal{S}|}}))\right] \approx_\lambda 0 .$$

Additionally, an SSB commitment scheme can but does not have to be *trapdoor*.

Almost Everywhere Statistical Trapdoor (AEST): $\forall\lambda$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, and unbounded $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \approx_\lambda 0$, where $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) =$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p},n,q) \text{ s.t. } \mathcal{S} \subseteq [1\,..\,n] \wedge |\mathcal{S}| \le q;\\ (\mathtt{ck},\mathtt{td} = (\mathtt{ek},\mathtt{tk})) \leftarrow \mathsf{KC}(\mathsf{p},n,q,\mathcal{S}); (\boldsymbol{x}_0,r_0,\boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} = \boldsymbol{x}_{1\mathcal{S}};\\ r_1 \leftarrow \mathsf{tdOpen}(\mathsf{p},\mathtt{tk};\boldsymbol{x}_0,r_0,\boldsymbol{x}_1) : \mathsf{Com}(\mathtt{ck};\boldsymbol{x}_0;r_0) \neq \mathsf{Com}(\mathtt{ck};\boldsymbol{x}_1;r_1)\end{bmatrix} .$$

It is *almost everywhere perfect trapdoor (AEPT)* if $\mathsf{Adv}^{\mathrm{aest}}_{\mathsf{COM},n,q}(\lambda) = 0$.

It is important to consider the case $|\mathcal{S}| \le q$ instead of only $|\mathcal{S}| = q$. For example, when $q = n$, the PB commitment key ($|\mathcal{S}| = n$) has to be indistinguishable from the PH commitment key ($|\mathcal{S}| = 0$). Moreover, in the applications to construct QA-NIZK argument systems [15, 26, 27], one should not be able to distinguish between the cases $|\mathcal{S}| = 0$ and $|\mathcal{S}| = q$.

F-extractability [5] allows one to model the situation where $x_i \in \mathbb{Z}_p$ but we can only extract the corresponding bracketed value $[x_i]_\iota \in \mathbb{G}_\iota$; similar limited extractability is satisfied say by the Groth-Sahai commitment scheme for scalars [31]. Note that in this case, F depends on p. Interestingly, extractability implies SSB, see Appendix C.1 for a proof.

Lemma 1 (F-SSE & F is injective $\Rightarrow$ SSB). *Let* COM *be an SSB commitment scheme. Fix n and q. Assume F is injective. For all PPT $\mathcal{A}$, there exists a PPT $\mathcal{B}$ such that* $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \le 2\cdot\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{B},F,\mathsf{COM},n,q}(\lambda)$.

If $q = 0$ then AESH is equal to the standard statistical hiding (SH) requirement, and AEST is equal to the standard statistical trapdoor requirement. If $q = n$ then SSB is equal to the standard statistical binding (SB) requirement, and F-SSE is equal to the standard statistical F-extractability requirement. We will show that any secure SSB commitment scheme must also be computationally hiding and binding in the following sense.

Computational Binding (CB): $\forall$ PPT $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, where $\mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \mathcal{S} \subseteq [1\,..\,n] \wedge |\mathcal{S}| \leq q; \\ (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1, r_0, r_1) \leftarrow \mathcal{A}(\mathsf{ck}) \\ \text{s.t. } \boldsymbol{x}_0 \neq \boldsymbol{x}_1; \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_0; r_0) = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_1; r_1)\end{array}\right] \approx_\lambda 0 \ .$$

Computational Hiding (CH): $\forall$ PPT $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathrm{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| \approx_\lambda 0$, where $\varepsilon^{\mathsf{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \mathcal{S} \subseteq [1\,..\,n] \wedge |\mathcal{S}| \leq q; \\ (\mathsf{ck}, \mathsf{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathsf{ck}); \beta \leftarrow_\$ \{0,1\}; \\ r \leftarrow_\$ \mathsf{RSP} : \mathcal{A}(\mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_\beta; r)) = \beta\end{array}\right] \ .$$

Theorem 1. *Let* COM *be an SSB commitment scheme. Fix* n *and* q*.*

(i) (ISH + SSB $\Rightarrow$ CB) For all PPT $\mathcal{A}$, there exist PPT $\mathcal{B}_1$ and unbounded $\mathcal{B}_2$, such that $\mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda) + n/(q - 4 \cdot \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda)) \cdot \mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda)$.*

(ii) (ISH + AESH $\Rightarrow$ CH) For all PPT $\mathcal{A}$, there exist PPT $\mathcal{B}_1$ and unbounded $\mathcal{B}_2$, such that $\mathsf{Adv}^{\mathrm{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda) + \mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda)$.

The full proof of this theorem is deferred to Appendix C.2.

4 Constructing SSB Commitment Schemes

In this section we generalize the notion of algebraic commitment scheme to general matrix distributions. We show that they work nicely with QA-NIZK arguments and that certain matrix distributions give us an SSB commitment scheme. We focus on the particular case of EMP in Section 4.2, where we propose a general version of EMP and prove that it is an SSB commitment scheme.

4.1 Algebraic Commitment Schemes

Ràfols and Silva [42] defined the notion of *algebraic commitment schemes (ACSs)*, where the commitment keys are matrices, already used implicitly in other works [10, 11]. Since they behave like linear maps, it is very natural to work with them. We give a more general definition in the following where the matrices are sampled from general distributions.

Definition 1. *Let $\iota \in \{1,2\}$, and let n, m, k be small integers. Let $\mathcal{D}_1$ be a distribution of matrices from $\mathbb{G}_\iota^{k\times n}$ and let $\mathcal{D}_2$ be a distribution of matrices from $\mathbb{G}_\iota^{k\times m}$. A commitment scheme* COM *is a* $(\mathcal{D}_1, \mathcal{D}_2)$-algebraic commitment scheme (ACS) *for vectors in $\mathbb{Z}_p^n$, if for commitment key $\mathsf{ck} = [\boldsymbol{U}_1, \boldsymbol{U}_2]_\iota \leftarrow_\$ \mathcal{D}_1 \times \mathcal{D}_2$ the commitment of a vector $\boldsymbol{x} \in \mathbb{Z}_p^n$ is computed as a linear map of $\boldsymbol{x}$ and randomness $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^m$, i.e., $\mathsf{Com}_{\mathsf{ck}}(\boldsymbol{x}, \boldsymbol{r}) := [\boldsymbol{U}_1]_\iota \boldsymbol{x} + [\boldsymbol{U}_2]_\iota \boldsymbol{r} \in \mathbb{G}_\iota^k$.*

Ràfols and Silva mention that given different commitment key matrices, their distributions are computationally indistinguishable under the MDDH assumption, and each concrete distribution defines which coordinates of the commitments are SB or SH. We prove in Appendix D.1 that it also gives a characterization of the coordinates of the key matrices for the different SSB properties (AECH, ISH, SPB, SPE) based on linear dependency. In Appendix D.1 we also prove that to extract n elements from an ACS we need at least $n+1$ rows.

4.2 The EMP Commitment Scheme

Extended Multi-Pedersen (EMP) [26,27] is a variant of the standard vector Pedersen commitment scheme [40]. In this section, we will depict a general version of the EMP commitment scheme[3] in group $\mathbb{G}$. We redefine EMP by using a division of the generator matrix $\boldsymbol{g}$ as a product of two matrices $\boldsymbol{R}$ and $\boldsymbol{M}$; this representation results in very short security proofs for EMP. To simplify notation, we will write Ext instead of $\mathsf{Ext}_{[\cdot]}$. We use a distribution $\mathcal{D}_{q+1}^{p,n,\mathcal{S}}$ that outputs $n+1$ vectors $\boldsymbol{g}^{(i)}$, such that if $i \in \mathcal{S}' = \mathcal{S} \cup \{n+1\}$ then $\boldsymbol{g}^{(i)}$ is distributed uniformly over $\mathbb{Z}_p^{q+1}$, and otherwise $\boldsymbol{g}^{(i)}$ is a random scalar multiple of $\boldsymbol{g}^{(n+1)}$.[4]

Definition 2. *Let $p = p(\lambda)$, $n = \mathsf{poly}(\lambda)$, and let $q \le n$ be a small positive integer. Let $\mathcal{S} \subseteq [1\,..\,n]$ with $|\mathcal{S}| \le q$. Then the distribution $\mathcal{D}_{q+1}^{p,n,\mathcal{S}}$ is defined as the first part of $\mathcal{D}_{gen}(p, n, \mathcal{S}, q)$ in Fig. 1 (i.e., just $\boldsymbol{g}$, without the associated extraction key or trapdoor).*

Note that [27] uses a distribution $\mathcal{D}_{q+1,k}$ instead of the uniform distribution $\mathcal{U}_{q+1}$ over $\mathbb{Z}_p^{q+1}$, which means that taking a larger k gives a weaker security assumption but with worse efficiency. Our version of EMP also works with a general distribution, but for ease of presentation we only use $\mathcal{U}_{q+1}$.

Example 1. In the Groth-Sahai commitment scheme, $n = q = 1$, so $\mathcal{D}_{gen}$ first samples $\boldsymbol{R} = \left(\begin{smallmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{smallmatrix}\right) \leftarrow_\$ \mathbb{Z}_p^{2\times 2}$. If $\mathcal{S} = \{1\}$ then $\boldsymbol{M} = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$ and $\boldsymbol{g} = \boldsymbol{RM} = \left(\begin{smallmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{smallmatrix}\right)$. On the other hand, if $\mathcal{S} = \emptyset$ then $\boldsymbol{M} = \left(\begin{smallmatrix} 0 & 0 \\ \delta_1 & 1 \end{smallmatrix}\right)$ and $\boldsymbol{g} = \boldsymbol{RM} = \left(\begin{smallmatrix} \delta_1 r_{12} & r_{12} \\ \delta_1 r_{22} & r_{22} \end{smallmatrix}\right)$ for $\delta_1 \leftarrow_\$ \mathbb{Z}_p$.

Consider the case $n = 3$, $q = 2$, and $\mathcal{S} = \{3\}$. Then

$$\boldsymbol{M} = \left(\begin{smallmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ \delta_1 & \delta_2 & 0 & 1 \end{smallmatrix}\right), \quad \boldsymbol{g} = \boldsymbol{RM} = \left(\begin{smallmatrix} \delta_1 r_{13} & \delta_2 r_{13} & r_{11} & r_{13} \\ \delta_1 r_{23} & \delta_2 r_{23} & r_{21} & r_{23} \\ \delta_1 r_{33} & \delta_2 r_{33} & r_{31} & r_{33} \end{smallmatrix}\right), \text{ for } \delta_1, \delta_2 \leftarrow_\$ \mathbb{Z}_p, \boldsymbol{R} \leftarrow_\$ \mathbb{Z}_p^{3\times 3} \ .$$

[3] González *et al.* [27] mostly considered the case $q = 1$; they also did not formalize its security by using notions like ISH

[4] We add $+1$ to the dimension (e.g., $q+1$) to accommodate the randomizer in EMP.

$\mathcal{D}_{gen}(p, n, \mathcal{S}, q)$

$\mathcal{S}' \leftarrow \mathcal{S} \cup \{n+1\}$; // $\mathcal{S}' = \{\sigma_1, \ldots, \sigma_{q+1}\}$
$\boldsymbol{R} \leftarrow_\$ \mathbb{Z}_p^{(q+1)\times(q+1)}$; $\boldsymbol{M} \leftarrow \mathbf{0}_{(q+1)\times(n+1)}$; $M_{q+1,n+1} \leftarrow 1$;
for $j = 1$ **to** n **do**
  if $j \notin \mathcal{S}'$ **then** $M_{q+1,j} = \delta_j \leftarrow_\$ \mathbb{Z}_p$; **else** let i be such that $j = \sigma_i$; $M_{i,\sigma_i} \leftarrow 1$;
endfor
$\boldsymbol{g} \leftarrow \boldsymbol{R}\boldsymbol{M}$; $\mathtt{tk} \leftarrow (\delta_j)_{j\in[1\,..\,n]\setminus\mathcal{S}}$; // $\boldsymbol{g} \in \mathbb{Z}_p^{(q+1)\times(n+1)}$;
return $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk})$;

Fig. 1. Generating $\mathcal{D}_{q+1}^{p,n,\mathcal{S}}$, with associated extraction key $\boldsymbol{R}$ and trapdoor $\mathtt{tk}$

$\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$: // $\mathcal{S} \subseteq \{1, 2, \ldots, n\}$ with $|\mathcal{S}| \le q$

Sample $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk}_\iota) \leftarrow_\$ \mathcal{D}_{gen}(p, n, \mathcal{S}, q)$ s.t. $\boldsymbol{R}$ has full rank;
$\mathtt{ck} \leftarrow [\boldsymbol{g}]$; $\mathtt{ek} \leftarrow \boldsymbol{R}$; // $\boldsymbol{g} \in \mathbb{Z}_p^{(q+1)\times(n+1)}$, $\boldsymbol{R} \in \mathbb{Z}_p^{(q+1)\times(q+1)}$
$\mathtt{td} \leftarrow (\mathtt{ek}, \mathtt{tk})$; **return** $(\mathtt{ck}, \mathtt{td})$;

$\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}_\iota; \boldsymbol{x}, r, \boldsymbol{x}^*)$

$r^* \leftarrow \sum_{i\in[1\,..\,n]\setminus\mathcal{S}}(x_i - x_i^*)\delta_i + r$;
return r^*;

$\mathsf{Ext}(\mathsf{p}, \mathsf{ek}; [\mathbf{c}])$

$[\boldsymbol{x}'] \leftarrow \boldsymbol{R}^{-1}[\mathbf{c}]$;
return $[\boldsymbol{x}_{\mathcal{S}}] \leftarrow [\boldsymbol{x}'_{[1\,..\,|\mathcal{S}|]}]$;

$\mathsf{Com}(\mathtt{ck}; \boldsymbol{x} \in \mathbb{Z}_p^n; r \in \mathbb{Z}_p)$

return $[\boldsymbol{g}]\binom{\boldsymbol{x}}{r}$; // $= \sum_{j=1}^n x_j[\boldsymbol{g}^{(j)}] + r[\boldsymbol{g}^{(n+1)}] \in \mathbb{G}^{q+1}$

Fig. 2. The EMP commitment scheme COM

The following lemma shows that distributions $[\mathcal{D}_{q+1}^{p,n,\mathcal{S}}]$ for different sets $\mathcal{S}$ are indistinguishable under the MDDH assumption. See Appendix D.2 for a proof.

Lemma 2. *Let $\iota \in \{1, 2\}$. Let $p = p(\lambda)$ be created by $\mathsf{Pgen}(1^\lambda)$, $n = \mathsf{poly}(\lambda)$, and let $q \le n$ be a positive integer. Let $\mathcal{S} \subseteq [1\,..\,n]$ with $|\mathcal{S}| \le q$. The distribution families $\mathcal{D}^0 := \{[\mathcal{D}_{q+1}^{p,n,\mathcal{S}}]\}_\lambda$ and $\mathcal{D}^1 := \{[\mathcal{D}_{q+1}^{p,n,\emptyset}]\}_\lambda$ are computationally indistinguishable under the $\mathcal{U}_{q+1}$-MDDH$_{\mathbb{G}_\iota}$ assumption relative to Pgen: for any PPT $\mathcal{A}$, there exists a PPT $\mathcal{B}$, such that $\mathsf{Adv}^{\mathrm{indist}}_{\mathcal{A},\mathcal{D}^0,\mathcal{D}^1}(\lambda) \le |\mathcal{S}| \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{B},\mathcal{U}_{q+1},\mathsf{Pgen}}(\lambda)$.*

We define EMP in Fig. 2. We claim that it is indeed an SSB commitment scheme in the following Theorem, see Appendix D.3 for a proof.

