Succinct Publicly Verifiable Computation

Alonso González*[1] and Alexandros Zacharakis†[2]

[1]ENS de Lyon, Laboratoire LIP (U. Lyon, CNRS, ENSL, INRIA, UCBL), France
[2]Universitat Pompeu Fabra, Barcelon, Spain

alonso.gonzalez@ens-lyon.fr, alexandros.zacharakis@upf.edu

March 17, 2021

Abstract

In this work we construct for the first time a delegation scheme for arithmetic circuits with proof-size and verification complexity comparable to those of pairing based zk-SNARKS (e.g. Gennaro et al. at Eurocrypt 2013 or Groth at Eurocrypt 2016), but based on standard assumptions. Each proof comprises $O(1)$ group elements of a bilinear group and verification requires $O(1)$ pairings plus n exponentiations, where n is the number of inputs. Soundness can be proven under any Matrix Diffie-Hellman (MDDH) assumption of size $k \geq 2$. The size of the reference string as well as the prover's complexity is quadratic in the size of the circuit.

Our techniques combine the ideas for constructing delegation schemes of Paneth and Rothblum (TCC 2017), and then refined by Kalai et al. (STOC 2019), with the so called Quasi-Adaptive NIZK arguments for linear languages (Jutla and Roy at Asiacrypt 2014 and Crypto 2015, Libert et al. Eurocrypt 2015, Kiltz and Wee Eurocrypt 2015) and for quadratic languages (González et al. at Asiacrypt 2015 and 2019). We obtain a delegation scheme with asymptotically shorter proofs and verification.

Our construction can be turned into a NIZK argument for NP of size $n + O(1)$ group elements under the same assumptions and can be used to construct zk-SNARKs from quantitatively weaker assumptions than the state of the art. Additionally, the NIZK argument for NP yields a compact NIZK for NP with proof size linear in the size of the witness by using the same techniques and improving on Katsumata et al. (Crypto 2019 and Eurocrypt 2020) which has proof size linear in the size of the circuit.

*This author was supported by the European Union PROMETHEUS project (Horizon 2020 Research and Innovation Program, grant 780701)

†Research Supported by fellowships from "la Caixa" Foundation (ID 100010434). The fellowship codes are LCF/BQ/DI18/11660052 and LCF/BQ/DI18/11660053. Funding is also from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 713673.

Contents

1 Introduction

In a delegation scheme a verifier with limited computational resources (a mobile device for example) wishes to delegate a heavy but still polynomial computation to an untrusted prover. The prover, with more computational power but still of polynomial time, computes a proof which the verifier accepts or rejects. Given the limitations of the verifier, the proof should be as short as possible and the verification process should consume as few computational resources as possible.

A delegation scheme can be easily constructed from a zero-knowledge Succinct Non-interactive Argument of Knowledge (zk-SNARK) for NP. Schemes like [GGPR13, Gro16] are very appealing in practice because a proof consists of only a constant number of group elements and verification requires the evaluation of a constant number of pairings.[1] The downside is that these zk-SNARKs are based on strong and controversial assumptions such as the knowledge of exponent assumption or the generic group model. Such assumptions are called non-falsifiable because there is no way of efficiently decide whether an adversary breaks the assumption or not. Actually, since these zk-SNARKs can handle even NP computations, soundness becomes an essentially non-falsifiable property where one needs to decide whether an adversary produces a true or false statement without any witness but only with a very short proof. Gentry and Wichs [GW11] proved that zk-SNARKs for NP are (in a broad sense) impossible to construct without resorting to non-falsifiable assumptions.

Interactive schemes such as [GKR08, RRR16] offer delegation even with unconditional soundness at the cost of many rounds of interaction between the prover and the verifier. The downside is that, in general, interactive protocols can't be publicly verified and they might be even deniable [DNS98]. Designated verifier schemes such as [KRR14] suffers from the same problem. Although interactive schemes can be made non-interactive and publicly verifiable in the random oracle model, such a strong assumption is again non-falsifiable.

In contrast to NP computations, soundness of a delegation scheme is a falsifiable statement. Indeed, determining whether the adversary breaks soundness becomes efficiently falsifiable since it requires to evaluate the delegated polynomial computation on some input x and check whether it is accepting or rejecting. This observation shows that there is no evident reason for using non-falsifiable assumptions in the construction of delegation schemes.

In fact, there are many arguments for lower complexity classes whose security is based on falsifiable assumptions. [KPY18] constructs a delegation scheme using (constant-size) non-falsifiable assumptions but limited to bounded depth log-space uniform computations. González and Ràfols [GR19] constructed a delegation scheme for bounded depth computations based on a q-assumption over bilinear groups. Canetti et al. [CCH$^+$19] constructed delegation for log-space uniform NC computations based on a form of perfect circular security for fully homomorphic encryption. Very recently in [JKKZ20], the authors give a delegation for log-space uniform circuits under the sub-exponential hardness of the LWE assumption. Kalai et al. [KPY19] constructed a delegation scheme for any poly-time computation based on a type of q-assumption in bilinear groups, where $q = \log \kappa$ and κ is the security parameter. For a computation taking T steps, the size of the proof is $\mathsf{polylog}(T)$ group elements which becomes $\mathsf{poly}(\kappa)$ group elements if $T \leq 2^\kappa$. In [KPY19] the authors introduced an interesting relaxation of arguments of knowledge which they called quasi-arguments of knowledge. Then they showed that their quasi-arguments become normal arguments in the case of polynomial time computations.

However, in spite ot the recent progress, there's still a gap in the proof size and verification with respect to paring based zk-SNARKs. Furthermore, we would like to use assumptions as standard as DDH or matrix Decisional Diffie-Hellman.

1.1 Our results

In this work we construct a succinct delegation scheme with public verification. To achieve this goal we: (1) introduce and construct efficient *no-signaling somewhere statistically binding commitments*; (2) construct

[1] Note that zero-knowledge is not necessary.

quasi-arguments for linear and quadratic relations with shorter proofs and verification complexity; and (3) we use and improve the *knowledge transfer arguments* of [GR19].

Succinct Publicly Verifiable Delegation. We construct a delegation scheme for arithmetic circuits in the pre-processing model. That is, there is a preprocessing stage where some common reference string (crs) is set up and additionally, this crs might depend on the particular arithmetic circuit. Each proof comprises 10+8 group elements of an asymmetric bilinear group. Verification requires n exponentiations plus 36 evaluations of the pairing function, where n is the size of the input. In symmetric groups, soundness can be proven under any Matrix Diffie-Hellman (MDDH) assumption of size $k \geq 2$, as for example the decisional linear assumption (DLin). In asymmetric groups soundness can be based on the natural translation of symmetric DLin where the challenge is encoded in both groups (the SDlin assumption of [GHR15a]). The size of the common reference string as well as the prover's complexity are quadratic in the size of the circuit. In Table 1 we provide a comparison with other delegation schemes.

Table 1: Comparison between different pairing based delegation schemes and our results.

	Language	Verification	Proof size	CRS size	Assumption
[GGPR13][Gro16]	AC	$n\mathsf{e} + O(1)\mathsf{p}$	$O(\kappa)$	$O(\|C\|)$	Non Falsifiable
[KPY19] (base case)	RM	$n\mathsf{e} + O(\log d)\mathsf{p}$	$O(\kappa \log d)$	$O((n + \log d)\kappa)$	$\log \kappa$-Assumption
[GR19]	AC	$n\mathsf{e} + O(d)\mathsf{p}$	$O(d\kappa)$	$O(\|C\|\kappa)$	s-Assumption
This work	AC	$n\mathsf{e} + O(1)\mathsf{p}$	$O(\kappa)$	$O(\|C\|^2\kappa)$	SDLin

Verification is given in number exponentiations (e) and pairings (p). d is the circuit depth/number of steps of a computation, n the number of inputs, s the circuit width/computation space and $|C|$ the circuit size. AC stands for "Arithmetic Circuit" and RM for "RAM Machine". For [KPY19] we only consider the "base case" and not the "bootstrapped" constructions, because bootstrapping adds a considerable overhead (although it is only $\mathsf{poly}(\kappa)$) and hence is of less practical interest.

No-Signaling SSB Commitments and Succinct Pairing-based Quasi-Arguments. We follow and extend the ideas of Rothblum and Paneth [PR17] and Kalai et al. [KPY19] for constructing delegation schemes for poly-time computations from what they called quasi-arguments of knowledge with no-signaling extractors. We show that the somewhere statistically binding (SSB) commitments of [GHR15a, FLPS20] are no-signaling when they also have what we call an "oblivious trapdoor generator". We then construct more efficient constant-sized quasi-arguments of knowledge for linear and quadratic relations. We do so by combining SSB commitment with the quasi-adaptive non-interactive zero-knowledge arguments for linear [JR13, LPJY13, JR14, KW15] and quadratic relations [GHR15a, DGP+19].

Delegation for Unbounded Space Computations. The quasi argument of [KPY19] works only for bounded-space computations, or for bounded width circuits in our case.[2] Kalai et al. solved this issue by simulating high-space computations with low-space computations using hash-trees. This approach has the inconvenient of being non structure-preserving, in the sense that it ultimately relies on expressing as a circuit (or boolean formula) the evaluation of a hash function. We note that in practice this is very inefficient and in fact, is the same bottleneck found in the deployment of zk-SNARKs [BCG+14].

We take a different approach and note that the so called "Arguments of Knowledge Transfer" of [GR19] allow to achieve the same goal in a *structure-preserving* fashion [AFG+16]. That is, the statement being proven is "NIZK-friendly" and a practically efficient argument for the satisfiability of the statement can be derived without reducing a cryptographic primitive to a circuit. Specifically, in [GR19] the authors show how to express the correct evaluation of a circuit as a set of d paring product equations in only three variables,

[2] Equivalently, we can think the size of the proof grows with the width of the circuit.

where d is the depth of the circuit (hence, independent of the width). In this work we construct a constant-size quasi-argument of knowledge of solutions to these pairing product equations, bypassing the reduction to a circuit of a cryptographic primitive (of the pairing function in this case). Furthermore, we extend the work of [GR19] constructing arguments of knowledge transfer from any MDDH assumption at the cost of a quadratic crs.

Applications. Our construction can be turned into a NIZK argument for NP of size $n + O(1)$ group elements under the same assumptions. In table 2 we provide a comparison of our NIZK construction and the literature. Using standard techniques, the argument implies compact NIZK for NP with proof size linear in the size of the witness. That is, the size of the proof is proportional to the size of the input and not the security parameter (recall that the size of each element of a bilinear group is $O(\kappa)$, where κ is the security parameter). In comparison, the state of the art is $O(|C|)$ for poly-sized boolean circuits and $O(n)$ for log-depth boolean circuits [KNYY19, KNYY20].

Table 2: Comparison between different pairing based NIZK schemes and our results.

	Language	Verification	Proof size	CRS size	Assumption
[GOS06]	AC	$O(\|C\|)\mathtt{p}$	$O(\|C\|\kappa)$	$O(\kappa)$	SXDH
[GGPR13][Gro16]	AC	$O(1)\mathtt{p}$	$O(\kappa)$	$O(\|C\|\kappa)$	Non Falsifiable
[GR19]	BC	$O(n+d)\mathtt{p}$	$O((n+d)\kappa)$	$O(\|C\|\kappa)$	s-Assumption
[KNYY20]	NC[1]	$O(\|C\|)\mathsf{poly}(\kappa)$	$n\mathsf{poly}(\kappa)$	$\mathsf{poly}(\|C\|,\kappa,2^d)$	DLin
This work	BC	$O(n)\mathtt{p}$	$nO(\kappa)$	$O(\|C\|^2\kappa)$	SDLin

Verification is given in number of pairings $\mathtt{p}$. d is the circuit depth, n the number of inputs, s the circuit width and $|C|$ the circuit size. AC stands for "Arithmetic Circuit" and BC for "Boolean Circuit".

Our argument can be also used to construct zk-SNARKs from quantitatively weaker assumptions than the state of the art. Indeed, the strongest assumption used in zk-SNARKs such as [GGPR13, Gro16] is a knowledge assumption which states that an adversary computing some elements of a bilinear group, satisfying a particular relation, must know their discrete logarithms.[3] Such assumption is used to extract an assignment to each of the circuit wires. The "size" of such assumption is proportional to the number of extracted values, which in this case is the size of the circuit. Since our argument only requires the reduction to know the input of the circuit, we can rely on a knowledge assumption only for extracting the input. As a consequence the size of the assumption is drastically shortened.

2 Technical Overview

In this work we follow a commit-and-prove approach, which means that we first commit to the witness and then show that this witness satisfies some relation. Our approach also follows the ideas of Kalai et al.[KPY19] to derive a delegation scheme from a quasi-argument.

We use somewhere statistically binding (SSB) commitments as those used in [GHR15a, GR16, FLPS20] and show that they have no-signaling extractors. Then we do the same for the so called quasi-adaptive NIZK arguments for linear spaces [JR13, LPJY13, JR14, KW15] and for quadric relations [GHR15a, DGP$^+$19]. From these primitives we can construct delegation for bounded-space computations/bounded width circuits with proof-size independent of the depth of the computation. To get a succinct proof-size in addition to the "depth compression" we must also perform a "width compression". To do so we use the delegation scheme for bounded depth computations of González and Ràfols [GR19] and get rid of the q-assumption to rely solely on constant size assumptions. To combine both "compressions" efficiently we exploit the fact that [GR19] is structure preserving and the verifier is a bounded width circuit.

[3] Actually, the adversary must know a representation of these values as a linear combination of a set of group elements that she receives as input.

Bilinear Groups. In this high-level overview we will be using symmetric bilinear groups as it leads to simpler constructions. That is $(\mathbb{G}, \mathbb{G}_T, e)$ of primer order p and generators $\mathcal{P}, \mathcal{P}_T = e(\mathcal{P}, \mathcal{P})$ and $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ an efficiently computable non-degenerated bilinear map. For the more efficient case of asymmetric groups, we essentially use the idea of [GHR15a] of splitting quadratic terms in two random shares encoded in different groups.

2.1 Quasi-Arguments of Knowledge of [KPY19]

Paneth and Rothblum and then Kalai et al. used a weakened version of an argument of knowledge, which Kalai et al. called quasi-argument, as an intermediate step for obtaining a delegation scheme. Unlike an argument of knowledge, a quasi-argument has only local extraction, meaning that only a small part of the witness of size at most K, the locality parameter, is extracted. This is formalized by means of an extractor which on input a set $S \subseteq [n]$ of size at most K, where n is the size of the witness, programs a crs so that it can later extract positions of the witness defined by S. Central to quasi-arguments is the notion of no-signaling local extraction which is aimed to capture a strong *local soundness* guarantee. Local soundness means that the extracted local witness is consistent with the relation and doesn't lead to a local contradiction. The *no-signaling* requirement is defined for any two sets S, S' where $S' \subseteq S$ and of size at most K. It states that the result of programming extraction for S and then output only the extracted value for S', should be indistinguishable from the result of programming extraction for S' and output the extracted value for S'. Intuitively, this strengthens locality by requiring that the small parts of the local witness are extracted independently from rest so that it doesn't matter if extraction is done with a trapdoor for S or S'.

Delegation for P. Kalai et al. showed that quasi-arguments become arguments if the underlying language is in P, for example the language of (x, C) s.t. C is a circuit or TM and $C(x) = 1$, which is exactly the case of delegation. An outline of the construction will be useful for understanding the power of no-signaling extraction, which is a rather technical notion.

Consider some polynomial time sequential computation which on input x outputs y, for example a Turing Machine or an arithmetic circuit. To do so, the computation goes through a sequence of states $\mathtt{st}_0, \mathtt{st}_1, \ldots, \mathtt{st}_d$ such that $\mathtt{st}_0$ is consistent with the input, state $\mathtt{st}_d$ contains the output y, and there's a functional relation between states $\mathtt{st}_i, \mathtt{st}_{i+1}$ where $\mathtt{st}_{i+1} = f(\mathtt{st}_i)$ and f is determined by the description of the computation. Kalai et al. constructed a quasi argument that is also an argument for the correct computation of y. The local extractor can extract any pair of consecutive states and they showed that any of such states must be consistent with the input, meaning that on input x an honest computation reaches such state in the corresponding number of steps. Note that consistency can be efficiently checked as long as the computation is polynomial time.

Consider an extractor programmed for retrieving $\mathtt{st}_0, \mathtt{st}_1$, i.e. locality parameter $K = 2|\mathtt{st}|$, where $|\mathtt{st}|$ is a bound on the size of the states. Local soundness asserts that state $\mathtt{st}_0$ is consistent with x. Local soundness also implies that $\mathtt{st}_1$ is consistent with $\mathtt{st}_0$ and hence with x (note that the statement $\mathtt{st}_1 = f(\mathtt{st}_0)$ depends only on local variables). Now, to show that $\mathtt{st}_2$ is also consistent, we jump to another game where first the extractor computes only $\mathtt{st}_1$, and in the next game the extractor computes $\mathtt{st}_1, \mathtt{st}_2$. The crucial observation is that $\mathtt{st}_1$ should be still consistent with x in both games. Otherwise, we can distinguish between the common output of extractors for $\mathtt{st}_0, \mathtt{st}_1$ and $\mathtt{st}_1$ or between $\mathtt{st}_1$ and $\mathtt{st}_1, \mathtt{st}_2$, which contradicts the no-signaling property. Similarly, consistency of $\mathtt{st}_1$ and local soundness imply that $\mathtt{st}_2$ is also consistent, and so on up to $\mathtt{st}_d$.

The issue is that setting $K = O(|\mathtt{st}|)$ yields a proof whose size is linear in the space of the computation and hence not succinct. To overcome this bounded-space limitation, Kalai et al. used hash-trees to construct a proof that a RAM machine (which can emulate an unbounded-space machine) transitions between two configurations $\mathtt{cf}, \mathtt{cf}'$. Essentially, the verifier is only given digests $\mathtt{h}, \mathtt{h}'$ of the states, which act as an aggregated form of the state in each time step. Kalai et al.'s hash-tree construction guarantees that an adversary can't produce a configuration $\mathtt{cf}$, digests $\mathtt{h}, \mathtt{h}'$ and valid proof such that $\mathtt{h} = \mathsf{Hash}(\mathtt{cf})$ but $\mathtt{h}' \neq \mathsf{Hash}(\mathtt{cf}')$, where $\mathtt{cf}' = f(\mathtt{cf})$ and f is the transition function of the RAM machine.

Note that the verification procedure of such proof is a computation with space bounded by $\mathsf{poly}(\kappa)$. Then, we can give a quasi-argument for the RAM delegation. That is, a quasi-argument of knowledge of $\mathtt{h}_1, \mathtt{h}_2, \ldots, \mathtt{h}_\mathsf{d} \in \{0,1\}^{\mathsf{poly}(\kappa)}$ such that, if $\mathtt{h}_{i-1} = \mathsf{Hash}(\mathtt{cf}_{i-1})$ then $\mathtt{h}_i = \mathsf{Hash}(f(\mathtt{cf}_{i-1}))$. However this comes at the cost of reducing the verification procedure of the hash-tree to a 3CNF formula. While the size of the final formula is still polynomial in the security parameter, in practice it has a considerable size which has direct impact on the size of the common reference string and the prover's computation.

2.2 Structure Preserving Delegation for Bounded-Depth Circuits.

González and Ràfols [GR19] constructed a delegation scheme with proof-size $O(d\kappa)$ and verification requiring n plus $O(d)$ cryptographic operations. Interestingly, the verification procedure of [GR19] can be described completely as a set of pairing product equations. As shown by Abe et al.[AFG+16], cryptographic primitives whose correctness can be stated as equations over bilinear groups are more suited for practically efficient arguments without resorting to generic reductions to a circuit or a 3CNF formula.

In the heart of the delegation scheme of [GR19] lie two of the so called "knowledge transfer arguments" with the following property. For a commitment C_1 and an opening x, such an argument allows to prove that some other commitment C_2 opens to $f(x)$, for some function f, even if C_2 is not extractable. The first of such arguments is a succinct (proofs of size $O(\kappa)$) knowledge transfer argument for linear functions. Soundness is shown under the hardness of the $\mathcal{G}^\top$-MDDH assumption, where $\mathcal{G}$ is the distribution of the matrix $\mathbf{G} \in \mathbb{Z}_p^{k\times n}$ containing the discrete logarithms of C_1's commitments keys. Note that when $\mathbf{G}$ is uniform (e.g. Pedersen commitments), the $\mathcal{G}^\top$-MDDH assumption can be reduced to DDH in asymmetric groups.[4] In the second argument, the function is the hadamard product $f(\boldsymbol{a}, \boldsymbol{b}) = \boldsymbol{a} \circ \boldsymbol{b}$ and security is based on the hardness of the "$\mathcal{R}$-Rational Strong Diffie Hellman" assumption. In contrast to the linear argument, the quadratic argument requires a specific distribution for the commitment key where the $k > 1$ rows of $\mathbf{G}$ are the result of evaluating n lagrangian polynomials at k different random points. As a result, when using linear and quadratic arguments together, the linear argument is based on the so called "q-Lagrangian assumption".

To delegate the computation of an arithmetic circuit, the multiplication gates are partitioned in d levels. Each level groups the gates at the same distance from the inputs, without counting linear gates. In this way, the inputs of level $i+1$ are linear combinations of outputs of the i previous levels. A prover commits to the left, right, and output wires of each level as L_i, R_i, O_i. In the first d arguments f is a linear function and the argument handles the linear relations between the input wires (the openings of L_i, R_i) of level i and the output wires of all previous levels (the openings of $O_1, \ldots, O_{i-1}$). In the next d arguments f is the hadamard product so that the opening of O_i is the the hadamard product of the openings of L_i and R_i. The fact that the verifier can check the commitment to the first level using the public input and a simple inductive argument over the levels shows that the output must be correct.

Although we don't explicitly construct it, underlying our quasi arguments lies a delegation scheme with proof size linear in d which closely follows [GR19]. The main difference is that we use uniform commitment keys that allows us to rely only on constant-size assumptions. In symmetric groups, when $\mathbf{G}$ is uniformly distributed over $\mathbb{Z}_p^{2\times n}$ the $\mathcal{G}^\top$-MDDH can be easily reduced to DLin. In asymmetric groups we can take $\mathbf{G} \leftarrow \mathbb{Z}_p^{1\times n}$ (a pedersen commitment key) and rely on DDH. We additionally modify the hadamard argument as follows. We first give a "kronecker argument" such that, if L_i opens to $\boldsymbol{a}_i$ and R_i opens to $\boldsymbol{b}_i$, then S_i opens to $\mathbf{a}_i \otimes \mathbf{b}_i$. Then we give a linear knowledge transfer argument from S_i to O_i, a commitment to $\boldsymbol{a}_i \circ \boldsymbol{b}_i$, using the fact that there's a linear relation between $\boldsymbol{a} \otimes \boldsymbol{b}$ and $\boldsymbol{a} \circ \boldsymbol{b}$.

The advantage of the kronecker argument is that, when commitments are (vectors of) bilinear group elements, the verifier can check if S_i is correct with probability 1 and without any proof. If we restrict to "algebraic commitments", the commitment keys are matrices defined over $\mathbb{G}$ and commitments L_i, R_i are group encodings of $\mathbf{U}\boldsymbol{a}_i$ and $\mathbf{V}\boldsymbol{b}_i$ respectively. If we define the commitment key of S_i to be the encoding of $\mathbf{U} \otimes \mathbf{V}$ in the base group, then e provides a trivial way of testing that O_i opens to $\boldsymbol{a}_i \otimes \boldsymbol{b}_i$. It suffices to check that $e(S_i, \mathcal{P}) = e(L_i, R_i)$, where $\mathcal{P}$ is a generator of $\mathbb{G}$, and then S_i opens to $\boldsymbol{a}_i \otimes \boldsymbol{b}_i$ since $\log S_i =$

[4] In symmetric bilinear groups the DDH assumptions is false. However, using Pedersen commitments of size 2 yields security based on the DLin assumption.

$(\mathbf{U} \otimes \mathbf{V})(\boldsymbol{a}_i \otimes \boldsymbol{b}_i)$. The last step is to show the hardness of the $(\mathbf{U} \otimes \mathbf{V})^\top$-MDDH, which is necessary for the soundness of the linear knowledge transfer argument from S_i to O_i. We show that $(\mathbf{U} \otimes \mathbf{V})^\top$-MDDH can be reduced to the $\mathbf{U}^\top$-MDDH and $\mathbf{V}^\top$-MDDH assumptions (and hence to DLin when the matrices are uniform).

2.3 No-Signaling Somewhere Statistically Binding Commitments/Hashing

Somewhere statistically binding (SSB) hashing/commitments[5] were introduced by Hubacek an Wichs [HW15] and then improved by [OPWW15], and have been used for constructing efficient NIZK proofs [GHR15a, GR16] as well as ring signatures [BDH+19]. SSB commitments are a generalization of dual-mode commitments where the commitment key can be sampled from many computationally indistinguishable distributions, each of which is statistically binding for some part of size ℓ of the input. Known SSB commitments constructions are also extractable.[6] That is, there exists an efficient procedure for retrieving the local opening $\boldsymbol{x}_S = (x_i : i \in S)$ from any commitment to $x_1, \ldots, x_n$, whenever the commitment keys are perfectly binding in $S \subseteq [n]$.

We note that the SSB extractor has many similarities with a no-signaling extractor. First, extractability of the local opening is just a local soundness guarantee (with locality parameter ℓ). Additionally, indistinguishability of the commitment keys is a weaker form of the no-signaling property. Indeed, a no-signaling extractor must produce commitment keys which are indistinguishable between them. Otherwise a distinguisher for sets S, S' can be used for wining in the no-signaling game even without the extracted value.

Although we don't know if any SSB commitment is also no-signaling, we show a strong connection between the two notions. First, we show that K extractable SSB commitments with locality $\ell = 1$ can be straightforwardly used to construct a no-signaling SSB commitment with locality K. However, this construction is not as efficient as directly using an SSB commitment with locality $\ell = K$. At least for known SSB constructions such as [GHR15a] and [FLPS20] the size of each commitment with locality $\ell = K$ is $K + k$ elements of a bilinear group, where k is the size of the underlying matrix assumption. In the simple construction the size of K commitments with locality 1 is $K(k+1)$, which amounts to $k(K-1)$ more group elements. Consequently, we ask whether SSB commitments with locality $\ell = K$ are also no-signaling. We show this is the case if the SSB commitment has also an *oblivious trapdoor generation procedure*.

2.3.1 SSB Commitments with Oblivious Trapdoor Generation.

We strengthen the key and trapdoor generation to be *oblivious*, and for short we will say oblivious SSB commitment. Intuitively, this notion captures that, for any subset S' of the larger set S of binding coordinates, the key generation algorithm can generate the commitment key for S' and a trapdoor for S' obliviously of $S \setminus S'$. That is, given only a commitment key generated for S and the description of S', which should (computationally) hide any information about $S \setminus S'$, the oblivious key generation algorithm outputs an identically distributed key together with a trapdoor for extracting $x_{S'}$. Intuitively, the key generation algorithm is oblivious of $S \setminus S'$ (it might even be that $S \setminus S' = \emptyset$) because the commitment keys are indistinguishable.

Oblivious SSB implies no-signaling SSB. It turns out that this notion also implies that the extracted value can't "signal" any information about $S \setminus S'$. Concretely, we show that any oblivious SSB commitment has a no-signaling extractor. Consider the extractor which on input S generates the commitment key binding at S together with some trapdoor τ. When the extractor receives some commitment from the adversary it uses the trapdoor to produce a local opening. Local consistency follows directly from local extractability. For showing the extractor no-signaling, we need to show that given two sets $S, S' \subseteq [n]$ where $S' \subseteq S$, the

[5] Through this paper we will refer to "commitments" while technically they are "hashes". We do so because in the context of NIZK proofs is traditional to commit to the witness and then prove that the committed value satisfy some relation. However, since we are less interested in zero-knowledge, the randomness of such commitments is 0 (or fixed/inexistent) and we end up with hashes.

[6] In the context of bilinear groups, we can consider f-extraction where one only extracts f applied to the witness. In particular, it is usual to consider f the (one-way) function that maps elements in $\mathbb{Z}_p$ to one of the base groups $\mathbb{G}_1$ or $\mathbb{G}_2$.

following two experiments are indistinguishable. In the first experiment the extractor is called on input S and when the extractor outputs some x_S, the experiment outputs $x_{S'}$. In the second experiment the extractor is called on input S' and the output of the experiment is whatever the extractor outputs. Now we construct a sequence of intermediate experiments by doing incremental modifications to the first one until we end up with the second.

The first intermediate experiment is as the first experiment but the extractor additionally calls the oblivious key generation on input S' and the original commitment key. It outputs a new identically distributed commitment key together with a trapdoor. Now the extractor extracts directly $x_{S'}$ using the obliviously generated trapdoor, which should be the same as extracting x_S and returning $x_{S'}$. Since the new commitment key follows the same distribution as the original key, the output of this modified experiment should be indistinguishable from the first experiment. Now we use the obliviousness of the key generation algorithm and change the commitment key given to the oblivious key generation algorithm with a key generated for the set S'. The indistinguishability of commitment keys implies that the output of the experiment is still the same. We finally get rid of the oblivious key generation algorithm and end up with experiment two, showing that the extractor is no-signaling.

2.3.2 Constructing Oblivious SSB Commitments.

First lets see that K SSB commitments with locality parameter 1 can be used to construct an oblivious SSB commitment with locality K (and hence it has a no-signaling extractor). For a set $S = \{s_1, \ldots, s_t\}$ the commitment key is just K commitment keys $ck_1, \ldots, ck_K$ for sets $\{s_1\}, \ldots, \{s_t\}$, complementing with extra $\{s_{t+1}\}, \ldots, \{s_K\}$ if necessary. To commit to some $\boldsymbol{x} \in \mathbb{Z}_p^n$ one simply gives $c_1 = \mathsf{Com}_{ck_1}(\boldsymbol{x}), \ldots, c_K = \mathsf{Com}_{ck_K}(\boldsymbol{x})$. Extraction of each x_{s_i} is done using c_{s_i} and the trapdoor τ_{s_i}, independently of the others. The oblivious extractor on input the commitment keys for some unknown S and the description of $S' \subseteq S$ just re-samples the commitment keys for S'.[7] Since it doesn't matter if the trapdoors for positions $i \notin S'$ are not known, we have that this trivial extractor can obliviously generate the trapdoor $\{\tau_i : i \in S'\}$.

