Somewhere Statistically Binding Commitment Schemes with Applications

December 29, 2022

Prastudy Fauzi[1], Helger Lipmaa[1,2], Zaira Pindado[3], and Janno Siim[2]

[1] Simula UiB, Bergen, Norway
[2] University of Tartu, Tartu, Estonia
[3] Universitat Pompeu Fabra, Barcelona, Spain

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 [BBB$^+$18], and are used in many privacy-preserving cryptocurrency designs such as Mimblewimble [Poe16, FOS19] and Quisquis [FMMO19].

Dual-mode commitment schemes [DN02,CV05,DFL$^+$09] 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 [GS08] 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 [HW15, OPWW15], 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 [Gro10, Lip12, GGPR13, DFGK14], 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 [DGP+19, GR19] using a generalization of the Pedersen commitment scheme called *Extended Multi-Pedersen* [GHR15, GR16] 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.[4] 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 [GHR15, GR16], 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 $\mathcal{S} \subseteq [1 \mathinner{..} n]$ of size at most $q \leq n$ and defines a commitment key 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-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 $n = q = 1$ and $|\mathcal{S}| \in \{0, 1\}$ 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 [EHK+13].

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 [DGP+19]; 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 7 we present a QA-NIZK for Square Arithmetic Programs (SAP, [GM17]) that follows a similar

[4] One cannot construct zk-SNARKs in a black-box way from falsifiable assumptions [GW11], hence any black-box construction from falsifiable assumptions will not be fully succinct.

strategy to prior work [DGP+19] but can be used for arithmetic circuit satisfiability instead of Boolean circuit satisfiability. Our QA-NIZK has comparable efficiency and its security reduces to falsifiable assumptions.

Relation to other primitives. The SSB requirement makes the EMP commitment scheme look similar to SSB hash functions [HW15, OPWW15], 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[5]. 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.

SSB commitments are directly related to two-message oblivious transfer (OT) protocols as defined in [AIR01]. 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 ck generator. This allows for more efficient constructions.

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

1.1 Corrections

This version of the paper contains some additional details and fixes a significant error compared to the conference version [FLPS21] and the prior eprint version.

1. Most importantly, our QA-NIZK for SAP contained an error. For our security proof to go through, it has to be possible to verify efficiently if the adversary has broken any SAP equations. This is possible if the prover makes a linear-length commitment to the SAP witness in both pairing groups. However, in the earlier versions of the paper, a linear-length commitment was only made in the first pairing group. We give more details about this issue in Section 7.
2. The ISH property of EMP has a much more detailed proof, and it is reduced to the standard DDH assumption. In prior work, ISH property was reduced to matrix DDH assumption.

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 $i \in [1 \mathinner{..} n]$, let α_i be its ith coefficient. Let $\boldsymbol{e}_i$ be the ith unit vector of implicitly understood dimension. For a tuple $\mathcal{S} = (\sigma_1, \dots, \sigma_q)$ with $\sigma_i < \sigma_{i+1}$, let $\boldsymbol{\alpha}_{\mathcal{S}} = (\alpha_{\sigma_1}, \dots, \alpha_{\sigma_q})$. 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 $|f - g| = \mathsf{negl}(\lambda)$. 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 $\mathcal{A}$, $\Pr[x \leftarrow\$\, \mathcal{D}^0_\lambda : \mathcal{A}(x) = 1] \approx_\lambda \Pr[x \leftarrow\$\, \mathcal{D}^1_\lambda : \mathcal{A}(x) = 1]$.

2.1 Bilinear Groups

In the case of groups, we will use additive notation together with the bracket notation [EHK+13], 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

[5] The properties of SSB and local opening are orthogonal: it is possible to construct efficient SSB hashes without local opening [OPWW15] and efficient vector commitments [LY10, CF13] (which have a local opening) without the SSB property

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}.

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

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

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 [EHK$^+$13], 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 $\iota \in \{1,2\}$. *$\mathcal{D}_{\ell k}$-$MDDH_{\mathbb{G}_\iota}$* [EHK$^+$13] 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

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

Common distributions for the MDDH assumption are $\mathcal{U}_k := \mathcal{U}_{k+1,k}$ and the linear distribution $\mathcal{L}_k$ over $\boldsymbol{A} = \left(\begin{smallmatrix} & \boldsymbol{A'} & \\ 1 & \dots & 1 \end{smallmatrix}\right)$, where $\boldsymbol{A'} \in \mathbb{Z}_p^{k\times k}$ is a diagonal matrix with $a'_{ii} \leftarrow\!\!\$\, \mathbb{Z}_p$.

Rank Assumption.

Definition 1 (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 in $\mathbb{G}_\iota$ 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*

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

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

Theorem 1 ([Vil12]). *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*,*

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

We give an alternative definition of rank assumption which has β sampled randomly. This is more consistent with other definitions in this paper.

Definition 2 ((Randomized) Rank Assumption). *Let $\iota \in \{1,2\}$.* (ℓ,k,r_0,r_1)-(Randomized) Rank assumption *for $1 \leq r_0 < r_1 \leq \min(\ell,k)$ holds in $\mathbb{G}_\iota$ relative to* Pgen*, if $\forall$ PPT $\mathcal{A}$,* $\mathsf{Adv}^{\mathrm{r\text{-}rank}}_{\mathcal{A},\ell,k,r_0,r_1,\iota,\mathsf{Pgen}}(\lambda) := 2\cdot|\varepsilon_{\mathcal{A}}(\lambda) - 1/2| \approx_\lambda 0$*, where*

$$\varepsilon_{\mathcal{A}}(\lambda) := \Pr\left[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \beta \leftarrow\!\!\$\, \{0,1\}; \mathbf{A} \leftarrow\!\!\$\, \mathcal{U}^{(r_\beta)}_{\ell,k}; \beta' \leftarrow \mathcal{A}(\mathsf{p},[\mathbf{A}]_\iota) : \beta' = \beta \quad\right].$$

If we consider adversaries $\mathcal{A}$ that output only 0 or 1 (which is always sufficient for decisional assumptions), then the advantages are in fact the same.

$$\begin{aligned}
\mathsf{Adv}^{\mathrm{r\text{-}rank}}_{\mathcal{A},\ell,k,r_0,r_1,\iota,\mathsf{Pgen}}(\lambda) :=& 2\cdot|\varepsilon_{\mathcal{A}}(\lambda) - 1/2| = 2\cdot|\Pr[\beta'=\beta] - 1/2| \\
=& 2\cdot|\frac{1}{2}\Pr[\beta'=1 \mid \beta=1] + \frac{1}{2}\Pr[\beta'=0 \mid \beta=0] - \frac{1}{2}| \\
=& |\Pr[\beta'=1 \mid \beta=1] + (1-\Pr[\beta'=1 \mid \beta=0]) - 1| \\
=& |\varepsilon^1_{\mathcal{A}} - \varepsilon^0_{\mathcal{A}}| = \mathsf{Adv}^{\mathrm{rank}}_{\mathcal{A},\ell,k,r_0,r_1,\iota,\mathsf{Pgen}}(\lambda).
\end{aligned}$$

In the following we will use the second definition.

2.2 Quasi-adaptive NIZK

A quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proof [JR13] 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 $\mathcal{D}_{\mathsf{gk}}^{\mathsf{par}}$ 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 $\Pi = (\mathsf{K}_0, \mathsf{K}_1, \mathsf{P}, \mathsf{V})$ 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:

$$\mathsf{Adv}^{\mathrm{snd}}_{\mathcal{A}_2,\Pi}(\lambda) := \Pr\left[\begin{array}{l}\mathsf{gk} \leftarrow \mathsf{K}_0(1^\lambda); \rho \leftarrow \mathcal{D}_{\mathsf{gk}}; \\ \mathtt{crs} \leftarrow \mathsf{K}_1(\mathsf{gk}, \rho); \\ (x, \pi) \leftarrow \mathcal{A}_2(\mathsf{gk}, \mathtt{crs})\end{array} : \mathsf{V}(\mathtt{crs}, x, \pi) = 1 \wedge \neg(\exists w : \mathcal{R}_\rho(x, w))\right],$$

where $\mathsf{Adv}^{\mathrm{snd}}_{\mathcal{A}_2,\Pi}(\lambda) \approx_\lambda 0$.

Computational Strong Quasi-Adaptive Soundness:

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

where $\mathsf{Adv}^{\mathrm{s\text{-}snd}}_{\mathcal{A}_2,\Pi}(\lambda) \approx_\lambda 0$.

Perfect Quasi-Adaptive Zero-Knowledge:

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

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.

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 [GS08], $n = q = 1$ 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* [BCKL08] 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 [HW15, OPWW15] who defined SSB hashing; however, the consideration of the hiding property makes the case of SSB commitments sufficiently different.

Table 1. Properties of an SSB commitment scheme

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 $F(\boldsymbol{x}_S)$

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, $n \in \mathsf{poly}(\lambda)$, $q \in [1 \mathinner{..} n]$, and a tuple $\mathcal{S} \subseteq [1 \mathinner{..} n]$ 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 $\mathtt{RSP}$, the extraction space $\mathtt{ESP}$, and the commitment space $\mathtt{CSP}$, such that $F(\mathtt{MSP}) \subseteq \mathtt{ESP}$. For invalid input, KC outputs $(\mathtt{ck}, \mathtt{td}) = (\perp, \perp)$.

Commitment: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathtt{ck} \neq \perp$, 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 $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathcal{S} \subseteq [1 \mathinner{..} n]$ with $|\mathcal{S}| \leq q$, $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$, two messages $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathtt{MSP}^n$, and a randomizer $r_0 \in \mathtt{RSP}$, $\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}_0, r_0, \boldsymbol{x}_1)$ returns a randomizer $r_1 \in \mathtt{RSP}$.

Extraction: for $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $\mathcal{S} = (\sigma_1, \ldots, \sigma_{|\mathcal{S}|}) \subseteq [1 \mathinner{..} n]$ 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 $\mathcal{A}$, $n \in \mathsf{poly}(\lambda)$, $q \in [1 \mathinner{..} n]$, $\mathsf{Adv}^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) := 2 \cdot |\varepsilon^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) - 1/2| \approx_\lambda 0$, where $\varepsilon^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

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

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

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

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 $\varepsilon^{\mathsf{aesh}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

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

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 $|\mathcal{S}| \leq q$, $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$, and PPT $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) :=$

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

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 $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \approx_\lambda 0$, where $\mathsf{Adv}^{\mathsf{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) =$

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

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

It is important to consider the case $|\mathcal{S}| \leq q$ instead of only $|\mathcal{S}| = q$. For example, when $q = n$, the perfectly binding (PB) commitment key ($|\mathcal{S}| = n$) has to be indistinguishable from the perfectly hiding (PH) commitment key ($|\mathcal{S}| = 0$). Moreover, in the applications to construct QA-NIZK argument systems [GHR15, GR16, DGP$^+$19], one should not be able to distinguish between the cases $|\mathcal{S}| = 0$ and $|\mathcal{S}| = q$.

F-extractability [BCKL08] 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 [GS08]. Note that in this case, F depends on p. Interestingly, extractability implies SSB.

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* $\mathsf{Adv}^{\mathsf{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq 2 \cdot \mathsf{Adv}^{\mathsf{sse}}_{\mathcal{B},F,\mathsf{COM},n,q}(\lambda)$.

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

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 $\beta \leftarrow_{\$} \{0,1\}$, $\mathsf{Adv}^{\mathsf{sse}}_{\beta,F,\mathsf{COM},n,q}(\lambda) \geq \mathsf{Adv}^{\mathsf{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)/2$ and hence $\mathsf{Adv}^{\mathsf{ssb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq 2 \cdot \mathsf{Adv}^{\mathsf{sse}}_{\beta,F,\mathsf{COM},n,q}(\lambda)$. □

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 $\mathsf{Adv}^{\mathsf{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$

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

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

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

Theorem 2. *Let* COM *be an SSB commitment scheme. Fix* n *and* q.

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

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 $\varepsilon_1 = \mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)$.

$\underline{\mathrm{Game}_2}$: Game_1, but instead of $\mathtt{ck}$, $\mathcal{A}$ gets $\mathtt{ck}'$ where $(\mathtt{ck}', \mathtt{td}') \leftarrow \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S}_1)$ for $\mathcal{S}_1 \leftarrow_{\$} \mathbb{P}([1 \mathinner{..} n], q)$. Note that a distinguisher $\mathcal{B}_1$ for Game_1 and Game_2 can be used to break the ISH game with advantage $\varepsilon_{\mathsf{ish}} = \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda)$. 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 2. *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* $\mathrm{Game}_2] \geq q/n - 4 \cdot \varepsilon_{\mathsf{ish}}$.

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 $\Pr[(x_0)_{\mathcal{S}_1} \neq (x_1)_{\mathcal{S}_1} \text{ in } \mathrm{Game}_2] = \varepsilon$.