Theorem 2. *Let Pgen_{bg} be a bilinear group generator. Fix λ, n, and q. The EMP commitment scheme is (i) ISH under the $\mathcal{U}_{(q+1)\times(n+1)}$-MDDH$_{\mathbb{G}_\iota}$ assumption, (ii) F-SSE for $F = [\cdot]$ (thus, F depends on p), (iii) AEPT, (iv) SPB, (v) AEPH, (vi) CB and CH under the $\mathcal{U}_{(q+1)\times(n+1)}$-MDDH$_{\mathbb{G}_\iota}$ assumption.*

Alternative constructions. One can also construct a SSB commitment from any IND-CPA secure cryptosystem if both the message space and the random-

ness space are additively homomorphic, i.e., $\mathsf{Enc}_{\mathsf{pk}}(m_1; r_1) + \mathsf{Enc}_{\mathsf{pk}}(m_2; r_2) = \mathsf{Enc}_{\mathsf{pk}}(m_1 + m_2; r_1 + r_2)$ for any public key pk, messages m_1, m_2 and randomness $r_1, r_2 \in \mathcal{R}$. For simplicity, consider the case when $q = 1$ and the i-th index is binding. We can set $\mathtt{ck} = (\mathsf{pk}, \boldsymbol{c} := (\mathsf{Enc}_{\mathsf{pk}}(e_{i,1}; r_1), \ldots, \mathsf{Enc}_{\mathsf{pk}}(e_{i,n}; r_n)), \mathtt{tk} = \mathsf{sk}$ where $\boldsymbol{e}_i$ is the i-th unit vector. In order to commit to $\boldsymbol{x}$, we compute $\boldsymbol{c} \cdot \boldsymbol{x} + \mathsf{Enc}_{\mathsf{pk}}(0; r) = \mathsf{Enc}_{\mathsf{pk}}(x_i, r + \sum_{i=1}^{n} r_i)$ for $r \leftarrow_{\$} \mathcal{R}$. Now, ISH follows directly from the IND-CPA security, SSB and F-SSE follow from the correctness of the cryptosystem, and AESH follows since $\mathsf{Enc}_{\mathsf{pk}}(x_i, r + \sum_{i=1}^{n} r_i)$ only depends on x_i. However, we obtain a less efficient construction than EMP. E.g., if we instantiate with Elgamal we would have a commitment size of $2q$ group elements, whereas EMP has $q+1$.

The above is similar to the technique of obtaining 2-message oblivious transfer (OT) from additively homomorphic cryptosystems [2] and this is no coincidence. SSB commitments can indeed be constructed from OT, and we can conversely construct OT from SSB commitments. Hence there are various alternative constructions of SSB, but in this paper we concentrate on EMP due to the applications we are interested in. See Appendix B.2 for more details.

5 Functional SSB Commitments

We generalize the notion of SSB commitment from being statistically binding on an index-set $\mathcal{S} \subseteq [1\,..\,n]$ to being statistically binding on outputs of the functions $\{f_i\}_{i=1}^{q}$ from some function family $\mathcal{F}$. We construct a functional SSB commitment for the case when $\mathcal{F}$ is the set of linear functions. In particular, this covers functions $f_j(\boldsymbol{x}) = x_j$ and hence we also have the index-set functionality of EMP commitment.

In our definition, given a family of functions $\mathcal{F}$ we require that the commitment key $\mathtt{ck}$ will hide the functions $\{f_i\}_{i=1}^{q} \subset \mathcal{F}$ and given a commitment $\mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; r)$ and an extraction key $\mathtt{ek}$ it is possible to F-extract $f_i(\boldsymbol{x})$ for $i \in [1\,..\,q]$. The commitment uniquely determines the outputs of the functions (due to the SSB property) and commitments to messages which produce equal function outputs are statistically indistinguishable (due to the AESH property). Our definition is similar to Döttling et al.'s [16] definition for trapdoor hash functions for a family of predicates $\mathcal{F}$.

Definition of functional SSB. An *F-extractable functional SSB commitment scheme* $\mathsf{COM} = (\mathsf{Pgen}, \mathsf{KC}, \mathsf{Com}, \mathsf{tdOpen}, \mathsf{Ext}_F)$ for a function family $\mathcal{F}$ follows the definitions of SSB commitments in Section 3.1, but with the following changes: (i) $\mathcal{S}$ is now a set of functions rather than a set of indices. (ISH then becomes function set hiding (FSH)). (ii) For $\mathcal{S} = \{f_i\}_{i=1}^{q} \subseteq \mathcal{F}$ and vector $\boldsymbol{x}$ we redefine $\boldsymbol{x}_{\mathcal{S}} := (f_1(\boldsymbol{x}), \ldots, f_q(\boldsymbol{x}))$. The full definitions are given in Appendix E.1. Relations that hold between properties of SSB commitments also hold for functional SSB commitments; the proofs are very similar.

Linear EMP. We construct a functional SSB commitment for a family of linear functions. Our construction follows the ideas in [15] which only dealt with some concrete functions and never formalized the ideas.

$\mathsf{KC}_\iota(\mathsf{p}, n, q, \boldsymbol{M} \in \mathbb{Z}_p^{q\times n})$:

Set implicitly $\mathtt{MSP} = \mathtt{RSP} = \mathbb{Z}_p^n$ and $\mathtt{CSP} = \mathbb{G}_\iota^{q+1}$;
Sample $\boldsymbol{R} \leftarrow\!\!\$\, \mathbb{Z}_p^{(q+1)\times(q+1)}$ so that it has full rank; Sample $\boldsymbol{r} \leftarrow\!\!\$\, \mathbb{Z}_p^n$;
Set $\boldsymbol{M}' \leftarrow \left(\begin{smallmatrix} \boldsymbol{M} & \mathbf{0} \\ \boldsymbol{r}^\intercal & 1 \end{smallmatrix}\right) \in \mathbb{Z}_p^{(q+1)\times(n+1)}$;
Set $\mathtt{ck} \leftarrow [\boldsymbol{R}\boldsymbol{M}']_\iota \in \mathbb{G}_\iota^{(q+1)\times(n+1)}$, $\mathtt{td} \leftarrow (\mathtt{ek} \leftarrow \boldsymbol{R}^{-1}, \mathtt{tk} \leftarrow \boldsymbol{r})$;
return $(\mathtt{ck}, \mathtt{td})$;

$\mathsf{Com}(\mathtt{ck}; \boldsymbol{x} \in \mathbb{Z}_p^n; r \in \mathbb{Z}_p)$

return $\mathtt{ck}\left(\begin{smallmatrix} \boldsymbol{x} \\ r \end{smallmatrix}\right)$;

$\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}_\iota; \boldsymbol{x}, r, \boldsymbol{x}^*) \;/\!\!/\; \boldsymbol{M}\boldsymbol{x} = \boldsymbol{M}\boldsymbol{x}^*$

return $r^* \leftarrow \sum_{i\in[1\,..\,n]} (x_i - x_i^*)\mathtt{tk}_i + r$;

$\mathsf{Ext}(\mathsf{p}, \mathtt{ek}; [\mathbf{c}]_\iota)$

return $\mathtt{ek}[\boldsymbol{c}]_\iota$ without the last element;

Fig. 3. Functional SSB commitment for linear functions

We represent q linear functions by a matrix $\boldsymbol{M} \in \mathbb{Z}_p^{q\times n}$ where each row contains coefficients of one function. From a commitment to vector $\boldsymbol{x} \in \mathbb{Z}_p^n$, our construction allows to extract $[\boldsymbol{M}\boldsymbol{x}]_\iota$. In particular, if we take $\boldsymbol{M} = (\boldsymbol{e}_{i_1}|\ldots|\boldsymbol{e}_{i_q})^\top$ where $\boldsymbol{e}_{i_j} \in \mathbb{Z}_p^n$ is the i_jth unit vector, then $[\boldsymbol{M}\boldsymbol{x}]_\iota = [x_{i_1}, \ldots, x_{i_q}]_\iota^\top$. A detailed construction is given in Fig. 3. Moreover, if we take an ACSP, the commitment key is $\mathtt{ck} = [\boldsymbol{U}_1, \boldsymbol{U}_2]_\iota \in \mathbb{G}_\iota^{(q+1)\times n} \times \mathbb{G}_\iota^{(q+1)\times 1}$, which is optimal size for extraction in q coordinates, as proven in Corollary 1. The main differences with the EMP construction in Section 4.2 is that in EMP $\boldsymbol{M}$ is a matrix in reduced row echelon form (with multiples of the column vector $(0, \ldots, 0, 1)^T$ possibly inserted in between). We prove security of linear EMP in Appendix E.2.

6 Applications of Functional SSB Commitments

We present three applications of functional SSB commitments. In Section 6.1 we have two straightforward applications for linear EMP commitments: Oblivious Database Queries (ODQ) and Oblivious Linear Function Evaluation (OLE) [17, 18, 25]. OLE allows the receiver to learn $f(\boldsymbol{x})$ where $\boldsymbol{x}$ is the receiver's private vector and f is the sender's private linear function. ODQ essentially switches the roles of receiver and sender: the receiver wants to learn $f(\boldsymbol{x})$ where $\boldsymbol{x}$ is the sender's private database and f is the receiver's linear query function. In Section 6.2 we present a new QA-NIZK argument for SAP relations that uses linear EMP commitments as a technical tool in the security proof.

6.1 ODQ & OLE

A very straight-forward application of linear EMP is oblivious database queries (ODQ). We consider a scenario where the sender knows a private database $\boldsymbol{x}$

and the receiver knows a set of private linear functions $f_i(X_1, \ldots, X_n) = b_i + \sum_{j=1}^{n} a_{i,j} X_j$ for $i \in [1\,..\,q]$ that he wants to evaluate on that database.

Our ODQ protocol works as follows:

- Receiver defines matrices $\boldsymbol{A} = (a_{ij}) \in \mathbb{Z}_p^{q \times n}$, $\boldsymbol{B} = \mathrm{diag}(b_1, \ldots, b_q) \in \mathbb{Z}_p^{q \times q}$, and $\boldsymbol{M} = (\boldsymbol{A} \mid \boldsymbol{B}) \in \mathbb{Z}_p^{q \times (n+q)}$. Following the KC algorithm it creates the commitment key $\mathtt{ck}$, the extraction key $\mathtt{ek}$, and sends $\mathtt{ck}$ to the sender.
- Sender has $\boldsymbol{x} \in \mathbb{Z}_p^n$ and $\mathtt{ck}$ as input. It sets $\boldsymbol{x'} = \left(\begin{smallmatrix} \boldsymbol{x} \\ \mathbf{1}_q \end{smallmatrix}\right)$, picks random $r \leftarrow_{\$} \mathbb{Z}_p$ and sends $\mathsf{COM} = \mathtt{ck}\left(\begin{smallmatrix} \boldsymbol{x'} \\ r \end{smallmatrix}\right)$ to the receiver.
- Receiver extracts $[\boldsymbol{M} \cdot \boldsymbol{x'}]$ from COM using the Ext algorithm with $\mathtt{ek}$.

Privacy and Correctness. We follow privacy and correctness definitions proposed by Döttling et al. [16] (see Section 5.1 of their paper for full definitions). From the SSE property we know that the receiver can recover $[\boldsymbol{M}\left(\begin{smallmatrix} \boldsymbol{x} \\ \mathbf{1}_q \end{smallmatrix}\right)]_\iota = [\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}]_\iota$ and thus correctness holds. Receiver's (computational) privacy follows directly from the FSH property, that is, any two function sets of size at most q are indistinguishable. Sender's privacy is defined through simulatability of the protocol transcript given only receiver's input $\boldsymbol{M}$ and receiver's output $[\boldsymbol{M}\boldsymbol{x'}]$ to the simulator. Simulatability is slightly stronger than the AEPH property but still holds for linear EMP. As a first message, the simulator can generate $\mathtt{ck}$ with $\boldsymbol{M}$ and store $\boldsymbol{R}$. An honestly computed second message has the form $[\boldsymbol{R}\left(\begin{smallmatrix} \boldsymbol{M} & \mathbf{0} \\ \boldsymbol{r}^\intercal & 1 \end{smallmatrix}\right)]\left(\begin{smallmatrix} \boldsymbol{x'} \\ r \end{smallmatrix}\right) = \boldsymbol{R}\left[\begin{smallmatrix} \boldsymbol{M}\boldsymbol{x'} \\ \boldsymbol{x'}\boldsymbol{r}^\top + r \end{smallmatrix}\right]$ and therefore we can simulate it by sampling $r^* \leftarrow_{\$} \mathbb{Z}_p$ and computing $\boldsymbol{R}\left(\begin{smallmatrix} [\boldsymbol{M}\boldsymbol{x'}] \\ r^* \end{smallmatrix}\right)$. Thus sender's privacy also holds.

Efficiency. We define download rate as the ratio between output size and sender's message and total rate as the ratio between output size and total transcript size. The total rate of our protocol is $|[\boldsymbol{M}\boldsymbol{x'}]|/(|\mathtt{ck}| + |\mathsf{COM}|) = q/((n + q + 2)(q + 1))$. However, we achieve very good download rate $|[\boldsymbol{M}\boldsymbol{x'}]|/|\mathsf{COM}| = q/(q + 1)$ which tends to 1. This is similar to Döttling et al. [16] where they achieve an optimal download rate but sub-optimal total rate.

OLE. We can achieve OLE in a very similar way. Suppose that now the sender has a function $f(X_1, \ldots, X_n) = b + \sum_{i=1}^{n} a_i X_i$ and the receiver has $\boldsymbol{x}$. Then the receiver can send a commitment key with $\boldsymbol{M} = (x_1, \ldots, x_n, 1)$ and the sender responds with a commitment to $(a_1, \ldots, a_n, b)$. The receiver extracts to obtain $[f(\boldsymbol{x})]_\iota$. The proof is identical to the ODQ case. However, the resulting OLE is less efficient with download rate $1/2$ and total rate $1/(2n + 4)$.

6.2 QA-NIZK Argument for Quadratic Equations

We present a QA-NIZK argument which uses linear EMP commitments as an important technical tool in the security proof, inspired by Daza et al. [15] who presented a commit-and-prove QA-NIZK argument for Square Span Programs (SSP, [14]) which can be used to encode the Boolean circuit satisfiability language. Their construction uses a specific setting of linear EMP commitments without explicitly formalizing it. Our QA-NIZK is for Square Arithmetic Programs (SAP) [30] which can be used to encode the arithmetic circuit satisfiability language, has roughly the same complexity as the argument in [15] and follows

a similar overall strategy. However, we use linear EMP commitments as a black-box and thus have a more compact and clear presentation.

A rough intuition of our commit-and-prove QA-NIZK is as follows. The statement of our language $\mathscr{L}_{\mathtt{SAP},\mathtt{ck}}$ contains a linear-length perfectly binding (and $[\cdot]_1$-extractable) commitment $[\boldsymbol{c}]_1$ of the SAP witness. Note that the commitment is only computed once but can be reused for many different SAP relations. For simplicity, we use ElGamal encryption in this role (see PB commitment in Appendix F.1) and the commitment key $\mathtt{ck}$ as a parameter of the language. The argument itself is succinct and contains the following elements:

- a succinct SNARK-type argument $[V, H, W]_1$, $[V]_2$ for the SAP relation,
- a succinct linear EMP commitment $[\tilde{\boldsymbol{c}}]_2$ that commits to the SAP witness and to the randomness of the SNARK,
- a succinct linear subspace argument bls [26] that shows that commitments open to consistent values (see bls argument in Appendix F.1). I.e., it guarantees that the opening of $[\boldsymbol{c}]_1$ is also used in the SNARK and in $[\tilde{\boldsymbol{c}}]_2$.

Square Arithmetic Program (SAP). A square arithmetic program is a tuple $\mathtt{SAP} = (\mathsf{p}, n, d, \mathbf{V} \in \mathbb{Z}_p^{n\times d}, \mathbf{W} \in \mathbb{Z}_p^{n\times d})$. We define a commit-and-prove language for $\mathtt{SAP}$ as the following language with n variables and d quadratic equations

$$\mathscr{L}_{\mathtt{SAP},\mathtt{ck}} = \left\{ [\boldsymbol{c}]_1 \in \mathbb{G}_1^{2n} \;\middle|\; \begin{array}{c} \exists \boldsymbol{a}, \boldsymbol{r} \in \mathbb{Z}_p^n\colon [\boldsymbol{c}]_1 = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{r}) \wedge \\ \left\{ \left(\boldsymbol{a}^\top \boldsymbol{v}_j\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_j = 0 \right\}_{j=1}^{d} \end{array} \right\}$$

where Com_{ck} is a perfectly binding commitment scheme, $\boldsymbol{v}_j$ is j-th column of the matrix $\boldsymbol{V}$ and $\boldsymbol{w}_j$ is the j-th column of the matrix $\boldsymbol{W}$.