Efficient Oblivious SSB Commitments. To construct more efficient SSB commitments with oblivious trapdoor generation we use the implicit SSB commitments of [GHR15a] later generalized in [FLPS20]. For $x \in \mathbb{Z}_p$, we write $[x]$ to denote $x\mathcal{P}$. For message space $\mathbb{Z}_p^n$, locality parameter $K \in \mathbb{N}$ and a subset $S \subseteq [n]$ of size $t \leq K$, the commitment key is defined by $[\mathbf{G}]$ where $\mathbf{G} = (\mathbf{G}_S | \mathbf{G}_{\overline{S}})\mathbf{P}$ and

$$\mathbf{G}_S \leftarrow \mathbb{Z}_p^{(K+1)\times t}, \mathbf{G}_{\overline{S}} = \mathbf{G}_0\boldsymbol{\Gamma}, \mathbf{G}_0 \leftarrow \mathbb{Z}_p^{(K+1)\times(K+1-t)}, \boldsymbol{\Gamma} \leftarrow \mathbb{Z}_p^{(K+1-t)\times(d-t)}.$$

Matrix $\mathbf{P} \in \{0,1\}^{d\times d}$ is a permutation matrix associated to S such that $\mathbf{P}\boldsymbol{e}_{s_i} = \boldsymbol{e}_i$, for $i \leq t$ and $\boldsymbol{e}_i$ the i-th vector of the canonical basis. A commitment to $\boldsymbol{x} \in \mathbb{Z}_p^d$ is computed as $[\boldsymbol{c}] = [\mathbf{G}]\boldsymbol{x} = [\mathbf{G}_S|\mathbf{G}_{\overline{S}}]\mathbf{P}\boldsymbol{x} = [\mathbf{G}_S]\boldsymbol{x}_S + [\mathbf{G}_{\overline{S}}]\boldsymbol{x}_{\overline{S}}$. Note that the columns of $\mathbf{G}_S$ are linearly independent from the columns of $\mathbf{G}_{\overline{S}}$ with overwhelming probability, since $\mathbf{Im}(\mathbf{G}_{\overline{S}}) \subseteq \mathbf{Im}(\mathbf{G}_0)$ and $(\mathbf{G}_S|\mathbf{G}_0)$ is a basis of $\mathbb{Z}_p^{K+1}$ w.o.p.

This distribution of commitment keys implies that some parts of the input in S go to the space spanned by $\mathbf{G}_S$ of dimension t, while the other part is mapped to the space spanned by $\mathbf{G}_0$ of dimension $K+1-t$. Since $\mathrm{rank}(\mathbf{G}_S) = t$ with overwhelming probability, all the information of $\boldsymbol{x}_S \in \mathbb{Z}_p^t$ can be retrieved from $\boldsymbol{c}$. Even more, there exists an efficiently computable trapdoor $\mathbf{T}_S \in \mathbb{Z}_p^{(K+1)\times t}$ such that $\mathbf{G}_S^\top \mathbf{T}_S = \mathbf{I}_{t\times t}$ and $\mathbf{G}_{\overline{S}}^\top \mathbf{T}_S = \mathbf{0}_{(d-t)\times t}$, and hence $\mathbf{T}_S^\top[\boldsymbol{c}] = [\boldsymbol{x}_S]$. Note that this shows also that the commitment is SSB. The indistinguishability of commitment keys can be shown with a tight reduction to the DDH assumption as in [FLPS20].

Oblivious Trapdoor Generation. One of the main technical contributions of this work is an oblivious trapdoor generator for this commitment scheme. The algorithm receives a set S' of size t' and a commitment

[7] Actually, the oblivious key generation needs to know which of the commitments keys $ck_1, \ldots, ck_K$ are perfectly binding for $s' \in S'$. Nevertheless, it should be still oblivious of whether the rest of commitment keys are binding or not. See i section 4.2 for more details.

key $[\mathbf{G}]$ sampled for being binding at some unknown $S \supseteq S'$. The procedure must compute a new commitment key $[\mathbf{H}]$ distributed as $[\mathbf{G}]$ together with the trapdoor $\mathbf{T}_{S'}$. Since we know that columns in S' are uniform, we could pick $\mathbf{H}_{S'} \leftarrow \mathbb{Z}_p^{(K+1)\times t'}$ and solve $\mathbf{H}_{S'}^\top \mathbf{T}_{S'} = \mathbf{I}_{t'\times t'}$ for some $\mathbf{T}_{S'}$ (note that there are many such $\mathbf{T}_{S'}$ if $t' \leq K$). However, since we don't know the distribution of $[\mathbf{G}_{\overline{S}'}]$ the only hope is to define $[\mathbf{H}_{\overline{S}'}] = [\mathbf{G}_{\overline{S}'}]$ and try to find some $\mathbf{T}_{S'}$ such that $\mathbf{G}_{\overline{S}'}^\top \mathbf{T}_{S'} = \mathbf{0}_{(d-t')\times t'}$. Unfortunately, this amounts to find an element in the kernel of $[\mathbf{G}_{\overline{S}'}]^\top$ which is in general a computationally hard problem [MRV16].

Instead we make the following observation. Regardless of $S \setminus S'$, the t' lower rows of $\mathbf{G}_{\overline{S}}$ can be always written as a random linear combination of the first $K+1-t'$ rows. That is $\mathbf{G}_{\overline{S}'} = \left(\begin{smallmatrix} \mathbf{A} \\ \mathbf{RA} \end{smallmatrix}\right)$, where $\mathbf{A} \in \mathbb{Z}_p^{K+1-t'\times d-t'}$ and $\mathbf{R} \leftarrow \mathbb{Z}_p^{t'\times K+1-t'}$. In this case it is possible to compute $\mathbf{T}_{S'} = \left(\begin{smallmatrix} -\mathbf{R}^\top \mathbf{C} \\ \mathbf{C} \end{smallmatrix}\right)$, for any $\mathbf{C} \in \mathbb{Z}_p^{t'\times t'}$, which satisfies $\mathbf{G}_{\overline{S}'}^\top \mathbf{T}_{S'} = \mathbf{0}_{(d-t')\times t'}$. Adding the restriction $\mathbf{H}_{S'}^\top \mathbf{T}_{S'} = \mathbf{I}_{t'\times t'}$ yields the desired trapdoor.

Lets see that the previous observation holds. Indeed, this is the case for $\mathbf{G}_0 \in \mathbb{Z}_p^{(K+1)\times(K+1-t)}$ since the upper part $\overline{\mathbf{G}}_0$ is a random matrix with more rows than columns and hence $\mathbf{R}\overline{\mathbf{G}}_0$, for $\mathbf{R} \leftarrow \mathbb{Z}_p^{t'\times(K+1-t')}$, is uniformly distributed. This is also valid for all non-binding coordinates since $\mathbf{G}_{\overline{S}} = \mathbf{G}_0\mathbf{\Gamma}$ and then the lower rows follow distribution $\mathbf{R}\overline{\mathbf{G}}_{\overline{S}}$. The same is true for $\mathbf{G}_{S\setminus S'} \in \mathbb{Z}_p^{(K+1)\times(t-t')}$, that is $\mathbf{R}'\overline{\mathbf{G}}_{S\setminus S'}$ is uniform when $\mathbf{R}' \leftarrow \mathbb{Z}_p^{t'\times(K+1-t')}$. Now we show that using the same matrix $\mathbf{R}$ doesn't alter the distribution and in fact $\mathbf{R}\overline{\mathbf{G}}_0$ is independent from $\mathbf{R}\overline{\mathbf{G}}_{S\setminus S'}$. Since the columns of $\overline{\mathbf{G}}_0 \in \mathbb{Z}_p^{(K+1-t')\times(K+1-t)}$ and of $\overline{\mathbf{G}}_{S\setminus S'} \in \mathbb{Z}_p^{(K+1-t')\times(t-t')}$ form a basis of $\mathbb{Z}_p^{K+1-t'}$, the matrix $\mathbf{R}^\top$ can be decomposed into two independent components: a random element in $\mathrm{Im}(\overline{\mathbf{G}}_{S\setminus S'}^\perp)$ and another in $\mathrm{Im}(\overline{\mathbf{G}}_0^\perp)$. This shows that $\mathbf{R}\overline{\mathbf{G}}_0 = \mathbf{R}_2(\mathbf{G}_{\overline{S}\setminus S'}^\perp)^\top \overline{\mathbf{G}}_0$ and $\mathbf{R}\overline{\mathbf{G}}_{S\setminus S'} = \mathbf{R}_1(\mathbf{G}_0^\perp)^\top \overline{\mathbf{G}}_{S\setminus S'}$ are independent and then $\left(\begin{smallmatrix} \overline{\mathbf{G}}_{S\setminus S'} & \overline{\mathbf{G}}_0\mathbf{\Gamma} \\ \mathbf{R}\overline{\mathbf{G}}_{S\setminus S'} & \mathbf{R}\overline{\mathbf{G}}_0\mathbf{\Gamma} \end{smallmatrix}\right)$ is correctly distributed.

2.4 Quasi-Arguments of Membership in a Linear Space

Quasi-Adaptive NIZK (QA-NIZK) proofs are NIZK proofs where the CRS is allowed to depend on the specific language for which proofs have to be generated [JR13]. Such language dependent preprocessing boosts efficiency leading to proofs of size as short as a single group element [JR14].

However, QA-NIZK arguments are in general not arguments of knowledge. Although it is possible to show that they are arguments of knowledge in the generic/algebraic group model [CFQ19], nothing is known under falsifiable assumptions. Actually it is quite plausible that results for the feasibility of SNARKs under falsifiable assumption also apply to arguments of knowledge for linear spaces. Hence, we can only look for relaxed notions of arguments of knowledge such as the quasi-arguments of Kalai et al.

A first attempt to define quasi-arguments for linear spaces is to relax the argument of knowledge: extract only a small part of $\boldsymbol{w}$ such that $[\boldsymbol{x}] = [\mathbf{U}]\boldsymbol{w}$. However, if $\mathbf{U}$ follows an arbitrary distribution, such local witness might not even be defined. It is not hard to see that one should require some local binding property for $\mathbf{U}$ which is exactly the case of SSB commitments we have just seen. Instead, we define a quasi-argument of knowledge of some vector $[\boldsymbol{x}] \in \mathbb{G}^\ell$ belonging to the image of a matrix $[\mathbf{U}] \in \mathbb{G}^{\ell\times n}$.

We use Kiltz at Wee argument of membership in linear spaces [KW15] to construct a quasi argument for linear relations. We show that there is a local and no-signaling extractor which given some $S \subseteq [d]$ of size $t \leq K$ extracts $[\boldsymbol{x}_S] \in \mathrm{Im}([\mathbf{U}_S])$, where $\boldsymbol{x}_S \in \mathbb{Z}_p^t$ is the vector whose entries are x_i and $\mathbf{U}_S \in \mathbb{Z}_p^{t\times n}$ is the matrix whose rows are the rows of $\mathbf{U}$ indexed by i, where i ranges over S in some fixed order.

2.4.1 The argument.

Our construction is Kiltz and Wee linear membership argument [KW15] for the matrix $[\mathbf{GU}]$, where $\mathbf{G}$ is an SSB commitment key with locality parameter K. For simplicity, here we consider here the argument with proof size $k+1$ of [KW15] but our construction is also sound for the more efficient instantiation of size k.

The argument is essentially a hash proof system [CS02] with public verifiability. For a secret hash key $\mathbf{K} \leftarrow \mathbb{Z}_p^{(K+1)\times(k+1)}$ the crs contains the projection $[\mathbf{B}] = [\mathbf{U}^\top \mathbf{G}^\top \mathbf{K}]$ from which a proof that $\boldsymbol{c} = \mathbf{GU}\boldsymbol{w}$ is

computed as $[\boldsymbol{\pi}] = \boldsymbol{w}^\top[\mathbf{B}] = [\boldsymbol{c}^\top]\mathbf{K}$. Secret verification is just $[\boldsymbol{\pi}] = [\boldsymbol{c}^\top]\mathbf{K}$ and is sound because for any $\boldsymbol{c} \notin \mathrm{Im}(\mathbf{GU})$ the value $\boldsymbol{c}^\top\mathbf{K}$ is completely random. To publicly verify proofs the crs additionally contains a "partial commitment" to the secret key $[\mathbf{C}] = [\mathbf{KA}]$ plus $[\mathbf{A}]$, where $\mathbf{A} \in \mathbb{Z}_p^{(k+1)\times k}$ is sampled from some matrix distribution $\mathcal{D}_k$. The verifier checks whether $e([\boldsymbol{\pi}], [\mathbf{A}]) = e([\boldsymbol{c}^\top], [\mathbf{C}])$ and, while now there exist proofs satisfying the verification equation for false statements with high probability, cheating proofs can be used to compute elements in the kernel of $\mathbf{A}^\top$ (i.e. breaking the $\mathcal{D}_k^\top$-KerMDH assumption of [MRV16]).

2.4.2 Local and No-Signaling extraction.

Our strategy to prove local soundness is to show that, apart from extracting $[\boldsymbol{x}_S]$ from $[\boldsymbol{c}]$, we are also able to produce a verifying proof $[\boldsymbol{\pi}^\dagger]$ that $[\boldsymbol{x}_S] \in \mathbf{Im}(\mathbf{U}_S)$. More concretely, on input a crs $[\mathbf{A}^\dagger], [\mathbf{B}^\dagger], [\mathbf{C}^\dagger]$, we can construct another crs that is statistically close to the original and, more importantly, we can extract $[\boldsymbol{x}_S]$ and $[\boldsymbol{\pi}^\dagger]$ satisfying the corresponding verification equation.

The high-level idea for the proof of local soundness is to embed the public parameters $[\mathbf{A}^\dagger], [\mathbf{B}^\dagger], [\mathbf{C}^\dagger]$ of the local linear space argument for $\mathbf{U}_S$ in the larger one. Although the secret hash key $\mathbf{K}^\dagger$ of the local linear argument is statistically hidden, we can still pick a random hash key for all the coordinates by picking another secret key and implicitly define the full secret key as some composition of the two keys. Concretely, given the trapdoor $\mathbf{T}_S$ for locally opening SSB commitments we implicitly define $\mathbf{K} = \mathbf{T}_S\mathbf{K}^\dagger + \mathbf{R}$, where $\mathbf{R}$ is the additional key, so that the proofs for $\boldsymbol{c} = \mathbf{GP}\left(\begin{smallmatrix}\boldsymbol{x}_S\\ \boldsymbol{x}_{\overline{S}}\end{smallmatrix}\right) = \mathbf{G}_S\boldsymbol{x}_S + \mathbf{G}_{\overline{S}}\boldsymbol{x}_{\overline{S}}$ are of the form $\boldsymbol{\pi} = \boldsymbol{c}^\top\mathbf{K} = (\mathbf{G}_S\boldsymbol{x}_S + \mathbf{G}_{\overline{S}}\boldsymbol{x}_{\overline{S}})^\top(\mathbf{T}_S\mathbf{K}^\dagger + \mathbf{R}) = \boldsymbol{x}_S^\top\mathbf{K}^\dagger + \boldsymbol{c}^\top\mathbf{R}$. In this way a proof for the local argument can be retrieved as $[\boldsymbol{\pi}^\dagger] = [\boldsymbol{\pi}] - [\boldsymbol{c}^\top]\mathbf{R}$. This equivalent way of sampling $\mathbf{K}$ allows to compute the crs of the larger linear argument using only $[\mathbf{A}^\dagger], [\mathbf{B}^\dagger], [\mathbf{C}^\dagger]$ and $\mathbf{T}_S, \mathbf{R}$. Indeed, we can define $[\mathbf{A}] = [\mathbf{A}^\dagger]$, $[\mathbf{B}] = [\mathbf{B}^\dagger] + [\mathbf{U}^\top\mathbf{G}^\top]\mathbf{R}$ and $[\mathbf{C}] = \mathbf{T}_S[\mathbf{C}^\dagger] + \mathbf{R}[\mathbf{A}^\dagger]$.

We also show that the crs is indistinguishable for different sets and that there is an oblivious trapdoor generation strategy, and hence we also have a no-signaling extraction strategy. The indistinguishability of the crs follows directly from the indistinguishability of SSB commitment keys if additionally the matrix $[\mathbf{U}]$ is witness samplable (i.e. one can sample the discrete log of $\mathbf{U}$) which is usually the case. The trapdoor can be computed from the oblivious SSB trapdoor generation.

2.4.3 Extension to Knowledge Transfer, Bilateral Spaces and Sum Arguments.

We constructed a local extractor which on input a set $S \subseteq [d]$ extracts some local opening $[\boldsymbol{x}_S]$ and a proof $[\boldsymbol{\pi}^\dagger]$ that $\boldsymbol{x}_S \in \mathbf{Im}(\mathbf{U}_S)$. While this form of local soundness is enough for constructing delegation for bounded-width circuits, in the general case we require the following variation of local soundness. The statement is split in two parts, $[\boldsymbol{x}]$ and $[\boldsymbol{y}]$, as well as the matrix generating the linear space is split in $[\mathbf{U}]$ and $[\mathbf{V}]$ such that $\left(\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{smallmatrix}\right) = \left(\begin{smallmatrix}\mathbf{U}\\ \mathbf{V}\end{smallmatrix}\right)\boldsymbol{w}$. For $S_1, S_2 \subset [d]$, the adversary is requested to produce some $\boldsymbol{w}^*$ such that $\boldsymbol{x}_{S_1} = \mathbf{U}_{S_1}\boldsymbol{w}^*$ but $\boldsymbol{y}_{S_2} \neq \mathbf{V}_{S_2}\boldsymbol{w}^*$. In [GR19] it is shown that, provided the $\mathbf{U}_{S_1}^\top$-MDDH assumption is hard, a QA-NIZK proof that $\left(\begin{smallmatrix}\boldsymbol{x}_{S_1}\\ \boldsymbol{y}_{S_2}\end{smallmatrix}\right) \in \mathbf{Im}\left(\begin{smallmatrix}\mathbf{U}_{S_1}\\ \mathbf{V}_{S_2}\end{smallmatrix}\right)$ implies that such an adversary wins only with negligible probability.[8] Since we can also extract a proof $[\boldsymbol{\pi}^\dagger]$ for $\left(\begin{smallmatrix}\boldsymbol{x}_{S_1}\\ \boldsymbol{y}_{S_2}\end{smallmatrix}\right)$, we conclude that the adversary can't cheat.

This argument was called an argument of knowledge transfer by González and Ràfols [GR19] while Kalai et al. [KPY19] used a similar soundness property in their delegation for RAM machines. For knowledge transfer arguments, in general, the security only holds if the matrix $\mathbf{A}$ has more rows than columns, and hence proof size is at least $k+1$. If $k > 1$ we can also prove soundness with proof size k using an additional decisional assumption.

Another variant given in [GHR15a], and extended to knowledge transfer arguments in [GR19], considers the statement as well as the matrix split between the two groups. We call this argument a linear argument for bilateral spaces. We consider a particular type of bilateral linear spaces defined in [GHR15a] and called "sum in subspace argument". The statement is $[\boldsymbol{x}]_1, [\boldsymbol{y}]_2$ and soundness means that $\boldsymbol{x} + \boldsymbol{y} \in \mathrm{Im}(\mathbf{M} + \mathbf{N})$ given $[\mathbf{M}]_1, [\mathbf{N}]_2$. We construct quasi arguments for all these variants with knowledge transfer soundness.

[8] See section 3.3 .

Luckily, the constructions as well as the security proofs are minor modifications of the original argument. Security is based on constant-size assumptions.

2.5 Quasi-Argument of Hadamard Products

We show that the "bit-string" argument of [GHR15a] was implicitly a quasi-argument for the set of equations $b_i(b_i - 1) = 0$, for all $i \in [d]$. It will be convenient to directly work with equations of the form $x_i y_i = z_i$, that is $\boldsymbol{x} \circ \boldsymbol{y} = \boldsymbol{z}$ where $\circ$ denotes the hadamard product, instead of the bit-string argument equations.

The reference string in [GHR15a] contains what we interpret as two SSB commitment keys $[\mathbf{G}] \in \mathbb{G}^{(k+1)\times d}, [\mathbf{H}] \in \mathbb{G}^{(k+1)\times d}$ with locality parameter $K = 1$. The crs additionally includes the product $[\mathbf{G} \otimes \mathbf{H}]$ so that a quasi argument of knowledge of $[\boldsymbol{x}] \in \mathbb{G}^d, [\boldsymbol{y}] \in \mathbb{G}^d, [\boldsymbol{z}'] \in \mathbb{G}^{d^2}$ such that $\boldsymbol{z}' = \boldsymbol{x} \otimes \boldsymbol{y}$, i.e. a kronecker product, is just $[\boldsymbol{c}] = [\mathbf{G}]\boldsymbol{x}, [\boldsymbol{d}] = [\mathbf{H}]\boldsymbol{y}$ and $[\boldsymbol{t}] = [\mathbf{G} \otimes \mathbf{H}]\boldsymbol{z}'$. A verifier should check that $[\boldsymbol{c}] \otimes [\boldsymbol{d}] = e([\boldsymbol{t}], [1])$, where $\otimes$ is naturally defined in terms of the pairing function. Note that this is locally extractable for a set $S = \{i, j\}$ since $[x_i] = \mathbf{T}_i[\boldsymbol{c}], [y_j] = \mathbf{T}'_j[\boldsymbol{d}]$ and $[z'_{n(i-1)+j}] = (\mathbf{T}_i \otimes \mathbf{T}'_j)[\boldsymbol{t}]$, where $\mathbf{T}_i, \mathbf{T}'_j$ are the trapdoors for locally open the respective SSB commitment at coordinates i, j. Moreover, in section 4.3 we show that $[\boldsymbol{t}]$ is also a no-signaling SSB commitment.

To show that some $[\boldsymbol{f}] = [\mathbf{F}]\boldsymbol{z}$, for commitment key $[\mathbf{F}] \in \mathbb{G}^{k+1\times d}$, opens to $\boldsymbol{z} = \boldsymbol{a} \circ \boldsymbol{b}$ we use the fact that there's a linear relation between $\boldsymbol{a} \circ \boldsymbol{b}$ and $\boldsymbol{a} \otimes \boldsymbol{b}$. Hence we show that $\left(\begin{smallmatrix} \boldsymbol{t} \\ \boldsymbol{f} \end{smallmatrix}\right) \in \mathbf{Im}\left(\begin{smallmatrix} \mathbf{G}\otimes\mathbf{H} \\ \mathbf{F}\cdot\mathsf{Had} \end{smallmatrix}\right)$, where Had is a matrix such that $\boldsymbol{x} \circ \boldsymbol{y} = \mathsf{Had}(\boldsymbol{x} \otimes \boldsymbol{y})$.

2.5.1 Local and No-Signaling Extraction.

The argument of the hadamard product is locally extractable for any set $\{i, j, k\}$ and sound when $i = j = k$ (otherwise local soundness holds vacuously). We can extract $[x_i] = \mathbf{T}_i[\boldsymbol{c}], [y_i] = \mathbf{T'}_i[\boldsymbol{d}], [z^\dagger] = \mathbf{T}''_i[\boldsymbol{f}]$ as well as $[z_i] = (\mathbf{T}_i \otimes \mathbf{T}'_i)[\boldsymbol{t}]$ such that $z_i = x_i y_i$. Assume for the sake of a contradiction that $z^\dagger \neq z_i$. Since $\boldsymbol{g}_i, \boldsymbol{h}_i, \boldsymbol{f}_i$ are linearly independent from the other columns in $\mathbf{G}, \mathbf{H}, \mathbf{F}$, respectively, if $[\boldsymbol{c}], [\boldsymbol{d}], [\boldsymbol{t}]$ satisfies $[\boldsymbol{c}] \otimes [\boldsymbol{d}] = e([\boldsymbol{t}], [1])$, then the unique openings at coordinate i satisfy $z_i = x_i y_i$. Since also $\boldsymbol{g}_i \otimes \boldsymbol{h}_i$ and $\boldsymbol{f}_i$ are linearly independent from the other columns in the respective commitment keys, it holds that $\left(\begin{smallmatrix} \boldsymbol{t} \\ \boldsymbol{f} \end{smallmatrix}\right)$ does not belong to the span of the matrix $\left(\begin{smallmatrix} \mathbf{G}\otimes\mathbf{H} \\ \mathbf{F}\cdot\mathsf{Had} \end{smallmatrix}\right)$. Hence, $z^\dagger$ must be equal to $x_i y_i$ or we can break soundness of linear argument.

2.5.2 Extension to Knowledge Transfer Arguments.

We extend the quasi-argument local soundness to offer a "knowledge transfer" guarantee. That is, we can extract $[x_i], [y_i], [z_i]$ and the adversary can't also produce an opening $\boldsymbol{a}, \boldsymbol{b}$ such that $x_i = \mathbf{U}_i\boldsymbol{a}, y_i = \mathbf{V}_i\boldsymbol{b}$ but $z_i \neq \mathbf{W}_i\boldsymbol{a} \circ \boldsymbol{b}$. Matrices $[\mathbf{U}_i], [\mathbf{V}_i], [\mathbf{W}_i]$ can be thought as commitment keys, but in general they should be such that the $\mathbf{U}_i^\top$-MDDH and $\mathbf{V}_i^\top$-MDDH assumptions are hard.

SSB commitments $[\boldsymbol{c}]$,$[\boldsymbol{d}]$ and $[\boldsymbol{f}]$ are now computed as $[\boldsymbol{c}] = [\mathbf{GU}]\boldsymbol{a}$, $[\boldsymbol{d}] = [\mathbf{HV}]\boldsymbol{b}$ and $[\boldsymbol{f}] = [\mathbf{FW}]\boldsymbol{a}\circ\boldsymbol{b}$. To compute proofs we add to the crs $[\mathbf{Q}] = [(\mathbf{G}\otimes\mathbf{H})(\mathbf{U}\otimes\mathbf{V})]$ so that $[\boldsymbol{t}] = [\mathbf{Q}](\boldsymbol{a}\otimes\boldsymbol{b})$ satisfy $[\boldsymbol{c}]\otimes[\boldsymbol{d}] = e([\boldsymbol{t}], [1])$. Then we give a quasi-argument for linear knowledge transfer from $\boldsymbol{t}$ to $\boldsymbol{f}$.

2.6 From our Quasi-Arguments to Delegation.

In section 2.2 we saw that, in the delegation scheme from [GR19], the prover gives $3d$ commitments $[L_1], \ldots, [L_d], [R_1], \ldots, [R_d], [O_1], \ldots, [O_d]$ to, respectively, the left, right and output wires of each level of the circuit. Then, it gives a linear and quadratic knowledge transfer arguments to "transfer" knowledge of the opening from the input level, which is known to the verifier, to the next levels. Finally, the verifier checks that the commitment to the output opens to $\boldsymbol{y}$.

In section 6 we give a "compressed" version of [GR19] where the $3d$ commitments are shrunken into 3 no-signaling SSB commitments, and the $2d$ knowledge transfer arguments are shrunken into 2 quasi arguments. From the SSB commitments we can extract $[L_i], [R_i], [O_j]$ for $j = i - 1$ or $j = i$. Local knowledge soundness of the quasi arguments imply that knowledge is "transferred" from $[O_{i-1}]$ to $[L_i], [R_i]$ or from $[L_i], [R_i]$ to

$[O_i]$. One important technical problem with this approach is that the linear knowledge transfer argument is between the next level and all previous levels. That is, the knowledge is transferred from commitments to the output in all previous levels $[O_1], \ldots, [O_i]$, to commitments to the left and right wires in the next level $[L_{i+1}], [R_{i+1}]$. This means the quasi-argument must extract $O(d)$ values and hence is not succinct. We solve this issue by computing L_i, R_i, O_i as commitments also to the respective wires of all previous levels.

2.7 NIZK, SNARKs and Compact NIZK

We can use standard techniques to turn our delegation scheme into a NIZK argument. Essentially, the prover needs to prove knowledge of (additional) secret input wires w and proof that $C(x, w) = y$ for some secret input w. Given the "structure preserving" properties of our delegation scheme, we can directly apply the Groth Sahai proof system [GS08][9] on the set of verification equations. In general, to achieve knowledge soundness, all we need to prove soundness is an extractable (and hiding) commitment for extracting the witness w. Depending on the properties of the extractable commitment scheme we get different NIZK flavors.

If the commitments to the inputs are succinct, the construction yields a SNARK for NP. Such commitments are widely employed in SNARKs, but their security relies on non-standard assumptions: either knowledge type assumptions such as q-Knowledge of Exponents assumption [GGPR13] or the generic group model [Gro16]. If we take for example the zk-SNARK from [DFGK14], the size of q is the number of field elements extracted from a valid proof. Indeed, the proof of soundness requires the extraction of all the circuit wires, which are later used to break some falsifiable q-assumption. Consequently, the knowledge assumption is of size $q = O(|C|)$. By reducing the number of extracted values from $O(|C|)$ to $|w|$, we reduce the size of the underlying knowledge assumption to $q = |w| < |C|$.

If we use the "bit-string" argument of [GHR15a] to show knowledge of $\boldsymbol{b} \in \{0,1\}^n$, we get extractable commitments of size $n + O(1)$ group elements based on a constant-size falsifiable assumption. Combining this extractable commitment with our delegation scheme yields a NIZK argument for circuit satisfiability with proof size $n + O(1)$ groups elements, or equivalently of size $O(n\kappa)$.

Finally, we can then use the techniques of Katsumata et al. [KNYY19, KNYY20] to construct a compact NIZK. The construction of Katsumata et al. is based on a non-compact NIZK argument for NC^1 plus a symmetric key encryption scheme $(\mathsf{K}, \mathsf{E}, \mathsf{D})$ where the size of $\mathsf{E}(K, m)$ is $|m| + \mathsf{poly}(\kappa)$. Instead of committing to the input $\boldsymbol{x}$ of a circuit C, they computed $K \leftarrow \mathsf{K}(1^\kappa)$ to obtain $ct \leftarrow \mathsf{E}(K, \boldsymbol{x})$ and give a NIZK argument of knowledge of some $K \in \{0,1\}^{\mathsf{poly}(\kappa)}$ such that $C(\mathsf{D}(K, ct)) = 1$. We note that we can straightforward use this idea to construct compact NIZK for any circuit by simply plugging our NIZK argument based on the commitments of [GHR15a]. The final proof is of size $|ct| + |K|\mathsf{poly}(\kappa) + |\pi| = n + \mathsf{poly}(\kappa)$ and is sound for any polynomial size circuit.