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

1. Given p, n, q, $\mathcal{B}$ sets $\mathcal{S}_1 \leftarrow_{\$} \mathbb{P}([1 \mathinner{..} n], q)$ 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,

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

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

$$\begin{aligned}\mathsf{Adv}^{\mathrm{ssb}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda) &\geq \varepsilon_2 \cdot \Pr[(\boldsymbol{x}_0)_{\mathcal{S}_1} \neq (\boldsymbol{x}_1)_{\mathcal{S}_1} \text{ in Game}_2 | \boldsymbol{x}_0 \neq \boldsymbol{x}_1] \\ &\geq (\varepsilon_1 - \varepsilon_{\mathsf{ish}})(q/n - 4 \cdot \varepsilon_{\mathsf{ish}}) \text{ (due to Lemma 2).}\end{aligned}$$

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

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

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, $\mathsf{Adv}^{\mathrm{ch}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq |\varepsilon_2 - \varepsilon_1| + \varepsilon_2 \leq \mathsf{Adv}^{\mathrm{ish}}_{\mathcal{B}_1,\mathsf{COM},n,q}(\lambda) + \mathsf{Adv}^{\mathrm{aesh}}_{\mathcal{B}_2,\mathsf{COM},n,q}(\lambda)$. □

4 Constructing SSB Commitment Schemes

In this section we generalize the notion of algebraic commitment schemes 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 [RS20] defined the notion of *algebraic commitment schemes (ACSs)*, where the commitment keys are matrices, already used implicitly in other works [CGM16, CFS17]. 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 3. *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 $\mathtt{ck} = [\boldsymbol{U}_1, \boldsymbol{U}_2]_\iota \leftarrow\$ \mathcal{D}_1 \times \mathcal{D}_2$ 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., $\mathsf{Com}_{\mathtt{ck}}(\boldsymbol{x}, \boldsymbol{r}) := [\boldsymbol{U}_1]_\iota \boldsymbol{x} + [\boldsymbol{U}_2]_\iota \boldsymbol{r} \in \mathbb{G}_\iota^k$.*

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 B.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 B.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) [GHR15, GR16] is a variant of the standard vector Pedersen commitment scheme [Ped92]. In this section, we will depict a general version of the EMP commitment scheme[6] 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 $\mathsf{Ext}_{[\cdot]_\iota}$. We use a distribution $\mathcal{D}_{q+1}^{p,n,q+1,\mathcal{S}}$ 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 $\boldsymbol{g}^{(n+1)}$.[7]

Definition 4. *Let $p = p(\lambda)$, $n = \mathsf{poly}(\lambda)$, and let $\mathfrak{r} \leq q+1$, $q \leq n$ be a small positive integers. Let $\mathcal{S} \subseteq [1 \mathinner{..} n]$ with $|\mathcal{S}| \leq q$. Then the distribution $\mathcal{D}_{q+1}^{p,n,\mathfrak{r},\mathcal{S}}$ is defined as the first part of $\mathcal{D}_{gen}(p, n, \mathfrak{r}, \mathcal{S}, q)$ in Fig. 1 (i.e., just $\boldsymbol{g}$, without the associated extraction key or trapdoor).*

Note that [GR16] uses a distribution $\mathcal{D}_{q+1}$, 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 $\mathcal{U}_{q+1}$.

```
$\mathcal{M}_{gen}(p, n, \mathcal{S}, q)$
$\mathcal{S}' \leftarrow \mathcal{S} \cup \{n+1\}$;    // $\mathcal{S}' = \{\sigma_1, \ldots, \sigma_{q+1}\}$
$\boldsymbol{M} \leftarrow \mathbf{0}_{(q+1)\times(n+1)}$; $M_{q+1,n+1} \leftarrow 1$;
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 $j = \sigma_i$; $M_{i,j} \leftarrow 1$;
$\mathtt{tk} \leftarrow (\delta_j)_{j \in [1 \mathinner{..} n] \setminus \mathcal{S}}$;
return $(\boldsymbol{M}, \mathtt{tk})$;
```

```
$\mathcal{D}_{gen}(p, n, \mathfrak{r}, \mathcal{S}, q)$
$\boldsymbol{R} \leftarrow_\$ \mathcal{U}_{q+1,q+1}^{(\mathfrak{r})}$;
$(\boldsymbol{M}, \mathtt{tk}) \leftarrow \mathcal{M}_{gen}(p, n, \mathcal{S}, q)$;
$\boldsymbol{g} \leftarrow \boldsymbol{R}\boldsymbol{M}$;    // $\boldsymbol{g} \in \mathbb{Z}_p^{(q+1)\times(n+1)}$;
return $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk})$;
```

Fig. 1. Generating $\boldsymbol{M}$ from $\mathcal{S}$ (the left hand side) and $\mathcal{D}_{q+1}^{p,n,\mathfrak{r},\mathcal{S}}$ with an associated extraction key $\boldsymbol{R}$ and trapdoor $\mathtt{tk}$ (the right hand side)

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 $\boldsymbol{g} = \boldsymbol{R}\boldsymbol{M} = \left(\begin{smallmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{smallmatrix}\right)$. On the other hand, if $\mathcal{S} = \emptyset$ then $\boldsymbol{M} = \left(\begin{smallmatrix} 0 & 0 \\ \delta_1 & 1 \end{smallmatrix}\right)$ and $\boldsymbol{g} = \boldsymbol{R}\boldsymbol{M} = \left(\begin{smallmatrix} \delta_1 r_{12} & r_{12} \\ \delta_1 r_{22} & r_{22} \end{smallmatrix}\right)$ for $\delta_1 \leftarrow_\$ \mathbb{Z}_p$.

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

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

We define EMP in Fig. 2. We claim that it is indeed an SSB commitment scheme in Theorem 3. However, before that we prove a small lemma that will help us prove the ISH property.

Lemma 3. *Let $q \leq n$ and $\lambda \in \mathbb{N}$. For any (even unbounded) adversary $\mathcal{A}$,*

$$\Big| \Pr[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q); \boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1]$$
$$- \Pr[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \mathcal{S} \leftarrow \mathcal{A}(\mathsf{p}, n, q); \boldsymbol{g} \leftarrow \mathcal{U}_{q+1,n+1}^{(1)} : \mathcal{A}(\boldsymbol{g}) = 1] \Big| \leq 1/p.$$

[6] González *et al.* [GR16] mostly considered the case $q = 1$; they also did not formalize its security by using notions like ISH

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

```
$\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$    // $\mathcal{S} \subseteq \{1, 2, \ldots, n\}$ with $|\mathcal{S}| \le q$
    Sample $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk}_\iota) \leftarrow\!\!\$\, \mathcal{D}_{gen}(p, n, \mathfrak{r}, \mathcal{S}, q)$ where $\mathfrak{r} = q + 1$;
    $\mathtt{ck} \leftarrow [\boldsymbol{g}]_\iota$; $\mathsf{ek} \leftarrow \boldsymbol{R}$;    // $\boldsymbol{g} \in \mathbb{Z}_p^{(q+1)\times(n+1)}$, $\boldsymbol{R} \in \mathbb{Z}_p^{(q+1)\times(q+1)}$
    $\mathtt{td} \leftarrow (\mathsf{ek}, \mathtt{tk})$; return $(\mathtt{ck}, \mathtt{td})$;

$\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}_\iota; \boldsymbol{x}_0, r_0, \boldsymbol{x}_1)$
    $r_1 \leftarrow \sum_{i \in [1\,..\,n] \setminus \mathcal{S}} (x_{0,i} - x_{1,i})\delta_i + r_0$;
    return $r_1$;

$\mathsf{Ext}(\mathsf{p}, \mathsf{ek}; [\mathbf{c}]_\iota)$
    $[\boldsymbol{x}']_\iota \leftarrow \boldsymbol{R}^{-1}[\mathbf{c}]_\iota$;
    return $[\boldsymbol{x}_{\mathcal{S}}]_\iota \leftarrow [\boldsymbol{x}'_{[1\,..\,|\mathcal{S}|]}]_\iota$;

$\mathsf{Com}(\mathtt{ck}; \boldsymbol{x} \in \mathbb{Z}_p^n; \mathsf{r} \in \mathbb{Z}_p)$
    return $[\boldsymbol{g}]_\iota \binom{\boldsymbol{x}}{\mathsf{r}}$;    // $= \sum_{j=1}^{n} x_j [\boldsymbol{g}^{(j)}]_\iota + \mathsf{r}[\boldsymbol{g}^{(n+1)}]_\iota \in \mathbb{G}^{q+1}$
```

Fig. 2. The EMP commitment scheme COM

Proof. Let $\boldsymbol{R}$ be a uniformly random rank 1 matrix over $\mathbb{Z}_p^{(q+1)\times(q+1)}$. Alternatively, we can sample $\boldsymbol{R}$ in the following way: we sample non-zero vectors $\boldsymbol{a}, \boldsymbol{b} \leftarrow\!\!\$\, (\mathbb{Z}_p^*)^{q+1}$ and compute $\boldsymbol{R} = \boldsymbol{a}^\top \otimes \boldsymbol{b}$.

Then $\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q)$ has the form

$$\boldsymbol{g} = \boldsymbol{R}\boldsymbol{M} = (\boldsymbol{a}^\top \otimes \boldsymbol{b})\boldsymbol{M} = (\boldsymbol{a}^\top \boldsymbol{M}) \otimes \boldsymbol{b},$$

where $\boldsymbol{M}$ is the output of $\mathcal{M}_{gen}(p, n, \mathcal{S}, q)$. Let us denote $\boldsymbol{c} := \boldsymbol{a}^\top \boldsymbol{M} \in \mathbb{Z}_p^{n+1}$. We have 3 cases for elements of $\boldsymbol{c}$:

1. If $j \notin \mathcal{S}$ and $j \neq n+1$, then $c_j = \sum_{i=1}^{q+1} a_i M_{i,j} = a_{q+1} \cdot \delta_j$ since by construction of $\boldsymbol{M}$, $M_{i,j} = 0$ for $i \le q$ and $M_{q+1,j} = \delta_j$.
2. If $j \in \mathcal{S}$, then $c_j = a_i$ where i is such that $j = \sigma_i$ since $M_{i,j} = 1$ and $M_{s,j} = 0$ for all $s \neq j$.
3. If $j = n+1$, then $c_{n+1} = a_{q+1}$ since $M_{i,n+1} = 0$ for $i \le q$ and $M_{q+1,n+1} = 1$.

We get that $\boldsymbol{c}$ is distributed uniformly randomly unless $a_{q+1} = 0$. The latter happens with a probability $(p^q - 1)/(p^{q+1} - 1)$. Importantly, when $\boldsymbol{c}$ is uniformly random, then $\boldsymbol{g} = \boldsymbol{c} \otimes \boldsymbol{b}$ is a uniformly random rank 1 matrix. Thus, for any adversary $\mathcal{A}$,

$$\begin{aligned}
&\Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1] \\
&\qquad = \Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1 \mid a_{q+1} = 0] \cdot \Pr[a_{q+1} = 0] + \\
&\qquad\quad \Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1 \mid a_{q+1} \neq 0] \cdot \Pr[a_{q+1} \neq 0] \\
&\qquad = \Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1 \mid a_{q+1} = 0] \cdot \frac{p^q - 1}{p^{q+1} - 1} + \\
&\qquad\quad \Pr[\boldsymbol{g} \leftarrow \mathcal{U}^{(1)}_{q+1,n+1} : \mathcal{A}(\boldsymbol{g}) = 1] \cdot \left(1 - \tfrac{p^q - 1}{p^{q+1} - 1}\right) \quad .
\end{aligned}$$

Then,

$$\begin{aligned}
&\Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1] - \Pr[\boldsymbol{g} \leftarrow \mathcal{U}^{(1)}_{q+1,n+1} : \mathcal{A}(\boldsymbol{g}) = 1] = \\
&\quad \Big(\Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1 \mid a_{q+1} = 0] - \Pr[\boldsymbol{g} \leftarrow \mathcal{U}^{(1)}_{q+1,n+1} : \mathcal{A}(\boldsymbol{g}) = 1]\Big) \cdot \tfrac{p^q - 1}{p^{q+1} - 1} \quad .
\end{aligned}$$

It follows that

$$\begin{aligned}
&|\Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1] - \Pr[\boldsymbol{g} \leftarrow \mathcal{U}^{(1)}_{q+1,n+1} : \mathcal{A}(\boldsymbol{g}) = 1]| = \\
&\quad \Big|\Pr[\boldsymbol{g} \leftarrow \mathcal{D}_{gen}(\mathsf{p}, n, 1, \mathcal{S}, q) : \mathcal{A}(\boldsymbol{g}) = 1 \mid a_{q+1} = 0] - \Pr[\boldsymbol{g} \leftarrow \mathcal{U}^{(1)}_{q+1,n+1} : \mathcal{A}(\boldsymbol{g}) = 1]\Big| \cdot \tfrac{p^q - 1}{p^{q+1} - 1} \le \tfrac{p^q - 1}{p^{q+1} - 1} \quad .
\end{aligned}$$

Note that since $p \cdot (p^q - 1) = p^{q+1} - p \leq p^{q+1} - 1$ for all primes p and all integers $q \geq 1$, we have

$$\tfrac{p^q-1}{p^{q+1}-1} \leq \tfrac{p^q-1}{p \cdot (p^q-1)} \leq \tfrac{1}{p} \ .$$

We have proven the lemma. □

Remark 1. Since $\mathcal{A}$ is unbounded, then the same result holds even if $\mathcal{A}$ gets as an input $[\boldsymbol{g}]_\iota$ and in the first game $[\boldsymbol{g}]_\iota$ is computed as $\boldsymbol{R} \leftarrow\!\$\, \mathcal{U}^{(\mathfrak{r})}_{q+1,q+1}; (\boldsymbol{M}, \mathtt{tk}) \leftarrow \mathcal{M}_{gen}(p, n, \mathcal{S}, q); [\boldsymbol{g}]_\iota \leftarrow [\boldsymbol{R}]_\iota \boldsymbol{M}$. This is the variation of the above lemma that we will use in the ISH proof.