QA-NIZK Argument scheme. Given $n, d \in \mathbb{N}$ we construct a QA-NIZK argument for $\mathscr{L}_{\mathtt{SAP},\mathtt{ck}}$.

- $\mathsf{K}_0(\lambda)$ returns $\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda)$.
- $\mathcal{D}_{\mathsf{p}}(n, d)$ returns a commitment key $\mathtt{ck} = [\boldsymbol{u}]_1 = [1, u]_1^\top$ where $u \leftarrow_\$ \mathbb{Z}_p$.
- $\mathsf{K}_1(\mathsf{p}, n, d, \mathtt{ck})$ picks $s \leftarrow_\$ \mathbb{Z}_p$, then sets $q_v = 4$, $n' = n + 1$, $\boldsymbol{M} = \mathbf{0} \in \mathbb{Z}_p^{q_v \times n'}$ (i.e., $S_v = \emptyset$) and generates a linear EMP key $\mathtt{ck}' = [\mathbf{K}]_2 \leftarrow \mathsf{KC}_2(\mathsf{p}, n', q_v, \boldsymbol{M}) \in \mathbb{G}_2^{5\times(n+2)}$. Finally, it runs $(\mathsf{crs}_{\mathsf{bls}}, \mathsf{td}_{\mathsf{bls}}) \leftarrow \mathsf{K}_{\mathsf{bls}}([\boldsymbol{N}_1]_1 \in \mathbb{G}_1^{(2n+2)\times(2n+3)}, [\boldsymbol{N}_2]_2 \in \mathbb{G}_2^{5\times(2n+3)})$ for

$$[\mathbf{N}_1]_1 = \left[\begin{array}{ccc|ccc|c} \boldsymbol{e}_2 & & & \boldsymbol{u} & & & \\ & \ddots & & & \ddots & & \mathbf{0} \\ & & \boldsymbol{e}_2 & & & \boldsymbol{u} & \\ \hline v_1(s) & \dots & v_n(s) & & \mathbf{0} & & t(s) \;\; 0 \;\; 0 \\ w_1(s) & \dots & w_n(s) & & & & 0 \;\; t(s) \; 0 \end{array}\right]_1,$$

$$[\mathbf{N}_2]_2 = \left[\begin{array}{ccc|c|c} v_1(s) & \dots & v_n(s) & & & t(s) & 0\,0 \\ \mathbf{K}^{(1)} & \dots & \mathbf{K}^{(n)} & \multicolumn{1}{c|}{\mathbf{0}} & & \mathbf{K}^{(n+1)} & 0\,\mathbf{K}^{(n+2)} \end{array}\right]_2.$$

Return the CRS $\mathtt{crs} = (\mathsf{p}, \mathtt{ck}, \mathtt{ck}', \{[s^i]_{1,2}\}_{i=1}^d, \mathsf{crs}_{\mathsf{bls}})$ with trapdoor $(s, \mathsf{td}_{\mathsf{bls}})$.

– The prover P receives an input $(\mathtt{crs}, ([\boldsymbol{c}]_1, \mathbf{V}, \mathbf{W}), (\boldsymbol{a}, \boldsymbol{r}))$. Let $v_i(X)$ and $w_i(X)$ be the interpolation polynomials at some points $\{\chi_j\}_j$ for the i-th column of $\mathbf{V}$ and $\mathbf{W}$ respectively for $i \in [1\,..\,n]$, and set $t(X) = \prod_{i=j}^{d}(X - \chi_j)$. The prover picks $\delta_v, \delta_w, r_v \leftarrow_\$ \mathbb{Z}_p$ and defines:

$$\begin{array}{ll} V(X) := \sum_{i=1}^{n} a_i v_i(X) + \delta_v t(X), & W(X) := \sum_{i=1}^{n} a_i w_i(X) + \delta_w t(X) \\ P(X) := V(X)^2 - W(X) & H(X) := P(X)/t(X) \end{array} \tag{1}$$

The prover computes group elements $[V]_{1,2} = [V(s)]_{1,2}$, $[W]_1 = [W(s)]_1$, $[H]_1 = [H(s)]_1$ and a linear EMP commitment $[\tilde{\boldsymbol{c}}]_2 = \mathsf{Com}(\mathsf{ck}'; (\boldsymbol{a}, \delta_v), r_v)$. The prover also computes a bls argument ψ for the statement $\mathtt{x}_{\mathsf{bls}} := ([\boldsymbol{c}]_1, [V]_1, [W]_1, [V]_2, [\tilde{\boldsymbol{c}}]_2)^\top \in \mathbf{Im}\begin{pmatrix} [\mathbf{N}_1]_1 \\ [\mathbf{N}_2]_2 \end{pmatrix}$ with witness $(\boldsymbol{a}, \boldsymbol{r}, \delta_v, \delta_w, r_v)^\top \in \mathbb{Z}_p^{2n+3}$. Finally, it outputs the argument $\pi := \left([H]_1, [V]_{1,2}, [W]_1, [\tilde{\boldsymbol{c}}]_2, \psi\right)$.

– The verifier V with input $(\mathtt{crs}, [\boldsymbol{c}]_1, \mathbf{V}, \mathbf{W}, \pi)$ returns 1 iff $[V]_1[V]_2 - [W]_1[1]_2 = [H]_1[t(s)]_2$ and $\mathsf{V}_{\mathsf{bls}}(\mathsf{crs}_{\mathsf{bls}}, \mathtt{x}_{\mathsf{bls}}, \psi) = 1$.

SSB functionality in the security proof. The security proof of the argument uses similar techniques as [15] but simplified because we rely on the properties of SSB commitments. Intuitively, in the security reduction we need to compute some elements of the form $[\sum_i a_i y_i]_2$ where $(a_1, \ldots, a_n)$ is the witness and $[y_1, \ldots, y_n]_2$ are elements that can be computed from the challenge of some falsifiable assumption or public elements. The actual reduction requires us to extract multiple such linear combinations.

If an adversary wins the soundness game, its argument passes verification but at least one SAP equation does not hold. In the security proof, the soundness game is first changed by randomly picking one of the SAP equations $\left(\boldsymbol{a}^\top \boldsymbol{v}_{j^*}\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_{j^*} = 0$ for some $j^* \in [1\,..\,d]$. To complete the proof, we have to check the equation and break a computational assumption. For the former, since our perfectly binding commitment is only $[\cdot]_1$-extractable, we can at best extract $[a_i]_1$ which is not enough to check the j^*-th equation, even if $\boldsymbol{v}_{j^*}$ and $\boldsymbol{w}_{j^*}$ are public. We need a square of $\boldsymbol{a}$, so it suffices to extract $\sum [a_i]_2 v_{j^*,i}$ in $\mathbb{G}_2$ and prove the equation in the target group. For the latter, we break the d-SATSDH assumption (see Appendix F.1) that is a version of the d-TSDH assumption with some extra elements that are linear combinations of the witness.

Next, we switch the EMP commitment key that is in perfectly hiding mode in the honest proof ($\mathcal{S} = \emptyset$) to the mode that encodes the functions $f(a_1, \ldots, a_n) = \sum_i a_i [y_i]_2$ that we need. Then, from $[\tilde{\boldsymbol{c}}]_2$ we can extract $[\sum_i a_i v_{j^*,i}]_2$, and so check the equation in $\mathbb{G}_T$, and also the linear combinations to break the assumption.

The *FSH* property guarantees that the adversary cannot learn the index j^* and thus the j^*-th SAP equation is not satisfied with probability $\geq 1/d$. The $[\cdot]_2$-*SSE* property allows us to extract some linear combinations of the claimed witness and break the d-SATSDH assumption. Zero-knowledge is straightforwardly guaranteed by the *AEPH* property. The full security proof and more intuition of it are deferred to Appendix F.2.

References

1. Abdolmaleki, B., Baghery, K., Lipmaa, H., Zajac, M.: A subversion-resistant SNARK. In: ASIACRYPT 2017, Part III. LNCS, vol. 10626, pp. 3–33
2. Aiello, W., Ishai, Y., Reingold, O.: Priced oblivious transfer: How to sell digital goods. In: EUROCRYPT 2001. LNCS, vol. 2045, pp. 119–135
3. Arora, S., Lund, C., Motwani, R., Sudan, M., Szegedy, M.: Proof verification and hardness of approximation problems. In: 33rd FOCS, pp. 14–23
4. Arora, S., Safra, S.: Probabilistic checking of proofs; A new characterization of NP. In: 33rd FOCS, pp. 2–13
5. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: P-signatures and non-interactive anonymous credentials. In: TCC 2008. LNCS, vol. 4948, pp. 356–374
6. Boneh, D., Boyen, X.: Secure identity based encryption without random oracles. In: CRYPTO 2004. LNCS, vol. 3152, pp. 443–459
7. Bünz, B., Bootle, J., Boneh, D., Poelstra, A., Wuille, P., Maxwell, G.: Bulletproofs: Short proofs for confidential transactions and more. In: 2018 IEEE Symposium on Security and Privacy, pp. 315–334
8. Catalano, D., Fiore, D.: Vector commitments and their applications. In: PKC 2013. LNCS, vol. 7778, pp. 55–72
9. Catalano, D., Visconti, I.: Hybrid trapdoor commitments and their applications. In: ICALP 2005. LNCS, vol. 3580, pp. 298–310
10. Chase, M., Ganesh, C., Mohassel, P.: Efficient zero-knowledge proof of algebraic and non-algebraic statements with applications to privacy preserving credentials. In: CRYPTO 2016, Part III. LNCS, vol. 9816, pp. 499–530
11. Chiesa, A., Forbes, M.A., Spooner, N.: A zero knowledge sumcheck and its applications. Cryptology ePrint Archive, Report 2017/305 (2017) `http://eprint.iacr.org/2017/305`.
12. Damgård, I., Fehr, S., Lunemann, C., Salvail, L., Schaffner, C.: Improving the security of quantum protocols via commit-and-open. In: CRYPTO 2009. LNCS, vol. 5677, pp. 408–427
13. Damgård, I., Nielsen, J.B.: Perfect hiding and perfect binding universally composable commitment schemes with constant expansion factor. In: CRYPTO 2002. LNCS, vol. 2442, pp. 581–596
14. Danezis, G., Fournet, C., Groth, J., Kohlweiss, M.: Square span programs with applications to succinct NIZK arguments. In: ASIACRYPT 2014, Part I. LNCS, vol. 8873, pp. 532–550
15. Daza, V., González, A., Pindado, Z., Ràfols, C., Silva, J.: Shorter quadratic QA-NIZK proofs. In: PKC 2019, Part I. LNCS, vol. 11442, pp. 314–343
16. Döttling, N., Garg, S., Ishai, Y., Malavolta, G., Mour, T., Ostrovsky, R.: Trapdoor hash functions and their applications. In: CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 3–32
17. Döttling, N., Ghosh, S., Nielsen, J.B., Nilges, T., Trifiletti, R.: TinyOLE: Efficient actively secure two-party computation from oblivious linear function evaluation. In: ACM CCS 2017, pp. 2263–2276
18. Döttling, N., Kraschewski, D., Müller-Quade, J.: Statistically secure linear-rate dimension extension for oblivious affine function evaluation. In: ICITS 12. LNCS, vol. 7412, pp. 111–128
19. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.: An algebraic framework for Diffie-Hellman assumptions. In: CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 129–147

20. Fauzi, P., Meiklejohn, S., Mercer, R., Orlandi, C.: Quisquis: A new design for anonymous cryptocurrencies. In: ASIACRYPT 2019, Part I. LNCS, vol. 11921, pp. 649–678
21. Fuchsbauer, G., Orrù, M., Seurin, Y.: Aggregate cash systems: A cryptographic investigation of Mimblewimble. In: EUROCRYPT 2019, Part I. LNCS, vol. 11476, pp. 657–689
22. Gennaro, R., Gentry, C., Parno, B., Raykova, M.: Quadratic span programs and succinct NIZKs without PCPs. In: EUROCRYPT 2013. LNCS, vol. 7881, pp. 626–645
23. Gentry, C., Ramzan, Z.: Single-database private information retrieval with constant communication rate. In: ICALP 2005. LNCS, vol. 3580, pp. 803–815
24. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: 43rd ACM STOC, pp. 99–108
25. Ghosh, S., Nielsen, J.B., Nilges, T.: Maliciously secure oblivious linear function evaluation with constant overhead. In: ASIACRYPT 2017, Part I. LNCS, vol. 10624, pp. 629–659
26. González, A., Hevia, A., Ràfols, C.: QA-NIZK arguments in asymmetric groups: New tools and new constructions. In: ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 605–629
27. González, A., Ràfols, C.: New techniques for non-interactive shuffle and range arguments. In: ACNS 16. LNCS, vol. 9696, pp. 427–444
28. González, A., Ràfols, C.: Shorter pairing-based arguments under standard assumptions. In: ASIACRYPT 2019, Part III. LNCS, vol. 11923, pp. 728–757
29. Groth, J.: Short pairing-based non-interactive zero-knowledge arguments. In: ASIACRYPT 2010. LNCS, vol. 6477, pp. 321–340
30. Groth, J., Maller, M.: Snarky signatures: Minimal signatures of knowledge from simulation-extractable SNARKs. In: CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 581–612
31. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432
32. Hubacek, P., Wichs, D.: On the communication complexity of secure function evaluation with long output. In: ITCS 2015, pp. 163–172
33. Jutla, C.S., Roy, A.: Shorter quasi-adaptive NIZK proofs for linear subspaces. In: ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 1–20
34. Kilian, J.: On the complexity of bounded-interaction and noninteractive zero-knowledge proofs. In: 35th FOCS, pp. 466–477
35. Libert, B., Yung, M.: Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In: TCC 2010. LNCS, vol. 5978, pp. 499–517
36. Lipmaa, H.: An oblivious transfer protocol with log-squared communication. In: ISC 2005. LNCS, vol. 3650, pp. 314–328
37. Lipmaa, H.: Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In: TCC 2012. LNCS, vol. 7194, pp. 169–189
38. Naor, M., Pinkas, B.: Efficient oblivious transfer protocols. In: 12th SODA, pp. 448–457
39. Okamoto, T., Pietrzak, K., Waters, B., Wichs, D.: New realizations of somewhere statistically binding hashing and positional accumulators. In: ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 121–145
40. Pedersen, T.P.: Non-interactive and information-theoretic secure verifiable secret sharing. In: CRYPTO'91. LNCS, vol. 576, pp. 129–140

41. Poelstra, A.: Mimblewimble (2016) Available at `https://download.wpsoftware.net/bitcoin/wizardry/mimblewimble.pdf`.
42. Ràfols, C., Silva, J.: QA-NIZK arguments of same opening for bilateral commitments. In: AFRICACRYPT 20. LNCS, vol. 12174, pp. 3–23
43. Villar, J.L.: Optimal reductions of some decisional problems to the rank problem. In: ASIACRYPT 2012. LNCS, vol. 7658, pp. 80–97

A Full QA-NIZK Definitions

A quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proof [33] enables one to prove membership in a language defined by a relation $\mathcal{R}_\rho$, which is determined by some parameter ρ sampled from a distribution $\mathcal{D}_{\mathsf{gk}}$. A distribution $\mathcal{D}_{\mathsf{gk}}$ is *witness-sampleable* if there exists an efficient algorithm that samples (ρ, ω_ρ) from a distribution $\mathcal{D}^{\mathsf{par}}_{\mathsf{gk}}$ such that ρ is distributed according to $\mathcal{D}_{\mathsf{gk}}$, and membership of ρ in the *parameter language* $\mathcal{L}_{\mathsf{par}}$ can be efficiently verified by using this witness ω_ρ.

A tuple of algorithms $(\mathsf{K}_0, \mathsf{K}_1, \mathsf{P}, \mathsf{V})$ is called a *QA-NIZK proof system* for witness-relations $\mathcal{R}_{\mathsf{gk}} = \{\mathcal{R}_\rho\}_{\rho \in \sup(\mathcal{D}_{\mathsf{gk}})}$ with parameters sampled from a distribution $\mathcal{D}_{\mathsf{gk}}$ 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:

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

Computational Quasi-Adaptive Soundness:

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

Computational Strong Quasi-Adaptive Soundness:

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

Perfect Quasi-Adaptive Zero-Knowledge:

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

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

We assume that $\mathtt{crs}$ contains an encoding of ρ, which is thus available to V.