3 Preliminaries

3.1 Notation

For $n \in \mathbb{N}$, let $[n]$ be the set $\{1, \ldots, n\}$. For vectors $\boldsymbol{a} = (a_i)_{i\in[n]}, \boldsymbol{b} = (b_i)_{i\in[n]} \in \mathbb{Z}_p^n$, we denote $\boldsymbol{a} \circ \boldsymbol{b} = (a_i b_i)_{i\in[n]}$ the Hadamard product of them, and for matrices $\mathbf{A} = (a_{i,j})_{i\in[n_1],j\in[m_1]} \in \mathbb{Z}_p^{n_1\times m_1}$, $\mathbf{B} \in \mathbb{Z}_p^{n_2\times m_2}$ we denote $\mathbf{A} \otimes \mathbf{B} = (a_{i,j}\mathbf{B})_{i\in[n_1],j\in[m_1]} \in \mathbb{Z}_p^{n_1 n_2\times m_1 m_2}$ their Kronecker product. We will be using the mixed-product property of kronecker products, which says that $(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD})$ whenever $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ have the appropriate dimensions. When $n_1 = n_2$ we denote by $\mathbf{A}|\mathbf{B} \in \mathbb{Z}_p^{n_1\times m_1+m_2}$ their vertical concatenation. For $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{Z}_p^n$ we write $\boldsymbol{x} \leq \boldsymbol{y}$ if and only if $x_i \leq y_i$ for all $i \in [n]$. We consider vectors of sets $\boldsymbol{S} = (S_1, \ldots, S_\ell)$, where $S_i \subseteq [n_i]$ for $i \in [\ell]$ and $n_i \in \mathbb{N}$, and extend set operations entry-wise. That is $\boldsymbol{S'} \subseteq \boldsymbol{S}$ if and only if $S'_i \subseteq S_i$ for all $i \in [\ell]$, and $|\boldsymbol{S}| = (|S_1|, \ldots, |S_\ell|)$. For $\boldsymbol{n} \in \mathbb{N}^\ell$, $[\boldsymbol{n}] = ([n_1], \ldots, [n_2])$.

[9] This can be also achieved in a more efficient way (concretely) by directly using hiding commitments for the delegation scheme.

We use implicit group notation. Let $gk = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \mathcal{P}_1, \mathcal{P}_2) \leftarrow \mathcal{G}(1^\kappa)$ be the description of an asymmetric bilinear group of size $p = O(2^\kappa)$ equipped with an efficient bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$, where $\mathcal{P}_\mu$ is a generator of $\mathbb{G}_\mu$, $\mu \in \{1, 2\}$. We assume all our algorithms receive as input gk sampled from $\mathcal{G}(1^\lambda)$, although in some abstract definitions is not necessarily the description of a bilinear group. For $r \in \mathbb{Z}_p$ we denote $[r]_\mu = r\mathcal{P}_\mu$ for $\mu \in \{1, 2, T\}$ and $\mathcal{P}_T = e(\mathcal{P}_1, \mathcal{P}_2)$. For a vector $a \in \mathbb{Z}_p^n$ and matrix $\mathbf{A} \in \mathbb{Z}_p^{n\times m}$ we denote with $[\boldsymbol{a}]_\mu$, $[\mathbf{A}]_\mu$ the natural embedding of $\boldsymbol{a}$, $\mathbf{A}$ in $\mathbb{G}_\mu$, respectively.

Sub-vectors and Sub-matrices. Let $S = \{s_1, \ldots, s_t\} \subseteq [n]$ and $\overline{S} = \{\overline{s}_1, \ldots, \overline{s}_{n-t}\}$ the set $[n] \setminus S$. We use an algebraic notation for the sub-vector $\boldsymbol{x}_S$ and sub-matrix $\mathbf{G}_S$ of some $\boldsymbol{x} \in \mathbb{Z}_p^n$ and $\mathbf{G} \in \mathbb{Z}_p^{m\times n}$ respectively. Let $\mathbf{P}_S \in \{0, 1\}^{n\times n}$ the permutation matrix defining the ordering $s_1, \ldots, s_t, \overline{s}_1, \ldots, \overline{s}_{n-t}$. That is, $\mathbf{P}_S \boldsymbol{e}_{s_i} = \boldsymbol{e}_i$ and $\mathbf{P}_{\overline{S}} \boldsymbol{e}_{\overline{s}_i} = \boldsymbol{e}_{i+t}$, where $\boldsymbol{e}_i$ is the i-th unitary vector of size n. We may simply write $\mathbf{P}$ when n, S are clear from the context. We also define the matrix $\mathbf{\Sigma}_S = (\mathbf{I}_t | \mathbf{0}_{t\times n-t})$. We may omit the subscript when the values are clear from the context.

We denote by $\boldsymbol{x}_S \in \mathbb{Z}_p^t, \mathbf{G}_S \in \mathbb{Z}_p^{k\times t}$ the sub-vector and sub-matrix containing the elements or columns with indices in $S \subseteq [n]$ of $\boldsymbol{x} \in \mathbb{Z}_p^n$ and $\mathbf{G} \in \mathbb{Z}_p^{k\times n}$, respectively.

Fact 1. *For any $\boldsymbol{x} \in \mathbb{Z}_p^n$ and any $S' \subseteq S \subseteq [n]$ it holds that:*

i. $\mathbf{P}_S \boldsymbol{x} = \left(\begin{smallmatrix} \boldsymbol{x}_S \\ \boldsymbol{x}_{\overline{S}} \end{smallmatrix}\right)$ *and* $\mathbf{G}\mathbf{P}_S^\top = (\mathbf{G}_S | \mathbf{G}_{\overline{S}})$.

ii. $\boldsymbol{x}_S = \mathbf{\Sigma}_S \mathbf{P}_S \boldsymbol{x}$ *and* $\mathbf{G}_S = \mathbf{G}\mathbf{P}_S^\top \mathbf{\Sigma}_S^\top$.

iii. $\mathbf{G}\boldsymbol{x} = \mathbf{G}_S \boldsymbol{x}_S + \mathbf{G}_{\overline{S}} \boldsymbol{x}_{\overline{S}}$.

iv. *Let $\boldsymbol{x}_{S'|S} = \mathbf{\Sigma}_{S|S'} \mathbf{P}_{S'|S} \boldsymbol{x}_S$, where $\mathbf{P}_{S'|S}$ is some permutation matrix such that $\mathbf{P}_{S'|S} \boldsymbol{x}_S = \left(\begin{smallmatrix} \boldsymbol{x}_{S'} \\ \boldsymbol{x}_{S\setminus S'} \end{smallmatrix}\right)$ and $\mathbf{\Sigma}_{S'|S} = (\mathbf{I}_{|S'|} | \mathbf{0}_{|S'|\times t-|S'|})$. $\boldsymbol{x}_{S'|S} = \boldsymbol{x}_{S'}$ and $\mathbf{G}_{S'|S} = \mathbf{G}_{S'}$.*

When $\boldsymbol{x} = \mathbf{U}\boldsymbol{w}$, for some matrix $\mathbf{U} \in \mathbb{Z}_p^{n\times m}$ and $\boldsymbol{w} \in m$, we abuse of notation and also write $\mathbf{U}_S$ for $\mathbf{\Sigma}_S \mathbf{P}_S \mathbf{U}$ so that $\boldsymbol{x}_S = \mathbf{U}_S \boldsymbol{w}$.

We extend this notation to two sets $S_1 \subseteq [n_1], S_2 \subseteq [n_2]$ and for $\boldsymbol{x} \in \mathbb{Z}_p^{n_1 n_2}$ define $\boldsymbol{x}_{S_1,S_2} \in \mathbb{Z}_p^{|S_1|\cdot|S_2|}$ as $\boldsymbol{x}_{S_1,S_2} = (\boldsymbol{x}_{(i-1)n_2+j} : i \in S_1 \text{ and } j \in S_2)$ in some fixed order. For matrices instead we define $\mathbf{G}_{S_1,S_2} = (q_{\ell,(i-1)n_2+j} : \ell \in [k], i \in S_1 \text{ and } j \in S_2) \in \mathbb{Z}_p^{k\times|S_1|\cdot|S_2|}$, where k is the number of columns of $\mathbf{G}$. Similarly as before, the following holds.

Fact 2. *For any $\boldsymbol{x} \in \mathbb{Z}_p^{n_1 n_2}$ and any $S_1' \subseteq S_1 \subseteq [n_1], S_2' \subseteq S_2 \subseteq [n_2]$ it holds that:*

i. *For some permutation matrix $\mathbf{\Pi} \in \mathbb{Z}_p^{n_1 n_2 \times n_1 n_2}$, $(\mathbf{P}_{S_1} \otimes \mathbf{P}_{S_2})\boldsymbol{x} = \mathbf{\Pi} \begin{pmatrix} \boldsymbol{x}_{S_1,S_2} \\ \boldsymbol{x}_{S_1,\overline{S}_2} \\ \boldsymbol{x}_{\overline{S}_1,S_2} \\ \boldsymbol{x}_{\overline{S}_1,\overline{S}_2} \end{pmatrix}$ and $\mathbf{G}(\mathbf{P}_{S_1}^\top \otimes \mathbf{P}_{S_2}^\top) = (\mathbf{G}_{S_1,S_2} | \mathbf{G}_{S_1,\overline{S}_2} | \mathbf{G}_{\overline{S}_1,S_2} | \mathbf{G}_{\overline{S}_1,\overline{S}_2}) \mathbf{\Pi}^\top$.*

ii. $\boldsymbol{x}_{S_1,S_2} = (\mathbf{\Sigma}_{S_1} \otimes \mathbf{\Sigma}_{S_2})(\mathbf{P}_{S_1} \otimes \mathbf{P}_{S_2})\boldsymbol{x}$ *and* $\mathbf{G}_{S_1,S_2} = \mathbf{G}(\mathbf{P}_{S_1}^\top \otimes \mathbf{P}_{S_2}^\top)(\mathbf{\Sigma}_{S_1}^\top \otimes \mathbf{\Sigma}_{S_2}^\top)$.

iii. $\mathbf{G}\boldsymbol{x} = \mathbf{G}_{S_1,S_2}\boldsymbol{x}_{S_1,S_2} + \mathbf{G}_{S_1,\overline{S}_2}\boldsymbol{x}_{S_1,\overline{S}_2} + \mathbf{G}_{\overline{S}_1,S_2}\boldsymbol{x}_{\overline{S}_1,S_2} + \mathbf{G}_{\overline{S}_1,\overline{S}_2}\boldsymbol{x}_{\overline{S}_1,\overline{S}_2}$.

iv. *Let $\boldsymbol{x}_{S_1',S_2'|S_1,S_2} = (\mathbf{\Sigma}_{S_1'|S_1}^\top \otimes \mathbf{\Sigma}_{S_2'|S_2}^\top)(\mathbf{P}_{S_1'|S_1} \otimes \mathbf{P}_{S_2'|S_2})\boldsymbol{x}_{S_1,S_2}$ and $\mathbf{G}_{S_1',S_2'|S_1,S_2} = \mathbf{G}(\mathbf{P}_{S_1'|S_1}^\top \otimes \mathbf{P}_{S_2'|S_2}^\top)(\mathbf{\Sigma}_{S_1'|S_1}^\top \otimes \mathbf{\Sigma}_{S_2'|S_2}^\top)$. Then $\boldsymbol{x}_{S_1',S_2'|S_1,S_2} = \boldsymbol{x}_{S_1',S_2'}$ and $\mathbf{G}_{S_1',S_2'|S_1,S_2} = \mathbf{G}_{S_1',S_2'}$*

3.2 Cryptographic Assumptions

Definition 1. *Let $k, \ell \in \mathbb{N}$. We call $\mathcal{D}_{\ell,k}$ (resp. $\mathcal{D}_k$) a matrix distribution if it outputs in PPT time, with overwhelming probability matrices in $\mathbb{Z}_p^{\ell\times k}$ (resp. in $\mathbb{Z}_p^{(k+1)\times k}$). For a matrix distribution $\mathcal{D}_k$, we denote as $\overline{\mathcal{D}}_k$ the distribution of the first k rows of the matrices sampled according to $\mathcal{D}_k$.*

Assumption 1. *Let $\mathcal{D}_{\ell,k}$ be a matrix distribution. For all non-uniform PPT adversaries $\mathcal{A}$ and relative to $gk \leftarrow \mathcal{G}(1^\kappa)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_p^k, [\boldsymbol{z}]_\gamma \leftarrow \mathbb{G}_\gamma^\ell$ and the coin tosses of adversary $\mathcal{A}$,*

1. *the Kernel Matrix Diffie-Hellman Assumption holds in $\mathbb{G}_\gamma$ [MRV16] if*

$$\Pr\left[[\boldsymbol{r}]_{3-\gamma} \leftarrow \mathcal{A}(gk, [\mathbf{A}]_\gamma) : \boldsymbol{r}^\top \mathbf{A} = 0\right] = \mathsf{negl}(\kappa),$$

2. *the Split Kernel Matrix Diffie-Hellman Assumption [GHR15a] holds if*

$$\Pr\left[[\boldsymbol{r}]_1, [\boldsymbol{s}]_2 \leftarrow \mathcal{A}(gk, [\mathbf{A}]_1, [\mathbf{A}]_2) : \boldsymbol{r} \neq \boldsymbol{s} \wedge \boldsymbol{r}^\top \mathbf{A} = \boldsymbol{s}^\top \mathbf{A}\right] = \mathsf{negl}(\kappa).$$

Assumption 2. *Let $\mathcal{D}_{\ell,k}$ be a matrix distribution and $gk \leftarrow \mathcal{G}(1^\kappa)$. For all non-uniform PPT adversaries $\mathcal{A}$ and relative to $gk \leftarrow \mathcal{G}(1^\kappa)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_p^k, [\boldsymbol{z}]_\gamma \leftarrow \mathbb{G}_\gamma^\ell$ and the coin tosses of adversary $\mathcal{A}$,*

1. *the Matrix Decisional Diffie-Hellman Assumption in $\mathbb{G}_\gamma$ ($\mathcal{D}_k$-MDDH_γ) holds if*

$$|\Pr[\mathcal{A}(gk, [\mathbf{A}]_\gamma, [\mathbf{A}\boldsymbol{w}]_\gamma) = 1] - \Pr[\mathcal{A}(gk, [\mathbf{A}]_\gamma, [\boldsymbol{z}]_\gamma) = 1]| \leq \mathsf{negl}(\kappa),$$

2. *the Split Matrix Decisional Diffie-Hellman Assumption in $\mathbb{G}_\gamma$ ($\mathcal{D}_k$-SMDDH_γ) holds if*

$$|\Pr[\mathcal{A}(gk, [\mathbf{A}]_{1,2}, [\mathbf{A}\boldsymbol{w}]_\gamma) = 1] - \Pr[\mathcal{A}(gk, [\mathbf{A}]_{1,2}, [\boldsymbol{z}]_\gamma) = 1]| \leq \mathsf{negl}(\kappa).$$

Assumption 3. *Let $(\mathcal{D}^1_{\ell,k}, \mathcal{D}^2_{\ell,k})$ be (possibly correlated) matrix distributions and $gk \leftarrow \mathcal{G}(1^\kappa)$. For all non-uniform PPT adversaries $\mathcal{A}$ and relative to $gk \leftarrow \mathcal{G}(1^\kappa)$, $(\mathbf{A}, \mathbf{B}) \leftarrow (\mathcal{D}^1_{\ell,k}, \mathcal{D}^2_{\ell,k}), \boldsymbol{w} \leftarrow \mathbb{Z}_p^k, [\boldsymbol{z}]_\gamma \leftarrow \mathbb{G}_\gamma^\ell$ and the coin tosses of adversary $\mathcal{A}$, the $(\mathcal{D}^1_{\ell,k}, \mathcal{D}^2_{\ell,k})$-Matrix Decisional Diffie-Hellman Assumption ($(\mathcal{D}^1_{\ell,k}, \mathcal{D}^2_{\ell,k})$-$\mathsf{MDDH}_\gamma$) holds if*

$$|\Pr[\mathcal{A}(gk, [\mathbf{A}]_1, [\mathbf{B}]_2, [\mathbf{A}\boldsymbol{w}]_1, [\mathbf{B}\boldsymbol{w}]_2) = 1] - \Pr[\mathcal{A}(gk, [\mathbf{A}]_1, [\mathbf{B}]_2, [\boldsymbol{s}]_1, [\boldsymbol{t}]_2) = 1]| \leq \mathsf{negl}(\kappa).$$

We also consider stronger versions of these definitions, denoted $(\mathcal{D}_{\ell,k}, h)$-MDDH, $(\mathcal{D}_{\ell,k}, h)$-SMDDH, $(\mathcal{D}^1_{\ell,k}, \mathcal{D}^2_{\ell,k}, h)$-MDDH, where the adversary is also given $h(\mathbf{A})$ ($h(\mathbf{A}, \mathbf{B})$ in the latter) for some (possibly probabilistic) function h.

3.3 Argument of Knowledge Transfer

In this section we recall arguments of knowledge transfer for membership in linear spaces as defined in [GR19] which in turn is just an instantiation of [KW15]. We also slightly modify the construction to turn it into an argument of knowledge transfer for the sum language, which we will use in later constructions.

Let gk be a bilinear group of order p and $\mathcal{M}, \mathcal{N}, \mathcal{P}, \mathcal{Q}$ be matrix distributions outputting matrices $[\mathbf{M}]_1 \in \mathbb{G}_1^{\ell_1 \times n}, [\mathbf{N}]_2 \in \mathbb{G}_2^{\ell_2 \times n} [\mathbf{P}]_1 \in \mathbb{G}_1^{\ell_3 \times n} [\mathbf{Q}]_2 \in \mathbb{G}_2^{\ell_4 \times n}$ respectively. In Fig. 1, we present two arguments of knowledge transfer for (1) the linear membership language

$$\mathcal{L}^{\mathsf{yes}}_{\mathsf{lin}} = \{([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2) \mid \exists \boldsymbol{w} \text{ s.t } \left(\begin{smallmatrix}\boldsymbol{c}_1\\ \boldsymbol{c}_2\end{smallmatrix}\right) = \left(\begin{smallmatrix}\mathbf{M}\\ \mathbf{N}\end{smallmatrix}\right)\boldsymbol{w} \text{ and } \left(\begin{smallmatrix}\boldsymbol{d}_1\\ \boldsymbol{d}_2\end{smallmatrix}\right) = \left(\begin{smallmatrix}\mathbf{P}\\ \mathbf{Q}\end{smallmatrix}\right)\boldsymbol{w}\}$$
$$\mathcal{L}^{\mathsf{no}}_{\mathsf{lin}} = \{([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, \boldsymbol{w}) \mid \left(\begin{smallmatrix}\boldsymbol{c}_1\\ \boldsymbol{c}_2\end{smallmatrix}\right) = \left(\begin{smallmatrix}\mathbf{M}\\ \mathbf{N}\end{smallmatrix}\right)\boldsymbol{w} \text{ and } \left(\begin{smallmatrix}\boldsymbol{d}_1\\ \boldsymbol{d}_2\end{smallmatrix}\right) \neq \left(\begin{smallmatrix}\mathbf{P}\\ \mathbf{Q}\end{smallmatrix}\right)\boldsymbol{w}\},$$

and (2) the sum knowledge transfer language

$$\mathcal{L}^{\mathsf{yes}}_{\mathsf{sum}} = \{([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2) \mid \exists \boldsymbol{w} \text{ s.t } \boldsymbol{c}_1 + \boldsymbol{c}_2 = (\mathbf{M} + \mathbf{N})\boldsymbol{w} \text{ and } \left(\begin{smallmatrix}\boldsymbol{d}_1\\ \boldsymbol{d}_2\end{smallmatrix}\right) = \left(\begin{smallmatrix}\mathbf{P}\\ \mathbf{Q}\end{smallmatrix}\right)\boldsymbol{w}\}$$
$$\mathcal{L}^{\mathsf{no}}_{\mathsf{sum}} = \{([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, \boldsymbol{w}) \mid \boldsymbol{c}_1 + \boldsymbol{c}_2 = (\mathbf{M} + \mathbf{N})\boldsymbol{w} \text{ and } \left(\begin{smallmatrix}\boldsymbol{d}_1\\ \boldsymbol{d}_2\end{smallmatrix}\right) \neq \left(\begin{smallmatrix}\mathbf{P}\\ \mathbf{Q}\end{smallmatrix}\right)\boldsymbol{w}\}.$$

A knowledge transfer argument is just an argument for the promise problem defined by $\mathcal{L}^{\mathsf{yes}}$ and $\mathcal{L}^{\mathsf{no}}$. Completeness means that an honest proof is accepting for any statement in $\mathcal{L}^{\mathsf{yes}}$. Soundness that any proof for a statement in $\mathcal{L}^{\mathsf{no}}$, which comes with an "advice" $\boldsymbol{w}$, is accepting only with negligible probability.

$\mathsf{K}(gk, [\mathbf{M}]_1, [\mathbf{N}]_2, [\mathbf{P}]_1, [\mathbf{Q}]_2)$:

- $\mathbf{K}_1 \leftarrow \mathbb{Z}_p^{\ell_1 \times \overline{k}}$; $\mathbf{K}_2 \leftarrow \mathbb{Z}_p^{\ell_2 \times \overline{k}}$; $\mathbf{K}_3 \leftarrow \mathbb{Z}_p^{\ell_3 \times \overline{k}}$; $\mathbf{K}_4 \leftarrow \mathbb{Z}_p^{\ell_4 \times \overline{k}}$.
- Sample $\mathbf{A} \leftarrow \mathcal{D}_k$; $\mathbf{\Gamma} \leftarrow \mathbb{Z}_p^{n \times \overline{k}}$.
- $[\mathbf{B}]_1 = [\mathbf{M}^\top \mathbf{K}_1 + \mathbf{N}^\top \mathbf{K}_2 + \mathbf{\Gamma}]_1$; $[\mathbf{D}]_2 = [\mathbf{P}^\top \mathbf{K}_3 + \mathbf{Q}^\top \mathbf{K}_4 - \mathbf{\Gamma}]_2$.
- $\mathbf{C}_1 = \mathbf{K}_1 \mathbf{A}$; $\mathbf{C}_2 = \mathbf{K}_2 \mathbf{A}$; $\mathbf{C}_3 = \mathbf{K}_3 \mathbf{A}$; $\mathbf{C}_4 = \mathbf{K}_4 \mathbf{A}$.
- Output $\mathsf{crs} = (gk, [\mathbf{A}]_{1,2}, [\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1, [\mathbf{C}_3]_2, [\mathbf{C}_4]_1)$.

$\mathsf{Prove}(\mathsf{crs}, ([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2), \boldsymbol{w})$:

- Sample $\boldsymbol{\rho} \leftarrow \mathbb{Z}_p^{\overline{k}}$; $[\boldsymbol{\pi}]_1 := \boldsymbol{w}^\top [\mathbf{B}]_1 + [\boldsymbol{\rho}]_1$; $[\boldsymbol{\theta}]_2 := \boldsymbol{w}^\top [\mathbf{D}]_2 - [\boldsymbol{\rho}]_2$.
- Output $([\boldsymbol{\pi}]_1, [\theta]_2)$

$\mathsf{Verify}(\mathsf{crs}, ([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2), ([\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2))$:

- Output 1 iff $e([\boldsymbol{\pi}]_1, [\mathbf{A}]_2) + e([\boldsymbol{\theta}]_2, [\mathbf{A}]_1) - e([\boldsymbol{c}_1^\top]_1, [\mathbf{C}_1]_2) - e([\boldsymbol{c}_2^\top]_2, [\mathbf{C}_2]_1) - e([\boldsymbol{d}_1^\top]_1, [\mathbf{C}_3]_2) - e([\boldsymbol{d}_2^\top]_2, [\mathbf{C}_4]_1)$;

Figure 1: Construction $\Pi_{\mathsf{kt\text{-}lin}}$ for $\mathcal{L}_{\mathsf{lin}}^{\mathsf{yes}}, \mathcal{L}_{\mathsf{lin}}^{\mathsf{no}}$. For $\ell_1 = \ell_2$, construction $\Pi_{\mathsf{kt\text{-}sum}}$ for $\mathcal{L}_{\mathsf{sum}}^{\mathsf{yes}}, \mathcal{L}_{\mathsf{sum}}^{\mathsf{no}}$ is identical with the only difference that $\mathbf{K}_2 = \mathbf{K}_1$.

We use this construction with (1) $\mathbf{Q} = \mathbf{0}$ for the case of linear knowledge transfer and (2) $\mathbf{N} = \mathbf{0}$ for the case of sum knowledge transfer so we prove only these two cases. We stress out that the proofs are easily extended to accommodate for the more general cases. We also strengthen the security requirements by allowing the adversary to get some extra information about the language parameters through some (possibly probabilistic) function h. We call this property h-strong soundness.

For the case of $\Pi_{\mathsf{kt\text{-}lin}}$, when setting $\mathbf{Q} = \mathbf{0}$, the security is shown in [GR19]. The only modification is that we allow the adversary $\mathcal{A}$ to get the discrete logarithms $\mathbf{N}, \mathbf{P}$ and the h information of the MDDH challenge, which does not affect the result of [GR19]. We extend the results of [GR19] to the sum argument. The security proof is essentially identical to the one for the bilateral case of [GR19]. For completeness we give the full proof in Appendix A.

4 No-Signaling Somewhere Statistically Binding Commitments

In this section we recall Somewhere Statistically Binding (SSB) commitments and then define two additional notions for SSB commitments: no-signaling extraction and oblivious key generation. The former is a natural adaptation of the definitions of no-signaling extractors from previous works [PR17, KPY19]. We show that the latter implies the former, and we give an efficient instantiation based on any $\mathcal{D}_k$-MDDH assumption. Finally, we consider the kronecker product of two of these commitments.

We now define somewhere Statistically Binding (SSB) commitment schemes [HW15, FLPS20]. An SSB commitment scheme, as the name suggests, is statistically binding only w.r.t. some variables which are determined during key generation. The commitment key computationally hides any information about this set, meaning that for all "modes" the commitment keys are computationally indistinguishable. Furthermore, the KeyGen outputs a trapdoor which allows to extract (a function of) the values in this set.

It will be useful to consider SSB commitments where committed vectors live in $\mathcal{M}^{n_1 n_2}$ and can be indexed by $i_1 \in [n_1], i_2 \in [n_2]$. We consider also 2 locality parameters $\boldsymbol{K} = (K_1, K_2)$ with $\boldsymbol{K}_i \leq n_i$, and extraction sets are of the form $\boldsymbol{S} = (S_1, S_2)$ where $S_i \subseteq [n_i]$ and $|S_i| \leq K_i$, for $i \in \{1, 2\}$. We put forward a stronger variant of the index set hiding property, where the distinguisher is also given $h(sk)$ for some function h. In this case we will say the SSB commitment is h-strong ISH.

Definition 2. *Let $[\cdot] : \mathcal{M} \to G$ be a function, where $\mathcal{M}$ is the message space and G some set. Syntactically,*

a Somewhere Statistically Binding Commitment Scheme CS *is a tuple of algorithms* $\mathsf{CS} = (\mathsf{KeyGen}, \mathsf{Com}, \mathsf{Extract})$

- $(ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S})$*:* KeyGen *takes as input the parameters* gk*,* $\boldsymbol{n} \in \mathbb{N}^\ell$*, locality parameters* $\boldsymbol{K} \in [\boldsymbol{n}]$ *and the sets* $\boldsymbol{S} \subseteq [\boldsymbol{n}], |\boldsymbol{S}| \leq \boldsymbol{K}$*. It outputs a commitment key* ck*, which may also contain some auxiliary information* aux*, a secret key* sk*, containing a trapdoor* τ *and possibly the random coins used by* KeyGen*.*
- $c \leftarrow \mathsf{Com}(ck, \boldsymbol{x})$*:* Com *takes as input the commitment key* ck *and a vector* $\boldsymbol{x} \in \mathcal{M}^{n_1 \cdot n_2}$ *and outputs a commitment* c*,*
- $\boldsymbol{y} \leftarrow \mathsf{Extract}(\tau, c)$*:* $\mathsf{Extract}$ *takes as input the trapdoor* τ *and a commitment* c*, and outputs the value* $\boldsymbol{y} \in G$ *allegedly equaling* $[\boldsymbol{x}_S]$*, where* $\boldsymbol{x}$ *is a valid opening for* c*.*

For all $\kappa \in \mathbb{N}, \boldsymbol{n} \in \mathbb{N}^2, \boldsymbol{K} \in [\boldsymbol{n}], \boldsymbol{S}_0, \boldsymbol{S}_1 \subseteq [\boldsymbol{n}]$ *with* $|\boldsymbol{S}_0|, |\boldsymbol{S}_1| \leq \boldsymbol{K}$*,* CS *must satisfy the following properties:*

- h-Strong Index Set Hiding*: for all PPT* $\mathcal{D}$

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(ck, h(sk)) = b \;\middle|\; \begin{array}{r} b \leftarrow \{0,1\} \\ (ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}_b) \end{array} \right] \leq \frac{1}{2} + \mathsf{negl}(\kappa).$$

- Somewhere Statistically Binding*: for all all, even unbounded* $\mathcal{A}$*,*

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\begin{array}{c} \mathsf{Com}(ck, \boldsymbol{x}) = \mathsf{Com}(ck, \boldsymbol{x}') \\ \text{and } \boldsymbol{x}_{\boldsymbol{S}} \neq \boldsymbol{x}'_{\boldsymbol{S}} \end{array} \;\middle|\; \begin{array}{r} (ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}); \\ (\boldsymbol{x}, \boldsymbol{x}') \leftarrow \mathcal{A}(ck); \end{array} \right] \leq \mathsf{negl}(\kappa).$$

- G-Extractability*: for all, even unbounded* $\mathcal{A}$

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\begin{array}{c} \exists \boldsymbol{x} \text{ s.t. } c = \mathsf{Com}(ck, \boldsymbol{x}) \\ \text{and } \boldsymbol{y} \neq [\boldsymbol{x}_{\boldsymbol{S}}] \end{array} \;\middle|\; \begin{array}{r} (ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}); c \leftarrow \mathcal{A}(ck); \\ \boldsymbol{y} \leftarrow \mathsf{Extract}(\tau, c), \text{ where } sk = (\tau, r); \end{array} \right] \leq \mathsf{negl}(\kappa)$$

Note that an SSB commitment is also "everywhere" computationally binding. This is the case since a breach in binding, namely the ability to produce c that opens to both $\boldsymbol{x} \neq \boldsymbol{x}'$, implies the ability to distinguish where the commitment is not statistically binding contradicting the index set hiding property.

We next present an extra property for an SSB commitment scheme which we call h-strong no-signaling extraction and is a natural adaptation of the definitions in [PR17, KPY19].

Definition 3. *We say the extractor of an SSB commitment scheme* $\mathsf{CS} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Com}, \mathsf{Extract})$ *is* h*-strong no-signaling if for any* $\boldsymbol{S}' \subseteq \boldsymbol{S} \subseteq [\boldsymbol{n}]$*, where* $|\boldsymbol{S}| \leq \boldsymbol{K}$*, and any PPT adversary* $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2)$*,*

$$\left| \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}_2(ck_{S'}, c, \boldsymbol{y}') = 1 \;\middle|\; \begin{array}{c} (ck_{S'}, sk_{S'}) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}') \\ c \leftarrow \mathcal{D}_1(ck, h(sk_{S'})); \text{if } c \notin \mathcal{C}\text{: } c \leftarrow \perp \\ \boldsymbol{y}' \leftarrow \mathsf{Extract}(\tau, c), \text{ where } sk = (\tau, r). \end{array} \right] - \right.$$
$$\left. \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}_2(ck_S, c, \boldsymbol{y}_{S'}) = 1 \;\middle|\; \begin{array}{c} (ck_S, sk_S) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}) \\ c \leftarrow \mathcal{D}_1(ck_S, h(sk_S)); \text{if } c \notin \mathcal{C}\text{: } c \leftarrow \perp \\ \boldsymbol{y} \leftarrow \mathsf{Extract}(\tau, c), \text{ where } sk = (\tau, r). \end{array} \right] \right| \leq \mathsf{negl}(\kappa).$$

We define also oblivious trapdoor generation. This property states that there exists an oblivious key generation algorithm, that takes a commitment key ck that allows extraction in $\boldsymbol{S}$ and a set $\boldsymbol{S}' \subseteq \boldsymbol{S}$, and can produce a fresh commitment key ck' and a trapdoor to extract $\boldsymbol{S}'$. The distribution of the new key ck' is statistically close to that of ck and – importantly – the oblivious key generation algorithm does not get as input the original extraction set $\boldsymbol{S}$. In other words, given a commitment key ck that we know allows extraction for some superset of $\boldsymbol{S}$, we can create a new key *with* a trapdoor for $\boldsymbol{S}'$ without skewing the distribution of ck.