Theorem 3. *Let Pgen_{bg} be a bilinear group generator. Fix λ, n, and q. The EMP commitment scheme in group $\mathbb{G} = \mathbb{G}_\iota$ is*

1. *ISH relative to Pgen_{bg} under the $DDH_{\mathbb{G}_\iota}$ assumption: for each PPT $\mathcal{A}$, there exists a PPT $\mathcal{B}$, such that $\mathsf{Adv}^{\mathrm{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq 2\lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B},\mathbb{G}_\iota,\mathsf{Pgen}}(\lambda) + 2/p$.*
2. *F-SSE for $F = [\cdot]_\iota$ (thus, F depends on p),*
3. *AEPT,*
4. *SPB,*
5. *AEPH,*
6. *CB and CH under the $DDH_{\mathbb{G}_\iota}$ assumption.*

Proof. **(1: ISH)** Assume $\mathcal{A}$ is an ISH adversary (see page 6) that succeeds with some probability $\varepsilon^{\mathsf{ish}}_{\mathcal{A}}$. We construct the adversary $\mathcal{C}$ for the rank assumption in Definition 2. Let $\boldsymbol{R}_0 \leftarrow\!\$\, \mathcal{U}^{(1)}_{q+1,q+1}$ be a random rank-1 matrix and $\boldsymbol{R}_1 \leftarrow\!\$\, \mathcal{U}^{(q+1)}_{q+1,q+1}$ be a random full-rank (rank-$(q+1)$) matrix. Let $\beta_{\mathcal{C}} \leftarrow\!\$\, \{0,1\}$ and $[\boldsymbol{R}]_\iota \leftarrow [\boldsymbol{R}_{\beta_{\mathcal{C}}}]_\iota$. The rank adversary $\mathcal{C}(\mathsf{p}, [\boldsymbol{R}]_\iota)$ has to guess the value of $\beta_{\mathcal{C}}$.

Given $(\mathsf{p}, [\boldsymbol{R}]_\iota)$, $\mathcal{C}$ does the following:

```
$\mathcal{C}(\mathsf{p}, [\boldsymbol{R}]_\iota)$
$(\mathcal{S}_0, \mathcal{S}_1) \leftarrow \mathcal{A}(\mathsf{p}, n, q)$;
$\beta_{\mathcal{A}} \leftarrow\!\$\, \{0,1\}; (\boldsymbol{M}, \mathtt{tk}) \leftarrow \mathcal{M}_{gen}(p, n, \mathcal{S}_{\beta_{\mathcal{A}}}, q); \mathtt{ck}_{\beta_{\mathcal{A}}} \leftarrow [\boldsymbol{R}]_\iota \cdot \boldsymbol{M}$;
$\beta'_{\mathcal{A}} \leftarrow \mathcal{A}(\mathtt{ck}_{\beta_{\mathcal{A}}})$;
if $\beta'_{\mathcal{A}} = \beta_{\mathcal{A}}$ then $\beta'_{\mathcal{C}} \leftarrow 1$ else $\beta'_{\mathcal{C}} \leftarrow 0$;
return $\beta'_{\mathcal{C}}$;
```

Note that $\beta_{\mathcal{C}} = 1$ iff $\boldsymbol{R}$ is a full-rank matrix and $\beta_{\mathcal{A}} = 1$ iff one uses $\boldsymbol{R}$ to mask a matrix $\boldsymbol{M}$ formed from $\mathcal{S}_1$. For the following proof, observe that if $\beta_{\mathcal{C}} = 0$ then $\boldsymbol{R}$ masks $\boldsymbol{M}$ perfectly (except with probability $1/p$ as shown in Lemma 3) and thus $\mathcal{A}$ has no advantage in guessing $\beta_{\mathcal{A}}$.

On the other hand, if $\beta_{\mathcal{C}} = 1$ then $\mathcal{A}$ has advantage $\varepsilon^{\mathsf{ish}}_{\mathcal{A}}$.

Then,

$$\begin{aligned}
\Pr[\beta'_{\mathcal{C}} = \beta_{\mathcal{C}}] &= (\Pr[\beta'_{\mathcal{C}} = 1 | \beta_{\mathcal{C}} = 1] + \Pr[\beta'_{\mathcal{C}} = 0 | \beta_{\mathcal{C}} = 0])/2 \\
&= (\Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}} | \beta_{\mathcal{C}} = 1] + \Pr[\beta'_{\mathcal{A}} \neq \beta_{\mathcal{A}} | \beta_{\mathcal{C}} = 0])/2 \\
&= \left(\varepsilon^{\mathsf{ish}}_{\mathcal{A}} + 1 - \delta\right)/2 \ ,
\end{aligned}$$

where $\delta := \Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}} | \beta_{\mathcal{C}} = 0]$. In particular,

$$\varepsilon^{\mathsf{ish}}_{\mathcal{A}} = 2\Pr[\beta'_{\mathcal{C}} = \beta_{\mathcal{C}}] - 1 + \delta.$$

To finish the proof, we have to bound δ. For this, we define two games.

Let Game 1 be the game where $\mathcal{C}$ is defined as above and $\beta_{\mathcal{C}} = 0$. Thus, $\delta = \Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}} | \text{Game } 1]$.

In Game 2, $\mathcal{C}$ operates as above, except in the case $\beta_{\mathcal{C}} = 0$, $\mathcal{A}$ gets as input $\mathtt{ck}_{\beta_{\mathcal{A}}} = [\boldsymbol{g}]_\iota$ a random rank-1 matrix.

By Lemma 3, $|\Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}}|\text{Game 1}] - \Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}}|\text{Game 2}]| \leq 1/p$. On the other hand, $\Pr[\beta'_{\mathcal{A}} = \beta_{\mathcal{A}}|\text{Game 2}] = 1/2$. Thus, $\delta \leq 1/2 + 1/p$.

Since $\mathsf{Adv}^{\mathrm{r-rank}}_{\mathcal{C},\ell,k,1,q+1,\iota,\mathsf{Pgen}}(\lambda) = |2\Pr[\beta'_{\mathcal{C}} = \beta_{\mathcal{C}}] - 1|$, we get $\varepsilon^{\mathsf{ish}}_{\mathcal{A}} = 2\Pr[\beta'_{\mathcal{C}} = \beta_{\mathcal{C}}] - 1 + \delta \leq \mathsf{Adv}^{\mathrm{r-rank}}_{\mathcal{C},\ell,k,1,q+1,\iota,\mathsf{Pgen}}(\lambda) + \delta$. Next,

$$\begin{aligned}\mathsf{Adv}^{\mathsf{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) =&|2\varepsilon^{\mathsf{ish}}_{\mathcal{A}} - 1| \leq 2\varepsilon^{\mathsf{ish}}_{\mathcal{A}} - 1\\ &\leq 2\mathsf{Adv}^{\mathrm{r-rank}}_{\mathcal{C},\ell,k,1,q+1,\iota,\mathsf{Pgen}}(\lambda) + 2\delta - 1\\ &\leq 2\mathsf{Adv}^{\mathrm{r-rank}}_{\mathcal{C},\ell,k,1,q+1,\iota,\mathsf{Pgen}}(\lambda) + 2/p\ .\end{aligned}$$

By Theorem 1, there exists a PPT adversary $\mathcal{B}$, such that $\mathsf{Adv}^{\mathrm{ish}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) \leq 2\lceil\log_2(q+1)\rceil \cdot \mathsf{Adv}^{\mathrm{ddh}}_{\mathcal{B},\mathbb{G}_\iota,\mathsf{Pgen}}(\lambda) + 2/p$. □

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

(3: AEPT) Let $\boldsymbol{x}_0 \neq \boldsymbol{x}_1$ but $\boldsymbol{x}_{0,\mathcal{S}} = \boldsymbol{x}_{1,\mathcal{S}}$. Then $\mathsf{Com}(\mathtt{ck};\boldsymbol{x}_0;r_0) - \mathsf{Com}(\mathtt{ck};\boldsymbol{x}_1;r_1) = \boldsymbol{RM}\binom{\boldsymbol{x}_0-\boldsymbol{x}_1}{r_0-r_1} = \boldsymbol{R}\binom{\mathbf{0}_q}{\sum_{i\in[1\,..\,n]\setminus\mathcal{S}}(x_{0,i}-x_{1,i})\delta_i+(r_0-r_1)} = \mathbf{0}_{q+1}$, since from tdOpen, $r_1 = \sum_{i\in[1\,..\,n]\setminus\mathcal{S}}(x_{0,i} - x_{1,i})\delta_i + r_0$.

(4: SPB) Since $F = [\cdot]_\iota$ is injective (because the bilinear group has a prime order), this follows from Item 2 and Lemma 1.

(5: AEPH) Let $\boldsymbol{x}_0, \boldsymbol{x}_1$ be such that $\boldsymbol{x}_{0,\mathcal{S}} = \boldsymbol{x}_{1,\mathcal{S}}$. Then $\boldsymbol{M}\binom{\boldsymbol{x}_0}{r_0} = (\boldsymbol{x}^\top_{0,\mathcal{S}}, 0, \ldots, 0, r_0 + \sum_{i\in[1\,..\,n]\setminus\mathcal{S}} x_{0,i}\sigma_i)^\top$ and similarly $\boldsymbol{M}\binom{\boldsymbol{x}_1}{r_1} = ((\boldsymbol{x}_{1,\mathcal{S}})^\top, 0, \ldots, 0, r_1 + \sum_{i\in[1\,..\,n]\setminus\mathcal{S}} x_{1,i}\sigma_i)^\top$. Thus, both have first q elements equal and the last element is uniformly random. Clearly then also $\mathsf{Com}(\mathtt{ck};\boldsymbol{x}_0;r_0) = \boldsymbol{RM}\binom{\boldsymbol{x}_0}{r_0}[1]_\iota$ and $\mathsf{Com}(\mathtt{ck};\boldsymbol{x}_1;r_1) = \boldsymbol{RM}\binom{\boldsymbol{x}_1}{r_1}[1]_\iota$ are indistinguishable.

(6: CB and CH): Follows from Theorem 2, Item 1, SPB and AEPH. □

Alternative constructions. One can also construct a SSB commitment from any IND-CPA secure cryptosystem if both the message space and the randomness 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 $\boldsymbol{c}\cdot\boldsymbol{x} + \mathsf{Enc}_{\mathsf{pk}}(0;\mathsf{r}) = \mathsf{Enc}_{\mathsf{pk}}(x_i, \mathsf{r} + \sum_{i=1}^n r_i)$ for $\mathsf{r} \leftarrow\$\, \mathcal{R}$. Now, ISH follows directly from the IND-CPA security, SSB and F-SSE follow from the correctness of the cryptosystem, and AESH follows since $\mathsf{Enc}_{\mathsf{pk}}(x_i, \mathsf{r} + \sum_{i=1}^n r_i)$ only depends on x_i. However, we obtain a less efficient construction than EMP. E.g., if we instantiate with lifted 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 [AIR01] 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 A.2 for more details.

5 Functional SSB Commitments

We generalize the notion of SSB commitments 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 scheme 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 $i \in [1\,..\,q]$, i.e. if F is the scalar multiplication function in the group, $[f_i(\boldsymbol{x})]_\iota$. 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 [DGI+19] 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 $\boldsymbol{x}_{\mathcal{S}} := (f_1(\boldsymbol{x}), \ldots, f_q(\boldsymbol{x}))$.

Essentially the only difference between an SSB commitment and a functional SSB commitment is that in the former $\mathcal{S}$ is a subset of $[1 \mathinner{..} q]$ 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 5. *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* $q \in [1 \mathinner{..} n]$*, and a tuple* $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ *with* $|\mathcal{S}| \le q$*,* $\mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$ *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* $|\mathcal{S}| \le q$*,* $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$*, two messages* $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathtt{MSP}^n$*, and a randomizer* $r_0 \in \mathtt{RSP}$*,* $\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}_0, r_0, \boldsymbol{x}_1)$ *returns a randomizer* $r_1 \in \mathtt{RSP}$.

Extraction: *for* $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$*,* $\mathcal{S} = (f_1, \ldots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ *with* $1 \le |\mathcal{S}| \le q$*,* $(\mathtt{ck}, (\mathtt{ek}, \mathtt{tk})) \in \mathsf{KC}(\mathsf{p}, n, q, \mathcal{S})$*, 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 6. *An F-extractable functional SSB commitment scheme* COM *for function family $\mathcal{F}$ is* secure *if it satisfies the following security requirements.*

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

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

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

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

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

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

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): *$\forall \lambda$, $\mathsf{p} \in \mathsf{Pgen}(1^\lambda)$, $n \in \mathsf{poly}(\lambda)$, $q \in [1\,..\,n]$, $\mathcal{S} = (f_1, \dots, f_{|\mathcal{S}|}) \subseteq \mathcal{F}$ 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 $\mathsf{Adv}^{\mathrm{sse}}_{\mathcal{A},F,\mathsf{COM},n,q}(\lambda) :=$*

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

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 $\mathsf{Adv}^{\mathrm{aest}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda)(\lambda) =$*

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

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 $\mathsf{Adv}^{\mathrm{cb}}_{\mathcal{A},\mathsf{COM},n,q}(\lambda) :=$*

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

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

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

Relations that hold between properties of SSB commitments also hold for functional SSB commitments; the proofs are very similar.

```
$\mathcal{M}^{\mathsf{FSSB}}_{gen}(\mathsf{p}, n, q, \boldsymbol{M} \in \mathbb{Z}_p^{q \times n})$
$\boldsymbol{\varrho} \leftarrow\!\$\, \mathbb{Z}_p^n; \mathtt{tk} \leftarrow \boldsymbol{\varrho}$;
Set $\boldsymbol{M}' \leftarrow \left( \begin{smallmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{\varrho}^\intercal & 1 \end{smallmatrix} \right) \in \mathbb{Z}_p^{(q+1) \times (n+1)}$;
return $(\boldsymbol{M}', \mathtt{tk})$;
```

```
$\mathcal{D}^{\mathsf{FSSB}}_{gen}(p, n, \mathfrak{r}, \mathcal{S}, q)$
$\boldsymbol{R} \leftarrow\!\$\, \mathcal{U}^{(\mathfrak{r})}_{q+1,q+1}$;
$(\boldsymbol{M}', \mathtt{tk}) \leftarrow \mathcal{M}^{\mathsf{FSSB}}_{gen}(p, n, \mathcal{S}, q)$;
$\boldsymbol{g} \leftarrow \boldsymbol{R}\boldsymbol{M}'$;    // $\boldsymbol{g} \in \mathbb{Z}_p^{(q+1) \times (n+1)}$;
return $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk})$;
```