B Relation to Existing Primitives

B.1 Relation to SSB Hashes

The SSB requirement makes the EMP commitment scheme look similar to SSB hash functions [32, 39], in which one can compute a hash of a vector v such that the computed hash is statistically binding in one coordinate of v. However, there are also obvious differences. First, to obtain zero-knowledge, we need hiding (AESH) that is not required from hash functions. This is, intuitively, a natural distinction and corresponds to the difference between collision-resistant hash families and statistically hiding commitment schemes.

Second, [32, 39] require that an SSB hash has the local opening property, meaning that the committer can efficiently open just one coordinate of the committed vector. In the QA-NIZK application, we do not need this property: the commitment key $\mathtt{ck}$ is created by a trusted third party, and there is no need for the honest parties to ever open the commitment. Instead, in the soundness proof, we need *somewhere statistical extractability* (SSE), stating that the creator of $\mathtt{ck}$ (e.g., the adversary $\mathcal{B}$) must be able to extract the succinct guilt witness. SSE is not needed in the case of SSB hashes. Although not needed in our concrete applications, it is also desirable to have the *almost everywhere statistical trapdoor* (AEST) property, where the creator of $\mathtt{ck}$ is able to replace non-SB coordinates with anything she wishes. Finally, we allow $\mathtt{ck}$ to be long, but require commitments to be succinct.

The properties of SSB and local opening are orthogonal: it is possible to construct efficient SSB hashes without local opening [39] and efficient vector commitments [8, 35] (which have a local opening) without the SSB property.

B.2 Relation to Oblivious Transfer (OT)

SSB commitments are directly related to two-message OT protocols as defined in [2]. In an OT protocol, the sender has an n-element database and the chooser has an index-set $\mathcal{S}$ with $|\mathcal{S}| \leq q$. The chooser wants to obtain $\boldsymbol{x}_{\mathcal{S}}$; no additional information should be leaked either to the chooser or the sender. In a two-message OT protocol (in the plain model), the chooser sends the first message otq (an encoding of $\mathcal{S}$) to the sender who replies with the second message otr (an encoding of $\boldsymbol{x}_{\mathcal{S}}$). OT protocols have very wide applications in many areas of cryptography, with two-message OT protocols in the plain model such as [2, 23, 36, 38] being of special interest because of their efficiency.

Essentially, SSB commitments are non-interactive analogs of such protocols, the commitment key corresponding to the first OT message ot_1, and the commitment corresponding to the second OT message ot_2. However, the connection is not completely one-to-one, since there are subtle differences in the security definitions between SSB commitment schemes and OT protocols. Importantly, while in OT, the ot_1 generator is always untrusted, in our applications it is sufficient to consider a trusted $\mathtt{ck}$ generator, which allows for more efficient constructions. Additionally, SSB commitment schemes (such as EMP) result in a flavour of OT

where the receiver's message ot_1 is long but can be reused multiple times, while the sender's message ot_2 is much shorter.

Thus, all secure two-message OT protocols are also secure SSB commitment schemes. Unfortunately, none of the known efficient two-message OT protocols have the required algebraic structure to construct QA-NIZKs, and thus they are unsuitable for our main application.

B.3 Relation to PCP-Based SNARKs

The QA-NIZK application of SSB commitments is based on the observation that the language of bit-strings (resp., CircuitSAT) has a local verifiability property, similar to PCP [3, 4]: one can establish, by checking one random coordinate of the bit-string (resp., all adjacent wires of a random gate), whether an input belongs to the language or not. Typical PCP-based zero-knowledge arguments like [34] use PCPs with small soundness error; as a drawback, such PCPs have a long proof and an inefficient reduction from CircuitSAT. Daza *et al.* [15] and the current paper use a trivial PCP with a large soundness error but with a trivial reduction from CircuitSAT. The use of SSB commitments means that the efficiency loss is logarithmic in n (one needs to use $\approx 2\log n$-bit longer group elements) while in the case of earlier PCP-based arguments the efficiency loss is much larger. Nevertheless, the use of SSB commitments is not limited to trivial PCP; one can use them together with any PCP that has a small number of queries and short proof length.

C Missing Proofs in Section 3

C.1 Proof of Lemma 1

Proof. Assume that for given n and q, $\mathcal{A}$ breaks SSB with probability $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)$. This means that for some $\mathcal{S}$ of cardinality $\le q$ and honestly generated ck (w.r.t. $\mathcal{S}$), $\mathcal{A}$ outputs $(\boldsymbol{x}_0, \boldsymbol{x}_1, r_0, r_1)$ such that $\boldsymbol{x}_{0\mathcal{S}} \neq \boldsymbol{x}_{1\mathcal{S}}$ and $C := \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_0; r_0) = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_1; r_1)$.

Since $\boldsymbol{x}_{0\mathcal{S}} \neq \boldsymbol{x}_{1\mathcal{S}}$ and F is injective, we get that $\boldsymbol{F}_0 := (F(x_{0\sigma_1}), \ldots, F(x_{0\sigma_{|\mathcal{S}|}})) \neq (F(x_{1\sigma_1}), \ldots, F(x_{1\sigma_{|\mathcal{S}|}})) =: \boldsymbol{F}_1$. Therefore, there exists $\beta \in \{0,1\}$, such that $\mathsf{Ext}_F(\mathsf{p}, \mathsf{ek}; C) \neq \boldsymbol{F}_\beta$. Thus, if $\mathcal{B}$ outputs $(\boldsymbol{x}_\beta, r_\beta)$ for $\beta \leftarrow\!\!\$\, \{0,1\}$, $\mathsf{Adv}^{\mathrm{sse}}_{\beta,F,\mathsf{COM},n,q}(\lambda) \ge \mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)/2$ and hence $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \le 2 \cdot \mathsf{Adv}^{\mathrm{sse}}_{\beta,F,\mathsf{COM},n,q}(\lambda)$. □

C.2 Proof of Theorem 1

Proof. Let $\Pr[\mathrm{Game}_i(\mathcal{A}) = 1]$ denote the probability $\mathcal{A}$ wins in Game_i.

(i: ISH + SSB ⇒ CB) We prove the theorem using a sequence of hybrid games, defined as follows, where $\varepsilon_i := \Pr[\mathrm{Game}_i(\mathcal{A}) = 1]$.

$\underline{\mathrm{Game}_1}$: The original computational binding game. For given n and q, by definition $\mathcal{A}$ can break CB with probability $\varepsilon_1 = \mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)$.

$\underline{\text{Game}_2}$: Game_1, but instead of $\mathtt{ck}$, $\mathcal{A}$ gets $\mathtt{ck}'$ where $(\mathtt{ck}', \mathtt{td}') \leftarrow \overline{\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_1)}$ for $\mathcal{S}_1 \leftarrow\!\!\$\ \mathbb{P}([1\,..\,n], q)$. Note that a distinguisher $\mathcal{B}_1$ for Game_1 and Game_2 can be used to break the ISH game with advantage $\varepsilon_{\mathsf{ish}} = \mathsf{Adv}^{\text{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda)$. Hence $|\varepsilon_1 - \varepsilon_2| \leq \varepsilon_{\mathsf{ish}}$, which implies that $\varepsilon_2 \geq \varepsilon_1 - \varepsilon_{\mathsf{ish}}$.

We now require the following lemma.

Lemma 3. *Assume $\mathcal{A}$ outputs $(\boldsymbol{x}_0, r_0, \boldsymbol{x}_1, r_1)$ with $\boldsymbol{x}_0 \neq \boldsymbol{x}_1$. Then $\Pr[(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1}$ in $\text{Game}_2] \geq q/n - 4 \cdot \varepsilon_{\mathsf{ish}}$.*

Proof. Assume for any $\mathcal{S}_1$ of size q sampled uniformly at random, $\mathcal{A}$ can output distinct $\boldsymbol{x}_0, \boldsymbol{x}_1$ such that $\Pr[(x_0)_{\mathcal{S}_1} \neq (x_1)_{\mathcal{S}_1}$ in $\text{Game}_2] = \varepsilon$.

We construct an adversary $\mathcal{B}$ that uses $\mathcal{A}$ to break ISH as follows.

1. Given p, n, q, $\mathcal{B}$ sets $\mathcal{S}_1 \leftarrow\!\!\$\ \mathbb{P}([1\,..\,n], q)$ and receives $S_0 \leftarrow \mathcal{A}(\mathsf{p}, n, q)$.
2. $\mathcal{B}$ sends $(\mathcal{S}_0, \mathcal{S}_1)$ to the ISH challenger, and receives $\mathtt{ck}$ corresponding to $\mathcal{S}_\beta$.
3. $\mathcal{B}$ gets $(\boldsymbol{x}_0, r_0, \boldsymbol{x}_1, r_1) \leftarrow \mathcal{A}(\mathtt{ck})$.
   - If $\mathcal{A}$ does not win, abort.
   - If $(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1}$ return $\beta' \leftarrow\!\!\$\ \{0,1\}$.
   - Else return 1.

Note that $\beta = 0$ corresponds to Game_1, and $\beta = 1$ corresponds to Game_2. Moreover, for $\beta = 0$, $\mathcal{A}$'s output $(\boldsymbol{x}_0, r_0, \boldsymbol{x}_1, r_1)$ is independent of S_1, in which case $\Pr[(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1}] \geq |\mathcal{S}_1|/n = q/n$. Hence we get that if $\mathcal{A}$ wins,

$$\begin{aligned}
\Pr[\text{Game}_{ISH}(\mathcal{B}) = 1] &= \frac{1}{2}\Pr[\text{Game}_{ISH}(\mathcal{B}) = 1 | \beta = 0] + \frac{1}{2}\Pr[\text{Game}_{ISH}(\mathcal{B}) = 1 | \beta = 1] \\
&= \frac{1}{2}\Pr[(x_0)_{\mathcal{S}_1} \neq (x_1)_{\mathcal{S}_1} \text{ in } \text{Game}_1 \wedge \beta' = 0] \\
&+ \frac{1}{2}\Pr[(x_0)_{\mathcal{S}_1} = (x_1)_{\mathcal{S}_1} \text{ in } \text{Game}_2] \\
&+ \frac{1}{2}\Pr[(x_0)_{\mathcal{S}_1} \neq (x_1)_{\mathcal{S}_1} \text{ in } \text{Game}_2 \wedge \beta' = 1] \\
&\geq \frac{q}{4n} + \frac{1-\epsilon}{2} + \frac{\epsilon}{4} \\
&= \frac{1}{2} + \frac{q - n\epsilon}{4n} \ .
\end{aligned}$$

Hence $4 \cdot \varepsilon_{\mathsf{ish}} \geq q/n - \epsilon$, as required. □

It is easy to see that an adversary that wins Game_2 with $(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1}$ also wins the SSB game. Hence there exists an adversary $\mathcal{B}_2$ such that

$$\begin{aligned}
\mathsf{Adv}^{\text{ssb}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda) &\geq \varepsilon_2 \cdot \Pr[(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1} \text{ in } \text{Game}_2 | \boldsymbol{x}_0 \neq \boldsymbol{x}_1] \\
&\geq (\varepsilon_1 - \varepsilon_{\mathsf{ish}})(q/n - 4 \cdot \varepsilon_{\mathsf{ish}}) \text{ (due to Lemma 3).}
\end{aligned}$$

This is equivalent to $\varepsilon_1 \leq \varepsilon_{\mathsf{ish}} + \frac{n}{q - 4 \cdot n \cdot \varepsilon_{\mathsf{ish}}} \cdot \mathsf{Adv}^{\text{ssb}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda)$.

(ii: ISH + AESH ⇒ CH) Assume that for given n and q, $\mathcal{A}$ can break CH with probability $\mathsf{Adv}^{\text{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)$. Consider the following sequence of games with $\varepsilon_i := \Pr[\text{Game}_i(\mathcal{A}) = 1]$.

Game$_1$: In this game, $\mathcal{A}$ breaks CH with probability ε_1. That is, given p, $\mathcal{A}(\mathsf{p}, n, q)$ outputs $\mathcal{S}_0$ such that $|\mathcal{S}_0| \leq q$, and for $(\mathtt{ck}_0, \mathtt{td}_0) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_0)$, $\mathcal{A}(\mathtt{ck}_0)$ outputs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$, s.t. $\Pr[\beta \leftarrow\!\!\$\, \{0,1\} : \mathcal{A}(\mathsf{Com}(\mathtt{ck}_0; \boldsymbol{x}_\beta; r)) = \beta] = \varepsilon_1$.

Game$_2$: In this game, instead of $\mathtt{ck}_0$, $\mathcal{A}$ obtains $\mathtt{ck}_1$ where $(\mathtt{ck}_1, \mathtt{td}_1) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_1)$ for $\mathcal{S}_1 = \emptyset$. Clearly, for any PPT $\mathcal{A}$ that tries to distinguish Game$_1$ and Game$_2$, there exists a PPT $\mathcal{B}_1$, such that $|\varepsilon_2 - \varepsilon_1| \leq \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda)$.

Let us consider the following AESH adversary $\mathcal{B}_2$ in Game$_2$.

1. Given p, n, q, $\mathcal{B}_2$ sets $\mathcal{S}_1 \leftarrow \emptyset$ and receives $S_0 \leftarrow \mathcal{A}(\mathsf{p}, n, q)$.
2. $\mathcal{B}_2$ computes $(\mathtt{ck}_1, \mathtt{td}_1) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_1)$ and receives $(\boldsymbol{x}_0, \boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathtt{ck})$.
3. $\mathcal{B}_2$ forwards $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ to the AESH challenger, and receives $c \leftarrow \mathsf{Com}(\mathtt{ck}_1, \boldsymbol{x}_\beta; r)$ for some $\beta \leftarrow\!\!\$\, \{0,1\}$, $r \leftarrow\!\!\$\, \mathsf{RSP}$.
4. $\mathcal{B}$ gets and outputs $\beta' \leftarrow \mathcal{A}(c)$.

If $\mathcal{A}$ returns the correct β' then clearly also $\mathcal{B}_2$ returns the correct β'. For the success of $\mathcal{B}_2$, it is also needed that $\boldsymbol{x}_{0\mathcal{S}_1} = \boldsymbol{x}_{1\mathcal{S}_1}$, which clearly holds since $\mathcal{S}_1 = \emptyset$. Thus, $\mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda) = \varepsilon_2$. Hence, $\mathsf{Adv}^{\mathrm{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq |\varepsilon_2 - \varepsilon_1| + \varepsilon_2 \leq \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda) + \mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda)$. □

D Details of Algebraic Commitments Schemes (ACS)

D.1 Characterisation of ACS

ACS as SSB commitment schemes. We will show that ACS defined in Section 4 are computationally hiding under MDDH. They are also perfectly binding in those components that correspond to the linearly independent columns of $\boldsymbol{U}_1$. If they are also pair-wise to columns of $\boldsymbol{U}_2$, the system of equations has maximum rank and unique solution. We give this characterisation in Lemma 4.

Moreover, for extraction assume that $\mathrm{span}\{\boldsymbol{U}_1\} \cap \mathrm{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$. Intuitively, $\boldsymbol{U}_1$ defines the space of the opening $\boldsymbol{x}$, while $\boldsymbol{U}_2$ defines the randomness space. To extract in q positions, we hence need $\mathtt{ek}$ is such that $\mathtt{ek}[\boldsymbol{U}_2]_\iota = \mathbf{0}$ and $\mathtt{ek}[\boldsymbol{U}_1]_\iota = (\boldsymbol{b}_i)_{i=1}^n$, where $\boldsymbol{b}_i$ is $\boldsymbol{e}_i$ in q positions and $\mathbf{0}$ elsewhere. Then by the linearity of ACS, $\mathtt{ek} \cdot \mathsf{Com}_{\mathtt{ck}}(\boldsymbol{x}, \boldsymbol{r}) = \mathtt{ek} \cdot [\boldsymbol{U}_1]_\iota \boldsymbol{x} = [\boldsymbol{x}]_\iota$.