Definition 4. *An SSB commitment scheme has oblivious trapdoor generation if there exists a PPT algorithm* $\mathsf{OblKeyGen}$ *such that for all* $\kappa \in \mathbb{N}, \boldsymbol{n} \in \mathbb{N}^2, \boldsymbol{K} \in [\boldsymbol{n}]$, $\boldsymbol{S} \subseteq [\boldsymbol{n}]$, *with* $|\boldsymbol{S}| \leq \boldsymbol{K}$, *and any* $\boldsymbol{S}'$ *such that* $\boldsymbol{S}' \subseteq \boldsymbol{S}$, *and for all, even unbounded* $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2)$,

$$\left| \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}_2(ck', c, \boldsymbol{y}') = 1 \;\middle|\; \begin{array}{r} (ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}); \\ (ck', \tau') \leftarrow \mathsf{OblKeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}', ck); \\ c \leftarrow \mathcal{D}_1(ck'); \boldsymbol{y}' \leftarrow \mathsf{Extract}(\tau', c), \text{ where } sk = (\tau, r) \end{array} \right] - \right.$$

$$\left. \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}_2(ck, c, \boldsymbol{y}_{\boldsymbol{S}'}) = 1 \;\middle|\; \begin{array}{r} (ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}); \\ c \leftarrow \mathcal{D}_1(ck); \boldsymbol{y} \leftarrow \mathsf{Extract}(\tau, c), \text{ where } sk = (\tau, r) \end{array} \right] \right| \leq \mathsf{negl}(\kappa)$$

Next, we show that an SSB commitment scheme with oblivious trapdoor generation is also no-signaling. We leave as an open problem to prove or disprove the opposite implication.

Theorem 1. *Let* $\mathsf{CS} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{OblKeyGen}, \mathsf{Com}, \mathsf{Extract})$ *be an SSB commitment scheme with oblivious trapdoor generation and* h*-strong ISH. Then,* CS *is also* h*-strong no-signaling.*

Proof. Fix any $\boldsymbol{S}' \subseteq \boldsymbol{S} \subseteq [\boldsymbol{n}]$ with $|\boldsymbol{S}| \leq \boldsymbol{K}$, and let $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2)$ be a distinguisher against no signaling extraction for these values. We show by a sequence of games that its success probability is negligible.

$\mathsf{Game}_0^{\mathcal{D}}(1^\kappa)$: In this game, we execute $(ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S})$. We then get $c \leftarrow \mathcal{D}_1(ck, h_{ns}(\mathsf{sk}))$, change it to $\perp$ if $c \notin \mathcal{C}$, and compute $\boldsymbol{y} \leftarrow \mathsf{Extract}(\tau, c)$ for $sk = (\tau, r)$. The output is $\mathcal{D}_2(ck, c, \boldsymbol{y}_{\boldsymbol{S}'})$.

$\mathsf{Game}_1^{\mathcal{D}}(1^\kappa)$: In this game, we execute $(ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S})$ and $(ck_{\mathsf{obl}}, \tau_{\mathsf{obl}}) \leftarrow \mathsf{OblKeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}', ck)$. We then compute $h(sk_{\mathsf{obl}})$ corresponding to ck_{obl} and get $c \leftarrow \mathcal{D}_1(ck_{\mathsf{obl}}, h(sk_{\mathsf{obl}}))$, change it to $\perp$ if $c \notin \mathcal{C}$, and compute $\boldsymbol{y}' \leftarrow \mathsf{Extract}(\tau_{\mathsf{obl}}, c)$. The output is $\mathcal{D}_2(ck_{\mathsf{obl}}, c, \boldsymbol{y}')$.

$\mathsf{Game}_1^{\mathcal{D}}(1^\kappa)$: In this game, we execute $(ck, sk) \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}')$ and $(ck_{\mathsf{obl}}, \tau_{\mathsf{obl}}) \leftarrow \mathsf{OblKeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}', ck)$. We then compute $h(sk_{\mathsf{obl}})$ corresponding to ck_{obl} and get $c \leftarrow \mathcal{D}_1(ck_{\mathsf{obl}}, h(sk_{\mathsf{obl}}))$, change it to $\perp$ if $c \notin \mathcal{C}$, and compute $\boldsymbol{y}' \leftarrow \mathsf{Extract}(\tau_{\mathsf{obl}}, c)$. The output is $\mathcal{D}_2(ck_{\mathsf{obl}}, c, \boldsymbol{y}')$.

$\mathsf{Game}_3^{\mathcal{D}}(1^\kappa)$: In this game, we execute $(ck', sk') \leftarrow \mathsf{KeyGen}(gk, \boldsymbol{n}, \boldsymbol{K}, \boldsymbol{S}')$. We then get $c \leftarrow \mathcal{D}_1(ck', h_{ns}(sk'))$, change it to $\perp$ if $c \notin \mathcal{C}$, and compute $\boldsymbol{y} \leftarrow \mathsf{Extract}(\tau', c)$ for $sk = (\tau, r)$. The output is $\mathcal{D}_2(ck, c, \boldsymbol{y}')$.

Now we show the output of games i and $i+1$ is indistinguishable for $i = 0$ to 2.

- *Cases* $i = 0$, $i = 2$. For $i = 0$, the two games are distributed identically to the two cases of the oblivious trapdoor generation definition for $\boldsymbol{S}' \subseteq \boldsymbol{S}$. Thus, the outputs of the games are statistically close. For $i = 2$, the same argument holds for $\boldsymbol{S} = \boldsymbol{S}'$. Note that in both cases, the oblivious trapdoor generation distinguisher is unbounded so it can compute sk_{obl}.
- *Case* $i = 1$. The difference in the two games is how we sample the (ck, sk) pair, either programmed to extract $\boldsymbol{S}$ or $\boldsymbol{S}'$. By the h-index set hiding property the outputs of the two games are computationally indistinguishable.

Finally, noting that $\mathsf{Game}_0^{\mathcal{D}}$, $\mathsf{Game}_3^{\mathcal{D}}$ correspond to the two cases of no signaling extraction, the result follows. □

4.1 Algebraic SSB Commitments.

In this section, we define algebraic SSB commitments following the definition of algebraic commitment schemes of [RS20] and extend them to what we call *split* algebraic SSB commitments.

Informally, an algebraic SSB commitment scheme is a commitment scheme where the commitment key is a matrix $[\mathbf{G}]$ of group elements such that (1) committing to a vector $\boldsymbol{x}$ is done by multiplying on the left with $[\mathbf{G}]$, that is $[\boldsymbol{c}] = [\mathbf{G}]\boldsymbol{x}$ and (2) the trapdoor is a matrix of field elements $\mathbf{T}$ and local extraction is done by multiplying the commitment on the left with $\mathbf{T}^\top$, that is $[\boldsymbol{x}_S] = \mathbf{T}^\top[\boldsymbol{c}]$. We also allow the commitment key to output some public auxiliary information which is not used in committing nor extraction.

Definition 5. *An SSB commitment scheme* $\mathsf{CS} = (\mathsf{KeyGen}, \mathsf{Com}, \mathsf{Extract})$ *is* algebraic *if, given* $gk \leftarrow \mathcal{G}(1^\kappa)$, $\mathsf{KeyGen}(gk, n, \boldsymbol{K}, \boldsymbol{S})$ *outputs* $ck = [\mathbf{G}] \in \mathbb{G}^{\overline{\boldsymbol{K}} \times n}$ *and* $sk = (\mathbf{T} \in \mathbb{Z}_p^{\overline{\boldsymbol{K}} \times |S|}, \mathbf{G})$ *where* $\overline{\boldsymbol{K}} \geq \boldsymbol{K}$, $\mathsf{Com}([\mathbf{G}], \boldsymbol{x}) = [\mathbf{G}]\boldsymbol{x}$ *and* $\mathbf{T}^\top \mathbf{G} = \boldsymbol{\Sigma}_{\boldsymbol{S}} \mathbf{P}_{\boldsymbol{S}}$.

We also define a subtype of algebraic commitments which are specific to asymmetric groups, where the commitment key is "split" between the two groups.

Definition 6. *An SSB commitment scheme* $\mathsf{CS} = (\mathsf{KeyGen}, \mathsf{Com}, \mathsf{Extract})$ *is* split algebraic *if* $\mathsf{KeyGen}(gk, n, \boldsymbol{K}, \boldsymbol{S})$ *outputs* $ck = ([\mathbf{G}]_1 \in \mathbb{G}_1^{\overline{\boldsymbol{K}} \times n}, [\mathbf{H}]_2 \in \mathbb{G}_2^{\overline{\boldsymbol{K}} \times n})$ *and* $sk = (\mathbf{T} \in \mathbb{Z}_p^{\overline{\boldsymbol{K}} \times |\boldsymbol{S}|}, (\mathbf{G}, \mathbf{H}))$, *for* $\overline{\boldsymbol{K}} \geq \boldsymbol{K}$, $\mathsf{Com}([\mathbf{G}]_1, [\mathbf{H}]_2, \boldsymbol{x}) = ([\mathbf{G}]_1\boldsymbol{x}, [\mathbf{H}]_2\boldsymbol{x})$ *and* $\mathbf{T}^\top \mathbf{G} + \mathbf{T}^\top \mathbf{H} = \boldsymbol{\Sigma}_{\boldsymbol{S}} \mathbf{P}_{\boldsymbol{S}}$.

All SSB commitment schemes in this work are algebraic or split-algebraic. Note that all (split-)SSB commitments only differ on the key generation algorithm. For that reason we sometimes refer to a commitment key distribution as the commitment scheme itself.

In the case of non-split algebraic SSB commitments, we can $\mathbb{G}$-extract by computing

$$\mathbf{T}^\top [\boldsymbol{c}] = \mathbf{T}^\top [\mathbf{G}\boldsymbol{x}] = [\boldsymbol{\Sigma}_S \mathbf{P}_S \boldsymbol{x}] = [\boldsymbol{x}_S],$$

while in the case of split-algebraic commitments, we can only $\mathbb{G}_T$ extract. That is, we can compute values $[\boldsymbol{u}_S]_1, [\boldsymbol{v}_S]_2$ such that $e([\boldsymbol{u}_S]_1, [1]_2) + e([1]_1, [\boldsymbol{v}_S]_2) = [\boldsymbol{x}_S]_T$. Indeed, if $[\boldsymbol{c}]_1 = [\mathbf{G}]_1\boldsymbol{x}$ and $[\boldsymbol{d}]_2 = [\mathbf{H}]_2\boldsymbol{x}$ then we can compute $[\boldsymbol{u}_S]_1 = \mathbf{T}[\boldsymbol{c}]_1$ and $[\boldsymbol{v}_S]_2 = \mathbf{T}[\boldsymbol{d}]_2$ and it holds that

$$\boldsymbol{u}_S + \boldsymbol{v}_S = \mathbf{T}^\top \boldsymbol{c} + \mathbf{T}^\top \boldsymbol{d} = \mathbf{T}^\top \mathbf{G}\boldsymbol{x} + \mathbf{T}^\top \mathbf{H}\boldsymbol{x} = (\mathbf{T}^\top \mathbf{G} + \mathbf{T}^\top \mathbf{H})\boldsymbol{x} = \boldsymbol{\Sigma}_S \mathbf{P}_S \boldsymbol{x} = \boldsymbol{x}_S.$$

Note that by definition, if the commitment key generation does not fail, the commitments are perfectly binding/extractable at S. This will be the case for commitment schemes with perfect completeness. We will utilize this fact in our constructions to simplify some of the arguments.

4.2 Somewhere Statistically Binding Commitments with Oblivious Trapdoor Generation

We present in Fig. 2 a simple construction of an SSB with Oblivious Key Generation from plain SSB commitments with locality parameter 1. The setup algorithm instantiates K different commitment keys and, given a set S, each of the first $|S|$ commitment keys is extractable in a different position $s \in S$. The last $K - |S|$ are binding for the empty set. To commit to a value $\boldsymbol{x}$, one gives K commitments to this value with each of the commitment keys. To verify an opening, one verifies each individual opening and that all the openings are the same.

Note that the ordering of the elements in S is arbitrary and, in some sense, there's no unique key generation algorithm for a set S. Indeed, it is only necessary that the commitment key contains K commitment keys for locality 1 such that $ck_{i_1}, \ldots, ck_{i_{|S|}}$ are binding at $s_1, \ldots, s_{|S|}$ respectively. Note that there are $\binom{K}{|S|}$ different choices of $i_1, \ldots, i_n$. For this reason, if the input of the oblivious generator is just S', it is impossible to know which commitment keys are the ones corresponding to S'. To alleviate this, the oblivious key generator receives as advice the indices where S' "appears" in S that is, $i_1, \ldots, i_{|S'|}$ such that $s_{i_1} = s'_1$.

In this case we need to change a little the proof that oblivious trapdoor generation implies no-signaling. We add a game $\mathsf{Game}_{1/2}^{\mathcal{D}}(1^\kappa)$, between games 0 and 1, which is identical to $\mathsf{Game}_0^{\mathcal{D}}(1^\kappa)$ but $\mathcal{E}_1$ samples ck_i binding at $\{s_i\}$ if $s_i \in S'$ and at $\emptyset$ if not. By the index-set hiding property of $ck_1, \ldots, ck_K$ the output of both games is indistinguishable. $\mathsf{Game}_1^{\mathcal{D}}(1^\kappa)$ is as before but the oblivious key generator receives also the advise. The rest of the proof is exactly as before

Theorem 2. *Let* CS *be an SSB commitment with locality parameter* $K = 1$. *Then construction* CS' *of Fig. 2 is an SSB commitment with Oblivious Trapdoor Generation.*

Proof. First, we show that CS' is an SSB commitment. For index-hiding we can use a standard hybrid argument to show that the concatenation of K commitment keys are indeed indistinguishable. Somewhere

$\mathsf{CS}'.\mathsf{KeyGen}(gk, n, K, S)$**:**

- For $s_i \in S$ set $(ck_i, \tau_i) \leftarrow \mathsf{CS.KeyGen}(gk, n, 1, \{s_i\})$.
- For $|S| + 1 \leq i \leq K$ set $(ck_i, \tau_i) \leftarrow \mathsf{CS.KeyGen}(gk, n, 1, \emptyset)$.
- Set $ck = (ck_1, \ldots, ck_K), \tau = ((\tau_1, s_1), \ldots, (\tau_{|S|}, s_{|S|}))$ and output (ck, τ)

$\mathsf{CS}'.\mathsf{Com}(ck = (ck_1, \ldots, ck_K), \boldsymbol{x})$**:**

- For $1 \leq i \leq K$ compute $c_i \leftarrow \mathsf{CS.Com}(ck_i, \boldsymbol{x})$.
- Set $c = (c_1, \ldots, c_K)$ and output c.

$\mathsf{CS}'.\mathsf{Extract}(\tau = ((\tau_1, s_1), \ldots, (\tau_{|S|}, s_{|S|})), c = (c_1, \ldots, c_K))$**:**

- For all $s_i \in S$ compute $y_{s_i} \leftarrow \mathsf{CS.Extract}(\tau_i, c_i)$.
- Set $\boldsymbol{y} = (y_1, \ldots, y_{|S|})$ and output $\boldsymbol{y}$.

$\mathsf{CS}'.\mathsf{OblKeyGen}(gk, n, K, S', a, ck = (ck_1, \ldots, ck_K))$**:**

- Parse a as $i_1, \ldots, i_{|S'|}$, the indices of the commitment keys binding at $s_{i_j} \in S'$.
- For $1 \leq j \leq |S'|$ set $(ck_{i_j}, \tau_{i_j}) \leftarrow \mathsf{CS.KeyGen}(gk, n, 1, \{s'_j\})$.
- Set $ck = (ck_1, \ldots, ck_K), \tau = ((\tau_{i_1}, s_{i_1}), \ldots, (\tau_{i_{|S|'}}, (s_{i_{|S|'}}))$ and output (ck, τ)

Figure 2: Oblivious SSB commitment scheme from K SSB commitments with locality parameter 1.

Statistical Binding and G-extractability of CS' follow from the respective properties of CS. Indeed, for the former, note that each individual commitment is statistically binding in one coordinate, and for a commitment-opening to verify, all commitments are checked w.r.t. to the same opening; thus, effectively the commitment is statistically binding in the set S. For the latter, we use the same argument and the fact that the extractor of CS can G-extract each value independently.

For oblivious trapdoor generation, note that the crs output by $\mathsf{OblKeyGen}$ follow exactly the same distribution as the one output by KeyGen as well as a valid trapdoor for S'. □

Next, we present a more efficient SSB commitment scheme with oblivious trapdoor generation. The scheme is parameterized by a group $\mathbb{G}_\mu$, the message space is $\mathbb{Z}_p^n$ and we extract $[\boldsymbol{x}_S]_\mu$. The construction is essentially the one given in [FLPS20], which in turn is a generalization of the so called *Multi-Pedersen commitments* from [GHR15a], with a minor change in the key generation algorithm.

$\mathsf{KeyGen}(gk, n, K, S)$**:**

- Let $\mathbf{A} \leftarrow \mathcal{D}_k$, $\mathbf{B} \leftarrow \mathbb{Z}_p^{K+k\times K-|S|}$, $\mathbf{W} \leftarrow \mathbb{Z}_p^{K-1\times k+1}$ and define $\mathbf{G}_0 = \begin{pmatrix} \mathbf{B} & \mathbf{A} \\ & \mathbf{WA} \end{pmatrix}$.
- Let $\mathbf{G}_S \leftarrow \mathbb{Z}_p^{K+k\times|S|}$ and $\mathbf{\Gamma} \leftarrow \mathbb{Z}_p^{K+k-|S|\times n-|S|}$.
- Let $\mathbf{T}_S \in \mathbb{Z}_p^{K+k\times|S|}$ s.t. $\mathbf{T}_S^\top \mathbf{G}_S = \mathbf{I}_{|S|}$ and $\mathbf{T}_S^\top \mathbf{G}_0 = \mathbf{0}_{|S|\times K+k-|S|}$. Abort if such a matrix does not exist.
- Let $\mathbf{G} = (\mathbf{G}_S | \mathbf{G}_0\mathbf{\Gamma})\,\mathbf{P}_S$. Output $(ck, sk) = ([\mathbf{G}]_\mu, (\mathbf{T}_S, \mathbf{G}))$.

$\mathsf{OblKeyGen}(gk, n, K, S', ck = [\mathbf{G}]_\mu)$**:** $\quad //S' \subseteq S$

- Sample $\mathbf{G}_1 \leftarrow \mathbb{Z}_p^{K+k-|S'|\times|S'|}$, $\mathbf{G}_2 \leftarrow \mathbb{Z}_p^{|S'|\times|S'|}$, $\mathbf{R} \leftarrow \mathbb{Z}_p^{|S'|\times K+k-|S'|}$.
- Compute a matrix $\mathbf{T} \in \mathbb{Z}_p^{|S'|\times|S'|}$ such that $(\mathbf{G}_1^\top \mathbf{R}^\top - \mathbf{G}_2^\top)\mathbf{T} = \mathbf{I}_{|S'|}$. Abort if such a matrix does not exist.
- Denote by $[\overline{\mathbf{G}}_{\overline{S}'}]_\mu$ the matrix containing the first $K + k - |S'|$ rows of $[\mathbf{G}_{\overline{S}'}]_\mu$.
- Output $ck_{\mathsf{ob}} = [\mathbf{G}^*]_\mu = \begin{pmatrix} [\mathbf{G}_1]_\mu & [\overline{\mathbf{G}}_{\overline{S}'}]_\mu \\ [\mathbf{G}_2]_\mu & \mathbf{R}[\overline{\mathbf{G}}_{\overline{S}'}]_\mu \end{pmatrix} \mathbf{P}_{S'}$ and $\tau_{\mathsf{ob}} = \mathbf{T}^* = \begin{pmatrix} \mathbf{R}^\top\mathbf{T} \\ -\mathbf{T} \end{pmatrix}$

$\mathsf{Com}(ck, \boldsymbol{x})$**:** Parse $ck = [\mathbf{G}]_\mu$ and output $[\boldsymbol{c}]_\mu = [\mathbf{G}]_\mu \boldsymbol{x}$.

$\mathsf{Extract}(\tau, [\hat{\boldsymbol{x}}]_\mu)$**:** Output $[\boldsymbol{x}_S]_\mu = \mathbf{T}_S^\top [\hat{\boldsymbol{x}}]_\mu$.

Figure 3: SSB commitment scheme with oblivious trapdoor generation parametrized by the matrix distribution $\mathcal{D}_k$.

For simplicity, we describe the oblivious key generation algorithm in terms of the permutation $\mathbf{P}_S$ while it is not really needed. Indeed, it only needs to randomly sample itself the columns corresponding to S' and sample the lower rows as a random combination of upper rows or columns in $\overline{S}'$.

In [FLPS20] it is shown that the Index Set Hiding property can be reduced to DDH with a security lost of $2\log K$ when $\mathbf{G}_0$ is uniform using the results of [Vil12]. In our case $\mathbf{G}_0$ it is not completely uniform as some part depends on $\mathcal{D}_k$. Although it seems still possible to use [Vil12], for simplicity we use a naive hybrid argument at the cost of a less tight reduction. Although the security lost is $2K$ instead of $2\log K$, in general K is small (constant in our instantiations) and hence it doesn't make much difference. We give a proof of the following theorem.

Theorem 3. *Construction* CS *of Fig. 3 is an SSB commitment scheme. It is somewhere statistically binding and* $\mathbb{G}$*-Extractable with probability at least* $1 - \frac{K}{p}$ *and Index Set Hiding with probability at least* $1 - 2K \cdot \mathsf{Adv}_{\mathsf{MDDH}\text{-}\mathcal{D}_k}(\mathcal{D})$*, where* $\mathcal{D}$ *is a PPT adversary against the* $\mathsf{MDDH}\text{-}\mathcal{D}_k$ *assumption.*

Proof. We first show that $\mathsf{CS.KeyGen}$ aborts only with probability $\frac{K}{p}$. Let $\mathbf{G}_0^\perp$ be a matrix whose columns are a basis of the kernel of $\mathbf{G}_0^\top$. Since $\mathbf{G}_0$ is uniformly distributed, by the Schwartz-Zippel lemma, $\mathbf{G}_0$ has rank $K + k - |S|$ with probability at least $1 - \frac{K+k-|S|}{p}$. Now, consider the matrix $\mathbf{G}_S^\top \mathbf{G}_0^\perp$. Again, by the Schwartz-Zippel lemma and the fact that $\mathbf{G}_S$ is uniformly distributed, this matrix has rank $|S|$ with probability at least $1 - \frac{|S|}{p}$, and thus, it is invertible. Let $\mathbf{T}$ be its inverse. This matrix exists except with probability $\frac{K+k-|S|+|S|}{p} = \frac{K+k}{p}$. Now, set $\mathbf{T}_S = \mathbf{G}_0^\perp \mathbf{T}$. We have that $\mathbf{G}_S^\top \mathbf{T}_S = \mathbf{G}_S^\top \mathbf{G}_0^\perp \mathbf{T}' = \mathbf{I}_{|S|}$ and $\mathbf{G}_0^\top \mathbf{T}_S = \mathbf{G}_0^\top \mathbf{G}_0^\perp \mathbf{T}' = \mathbf{0}_{K+k-|S|\times|S|}$, which concludes the proof.

Index Set Hiding. Consider the following sequence of hybrid games.

- $\mathsf{Game}_0^{\mathcal{D}}$: In this game we sample $(ck, sk) \leftarrow \mathsf{KeyGen}(1^\lambda, gk, n, K, S_0)$ and output $\mathcal{D}(ck)$.

- $\mathsf{Game}_1^{\mathcal{D}}$: In this game we sample $(ck, sk) \leftarrow \mathsf{KeyGen}(1^\lambda, gk, n, K, \emptyset)$ and output $\mathcal{D}(ck)$.
- $\mathsf{Game}_2^{\mathcal{D}}$: In this game we sample $(ck, sk) \leftarrow \mathsf{KeyGen}(1^\lambda, gk, n, K, S_1)$ and output $\mathcal{D}(ck)$.

Noting that in Game_0 and in Game_1 the difference in the distributions of ck is that in the former $\mathbf{G}_{S_0}$ is uniform, while in the later $\mathbf{G}_{S_0} = \mathbf{G}_0\mathbf{\Gamma}_{S_0}$, where $\mathbf{\Gamma}_{S_0} \in \mathbb{Z}_p^{K+k-|S|\times|S_0|}$. Using a standard hybrid argument, we can bound the advantage of distinguishing these games by $|S_0| \leq K$ times the advantage of breaking the $\mathbf{G}_0$-MDDH assumption. It is not hard to see that the $\mathbf{G}_0$-MDDH can be reduced (without security lost) to the $\mathcal{D}_k$-MDDH assumption. We conclude that the advantage of distinguishing Game_0 and Game_1 can be bounded by $K \cdot \mathsf{Adv}_{\mathcal{D}_k\text{-MDDH}}$. The same argument applies to Game_1 and in Game_2.

Somewhere Statistically Binding. Finally we show the somewhere statistically binding and extractability property. Let $\mathbf{G}_S, \mathbf{G}_0, \mathbf{\Gamma}$, implicitly defined by $(ck, sk) \leftarrow \mathsf{CS.KeyGen}(gk, n, K, S)$. Conditioned on $\mathsf{CS.KeyGen}$ not failing, which only happens with probability at most $1 - \frac{K}{p}$, the matrix $\mathbf{T}_S \in \mathbb{Z}_p^{K+k\times|S|}$ satisfies $\mathbf{T}_S^\top \mathbf{G} = \mathbf{\Sigma}_S \mathbf{P}_S$.

Now let $\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{Z}^n$. For extractability, note that $\mathbf{T}^\top \mathsf{CS.Com}([\mathbf{G}]_\mu, \boldsymbol{x}) = \mathbf{T}^\top[\mathbf{G}]_\mu \boldsymbol{x} = [\mathbf{\Sigma}_S \mathbf{P}_S]_\mu \boldsymbol{x} = [\boldsymbol{x}_S]_\mu$. Additionally, if $\mathsf{CS.com}([\mathbf{G}]_\mu, \boldsymbol{x}) = \mathsf{CS.com}([\mathbf{G}]_\mu, \boldsymbol{x}')$ and we multiply by $\mathbf{T}^\top$ on both sides, we get that $\boldsymbol{x}_S = \boldsymbol{x}'_S$ □

In the next Theorem we assume $\mathcal{D}_k$ outputs full rank matrices with overwhelming probability. Note that this is true for most matrix distributions such as the uniform and the linear family.

Theorem 4. *Construction* CS *of Fig. 3 satisfies Oblivious Trapdoor Generation. Furthermore, for all even unbounded* $\mathcal{D} = (\mathcal{D}_1, \mathcal{D}_2)$*, against oblivious trapdoor generation,* $\mathsf{Adv}^{\mathsf{CS}}_{\mathsf{Oblv}}(\mathcal{D}) \leq \frac{K}{p}$.

Proof. Let $K \leq n$ and $S' \subseteq S \subseteq [n]$. We first show that the oblivious key follows exactly the same distribution as the original key. Let $ck := [\mathbf{G}]_\mu$ be the output of $\mathsf{KeyGen}(gk, n, K, S)$ and $ck_{\mathsf{ob}} = [\mathbf{G}^*]_\mu$ be the output of $\mathsf{OblKeyGen}(gk, n, K, S', [\mathbf{G}])$. We can write ck as $\mathbf{G} = ((\mathbf{G}_{S'}|\mathbf{G}_{S'|S})\mathbf{P}_{S'|S}|\mathbf{G}_0\mathbf{\Gamma})\mathbf{P}_S$.

Let $\overline{\mathbf{G}}_{S'|S} \in \mathbb{Z}_p^{K+k-|S'|\times K-|S'|}$, $\underline{\mathbf{G}}_{S'|S} \in \mathbb{Z}_p^{|S'|\times K-|S'|}$, $\overline{\mathbf{G}}_0 \in \mathbb{Z}_p^{K+k-|S'|\times k}$, $\underline{\mathbf{G}}_0 \in \mathbb{Z}_p^{|S'|\times k}$, such that $\mathbf{G}_{S'|S} = \begin{pmatrix}\overline{\mathbf{G}}_{S'|S}\\ \underline{\mathbf{G}}_{S'|S}\end{pmatrix}, \mathbf{G}_0 = \begin{pmatrix}\overline{\mathbf{G}}_0\\ \underline{\mathbf{G}}_0\end{pmatrix}$. We claim that there exists a matrix $\mathbf{R} \in \mathbb{Z}_p^{|S'|\times K+k-|S'|}$, uniformly distributed, such that $\left(\underline{\mathbf{G}}_{S'|S}|\underline{\mathbf{G}}_0\right) = \mathbf{R}\left(\overline{\mathbf{G}}_{S'|S}|\overline{\mathbf{G}}_0\right)$ as in the output of $\mathsf{OblKeyGen}$. If this is the case, the distributions of ck output by KeyGen and ck_{ob} output by $\mathsf{OblKeyGen}$ are identical, since we can write

$$\begin{aligned}
\mathbf{G} &= \left(\mathbf{G}_{S'} \quad \left(\begin{pmatrix}\overline{\mathbf{G}}_{S'|S} & \overline{\mathbf{G}}_0\\ \underline{\mathbf{G}}_{S'|S} & \underline{\mathbf{G}}_0\end{pmatrix} \quad \begin{pmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{\Gamma}\end{pmatrix}\right)\mathbf{P}_{S'|S}\right)\mathbf{P}_S\\
&= \left(\mathbf{G}_{S'} \quad \left(\begin{pmatrix}\overline{\mathbf{G}}_{S'|S} & \overline{\mathbf{G}}_0\\ \mathbf{R}\left(\overline{\mathbf{G}}_{S'|S} & \overline{\mathbf{G}}_0\right)\end{pmatrix} \quad \begin{pmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{\Gamma}\end{pmatrix}\right)\mathbf{P}_{S'|S}\right)\mathbf{P}_S\\
&= \begin{pmatrix}\mathbf{G}_{S'} & \overline{\mathbf{G}}_{\overline{S}}\\ & \mathbf{R}\overline{\mathbf{G}}_{\overline{S}}\end{pmatrix}\mathbf{P}_S.
\end{aligned}$$

First we show that the matrix $(\overline{\mathbf{G}}_{S'|S}|\overline{\mathbf{G}}_0)$ is full rank with overwhelming probability. Indeed, $\overline{\mathbf{G}}_0 = \left(\frac{\mathbf{A}}{\overline{\mathbf{W}}\mathbf{A}}\right)$, where $\mathbf{A} \leftarrow \mathcal{D}_k, \overline{\mathbf{W}} \leftarrow \mathbb{Z}_p^{K-1-|S'|\times k+1}$, and it has rank k. By the fact that $\overline{\mathbf{G}}_{S'|S}$ is uniform, using the Schwartz-Zippel lemma we get that $(\overline{\mathbf{G}}_{S'|S}|\overline{\mathbf{G}}_0)$ has rank $K + k - |S'|$ except with probability $\frac{K-|S'|}{p}$. This means that the matrix is invertible and we can set $\mathbf{R} = (\underline{\mathbf{G}}_{S'|S}|\underline{\mathbf{G}}_0)(\overline{\mathbf{G}}_{S'|S}|\overline{\mathbf{G}}_0)^{-1}$. Furthermore, both $\underline{\mathbf{G}}_{S'|S}$ and $\underline{\mathbf{G}}_0 = \underline{\mathbf{W}}\mathbf{A}$ are uniform, the latter since $\underline{\mathbf{W}} \in \mathbb{Z}_p^{|S'|\times k+1}$ is uniformly distributed and $\mathbf{A}$ is full rank, and the former by construction.