Fig. 3. Generating $\boldsymbol{M}'$ from function set $\mathcal{S} \subseteq \mathcal{F}$ (the left hand side) and the discrete logarithm of the commitment key with an associated extraction key $\boldsymbol{R}$ and trapdoor $\mathtt{tk}$ (the right hand side)

5.1 Linear EMP

We construct a functional SSB commitment for a family of linear functions. Our construction follows the ideas in [DGP+19] 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 $\mathtt{CSP} = \mathbb{G}_\iota^{q+1}$;
    Set $(\boldsymbol{g}, \boldsymbol{R}, \mathtt{tk}) \leftarrow \mathcal{D}_{gen}^{\mathsf{FSSB}}(p, n, \mathfrak{r}, \mathcal{S}, q)$ where $\mathfrak{r} = q + 1$;
    Set $\mathtt{ck} \leftarrow [\boldsymbol{g}]_\iota \in \mathbb{G}_\iota^{(q+1)\times(n+1)}, \mathtt{td} \leftarrow (\mathtt{ek} \leftarrow \boldsymbol{R}^{-1}, \mathtt{tk})$;
    return $(\mathtt{ck}, \mathtt{td})$;

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

$\mathsf{tdOpen}(\mathsf{p}, \mathtt{tk}; \boldsymbol{x}_0, r_0, \boldsymbol{x}_1)$    // $[\boldsymbol{M}]_\iota \boldsymbol{x}_0 = [\boldsymbol{M}]_\iota \boldsymbol{x}_1$
    return $r_1 \leftarrow \sum_{i\in[1\,..\,n]} (x_{0,i} - x_{1,i})\mathtt{tk}_i + r_0$;