Lemma 4. *Let $n \geq 1$ and $q \leq n$. Let COM be an ACS with commitment key $\mathtt{ck} = [\boldsymbol{U}_1, \boldsymbol{U}_2]_\iota$ sampled from $\mathcal{D}_1 \times \mathcal{D}_2$ as defined in Definition 1.*

(i) COM *is AECH under $\mathcal{D}_2$-MDDH$_{\mathbb{G}_\iota}$.*
(ii) COM *is ISH under $\mathcal{D}_1, \mathcal{D}_2$-MDDH$_{\mathbb{G}_\iota}$.*
(iii) COM *is SPB if $\boldsymbol{U}_1$ has rank q and* $\mathrm{span}\{\boldsymbol{U}_1\} \cap \mathrm{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$.
(iv) COM *is $[\cdot]_\iota$-SPE if $\boldsymbol{U}_1$ has rank q and* $\mathrm{span}\{\boldsymbol{U}_1\} \cap \mathrm{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$.

Proof. Let $\mathcal{S} \subseteq [1\,..\,n]$, $|\mathcal{S}| \leq q$ be the indices of $\boldsymbol{x}$ one can extract during opening.

(i: AECH) Let $\mathcal{A}$ be an adversary that breaks AECH with non-negligible probability, say $\varepsilon_{\mathcal{A}}$. Consider the following $\mathbb{G}_\iota$-MDDH adversary $\mathcal{B}$. $\mathcal{B}$ receives a challenge $[\mathbf{A}, \boldsymbol{y}_\beta]_\iota$ where $\mathbf{A} \leftarrow\!\!\$\, \mathcal{D}_2$, $\boldsymbol{y}_0 \leftarrow\!\!\$\, \mathbb{Z}_p^k$, and $\boldsymbol{y}_1 \leftarrow \mathbf{A}\boldsymbol{r}$ for $\boldsymbol{r} \leftarrow\!\!\$\, \mathbb{Z}_p^m$. $\mathcal{B}$ sets $[\boldsymbol{U}_2]_\iota \leftarrow [\mathbf{A}]_\iota$, and generates $\boldsymbol{U}_1$ from the distribution $\mathcal{D}_1$. $\mathcal{B}$ sends $\mathtt{ck} = [\boldsymbol{U}_1, \boldsymbol{U}_2]_\iota$ to $\mathcal{A}$ who replies with two messages $\boldsymbol{x}_0, \boldsymbol{x}_1$, such that $\boldsymbol{x}_{0,\mathcal{S}}, \boldsymbol{x}_{1,\mathcal{S}}$. $\mathcal{B}$

computes $\boldsymbol{c}_0 \leftarrow [\boldsymbol{U}_1]_\iota \boldsymbol{x}_0 + [\boldsymbol{U}_2]_\iota \boldsymbol{r}$, for $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^m$, and $\boldsymbol{c}_1 \leftarrow [\boldsymbol{U}_1]_\iota \boldsymbol{x}_1 + [\boldsymbol{y}_\beta]_\iota$. $\mathcal{B}$ picks $\beta' \leftarrow \{0,1\}$ and sends $c_{\beta'}$ to $\mathcal{A}$. $\mathcal{A}$ guesses which message was committed by returning $\beta_\mathcal{A} \in \{0,1\}$ to $\mathcal{B}$. $\mathcal{B}$ sends $\beta_\mathcal{A}$ to the MDDH challenger. Clearly,

$$\begin{aligned}\Pr[\beta_\mathcal{A} = \beta] =& \Pr[\beta_\mathcal{A} = 0 | \beta = 0]/2 + \Pr[\beta_\mathcal{A} = 1 | \beta = 1]/2 \\ =& \varepsilon_\mathcal{A}/2 + (\Pr[\beta_\mathcal{A} = 1 | \beta = 1, \beta' = 0]/2 + \Pr[\beta_\mathcal{A} = 1 | \beta = 1, \beta' = 1]/2)/2 \\ =& \varepsilon_\mathcal{A}/2 + \varepsilon_\mathcal{A}/4 + \varepsilon_\mathcal{A}/8 = 7/8 \cdot \varepsilon_\mathcal{A} \ .\end{aligned}$$

Thus if $\mathcal{A}$ succeeded with non-negligible probability, then so did $\mathcal{B}$.

(ii: ISH) Firstly we prove that for any $\mathcal{S}_0$ with $|\mathcal{S}_0| \leq n$, if $\mathcal{S}_1 = \mathcal{S}_0 \cup \{i^*\}$ for some $i^* \notin \mathcal{S}_0$ and $\mathcal{S}_0, \mathcal{S}_1 \subseteq [1\,..\,n]$, then $\mathcal{D}_{1,2}^{0,q} := ([\mathcal{D}_{\mathcal{S}_0}^{n,k}]_\iota, [\mathcal{D}_{\mathcal{S}_0}^{m,k}]_\iota)$ and $\mathcal{D}_{1,2}^{1,q} := ([\mathcal{D}_{\mathcal{S}_1}^{n,k}]_\iota, [\mathcal{D}_{\mathcal{S}_1}^{m,k}]_\iota)$ are computationally indistinguishable under MDDH. Let $\mathcal{A}$ be an adversary that can distinguish $\mathcal{D}_{1,2}^0$ and $\mathcal{D}_{1,2}^1$. We construct the following MDDH adversary $\mathcal{B}$ that receives a challenge $[\mathbf{A}, \boldsymbol{y}_\beta]_\iota$ where $\mathbf{A}_1, \mathbf{A}_2 \leftarrow_\$ \mathcal{D}_{1,2}^0$, $\boldsymbol{y}_0 \leftarrow_\$ \mathbb{Z}_p^k$, and $\boldsymbol{y}_1 \leftarrow (\mathbf{A}_1^\top | \mathbf{A}_2^\top)\boldsymbol{r}$ for $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^m$. $\mathcal{B}$ sets $[\boldsymbol{U}_1]_\iota \leftarrow [\mathbf{A}_1]_\iota$, and $[\boldsymbol{U}_2]_\iota \leftarrow ([\mathbf{A}_2]_\iota | [\boldsymbol{y}_\beta]_\iota)$. $\mathcal{B}$ computes $\boldsymbol{c}_\beta \leftarrow [\boldsymbol{U}_1]_\iota \boldsymbol{x} + [\boldsymbol{U}_2]_\iota \boldsymbol{r}$, for $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^m$ and sends $\boldsymbol{c}_\beta$ to $\mathcal{A}$ who replies with $\beta_\mathcal{A}$. Thus, $\mathcal{B}$ has the same advantage in breaking MDDH as $\mathcal{A}$ has in distinguishing $\mathcal{D}_{1,2}^{0,q}$ and $\mathcal{D}_{1,2}^{1,q}$.

Now, for any sets $\mathcal{S}_0$ and $\mathcal{S}_1$ it holds that $\mathsf{Adv}^{\mathrm{indist}}_{\mathcal{A},\mathcal{D}_{1,2}^0,\mathcal{D}_{1,2}^1}(\lambda) \leq (|\mathcal{S}_0 \cup \mathcal{S}_1| - |\mathcal{S}_0 \cap \mathcal{S}_1|) \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{B},\mathcal{D}_{1,2}^{n,q},\mathsf{Pgen}}(\lambda)$.

(iii: SPB) Assume that all columns of $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are pairwise linearly independent. Consider the matrix system of equations defined by $(\boldsymbol{U}_1, \boldsymbol{U}_2)\binom{\boldsymbol{x}}{\boldsymbol{r}} = \mathsf{Com}_{\mathsf{ck}}(\boldsymbol{x}, \boldsymbol{r})$. This system has a unique solution because the matrix has full rank. Hence, each commitment corresponds to a unique vector $\binom{\boldsymbol{x}}{\boldsymbol{r}}$. Now, if $\boldsymbol{U}_1$ has q columns pair-wise linear independent and columns of $\boldsymbol{U}_2$ pair-wise linear independent to all of them, consider the system that has a matrix with those q columns of $\boldsymbol{U}_1$ and the whole $\boldsymbol{U}_2$. Its rank is maximum as well and the result follows.

(iv: $[\cdot]$-SPE) Since $k > m$, for any matrix $\boldsymbol{U}_2$ of size $k \times m$ there exist matrices $\mathsf{ek} \in \boldsymbol{U}_2^\perp$ that define orthogonal spaces of $\boldsymbol{U}_2$ of size $k' \times k$ for $k' \geq k - m$ such that $\mathsf{ek} \cdot \boldsymbol{U}_2 = \begin{pmatrix} \mathbf{0}_{(k-m)\times m} \\ \boldsymbol{a} \end{pmatrix}$ where $\boldsymbol{a} \in \mathbb{Z}_p^{(k'-k+m)\times m}$. This space has at least dimension 1 because $k > m$. Moreover, there exists an appropriate change of basis of the space such that $\mathsf{ek} \cdot \boldsymbol{U}_1 = \left(\begin{array}{c|c} \boldsymbol{I}_q & \\ \boldsymbol{b}_1 & \boldsymbol{b}_2 \end{array}\right)$ where $\boldsymbol{b}_1 \in \mathbb{Z}_p^{(k'-q)\times q}, \boldsymbol{b}_2 \in \mathbb{Z}_p^{k' \times (n-q)}$. This is well-defined since $k - m \geq q$ and if q columns of the matrices are pair-wise linear independent then $k' - q \geq k - m - q \geq 0$. □

Corollary 1. *The minimum size of the $k \times m$ matrix to guarantee $[\cdot]_\iota$-extraction of $n \geq 1$ elements is $k = n + 1$, $m = 1$.*

Proof. Information theoretically the commitment size should be no less than the dimension of the opening in order to extract it completely, so $k \geq n$. The orthogonal space has to be at least of dimension 1 in order to provide extraction,

so the minimal difference is $k-m \geq 1$. We have $k \geq n+m$ directly by the linear independence of the columns in matrices $\boldsymbol{U}_1, \boldsymbol{U}_2$. Hence, the minimal constants are $m=1$, $k=n+1$. □

ACS and QA-NIZK arguments. Algebraic commitments are suitable to work with QA-NIZK arguments for linear spaces because most of their properties can be expressed in terms of membership or non-membership to certain linear subspaces. For example, the works of González *et al.* [15, 26, 27] implicitly use an SSB commitment scheme COM to construct efficient QA-NIZK argument systems based on falsifiable assumptions. The soundness of their QA-NIZK system depends on the ISH, SSB, and SSE properties, while the zero-knowledge property depends on the AESH and CH properties. On the other hand, honest parties never need to actually open the commitment; the opening (more precisely, extraction) is only done inside the security proof by using the SSE property[5]. Moreover, in our QA-NIZK argument in Section 6.2, as well as [15], we use functional SSB commitments since linear EMP is more straightforward to our use of it in the soundness proof.

D.2 Proof of Lemma 2

Proof. Fix λ. We first prove that for any $\mathcal{S}_0$ with $|\mathcal{S}_0| \leq q-1$, if $\mathcal{S}_1 = \mathcal{S}_0 \cup \{i^*\}$ for $i^* > \max_i\{i \in \mathcal{S}_0\}$ and $\mathcal{S}_0, \mathcal{S}_1 \subseteq [1\,..\,n]$, then $\mathcal{D}_0 := [\mathcal{D}_{q+1}^{p,n,\mathcal{S}_0}]$ and $\mathcal{D}_1 := [\mathcal{D}_{q+1}^{p,n,\mathcal{S}_1}]$ are computationally indistinguishable.

Let $\mathcal{A}$ be an adversary that can distinguish $\mathcal{D}_0$ and $\mathcal{D}_1$. We construct the following MDDH adversary $\mathcal{B}$. The challenger $\mathcal{C}$ of the MDDH game samples $\mathbf{A} \leftarrow_{\$} \mathbb{Z}_p^{q+1}$ and $\mathtt{w} \leftarrow_{\$} \mathbb{Z}_p$. If $\beta = 0$ then $\mathcal{C}$ samples $\boldsymbol{y} \leftarrow_{\$} \mathbb{Z}_p^{q+1}$, otherwise $\mathcal{C}$ sets $\boldsymbol{y} \leftarrow \mathbf{A}\mathtt{w}$. $\mathcal{C}$ sends $(\mathsf{p}, [\mathbf{A}, \boldsymbol{y}]_\iota)$ to $\mathcal{B}$. $\mathcal{B}$ does the following:

$\mathcal{B}(\mathsf{p}, [\mathbf{A}, \boldsymbol{y}])$

$[\boldsymbol{g}^{(n+1)}] \leftarrow [\mathbf{A}]$;
for i **in** $[1\,..\,n]$ **do**
  if $i = i^*$ **then** $[\boldsymbol{g}^{(i)}] \leftarrow [\boldsymbol{y}]$;
  elseif $i \in \mathcal{S}_0$ **then** $\boldsymbol{g}^{(i)} \leftarrow_{\$} \mathbb{Z}_p^{q+1}$;
  else $\delta_i \leftarrow_{\$} \mathbb{Z}_p$; $[\boldsymbol{g}^{(i)}] \leftarrow [\boldsymbol{g}^{(n+1)}]\delta_i$; **fi endfor**
return $\beta \leftarrow \mathcal{A}(\mathsf{p}, [\boldsymbol{g}])$;

Clearly, $[\boldsymbol{g}]$ is distributed according to $\mathcal{D}_\beta$. Thus, $\mathcal{B}$ has the same advantage in breaking MDDH as $\mathcal{A}$ has in distinguishing $\mathcal{D}_0$ from $\mathcal{D}_1$. By using a standard hybrid argument, $\mathsf{Adv}^{\mathrm{indist}}_{\mathcal{A},\mathcal{D}^0,\mathcal{D}^1}(\lambda) \leq |\mathcal{S}| \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{B},\mathcal{U}_{q+1},\mathsf{Pgen}}(\lambda)$. □

As a simple generalization of Lemma 2, for any $\mathcal{S}_0, \mathcal{S}_1 \subseteq [1\,..\,n]$ with $\mathcal{S}_i \leq q$, $\mathsf{Adv}^{\mathrm{indist}}_{\mathcal{A},[\mathcal{D}_{q+1}^{p,n,\mathcal{S}_0}],[\mathcal{D}_{q+1}^{p,n,\mathcal{S}_1}]}(\lambda) \leq |\mathcal{S}_1 \triangle \mathcal{S}_2| \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{B},\mathcal{U}_{q+1},\mathsf{Pgen}}(\lambda)$.

[5] In this sense, one could also call them trapdoor hash functions [16] with the SSB and AESH properties

D.3 Proof of Theorem 2

Proof. **(i: ISH)** Due to the properties of $\mathcal{D}_{q+1}^{p,n,\mathcal{S}}$, $\boldsymbol{g}^{(\mathcal{S}\cup\{n+1\})}$ has columns distributed uniformly over $\mathbb{Z}_p^{q+1}$ and hence by the Schwartz-Zippel lemma has full rank with probability $\geq 1-(q+1)/p$. It follows from Lemma 2 that for any PPT $\mathcal{A}$, there exists a PPT $\mathcal{B}$, such that $\mathsf{Adv}^{\mathrm{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq q \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{B},\mathcal{U}_{(q+1)\times(n+1)},\iota,\mathsf{Pgen}}(\lambda) + (q+1)/p$.

(ii: $[\cdot]$-SSE) We have $[\mathsf{c}] = [\boldsymbol{g}]\binom{\boldsymbol{x}}{r} = [\boldsymbol{RM}]\binom{\boldsymbol{x}}{r}$ for *some* $\binom{\boldsymbol{x}}{r}$, where $\boldsymbol{R}$ has full rank. But then $[\boldsymbol{x}'] = \boldsymbol{R}^{-1}[\mathsf{c}] = [\boldsymbol{M}]\binom{\boldsymbol{x}}{r}$. Let $\mathcal{S} = \{\sigma_i\}$. By the definition of $\boldsymbol{M}$, clearly $x'_i = \boldsymbol{M}_i\binom{\boldsymbol{x}}{r} = x_{\sigma_i}$ for $i \leq |\mathcal{S}|$.