To conclude the proof, it remains to show that the trapdoor output by $\mathsf{OblKeyGen}(gk, n, K, S', [\mathbf{G}])$ is correct w.r.t ck_{ob}, that is $\mathbf{T}^{*\top}\mathbf{G}^* = \mathbf{\Sigma}_{S'}$. By a simple calculation,

$$\mathbf{T}^{*\top}\mathbf{G}^* = \begin{pmatrix}\mathbf{T}^\top\mathbf{R} & -\mathbf{T}\end{pmatrix}\begin{pmatrix}\mathbf{G}_1 & \overline{\mathbf{G}}_{\overline{S}'}\\ \mathbf{G}_2 & \mathbf{R}\overline{\mathbf{G}}_{\overline{S}'}\end{pmatrix} = \begin{pmatrix}\mathbf{T}^\top(\mathbf{R}\mathbf{G}_1 - \mathbf{G}_2) & \mathbf{T}^\top\mathbf{R}\overline{\mathbf{G}}_{\overline{S}'} - \mathbf{T}^\top\mathbf{R}\overline{\mathbf{G}}_{\overline{S}'}\end{pmatrix} = \begin{pmatrix}\mathbf{I}_{|S'|} & \mathbf{0}\end{pmatrix} = \mathbf{\Sigma}_{S'}$$

where $\mathbf{T}^\top(\mathbf{R}\mathbf{G}_1 - \mathbf{G}_2) = \mathbf{I}_{S'}$ by construction. □

In the next sections we assume that KeyGen and OblKeyGen do not abort. This is w.l.o.g. since we can always re-sample values when an abort happens. Note that in this case, the keys of both KeyGen and OblKeyGen are "somewhere perfectly binding".

4.3 Kronecker Product of two SSB commitments

Let CS be an algebraic commitment scheme and let $[\mathbf{G}]_1 \in \mathbb{G}_1^{\ell_1 \times n_1}$ and $[\mathbf{H}]_2 \in \mathbb{G}_2^{\ell_2 \times n_2}$ commitment keys. We note there's the following key and input homomorphism

$$\mathsf{CS.Com}([\mathbf{G}]_1, \boldsymbol{x}) \otimes \mathsf{CS.Com}([\mathbf{H}]_2, \boldsymbol{y}) = \mathsf{CS.Com}([\mathbf{G} \otimes \mathbf{H}]_T, \boldsymbol{x} \otimes \boldsymbol{y}),$$

where $\otimes$ is the Kronecker product and is naturally defined w.r.t. the pairing function when the operands are group elements. To get a structure preserving primitive, so that we can later efficiently show that committed values satisfy some relation, it is better to consider all keys defined over one of the base groups [AFG+16]. However, as noted in [GHR15a], in asymmetric groups it is not clear whether $[\mathbf{G} \otimes \mathbf{H}]_1$ (or $[\mathbf{G} \otimes \mathbf{H}]_2$) defines an SSB commitment. Indeed, if we use the ISH of CS_2 to prove that $[\mathbf{G} \otimes \mathbf{H}]_1$ is ISH, it turns out that we only know $[\mathbf{H}]_2$ in group $\mathbb{G}_2$ and hence we can only compute $\mathbf{G} \otimes [\mathbf{H}]_2$ which is trivially distinguishable from the original key. To overcome this problem, the authors in [GHR15a] used the split key $[\mathbf{Q}_1]_1 = [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1 \in \mathbb{G}_1^{\ell_1\ell_2 \times n_1 n_2}, [\mathbf{Q}_2]_2 = [-\mathbf{Z}] \in \mathbb{G}_2^{\ell_1\ell_2 \times n_1 n_2}$, for $\mathbf{Z} \leftarrow \mathbb{Z}_p^{\ell_1\ell_2 \times n_1 n_2}$. In this case we can write the homomorphism as follows

$$\begin{aligned} &\mathsf{CS.Com}([\mathbf{G}]_1, \boldsymbol{x}) \otimes \mathsf{CS.Com}([\mathbf{H}]_2, \boldsymbol{y}) = \\ &\quad e(\mathsf{CS.Com}([\mathbf{Q}_1]_1, \boldsymbol{x} \otimes \boldsymbol{y}), [1]_2) + e([1]_1, \mathsf{CS.Com}([\mathbf{Q}_2]_2, \boldsymbol{x} \otimes \boldsymbol{y}). \end{aligned} \tag{1}$$

If additionally CS is an instance of the scheme defined in figure 3, we prove the following theorem.

Theorem 5. *For $n_i \in \mathbb{N}, K_i \leq n_i, S_i \subseteq [n_i]$ and $|S_i| \leq K_i$, let CS_1 and CS_2 be two instances of the SSB commitment of figure 3 such that $(ck_i, sk_i) \leftarrow \mathsf{CS}_i.\mathsf{KGen}(gk_i, m_i, K_i, S_i)$ outputs a key over $\mathbb{G}_i$, where $i \in \{1, 2\}$. Then the commitment scheme kCS, where $\mathsf{kCS.KGen}(gk, (n_1, n_2), (K_1, K_2), (S_1, S_2))$ is defined as*

$\mathsf{kCS.KGen}(gk, ck_1, ck_2, sk_1, sk_2)$://$(ck_i, sk_i) \leftarrow \mathsf{CS}_1.\mathsf{KGen}(gk, m_i, K_i, S_i)$

1. *Parse sk_1 as $(\mathbf{G}, \mathbf{T}_\mathbf{G})$ and sk_2 as $(\mathbf{H}, \mathbf{T}_\mathbf{H})$.*
2. *Let $\mathbf{Q}_1 = \mathbf{G} \otimes \mathbf{H} + \mathbf{Z}$ and $\mathbf{Q}_2 = -\mathbf{Z}$, where $\mathbf{Z} \leftarrow \mathbb{Z}_p^{\overline{K}_1\overline{K}_2 \times n_1 n_2}$.*
3. *Let $\mathbf{T}_\mathbf{Q} = \mathbf{T}_\mathbf{G} \otimes \mathbf{T}_\mathbf{H}$ and $\mathsf{aux} = (ck_1, ck_2)$.*
4. *output $ck = ([\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathsf{aux})$ and $sk = (\mathbf{T}_\mathbf{Q}, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{G}, \mathbf{H})$.*

is a split algebraic oblivious SSB commitment scheme.

Proof. Index Set Hiding. Let $S_1, S_1' \subseteq [n_1], |S_1|, |S_1'| \leq K_1$ and S_2. The result follows from the indistinguishability of the following distributions (this is essentially part of the proof in [GHR15b, Theorem 6]). For simplicity we write $\mathbf{X} \leftarrow \mathsf{CS.KGen}$, where $\mathbf{X}$ is some part of (ck, sk), meaning that after running KGen we discard everything but $\mathbf{X}$. Recall that $\mathsf{aux} = ([\mathbf{G}]_1, [\mathbf{H}]_2)$.

1. $\mathsf{aux}, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [-\mathbf{Z}]_2,$ $\quad \mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1),\ \mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2),$
2. $\mathsf{aux}, [\mathbf{G}]_1 \otimes \mathbf{H} + [\mathbf{Z}]_1, [-\mathbf{Z}]_2,$ $\quad \mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1),\ \mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2),$
3. $\mathsf{aux}, [\mathbf{G}]_1 \otimes \mathbf{H} + [\mathbf{Z}]_1, [-\mathbf{Z}]_2,$ $\quad \mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1'),\ \mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2),$
4. $\mathsf{aux}, [\mathbf{Z}]_1, \mathbf{G} \otimes [\mathbf{H}]_2 - [\mathbf{Z}]_2,$ $\quad \mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1'),\ \mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2),$
5. $\mathsf{aux}, [\mathbf{Z}]_1, \mathbf{G} \otimes [\mathbf{H}]_2 - [\mathbf{Z}]_2,$ $\quad \mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1'),\ \mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2'),$

6. $\mathsf{aux}, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [-\mathbf{Z}]_2$, $\mathbf{G} \leftarrow \mathsf{CS.Setup}(gk, n_1, K_1, S_1')$, $\mathbf{H} \leftarrow \mathsf{CS.Setup}(gk, n_2, K_2, S_2')$.

Perfect indistinguishability between distributions 1-2, 3-4 and 5-6 follows from the fact that always both distributions are uniformly distributed conditioned on their sum being equal to $\mathbf{G} \otimes \mathbf{H}$. On the other hand, computational indistinguishability of distributions 2-3 and 4-5 follows from the ISH of CS_1 and CS_2 respectively.

Somewhere Statistically Binding and G-Extractability. Let $\boldsymbol{z}, \boldsymbol{z}' \in \mathbb{Z}_p^{n_1 n_2}$ such that $\mathsf{kCS.Com}(ck, \boldsymbol{z}) = \mathsf{kCS.Com}(ck, \boldsymbol{z}')$. Let $\mathbf{T_G}$ and $\mathbf{T_H}$ the trapdoors associated to $[\mathbf{G}]_1$ and $[\mathbf{H}]_2$, respectively, then

$$\begin{aligned}
0 &= (\mathbf{T_G} \otimes \mathbf{T_H})(\mathbf{G} \otimes \mathbf{H} + \mathbf{Z})(\boldsymbol{z} - \boldsymbol{z}') - (\mathbf{T_G} \otimes \mathbf{T_H})\mathbf{Z}(\boldsymbol{z} - \boldsymbol{z}') \\
&= (\mathbf{T_G} \otimes \mathbf{T_H})(\mathbf{G} \otimes \mathbf{H})(\boldsymbol{z} - \boldsymbol{z}') \\
&= (\mathbf{T_G}\mathbf{G}) \otimes (\mathbf{T_H}\mathbf{H})(\boldsymbol{z} - \boldsymbol{z}') \\
&= (\boldsymbol{\Sigma}_{S_1}\mathbf{P}_{S_1}) \otimes (\boldsymbol{\Sigma}_{S_2}\mathbf{P}_{S_2})(\boldsymbol{z} - \boldsymbol{z}') \\
&= (\boldsymbol{\Sigma}_{S_1} \otimes \boldsymbol{\Sigma}_{S_2})(\mathbf{P}_{S_1} \otimes \mathbf{P}_{S_2})(\boldsymbol{z} - \boldsymbol{z}') \\
&= \boldsymbol{z}_{S_1,S_2} - \boldsymbol{z}'_{S_1,S_2},
\end{aligned}$$

Note that this also shows that the trapdoors correctly extracts $[\boldsymbol{z}_{S_1,S_2}]_T$ from $\mathsf{kCS.Com}(ck, \boldsymbol{z})$.

Oblivious Trapdoor Generation. We first recall the following commutative property of kronecker products.

Fact 3. *For every $m_1, m_2, n_1, n_2 \in \mathbb{N}$ there exists permutation matrices $\boldsymbol{\Pi}_1 \in \{0,1\}^{m_1 n_1 \times m_1 n_1}, \boldsymbol{\Pi}_2 \in \{0,1\}^{m_2 n_2 \times m_2 n_2}$ such that for any pair of matrices $\mathbf{M} \in \mathbb{Z}_p^{m_1 \times m_2}, \mathbf{N} \in \mathbb{Z}_p^{n_1 \times n_2}$ it holds that $\mathbf{M} \otimes \mathbf{N} = \boldsymbol{\Pi}_1(\mathbf{N} \otimes \mathbf{M})\boldsymbol{\Pi}_2$. Note that $\boldsymbol{\Pi}_1$ and $\boldsymbol{\Pi}_2$ depend only on the size of $\mathbf{M}$ and $\mathbf{N}$ but not the values of their entries.*

We construct an oblivious key generation algorithm as follows.

$\mathsf{kCS.OblKeyGen}(gk, (n_1, n_2), (K_1, K_2), (S_1, S_2), ck)$:

1. Parse ck as $[\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2$ and $\mathsf{aux} = ([\mathbf{G}]_1, [\mathbf{H}]_2)$.
2. Run $([\mathbf{G}^*], \mathbf{T}_1) \leftarrow \mathsf{CS}_1.\mathsf{OblKeyGen}(gk, n_1, K_1, S_1, [\mathbf{G}]_1)$ and $([\mathbf{H}^*]_2, \mathbf{T}_2) \leftarrow \mathsf{CS}_2.\mathsf{OblKeyGen}(gk, m_2, K_2, S_2, [\mathbf{H}]_2)$ and use the random coins of $\mathsf{OblKeyGen}$ to retrieve $\mathbf{G}^*_{S_1}, \mathbf{R}_1$ and $\mathbf{H}^*_{S_1}, \mathbf{R}_2$ such that

$$[\mathbf{G}^*]_1 = \begin{pmatrix} [\mathbf{G}^*_{S_1}]_1 & [\overline{\mathbf{G}}_{\overline{S}_1}]_1 \\ \mathbf{R}_1[\overline{\mathbf{G}}_{\overline{S}_1}]_1 \end{pmatrix} \mathbf{P}_{S_1} \text{ and } \mathbf{H}^* = \begin{pmatrix} [\mathbf{H}^*_{S_2}]_2 & [\overline{\mathbf{H}}_{\overline{S}_2}]_2 \\ \mathbf{R}_2[\overline{\mathbf{H}}_{\overline{S}_2}]_2 \end{pmatrix} \mathbf{P}_{S_2},$$

   as defined in Fig. 3.
3. Let $[\mathbf{A}_1]_1, [\mathbf{A}_2]_2$ be the matrices containing the first $(K_1 + k - |S_1|)(K_2 + k)$ rows of $[(\mathbf{Q}_1)_{\overline{S}_1,\overline{S}_2}]_1$ and $[(\mathbf{Q}_2)_{\overline{S}_1,\overline{S}_2}]_2$, respectively.
4. Let $\boldsymbol{\Pi}_1$ and $\boldsymbol{\Pi}_2$ the permutation matrices of Fact 3 for matrices with $(K_1 + k - |S|_1)$ and $(K_2 + k)$ rows, and $n_1 - |S_1|$ and $n_2 - |S_2|$ columns.
5. Define $[\mathbf{B}_1]_1$ and $[\mathbf{B}_2]_2$ be the matrices of the first $(K_1 + k - |S_1|)(K_2 + k - |S_2|)$ columns of $\boldsymbol{\Pi}_1^\top[\mathbf{A}_1]_1\boldsymbol{\Pi}_2^\top$ and $\boldsymbol{\Pi}_1^\top[\mathbf{A}_2]_2\boldsymbol{\Pi}_2^\top$, respectively.
6. Let $[\mathbf{A}_1^*]_1 = \boldsymbol{\Pi}_1 \begin{pmatrix} [\mathbf{B}_1]_1 \\ (\mathbf{R}_2 \otimes \mathbf{I}_{K_1+k-|S_1|})[\mathbf{B}_1]_1 \end{pmatrix} \boldsymbol{\Pi}_2$ and $[\mathbf{A}_2^*]_2 = \boldsymbol{\Pi}_1 \begin{pmatrix} [\mathbf{B}_2]_2 \\ (\mathbf{R}_2 \otimes \mathbf{I}_{K_1+k-|S_1|})[\mathbf{B}_2]_2 \end{pmatrix} \boldsymbol{\Pi}_2$.
7. Pick $\mathbf{Z} \leftarrow \mathbb{Z}_p^{(K_1+k)(K_2+k) \times n_1 n_2}$ and let

$$[\mathbf{Q}_1^*]_1 = \left([\mathbf{Z}_{S_1,[n_2]}]_1 \middle| [\mathbf{G}^*_{\overline{S}_1}]_1 \otimes \mathbf{H}^*_{S_2} + [\mathbf{Z}_{\overline{S}_1,S_2}]_1 \middle| \begin{pmatrix} [\mathbf{A}_1^*]_1 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})[\mathbf{A}_1^*]_1 \end{pmatrix} + [\mathbf{Z}_{\overline{S}_1,\overline{S}_2}]_1 \right) (\mathbf{P}_{S_1} \otimes \mathbf{P}_{S_2})$$

$$[\mathbf{Q}_2^*]_2 = \left(\mathbf{G}^*_{S_1} \otimes [\mathbf{H}^*]_2 - [\mathbf{Z}_{S_1,[n_2]}]_1 \middle| - [\mathbf{Z}_{\overline{S}_1,S_2}]_2 \middle| \begin{pmatrix} [\mathbf{A}_2^*]_2 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})[\mathbf{A}_2^*]_2 \end{pmatrix} - [\mathbf{Z}_{\overline{S}_1,\overline{S}_2}]_2 \right) (\mathbf{P}_{S_1} \otimes \mathbf{P}_{S_2})$$

8. Let $\mathsf{aux} = ([\mathbf{G}^*]_1, [\mathbf{H}^*]_2)$ and $\mathbf{T} = \mathbf{T}_1 \otimes \mathbf{T}_2$.

9. Return $(ck = ([\mathbf{Q}_1^*]_1, [\mathbf{Q}_2^*]_2, \mathsf{aux}), \tau = \mathbf{T})$.

Now we show that ck is correctly distributed. Since CS_1 and CS_2 are both oblivious SSB commitments, it holds that $\mathsf{aux} = [\mathbf{G}^*]_1, [\mathbf{H}^*]$ follows the same distribution as the honest aux. It is enough to show that $\mathbf{Q}^* = \mathbf{Q}_1^* + \mathbf{Q}_2^* = \mathbf{G}^* \otimes \mathbf{H}^*$. This is the case since if this holds, the commitment key $[\mathbf{Q}_1^*]_1, [\mathbf{Q}_2^*]_2$ consists of two uniform matrices, conditioned on their sum equaling $\mathbf{G}^* \otimes \mathbf{H}^*$, and this is the distribution of the honest key as well.

It is clear that this is the case for $\mathbf{Q}^*_{S_1,[n_2]}$ and $\mathbf{Q}^*_{\overline{S}_1,S_2}$, so we show it is also the case for $\mathbf{Q}^*_{\overline{S}_1,\overline{S}_2}$.

First, note that $\mathbf{Q}_{\overline{S}_1,\overline{S}_2} = (\mathbf{Q}_1 + \mathbf{Q}_2)_{\overline{S}_1,\overline{S}_2} = \begin{pmatrix} \overline{\mathbf{G}}_{\overline{S}_1} \otimes \mathbf{H}_{\overline{S}_2} \\ \underline{\mathbf{G}}_{\overline{S}_1} \otimes \mathbf{H}_{\overline{S}_2} \end{pmatrix}$ and then $\mathbf{A} = \mathbf{A}_1 + \mathbf{A}_2 = \overline{\mathbf{G}}_{\overline{S}_1} \otimes \mathbf{H}_{\overline{S}_2}$. It follows that $\mathbf{\Pi}_1^\top \mathbf{A} \mathbf{\Pi}_2^\top = \mathbf{\Pi}_1^\top \mathbf{\Pi}_1 \mathbf{H}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \mathbf{\Pi}_2 \mathbf{\Pi}_2^\top = \begin{pmatrix} \overline{\mathbf{H}}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \\ \underline{\mathbf{H}}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \end{pmatrix}$ and hence $\mathbf{B} = \mathbf{B}_1 + \mathbf{B}_2 = \overline{\mathbf{H}}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1}$. Finally we have that

$$
\begin{aligned}
\mathbf{Q}^*_{\overline{S}_1,\overline{S}_2} &= \begin{pmatrix} \mathbf{A}_1^* + \mathbf{A}_2^* \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\mathbf{A}_1^* + \mathbf{A}_2^*) \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{\Pi}_1 \begin{pmatrix} \mathbf{B}_1 + \mathbf{B}_2 \\ (\mathbf{R}_2 \otimes \mathbf{I}_{K_1+k-|S_1|})(\mathbf{B}_1 + \mathbf{B}_2) \end{pmatrix} \mathbf{\Pi}_2 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\mathbf{A}_1^* + \mathbf{A}_2^*) \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{\Pi}_1 \begin{pmatrix} \overline{\mathbf{H}}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \\ (\mathbf{R}_2 \otimes \mathbf{I}_{K_1+k-|S_1|})(\overline{\mathbf{H}}_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1}) \end{pmatrix} \mathbf{\Pi}_2 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\mathbf{A}_1^* + \mathbf{A}_2^*) \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{\Pi}_1 \begin{pmatrix} \overline{\mathbf{H}}_{\overline{S}_2} \\ \mathbf{R}_2 \overline{\mathbf{H}}_{\overline{S}_2} \end{pmatrix} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \mathbf{\Pi}_2 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\mathbf{A}_1^* + \mathbf{A}_2^*) \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{\Pi}_1 \mathbf{H}^*_{\overline{S}_2} \otimes \overline{\mathbf{G}}_{\overline{S}_1} \mathbf{\Pi}_2 \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\mathbf{A}_1^* + \mathbf{A}_2^*) \end{pmatrix} \\
&= \begin{pmatrix} \overline{\mathbf{G}}_{\overline{S}_1} \otimes \mathbf{H}^*_{\overline{S}_2} \\ (\mathbf{R}_1 \otimes \mathbf{I}_{K_2+k})(\overline{\mathbf{G}}_{\overline{S}_1} \otimes \mathbf{H}^*_{\overline{S}_2}) \end{pmatrix} \\
&= \mathbf{G}^*_{\overline{S}_1} \otimes \mathbf{H}^*_{\overline{S}_2}.
\end{aligned}
$$

For finishing the proof it suffices to show that the rest of the input given to the distinguisher is correctly distributed. Note that, following definition 4 and fact 2, $[\boldsymbol{y}_{S_1',S_2'}]_T = (\mathsf{Extract}(\mathbf{T}_{S_1,S_2}, [\boldsymbol{c}]_1, [\boldsymbol{d}]_2))_{S_1',S_2'} = [\boldsymbol{z}_{S_1',S_2'|S_1',S_2}]_T = [\boldsymbol{z}_{S_1',S_2'}]_T = \mathsf{Extract}(\mathbf{T}_{S_1',S_2'}, [\boldsymbol{c}]_1, [\boldsymbol{d}]_2)$. □

Corollary 1. *Construction from fig. 3 instantiated in $\mathbb{G}_1$ is ISH even when the adversary is given $h(sk) = ([\mathbf{H}]_2, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [-\mathbf{Z}]_2)$. Similarly, it is also ISH when instantiated in $\mathbb{G}_2$ when the adversary is given $h(sk) = ([\mathbf{G}]_1, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [-\mathbf{Z}]_2)$.*

Proof. Follows directly from the ISH of the kronecker SSB commitment of Theorem 5. Specifically, ISH for $\mathbb{G}_1$ follows from the indistinguishability of distributions 1 to 3 from Theorem 5, and ISH for $\mathbb{G}_2$ follows from the indistinguishability of distributions 3 to 6. □

5 Quasi-Arguments with Preprocessing

In this section we introduce an extension of Quasi Arguments as defined in [KPY19] which adds support for language dependent crs or preprocessing such as the so called QA-NIZK arguments [JR13]. Additionally we use different languages for completeness and local soundness, i.e. promise problems, to incorporate the "knowledge transfer" soundness of [GR19].

Following [JR13], languages are parametrized by $\rho \in \mathcal{L}_{\mathsf{par}}$ and ρ sampled from some distribution $\mathcal{D}_{\mathsf{par}}$. We say tat $\mathcal{D}_{\mathsf{par}}$ is witness samplable if ρ can be efficiently sampled together with a witness θ for $\rho \in \mathcal{L}_{\mathsf{par}}$. We simply write $(\theta, \rho) \leftarrow \mathcal{D}_{\mathsf{par}}$. Each $\rho \in \mathcal{L}_{\mathsf{par}}$ defines a language $\mathcal{L}_\rho$ with the corresponding relations $\mathcal{R}^{\mathsf{yes}}_\rho$, that is $\mathcal{L}_\rho = \{x \mid \exists w \text{ s.t. } (x, w) \in \mathcal{R}^{\mathsf{yes}}_\rho\}$. After the language is fixed there is a (language dependent) prepossessing stage where a common reference string is generated. Going a step forward, we would like our statements to be commitments and that $\mathcal{R}^{\mathsf{yes}}_\rho$ puts some restriction on the commitment opening. Since we will be using SSB commitments, the language parameter must contain the SSB commitment key. Therefore, we assume distribution $\mathcal{D}_{\mathsf{par}}$ receives as input $d \in \mathbb{N}$ (the size of the opening), a locality parameter $K \leq d$ and a set $S \subseteq [d]$. It will be useful to define $\mathcal{L}^{\mathsf{yes}}_\rho = \mathcal{L}_\rho$ and $\mathcal{L}^{\mathsf{no}}_\rho$ the complement of $\mathcal{L}^{\mathsf{yes}}_\rho$, and similarly define $\mathcal{R}^{\mathsf{yes}}_\rho$ and $\mathcal{R}^{\mathsf{no}}_\rho$. Traditional arguments of knowledge require that from any accepting statement and proof pair one can extract a witness w such that $(x, w) \in \mathcal{R}^{\mathsf{no}}_\rho$ only with negligible probability.In a quasi-argument of knowledge only a small part of the witness w_S is extracted and $(x, w_S) \in \mathcal{R}^{\mathsf{yes}}_{\rho,S}$ with overwhelming probability, where $\mathcal{R}^{\mathsf{yes}}_{\rho,S}$ is a "local version" of $\mathcal{R}^{\mathsf{yes}}_\rho$. [10]

Our final addition is support for arguments of knowledge transfer (AoKT) [GR19]. In a nutshell, an AoKT enables to "succinctly reuse" an AoK of the opening of some commitment C for constructing another AoK for commitment D. That is, given an opening w for C, it enables to give a succinct proof that D opens to $g(w)$. Importantly, AoKTs can be based on falsifiable assumptions. Following [GR19], $\rho \in \mathcal{L}_{\mathsf{par}}$ defines languages $\mathcal{L}^{\mathsf{yes}}_\rho$ and $\mathcal{L}^{\mathsf{no}}_\rho$, with $\mathcal{L}^{\mathsf{no}}_\rho$ not necessarily the complement of $\mathcal{L}^{\mathsf{yes}}_\rho$ (i.e. a promise problem), with their corresponding relations $\mathcal{R}^{\mathsf{yes}}_\rho$ and $\mathcal{R}^{\mathsf{no}}_\rho$. For no instances, the adversary provides a promise w^* for x. In [GR19] $x = (C, D)$ and $(C, D, w^*) \in \mathcal{L}^{\mathsf{no}}_\rho$ if w^* is an opening for C but $g(w^*)$ is not an opening for D. In our instantiations x will be two SSB commitments to $C_1, \ldots, C_d$ and $D_1, \ldots, D_d$ such that C_i opens to w and D_i to $g_i(w)$. From the two SSB commitments we can extract C_S and D_S. Furthermore, C_i and D_i might not be extractable (actually, they will be Pedersen commitments) an hence the extractor can only compute $f(w, S) = \{\mathsf{Com}(ck_i, w) : i \in S\}$.

We define the yes and no languages as

$$\mathcal{L}^{\mathsf{yes}}_\rho = \{x \mid \exists w \text{ s.t. } (x, w) \in \mathcal{R}^{\mathsf{yes}}_{\rho,S}\}, \qquad \mathcal{L}^{\mathsf{no}}_\rho = \{(x, w^*) \mid \exists y \text{ s.t. } (x, y, w^*) \in \mathcal{R}^{\mathsf{no}}_{\rho,S}\},$$

where w^* is the promise of the adversary and y is the local f-witness that we can extract from the adversary. Intuitively, the two witnesses of the languages are different kind of objects. Witness y is the value we extract from the adversary, which can't be equal to $f(w, S)$ for successful adversaries, but should lie the image of f anyway. On the other hand w is a "proper" witness from which an y can be computed and hence belongs to the preimage of f.[11]

5.1 Arguments with No-signaling extraction and Oblivious CRS Generation

Similarly to the way we treated commitment schemes, we don't directly prove the existence of no-signaling extractors but first show the existence of an Oblivious CRS Generation algorithm. We then show the latter notion implies the former. For convenience, we start defining a quasi argument without no-signaling extraction but only local soundness. For local soundness, we use a weaker variant of the strong Quasi-Adaptive soundness of [JR13] where the adversary chooses $(\rho, \theta) \in \mathcal{L}_{\mathsf{par}}$. Instead, we honestly sample parameter ρ and reveal part of the witness $h_{ls}(\theta)$ to the adversary, for some function h_{ls}. When we don't require computational assumptions on ρ, as in quasi arguments of membership in a linear space, h_{ls} might be the identity function and then our definition becomes strong soundness as defined in [JR13]. In knowledge transfer arguments, soundness holds provided the hardness of some computational assumption defined by ρ. For this reason h_{ls} can't be the identity and some part of θ must remain hidden.

In practice h_{ls} models correlated information leaked by another protocol, typically as a result of sharing the commitment keys. If local knowledge soundness holds even when the adversary is given $h_{ls}(\theta)$, it means

[10] In the case x is a 3-CNF formula, in [KPY19] the authors define $\mathcal{R}^{\mathsf{yes}}_{\rho,S}$ as the pairs (x, w) where w is a "locally satisfying assignment". This means that every clause C in x with all variables in S, is satisfied by w.

[11] The original definition from [GR19] is syntactically different as w is part of the statement in the yes language. However, as the authors said, the verifier can't read w as it will render the verification process not succinct. Since y becomes irrelevant, we prefer to eliminate it from the yes language.

that any other protocol for which the crs can be derived from $h_{ls}(\theta)$ can be safely executed with a "correlated crs".