$\mathsf{Ext}(\mathsf{p}, \mathtt{ek}; [\boldsymbol{c}]_\iota)$
    return $\mathtt{ek}[\boldsymbol{c}]_\iota$ without the last component;
```

Fig. 4. 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. (For the FSH proof to work, $\boldsymbol{M}$ needs to be available as an integer matrix; this is fine in all our applications.) 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 $\boldsymbol{M} = (\boldsymbol{e}_{i_1}|\dots|\boldsymbol{e}_{i_q})^\top$ where $\boldsymbol{e}_{i_j} \in \mathbb{Z}_p^n$ is the i_jth unit vector, then $[\boldsymbol{M}\boldsymbol{x}]_\iota = [x_{i_1}, \dots, x_{i_q}]_\iota^\top$. A detailed construction is given in Fig. 4.

We want to note that the matrix $\boldsymbol{M}$ is extended into one row to place the randomness vector $\boldsymbol{\varrho}$ and one column to place the randomizer of the commitment, r, to perfectly hide the secret vector $\boldsymbol{x}$ when we extract (see Fig. 3). Concretely, in the extraction phase we obtain $\left(\begin{smallmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{\varrho}^\intercal & 1 \end{smallmatrix}\right) \cdot \left[\begin{smallmatrix} \boldsymbol{x} \\ r \end{smallmatrix}\right]_\iota = \left[\begin{smallmatrix} \boldsymbol{M}\boldsymbol{x} \\ \boldsymbol{\varrho}^\intercal \boldsymbol{x} + r \end{smallmatrix}\right]_\iota$ from multiplying the commitment by the inverse matrix of $\boldsymbol{R}$. The first q rows contain the functions of $\boldsymbol{x}$ in the group that we want and the last component contains a combination of $\boldsymbol{x}$ with $\boldsymbol{\varrho}$ that is completely masked by r.

Moreover, if we take an ACS (Definition 3), 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 $(0, \dots, 0, 1)^T$ possibly inserted in between).

5.2 Security proofs

We start by proving an analogue for Lemma 3.

Lemma 4. *Let $q \le n$ and $\lambda \in \mathbb{N}$. For any (even unbounded) adversary $\mathcal{A}$,*

$$|\Pr[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \boldsymbol{M} \leftarrow \mathcal{A}(\mathsf{p}, n, q); \boldsymbol{g} \leftarrow \mathcal{D}_{gen}^{\mathsf{FSSB}}(\mathsf{p}, n, 1, \boldsymbol{M}, q) : \mathcal{A}(\boldsymbol{g}) = 1]$$

$$- \Pr[\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda); \boldsymbol{M} \leftarrow \mathcal{A}(\mathsf{p}, n, q); \boldsymbol{g} \leftarrow \mathcal{U}_{q+1,n+1}^{(1)} : \mathcal{A}(\boldsymbol{g}) = 1]| \le 1/p.$$

Proof. The proof is very similar to the one in Lemma 3. We can sample uniformly random rank 1 matrix $\boldsymbol{R}$ by sampling $\boldsymbol{a}, \boldsymbol{b} \leftarrow\!\!\$\, \mathbb{Z}_p^{q+1} \setminus \{\boldsymbol{0}\}$ and computing $\boldsymbol{R} = \boldsymbol{a}^\top \otimes \boldsymbol{b}$. Then, $\boldsymbol{g} = \boldsymbol{R}\boldsymbol{M}' = (\boldsymbol{a}^\top \otimes \boldsymbol{b})\boldsymbol{M}' = (\boldsymbol{a}^\top \boldsymbol{M}') \otimes \boldsymbol{b}$. Let us define $\boldsymbol{c} = \boldsymbol{a}^\top \boldsymbol{M}'$.

For $i = 1, \dots, n$, we have $c_i = (\sum_{j=1}^q a_j M_{j,i}) + a_{q+1}\rho_i$ and $c_{n+1} = a_{q+1}$. Thus, $\boldsymbol{c}$ is uniformly random in $(\mathbb{Z}_p \setminus \{\boldsymbol{0}\})^{q+1}$ exactly when $a_{q+1} \neq 0$. The probability that $a_{q+1} = 0$ is $(p^q - 1)/(p^{q+1} - 1) \le 1/p$.

Hence, follows the claim in the lemma's statement. □

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

1. *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 $\mathsf{Adv}_{\mathcal{A},\mathsf{COM},n,q}^{\mathrm{fsh}}(\lambda) \le \lceil \log_2(q+1) \rceil \cdot \mathsf{Adv}_{\mathcal{B},\mathbb{G}_\iota,\mathsf{Pgen}}^{\mathrm{ddh}}(\lambda)$.*

2. *F-SSE for $F = [\cdot]_\iota$ (thus, F depends on p),*
3. *SPB,*
4. *AEPH,*
5. *AEPT,*
6. *CB and CH under the $\mathrm{DDH}_{\mathbb{G}_\iota}$ assumption.*

Proof. **(1: FSH)** The proof is almost identical to the ISH proof in Theorem 3. The only significant difference is that since $\boldsymbol{M}$ has a different distribution, we use Lemma 4 instead of Lemma 3.

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

(3: SPB) Clearly, there are no $\boldsymbol{x}_0, \boldsymbol{x}_1 \in \mathbb{Z}_p^n$ such that $\boldsymbol{M}\boldsymbol{x}_0 \neq \boldsymbol{M}\boldsymbol{x}_1$ and $[\boldsymbol{c}]_\iota := \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_0; r_0) = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_1; r_1)$ since by the F-SSE property we have that $\mathsf{Ext}(\mathsf{p}, \mathsf{ek} = \boldsymbol{R}^{-1}; [\boldsymbol{c}]_\iota) = [\boldsymbol{M}\boldsymbol{x}_0]_\iota = [\boldsymbol{M}\boldsymbol{x}_1]_\iota$.

(4: 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 $\boldsymbol{M}' = \left(\begin{smallmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{\varrho}^\top & 1 \end{smallmatrix}\right)$, $\boldsymbol{R} \in \mathbb{Z}_p^{(q+1)(q+1)}$ is some full rank matrix, and $\boldsymbol{\varrho} \in \mathbb{Z}_p^n$, and finally outputs $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ such that $\boldsymbol{M}\boldsymbol{x}_0 = \boldsymbol{M}\boldsymbol{x}_1$.

Let us analyze distributions of $C_0 = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_0;_0)$ and $C_1 = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_1; r_1)$ for a uniformly random r_0, r_1. For $\beta \in \{0, 1\}$, we can define $[\boldsymbol{u}_\beta]_\iota := [\boldsymbol{M'}(\begin{smallmatrix} \boldsymbol{x}_\beta \\ r_\beta \end{smallmatrix})]_\iota = \left[\begin{smallmatrix} \boldsymbol{M}\boldsymbol{x}_\beta \\ \boldsymbol{\varrho}^\top \boldsymbol{x}_\beta + r_\beta \end{smallmatrix}\right]$. 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 $C_\beta = \mathsf{Com}(\mathtt{ck}; \boldsymbol{x}_\beta; r_\beta) = \boldsymbol{R}[\boldsymbol{u}_\beta]_\iota$, then also C_1 and C_2 are indistinguishable.

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

(6: CB and CH) Follows directly from the analog of Theorem 2. □

6 Applications of Functional SSB Commitments

In the rest of the paper, we present three applications of functional SSB commitments. In Section 6.1 we have two straightforward applications for FSSB: Oblivious Database Queries (ODQ) and Oblivious Linear Function Evaluation (OLE) [DKM12, GNN17, DGN+17]. 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 7 we present a new QA-NIZK argument for SAP relations that uses FSSB as a technical tool in the security proof.

6.1 ODQ & OLE

A very straight-forward application of FSSB 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 $i \in [1 \mathinner{..} q]$ 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 $\boldsymbol{x}' = (\begin{smallmatrix} \boldsymbol{x} \\ \boldsymbol{1}_q \end{smallmatrix})$, 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 $[\boldsymbol{M} \cdot \boldsymbol{x}']_\iota$ from COM using the Ext algorithm with $\mathtt{ek}$.

Privacy and Correctness. We follow privacy and correctness definitions proposed by Döttling et al. [DGI+19] (see Section 5.1 of their paper for full definitions). From the SSE property we know that the receiver can recover $[\boldsymbol{M}\left(\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{1}_q\end{smallmatrix}\right)]_\iota = [\boldsymbol{A}\boldsymbol{x}+\boldsymbol{b}]_\iota$ 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 $[\boldsymbol{M}\boldsymbol{x}']_\iota$ to the simulator. Simulatability is slightly stronger than the AEPH property but still holds for FSSB. 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 $[\boldsymbol{R}\left(\begin{smallmatrix}\boldsymbol{M} & \boldsymbol{0}\\ \boldsymbol{r}^\top & 1\end{smallmatrix}\right)]_\iota \left(\begin{smallmatrix}\boldsymbol{x}'\\ r\end{smallmatrix}\right) = \boldsymbol{R}\left[\begin{smallmatrix}\boldsymbol{M}\boldsymbol{x}'\\ \boldsymbol{x}'\boldsymbol{r}^\top+r\end{smallmatrix}\right]_\iota$ and therefore we can simulate it by sampling $r^* \leftarrow\!\!\$\, \mathbb{Z}_p$ and computing $\boldsymbol{R}\left(\begin{smallmatrix}[\boldsymbol{M}\boldsymbol{x}']_\iota\\ r^*\end{smallmatrix}\right)$. 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 $|[\boldsymbol{M}\boldsymbol{x}']_\iota|/(|\mathtt{ck}| + |\mathsf{COM}|) = q/((n+q+2)(q+1))$. However, we achieve very good download rate $|[\boldsymbol{M}\boldsymbol{x}']_\iota|/|\mathsf{COM}| = q/(q+1)$ which tends to 1. This is similar to Döttling et al. [DGI+19] 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 $1/(2n+4)$.

OT. SSB commitments can 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 A.2 for more details.

7 QA-NIZK Argument for Quadratic Equations

We present a QA-NIZK argument that uses FSSB commitments as a crucial technical tool in the security proof. It is inspired by Daza et al. [DGP+19], who presented a commit-and-prove QA-NIZK argument for Square Span Programs (SSP, [DFGK14]) that can be used to encode the Boolean circuit satisfiability language. Their construction implicitly uses FSSB commitments without explicitly formalizing it. Our QA-NIZK is for Square Arithmetic Programs (SAP) [GM17] which can be used to encode the arithmetic circuit satisfiability language, has roughly the same complexity as the argument in [DGP+19] and follows a similar overall strategy. However, we use FSSB 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 $([\cdot]_1, [\cdot]_2)$-extractable) commitments $[\boldsymbol{c}]_1$ and $[\boldsymbol{c}']_2$ of the SAP witness.[8] Note that the commitment is only computed once but can be reused for many different SAP relations. For simplicity, we use ElGamal encryption in both groups (with different public keys) in this role, and the commitment key $\mathtt{ck} = (\mathsf{pk}, \mathsf{pk}')$ is the 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 FSSB commitment $[\tilde{\boldsymbol{c}}]_2$ that commits to the SAP witness and the randomness of the SNARK,

[8] Daza et al. [DGP+19] needs an extractable commitment only in $\mathbb{G}_1$ since the constraints of an $[\cdot]_1$-extracted SSP witness can be verified without pairings (e.g, $[\boldsymbol{a}]_1^\top \boldsymbol{v}_j + b_j[1]_1 \in [0,2]_1$) whereas SAP constraints are quadratic and require pairings for verification (e.g., $([\boldsymbol{a}]_1^\top \boldsymbol{v}_j)([\boldsymbol{a}]_2^\top \boldsymbol{v}_j) - ([\boldsymbol{a}]_1^\top \boldsymbol{w}_j)[1]_2 = [0]_T$). Prior versions of this paper incorrectly contained the commitment only in $\mathbb{G}_1$.

- a succinct linear subspace argument bls [GHR15] that shows that commitments open to consistent values (see bls argument below). I.e., it guarantees that the opening of $[\boldsymbol{c}]_1$ is also used in the SNARK-type elements $[V, H, W]_1$, $[V]_2$, in linear-length commitment $[\boldsymbol{c}']_2$, and in FSSB commitment $[\tilde{\boldsymbol{c}}]_2$.

Below, we go over some of the technical background and then finally present our QA-NIZK argument for SAP.

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

Square Arithmetic Program (SAP). A square arithmetic program [DFGK14, GM17] 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

$$\mathscr{L}_{\mathtt{SAP},\mathtt{ck}} = \left\{ ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2) \in \mathbb{G}_1^{2n} \times \mathbb{G}_2^{2n} \;\middle|\; \begin{array}{c} \exists \boldsymbol{a}, \boldsymbol{r}, \boldsymbol{r}' \in \mathbb{Z}_p^n : \\ \left\{(\boldsymbol{a}^\top \boldsymbol{v}_j)^2 - \boldsymbol{a}^\top \boldsymbol{w}_j = 0\right\}_{j=1}^{d} \wedge \\ ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2) = \mathsf{Com}_{ck}(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{r}') \end{array} \right\}$$

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 $\boldsymbol{W}$. Following [DGP+19], our SAP instance does not take a public input. For public inputs one can choose the commitment randomness in corresponding positions to be 0.

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

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

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 satisfied 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 $\Pi_{\mathsf{bls}} = (\mathsf{K}_{\mathsf{bls}}, \mathsf{P}_{\mathsf{bls}}, \mathsf{V}_{\mathsf{bls}})$ for membership in linear spaces defined in [GHR15] 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. Π_{bls} 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 [GHR15] for more details. We leave it as an open question if the slightly more efficient construction by Ràfols and Silva [RS20] can be used: it has proof size of three group elements, but it is not known to be strongly sound.

New target assumption. The q-target strong Diffie-Hellman (q-TSDH) assumption [BB04] says that given $\{[s^i]_{1,2}\}_{i=1}^q$ for a random s, it is computationally output $(r, [\nu]_T = [1/(s-r)]_T)$ for some $r \in \mathbb{Z}_p$. We generalize q-TSDH by saying that it is hard to compute $(r, [\nu]_T = [c/(s-r)]_T)$ where $r \in \mathbb{Z}_p$ and c is a constant independent of s. 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 7 (q-SATSDH)**.** *The q-Square Arithmetic Target Strong Diffie-Hellman assumption holds relative to* Pgen*, if* $\forall$ *PPT adversaries* $\mathcal{A}$*,*

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

where $\mathsf{Adv}_{\mathcal{A},q}^{\mathsf{satsdh}}(\lambda) \approx_\lambda 0$.

SATSDH is similar to yet different from the GSDH, STSDH, and QTSDH assumptions defined by Daza et al. [DGP+19]. Hence, in Appendix C, we show that SATSDH is a falsifiable assumption and that assuming a particular (previously known) knowledge assumption, SATSDH and TSDH are equivalent. Alternatively, one can prove that SATSDH is secure in the algebraic group model.

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

- $\mathsf{K}_0(\lambda)$ returns $\mathsf{p} \leftarrow \mathsf{Pgen}(1^\lambda)$.
- $\mathcal{D}_\mathsf{p}(n, d)$ returns a commitment key $\rho = \mathtt{ck} = ([\boldsymbol{u}]_1, [\boldsymbol{u}']_2) = ([1, u]_1^\top, [1, u']_2^\top)$ where $u, u' \leftarrow\!\!\$\, \mathbb{Z}_p$.