(iii: AEPT) Let $\boldsymbol{x} \neq \boldsymbol{x}^*$ but $\boldsymbol{x}_{\mathcal{S}} = \boldsymbol{x}^*_{\mathcal{S}}$. Then $\mathsf{Com}(\mathsf{ck};\boldsymbol{x};r) - \mathsf{Com}(\mathsf{ck};\boldsymbol{x}^*;r^*) = \boldsymbol{RM}\binom{\boldsymbol{x}-\boldsymbol{x}^*}{r-r^*} = \boldsymbol{R}\binom{\mathbf{0}_q}{\sum_{i\in[1\,..\,n]\setminus\mathcal{S}}(x_i-x_i^*)\delta_i+(r-r^*)} = \mathbf{0}_{q+1}$, since from tdOpen, $r^* = \sum_{i\in[1\,..\,n]\setminus\mathcal{S}}(x_i - x_i^*)\delta_i + r$.

(iv: SPB) Since $F = [\cdot]$ is injective (because the bilinear group has a prime order), this follows from Item ii and Lemma 1.

(v: AEPH) Let $\boldsymbol{x}, \boldsymbol{x}^*$ be such that $\boldsymbol{x}_{\mathcal{S}} = \boldsymbol{x}^*_{\mathcal{S}}$. Then $\boldsymbol{M}\binom{\boldsymbol{x}}{r} = (\boldsymbol{x}_{\mathcal{S}}^\top, 0, \ldots, 0, r + \sum_{i\in[1\,..\,n]\setminus\mathcal{S}} x_i\sigma_i)^\top$ and similarly $\boldsymbol{M}\binom{\boldsymbol{x}^*}{r^*} = ((\boldsymbol{x}^*_{\mathcal{S}})^\top, 0, \ldots, 0, r^* + \sum_{i\in[1\,..\,n]\setminus\mathcal{S}} x_i^*\sigma_i)^\top$. Thus, both have first q elements equal and the last element is uniformly random. Clearly then also $\mathsf{Com}(\mathsf{ck};\boldsymbol{x};r) = \boldsymbol{RM}\binom{\boldsymbol{x}}{r}$ and $\mathsf{Com}(\mathsf{ck};\boldsymbol{x}^*;r^*) = \boldsymbol{RM}\binom{\boldsymbol{x}^*}{r^*}$ are indistinguishable.

(vi: CB and CH): Follows from Theorem 1, Item i, SPB and AEPH. □

E Details of Functional SSB Commitments

E.1 Definitions

Essentially the only difference between an SSB commitment and a functional SSB commitment is that in the former $\mathcal{S}$ is a subset of $[1\,..\,q]$ and in the latter $\mathcal{S}$ is a subset of some function set $\mathcal{F}$. For the sake of completeness we provide the formal definition below.

Definition 3. *An F-extractable functional SSB commitment scheme $\mathsf{COM} = (\mathsf{Pgen}, \mathsf{KC}, \mathsf{Com}, \mathsf{tdOpen}, \mathsf{Ext}_F)$ for a function family $\mathcal{F}$ consists of the following polynomial-time algorithms:*

Parameter generation: $\mathsf{Pgen}(1^\lambda)$ *returns parameters* p *(for example, group description). We allow F to depend on* p.

Commitment key generation: *for parameters* p*, a positive integer* $n \in \mathsf{poly}(\lambda)$*, an integer* $q \in [1\,..\,n]$*, and a tuple* $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ *with* $|\mathcal{S}| \leq q$*,* $\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$ *outputs a commitment key* $\mathtt{ck}$ *and a trapdoor* $\mathtt{td} = (\mathtt{ek}, \mathtt{tk})$*. Here,* $\mathtt{ck}$ *implicitly specifies* p*, the message space* $\mathtt{MSP}$*, the randomizer space* $\mathtt{RSP}$*, and the commitment space* $\mathtt{CSP}$*, such that* $F(\mathtt{MSP}) \subseteq \mathtt{CSP}$*,* $\mathtt{ek}$ *is the* extraction key*, and* $\mathtt{tk}$ *is the* trapdoor key*. For any other input,* KC *outputs* $(\mathtt{ck}, \mathtt{td}) = (\bot, \bot)$.

Commitment: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$*, a commitment key* $\mathtt{ck} \neq \bot$*, a message* $\boldsymbol{x} \in \mathtt{MSP}^n$*, and a randomizer* $r \in \mathtt{RSP}$*,* $\mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; r)$ *outputs a commitment* $c \in \mathtt{CSP}$*.*

Trapdoor opening: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$*,* $\mathcal{S} \subseteq \mathcal{F}$ *with* $|\mathcal{S}| \leq q$*,* $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$*, two messages* $\boldsymbol{x}, \boldsymbol{x}^* \in \mathtt{MSP}^n$*, and a randomizer* $r \in \mathtt{RSP}$*,* $\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}, r, \boldsymbol{x}^*)$ *returns a randomizer* $r^* \in \mathtt{RSP}$*.*

Extraction: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$*,* $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ *with* $1 \leq |\mathcal{S}| \leq q$*,* $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$*, and* $c \in \mathtt{CSP}$*,* $\mathsf{Ext}_F(\mathsf{p}, \mathtt{ek}; c)$ *returns a tuple* $\big(F(f_1(x)), \ldots, F(f_{|\mathcal{S}|}(x))\big) \in \mathtt{MSP}^{|\mathcal{S}|}$*;*

For $\{f_i\}_{i=1}^q \subseteq \mathcal{F}$ and vector $\boldsymbol{x}$ let us denote $\boldsymbol{x}_\mathcal{S} = (f_1(\boldsymbol{x}), \ldots, f_q(\boldsymbol{x}))$.

Definition 4. *An F-extractable functional SSB commitment scheme* COM *for function family* $\mathcal{F}$ *is* secure *if it satisfies the following security requirements.*

Somewhere Statistically Binding (SSB): $\forall \lambda$*, unbounded* $\mathcal{A}$*,* $n \in \mathsf{poly}(\lambda)$*,* $q \in [1\,..\,n]$*,* $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \approx_\lambda 0$*, where* $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \mathcal{S} \subseteq \mathcal{F} \wedge |\mathcal{S}| \leq q; \\ (\mathtt{ck}, \mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{r}_0, \boldsymbol{r}_1) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} \neq \boldsymbol{x}_{1\mathcal{S}} : \\ \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_0; \boldsymbol{r}_0) = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_1; \boldsymbol{r}_1)\end{bmatrix} .$$

We say that COM *is* somewhere perfectly binding *(SPB) if* $\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) = 0$*.*

Almost Everywhere Statistically Hiding (AESH): $\forall \lambda$*, unbounded* $\mathcal{A}$*,* $n \in \mathsf{poly}(\lambda)$*,* $q \in [1\,..\,n]$*,* $\mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| \approx_\lambda 0$*, where* $\varepsilon^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \mathcal{S} \subseteq \mathcal{F} \wedge |\mathcal{S}| \leq q; \\ (\mathtt{ck}, \mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} = \boldsymbol{x}_{1\mathcal{S}}; \\ \beta \leftarrow_\$ \{0,1\}; \boldsymbol{r} \leftarrow_\$ \mathtt{RSP} : \mathcal{A}(\mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_\beta; \boldsymbol{r})) = \beta\end{bmatrix} .$$

COM *is* almost everywhere perfectly hiding *(AEPH) if* $\mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) = 0$*.*

Somewhere Statistical F-Extractability (F-SSE): $\forall \lambda$*,* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$*,* $n \in \mathsf{poly}(\lambda)$*,* $q \in [1\,..\,n]$*,* $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ *with* $|\mathcal{S}| \leq q$*,* $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$*, and PPT* $\mathcal{A}$*,* $\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) \approx_\lambda 0$*, where* $\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\big[\boldsymbol{x}, r \leftarrow \mathcal{A}(\mathtt{ck}) : \mathsf{Ext}_F(\mathsf{p}, \mathtt{ek}; \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; r)) \neq \big(F(f_1(\boldsymbol{x})), \ldots, F(f_{|\mathcal{S}|}(\boldsymbol{x}))\big)\big] .$$

It is somewhere perfect extractable if $\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) = 0$*.*

Almost Everywhere Statistical Trapdoor (AEST): $\forall \lambda$*,* $n \in \mathsf{poly}(\lambda)$*,* $q \in [1\,..\,n]$ *and unbounded* $\mathcal{A}$*,* $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)(\lambda) \approx_\lambda 0$*, where* $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)(\lambda) =$

$$\Pr\begin{bmatrix}\mathsf{p} \in \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \text{ s.t. } \mathcal{S} \subseteq \mathcal{F} \wedge |\mathcal{S}| \leq q; \\ (\mathtt{ck}, \mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{r}_0) \leftarrow \mathcal{A}(\mathtt{ck}) \text{ s.t. } \boldsymbol{x}_{0\mathcal{S}} = \boldsymbol{x}_{1\mathcal{S}} : \\ \boldsymbol{r}^* \leftarrow \mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}, \boldsymbol{r}, \boldsymbol{x}^*) : \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}; \boldsymbol{r}) \neq \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}^*; \boldsymbol{r}^*)\end{bmatrix} .$$

It is AEPT (almost everywhere perfect trapdoor) if $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)(\lambda) = 1$.

Computational Binding (CB): $\forall$ *PPT* $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) = \mathsf{negl}(\lambda)$, *where* $\mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \ \textit{s.t.}\ \mathcal{S} \subseteq \mathcal{F} \wedge |\mathcal{S}| \leq q; \\ (\mathtt{ck}, \mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{r}_0, \boldsymbol{r}_1) \leftarrow \mathcal{A}(\mathtt{ck}) \ \textit{s.t.}\ \boldsymbol{x}_0 \neq \boldsymbol{x}_1 : \\ \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_0; \boldsymbol{r}_0) = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_1; \boldsymbol{r}_1)\end{bmatrix} .$$

Computational Hiding (CH): $\forall$ *PPT* $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathsf{Adv}^{\mathrm{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2{\cdot}|\varepsilon^{\mathsf{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| = \mathsf{negl}(\lambda)$, *where* $\varepsilon^{\mathsf{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

$$\Pr\begin{bmatrix}\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q) \ \textit{s.t.}\ \mathcal{S} \subseteq \mathcal{F} \wedge |\mathcal{S}| \leq q; \\ (\mathtt{ck}, \mathtt{td}) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}); (\boldsymbol{x}_0, \boldsymbol{x}_1) \leftarrow \mathcal{A}(\mathtt{ck}); \beta \leftarrow_{\$} \{0,1\}; \\ \boldsymbol{r} \leftarrow_{\$} \mathsf{RSP} : \mathcal{A}(\mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_\beta; \boldsymbol{r})) = \beta\end{bmatrix} .$$

E.2 Security proofs

Before proving the security of linear EMP, let us recall some well-known decisional assumptions.

Decisional Diffie-Hellman (DDH) Assumption. Let $\iota \in \{1,2\}$. $DDH_{\mathbb{G}_\iota}$ holds relative to Pgen, if $\forall$ PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{A},\iota,\mathsf{Pgen}}(\lambda) := |\varepsilon^0_{\mathcal{A}}(\lambda) - \varepsilon^1_{\mathcal{A}}(\lambda)| = \mathsf{negl}(\lambda)$, where

$$\varepsilon^\beta_{\mathcal{A}}(\lambda) := \Pr\left[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); x, y, z \leftarrow_{\$} \mathbb{Z}_p : \mathcal{A}(\mathsf{p}, [x, y, xy + \beta z]_\iota) = 1 \right] .$$

Rank Assumption. Let $\iota \in \{1,2\}$. (ℓ, k, r_0, r_1)-*Rank assumption* for $1 \leq r_0 < r_1 \leq \min(\ell, k)$ holds relative to Pgen, if $\forall$ PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{rank}}_{\mathcal{A},\ell,k,r_0,r_1,\iota,\mathsf{Pgen}}(\lambda) := |\varepsilon^0_{\mathcal{A}}(\lambda) - \varepsilon^1_{\mathcal{A}}(\lambda)| = \mathsf{negl}(\lambda)$, if

$$\varepsilon^\beta_{\mathcal{A}}(\lambda) := \Pr\left[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathbf{A} \leftarrow_{\$} \mathcal{U}^{(r_\beta)}_{\ell k} : \mathcal{A}(\mathsf{p}, [\mathbf{A}]_\iota) = 1 \right] ,$$

where $\mathcal{U}^{(r_\beta)}_{\ell k}$ is the uniform distribution over rank r_β matrices $\mathbb{Z}^{\ell \times k}_p$.

Theorem 3 ([43]). *Let* $\iota \in \{1,2\}$. *For any* $\ell, k, r_0, r_1 \in \mathbb{Z}$ *such that* $1 \leq r_0 < r_1 \leq \min(\ell, k)$, *any PPT* $\mathcal{A}$, *and any* Pgen,

$$\mathsf{Adv}^{\mathrm{rank}}_{\mathcal{A},\ell,k,r_0,r_1,\iota,\mathsf{Pgen}}(\lambda) \leq \lceil \log_2(r_1/r_0) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{A},\iota,\mathsf{Pgen}}(\lambda) .$$

Theorem 4. *Let* Pgen_{bg} *be a bilinear group generator. Fix* n *and* q. *The commitment scheme in Fig. 3 is*

(i) FSH relative to Pgen_{bg} *under the* $DDH_{\mathbb{G}_\iota}$ *assumption: for each PPT* $\mathcal{A}$, *there exists a PPT* $\mathcal{B}$, *such that* $\mathsf{Adv}^{\mathrm{fsh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq \lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B},\iota,\mathsf{Pgen}}(\lambda)$.

(ii) F-SSE for $F = [\cdot]_\iota$ *(thus,* F *depends on* p*),*

(iii) SPB,

(iv) AEPH,
(v) AEPT,
(vi) CB and CH.

Proof. **(i: FSH)** Since given a matrix $\boldsymbol{M'}$ of rank $r \in [1\,..\,q+1]$, the matrix $\boldsymbol{RM'}$ is a random matrix of rank r with an overwhelming probability. Then, distinguishing commitment keys $\mathsf{ck}_1 = [\boldsymbol{R}_1\boldsymbol{M'}_1]_\iota$ and $\mathsf{ck}_2 = [\boldsymbol{R}_2\boldsymbol{M'}_2]_\iota$ is equivalent to breaking the rank assumption. Now, considering Theorem 3 we get that for each adversary $\mathcal{A}$ against FSH, there exists an adversary $\mathcal{B}$ against the DDH in $\mathbb{G}_\iota$ such that the bound $\mathsf{Adv}^{\mathrm{fsh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \simeq \mathsf{Adv}^{\mathrm{rank}}_{\mathcal{B},\iota,\mathsf{Pgen}}(\lambda) \leq \lceil \log_2(r_1/r_0) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B},\iota,\mathsf{Pgen}}(\lambda)$ holds. In the worst case one matrix has rank $r_0 = 1$ and the other has rank $r_1 = q+1$, so the worst bound is $\lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B},\iota,\mathsf{Pgen}}(\lambda)$.

(ii: F-SSE) For any $\boldsymbol{x} \in \mathbb{Z}_p^n$ and $\boldsymbol{r} \in \mathbb{Z}_p^{q+1}$, we have $\mathsf{Com}(\mathsf{ck}; \boldsymbol{x}; r) = [\boldsymbol{RM'}\left(\begin{smallmatrix}\boldsymbol{x}\\ r\end{smallmatrix}\right)]_\iota = [\boldsymbol{c}]_\iota =$. Then, $\mathsf{Ext}(\mathsf{p}, \mathsf{ek} = \boldsymbol{R}^{-1}; [\boldsymbol{c}]_\iota)$ computes $\boldsymbol{R}^{-1}[\boldsymbol{c}]_\iota = [\boldsymbol{M'}\left(\begin{smallmatrix}\boldsymbol{x}\\ r\end{smallmatrix}\right)]_\iota = \left[\begin{smallmatrix}\boldsymbol{Mx}\\ \boldsymbol{r}^\top\boldsymbol{x}+r\end{smallmatrix}\right]_\iota$ and outputs $[\boldsymbol{Mx}]_\iota$ which is exatly what we wanted to extract.

(iii: SPB) Clearly, there are no $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{Z}_p^n$ such that $\boldsymbol{Mx}_0 \neq \boldsymbol{Mx}_1$ and $[\boldsymbol{c}]_\iota := \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_0; \boldsymbol{r}_0) = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_1; \boldsymbol{r}_1)$ since by the F-SSE property we have that $\mathsf{Ext}(\mathsf{p}, \mathsf{ek} = \boldsymbol{R}^{-1}; [\boldsymbol{c}]_\iota) = [\boldsymbol{Mx}_0]_\iota = [\boldsymbol{Mx}_1]_\iota$.