It will be useful to consider vectors of sets of size t. Namely $\boldsymbol{S} = (\boldsymbol{S}_1, \ldots, \boldsymbol{S}_t)$, for some $t \in \mathbb{N}$.

Definition 7. *An h_{ls}-*strong locally extractable proof system Π *for the parameter language $\mathcal{L}_{\mathsf{par}}$ and relations $\mathcal{R}^{\mathsf{yes}}_{\rho}, \mathcal{R}^{\mathsf{no}}_{\rho,\boldsymbol{S}}$ is a tuple of PPT algorithms $\Pi = (\mathsf{K}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Extract})$ where*

- $(\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S})$*: Parameter generation $\mathcal{D}_{\mathsf{par}}$ takes as input a group key gk, the locality parameter $\boldsymbol{K}$ and a set $\boldsymbol{S} \subseteq ([d], \ldots, [d])$ with $|\boldsymbol{S}| \leq \boldsymbol{K}$; it outputs an instance witness pair (ρ, θ) of $\mathcal{L}_{\mathsf{par}}$.*
- $(\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta)$*: K takes as input an instance-witness pair (ρ, θ) of $\mathcal{L}_{\mathsf{par}}$; it outputs a common reference string crs and an extraction trapdoor τ.*
- $\pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w)$*: Prove takes as input crs and a statement-witness pair (x, w) of $\mathcal{L}^{\mathsf{yes}}_{\rho}$; it outputs a proof π.*
- $b \leftarrow \mathsf{Verify}(\mathsf{crs}, x, \pi)$*: Verify takes as input crs, a statement x and a proof π; it outputs a bit b indicating if the proof π is a valid proof.*
- $y \leftarrow \mathsf{Extract}(\tau, x, \pi)$*: $\mathsf{Extract}$ takes as input the extraction trapdoor τ, a statement x and a proof π, and outputs a local witness y for the set $\boldsymbol{S}$.*

For all $\boldsymbol{\kappa} \in \mathbb{N}^t, \boldsymbol{K} \leq (d, \ldots, d) \in \mathbb{N}^t, \boldsymbol{S} \subseteq ([d], \ldots, [d])$, with $|\boldsymbol{S}| \leq \boldsymbol{K}$, Π satisfies the following properties:

- Completeness: *For all $(\rho, \theta) \in \mathcal{L}_{\mathsf{par}}$ and $x, w \in \{0,1\}^*$*

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\begin{array}{c|r} \mathsf{Verify}(\mathsf{crs}, x, \pi) = 1 & (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta); \\ \vee\ (x, w) \notin \mathcal{R}^{\mathsf{yes}}_{\rho,\boldsymbol{S}} & \pi \leftarrow \mathsf{Prove}(\mathsf{crs}, x, w) \end{array} \right] \geq 1 - \mathsf{negl}(\kappa)$$

- h_{ls}-Strong Local Knowledge Soundness: *For all PPT $\mathcal{A}$*

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\begin{array}{c|r} \mathsf{Verify}(\mathsf{crs}, x, \pi) = 0 & (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}); \\ \vee\ (x, y, w^*) \notin \mathcal{R}^{\mathsf{no}}_{\rho,\boldsymbol{S}} & (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta); \\ & (x, w^*, \pi) \leftarrow \mathcal{A}(\rho, h_{ls}(\theta), \mathsf{crs}); \\ & y \leftarrow \mathsf{Extract}(\tau, x, \pi) \end{array} \right] \geq 1 - \mathsf{negl}(\kappa)$$

Next, we define the no-signaling property of quasi-arguments. Similarly as with strong knowledge soundness, we consider a stronger definition where the adversary is given some function of θ, namely $h_{ns}(\theta)$.

Definition 8. *An h_{ls}-strong locally extractable proof system Π for the parameter language $\mathcal{L}_{\mathsf{par}}$ and relations $\mathcal{R}^{\mathsf{yes}}_{\rho,\boldsymbol{S}}, \mathcal{R}^{\mathsf{no}}_{\rho,\boldsymbol{S}}$ is an (h_{ls}, h_{ns})-*quasi argument *if it satisfies h_{ns}-*strong no-signaling extraction*. That is, for all $\kappa \in \mathbb{N}, \boldsymbol{K} \leq d \in \mathbb{N}^t, \boldsymbol{S}' \subseteq \boldsymbol{S} \subseteq ([d], \ldots, [d])$ with $|\boldsymbol{S}| \leq \boldsymbol{K}$, and all PPT $\mathcal{A}$ and PPT $\mathcal{D}$*

$$\left| \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\mathsf{crs}, x, \pi, y_{\boldsymbol{S}'}) = 1 \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}); \\ (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta); \\ (x, \pi) \leftarrow \mathcal{A}(\rho, h_{ns}(\theta), \mathsf{crs}); \\ \textit{if } \mathsf{Verify}(\mathsf{crs}, x, \pi) = 0\textit{: set } x = \bot; \\ y \leftarrow \mathsf{Extract}(\tau, x, \pi) \end{array} \right] - \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\mathsf{crs}, x, \pi, y') = 1 \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}'); \\ (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta); \\ (x, \pi) \leftarrow \mathcal{A}(\rho, h_{ns}(\theta), \mathsf{crs}); \\ \textit{if } \mathsf{Verify}(\mathsf{crs}, x, \pi) = 0\textit{: set } x = \bot; \\ y' \leftarrow \mathsf{Extract}(\tau, x, \pi) \end{array} \right] \right| \leq \mathsf{negl}(\kappa)$$

Finally, we define the notion of oblivious locally extractable proof systems. The requirements are that (1) the crs alone does not help PPT adversaries gain information about the extraction set used to sample the parameters ρ; (2) there exists a PPT algorithm $\mathsf{OblSetup}$ that on input a set $\boldsymbol{S}' \subseteq \boldsymbol{S}$ and (ρ, crs), sampled for extraction on the superset of $\boldsymbol{S}$, outputs new values (ρ', crs') that are statistically close to (ρ, crs) and additionally, it outputs a trapdoor τ' for $\boldsymbol{S}'$ that outputs indistinguishable witnesses to the ones output for $\boldsymbol{S}$ and restricted to $\boldsymbol{S}'$.

We consider also a "h_{ns}-strong" variant of (1). Note that (2) holds against unbounded adversaries which can compute θ by themselves.

Definition 9. *A locally extractable proof system* Π *for the parameter language* $\mathcal{L}_{\mathsf{par}}$ *and relations* $\mathcal{R}^{\mathsf{yes}}_{\rho}, \mathcal{R}^{\mathsf{no}}_{\rho,\boldsymbol{S}}$ *is* h_{ns}-Strong Oblivious *if there exist a PPT algorithm* $\mathsf{OblSetup}$ *such that, for all* $\kappa \in \mathbb{N}, \boldsymbol{K} \leq (d, \ldots, d) \in \mathbb{N}^t, \boldsymbol{S}', \boldsymbol{S} \subseteq ([d], \ldots, [d])$ *with* $|\boldsymbol{S}'|, |\boldsymbol{S}| \leq K$,

1. h_{ns}-Strong Index Set Hiding*: for all PPT* $\mathcal{D}$

$$\left| \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\rho, \mathsf{crs}, h_{ns}(\theta)) \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}) \\ (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta) \end{array} \right] - \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\rho, \mathsf{crs}, h_{ns}(\theta)) \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}') \\ (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta) \end{array} \right] \right| \leq \mathsf{negl}(\kappa)$$

2. Oblivious trapdoor Generation: *if* $\boldsymbol{S}' \subseteq \boldsymbol{S}$ *then for all, (even unbounded) adversaries* $\mathcal{A}$ *and distinguishers* $\mathcal{D}$

$$\left| \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\rho', \mathsf{crs}', y') = 1 \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}); (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta) \\ (\rho', \mathsf{crs}', \tau') \leftarrow \mathsf{OblSetup}(\rho, \mathsf{crs}, \boldsymbol{S}') \\ (x, \pi) \leftarrow \mathcal{A}(\rho, \mathsf{crs}') \\ \textit{if } \mathsf{Verify}(\mathsf{crs}, x, \pi) = 0\textit{: set } x = \bot; \\ y' \leftarrow \mathsf{Extract}(\tau', x, \pi) \end{array} \right] - \Pr_{gk \leftarrow \mathcal{G}(1^\kappa)} \left[\mathcal{D}(\rho, \mathsf{crs}, y_{\boldsymbol{S}'}) = 1 \;\middle|\; \begin{array}{r} (\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}); (\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta) \\ (x, \pi) \leftarrow \mathcal{A}(\rho, \mathsf{crs}) \\ \textit{if } \mathsf{Verify}(\mathsf{crs}, x, \pi) = 0\textit{: set } x = \bot; \\ y \leftarrow \mathsf{Extract}(\tau, x, \pi) \end{array} \right] \right| \leq \mathsf{negl}(\kappa)$$

Next, we present a proof that if a locally extractable proof system satisfies oblivious crs generation, then it is no-signaling. The proof is similar to the proof of Thm. 1.

Theorem 6. *Let* $\Pi = (\mathsf{K}, \mathsf{Prove}, \mathsf{Verify}, \mathsf{Extract}, \mathsf{OblSetup})$ *be an* h_{ns}*-strong Locally Extractable Proof System for the parameter language* $\mathcal{L}_{\mathsf{par}}$ *and relations* $\mathcal{R}^{\mathsf{yes}}_{\rho}, \mathcal{R}^{\mathsf{no}}_{\rho,\boldsymbol{S}}$*. Then,* Π *has* h_{ns}*-strong no signaling extraction.*

Proof. Fix any $\boldsymbol{S}' \subseteq \boldsymbol{S} \subseteq ([d], \ldots, [d])$ with $|\boldsymbol{S}| \leq \boldsymbol{K}$, and let $\mathcal{D}$ be a PPT distinguisher against no signaling extraction for these values, on instance-proof pairs output by a PPT $\mathcal{A}$. We show by a sequence of games that its success probability is negligible.

$\mathsf{Game}_0^{\mathcal{D},\mathcal{A}}(1^\kappa)$: We execute $(\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S})$; $(\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta)$; we then get $(x, \pi) \leftarrow \mathcal{A}(\rho, \mathsf{crs}, h_{ns}(\theta))$ and change x to $\bot$ if $\mathsf{Verify}(\mathsf{crs}, x, \pi) = 0$; we compute $y \leftarrow \mathsf{Extract}(\tau, x, \pi)$. The output is $\mathcal{D}(\mathsf{crs}, x, \pi, y_{\boldsymbol{S}'})$.

$\mathsf{Game}_1^{\mathcal{D},\mathcal{A}}(1^\kappa)$: We execute $(\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S})$; $(\mathsf{crs}, \tau) \leftarrow \mathsf{K}(\rho, \theta)$; we use the oblivious extractor to get $(\rho', \mathsf{crs}', \tau') \leftarrow \mathsf{OblSetup}(\rho, \mathsf{crs}, \boldsymbol{S}')$; we then get $(x, \pi) \leftarrow \mathcal{A}(\rho', \mathsf{crs}', h_{ns}(\theta))$ and change x to $\bot$ if $\mathsf{Verify}(\mathsf{crs}, x, \pi) = 0$; we compute $y' \leftarrow \mathsf{Extract}(\tau', x, \pi)$. The output is $\mathcal{D}(\mathsf{crs}, x, \pi, y')$.

$\mathsf{Game}_2^{\mathcal{D},\mathcal{A}}(1^\kappa)$: This is the same as $\mathsf{Game}_1^{\mathcal{D},\mathcal{A}}$ but in the first step we sample parameters for $\boldsymbol{S}'$, that is we execute $(\rho, \theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S}')$.

$\mathsf{Game}_3^{\mathcal{D},\mathcal{A}}(1^\kappa)$: We execute $(\rho,\theta) \leftarrow \mathcal{D}_{\mathsf{par}}(gk,d,K,\boldsymbol{S}')$; $(\mathsf{crs},\tau) \leftarrow \mathsf{K}(\rho,\theta)$; we then get $(x,\pi) \leftarrow \mathcal{A}(\rho,\mathsf{crs},h_{ns}(\theta))$ and change x to $\perp$ if $\mathsf{Verify}(\mathsf{crs},x,\pi)=0$; we compute $y' \leftarrow \mathsf{Extract}(\tau,x,\pi)$. The output is $\mathcal{D}(\mathsf{crs},x,\pi,y')$.

We next show that for all $1 \le i \le 3$,

$$\left|\Pr\left[\mathsf{Game}_i^{\mathcal{D},\mathcal{A}}(1^\kappa)=1\right] - \Pr\left[\mathsf{Game}_{i-1}^{\mathcal{D},\mathcal{A}}(1^\kappa)=1\right]\right| \le \mathsf{negl}(\kappa). \tag{2}$$

- *Case* $i=1$, $i=3$. Note that for $i=1$, the difference in the two games is exactly as in the two cases of the oblivious trapdoor generation property for $\boldsymbol{S}' \subseteq \boldsymbol{S}$, so the outputs of games are statistically close. For case 3, we use the same argument for $\boldsymbol{S}' \subseteq \boldsymbol{S}'$.
- *Case* $i=2$ The only difference in the games is how we setup the initial crs, either by sampling for $\boldsymbol{S}'$ or for $\boldsymbol{S}$. The output of the two games are computationally indistinguishable by the index set hiding property, even when the adversary is given $h_{ns}(\theta)$.

By a standard argument we get that, for all PPT $\mathcal{D},\mathcal{A}$,

$$\left|\Pr\left[\mathsf{Game}_0^{\mathcal{D},\mathcal{A}}(1^\kappa)=1\right] - \Pr\left[\mathsf{Game}_5^{\mathcal{D},\mathcal{A}}(1^\kappa)=1\right]\right| \le \mathsf{negl}(\kappa).$$

Finally, noting that $\mathsf{Game}_0^{\mathcal{D},\mathcal{A}}$, $\mathsf{Game}_3^{\mathcal{D},\mathcal{A}}$ correspond to the two cases of no signaling extraction, we conclude the proof. □

5.2 Succinct Pairing Based Quasi-Arguments

In this section we present quasi arguments for various languages using SSB commitments with oblivious trapdoor generation. We first present the simpler case, membership in linear spaces, and then we present some extensionsof it, specifically a knowledge transfer version, and a knowledge transfer version for statements split in the two groups. Finally,we use the latter to build a quasi argument of knowledge transfer for hadamard products.

5.2.1 Quasi Arguments of Membership in Linear Spaces

Let $\mathcal{U}$ be a witness samplable distributions sampling $([\mathbf{U}]_1,\mathbf{U})$, where $\mathbf{U} \in \mathbb{Z}_p^{d\times n}$. We assume that for any $S \subseteq [d]$, given only $[\mathbf{U}_S]_1$ such that $\mathbf{U} = \mathbf{P}_S^\top \begin{pmatrix} \mathbf{U}_S \\ \mathbf{U}_{\overline{S}} \end{pmatrix}$ there is an efficient way of sampling $[\mathbf{U}_{\overline{S}}]$.[12] Also, let CS be an algebraic SSB commitment key. The parameter language is

$$\begin{aligned}\mathcal{L}_{\mathsf{par}} = \{[\mathbf{U}]_1,[\mathbf{G}]_1 \mid \exists \mathbf{U},\mathbf{G} \text{ s.t. } &([\mathbf{U}]_1,\mathbf{U}) \in \mathsf{Sup}(\mathcal{U}) \text{ and} \\ &([\mathbf{G}]_1,\mathbf{G},\mathbf{T}) \in \mathsf{Sup}(\mathsf{CS.KeyGen}(gk,d,K,S))\}\end{aligned}$$

We assume that the corresponding relation is efficiently verifiable[13]. The parameters $\rho = ([\mathbf{U}]_1,[\mathbf{G}]_1) \leftarrow (\mathcal{U},\mathsf{CS.KeyGen}(gk,d,K,S))$ define the following relations:

$$\begin{aligned}&\mathcal{RL}_\rho^{\mathsf{yes}} = \{([\boldsymbol{c}]_1,\boldsymbol{w}) : \boldsymbol{c} = \mathbf{G}\mathbf{U}\boldsymbol{w}\},\\ &\mathcal{RL}_{\rho,S}^{\mathsf{no}} = \{([\boldsymbol{c}]_1,[\boldsymbol{y}]_1) : \boldsymbol{y} \text{ is a valid } S\text{-opening of } \boldsymbol{c} \text{ and } \boldsymbol{y} \notin \mathrm{Im}(\mathbf{U}_S)\}\end{aligned}$$

[12] We will instantiate the argument with $\mathbf{U}$ a block lower triangular matrix where each row is of the form $(\mathbf{U}_1,\mathbf{U}_2,\ldots,\mathbf{U}_i,\mathbf{0},\ldots,\mathbf{0})$ where $\{\mathbf{U}_i\}_i$ are independent random variables. Then is clear that from $[\mathbf{U}_S]_1$ we know $[\mathbf{U}_i]_1$ up to $i=\max S$, and the rest $\{\mathbf{U}_j : j \notin S\}$ can be sampled independently.

[13] This is w.l.o.g. since one can extend the witness to include the randomness used to sample the parameters.

The advice is the empty string while the extractor should retrieve $f(\boldsymbol{w}, S) = [\mathbf{U}_S]_1 \boldsymbol{w}$ from any accepting statement and proof pair. We present the construction QALin in Fig. 4. The construction is essentially the quasi adaptive construction of membership in linear space of [KW15] for the matrix $\mathbf{GU}$.

$\mathcal{D}_{\mathsf{par}}(gk, d, K, S)$**:**

- $([\mathbf{U}]_1, \mathbf{U}) \leftarrow \mathcal{U}$; $(pk, sk) \leftarrow \mathsf{CS.KeyGen}(gk, d, K, S)$
- Output (ρ, θ) where $\rho = (gk, pk, [\mathbf{U}]_1), \theta = (sk, \mathbf{U})$

$\mathsf{K}(\rho = (gk, [\mathbf{G}]_1, [\mathbf{U}]_1), \theta = (\mathbf{G}, \mathbf{T}, \mathbf{U}))$**:**

- Sample $\mathbf{K} \leftarrow \mathbb{Z}_p^{\overline{K} \times k}$, $\mathbf{A} \leftarrow \mathcal{D}_k$, and redefine $\mathbf{A}$ as its first k columns.
- Compute $[\mathbf{B}]_1 = [\mathbf{U}^\top]_1 \mathbf{G}^\top \mathbf{K}$, $\mathbf{C} = \mathbf{K}\mathbf{A}$.
- Output (crs, τ) where $\mathsf{crs} = ([\mathbf{A}]_2, [\mathbf{B}]_1, [\mathbf{C}]_2)$, $\tau = \mathbf{T}$.

$\mathsf{Prove}(\mathsf{crs} = ([\mathbf{A}]_2, [\mathbf{B}]_1, [\mathbf{C}]_2), [\boldsymbol{c}]_1, \boldsymbol{w})$**:** Output $[\boldsymbol{\pi}]_1 \leftarrow \boldsymbol{w}^\top [\mathbf{B}]_1$.

$\mathsf{Verify}(\mathsf{crs} = ([\mathbf{A}]_2, [\mathbf{B}]_1, [\mathbf{C}]_2), [\boldsymbol{c}]_1, [\boldsymbol{\pi}]_1)$**:** Output 1 if $e([\boldsymbol{\pi}]_1, [\mathbf{A}]_2) = e([\boldsymbol{c}^\top]_1, [\mathbf{C}]_2)$ and 0 otherwise.

$\mathsf{Extract}(\tau = \mathbf{T}, [\boldsymbol{c}]_1, [\boldsymbol{\pi}]_1)$**:** Output $[\boldsymbol{y}]_1 \leftarrow \mathbf{T}^\top [\boldsymbol{c}_S]_1$, otherwise output $\perp$.

Figure 4: Construction QALin for membership in linear spaces. Note that this is just the argument of [KW15] for matrix $[\mathbf{GU}]_1$.

Theorem 7. *Let $\mathcal{U}$ be a witness samplable distribution, $\mathcal{D}_k$ be a matrix distribution and CS an algebraic SSB commitment. Then, construction QALin of Fig. 4 is a locally extractable proof system with h_{ls}-strong local knowledge soundness where $h_{\mathsf{ls}}(\theta) = \theta$. Furthermore, completeness holds with probability 1 and h_{ls}-strong local knowledge soundness holds with probability at least $1 - \mathsf{Adv}^{\Pi_{\mathsf{lin}}}_{\mathsf{snd}}(\mathcal{B})$, whee $\mathcal{B}$ is a PPT adversary against the strong soundness of Π_{lin} of [KW15].*

Proof. For completeness, we have that if $\boldsymbol{c} = \mathbf{GU}\boldsymbol{w}$, then

$$\boldsymbol{c}^\top \mathbf{C} = (\mathbf{GU}\boldsymbol{w})^\top \mathbf{C} = \boldsymbol{w}^\top \mathbf{U}^\top \mathbf{G}^\top \mathbf{C} = \boldsymbol{w}^\top \mathbf{U}^\top \mathbf{G}^\top \mathbf{K}\mathbf{A} = \boldsymbol{w}^\top \mathbf{B}\mathbf{A} = \boldsymbol{\pi}\mathbf{A}.$$

Local knowledge soundness is guaranteed by the local extractability of the SSB commitment scheme and soundness of Kiltz and Wee proof system. Note that the extractor always outputs a valid partial opening of $[\boldsymbol{c}]_1$ given an accepting proof $[\pi]_1$, by the local extractability property of the SSB commitments. We claim that this opening must lie in $\mathrm{Im}([\mathbf{U}_S]_1)$. Assume otherwise, and let $\mathcal{A}$ be a PPT adversary that makes the extraction fail. We construct a PPT adversary $\mathcal{B}_S$ that breaks strong soundness of Kiltz and Wee for the matrix $\mathbf{U}_S$, conditioned on $\mathcal{A}$ giving a valid proof. $\mathcal{B}_S$ works as follows: it takes input crs_S containing $[\mathbf{U}_S]_1 \in \mathbb{G}^{|S| \times d}$, $[\mathbf{A}]_2 \in \mathbb{G}_2^{k \times k}$, $[\mathbf{B}^\dagger]_1 \in \mathbb{G}^{d \times k}$, $[\mathbf{C}^\dagger]_2 \in \mathbb{G}_2^{|S| \times k}$ and the discrete logarithms of matrix $\mathbf{U}_S$ and does the following:

- It samples $([\mathbf{U}_{\overline{S}}]_1, \mathbf{U}_{\overline{S}})$ s.t. $\mathbf{U} = \mathbf{P}_S^\top (\mathbf{U}_S / \mathbf{U}_{\overline{S}})$.
- It samples $([\mathbf{G}]_1, \mathbf{G}, \mathbf{T}) \leftarrow \mathsf{CS.KeyGen}(gk, n, d, K, S)$ and a random matrix $\mathbf{R} \leftarrow \mathbb{Z}_p^{K+k \times k}$.
- It computes $[\mathbf{B}]_1 = [\mathbf{B}^\dagger]_1 + [\mathbf{U}]^\top \mathbf{G}^\top \mathbf{R}$, $[\mathbf{C}]_2 = \mathbf{T}[\mathbf{C}^\dagger]_2 + \mathbf{R}[\mathbf{A}]_2$.
- It sets $\rho := (gk, [\mathbf{G}]_1, [\mathbf{U}]_1), \theta := (\mathbf{G}, \mathbf{U}, \mathbf{T})$ and $\mathsf{crs} := ([\mathbf{A}]_2, [\mathbf{B}]_1, [\mathbf{C}]_2)$.

It then executes $\mathcal{A}(\rho, \theta, \mathsf{crs})$ until it outputs $[\boldsymbol{c}]_1, [\pi]_1$. If this is an accepting proof pair, $\mathcal{B}_S$ sets $[\boldsymbol{x}^\dagger] := \mathbf{T}[\boldsymbol{c}]$ and $[\boldsymbol{\pi}^\dagger] := [\boldsymbol{\pi}]_1 - [\boldsymbol{c}]_1^\top \mathbf{R}$.

First, we claim that the values $\rho, \theta, \mathsf{crs}$ given as input to $\mathcal{A}$ are identically distributed to honestly created ones and thus do not skew the probability that $\mathcal{A}$ outputs a valid proof. This is immediate for ρ, θ since

they are sampled honestly. We show that this is true for crs as well. Let $\mathbf{K}^\dagger \in \mathbb{Z}^{|S|\times k}$ be the implicit matrix in crs_S, that is it satisfies $\mathbf{B}^\dagger = \mathbf{U}_S^\top \mathbf{K}^\dagger$ and $\mathbf{C}^\dagger = \mathbf{K}^\dagger \mathbf{A}$. Consider the matrix $\mathbf{K} = \mathbf{T}\mathbf{K}^\dagger + \mathbf{R}$, and note that this matrix is uniformly distributed since $\mathbf{R}$ is uniformly distributed. Thus $\mathbf{K}$ is distributed identically to an honestly generated $\mathbf{K}'$ for generating a crs. We claim that the crs crs output by $\mathcal{B}_S$ is identically distributed to sampling this matrix and computing the other values honestly. Indeed we have that

$$\begin{aligned}\mathbf{C} &= \mathbf{T}\mathbf{C}^\dagger + \mathbf{R}\mathbf{A} \\ &= \mathbf{T}\mathbf{K}^\dagger\mathbf{A} + \mathbf{R}\mathbf{A} \\ &= (\mathbf{T}\mathbf{K}^\dagger + \mathbf{R})\mathbf{A} \\ &= \mathbf{K}\mathbf{A}\end{aligned} \quad \text{and} \quad \begin{aligned}\mathbf{B} &= \mathbf{B}^\dagger + \mathbf{U}^\top\mathbf{G}^\top\mathbf{R} = \mathbf{U}_S^\top\mathbf{K}^\dagger + \mathbf{U}^\top\mathbf{G}^\top\mathbf{R} \\ &= \mathbf{U}^\top\mathbf{G}^\top\mathbf{T}\mathbf{K}^\dagger + \mathbf{U}^\top\mathbf{G}^\top\mathbf{R} \\ &= \mathbf{U}^\top\mathbf{G}^\top(\mathbf{T}\mathbf{K}^\dagger + \mathbf{R}) = (\mathbf{G}\mathbf{U})^\top\mathbf{K}\end{aligned}$$

where the second equality for $\mathbf{B}$ follows since by the properties of algebraic SSB commitments we have $\mathbf{T}^\top\mathbf{G} = (\mathbf{I}_{|S|}\ \mathbf{0})\mathbf{P}_S$ which gives

$$\mathbf{U}^\top\mathbf{G}^\top\mathbf{T} = \mathbf{U}^\top\mathbf{P}_S^\top \begin{pmatrix}\mathbf{I}_{|S|}\\ \mathbf{0}\end{pmatrix} = \mathbf{U}_S.$$

So, the outputted crs crs' is indeed identically distributed with an honest one.

Finally, we show that if $\mathcal{A}$ outputs a valid proof $[\boldsymbol{\pi}]_1$, then $\mathcal{B}_S$ outputs a valid statement-proof pair w.r.t. to crs_S. Indeed, by the local extractability property of the commitment scheme, $\mathcal{B}_S$ always outputs some $[\boldsymbol{x}^\dagger]_1$ consistent with $[\boldsymbol{c}]_1$, and also the proof verifies, since we have

$$\boldsymbol{\pi}\mathbf{A} = \boldsymbol{c}^\top\mathbf{C} = \boldsymbol{c}^\top\mathbf{K}\mathbf{A} = \boldsymbol{c}^\top(\mathbf{T}\mathbf{K}^\dagger + \mathbf{R})\mathbf{A} = (\boldsymbol{x}^\dagger)^\top\mathbf{K}^\dagger\mathbf{A} + \boldsymbol{c}^\top\mathbf{R}\mathbf{A}$$

which gives $\boldsymbol{\pi}^\dagger\mathbf{A} = \boldsymbol{\pi}\mathbf{A} - \boldsymbol{c}^\top\mathbf{R}\mathbf{A} = (\boldsymbol{x}^\dagger)^\top\mathbf{K}^\dagger\mathbf{A} = (\boldsymbol{x}^\dagger)\mathbf{C}^\dagger$. We conclude that $[\boldsymbol{\pi}^\dagger]_1$ is a valid proof for $[\boldsymbol{x}^\dagger]_1 \notin \mathrm{Im}([\mathbf{U}_S]_1)$ and $\mathcal{B}_S$ breaks soundness of Kiltz and Wee construction. □

Corollary 2. *Consider construction from Fig. 4 with a statement of the form* $\begin{pmatrix}[\boldsymbol{x}]_1\\ [\boldsymbol{y}]_1\end{pmatrix}$*, matrix* $\begin{pmatrix}\mathbf{U}\\ \mathbf{V}\end{pmatrix}$*, locality parameter* $\boldsymbol{L} \le (d,d) \in \mathbb{N}^2$ *and extraction set* $\boldsymbol{S} = (S_1, S_2) \subseteq ([d],[d]), |\boldsymbol{S}| \le \boldsymbol{L}$*, such that the* $(\mathbf{U}_{S_1}^\top, h)$-MDDH *assumption is hard for some function* h*. Assume also* $\mathbf{K} \leftarrow \mathbb{Z}_p^{L_1+L_2+2k\times k}$*,* $\mathbf{G} = \begin{pmatrix}\mathbf{G}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{G}_2\end{pmatrix}$*, where* $\mathbf{G}_i \leftarrow \mathsf{CS.KeyGen}(gk, d, L_i, S_i)$*, and* $\mathbf{A} \leftarrow \mathbb{Z}_p^{k\times k}, k \ge 2$*. Then construction from Fig. 4 is also a quasi argument for the relations* $\mathcal{KL}_\rho^{\mathsf{yes}} = \mathcal{RL}_\rho^{\mathsf{yes}}$ *and* $\mathcal{KL}_\rho^{\mathsf{no}} = \{[\boldsymbol{c}]_1, [\boldsymbol{d}]_1, [\boldsymbol{x}^*]_1, [\boldsymbol{y}^*]_1, \boldsymbol{w}^* : \begin{pmatrix}\boldsymbol{c}\\ \boldsymbol{d}\end{pmatrix}\ \boldsymbol{S}\text{-open to } \begin{pmatrix}\boldsymbol{x}^*\\ \boldsymbol{y}^*\end{pmatrix}$ *and* $\boldsymbol{x}^* = \mathbf{U}_{S_1}\boldsymbol{w}^*$ *but* $\boldsymbol{y}^* \neq \mathbf{V}_{S_2}\boldsymbol{w}^*\}$*, with* h_{ls}*-strong local soundness where* $h_{\mathsf{ls}}(\theta) = (h(\mathbf{U}_{S_1}^\top), \mathbf{G}, \mathbf{U}_{S_2}^\top)$*.*

Proof. In [GR19] it is shown that Kiltz and Wee argument is also a knowledge transfer argument whenever the $\mathbf{U}^\top$-MDDH assumption ($\mathbf{U}_{S_1}^\top$-MDDH in this case) holds and $\mathbf{A}$ is not full rank. Of course, this is still true if the stronger $(\mathbf{U}_S^\top, h)$-MDDH assumption holds. However in 4 $\mathbf{A}$ is full rank with overwhelming probability. Nevertheless, if $\mathbf{A}$ is uniform and $k \ge 2$ we can jump to a game (relying on the DDH assumption) where $\mathbf{A} \in \mathbb{Z}_p^{k\times k}$ is not full rank. Then the reduction of Thm. 7 yields also a reduction to the knowledge transfer of [KW15] (taking $\begin{pmatrix}\mathbf{T}_1\\ \mathbf{T}_2\end{pmatrix}$ as trapdoor, where $\mathbf{T}_i$ is the trapdoor for $\mathbf{G}_i$). □

The proof that QALin is oblivious essentially follows from the oblivious trapdoor generation and index set hiding of SSB commitments. Before proving oblivious trapdoor generation we present a lemma stating that we can also compute ρ, crs knowing only the commitment key $[\mathbf{G}]_1$ and $\mathbf{U}$, in both simple and knowledge transfer schemes.