- $\mathsf{K}_1(\mathsf{p}, n, d, \mathtt{ck})$ picks $s \leftarrow\!\!\$\, \mathbb{Z}_p$ such that $t(s) \neq 0$, then sets $q_v = 4$ (the reason behind this choice becomes clear in the soundness proof), $n' = n+1$, $\boldsymbol{M} = \mathbf{0} \in \mathbb{Z}_p^{q_v \times n'}$ (i.e., $S_v = \emptyset$) and generates a FSSB key $(\widetilde{\mathtt{ck}} \in \mathbb{G}_2^{(q_v+1)\times(n'+1)}, \widetilde{\mathtt{td}}) = ([\mathbf{K}]_2, \widetilde{\mathtt{td}}) \leftarrow \mathsf{KC}_2(\mathsf{p}, n', q_v, \boldsymbol{M})$. Finally, it runs $(\mathsf{crs_{bls}}, \mathsf{td_{bls}}) \leftarrow \mathsf{K_{bls}}([\boldsymbol{N}_1]_1 \in \mathbb{G}_1^{(2n+2)\times(3n+3)}, [\boldsymbol{N}_2]_2 \in \mathbb{G}_2^{(2n+q_v+2)\times(3n+3)})$ for

$$[\mathbf{N}_1]_1 = \left[\begin{array}{ccc|ccc|c|ccc} \boldsymbol{e}_2 & & & \boldsymbol{u} & & & & & & \\ & \ddots & & & \ddots & & \mathbf{0}_{2n\times n} & & \mathbf{0}_{2n\times 3} & \\ & & \boldsymbol{e}_2 & & & \boldsymbol{u} & & & & \\ \hline v_1(s) & \dots & v_n(s) & & & & & t(s) & 0 & 0 \\ w_1(s) & \dots & w_n(s) & & \mathbf{0}_{2\times n} & & \mathbf{0}_{2\times n} & 0 & t(s) & 0 \end{array}\right]_1,$$

$$[\mathbf{N}_2]_2 = \left[\begin{array}{ccc|c|ccc|ccc} \boldsymbol{e}_2 & & & & \boldsymbol{u}' & & & & & \\ & \ddots & & \mathbf{0}_{2n\times n} & & \ddots & & & \mathbf{0}_{2n\times 3} & \\ & & \boldsymbol{e}_2 & & & & \boldsymbol{u}' & & & \\ \hline v_1(s) & \dots & v_n(s) & & & & & t(s) & 0 & 0 \\ \mathbf{K}^{(1)} & \dots & \mathbf{K}^{(n)} & \mathbf{0}_{2\times n} & & \mathbf{0}_{2\times n} & & \mathbf{K}^{(n+1)} & 0 & \mathbf{K}^{(n+2)} \end{array}\right]_2.$$

  Return the CRS $\mathtt{crs} = (\mathsf{p}, \mathtt{ck}, \widetilde{\mathtt{ck}}, \{[s^i]_{1,2}\}_{i=1}^d, \mathsf{crs_{bls}})$ with the trapdoor $\mathsf{td_{bls}}$.
- The prover P receives an input $(\mathtt{crs}, ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2), (\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{r}'))$. 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:

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

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

- The verifier V with input $(\mathtt{crs}, ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2), \pi)$ returns 1 iff
  1. $[\boldsymbol{c}]_1, [\boldsymbol{c}']_2, \pi$ can be parsed as group elements as expected from an honest statement and proof,
  2. $[V]_1[V]_2 - [W]_1[1]_2 = [H]_1[t(s)]_2$, and
  3. $\mathsf{V_{bls}}(\mathsf{crs_{bls}}, \mathsf{x_{bls}}, \psi) = 1$.

Efficiency. The proof size in the original construction in [DGP+19] 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 $\mathbb{G}_2$. The minor loss in efficiency is expected by directly handling a more powerful language.

SSB functionality in the security proof. The security proof of Π_{SAP} uses similar techniques as [DGP+19], but it is conceptually simpler 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 either public elements or the challenge of some falsifiable assumption. The actual reduction requires us to extract multiple such linear combinations.

If an adversary wins the soundness game, its argument passes verification but either the commitments in $\mathbb{G}_1$ and $\mathbb{G}_2$ open to different vectors or at least one SAP equation does not hold. In the security proof, we first change the soundness game such that it the adversary fails if the commitments open to differenct values. This will only negligibly change the winning advantage due to the BLS argument. In the next game we randomly pick one of the SAP constraints $\left(\boldsymbol{a}^\top \boldsymbol{v}_{j^*}\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_{j^*}$ by sampling $j^* \leftarrow\!\!\$\, [1\,..\,d]$. If the adversary cheats, then this constraint does not hold with probability $1/d$. Our aim is to break the d-SATSDH assumption when the j^*-th constraint does not hold.

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 of the form $f(a_1, \ldots, a_n) = \sum_i a_i [y_i]_2$ that we need. Here, $[\boldsymbol{y}]_2$ depends on the security assumption and on the j^*-th equation. For the reduction to FSH to work, we need to be able to efficiently verify that the soundness was broken. The latter can be done by extracting $[\boldsymbol{a}]_1$ and $[\boldsymbol{a}']_2$ respectively from $[\boldsymbol{c}]_1$ and $[\boldsymbol{c}']_2$ and testing if $[\boldsymbol{a}]_1[1]_2 = [1]_1[\boldsymbol{a}']_2$ and $([\boldsymbol{a}]_1^\top \boldsymbol{v}_j)([\boldsymbol{a}]_2^\top \boldsymbol{v}_j) - ([\boldsymbol{a}]_1^\top \boldsymbol{w}_j)[1]_2 = [0]_T$ for all $j = 1, \ldots, d$.

The *FSH* property guarantees that the adversary's cheating probability in the j^*-th equation differs only negligibly when the function f is encoded inside $\widetilde{\mathtt{ck}}$. That is, when the adversary cheats, the SAP equation is still not satisfied (almost) with probability $\geq 1/d$. The $[\cdot]_2$-*SSE* property[9] 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.

Reduction to d-SATSDH. Let us now discuss in more detail what exactly we need to extract with FSSB (the functions f from above) to break the d-SATSDH assumption. By the characterization of the SAP, if the j^*-th equation does not hold, then $(X - \chi_{j^*}) \nmid P(X)$. Let $q_v(X), q_w(X) \in \mathbb{Z}_p[X]$ and $\beta_v, \beta_w \in \mathbb{Z}_p$ be the unique elements such that

$$V(X) = q_v(X) \cdot (X - \chi_{j^*}) + \beta_v,$$

$$W(X) = q_w(X) \cdot (X - \chi_{j^*}) + \beta_w.$$

Then we can express the division of $P(X) = V(X)^2 - W(X)$ by $X - \chi_{j^*}$ as follows.

[9] The security of Π_{SAP} relies crucially on the $[\cdot]_2$-SSE property, while we do not explicitly need the (weaker) SSB property by itself.

Lemma 5. *$P(X) = (q_v(X)\,(V(X) + \beta_v) - q_w(X)) \cdot (X - \chi_{j^*}) + \big(\beta_v^2 - \beta_w\big)$.*

Proof. The proof follows straightforwardly:

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

This concludes the proof. □

Since, $(X - \chi_{j^*}) \nmid P(X)$ we get that $(\beta_v^2 - \beta_w) \neq 0$ and thus $\beta_v^2 \neq \beta_w$. This is one of the requirements in the SATSDH assumption.

Let us further define unique elements $\alpha_i(X)$, $\hat{\alpha}_i(X) \in \mathbb{Z}_p[X]$, and $\beta_{v,i}, \beta_{w,i} \in \mathbb{Z}_p$, such that

$$v_i(X) = \alpha_i(X)(X - \chi_{j^*}) + \beta_{v,i}, \tag{3}$$

$$w_i(X) = \hat{\alpha}_i(X)(X - \chi_{j^*}) + \beta_{w,i}, \tag{4}$$

for $i = 1, \ldots, n$. Moreover, let us define $\alpha_t(X)$ and β_t such that

$$t(X) = \alpha_t(X)(X - \chi_{j^*}) + \beta_t, \tag{5}$$

but note that $\beta_t = 0$ since $t(X)$ is divisible by $(X - \chi_{j^*})$.

Lemma 6. *The following relations hold:*

$$q_v(X) = \sum_{i=1}^{n} a_i\alpha_i(X) + \delta_v\alpha_t(X), \quad \beta_v = \sum_{i=1}^{n} a_i\beta_{v,i}, \tag{6}$$

$$q_w(X) = \sum_{i=1}^{n} a_i\hat{\alpha}_i(X) + \delta_w\beta_t(X), \quad \beta_w = \sum_{i=1}^{n} a_i\beta_{w,i}. \tag{7}$$

Proof. On the one hand, we defined $V(X) = q_v(X) \cdot (X - \chi_{j^*}) + \beta_v$. If we now also consider Eq. (3) and Eq. (5), then

$$\begin{aligned} V(X) =& \sum_{i=1}^{n} a_i v_i(X) + \delta_v t(X) \\ =& \sum_{i=1}^{n} a_i\big(\alpha_i(X)(X - \chi_{j^*}) + \beta_{v,i}\big) + \delta_v\big(\alpha_t(X)(X - \chi_{j^*}) + \beta_t\big) \\ =& \left(\sum_{i=1}^{n} a_i\alpha_i(X) + \delta_v\alpha_t(X)\right) \cdot (X - \chi_{j^*}) + \left(\sum_{i=1}^{n} a_i\beta_{v,i} + \delta_v\beta_t\right) . \end{aligned}$$

Given that $\beta_t = 0$, we get $q_v(X) = \sum_{i=1}^{n} a_i\alpha_i(X) + \delta_v\alpha_t(X)$ and $\beta_v = \sum_{i=1}^{n} a_i\beta_{v,i}$. This proves Eq. (6). The proof for Eq. (7) is similar. □

The SATSDH reduction extracts the following functions of the witness $\boldsymbol{a}$ and δ_v, δ_w: $[q_v(s)]_2 = [\sum_{i=1}^{n} a_i\alpha_i(s) + \delta_v\alpha_t(s)]_2$, $[q_w(s)]_2 = [\sum_{i=1}^{n} a_i\hat{\alpha}_i(s) + \delta_w\alpha_t(s)]_2$, $[\beta_v z]_2 = [\sum_{i=1}^{n} a_i z\beta_{v,i}]_2$, and $[\beta_w z]_2 = [\sum_{i=1}^{n} a_i z\beta_{w,i}]_2$, where $z, s \in \mathbb{Z}_p$ are secrets of the SATSDH assumption. The idea is that we can break the d-SATSDH assumption by computing $[\beta_v]_1 = \sum_{i=1}^{n} \beta_{v,i}[a_i]_1$ (note that $[a_i]_1$ are extractable from the PB commitment), $[\beta_w]_1 = \sum_{i=1}^{n} \beta_{w,i}[a_i]_1$ and moreover by Lemma 5, $\left[\frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T = \left[\frac{P(s)}{s - \chi_{j^*}}\right]_T - ([V]_1 + [\beta_v]_1)[q_v(s)]_2 + [1]_1[q_w(s)]_2$, where $[\frac{P(s)}{s - \chi_{j^*}}]_T = [H]_1[\prod_{i \neq j^*}(s - \chi_i)]_2$ 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 7 for more details.

Proofs of security. We prove completeness, soundness, and zero-knowledge of the new QA-NIZK construction.

Theorem 5. *Π_{SAP} has quasi-adaptive completeness.*

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

Theorem 6. *Π_{SAP} has perfect quasi-adaptive zero-knowledge.*

Proof. 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 $\mathbf{0}_{n+1}$. Next, V and W are uniformly random and independently distributed in the honest proof. This is the case since s is chosen such that $t(s) \neq 0$. 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 [GHR15]) using the trapdoor $\mathsf{td}_{\mathsf{bls}}$. □

Remark 2. Note that the perfect zero-knowledge proof in [DGP⁺19] does not hold since $t(s) = 0$ is possible. In this case we would get $[V(s)]_1 = [\sum_{i=1}^{n} a_i v_i(s) + \delta_v t(s)]_1 = [\sum_{i=1}^{n} a_i v_i(s)]_1$, which may leak information about the witness. The argument of [DGP⁺19] still achieves statistical zero-knowledge since $t(X)$ has only polynomially many roots. In Π_{SAP}, we explicitly sample non-roots of $t(X)$ and thus obtain perfect zero-knowledge. A similar change would also make [DGP⁺19] argument perfectly zero-knowledge.

Theorem 7. *Let $\mathsf{Adv}_{snd}(\mathcal{A})$ be the advantage of any non-uniform PPT adversary $\mathcal{A}$ against the computational quasi-adaptive soundness of Π_{SAP}. Then, there exists non-uniform PPT adversaries $\mathcal{B}_0$ and $\mathcal{B}_2$ against strong soundness of the BLS argument, $\mathcal{B}_1$ against the FSH, and $\mathcal{B}_3$ against the d-*SATSDH *assumption such that,*

$$\mathsf{Adv}^{\mathsf{snd}}_{\mathcal{A},\Pi_{\mathsf{SAP}}}(\lambda) \le \mathsf{Adv}^{\mathsf{snd}}_{\mathcal{B}_0,\Pi_{\mathsf{bls}}}(\lambda) + d\big(\mathsf{Adv}^{\mathsf{fsh}}_{\mathsf{COM},n',q_v,\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathsf{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathsf{satsdh}}_{\mathcal{B}_3,d}(\lambda)\big).$$

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

- Real: This is the original quasi-adaptive soundness game (see Section 2.2). The output is 1 if the adversary produces a false accepting proof.
  In particular, it means that if $[\boldsymbol{c}]_1$ opens to $[\boldsymbol{a}]_1$ and $[\boldsymbol{c}']_2$ opens to $[\boldsymbol{a}']_2$, then either
  - $\boldsymbol{a} \neq \boldsymbol{a}'$, or
  - $(\boldsymbol{a}^\top \boldsymbol{v}_i) \cdot (\boldsymbol{a}^\top \boldsymbol{v}_i) - \boldsymbol{a}^\top \boldsymbol{w}_i \neq 0$ for some $i = 1, \ldots, d$.

  (Since the commitments are perfectly binding, $\boldsymbol{a}$ and $\boldsymbol{a}'$ are uniquely determined by $[\boldsymbol{c}]_1$ and $[\boldsymbol{c}']_2$.)
- Game_0: This game is identical to the previous one, except for the following differences. Instead of generating the commitment key as $\mathtt{ck} \leftarrow \mathcal{D}_{\mathsf{p}}(n, d)$, the game samples $u, u' \leftarrow\!\!\$\, \mathbb{Z}_p$ itself, sets $\mathtt{ck} = ([1, u]_1^\top, [1, u']_2^\top)$, and stores (u, u'). When $\mathcal{A}$ outputs $x = ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2), \pi$ accepted by the verifier, the game will extract $[\boldsymbol{a}]_1$ from $[\boldsymbol{c}]_1$ with the secret key u and $[\boldsymbol{a}']_2$ from $[\boldsymbol{c}']_1$ with the secret key u'. Most importantly, we also change the winning condition of this game. In addition to the standard soundness winning condition $\big(\mathsf{V}(\mathtt{crs}, x, \pi) = 1$ and $\neg(\exists w : \mathcal{R}_\rho(x, w))\big)$, we require that $\boldsymbol{a} = \boldsymbol{a}'$.
- Game_1: This game is identical to the previous one except that the game picks $j^* \leftarrow\!\!\$\, [1 \mathinner{..} d]$ and aborts if $\boldsymbol{a}$ satisfies the j^*-th equation. That is, $\mathcal{A}$ wins if in addition to the standard soundness winning condition, $\boldsymbol{a} = \boldsymbol{a}'$ and $\left(\boldsymbol{a}^\top \boldsymbol{v}_{j^*}\right)^2 - \boldsymbol{a}^\top \boldsymbol{w}_{j^*} \neq 0$.
- Game_2: This game is the same as the previous one except that we change the commitment key $\widetilde{\mathtt{ck}}$ by using a different matrix $\mathbf{M} \neq \mathbf{0}$ during its generation.
  For each $i \in [1 \mathinner{..} n]$, 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 $z \leftarrow\!\!\$\, \mathbb{Z}_{\mathsf{p}}$, which is going to be part of the SATSDH challenge, and change the EMP commitment key $\widetilde{\mathtt{ck}}$ by setting

$$\mathbf{M} = \begin{pmatrix} \alpha_1(s) \dots \alpha_n(s) & \alpha_t(s) \\ \hat{\alpha}_1(s) \dots \hat{\alpha}_n(s) & \alpha_t(s) \\ \beta_{v,1}z \dots \beta_{v,n}z & 0 \\ \beta_{w,1}z \dots \beta_{w,n}z & 0 \end{pmatrix} .$$

Let us now analyze the games.

Lemma 7. *There exists a non-uniform PPT adversary* $\mathcal{B}_0$*, such that* $\Pr[\mathsf{Real}(\mathcal{A}) = 1] \leq \Pr[\mathsf{Game}_0(\mathcal{A}) = 1] + \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_0, \Pi_{\mathsf{bls}}}(\lambda)$.

Proof. Let us start by observing that

$$\begin{aligned}\Pr[\mathsf{Real}(\mathcal{A}) = 1] =& \Pr[(\mathsf{Real}(\mathcal{A}) = 1 \wedge \boldsymbol{a} = \boldsymbol{a}') \vee (\mathsf{Real}(\mathcal{A}) = 1 \wedge \boldsymbol{a} \neq \boldsymbol{a}')] \\ =& \Pr[\mathsf{Game}_0(\mathcal{A}) = 1] + \Pr[\mathsf{Real}(\mathcal{A}) = 1 \wedge \boldsymbol{a} \neq \boldsymbol{a}'] \ .\end{aligned}$$

The probability of the latter event can be reduced to soundness of Π_{bls}. We will construct a Π_{bls} soundness adversary $\mathcal{B}_0$ to show this. Let $\rho_{\mathsf{bls}} = ([\mathbf{N}_1]_1, [\mathbf{N}_2]_2)$ be sampled as in the CRS of Π_{SAP} and let $\omega_{\rho_{\mathsf{bls}}} = (u, u', s, \mathbf{K})$ be the parameter witness of ρ_{bls}. Recall that since Π_{bls} is strongly quasi-adaptively sound, the adversary is allowed to get $\omega_{\rho_{\mathsf{bls}}}$ as an input.

Suppose that the adversary $\mathcal{B}_0$ gets as an input $(\mathsf{p}, \mathtt{crs}_{\mathsf{bls}}, \omega_{\rho_{\mathsf{bls}}})$. Since $\mathcal{B}_0$ knows $\omega_{\rho_{\mathsf{bls}}}$, it can reconstruct the CRS of Π_{SAP},