(iv: AEPH) Suppose that the adversary $\mathcal{A}$ on input (p, n, q) outputs $\mathcal{S} = \boldsymbol{M} \in \mathbb{Z}_p^{q\times n}$, then gets as an input the public key $\boldsymbol{g} = \boldsymbol{R}\cdot\boldsymbol{M'}$ where $\boldsymbol{M'} = \left(\begin{smallmatrix}\boldsymbol{M} & \boldsymbol{0}\\ \boldsymbol{r}^\top & 1\end{smallmatrix}\right)$, $\boldsymbol{R} \in \mathbb{Z}_p^{(q+1)(q+1)}$ is some full rank matrix, and $\boldsymbol{r} \in \mathbb{Z}_p^n$, and finally outputs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ such that $\boldsymbol{Mx}_0 = \boldsymbol{Mx}_1$.

Let us analyze distributions of $C_0 = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_0; _0)$ and $C_1 = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_1; r_1)$ for a uniformly random r_0, r_1. For $\beta \in \{0,1\}$, we can define $[\boldsymbol{u}_\beta] := [\boldsymbol{M'}\left(\begin{smallmatrix}\boldsymbol{x}_\beta\\ r_\beta\end{smallmatrix}\right)] = \left[\begin{smallmatrix}\boldsymbol{Mx}_\beta\\ \boldsymbol{r}^\top\boldsymbol{x}_\beta+r_\beta\end{smallmatrix}\right]$. We see that top q elements of $\boldsymbol{u}_0$ and $\boldsymbol{u}_1$ are equal and the last element is uniformly random. Thus, $\boldsymbol{u}_0$ and $\boldsymbol{u}_1$ are indistinguishable. Since $C_\beta = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_\beta; r_\beta) = \boldsymbol{R}[\boldsymbol{u}_\beta]$, then also C_1 and C_2 are indistinguishable.

(v: AEPT) Let $r_0 \in \mathbb{Z}_p$ and $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{Z}_p^n$ such that $\boldsymbol{Mx}_0 = \boldsymbol{Mx}_1$. In tdOpen, we define $r_1 = \sum_{i\in[1\,..\,n]}(x_{0,i} - x_{1,i})r_i + r_0$. Then, $\boldsymbol{r}^\top\boldsymbol{x}_1 + r_1 = \boldsymbol{r}^\top\boldsymbol{x}_0 + r_0$. Using, the definition of $\boldsymbol{u}_b$ from the previous property, we see that $\boldsymbol{u}_0 = \boldsymbol{u}_1$ and then also $\mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_0; r_0) = \mathsf{Com}(\mathsf{ck}; \boldsymbol{x}_1; r_1)$.

(vi: CB and CH) Follows directly from the analog of Theorem 1. □

F Details in QA-NIZK Application Section 6.2

F.1 Preliminaries

Perfectly binding commitment. We use ElGamal encryption as our perfectly binding commitment. In particular, the commitment key is $\mathsf{ck} = [\boldsymbol{u}]_1 = [1, u]_1^\top$ where $u \leftarrow_{\$} \mathbb{Z}_p$ and $\mathsf{Com}_{\mathsf{ck}}(\boldsymbol{a} \in \mathbb{Z}_p^n; \boldsymbol{r} \in \mathbb{Z}_p^n) = [\boldsymbol{c}]_1 := ([\boldsymbol{r}]_1, [\boldsymbol{a}]_1 + \boldsymbol{r}[u]_1)$. In matrix form $[\boldsymbol{c}_i]_1 = a_i[\boldsymbol{e}_2]_1 + r_i[\boldsymbol{u}]_1$. To $[\cdot]_1$-extract the message, we can simply decrypt each individual ciphertext, that is $[a_i]_1 = [c_{i,2}]_1 - u[c_{i,1}]_1$ where $[\boldsymbol{c}_i]_1 = [c_{i,1}, c_{i,2}]_1^\top$.

SNARK for SAP. Let $\chi_1, \ldots, \chi_d \in \mathbb{Z}_p$ be unique interpolation points. We define

$$v(X) = \sum_{i=1}^{n} a_i v_i(X), \quad w(X) = \sum_{i=1}^{n} a_i w_i(X) \tag{2}$$

where $v_i(X)$, $w_i(X)$ are polynomials of degree less than d such that $v_i(\chi_j) = v_{ij}$ and $w_i(\chi_j) = -w_{ij}$. Moreover, let us define $p(X) = v(X)^2 - w(X)$ and $t(X) = \prod_{j=1}^{d}(X - \chi_j)$. We have that $p(\chi_j) = (\boldsymbol{a}^\top \boldsymbol{v}_j)^2 - \boldsymbol{a}^\top \boldsymbol{w}_j$ and thus the j-th SAP equation is satified exactly when χ_j is a root of $p(X)$. In particular, when all interpolation points are roots of $p(X)$, then $t(X)$ divides $p(X)$ and all the SAP equations are satisfied.

We can use these polynomial representations to construct a SNARK. Our CRS will contain $\{[s^i]_{1,2}\}_{i=1}^{d}$ where $s \leftarrow\!\!\$\, \mathbb{Z}_p$ is a secret point. The prover will compute $[V]_{1,2} = [V(s)]_{1,2}$, $[W]_1 = [W(s)]_1$ and $[H]_1 = [H(s)]_1$ where $V(X) = v(X) + \delta_v t(X)$, $W(X) = w(X) + \delta_w t(X)$, and $H(X) = (V(X)^2 - W(X))/t(X)$. Elements δ_v and δ_w are picked randomly to hide the witness. The verifier checks that the equation $[V]_1[V]_2 - [W]_1[1]_2 = [H]_1[t(s)]_2$ is satisfied. Intuitively, we can use this to show that $t(X)$ divides $P(X) := V(X)^2 - W(X)$. It is easy to see that if $t(X) \mid P(X)$ then also $t(X) \mid p(X)$ and thus the SAP relation is satisfied.

BLS argument. As a subargument, we use a QA-NIZK argument for membership in linear spaces $(\mathsf{K}_{\mathsf{bls}}, \mathsf{P}_{\mathsf{bls}}, \mathsf{V}_{\mathsf{bls}})$ defined in [26] for the bilateral linear subspace (bls) language $\mathcal{L}_{[\boldsymbol{N}_1]_1,[\boldsymbol{N}_2]_2} := \{([\boldsymbol{x}]_1, [\boldsymbol{y}]_2) \mid \exists \boldsymbol{w} \in \mathbb{Z}_p^t : \boldsymbol{x} = \boldsymbol{N}_1\boldsymbol{w} \wedge \boldsymbol{y} = \boldsymbol{N}_2\boldsymbol{w}\}$ for $\boldsymbol{N}_1 \in \mathbb{Z}_p^{n\times t}$, $\boldsymbol{N}_2 \in \mathbb{Z}_p^{m\times t}$. We use it to prove that commitments in different groups open to the same value. It has perfect completeness, strong quasi-adaptive soundness under the SKerMDH assumption, and perfect zero-knowledge. The proof size is 2 elements in $\mathbb{G}_1$ and 2 elements in $\mathbb{G}_2$. We refer the reader to the original paper for more details. We leave it as an open question if the slightly more efficient construction by Ràfols and Silva [42] can be used.

New target assumption. The q-target strong Diffie-Hellman assumption [6] says that given $\{[s^i]_{1,2}\}_{i=1}^{q}$ for a random s, it is computationally hard to find $[\nu]_T = [1/(s-r)]_T$ for any $r \in \mathbb{Z}_p$. We generalize this assumption and intuitively say that it is hard to compute $[\nu]_T = [c/(s-r)]_T$ where $r \in \mathbb{Z}_p$ and c is a constant independent of s. In order to satisfy the latter requirement, we include a challenge value $[z]_2$ and let the adversary additionally output $[c]_1$ and $[c']_2$ such that $zc = c'$. Intuitively, then c cannot depend on s^i since otherwise c' should depend on zs^i which is not a part of the challenge. For technical reasons, c in our assumption has a slightly more structured form $\beta_1^2 - \beta_2$.

Definition 5 (q-SATSDH). *The q-Square Arithmetic Target Strong Diffie-Hellman assumption holds relative to* Pgen*, if $\forall$ PPT adversaries $\mathcal{A}$,*

$$\Pr\left[\begin{array}{l} \mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); s, z \leftarrow\!\!\$\, \mathbb{Z}_p; \\ \left(r, [\beta_1, \beta_2]_1, [\tilde{\beta}_1, \tilde{\beta}_2]_2, [\nu]_T\right) \leftarrow \mathcal{A}\left(\mathsf{p}, \{[s^i]_{1,2}\}_{i=1}^{q}, [z]_2\right) : \\ \tilde{\beta}_1 = z\beta_1 \wedge \tilde{\beta}_2 = z\beta_2 \wedge \beta_1^2 \neq \beta_2 \wedge \nu = \frac{\beta_1^2 - \beta_2}{s-r} \end{array}\right] \approx_\lambda 0.$$

We prove in the following that our new assumption is falsifiable and equivalent to TSDH assumption under a knowledge assumption.

Let us first see that q-SATSDH is falsifiable. Observe that the challenger knows $z, s \in \mathbb{Z}_p$. Thus, upon receiving $(r, [\beta_1, \beta_2]_1, [\tilde{\beta}_1, \tilde{\beta}_2]_2, [\nu]_T)$ it verifies that: (a) $[1]_1[\tilde{\beta}_1]_2 = [\beta_1]_1[z]_2$, (b) $[1]_1[\tilde{\beta}_2]_2 = [\beta_2]_1[z]_2$, (c) $\frac{1}{z}[\beta_1]_1[\tilde{\beta}_1]_2 \neq [\beta_2]_1[1]_2$, and (d) $(s-r)[\nu]_T = \frac{1}{z}[\beta_1]_1[\tilde{\beta}_1]_2 - [\beta_2]_1[1]_2$.

We prove that if the Knowledge of Exponent Assumption in bilinear groups holds, then both q-TSDH and q-SATSDH assumptions are equivalent. We recall in the following the definition of the Bilinear Bilinear Diffie-Hellman Knowledge of Exponent assumption.

Definition 6 (Bilinear Diffie-Hellman Knowledge of Exponent Assumption, BDH-KE [1]). *For all non-uniform PPT adversaries $\mathcal{A}$:*

$$\Pr\left[\left([\alpha_1]_1, [\alpha_2]_2 \| a\right) \leftarrow (\mathcal{A} \| \mathcal{X}_{\mathcal{A}})(\mathsf{gk}) : e\left([\alpha_1]_1, [1]_2\right) = e\left([1]_1, [\alpha_2]_2\right) \wedge a \neq \alpha_1\right] \approx 0,$$

where the probability is taken over $\mathsf{gk} \leftarrow \mathsf{Pgen}(1^\lambda)$ *and the coin tosses of adversary $\mathcal{A}$.*

Lemma 5. *Given a bilinear group* $\mathsf{gk} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$*, if the* q-SATSDH *assumption holds then the* q-TSDH *assumption holds.*

Proof. Assume that $\mathcal{A}$ is an adversary against the q-TSDH assumption, we construct another adversary $\mathcal{B}$ against q-SATSDH assumption that receives a challenge tuple $(\mathsf{gk}, \{[s^i]_{1,2}\}_{i=1}^q, [z]_2)$ and sends the elements $(\mathsf{gk}, \{[s^i]_{1,2}\}_{i=1}^q)$ to $\mathcal{A}$. $\mathcal{A}$ then returns $(r, [\nu]_T)$ that breaks q-TSDH. The adversary $\mathcal{B}$ chooses $\beta_1, \beta_2 \leftarrow \mathbb{Z}_p$ such that $\beta_1^2 \neq \beta_2$ and returns $\left(r, [\beta_1, \beta_2]_1, \beta_1[z]_2, \beta_2[z]_2, (\beta_1^2 - \beta_2)[\nu]_T\right)$ which breaks the q-SATSDH assumption. □

Lemma 6. *Given a bilinear group* $\mathsf{gk} = (q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ *where* BDHKE *assumption holds, if the* q-TSDH *assumption holds then the* q-SATSDH *assumption holds.*

Proof. Assume that $\mathcal{A}$ is an adversary against the q-SATSDH assumption, we construct an another adversary $\mathcal{B}$ against the q-TSDH assumption that receives a challenge tuple $(\mathsf{gk}, \{[s^i]_{1,2}\}_{i=1}^q)$. $\mathcal{B}$ chooses $z \leftarrow \mathbb{Z}_p$ and sends the elements $(\mathsf{gk}, \{[s^i]_{1,2}\}_{i=1}^q, [z]_2)$ to $\mathcal{A}$. The adversary $\mathcal{A}$ then returns $(r, [\beta_1, \beta_2]_1, [\beta_3, \beta_4]_2, [\nu]_T)$ that breaks q-SATSDH. Now $\mathcal{B}$ computes $[\hat{\beta}_1]_2 = \frac{1}{z}[\beta_3]_2$ and $[\hat{\beta}_2]_2 = \frac{1}{z}[\beta_4]_2$ which satisfy $e([\beta_i]_1, [1]_2) = e([1]_1, [\hat{\beta}_i]_2)$ for $i = 1, 2$. By the BDHKE assumption there exists and extractor of β_1, β_2 that solves the q-TSDH assumption with $\left(r, \frac{1}{\beta_1^2 - \beta_2}[\nu]_T\right)$. □

F.2 Security of our QA-NIZK in Section 6.2

Security intuition. In the security proof, the soundness game is first changed by randomly picking one of the SAP equations $\left(\boldsymbol{a}^\top \boldsymbol{v}_{j^*}\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_{j^*} = 0$ for some

$j^* \in [1\,..\,d]$; with probability $\geq 1/d$ this equation does not hold, assuming that the adversary is successful. By the characterization of the SAP, if the j^*-th equation does not hold, then $X - \chi_{j^*} \nmid P(X)$. In particular, let $q_v(X), q_w(X)$ be unique polynomials and $\beta_v, \beta_w \in \mathbb{Z}_p$ be unique values such that $V(X) = q_v(X)(X - \chi_{j^*}) + \beta_v$ and $W(X) = q_w(X)(X - \chi_{j^*}) + \beta_w$. Then we can express the division of $P(X) = V(X)^2 - W(X)$ by $X - \chi_{j^*}$ as follows,

$$\begin{aligned} P(X) &= V(X)(q_v(X)(X - \chi_{j^*}) + \beta_v) - q_w(X)(X - \chi_{j^*}) - \beta_w \\ &= (X - \chi_{j^*})\,(V(X)q_v(X) - q_w(X)) + V(X)\beta_v - \beta_w \\ &= (X - \chi_{j^*})\,(V(X)q_v(X) - q_w(X)) + (q_v(X)(X - \chi_{j^*}) + \beta_v)\beta_v - \beta_w \\ &= (X - \chi_{j^*})\,(q_v(X)\,(V(X) + \beta_v) - q_w(X)) + (\beta_v^2 - \beta_w)\ . \end{aligned} \tag{3}$$

Since, $X - \chi_{j^*} \nmid P(X)$ we get that $(\beta_v^2 - \beta_w) \neq 0$.