Lemma 1. *There exists a modified* crs *generation algorithm* K' *that on input* (ρ, θ')*, where* θ' *contains only* $\mathbf{U}$ *(resp.* $\mathbf{U}, \mathbf{V}$*) outputs a* crs *such that* (ρ, crs) *are identically distributed to the honest algorithm.*

The lemma follows directly by noting that $[\mathbf{B}]_1 = [\mathbf{U}^\top]_1\mathbf{G}\mathbf{K} = \mathbf{U}^\top[\mathbf{G}]_1\mathbf{K}$. (resp. $[\mathbf{B}]_1 = [\mathbf{U}^\top \mid \mathbf{V}^\top]_1\mathbf{G}\mathbf{K} = (\mathbf{U}^\top \mid \mathbf{V}^\top)[\mathbf{G}]_1\mathbf{K}$. Given that this result holds, we slightly abuse notation and refer to $\mathsf{K}'(\rho, \theta')$ as $\mathsf{K}(\rho, \theta')$, that is we use the same name for the honest and the simulated algorithm.

Theorem 8. *Let $\mathcal{U}$ (resp. $\mathcal{U}, \mathcal{V}$ for the knowledge transfer case) be a witness samplable distribution, and* CS *be an algebraic SSB commitment scheme with perfect completeness, h-strong index set hiding and oblivious trapdoor generation. Then Construction* QALin *of Fig. 4 (resp. construction* Π *of corollary 2) is h_{ns}-strong oblivious where $h_{\mathsf{ns}} = (h(sk), \mathbf{U})$ (resp. $h_{\mathsf{ns}} = (h(sk), \mathbf{U}, \mathbf{V})$). Furthermore,*

1. *For every PPT $\mathcal{A}$ against h_{ns}-strong index set hiding of Π, there exists an adversary $\mathcal{B}$ against h-strong index set hiding property of* CS*, such that $\mathsf{Adv}^{\Pi}_{\mathsf{ISH}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{CS}}_{\mathsf{ISH}}(\mathcal{B})$ where $h_{\mathsf{ns}}(\theta) = (h(sk), \mathbf{U})$.*
2. *For every $\mathcal{A}$ against oblivious trapdoor generation of Π, there exists an adversary $\mathcal{B}$ against oblivious trapdoor generation of* CS*, such that $\mathsf{Adv}^{\Pi}_{\mathsf{oblv}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{CS}}_{\mathsf{oblv}}(\mathcal{B})$.*

Proof. For index set hiding, it is enough to notice that in both cases, the crs of Π can be efficiently computed given only $ck = ([\mathbf{G}]_1, h(\mathbf{G}))$. Indeed by sampling $[\mathbf{U}]_1, \mathbf{U} \leftarrow \mathcal{U}$ (resp. $[\mathbf{U}]_1, \mathbf{U} \leftarrow \mathcal{U}$; $[\mathbf{V}]_1, \mathbf{V} \leftarrow \mathcal{V}$) all values of crs are efficiently computable, as noted in the previous lemma. Additionally, since we assume CS is h-strong ISH, $\mathcal{A}$ can be also given $h_{\mathsf{ns}}(\theta) = (h(sk), \mathbf{U})$ (resp. $h_{\mathsf{ns}}(\theta) = (h(sk), \mathbf{U}, \mathbf{V})$). Thus, a distinguishing advantage in index set hiding of Π immediately implies equal advantage on the respective property of CS.

For oblivious trapdoor generation we first describe the $\mathsf{OblSetup}$ algorithm. Let $S' \subseteq S$.

$\mathsf{OblSetup}(\rho = ([\mathbf{G}]_1, [\mathbf{U}]_1), \mathsf{crs})$:

- $([\mathbf{G}']_1, \mathbf{T}') \leftarrow \mathsf{CS.OblSetup}(gk, d, K, S, ck = [\mathbf{G}]_1)$.
- $([\mathbf{U}]_1, \mathbf{U}) \leftarrow \mathcal{U}$ (resp. $([\mathbf{U}]_1, \mathbf{U}) \leftarrow \mathcal{U}$; $([\mathbf{V}]_1, \mathbf{V}) \leftarrow \mathcal{U}$).
- $(\mathsf{crs}, \tau) \leftarrow \Pi.\mathsf{K}(\rho, \theta' = \mathbf{U})$ (resp. $(\mathsf{crs}, \tau) \leftarrow \Pi.\mathsf{K}(\rho, \theta' = (\mathbf{U}, \mathbf{V}))$).

Note that the only difference in sampling with S and with S' is how we sample the commitment key $\mathbf{G}$. The crs part crs is identically distributed to an honest one by Lemma 1. Finally, by the statistically binding property of the commitment key the extracted witness for S and S' are unique and thus do not help the (unbounded) distinguisher, who can compute them on its own.

□

Corollary 3. *When* CS *is the one from fig. 3, then* Π *from fig. 4 (resp. corollary 2) is h_{ns}-strong no-signaling where $h_{\mathsf{ns}}(\theta) = (h(sk), \mathbf{U})$ (resp. $h_{\mathsf{ns}}(\theta) = (h(sk), \mathbf{U}, \mathbf{V})$).*

Proof. Follows directly from Theorem 6 and the h_{ns}-strong ISH of QALin, which in turn follows from Theorem 8. □

Extensions. We consider several extensions of QALin such as bilateral linear spaces [GHR15a], where the statement as well as the generating matrix have components in both groups. We also consider a sum argument [GHR15a] which is akin to a bilateral language but one shows that the sum of the discrete logs of two vectors in $\mathbb{G}_1$ and $\mathbb{G}_2$ belong to the image of the sum of two matrices in $\mathbb{G}_1, \mathbb{G}_2$. Finally, we extend local soundness to consider knowledge transfer arguments. The security of all this extensions is almost verbatim of theorems 7 and 8.

Quasi Argument for Bilateral Linear Knowledge Transfer. Let $\mathcal{M}, \mathcal{N}_1, \mathcal{N}_2$ be 3 witness samplable distribution over matrices in $\mathbb{G}_1^{d\times n}, \mathbb{G}_1^{d\times n}$ and $\mathbb{G}_2^{d\times n}$, respectively, for $n, d \in \mathbb{N}$. Let $\boldsymbol{K} \leq d$ where $\boldsymbol{K} = (K_1, K_2)$ and $\boldsymbol{S} \subseteq ([d], [d])$ where $S = S_1 \cup S_2$ and $\boldsymbol{S} \leq \boldsymbol{K}$ Let CS be an algebraic SSB commitment schemes with commitment space $\mathbb{G}_\mu^{\overline{K}}$, where $\mathbb{G}_\mu$ is defined by the input gk. The parameter language is

$$\begin{aligned}\mathcal{L}_{\mathsf{par}} = \big\{ &[\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2, [\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2 \mid \exists \mathbf{M}, \mathbf{N}_1, \mathbf{N}_2, \mathbf{G}, \mathbf{H}, \mathbf{F} \text{ s.t.} \\ &([\mathbf{M}]_1, \mathbf{M}), ([\mathbf{N}_1]_1, \mathbf{N}_2), ([\mathbf{N}_2]_2, \mathbf{N}_2) \in \mathsf{Sup}(\mathcal{M}, \mathcal{N}_1, \mathcal{N}_2), \\ &([\mathbf{G}]_1, \mathbf{G}, \mathbf{T}_{\mathbf{G}}) \in \mathsf{Sup}(\mathsf{CS.KeyGen}(gk_1, d, K_1, S_1)), \\ &([\mathbf{H}]_1, \mathbf{H}, \mathbf{T}_{\mathbf{H}}) \in \mathsf{Sup}(\mathsf{CS.KeyGen}(gk_1, d, K_2, S_2)), \\ &([\mathbf{F}]_2, \mathbf{F}, \mathbf{T}_{\mathbf{F}}) \in \mathsf{Sup}(\mathsf{CS.KeyGen}(gk_2, d, K_2, S_2))\big\}\end{aligned}$$

We assume w.l.o.g. that the corresponding relation is efficiently verifiable.The parameters $\rho = ([\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2, [\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2)$, define the following relations:

$$\mathcal{R}_\rho^{\mathsf{yes}} = \left\{ [\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, \boldsymbol{w} \;\middle|\; \begin{pmatrix} \boldsymbol{c} \\ \boldsymbol{d}_1 \\ \boldsymbol{d}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{GM} \\ \mathbf{HN_1} \\ \mathbf{FN_2} \end{pmatrix} \boldsymbol{w} \right\},$$

$$\mathcal{R}_{\rho,\boldsymbol{S}}^{\mathsf{no}} = \left\{ \begin{array}{c} ([\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2), \boldsymbol{w}, \\ ([\boldsymbol{x}]_1, [\boldsymbol{y}_1]_1, [\boldsymbol{y}_2]_2) \end{array} \;\middle|\; \begin{array}{l} \boldsymbol{x}, \boldsymbol{y_1}, \boldsymbol{y_2} \text{ are valid } S_1, S_2, S_2 \text{ openings of} \\ \boldsymbol{c},\ \boldsymbol{d}_1,\ \boldsymbol{d}_2 \text{ w.r.t. } \mathbf{G}, \mathbf{H}, \mathbf{F} \text{ respectively and} \\ \boldsymbol{x}_1 = \mathbf{M}_{S_1}\boldsymbol{w} \text{ but } \boldsymbol{y}_1 \neq \mathbf{N}_{1,S_2}\boldsymbol{w} \text{ or } \boldsymbol{y}_2 \neq \mathbf{N}_{2,S_2}\boldsymbol{w} \end{array} \right\},$$

that is the partial witness for $\boldsymbol{S}$ is some valid local openings $[\boldsymbol{x}]_1, [\boldsymbol{y}_1]_1, [\boldsymbol{y}_2]_2$ w.r.t. to $\mathbf{G}, \mathbf{H}, \mathbf{F}$ respectively that satisfy the following: if $\boldsymbol{x}_{S_2} = \mathbf{M}_{S_1}\boldsymbol{w}$ then it should be the case that both $\boldsymbol{y_1} = \mathbf{N}_{1,S_2}\boldsymbol{w}$ and $\boldsymbol{y_2} = \mathbf{N}_{2,S_2}\boldsymbol{w}$ where $\boldsymbol{w}$ is the promise of the adversary. Note that if S_1 is the empty set the latter relations trivially hold. We present the protocol in Fig. 5. Security is almost verbatim to the unilateral case. For completeness we give the full proof in Appendix B.1.

$\mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S} = (S_0, S_1))$**:**

- $([\mathbf{M}]_1, \mathbf{M}) \leftarrow \mathcal{M}$; $([\mathbf{N}_1]_1, \mathbf{N}_1) \leftarrow \mathcal{N}_1$; $([\mathbf{N}_2]_2, \mathbf{N}_2) \leftarrow \mathcal{N}_2$.
- $([\mathbf{G}]_1, \mathbf{G}, \mathbf{T_G}) \leftarrow \mathsf{CS.KeyGen}(gk_1, n, d, K_0, S_0)$;
  $([\mathbf{H}]_1, \mathbf{H}, \mathbf{T_H}) \leftarrow \mathsf{CS.KeyGen}(gk_1, n, d, K_1, S_1)$;
  $([\mathbf{F}]_2, \mathbf{F}, \mathbf{T_F}) \leftarrow \mathsf{CS.KeyGen}(gk_2, n, d, K_2, S_1)$;
- Output (ρ, θ) where
$$\rho = (gk, [\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2, [\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2),$$
$$\theta = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F}, \mathbf{M}, \mathbf{N}_1, \mathbf{N}_2).$$

$\mathsf{K}(\rho, \theta)$**:**

- Parse $\rho = (gk, [\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2, [\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2)$,
  $\theta = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F}, \mathbf{M}, \mathbf{N}_1, \mathbf{N}_2)$.
- Sample $\mathbf{K}_0 \leftarrow \mathbb{Z}_p^{\overline{K}_0 \times k}$; $\mathbf{K}_1 \leftarrow \mathbb{Z}_p^{\overline{K}_1 \times k}$; $\mathbf{K}_2 \leftarrow \mathbb{Z}_p^{\overline{K}_2 \times k}$; $\mathbf{A} \leftarrow \mathcal{D}_k$ and redefine $\mathbf{A}$ as its first k columns.
- Compute $[\mathbf{B}]_1 = [\mathbf{M}^\top]_1\mathbf{G}^\top\mathbf{K}_0 + [\mathbf{N}_1^\top]_1\mathbf{H}^\top\mathbf{K}_1$ and $[\mathbf{D}]_2 = [\mathbf{N}_2^\top]_2\mathbf{F}^\top\mathbf{K}_2$.
- $\mathbf{C}_1 = \begin{pmatrix} \mathbf{K}_0 \\ \mathbf{K}_1 \end{pmatrix} \mathbf{A}$ and $\mathbf{C}_2 = \mathbf{K}_2\mathbf{A}$;
- Output (crs, τ) where $\mathsf{crs} = ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1)$ and $\tau = (\mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F})$.

$\mathsf{Prove}(\mathsf{crs} = ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1), [\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, \boldsymbol{w})$**:**

- Output $([\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2) \leftarrow (\boldsymbol{w}^\top[\mathbf{B}]_1, \boldsymbol{w}^\top[\mathbf{D}]_2)$.

$\mathsf{Verify}\,(\mathsf{crs}, [\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, [\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2)$**:**

- Output 1 iff $e([\boldsymbol{\pi}]_1, [\mathbf{A}]_2) + e([\boldsymbol{\theta}]_2, [\mathbf{A}]_1) = e([\boldsymbol{c}^\top \mid \boldsymbol{d}_1^\top]_1, [\mathbf{C}_1]_2) + e([\boldsymbol{d}_2^\top]_2, [\mathbf{C}_2]_1)$.

$\mathsf{Extract}\,(\tau, [\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, [\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2)$**:**

- Parse τ as $(\mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F})$ and output $[\boldsymbol{x}]_1 = \mathbf{T_G}^\top[\boldsymbol{c}]_1, [\boldsymbol{y}_1]_1 = \mathbf{T_H}^\top[\boldsymbol{d}_1]_1, [\boldsymbol{y}_2]_2 = \mathbf{T_F}^\top[\boldsymbol{d}_2]_2$.

Figure 5: Quasi argument QABlin for knowledge transfer of membership in linear space.

Quasi Argument for Sum Knowledge Transfer. Let $(\mathcal{M}_1, \mathcal{M}_1)$ be some (possibly correlated) witness samplable distributions outputting matrices in $\mathbb{G}_1^{d\times n} \times \mathbb{G}_2^{d\times n}$ and $\mathcal{N}$ be witness samplable distributions outputting matrices in $\mathbb{G}_1^{d\times n}$ for $n, d \in \mathbb{N}$. Let $\boldsymbol{K} \leq d$ where $\boldsymbol{K} = (K_0, K_1)$ and $\boldsymbol{S} \subseteq ([d], [d])$ where $S = S_1 \cup S_2$ and $\boldsymbol{S} \leq \boldsymbol{K}$. Let CS be an algebraic SSB commitment scheme and CS' be a split algebraic commitment key with commitment space $\mathbb{G}_1^{\overline{K}}$, $\mathbb{G}_1^{\overline{K}} \times \mathbb{G}_2^{\overline{K}}$ respectively. The parameter language is

$$\begin{aligned}\mathcal{L}_{\mathsf{par}} = \big\{ &[\mathbf{M}_1]_1, [\mathbf{M}_2]_2, [\mathbf{N}]_1, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2 \mid \exists \mathbf{M}_1, \mathbf{M}_2, \mathbf{N}_1, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F} \text{ s.t.} \\ &([\mathbf{M}_1]_1, [\mathbf{M}_2]_2, \mathbf{M}_1, \mathbf{M}_2) \in \mathsf{Sup}(\mathcal{M}_1, \mathcal{M}_2), ([\mathbf{N}]_1, \mathbf{N}) \in \mathsf{Sup}(\mathcal{N}), \\ &([\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{T}_{\mathbf{Q}}) \in \mathsf{Sup}(\mathsf{CS}'.\mathsf{KeyGen}(gk, n, K_0, S_1)), \\ &([\mathbf{F}]_1, \mathbf{F}, \mathbf{T}_{\mathbf{F}}) \in \mathsf{Sup}(\mathsf{CS}.\mathsf{KeyGen}(gk_1, n, K_1, S_2))\big\}\end{aligned}$$

We assume w.l.o.g. that the corresponding relation is efficiently verifiable. The parameters $\rho = ([\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2)$ define the following relations[14]

$$\mathcal{R}_\rho^{\mathsf{yes}} = \left\{ \; [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_2, \boldsymbol{w} \;\middle|\; \begin{pmatrix} \boldsymbol{c}_1 + \boldsymbol{c}_2 \\ \boldsymbol{d} \end{pmatrix} = \begin{pmatrix} (\mathbf{Q}_1 + \mathbf{Q}_2)(\mathbf{M}_1 + \mathbf{M}_2) \\ \mathbf{F}\mathbf{N} \end{pmatrix} \boldsymbol{w} \; \right\},$$

$$\mathcal{R}_{\rho,\boldsymbol{S}}^{\mathsf{no}} = \left\{ \begin{array}{c} ([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1), \boldsymbol{w}, \\ ([\boldsymbol{x}_1]_1, [\boldsymbol{x}_2]_2, [\boldsymbol{y}]_1) \end{array} \;\middle|\; \begin{array}{l} \boldsymbol{x}_1 + \boldsymbol{x}_2, \boldsymbol{y} \text{ are valid } S_0, S_1 \text{ openings of} \\ \boldsymbol{c}_1 + \boldsymbol{c}_2, \boldsymbol{d}_2 \text{ w.r.t. } \mathbf{Q}_1 + \mathbf{Q}_2, \mathbf{F} \text{ respectively and} \\ \boldsymbol{x}_1 + \boldsymbol{x}_2 = (\mathbf{M}_{1,S_0} + \mathbf{M}_{2,S_0})\boldsymbol{w} \text{ but } \boldsymbol{y} \neq \mathbf{N}_{S_2}\boldsymbol{w} \end{array} \right\},$$

that is the partial witness for $\boldsymbol{S}$ is some valid local openings $[\boldsymbol{x}_1]_1, [\boldsymbol{x}_2]_2, [\boldsymbol{y}]_1$ w.r.t. to $\mathbf{G}, \mathbf{H}, \mathbf{F}$ respectively that satisfy the following: if $\boldsymbol{x}_1 + \boldsymbol{x}_2 = (\mathbf{M}_{1,S_1} + \mathbf{M}_{2,S_1})\boldsymbol{w}$ then it should be the case that $\boldsymbol{y} = \mathbf{N}_{S_2}\boldsymbol{w}$ where $\boldsymbol{w}$ is the promise of the adversary. Note that if S_1 is the empty set the latter relations trivially hold. We present the protocol in Fig 6.

For completeness we give the full proof in Appendix B.2.

5.2.2 Quasi-Arguments for Hadamard Products.

The main result of [GHR15a] was implicitly a quasi-argument for the set of equations $b_i(b_i - 1) = 0$, for all $i \in [d]$. We extend their results to equations of the form $x_i y_i = z_i$, that is $\boldsymbol{x} \circ \boldsymbol{y} = \boldsymbol{z}$ where $\circ$ denotes the hadamard product. Let $\mathcal{U}, \mathcal{V}, \mathcal{W}$ be witness samplable distributions over matrices in $\mathbb{G}_1^{d\times n}, \mathbb{G}_2^{d\times n}$ and $\mathbb{G}_1^{d\times n}$, respectively, for $n, d \in \mathbb{N}$. Let $\boldsymbol{K} = (K, K)$ with $K \leq d$ and $\boldsymbol{S} = (S, S)$ with $S \subseteq [d]$ and $S \leq K$. Also let CS be an algebraic SSB commitment scheme with commitment space $\mathbb{G}_\mu^{\overline{K}}$. The parameter language is

$$\begin{aligned}\mathcal{L}_{\mathsf{par}} = \big\{ &[\mathbf{U}]_1, [\mathbf{V}]_2, [\mathbf{W}]_1, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}]_1 \mid \exists \mathbf{U}, \mathbf{V}, \mathbf{W}, \mathbf{G}, \mathbf{H}, \mathbf{F} \text{ s.t.} \\ &([\mathbf{U}]_1, \mathbf{U}) \in \mathsf{Sup}(\mathcal{U}), ([\mathbf{V}]_2, \mathbf{V}) \in \mathsf{Sup}(\mathcal{V}), ([\mathbf{W}]_1, \mathbf{W}) \in \mathsf{Sup}(\mathcal{W}), \\ &([\mathbf{G}]_1, \mathbf{G}, \mathbf{T}_{\mathbf{G}}) \in \mathsf{Sup}(\mathsf{CS}.\mathsf{KeyGen}(gk_1, n, K, S)) \\ &([\mathbf{H}]_2, \mathbf{H}, \mathbf{T}_{\mathbf{H}}) \in \mathsf{Sup}(\mathsf{CS}.\mathsf{KeyGen}(gk_2, n, K, S)) \\ &([\mathbf{F}]_1, \mathbf{F}, \mathbf{T}_{\mathbf{F}}) \in \mathsf{Sup}(\mathsf{CS}.\mathsf{KeyGen}(gk_1, n, K, S))\big\}\end{aligned}$$

We assume w.l.o.g. that the corresponding relation is efficiently verifiable. The parameters $\rho = ([\mathbf{U}]_1, [\mathbf{V}]_2, [\mathbf{W}]_1, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}]_1)$define the following relations:

$$\mathcal{R}_\rho^{\mathsf{yes}} = \left\{ \; [\boldsymbol{u}]_1, [\boldsymbol{v}]_2, [\boldsymbol{w}]_2, \boldsymbol{a}, \boldsymbol{b} \;\middle|\; \begin{array}{l} \boldsymbol{u} = \mathbf{G}\mathbf{U}\boldsymbol{a}, \boldsymbol{v} = \mathbf{H}\mathbf{V}\boldsymbol{b} \\ \boldsymbol{w} = \mathbf{F}\mathbf{W}(\boldsymbol{a} \circ \boldsymbol{b}) \end{array} \; \right\},$$

$$\mathcal{R}_{\rho,S}^{\mathsf{no}} = \left\{ \begin{array}{c} ([\boldsymbol{v}]_1, [\boldsymbol{u}]_2, [\boldsymbol{w}]_1), \boldsymbol{a}, \boldsymbol{b} \\ ([\boldsymbol{x}_1]_1, [\boldsymbol{x}_2]_2, [\boldsymbol{y}]_1) \end{array} \;\middle|\; \begin{array}{l} \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{y} \text{ are valid } S \text{ openings of} \\ \boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{d} \text{ w.r.t. } \mathbf{G}, \mathbf{H}, \mathbf{F} \text{ respectively and} \\ \boldsymbol{x}_1 = \mathbf{U}_S\boldsymbol{a}, \boldsymbol{x}_2 = \mathbf{V}_S\boldsymbol{b}, \text{ but } \boldsymbol{y} \neq \mathbf{W}_S(\boldsymbol{a} \circ \boldsymbol{b}) \end{array} \right\}.$$

[14] We allow both the distributions $\mathcal{M}_1, \mathcal{M}_2, \mathcal{N}$ and the commitment keys to include some auxiliary information with its associated witness which are included in ρ, θ respectively. This auxiliary information is not used in the protocol, but is public when the protocol is used inside other protocol. We omit it here to simplify the presentation but we consider it whenever needed.

$\mathcal{D}_{\mathsf{par}}(gk, d, \boldsymbol{K}, \boldsymbol{S} = (S_0, S_1))$**:**

- $([\mathbf{M}_1]_1, [\mathbf{M}_2]_2, \mathbf{M}_1, \mathbf{M}_2) \leftarrow (\mathcal{M}_1, \mathcal{M}_2)$; $([\mathbf{N}]_1, \mathbf{N}) \leftarrow \mathcal{N}$;
- $([\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{T}_\mathbf{Q}) \leftarrow \mathsf{CS}'.\mathsf{KeyGen}(gk, n, d, K_0, S_0)$;
  $([\mathbf{F}]_1, \mathbf{F}, \mathbf{T}_\mathbf{F}) \leftarrow \mathsf{CS}.\mathsf{KeyGen}(gk_1, n, d, K_1, S_1)$;
- Output (ρ, θ) where
$$\rho = (gk, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2, [\mathbf{M}_1]_1, [\mathbf{M}_2]_2, [\mathbf{N}]_1),$$
$$\theta = (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}, \mathbf{T}_\mathbf{Q}, \mathbf{T}_\mathbf{F}, \mathbf{M}_1, \mathbf{M}_2, \mathbf{N}).$$

$\mathsf{K}(\rho, \theta)$**:**

- Parse $\rho = (gk, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2, [\mathbf{M}_1]_1, [\mathbf{M}_2]_2, [\mathbf{N}]_1)$, $\theta = (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}, \mathbf{T}_\mathbf{Q}, \mathbf{T}_\mathbf{F}, \mathbf{M}_1, \mathbf{M}_2, \mathbf{N})$.
- Set $\mathbf{Q} = \mathbf{Q}_1 + \mathbf{Q}_2$ and sample $\mathbf{K}_0 \leftarrow \mathbb{Z}_p^{\overline{K}_0 \times k}$; $\mathbf{K}_1 \leftarrow \mathbb{Z}_p^{\overline{K}_1 \times k}$; $\mathbf{Z} \leftarrow \mathbb{Z}_p^{n \times k}$; $\mathbf{A} \leftarrow \mathcal{D}_k$ and redefine $\mathbf{A}$ as its first k columns.
- Compute $[\mathbf{B}]_1 = [\mathbf{M}_1^\top]_1 \mathbf{Q}^\top \mathbf{K}_0 + [\mathbf{N}^\top]_1 \mathbf{F}^\top \mathbf{K}_1 + [\mathbf{Z}]_1$ and $[\mathbf{D}]_2 = [\mathbf{M}_2^\top]_2 \mathbf{Q}^\top \mathbf{K}_0 - [\mathbf{Z}]_2$.
- $\mathbf{C}_1 = \begin{pmatrix} \mathbf{K}_0 \\ \mathbf{K}_1 \end{pmatrix} \mathbf{A}$ and $\mathbf{C}_2 = \mathbf{K}_0 \mathbf{A}$;
- Output (crs, τ) where $\mathsf{crs} = ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1)$ and $\tau = (\mathbf{T}_\mathbf{Q}, \mathbf{T}_\mathbf{F})$.

$\mathsf{Prove}(\mathsf{crs} = ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1), [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1, \boldsymbol{w})$**:**

Sample $\boldsymbol{z} \leftarrow \mathbb{Z}_p^k$ and output $([\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2) \leftarrow (\boldsymbol{w}^\top [\mathbf{B}]_1 - [\boldsymbol{z}^\top]_1, \boldsymbol{w}^\top [\mathbf{D}]_2 + [\boldsymbol{z}^\top]_2)$.

$\mathsf{Verify}\,(\mathsf{crs}, [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1, [\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2)$**:**

- Output 1 iff $e([\boldsymbol{\pi}]_1, [\mathbf{A}]_2) + e([\boldsymbol{\theta}]_2, [\mathbf{A}]_1) = e([\boldsymbol{c}_1^\top \mid \boldsymbol{d}^\top]_1, [\mathbf{C}_1]_2) + e([\boldsymbol{c}_2^\top]_2, [\mathbf{C}_2]_1)$.

$\mathsf{Extract}\,(\tau, [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1, [\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2)$**:**

- Parse τ as $(\mathbf{T}_\mathbf{Q}, \mathbf{T}_\mathbf{F})$ and output $[\boldsymbol{x}_1]_1 = \mathbf{T}_\mathbf{Q}{}^\top [\boldsymbol{c}_1]_1, [\boldsymbol{x}_2]_2 = \mathbf{T}_\mathbf{Q}{}^\top [\boldsymbol{c}_2]_1, [\boldsymbol{y}]_1 = \mathbf{T}_\mathbf{F}{}^\top [\boldsymbol{d}]_1$.

Figure 6: Quasi argument QASum for knowledge transfer of sum membership in linear space.

That is the partial witness for S is some valid local openings $[\boldsymbol{x}_1]_1, [\boldsymbol{x}_2]_2, [\boldsymbol{y}]_1$ w.r.t. to $\mathbf{G}, \mathbf{H}, \mathbf{F}$ respectively that satisfy the following: if $\boldsymbol{x}_1 = \mathbf{U}_S \boldsymbol{a}$ and $\boldsymbol{x}_2 = \mathbf{V}_S \boldsymbol{b}$ and then it should be the case that $\boldsymbol{y} = \mathbf{W}_S \boldsymbol{c}$ where $\boldsymbol{c} = \boldsymbol{a} \circ \boldsymbol{b}$. Here $\boldsymbol{a}, \boldsymbol{b}$ is the promise of the adversary. We present the protocol in Fig 7. Essentially, we first have the prover commit to the kronecker product $\boldsymbol{a} \otimes \boldsymbol{b}$ using a commitment scheme defined by the $\otimes$ operation of CS to itself, and then show that if the split opening of this commitment is $\boldsymbol{w} = \boldsymbol{a} \otimes \boldsymbol{b}$, then the opening of $\boldsymbol{d}$ is $\mathbf{D}\boldsymbol{w}$ where $\mathbf{D}$ is the linear operation that outputs $\boldsymbol{a} \circ \boldsymbol{b}$ on input $\boldsymbol{a} \otimes \boldsymbol{b}$. The former "promise", regarding the kronecker product, is verified by the pairing operation, while for the latter construction QASum is used.