$$\mathtt{crs} = (\mathsf{p}, \mathsf{ck} = \left([1,u]_1^\top, [1,u']_2^\top\right), \widetilde{\mathtt{ck}} = [\mathbf{K}]_2, \{\left[s^i\right]_{1,2}\}_{i=1}^{d}, \mathsf{crs}_{\mathsf{bls}}).$$

Next, it runs $\mathcal{A}(\mathsf{p}, \mathtt{crs})$ to obtain $x = ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2)$ and π, and extracts from it the statement $\mathtt{x}_{\mathsf{bls}} := ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2, [V]_1, [W]_1, [V]_2, [\tilde{\boldsymbol{c}}]_2)^\top$ and the BLS proof ψ. It returns $(\mathtt{x}_{\mathsf{bls}}, \psi)$.

The condition $\mathsf{Real}(\mathcal{A}) = 1 \wedge \boldsymbol{a} \neq \boldsymbol{a}'$ implies that $\mathsf{V}_{\mathsf{bls}}(\mathsf{crs}_{\mathsf{bls}}, \mathtt{x}_{\mathsf{bls}}, \psi) = 1 \wedge \mathtt{x}_{\mathsf{bls}} \notin \mathbf{Im}\begin{pmatrix}[\mathbf{N}_1]_1 \\ [\mathbf{N}_2]_2\end{pmatrix}$. Thus, we get that $\Pr[\mathsf{Real}(\mathcal{A}) = 1 \wedge \boldsymbol{a} \neq \boldsymbol{a}'] \leq \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_0, \Pi_{\mathsf{bls}}}(\lambda)$ and the result follows. □

Lemma 8. *The following holds,*

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

Proof. The winning condition in Game_1 is the winning condition of Game_0 (soundness condition and $\boldsymbol{a} = \boldsymbol{a}'$) and additionally that the j^*-th equation is invalid. Thus,

$$\begin{aligned}\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] = \Pr[\mathsf{Game}_0(\mathcal{A}) = 1 \wedge j^*\text{-th equation is invalid}] = \\ \Pr[\mathsf{Game}_0(\mathcal{A}) = 1] \cdot \Pr[j^*\text{-th equation is invalid} \mid \mathsf{Game}_0(\mathcal{A}) = 1] \ .\end{aligned}$$

In order for $\mathcal{A}$ to win in Game_0 one of the d equations has to be invalid. Given that j^* is picked uniformly at random and independently from $\mathcal{A}$'s input, we get that $\Pr[j^*\text{-th equation is invalid} \mid \mathsf{Game}_0(\mathcal{A}) = 1] \geq 1/d$. Therefore,

$$\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] \geq \Pr[\mathsf{Game}_0(\mathcal{A}) = 1] \cdot (1/d)$$

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

Lemma 9. *There exists a non-uniform PPT adversary* $\mathcal{B}_1$ *against FSH such that* $|\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] - \Pr[\mathsf{Game}_2(\mathcal{A}) = 1]| = \mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{COM}, n', q_v, \mathcal{B}_1}(\lambda)$.

Proof. Game_1 and Game_2 differ only in the FSSB commitment key that encode different functions, but these keys are indistinguishable due to the FSH property.

To show this, we construct an adversary $\mathcal{B}_1$ that first gets as an input (p, n', q_v). It then chooses two function sets $\mathcal{S}_0$ and $\mathcal{S}_1$. For simplicity, let us represent them as matrices $\mathcal{S}_0 = \boldsymbol{M}_0 = \boldsymbol{0}_{q_v \times n'}$ and $\mathcal{S}_1 = \boldsymbol{M}_1$ is the matrix described in Game_2. The values s, z, j^* that define $\boldsymbol{M}_1$ are stored by $\mathcal{B}_1$.

Then the FSH game picks $\beta \leftarrow\$ \{0,1\}$ and generates a commitment key $\widetilde{\mathtt{ck}}_\beta$ from $\boldsymbol{M}_\beta$. Next, $\mathcal{B}_1$ gets $\widetilde{\mathtt{ck}}_\beta$ as an input. Since it knows s, z, it can simulate $\mathtt{crs}$ of Π_{SAP}, and run $\mathcal{A}(\mathsf{p}, \mathtt{crs})$ to obtain $\big(x = ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2), \pi\big)$. It extracts $[\boldsymbol{a}]_1$ and $[\boldsymbol{a}']_2$ respectively from $[\boldsymbol{c}]_1$, $[\boldsymbol{c}']_2$, and then tests the winning condition of $\mathsf{Game}_1/\mathsf{Game}_2$:

- $\mathsf{V}(\mathtt{crs}, ([\boldsymbol{c}]_1, [\boldsymbol{c}']_2), \pi) = 1$,
- $[\boldsymbol{a}]_1 [1]_2 = [1]_1 [\boldsymbol{a}']_2$, which implies that $\boldsymbol{a} = \boldsymbol{a}'$, and
- $([\boldsymbol{a}]_1^\top \boldsymbol{v}_{j^*})([\boldsymbol{a}]_2^\top \boldsymbol{v}_{j^*}) - ([\boldsymbol{a}]_1^\top \boldsymbol{w}_{j^*})[1]_2 \neq [0]_T$ (Note that is also implies the more general soundness winning condition $x \notin \mathscr{L}_{\mathsf{SAP},\mathsf{ck}}$).

If all of the above hold, then $\mathcal{B}_1$ outputs 1 and otherwise it outputs 0. Since the case $\beta = 0$ simulates Game_1 and $\beta = 1$ simulates Game_2, we get that

$$\begin{aligned}
\Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_\beta) = \beta] =& \Pr[\beta = 1] \cdot \Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_1) = 1 \mid \beta = 1] \\
&+ \Pr[\beta = 0] \cdot \Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_0) = 0 \mid \beta = 0] \\
=& \frac{1}{2}\big(\Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_1) = 1 \mid \beta = 1] + (1 - \Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_0) = 1 \mid \beta = 0])\big) \\
=& \frac{1}{2}\big(\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] + (1 - \Pr[\mathsf{Game}_1(\mathcal{A}) = 1])\big).
\end{aligned}$$

The advantage of $\mathcal{B}_1$ for FSH property is defined as

$$\begin{aligned}
\mathsf{Adv}^{\text{fsh}}_{\mathsf{COM},n',q_v,\mathcal{B}_1}(\lambda) =& 2 \cdot | \Pr[\mathcal{B}_1(\widetilde{\mathtt{ck}}_\beta) = \beta] - \frac{1}{2}| \\
=& 2 \cdot | \frac{1}{2}\big(\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] + 1 - \Pr[\mathsf{Game}_1(\mathcal{A}) = 1]\big) - \frac{1}{2}| \\
=& | \Pr[\mathsf{Game}_2(\mathcal{A}) = 1] - \Pr[\mathsf{Game}_1(\mathcal{A}) = 1]|.
\end{aligned}$$

Therefore, we have proven the claim. □

Lemma 10. *There exists a non-uniform PPT $\mathcal{B}_2$ against the strong soundness of the Π_{bls} and a non-uniform PPT $\mathcal{B}_3$ against d-SATSDH such that*

$$\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \le \mathsf{Adv}^{\text{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathsf{satsdh}}_{\mathcal{B}_3,d}(\lambda).$$

Proof. 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,

$$\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \le \Pr[\mathsf{Game}_2(\mathcal{A}) = 1|E] + \Pr[\mathsf{Game}_2(\mathcal{A}) = 1|\overline{E}]. \tag{8}$$

For the latter event, we can easily construct from $\mathcal{A}$ a non-uniform PPT adversary $\mathcal{B}_2$ that breaks strong quasi-adaptive soundness of Π_{bls}. Such an adversary receives as an input $\big(\mathtt{crs}_{\mathsf{bls}}, \omega_{\rho_{\mathsf{bls}}} = (u, u', s, \mathbf{K})\big)$ sampled according to the distribution specified by Game_2. This is sufficient to construct the CRS of Π_{SAP} in the usual way. Now the adversary $\mathcal{B}_2$ can use the output of $\mathcal{A}$ to break the soundness of Π_{bls} in a straightforward way. Thus, $\Pr[\mathsf{Game}_2(\mathcal{A}) = 1|\overline{E}] \le \mathsf{Adv}^{\text{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda)$.

In the following, we bound the first term of the sum in Eq. (8) by constructing a non-uniform 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 $(\boldsymbol{a}, \boldsymbol{r}, \boldsymbol{r}', \delta_v, \delta_w, r_v)^\top$ for membership in $\mathbf{Im}\begin{pmatrix} [\mathbf{N}_1]_1 \\ [\mathbf{N}_2]_2 \end{pmatrix}$. Furthermore, if the $\mathcal{A}$ wins in Game_2, then the witness is unique since

- $[\boldsymbol{c}]_1$ is perfectly binding and thus uniquely fixes $\boldsymbol{a}$ and $\boldsymbol{r}$,

- $[\boldsymbol{c}']_2$ is perfectly binding and thus uniquely fixes $\boldsymbol{a}'$ and $\boldsymbol{r}'$,
- winning condition requires that $\boldsymbol{a} = \boldsymbol{a}'$,
- $[V]_1$ 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.

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 $\left(\mathsf{p}, \{\left[s^i\right]_{1,2}\}_{i=1}^{d}, [z]_2\right)$. Let us observe that from this it is possible to construct the the CRS of $\Pi_{\mathtt{SAP}}$ as it is specified in Game_2:

- $\mathtt{ck} \leftarrow \mathcal{D}_{\mathsf{p}}(n, d)$ does not depend on s or z and can be easily sampled (together with secret keys u, u').
- $(\widetilde{\mathtt{ck}}, \widetilde{\mathtt{td}})$ can be constructed but it needs some further explanation. Firstly, since $\boldsymbol{M}$ described in Game_2 contains some univariate polynomials in s of degree less that d and linear functions in z, then $\mathcal{B}_3$ is able to efficiently compute $[\boldsymbol{M}]_2$ using the challenge $\{\left[s^i\right]_2\}_{i=1}^{d-1}, [z]_2$. Secondly, even though KC_2 requires $\boldsymbol{M}$ as an input, if we look at the construction in Fig. 4 it is clear that $(\widetilde{\mathtt{ck}}, \widetilde{\mathtt{td}})$ can also be efficiently computed from $[\boldsymbol{M}]_2$.
- $\mathcal{B}_3$ can efficiently compute $[\mathbf{N}_1]_1$ and $[\mathbf{N}_2]_2$ from $\mathtt{ck}$, $\{\left[s^i\right]_{1,2}\}_{i=1}^{d}$, and $\widetilde{\mathtt{ck}}$. Then $(\mathsf{crs}_{\mathsf{bls}}, \mathsf{td}_{\mathsf{bls}}) \leftarrow \mathsf{K}_{\mathsf{bls}}([\boldsymbol{N}_1]_1, [\boldsymbol{N}_2]_2)$.
- By composing them, we get $\mathtt{crs} = (\mathsf{p}, \mathtt{ck}, \widetilde{\mathtt{ck}}, \{\left[s^i\right]_{1,2}\}_{i=1}^{d}, \mathsf{crs}_{\mathsf{bls}})$ of $\Pi_{\mathtt{SAP}}$.

Then $\mathcal{B}_3$ sends $\mathtt{crs}$ to the soundness adversary $\mathcal{A}$ that returns $([\boldsymbol{c}]_1, [\boldsymbol{c}']_2)$ and π.

The adversary $\mathcal{B}_3$ uses the secret key u to extract $[\boldsymbol{a}]_1 \in \mathbb{G}_1$ from $[\boldsymbol{c}]_1$ and uses the FSSB trapdoor $\widetilde{\mathtt{td}}$ to extract from $[\tilde{\boldsymbol{c}}]_2$ the following elements:

- $[q_v(s)]_2 = [\sum_{i=1}^{n} a_i \alpha_i(s) + \delta_v \alpha_t(s)]_2$,
- $[q_w(s)]_2 = [\sum_{i=1}^{n} a_i \hat{\alpha}_i(s) + \delta_v \alpha_t(s)]_2$,
- $[\beta_v z]_2 = [\sum_{i=1}^{n} a_i z \beta_{v,i}]_2$, and
- $[\beta_v z]_2 = [\sum_{i=1}^{n} a_i z \beta_{w,i}]_2$.

The above equalities hold because of Lemma 6.

Since verification succeeds, $[V]_1[V]_2 - [W]_T = [H]_1[t(s)]_2$. 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 $\{[s^i]_2\}_{i=0}^{d}$ in the CRS. On the other hand, based on Lemma 5

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

and we have $\beta_v^2 - \beta_w \neq 0$ (otherwise the j^*-th equation is satisfied, in which case the game aborts), that is $\beta_v^2 \neq \beta_w$.

According to Eq. (6) and Eq. (7), $\mathcal{B}_3$ can compute $[\beta_v]_1 = \sum_{i=0}^{n} [a_i]_1 \beta_{v,i}$, $[\beta_w]_1 = \sum_{i=0}^{n} [a_i]_1 \beta_{w,i}$. Thus, $\mathcal{B}_3$ can also compute

$$\left[\frac{\beta_v^2 - \beta_w}{s - \chi_{j^*}}\right]_T = \left[\frac{P(s)}{s - \chi_{j^*}}\right]_T - ([V]_1 + [\beta_v]_1)[q_v(s)]_2 + [1]_1[q_w(s)]_2 .$$

Finally $\mathcal{B}_3$ returns

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

Since $\beta_v^2 \neq \beta_w$, it breaks the d-SATSDH assumption.

Hence by the triangle inequality we have $\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \leq \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathsf{satsdh}}_{\mathcal{B}_3,d}(\lambda)$. □

Finally, we get that if there exists a non-uniform PPT soundness adversary $\mathcal{A}$, then there exist non-uniform PPT adversaries $\mathcal{B}_0$, $\mathcal{B}_1$, $\mathcal{B}_2$, and $\mathcal{B}_3$ such that

- according to Lemma 7, $\Pr[\mathsf{Real}(\mathcal{A}) = 1] \leq \Pr[\mathsf{Game}_0(\mathcal{A}) = 1] + \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_0,\Pi_{\mathsf{bls}}}(\lambda)$,
- according to Lemma 8, $\Pr[\mathsf{Game}_0(\mathcal{A}) = 1] \leq d \cdot \Pr[\mathsf{Game}_1(\mathcal{A}) = 1]$,
- according to Lemma 9, $|\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] - \Pr[\mathsf{Game}_2(\mathcal{A}) = 1]| = \mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{COM},n',q_v,\mathcal{B}_1}(\lambda)$, and
- according to Lemma 10, $\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \leq \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathsf{satsdh}}_{\mathcal{B}_3,d}(\lambda)$.

When combining those, we obtain

$$\mathsf{Adv}^{\mathrm{snd}}_{\mathcal{A},\Pi_{\mathsf{SAP}}}(\lambda) \leq \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_0,\Pi_{\mathsf{bls}}}(\lambda) + d\big(\mathsf{Adv}^{\mathrm{fsh}}_{\mathsf{COM},n',q_v,\mathcal{B}_1}(\lambda) + \mathsf{Adv}^{\mathrm{snd}}_{\mathcal{B}_2,\Pi_{\mathsf{bls}}}(\lambda) + \mathsf{Adv}^{\mathsf{satsdh}}_{\mathcal{B}_3,d}(\lambda)\big).$$

□

Acknowledgments. We would like to thank Carla Ràfols and Janno Veeorg for useful discussions. The authors were supported by the Estonian Research Council grant (PRG49) and by Dora Plus Grant funded by the European Regional Development Fund, Republic of Estonia and Archimedes Foundation. Janno Siim was additionally supported by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 780477 (project PRIViLEDGE). Part of this work was done while Janno Siim was affiliated with the University of Edinburgh.

References

ABLZ17. Behzad Abdolmaleki, Karim Baghery, Helger Lipmaa, and Michal Zajac. A subversion-resistant SNARK. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part III*, volume 10626 of *LNCS*, pages 3–33. Springer, Heidelberg, December 2017. doi:10.1007/978-3-319-70700-6_1.

AIR01. William Aiello, Yuval Ishai, and Omer Reingold. Priced oblivious transfer: How to sell digital goods. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 119–135. Springer, Heidelberg, May 2001. doi:10.1007/3-540-44987-6_8.

ALM+92. Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan, and Mario Szegedy. Proof verification and hardness of approximation problems. In *33rd FOCS*, pages 14–23. IEEE Computer Society Press, October 1992. doi:10.1109/SFCS.1992.267823.

AS92. Sanjeev Arora and Shmuel Safra. Probabilistic checking of proofs; A new characterization of NP. In *33rd FOCS*, pages 2–13. IEEE Computer Society Press, October 1992. doi:10.1109/SFCS.1992.267824.

BB04. Dan Boneh and Xavier Boyen. Secure identity based encryption without random oracles. In Matthew Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 443–459. Springer, Heidelberg, August 2004. doi:10.1007/978-3-540-28628-8_27.

BBB+18. Benedikt Bünz, Jonathan Bootle, Dan Boneh, Andrew Poelstra, Pieter Wuille, and Greg Maxwell. Bulletproofs: Short proofs for confidential transactions and more. In *2018 IEEE Symposium on Security and Privacy*, pages 315–334. IEEE Computer Society Press, May 2018. doi:10.1109/SP.2018.00020.