We denote by $\alpha_i(X)$ and $\beta_{v,i}$ the quotient and the remainder of the polynomial division of $v_i(X)$ by $X - \chi_{j^*}$, i.e., $v_i(X) = \alpha_i(X)(X - \chi_{j^*}) + \beta_{v,i}$. Similarly, we can also express $w_i(X) = \hat{\alpha}_i(X)(X - \chi_{j^*}) + \beta_{w,i}$. As a special case, we define $t(X) = \alpha_t(X)(X - \chi_{j^*}) + \beta_t$. The definition of $V(X)$ and Eq. (2) give us $V(X) = (\sum_{i=1}^n a_i\alpha_i(X) + \delta_v\alpha_t)\,(X - \chi_{j^*}) + \sum_{i=1}^n a_i\beta_{v,i} + \delta_v\beta_t$, and thus

$$q_v(X) = \sum_{i=1}^n a_i\alpha_i(X) + \delta_v\alpha_t, \quad \beta_v = \sum_{i=1}^n a_i\beta_{v,i} + \delta_v\beta_t. \tag{4}$$

Similarly, we get that

$$q_w(X) = \sum_{i=1}^n a_i\hat{\alpha}_i(X) + \delta_w\beta_t, \quad \beta_w = \sum_{i=1}^n a_i\beta_{w,i} + \delta_w\beta_t. \tag{5}$$

The security proof extracts the following functions of the witness $\boldsymbol{a}$ and δ_v, δ_w: $[q_v(s)]_2 = [\sum_{i=1}^n a_i\alpha_i(s) + \delta_v\beta_t]_2$, $[\beta_v z]_2 = [\sum_{i=1}^n a_i z\beta_{v,i} + \delta_v z\beta_t]_2$, and $[\beta_w z]_2 = [\sum_{i=1}^n a_i z\beta_{w,i} + \delta_w z\beta_t]_2$, where $z, s \in \mathbb{Z}_p$ are secrets of SATSDH assumption. The idea is that we can break the d-SATSDH assumption by computing $[\beta_v]_1 = \sum_{i=1}^n \beta_{v,i}[a_i]_1 + \beta_t[\delta_v]_1$ (note that $[a_i]_1$ and $[\delta_v]_1$ are extractable from the PB commitment and $[V]_1$), $[\beta_w]_1 = \sum_{i=1}^n \beta_{w,i}[a_i]_1 + \beta_t[\delta_w]_1$ and moreover by Eq. (3), $\left[\frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T = \left[\frac{P(s)}{s - \chi_{j^*}}\right]_T - ([V]_1 + [\beta_v]_1)[q_v(s)]_2 + [q_w(s)]_T$, where $[\frac{P(s)}{s - \chi_{j^*}}]_T$ can be computed from the verification equation. Together with other extracted elements, this is now enough to break the SATSDH assumption. We refer to Theorem 6 for more details.

Proofs of security. The following two theorems prove the completeness, zero-knowledge, and soundness properties of our QA-NIZK construction.

Theorem 5. *The QA-NIZK argument has perfect completeness and perfect zero-knowledge.*

Proof. Completeness. Since the BLS argument is perfectly complete, we only need to check the last verification equation: the left hand side is $[V]_1[V]_2 - [W]_1[1]_2 = \left[V^2 - W\right]_T = [P(s)]_T$, and the right hand side is $[H]_1[t(s)]_2 = [H(s)]_1[t(s)]_2 = [P(s)]_T$.

Zero-knowledge. We prove it by showing that the proof can be efficiently simulated given the BLS trapdoor $\mathsf{td}_{\mathsf{bls}}$. Since we set $S_v = \emptyset$, then the SSB commitments are perfectly hiding by the AEPH property. Thus we may simulate $[\tilde{\boldsymbol{c}}]_2$ by committing to $\mathbf{0}$. Next, V and W are uniformly random and independently distributed in the honest proof. Hence, the simulator can pick $\mu_1, \mu_2 \leftarrow_{\$} \mathbb{Z}_p$ and define $[V]_{1,2} = \mu_1[t(s)]_{1,2}$, $[W]_1 = \mu_2[t(s)]_1$. Then, $[H]_1 = \mu_1^2[t(s)]_1 - [\mu_2]_1$ and the verification equation will be satisfied. Finally, the BLS proof ψ can be perfectly simulated (see [26]) using the trapdoor $\mathsf{td}_{\mathsf{bls}}$. □

Theorem 6. *Let $\mathsf{Adv}_{snd}(\mathcal{A})$ be the advantage of any PPT adversary $\mathcal{A}$ against the soundness of the QA-NIZK argument. There exist PPT adversaries $\mathcal{B}_1$ against the* DDH *assumption in $\mathbb{G}_2$, $\mathcal{B}_2$ against strong soundness of the BLS argument, and $\mathcal{B}_3$ against the d-*SATSDH *assumption such that*

$$\mathsf{Adv}_{\mathrm{Snd}}(\mathcal{A}) \leq 3\mathsf{Adv}_{\mathsf{DDH},\mathbb{G}_2}(\mathcal{B}_1) + d\big(\mathsf{Adv}_{\mathsf{bls}}(\mathcal{B}_2) + \mathsf{Adv}_{d\text{-}\mathsf{SATSDH}}(\mathcal{B}_3)\big).$$

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 $\left(\boldsymbol{a}^\top \boldsymbol{v}_i\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_i \neq 0$ and the verifier accepts the proof. Note that $\boldsymbol{a}$ is uniquely determined since commitment $[\boldsymbol{c}]_1$ is perfectly binding.
- Game_0: This game is identical to the previous one, except instead of generating the commitment key as $\mathsf{ck} \leftarrow \mathcal{D}_{\mathsf{p}}(n,d)$, the game samples $u \leftarrow_{\$} \mathbb{Z}_p$ himself, sets $\mathsf{ck} = [1,u]_1^\top$, and stores u. Clearly, $\mathcal{A}$'s advantage is the same in Real and Game_0.
- Game_1: This game is identical to the previous one except that some $j^* \leftarrow_{\$} [1\,..\,d]$ is chosen randomly and we change the commitment key ck' by using a different matrix $\mathbf{M} \neq \mathbf{0}$ during its generation. For each $i \in [1\,..\,n]$, let us express

$$v_i(X) = \alpha_i(X)(X - \chi_{j^*}) + \beta_{v,i}$$
$$w_i(X) = \hat{\alpha}_i(X)(X - \chi_{j^*}) + \beta_{w,i}$$

  and $t(X) = \alpha_t(X)(X - \chi_{j^*}) + \beta_t$. We will pick $[z]_2 \leftarrow_{\$} \mathbb{G}_2$ that is part of the SATSDH challenge and change the EMP commitment key ck' by setting

$$\mathbf{M} = \begin{pmatrix} \alpha_1(s) & \dots & \alpha_n(s) & \alpha_{n+1}(s) \\ \beta_{v,1}z & \dots & \beta_{v,n}z & 0 \\ \beta_{w,1}z & \dots & \beta_{w,n}z & 0 \\ v_{j^*,1} & \dots & v_{j^*,n} & 0 \end{pmatrix}.$$

  It is important to note that from $\{[s^i]_{1,2}\}_{i=1}^d$ and $[z]_2$ we can only compute $[\mathbf{M}]_2$. However, looking at the KC algorithm in Fig. 3, it is clear that ck' can

be computed even if only $[\mathbf{M}]_2$ is known. The game aborts if $\boldsymbol{a}$ satisfies the j^*-th equation , i.e. if $\left(\boldsymbol{a}^\top \boldsymbol{v}_{j^*}\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_{j^*} = 0$[6].

Let us now analyze the games.

Lemma 7. *There exists an adversary $\mathcal{B}_1$ against DDH in $\mathbb{G}_2$ such that* $|\Pr[\mathsf{Game}_0(\mathcal{A}) = 1] - \Pr[\mathsf{Game}_1(\mathcal{A}) = 1]| \leq 3\mathsf{Adv}_{\mathsf{DDH},\mathbb{G}_2}(\mathcal{B}_1)$.

Proof. Game_0 and Game_1 differ only in the linear EMP commitment key that encode different functions, but these keys are indistinguishable due to the FSH property. In particular, we can bound the advantage of an adversary $\mathcal{B}_1$ against the $DDH_{\mathbb{G}_2}$ assumption as in Theorem 4: $\mathsf{Adv}^{\mathrm{fsh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq \lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B}_1,2,\mathsf{Pgen}}(\lambda)$ where in this case $q = 4$. □

Lemma 8. *There exists an adversary $\mathcal{B}_2$ against the strong soundness of the* bls *proof and a d-*SATSDH *adversary $\mathcal{B}_3$ such that*

$$\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] \leq d\left(\mathcal{A}_{\mathsf{bls}}(\mathcal{B}_2) + \mathcal{A}_{d\text{-}\mathsf{SATSDH}}(\mathcal{B}_3)\right).$$

Proof. First of all, if $\mathcal{A}$ breaks soundness, at least one equation j^* does not hold, and the challenger can guess j^* with probability at least $\frac{1}{d}$.

Let E be the event that $([\boldsymbol{c}]_1, [V]_1, [W]_1, [V]_2, [\tilde{\boldsymbol{c}}]_2)^\top \in \mathbf{Im}\begin{pmatrix}[\mathbf{N}_1]_1\\ [\mathbf{N}_2]_2\end{pmatrix}$ and $\overline{E}$ be the complementary event. Obviously,

$$\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] \leq \Pr[\mathsf{Game}_1(\mathcal{A}) = 1 | E] + \Pr[\mathsf{Game}_1(\mathcal{A}) = 1 | \overline{E}]. \tag{6}$$

For the latter event, we can easily construct from $\mathcal{A}$ a PPT adversary $\mathcal{B}_2$ that breaks strong quasi-adaptive soundness of the BLS argument. Such an adversary receives as an input $(\mathtt{crs}_{\mathsf{bls}}, \varrho = ([\mathbf{N}_1]_1, [\mathbf{N}_2]_2), \omega_\rho = (\mathbf{N}_1, \mathbf{N}_2))$ sampled according to the distribution specified by Game_3. In particular, $\mathbf{N}_2$ contains $t(s)$ and thus $\mathcal{B}_2$ can efficiently recover s by finding roots of the polynomial $t(X) - t(s)$. This is sufficient to construct the rest of the CRS chosen in the usual way. Now adversary $\mathcal{B}_2$ can use the output of $\mathcal{A}$ to break the soundness of bls in a straightforward way. Thus, $\Pr[\mathsf{Game}_1(\mathcal{A}) = 1 | \overline{E}] \leq \mathsf{Adv}_{\mathsf{bls}}(\mathcal{B}_2)$.

In the following, we bound the first term of the sum in Eq. (6) by constructing an adversary $\mathcal{B}_3$ which breaks the d-SATSDH assumption in the case that E happens. Note that in this case there exists a witness $(\boldsymbol{a}, \boldsymbol{r}, \delta_v, \delta_w, r_v)^\top$ for membership in $\mathbf{Im}\begin{pmatrix}[\mathbf{N}_1]_1\\ [\mathbf{N}_2]_2\end{pmatrix}$. Furthermore, this witness is unique since

- $[\boldsymbol{c}]_1$ is perfectly binding and thus uniquely fixes $\boldsymbol{a}$ and $\boldsymbol{r}$,
- $[V]_1$ and $\boldsymbol{a}$ uniquely fix δ_v,
- $[W]_1$ and $\boldsymbol{a}$ uniquely fix δ_w, and
- $[\boldsymbol{a}]_1$ and δ_v uniquely fix r_v.

[6] This statement is well-defined since $\boldsymbol{a}$ is uniquely determined by the commitment $[\boldsymbol{c}]_1$. The check can be done in $\mathbb{G}_T$ from $[a_i]_1$ and $[\sum a_i v_{j^*,i}]_2$.

In particular, this uniquely determines the polynomial $P(X) = (v(X) + \delta_v t(X))^2 - w(X) + \delta_w t(X)$.

We now describe the full reduction. Adversary $\mathcal{B}_3$ receives the d-SATSDH assumption challenge $\left(\mathsf{p}, \{[s^i]_{1,2}\}_{i=1}^q, [z]_2\right)$ and uses this to construct the CRS just as it is specified in Game_1. Note that to create the commitment key $\mathtt{ck}'$, it constructs the matrix $\mathbf{M}$ and the corresponding extraction key $\mathtt{ek}'$. The CRS is then sent to the soundness adversary $\mathcal{A}$ that returns $[\boldsymbol{c}]_1$ and π.

The adversary $\mathcal{B}_3$ extracts $[\boldsymbol{a}]_1, [\delta_v]_1, [\delta_w]_1 \in \mathbb{G}_1$ from $[\boldsymbol{c}]_1$ by using the secret key u; and extracts $[q_v(s)]_2 = [\sum_{i=1}^{n+1} a_i \alpha_i(s) + \delta_v \alpha_{n+1}(s)]_2$, $[\beta_v z]_2$, $[\beta_w z]_2$ and $[\sum_i a_i v_{j^*,i}]_2$ from $\mathtt{ek}'$. Then it aborts if the j^*-th equation is satisfied, i.e. if

$$\left(\sum_{i=1}^{n} [a_i]_1 v_{j^*,i}\right) \cdot \left[\sum_{i=1}^{n} a_i v_{j^*,i}\right]_2 - \left(\sum_{i=1}^{n} [a_i]_1 w_{j^*,i}\right) \cdot [1]_2 = [0]_T.$$

Since verification succeeds, $[V]_1[V]_2 - [W]_T = [H(s)]_1[t(s)]_2$. By the definition of $P(X)$, we have that the left hand side is $[V^2 - W]_T = [P(s)]_T$.

If we divide both sides of the verification equation by $s - \chi_{j^*}$, then

$$\left[\frac{P(s)}{s-\chi_{j^*}}\right]_T = [H]_1 \cdot \left[\frac{t(s)}{s-\chi_{j^*}}\right]_2 = [H]_1 \cdot \left[\prod_{i \neq j^*} (s - \chi_i)\right]_2,$$

so the adversary $\mathcal{B}_3$ can compute $\left[\frac{P(s)}{s-\chi_{j^*}}\right]_T$ from $[H]_1$ and the powers of $[s]_2$ in the CRS. On the other hand, if we use equation (3) on $P(X)$, then

$$\left[\frac{P(s)}{s-\chi_{j^*}}\right]_T = \left[(V(s) + \beta_v) q_v(s) - q_w(s) + \frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T, \tag{7}$$

and we have $\beta_v^2 - \beta_w \neq 0$ (otherwise the j^*-th equation is satisfied, in which case the game aborts). We describe in the following how $\mathcal{B}_3$ can compute the right hand side of Eq. (7) and the elements to break the d-SATSDH Assumption.

According to Eq. (4) and Eq. (5), $\mathcal{B}_3$ can compute $[\beta_v]_1 = \sum_{i=0}^{n} [a_i]_1 \beta_{v,i} + [\delta_v]_1 \beta_t$, $[\beta_w]_1 = \sum_{i=0}^{n} [a_i]_1 \beta_{w,i} + [\delta_w]_1 \beta_t$ and also $[V(s) + \beta_v]_1 = [V]_1 + [\beta_v]_1$, 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 $X - \chi_{j^*}$.

From these values, the extracted values and $[V(s) + \beta_v]_2$, $\mathcal{B}_3$ can derive $[(V(s) + \beta_v) q_v(s)]_T$ as $[V(s) + \beta_v]_1 \cdot [q_v(s)]_2$. Finally, it can directly compute $[q_w(s)]_T$ from extracted elements $[a_i]_1$ for $i \in [1\,..\,n]$ and $[\delta_w]_1$, and public $\hat{\alpha}_i(s)$: $[\sum_{i=1}^{n} a_i \hat{\alpha}_i(s) + \delta_w \beta_t]_1$. Thus, from equation (7) $\mathcal{B}_3$ recovers $\left[\frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T$ and returns

$$\left(\chi_{j^*}, [\beta_v]_1, [\beta_w]_1, [z\beta_v]_2, [z\beta_w]_2, \left[\frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T\right),$$

breaking the d-SATSDH assumption.

Hence by the triangle inequality we have $\frac{1}{d}\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] \leq \mathcal{A}_{\mathsf{bls}}(\mathcal{B}_2) + \mathcal{A}_{d\text{-SATSDH}}(\mathcal{B}_3)$. □

Finally, by Lemmas 7 and 8 we get that

$$\mathsf{Adv}_{\mathrm{Snd}}(\mathcal{A}) \leq 3\mathsf{Adv}_{\mathsf{DDH},\mathbb{G}_2}(\mathcal{B}_1) + d\big(\mathsf{Adv}_{\mathsf{bls}}(\mathcal{B}_2) + \mathsf{Adv}_{d\text{-}\mathsf{SATSDH}}(\mathcal{B}_3)\big).$$

□

Efficiency. The proof size in the original construction in [15] is 4 elements in $\mathbb{G}_1$ and 6 elements in $\mathbb{G}_2$, while our construction's proof size is 5 elements in $\mathbb{G}_1$ and 8 elements in $\mathbb{G}_2$.