BCKL08. Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. P-signatures and noninteractive anonymous credentials. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 356–374. Springer, Heidelberg, March 2008. doi:10.1007/978-3-540-78524-8_20.

CF13. Dario Catalano and Dario Fiore. Vector commitments and their applications. In Kaoru Kurosawa and Goichiro Hanaoka, editors, *PKC 2013*, volume 7778 of *LNCS*, pages 55–72. Springer, Heidelberg, February / March 2013. doi:10.1007/978-3-642-36362-7_5.

CFS17. Alessandro Chiesa, Michael A. Forbes, and Nicholas Spooner. A zero knowledge sumcheck and its applications. Cryptology ePrint Archive, Report 2017/305, 2017. https://eprint.iacr.org/2017/305.

CGM16. Melissa Chase, Chaya Ganesh, and Payman Mohassel. Efficient zero-knowledge proof of algebraic and non-algebraic statements with applications to privacy preserving credentials. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part III*, volume 9816 of *LNCS*, pages 499–530. Springer, Heidelberg, August 2016. doi:10.1007/978-3-662-53015-3_18.

CV05. Dario Catalano and Ivan Visconti. Hybrid trapdoor commitments and their applications. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP 2005*, volume 3580 of *LNCS*, pages 298–310. Springer, Heidelberg, July 2005. doi:10.1007/11523468_25.

DFGK14. George Danezis, Cédric Fournet, Jens Groth, and Markulf Kohlweiss. Square span programs with applications to succinct NIZK arguments. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 532–550. Springer, Heidelberg, December 2014. doi:10.1007/978-3-662-45611-8_28.

DFL$^+$09. Ivan Damgård, Serge Fehr, Carolin Lunemann, Louis Salvail, and Christian Schaffner. Improving the security of quantum protocols via commit-and-open. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 408–427. Springer, Heidelberg, August 2009. doi:10.1007/978-3-642-03356-8_24.

DGI$^+$19. Nico Döttling, Sanjam Garg, Yuval Ishai, Giulio Malavolta, Tamer Mour, and Rafail Ostrovsky. Trapdoor hash functions and their applications. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part III*, volume 11694 of *LNCS*, pages 3–32. Springer, Heidelberg, August 2019. doi:10.1007/978-3-030-26954-8_1.

DGN$^+$17. Nico Döttling, Satrajit Ghosh, Jesper Buus Nielsen, Tobias Nilges, and Roberto Trifiletti. TinyOLE: Efficient actively secure two-party computation from oblivious linear function evaluation. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *ACM CCS 2017*, pages 2263–2276. ACM Press, October / November 2017. doi:10.1145/3133956.3134024.

DGP$^+$19. Vanesa Daza, Alonso González, Zaira Pindado, Carla Ràfols, and Javier Silva. Shorter quadratic QA-NIZK proofs. In Dongdai Lin and Kazue Sako, editors, *PKC 2019, Part I*, volume 11442 of *LNCS*, pages 314–343. Springer, Heidelberg, April 2019. doi:10.1007/978-3-030-17253-4_11.

DKM12. Nico Döttling, Daniel Kraschewski, and Jörn Müller-Quade. Statistically secure linear-rate dimension extension for oblivious affine function evaluation. In Adam Smith, editor, *ICITS 12*, volume 7412 of *LNCS*, pages 111–128. Springer, Heidelberg, August 2012. doi:10.1007/978-3-642-32284-6_7.

DN02. Ivan Damgård and Jesper Buus Nielsen. Perfect hiding and perfect binding universally composable commitment schemes with constant expansion factor. In Moti Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 581–596. Springer, Heidelberg, August 2002. doi:10.1007/3-540-45708-9_37.

EHK$^+$13. Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Heidelberg, August 2013. doi:10.1007/978-3-642-40084-1_8.

FLPS21. Prastudy Fauzi, Helger Lipmaa, Zaira Pindado, and Janno Siim. Somewhere Statistically Binding Commitment Schemes with Applications. In Nikita Borisov and Claudia Diaz, editors, *FC 2021 (1)*, volume 12674 of *LNCS*, pages 436–456, Virtual, March 1–15, 2021. Springer, Cham. doi:10.1007/978-3-662-64322-8_21.

FMMO19. Prastudy Fauzi, Sarah Meiklejohn, Rebekah Mercer, and Claudio Orlandi. Quisquis: A new design for anonymous cryptocurrencies. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part I*, volume 11921 of *LNCS*, pages 649–678. Springer, Heidelberg, December 2019. doi:10.1007/978-3-030-34578-5_23.

FOS19. Georg Fuchsbauer, Michele Orrù, and Yannick Seurin. Aggregate cash systems: A cryptographic investigation of Mimblewimble. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part I*, volume 11476 of *LNCS*, pages 657–689. Springer, Heidelberg, May 2019. doi:10.1007/978-3-030-17653-2_22.

GGPR13. Rosario Gennaro, Craig Gentry, Bryan Parno, and Mariana Raykova. Quadratic span programs and succinct NIZKs without PCPs. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 626–645. Springer, Heidelberg, May 2013. doi:10.1007/978-3-642-38348-9_37.

GHR15. Alonso González, Alejandro Hevia, and Carla Ràfols. QA-NIZK arguments in asymmetric groups: New tools and new constructions. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part I*, volume 9452 of *LNCS*, pages 605–629. Springer, Heidelberg, November / December 2015. doi:10.1007/978-3-662-48797-6_25.

GM17. Jens Groth and Mary Maller. Snarky signatures: Minimal signatures of knowledge from simulation-extractable SNARKs. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part II*, volume 10402 of *LNCS*, pages 581–612. Springer, Heidelberg, August 2017. doi:10.1007/978-3-319-63715-0_20.

GNN17. Satrajit Ghosh, Jesper Buus Nielsen, and Tobias Nilges. Maliciously secure oblivious linear function evaluation with constant overhead. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 629–659. Springer, Heidelberg, December 2017. doi:10.1007/978-3-319-70694-8_22.

GR05. Craig Gentry and Zulfikar Ramzan. Single-database private information retrieval with constant communication rate. In Luís Caires, Giuseppe F. Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *ICALP 2005*, volume 3580 of *LNCS*, pages 803–815. Springer, Heidelberg, July 2005. doi:10.1007/11523468_65.

GR16. Alonso González and Carla Ràfols. New techniques for non-interactive shuffle and range arguments. In Mark Manulis, Ahmad-Reza Sadeghi, and Steve Schneider, editors, *ACNS 16*, volume 9696 of *LNCS*, pages 427–444. Springer, Heidelberg, June 2016. doi:10.1007/978-3-319-39555-5_23.

GR19. Alonso González and Carla Ràfols. Shorter pairing-based arguments under standard assumptions. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part III*, volume 11923 of *LNCS*, pages 728–757. Springer, Heidelberg, December 2019. doi:10.1007/978-3-030-34618-8_25.

Gro10. Jens Groth. Short pairing-based non-interactive zero-knowledge arguments. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 321–340. Springer, Heidelberg, December 2010. doi:10.1007/978-3-642-17373-8_19.

GS08. Jens Groth and Amit Sahai. Efficient non-interactive proof systems for bilinear groups. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 415–432. Springer, Heidelberg, April 2008. doi:10.1007/978-3-540-78967-3_24.

GW11. Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 99–108. ACM Press, June 2011. doi:10.1145/1993636.1993651.

HW15. Pavel Hubacek and Daniel Wichs. On the communication complexity of secure function evaluation with long output. In Tim Roughgarden, editor, *ITCS 2015*, pages 163–172. ACM, January 2015. doi:10.1145/2688073.2688105.

JR13. Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 1–20. Springer, Heidelberg, December 2013. doi:10.1007/978-3-642-42033-7_1.

Kil94. Joe Kilian. On the complexity of bounded-interaction and noninteractive zero-knowledge proofs. In *35th FOCS*, pages 466–477. IEEE Computer Society Press, November 1994. doi:10.1109/SFCS.1994.365744.

Lip05. Helger Lipmaa. An oblivious transfer protocol with log-squared communication. In Jianying Zhou, Javier Lopez, Robert H. Deng, and Feng Bao, editors, *ISC 2005*, volume 3650 of *LNCS*, pages 314–328. Springer, Heidelberg, September 2005.

Lip12. Helger Lipmaa. Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 169–189. Springer, Heidelberg, March 2012. doi:10.1007/978-3-642-28914-9_10.

LY10. Benoît Libert and Moti Yung. Concise mercurial vector commitments and independent zero-knowledge sets with short proofs. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 499–517. Springer, Heidelberg, February 2010. doi:10.1007/978-3-642-11799-2_30.

NP01. Moni Naor and Benny Pinkas. Efficient oblivious transfer protocols. In S. Rao Kosaraju, editor, *12th SODA*, pages 448–457. ACM-SIAM, January 2001.

OPWW15. Tatsuaki Okamoto, Krzysztof Pietrzak, Brent Waters, and Daniel Wichs. New realizations of somewhere statistically binding hashing and positional accumulators. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part I*, volume 9452 of *LNCS*, pages 121–145. Springer, Heidelberg, November / December 2015. doi:10.1007/978-3-662-48797-6_6.

Ped92. Torben P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 129–140. Springer, Heidelberg, August 1992. doi:10.1007/3-540-46766-1_9.

Poe16. Andrew Poelstra. Mimblewimble, 2016. Available at https://download.wpsoftware.net/bitcoin/wizardry/mimblewimble.pdf.

RS20. Carla Ràfols and Javier Silva. QA-NIZK arguments of same opening for bilateral commitments. In Abderrahmane Nitaj and Amr M. Youssef, editors, *AFRICACRYPT 20*, volume 12174 of *LNCS*, pages 3–23. Springer, Heidelberg, July 2020. doi:10.1007/978-3-030-51938-4_1.

Vil12. Jorge Luis Villar. Optimal reductions of some decisional problems to the rank problem. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 80–97. Springer, Heidelberg, December 2012. doi:10.1007/978-3-642-34961-4_7.

A Relation to Existing Primitives

A.1 Relation to SSB Hashes

The SSB requirement makes the EMP commitment scheme look similar to SSB hash functions [HW15, OPWW15], 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, [HW15, OPWW15] 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 [OPWW15] and efficient vector commitments [LY10, CF13] (which have a local opening) without the SSB property.

A.2 Relation to Oblivious Transfer (OT)

SSB commitments are directly related to two-message OT protocols as defined in [AIR01]. 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 $\mathtt{otq}$ (an encoding of $\mathcal{S}$) to the sender who replies with the second message $\mathtt{otr}$ (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 [NP01, AIR01, GR05, Lip05] 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.

A.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 [AS92, ALM$^+$92]: one can establish, by checking one random coordinate of the bit-string (resp., all adjacent wires of a random gate), whether an input belongs to the language or not. Typical PCP-based zero-knowledge arguments like [Kil94] use PCPs with small soundness error; as a drawback, such PCPs have a long proof and an inefficient reduction from CircuitSAT. Daza *et al.* [DGP$^+$19] 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.

B Details of Algebraic Commitments Schemes (ACS)

B.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 11.

Moreover, for extraction assume that $\operatorname{span}\{\boldsymbol{U}_1\} \cap \operatorname{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$. 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 $\mathtt{ek} \cdot [\boldsymbol{U}_1]_\iota = (\boldsymbol{b}_i)_{i=1}^n$, 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 11. *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 3.*

1. COM *is AECH under $\mathcal{D}_2$-MDDH$_{\mathbb{G}_\iota}$.*
2. COM *is ISH under $\mathcal{D}_1, \mathcal{D}_2$-MDDH$_{\mathbb{G}_\iota}$.*
3. COM *is SPB if $\boldsymbol{U}_1$ has rank q and* $\operatorname{span}\{\boldsymbol{U}_1\} \cap \operatorname{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$.
4. COM *is $[\cdot]_\iota$-SPE if $\boldsymbol{U}_1$ has rank q and* $\operatorname{span}\{\boldsymbol{U}_1\} \cap \operatorname{span}\{\boldsymbol{U}_2\} = \{\mathbf{0}\}$.

Proof. Let $\mathcal{S} \subseteq [1\mathinner{..}n]$, $|\mathcal{S}| \leq q$ 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 $\mathcal{S}_0, \mathcal{S}_1 \subseteq [1\mathinner{..}n]$, 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 $\mathsf{Adv}^{\text{indist}}_{\mathcal{A}, \mathcal{D}^0_{1,2}, \mathcal{D}^1_{1,2}}(\lambda) \leq (|\mathcal{S}_0 \cup \mathcal{S}_1| - |\mathcal{S}_0 \cap \mathcal{S}_1|) \cdot \mathsf{Adv}^{\text{mddh}}_{\mathcal{B}, \mathcal{D}^{n,q}_{1,2}, \mathsf{Pgen}}(\lambda)$.

(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 $(\boldsymbol{U}_1, \boldsymbol{U}_2)\left(\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{r}\end{smallmatrix}\right) = \mathsf{Com}_{\mathtt{ck}}(\boldsymbol{x}, \boldsymbol{r})$. This system has a unique solution because the matrix has full rank. Hence, each commitment corresponds to a unique vector $\left(\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{r}\end{smallmatrix}\right)$. 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 $\mathtt{ek} \in \boldsymbol{U}_2^{\perp}$ that define orthogonal spaces of $\boldsymbol{U}_2$ of size $k' \times k$ for $k' \geq k - m$ such that $\mathtt{ek} \cdot \boldsymbol{U}_2 = \begin{pmatrix} \boldsymbol{0}_{(k-m)\times m} \\ \boldsymbol{a} \end{pmatrix}$ 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 $\mathtt{ek} \cdot \boldsymbol{U}_1 = \left(\begin{array}{c|c} \boldsymbol{I}_q & \boldsymbol{b}_2 \\ \boldsymbol{b}_1 & \end{array} \right)$ 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.* [GHR15, GR16, DGP+19] 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[10]. Moreover, in our QA-NIZK argument in Section 7, as well as [DGP+19], we use functional SSB commitments since FSSB is more straightforward to our use of it in the soundness proof.

C Security of q-SATSDH

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 8 (Bilinear Diffie-Hellman Knowledge of Exponent Assumption, BDH-KE [ABLZ17]). *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 12. *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.*

[10] In this sense, one could also call them trapdoor hash functions [DGI+19] with the SSB and AESH properties

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 $\big(r, [\beta_1, \beta_2]_1, \beta_1[z]_2, \beta_2[z]_2, (\beta_1^2 - \beta_2)[\nu]_T\big)$ which breaks the q-SATSDH assumption. □

Lemma 13. *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 $\Big(r, \frac{1}{\beta_1^2 - \beta_2}[\nu]_T\Big)$. □