$\mathcal{D}_{\mathsf{par}}(gk, d, K, S)$**:**

- $([\mathbf{U}]_1, \mathbf{U}) \leftarrow \mathcal{U}$; $([\mathbf{V}]_2, \mathbf{V}) \leftarrow \mathcal{V}$. $([\mathbf{W}]_1, \mathbf{W}) \leftarrow \mathcal{W}$;
- $([\mathbf{G}]_1, \mathbf{G}, \mathbf{T_G}) \leftarrow \mathsf{CS.KeyGen}(gk_1, n, d, K, S)$;
  $([\mathbf{H}]_2, \mathbf{H}, \mathbf{T_H}) \leftarrow \mathsf{CS.KeyGen}(gk_2, n, d, K, S)$;
  $([\mathbf{F}]_1, \mathbf{F}, \mathbf{T_F}) \leftarrow \mathsf{CS.KeyGen}(gk_1, n, d, K, S)$;
- Output (ρ, θ) where $\rho := (gk, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}]_1, [\mathbf{U}]_1, [\mathbf{V}]_2, [\mathbf{W}]_1)$, and $\theta := (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F}, \mathbf{U}, \mathbf{V}, \mathbf{W})$.

$\mathsf{K}(\rho, \theta)$**:**

- Parse $\rho = (gk, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}]_1, [\mathbf{U}]_1, [\mathbf{V}]_2, [\mathbf{W}]_1)$, $\theta = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F}, \mathbf{U}, \mathbf{V}, \mathbf{W})$.
- $(ck, sk) \leftarrow \mathsf{kCS.KeyGen}(gk, [\mathbf{G}]_1, [\mathbf{H}]_2, \mathbf{G}, \mathbf{H})$ and parse ck as $[\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathsf{aux}$ and sk as $\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{T_Q}$.
- Sample $\mathbf{R} \in \mathbb{Z}_q^{d^2 \times n^2}$ and set $\mathbf{M}_1 = \mathbf{U} \otimes \mathbf{V} - \mathbf{R}$ and $\mathbf{M}_2 = \mathbf{R}$. Set $\mathbf{N} = \mathbf{WD}$.
- Set $\rho_{\text{sum}} := (gk, [\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2, [\mathbf{M}_1]_1, [\mathbf{M}_2]_2, [\mathbf{N}]_1)$,
  $\theta_{\text{sum}} := (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}, \mathbf{T_Q}, \mathbf{T_F}, \mathbf{M}_1, \mathbf{M}_2, \mathbf{N})$.
- Set $(\mathsf{crs}_{\text{sum}}, \tau_{\text{sum}}) \leftarrow \mathsf{QASum}(\rho_{\text{sum}}, \theta_{\text{sum}})$.
- Sample $\mathbf{R}' \leftarrow \mathbb{Z}_p^{\overline{K}^2 \times n^2}$ and set $[\mathbf{E}_1]_1 = [\mathbf{Q}_1(\mathbf{U} \otimes \mathbf{V}) - \mathbf{R}')]_1$, $[\mathbf{E}_2]_2 = [\mathbf{Q}_2(\mathbf{U} \otimes \mathbf{V}) + \mathbf{R}']_2$.
- Output $\mathsf{crs} = ([\mathbf{E}_1]_1, [\mathbf{E}_2]_2, \mathsf{crs}_{\text{sum}})$, $\tau = (\mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F})$.

$\mathsf{Prove}(\mathsf{crs}, [\boldsymbol{x}]_1, [\boldsymbol{y}]_2, [\boldsymbol{w}]_1, \boldsymbol{a}, \boldsymbol{b})$**:**

- Parse $\mathsf{crs} = ([\mathbf{E}_1]_1, [\mathbf{E}_2]_2, \mathsf{crs}_{\text{sum}})$.
- Set $[\boldsymbol{c}_1]_1 = [\mathbf{E}_1]_1(\boldsymbol{a} \otimes \boldsymbol{b}), [\boldsymbol{c}_2]_2 = [\mathbf{E}_2]_2(\boldsymbol{a} \otimes \boldsymbol{b}), [\boldsymbol{d}]_1 = [\boldsymbol{w}]_1$.
- $\pi_{\text{sum}} = \mathsf{QASum.Prove}(\mathsf{crs}_{\text{sum}}, [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_1, [\boldsymbol{d}]_1, \boldsymbol{a} \otimes \boldsymbol{b})$.
- Output $\pi := ([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_1, \pi_{\mathsf{sum}})$.

$\mathsf{Verify}\,(\mathsf{crs}, [\boldsymbol{u}]_1, [\boldsymbol{v}]_2, [\boldsymbol{w}]_1, \pi)$**:**

- Parse $\mathsf{crs} = ([\mathbf{E}_1]_1, [\mathbf{E}_2]_2, \mathsf{crs}_{\text{sum}})$, $\pi := ([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_1, \pi_{\mathsf{sum}})$.
- Compute $[\boldsymbol{u} \otimes \boldsymbol{v}]_T$ using the pairing operation and output 1 iff
  1. $\mathsf{QASum.Verify}\,(\mathsf{crs}_{\mathsf{sum}}, [\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{w}]_1) = 1$ and
  2. $[\boldsymbol{u} \otimes \boldsymbol{v}]_T = e([\boldsymbol{c}_1]_1, [1]_2) + e([1]_1, [\boldsymbol{c}_2]_2)$

$\mathsf{Extract}\,(\tau, [\boldsymbol{u}]_1, [\boldsymbol{v}]_2, [\boldsymbol{w}]_1, \pi)$**:** Parse τ as $(\mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F})$ and output $[\boldsymbol{x}_1]_1 := \mathbf{T_G}^\top[\boldsymbol{u}]_1, [\boldsymbol{x}_2]_2 := \mathbf{T_H}^\top[\boldsymbol{v}]_1, [\boldsymbol{y}]_1 := \mathbf{T_F}^\top[\boldsymbol{w}]_1$.

Figure 7: Quasi argument QAHad for knowledge transfer of hadammard product. Here $\mathbf{D} \in \mathbb{Z}_q^{n \times n^2}$ is the matrix such that $\mathbf{D}(\boldsymbol{a} \otimes \boldsymbol{b}) = \boldsymbol{a} \circ \boldsymbol{b}$

Theorem 9. *Let $\mathcal{U}, \mathcal{V}, \mathcal{W}$ be witness samplable distributions, $\mathcal{D}_k$ be a matrix distribution and* CS *an algebraic SSB commitment scheme with perfect completeness. Also, let $\mathcal{A}$ be an adversary against h_{ls}-strong local knowledge soundness of QAHad where $h_{ls}(\theta) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{W}, [\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2)$ for a uniformly distributed $\mathbf{R}$. Then completeness holds with probability 1 and for h_{ls}-strong local soundness it holds that $\mathsf{Adv}_{snd}^{QAHad}(\mathcal{A}) \leq \mathsf{Adv}_{snd}^{QASum}(\mathcal{B})$ where $\mathcal{B}$ is an adversary against $h_{ls\text{-}sum}$-strong local soundness of QASum for ρ_{sum} as computed in Fig. 7 and $h_{ls\text{-}sum}(\theta_{sum})$ outputs θ_{sum} except the matrices $\mathbf{M}_1, \mathbf{M}_2$.*

Proof. For completeness, we have that

$$\begin{aligned}\boldsymbol{u}\otimes\boldsymbol{v} &= \mathbf{G}\mathbf{U}\boldsymbol{a}\otimes\mathbf{G}\mathbf{U}\boldsymbol{b} = (\mathbf{G}\otimes\mathbf{H})(\mathbf{U}\otimes\mathbf{V})(\boldsymbol{a}\otimes\boldsymbol{b}) = \\ &= (\mathbf{G}\otimes\mathbf{H}-\mathbf{Z}+\mathbf{Z})(\mathbf{U}\otimes\mathbf{V}-\mathbf{R}+\mathbf{R})(\boldsymbol{a}\otimes\boldsymbol{b}) = \\ &= (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{M}_1+\mathbf{M}_2)(\boldsymbol{a}\otimes\boldsymbol{b})\end{aligned}$$

and also $\boldsymbol{c}_1+\boldsymbol{c}_2 = (\mathbf{E}_1+\mathbf{E}_2)(\boldsymbol{a}\otimes\boldsymbol{b}) = (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{U}\otimes\mathbf{V})(\boldsymbol{a}\otimes\boldsymbol{b}) = \boldsymbol{u}\otimes\boldsymbol{v}$, so the pairing test is successful. Finally, noting that $\boldsymbol{w} = \boldsymbol{d} = \mathbf{F}\mathbf{W}(\boldsymbol{a}\circ\boldsymbol{b}) = \mathbf{F}\mathbf{W}\mathbf{D}(\boldsymbol{a}\otimes\boldsymbol{b}) = \mathbf{F}\mathbf{N}(\boldsymbol{a}\otimes\boldsymbol{b})$, we see that the statement/witness pair $([\boldsymbol{c}_1]_1,[\boldsymbol{c}_2]_2,[\boldsymbol{d}]_1),\boldsymbol{a}\otimes\boldsymbol{b}$ is a yes instance of the sum language for parameters ρ_{sum} and the second condition for verification follows by the completeness of the QASum.

For local knowledge soundness, it is enough to note that the Kronecker part of the knowledge transfer holds unconditionally, that is, if for some promise $\boldsymbol{a},\boldsymbol{b}$ it holds that $\boldsymbol{u} = \mathbf{G}\mathbf{U}\boldsymbol{a}$ and $\boldsymbol{v} = \mathbf{H}\mathbf{V}\boldsymbol{b}$, then by the verification of the pairing condition, $\boldsymbol{c}_1+\boldsymbol{c}_2 = (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{M}_1+\mathbf{M}_2)(\boldsymbol{a}\otimes\boldsymbol{b})$, so we efficiently construct a promise for the sum language. Also, the value $h_{\text{ls-sum}}(\theta_{\mathsf{sum}})$ can be computed given $h_{ls}(\theta)$. Now, an accepting proof for the hadamard language contains an accepting proof for the sum language and we use that to break q-strong local soundness of QASum. Details follow.

Let $\mathcal{A}$ be an adversary against h_{ls}-strong local knowledge soundness of QAHad. We construct an adversary $\mathcal{B}$ against $h_{\text{ls-sum}}$-strong local knowledge soundness of QASum. $\mathcal{B}$ takes as input $(\rho_{\text{sum}}, h_{\text{ls-sum}}(\theta_{\text{sum}}), \mathsf{crs}_{\text{sum}})$ and works as follows:

- Parse

$$\rho_{\text{sum}} = (gk,[\mathbf{Q}_1]_1,[\mathbf{Q}_2]_2,[\mathbf{F}]_1,[\mathbf{M}_1]_1,[\mathbf{M}_2]_2,[\mathbf{N}]_1,\mathsf{aux}_{\mathsf{CS}} = (\mathbf{G},\mathbf{H}),\mathsf{aux}_{\mathcal{M}} = ([\mathbf{U}]_1,[\mathbf{V}]_2),$$

$$\theta_{q_{\text{sum}}} = (\mathbf{Q}_1,\mathbf{Q}_2,\mathbf{G},\mathbf{H},\mathbf{F},\mathbf{N})$$

- Set $\rho = (gk,[\mathbf{G}]_1,[\mathbf{H}]_2,[\mathbf{F}]_2,[\mathbf{U}]_1,[\mathbf{V}]_2,[\mathbf{N}]_1)$, $h_{ls}(\theta) = (\mathbf{G},\mathbf{H},\mathbf{F},\mathbf{N},[\mathbf{M}_1]_1,[\mathbf{M}_2]_2)$.
- It samples $\mathbf{R}'\leftarrow\mathbb{Z}_p^{\overline{K}}\times n^2$ and sets $[\mathbf{E}_1]_1 = (\mathbf{Q}_1+\mathbf{Q}_2)[\mathbf{M}_1]_1+[\mathbf{R}']_1, [\mathbf{E}_2]_2 = (\mathbf{Q}_1+\mathbf{Q}_2)[\mathbf{M}_2]_2-[\mathbf{R}']_2$.

It then executes $\mathcal{A}(\rho,h_{ls}(\theta),\mathsf{crs} = ([\mathbf{E}_1]_1,[\mathbf{E}_2]_2,\mathsf{crs}_{\text{sum}}))$ until it outputs a statement $([\boldsymbol{u}]_1,[\boldsymbol{v}]_2,[\boldsymbol{w}]_1,\boldsymbol{a},\boldsymbol{b})$ together with an accepting proof $([\boldsymbol{c}_1]_1,[\boldsymbol{c}_2]_2,\pi_{\text{sum}})$. It outputs the statement/advice/proof tuple

$$(([\boldsymbol{c}_1]_1,[\boldsymbol{c}_2]_2,[\boldsymbol{w}]_1),\boldsymbol{a}\otimes\boldsymbol{b},\pi_{\text{sum}}).$$

The crs is identically distributed to an honestly computed one. Indeed the only thing computed differently are the values $[\mathbf{E}_1]_1,[\mathbf{E}_2]_2$, but note that in the reduction they are distributed uniformly conditioned on $\mathbf{E}_1+\mathbf{E}_2 = (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{M}_1+\mathbf{M}_2) = (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{U}\otimes\mathbf{V})$, as in the honest crs generation.

Now, assuming an accepting proof, and a correct promise $\boldsymbol{a},\boldsymbol{b}$ given from $\mathcal{A}$ means that the promise of $\mathcal{B}$ is also correct. Indeed, we have

$$\begin{aligned}\boldsymbol{c}_1+\boldsymbol{c}_2 = \boldsymbol{u}\otimes\boldsymbol{v} &= \mathbf{G}\mathbf{U}\boldsymbol{a}\otimes\mathbf{H}\mathbf{V}\boldsymbol{b} = (\mathbf{G}\otimes\mathbf{H})(\mathbf{U}\otimes\mathbf{V})(\boldsymbol{a}\otimes\boldsymbol{b}) = \\ &= (\mathbf{G}\otimes\mathbf{H}-\mathbf{Z}+\mathbf{Z})(\mathbf{U}\otimes\mathbf{V}-\mathbf{R}+\mathbf{R})(\boldsymbol{a}\otimes\boldsymbol{b}) = \\ &= (\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{M}_1+\mathbf{M}_2)(\boldsymbol{a}\otimes\boldsymbol{b}).\end{aligned}$$

Now let $\boldsymbol{x}_1 = \mathbf{T}_{\mathbf{Q}}\boldsymbol{c}_1$, $\boldsymbol{x}_2 = \mathbf{T}_{\mathbf{Q}}\boldsymbol{c}_2$, $\boldsymbol{y} = \mathbf{T}_{\mathbf{F}}\boldsymbol{w}$ be the extracted values. We have that

$$\begin{aligned}\boldsymbol{x}_1+\boldsymbol{x}_2 &= \mathbf{T}_{\mathbf{Q}}(\boldsymbol{c}_1+\boldsymbol{c}_2) = \mathbf{T}_{\mathbf{Q}}(\mathbf{Q}_1+\mathbf{Q}_2)(\mathbf{M}_1+\mathbf{M}_2)(\boldsymbol{a}\otimes\boldsymbol{b}) \\ &= (\mathbf{M}_{S,1}+\mathbf{M}_{S,2})(\boldsymbol{a}\otimes\boldsymbol{b}).\end{aligned}$$

so indeed the promise is correct. Also assuming that the statement/advice given from $\mathcal{A}$ is a no-instance for the hadamard language w.r.t. to the set S, then the statement/advice given from $\mathcal{B}$ is a no-instance for the sum language w.r.t. the same set S. Indeed, we have

$$\boldsymbol{y}\neq\mathbf{W}_{\boldsymbol{S}}(\boldsymbol{a}\circ\boldsymbol{b}) = \mathbf{W}_{\boldsymbol{S}}\mathbf{D}(\boldsymbol{a}\otimes\boldsymbol{b}) = \mathbf{N}_{\boldsymbol{S}}(\boldsymbol{a}\otimes\boldsymbol{b}).$$

So, conditioned on a successful $\mathcal{A}$, $\mathcal{B}$ outputs an instance/advice such that (1) the extractor gets values that satisfy $\mathcal{R}^{\mathsf{no}}_{\rho_{\mathrm{sum}},\boldsymbol{S}}$ for ρ_{sum} and (2) a proof that verifies w.r.t. the instance.

□

We next show that when the distributions $\mathcal{U}, \mathcal{V}, \mathcal{W}$ guarantee that the sum knowledge transfer argument is secure w.r.t. all possible sets $\boldsymbol{S}$, construction QAHad has h_{ls}-strong local knowledge soundness where h_{ls} includes $\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{W}, [\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2$ for a uniform $\mathbf{R}$.

Corollary 4. *Let $\mathcal{D}_k$ be a matrix distribution for which $\mathcal{D}_k$-$\mathsf{SKerMDH}$ and let DDH hold in $\mathbb{G}_1, \mathbb{G}_2$. Denote $\mathcal{U}_S$ (resp. $\mathcal{V}_S$, $\mathcal{W}_S$) the distributions that sample matrices from $\mathcal{U}$ (res. $\mathcal{V}_2$, $\mathcal{W}$), and restricts them to rows corresponding to S. Then*

1. *If for all $S \subseteq [d]$ with $S \leq K_0$, $\mathcal{U}_S^\top$-MDDH and $\mathcal{V}_S^\top$-MDDH hold, QAHad is an h_{ls}-strong local knowledge sound proof system, where $h_{ls}(\theta) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N}, [\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2)$ for a uniform $\mathbf{R}$.*
2. *If for all $S, S \subseteq [d]$ with $S \leq K$ the distributions $\mathcal{U}_S, \mathcal{V}_S, \mathcal{W}_S$ output matrices with the last n' columns being $\mathbf{0}$, and ${\mathcal{U}'}_S^\top$-MDDH and ${\mathcal{V}'}_S^\top$-MDDH hold, with ${\mathcal{U}'}_S$, (resp. ${\mathcal{V}'}_S$) being $\mathcal{U}_S$ (resp. $\mathcal{V}_S$) where we delete the trailing zero columns, then QAHad is an h_{ls}-strong local knowledge sound proof system, where $h_{ls}(\theta) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{W}, [\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2)$.*

Proof. By Thm. 9 it is enough to show that QASum is secure for such distribution. This in turn hold when the sum knowledge transfer argument is sound (Thm. 17) which is true if $\mathcal{D}_k$-$\mathsf{SKerMDH}$ holds and $(\mathcal{U}_S, \mathcal{V}_S, h) - \mathsf{MDDH}$ assumption holds (similar in the second case for the distributions we remove the zeros) by Thm. 14. It remains to show that for these distribution the latter condition holds when we are given the extra information $h(\mathbf{U}, \mathbf{V}) = ([\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2)$ for a uniform $\mathbf{R}$. We show that this is the case if, additionally, DDH hold. That is we need to show that for all S the $(\mathcal{U}_S, \mathcal{V}_S, h)$-$\mathsf{MDDH}$ holds or equivalently the distributions

- $[\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top \otimes \mathbf{V}^\top - \mathbf{R}]_1, [\mathbf{R}]_2, [(\mathbf{U} \otimes \mathbf{V})^\top \boldsymbol{k} - \boldsymbol{r}]_1, [\boldsymbol{r}]_2 : \boldsymbol{k} \leftarrow \mathbb{Z}_q^{|S|^2}; \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}; k \leftarrow \mathbb{Z}_q$
- $[\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top \otimes \mathbf{V}^\top - \mathbf{R}]_1, [\mathbf{R}]_2, [\boldsymbol{s}]_1, [\boldsymbol{t}]_2 : \boldsymbol{s}, \boldsymbol{t} \leftarrow \mathbb{Z}_q^{n^2}$

where $\mathbf{U} \leftarrow \mathcal{U}_S; \mathbf{V} \leftarrow \mathcal{V}_S; \mathbf{R} \leftarrow \mathbb{Z}_q^{n^2 \times |S|^2}$ are computationally indistuinguishable.

Let $S \subseteq [d]$ with $|S| \leq K$. We show the indistinguishability of these distributions by showing indistinguishability of a sequence of hybrid distributions. In what follows denote $\alpha = ([\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top \otimes \mathbf{V}^\top - \mathbf{R}]_1, [\mathbf{R}]_2)$ where $\mathbf{U} \leftarrow \mathcal{U}_S, \mathbf{V} \leftarrow \mathcal{V}_S, \mathbf{R} \leftarrow \mathbb{Z}_q^{n^2 \times |S|^2}$.

We have

0. $\alpha, [(\mathbf{U} \otimes \mathbf{V})^\top \boldsymbol{k} - \boldsymbol{r}]_1, [\boldsymbol{r}]_2 :$ $\quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}, \boldsymbol{k} \leftarrow \mathbb{Z}_q^{n^2}$
1. $\alpha, [(\mathbf{U} \otimes \mathbf{V})^\top (\boldsymbol{k}_1 \otimes \boldsymbol{k}_2) - \boldsymbol{r}]_1, [\boldsymbol{r}]_2 :$ $\quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}, \quad \boldsymbol{k}_1, \boldsymbol{k}_2 \leftarrow \mathbb{Z}_q^n$
2. $\alpha, [(\mathbf{U}^\top \boldsymbol{k}_1) \otimes (\mathbf{V}^\top \boldsymbol{k}_2) - \boldsymbol{r}]_1, [\boldsymbol{r}]_2 :$ $\quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}, \quad \boldsymbol{k}_1, \boldsymbol{k}_2 \leftarrow \mathbb{Z}_q^n$
3. $\alpha, [\boldsymbol{u} \otimes (\mathbf{V}^\top \boldsymbol{k}_2) - \boldsymbol{r}]_1, [\boldsymbol{r}]_2 :$ $\quad \boldsymbol{u} \leftarrow \mathbb{Z}_q^n, \quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}, \quad \boldsymbol{k}_2 \leftarrow \mathbb{Z}_q^n$
4. $\alpha, [\boldsymbol{r}]_1, [\boldsymbol{u} \otimes (\mathbf{V}^\top \boldsymbol{k}_2) - \boldsymbol{r}]_2 :$ $\quad \boldsymbol{u} \leftarrow \mathbb{Z}_q^n, \quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}, \quad \boldsymbol{k}_2 \leftarrow \mathbb{Z}_q^n$
5. $\alpha, [\boldsymbol{r}]_1, [\boldsymbol{u} \otimes \boldsymbol{v} - \boldsymbol{r}]_2 :$ $\quad \boldsymbol{u}, \boldsymbol{v} \leftarrow \mathbb{Z}_q^n, \quad \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}$
6. $\alpha, [\boldsymbol{s}]_1, [\boldsymbol{t}]_2 :$ $\quad \boldsymbol{s}, \boldsymbol{t} \leftarrow \mathbb{Z}_q^{n^2}$

We next show that for all $1 \leq i \leq 5$ the distributions $i-1, i$ are computationally indistinguishable.

- *Case* $i = 1$. We show that distinguishing these two distributions reduces to the rank problem in $\mathbb{G}_1$ introduced in [Vil12], namely, distinguishing $[\mathbf{A}]_1 \in \mathbb{G}_1^{n\times n}$ sampled uniformly over all matrices in $\mathbb{G}_1^{n\times n}$ of rank 1, from $[\mathbf{A}]_1 \in \mathbb{G}_1^{n\times n}$ sampled uniformly over all matrices in $\mathbb{G}_1^{n\times n}$ of rank n. Now, assume there exists a distinguisher $\mathcal{A}$ for distributions 0 and 1. We construct a distinguisher $\mathcal{B}$ against the rank problem. The distinguisher works as follows: on input $[\mathbf{A}]_1$, it samples $\mathbf{U} \leftarrow \mathcal{U}_S, \mathbf{V} \leftarrow \mathcal{V}_S, \mathbf{R} \leftarrow \mathbb{Z}_q^{n^2\times|S|^2}, \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}$. It computes $\mathbf{M} = \mathbf{U}^\top[\mathbf{A}]_1\mathbf{V}$ and vectorizes it; denote the vectorization as $[\boldsymbol{m}]_1$. it then executes $\mathcal{A}([\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top \otimes \mathbf{V}^\top - \mathbf{R}]_1, [\mathbf{R}]_2, [\boldsymbol{m}]_1 - [\boldsymbol{r}]_1, [\boldsymbol{r}]_2)$ and outputs whatever $\mathcal{A}$ outputs. Now, note the vectorization $[\boldsymbol{m}]_1$ corresponds to the value $[(\mathbf{U}\otimes\mathbf{V})\boldsymbol{m}]_1$. If $[\mathbf{A}]$ is of rank 1, then we can write $\mathbf{A} = \boldsymbol{k}_1\boldsymbol{k}_2^\top$ and we have $\mathbf{M} = \mathbf{U}^\top\boldsymbol{k}_1\boldsymbol{k}_2^\top\mathbf{V} = \mathbf{U}^\top\boldsymbol{k}_1(\mathbf{V}^\top\boldsymbol{k}_2)^\top$ and the vectorization corresponds to $(\mathbf{U}^\top\boldsymbol{k}_1)\otimes(\mathbf{V}^\top\boldsymbol{k}_2)$, namely the case $i = 0$. Otherwise, $[\mathbf{A}]$ is of rank n, and we can write its vectorization as $\boldsymbol{k}$. Then, $\boldsymbol{m}$ correspond to $(\mathbf{U}^\top \otimes \mathbf{V}^\top)\boldsymbol{k})$, namely the case $i = 1$. As shown in [Vil12], the rank problem reduces to DDH with a security loss of $\log n$.
- *Case* $i = 2$. Distributions $1, 2$ are perfectly indistinguishability since the only difference is that the latter is computed as $[(\mathbf{U}^\top\boldsymbol{k}_1) \otimes (\mathbf{V}^\top\boldsymbol{k}_2) - \boldsymbol{r}]_1$, which equals to $[(\mathbf{U}^\top \otimes \mathbf{V}^\top)(\boldsymbol{k}_1 \otimes \boldsymbol{k}_2) - \boldsymbol{r}]_1$, which is the corresponding value of distribution 1.
- *Case* $i = 3$. This case reduces to the $\mathcal{U}_S^\top$-MDDH_1 assumption. The only difference is that in the forth distribution, we replace $\mathbf{U}^\top\boldsymbol{k}_1$ with a uniform element $\boldsymbol{u}$. It is enough to show that we can compute the rest of the values given $[\mathbf{U}]_1, [\boldsymbol{u}]_1$ where $[\boldsymbol{u}]$ is either $\mathbf{U}^\top\boldsymbol{k}_1$ or uniform. We can compute the values as
$$[\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top]_1 \otimes \mathbf{V}^\top - [\mathbf{R}]_1, [\mathbf{R}]_2, [\boldsymbol{u}]_1 \otimes (\mathbf{V}^\top\boldsymbol{k}_2) - [\boldsymbol{r}]_1, [\boldsymbol{r}]_2$$
where we sample $\mathbf{V} \leftarrow \mathcal{V}_S$, $\mathbf{R} \leftarrow \mathbb{Z}_q^{n^2\times|S|^2}$, $\boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}$, $\boldsymbol{k}_2 \leftarrow \mathbb{Z}_q^n$.
- *Case* $i = 4$. The distributions 4 and 5 are perfectly indistinguishable. It is enough to note that in both, the last two elements are uniformly distributed conditioned on their sum of discrete logarithms being equal to $\boldsymbol{u} \otimes (\mathbf{V}^\top\boldsymbol{k}_2)$.
- *Case* $i = 5$. This is the same as the case $i = 3$ for the value $[\boldsymbol{v}]_2$. This case reduces to the $\mathcal{V}_S^\top$-MDDH_2 assumption. The only difference is that in the last distribution, we replace $\mathbf{V}^\top\boldsymbol{k}_2$ with a uniform element $\boldsymbol{v}$. It is enough to show that we can compute the rest of the values given $[\mathbf{V}]_2, [\boldsymbol{v}]_2$ where $\boldsymbol{v}$ is either $\mathbf{V}^\top\boldsymbol{k}_2$ or uniform. We can compute the values as
$$[\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{R}]_1, \mathbf{U}^\top \otimes [\mathbf{V}^\top]_2 - [\mathbf{R}]_2, [\boldsymbol{r}]_1, \boldsymbol{u} \otimes [\boldsymbol{v}]_2 - [\boldsymbol{r}]_2,$$
where we sample $\mathbf{U} \leftarrow \mathcal{U}_S$, $\mathbf{R} \leftarrow \mathbb{Z}_q^{n^2\times|S|^2}$, $\boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}$, $\boldsymbol{u} \leftarrow \mathbb{Z}_q^n$.
- *Case* $i = 6$. This again reduces to the rank problem in $\mathcal{G}_2$. The only difference in the two distributions is that in distribution 5 the sum of the last two elements, namely $\boldsymbol{u} \otimes \boldsymbol{v}$ is a vectorized matrix of rank 1, namely $\boldsymbol{u}\boldsymbol{v}^\top$, while in distribution 6 is a uniformly distributed matrix of rank n (except w.n.p). Given $[\mathbf{A}]_2 \in \mathbb{G}_2^{n\times n}$ either uniform of rank 1 or uniform of rank n we can compute all the other values efficiently as follows. Let $\boldsymbol{a}$ be the vectorization of $\mathbf{T}$. We compute
$$[\mathbf{U}^\top]_1, [\mathbf{V}^\top]_2, [\mathbf{U}^\top \otimes \mathbf{V}^\top - \mathbf{R}]_1, [\mathbf{R}]_2, [\boldsymbol{r}]_1, [\boldsymbol{a}]_2 - [\boldsymbol{r}]_2,$$
where $\mathbf{U} \leftarrow \mathcal{U}_S, \mathbf{V} \leftarrow \mathcal{V}_S, \mathbf{R} \leftarrow \mathbb{Z}_q^{n^2\times|S|^2}, \boldsymbol{r} \leftarrow \mathbb{Z}_q^{n^2}$. This implies that distinguishing distributions $5, 6$ reduces to the rank problem, which in turn reduces to DDH in $\mathbb{G}_2$.

□

The proof of oblivious trapdoor generation essentially follows from the oblivious trapdoor generation and index set hiding of the SSB commitments and is similar to the corresponding proofs for the other constructions.

First we show the corresponding lemma to Lemma 8, that is, we construct an indistinguishable crs given only the commitment keys and the matrices $\mathbf{U}, \mathbf{V}, \mathbf{W}$.

Lemma 2. *There exists a modified* crs *generation algorithm* K' *that on input* (ρ, θ')*, where either* $\theta' = (\mathbf{U}, \mathbf{V}, \mathbf{W}, [\mathbf{G} \otimes \mathbf{H} - \mathbf{Z}]_1, [\mathbf{Z}]_2)$ *or* $\theta' = (\mathbf{G}, \mathbf{H}, \mathbf{F}, [\mathbf{U} \otimes \mathbf{V} - \mathbf{R}]_1, [\mathbf{R}]_2)$ *and outputs a* crs *such that* (ρ, crs) *are identically distributed to the honest algorithm.*

The lemma follows by inspection an by noting that with the given values we can compute the crs for the sum as explained in Lemma 8. Again, w.l.o.g. we use the same name for the two algorithms, namely K and differentiate them by their input.

Theorem 10. *Let* $\mathcal{U}, \mathcal{V}, \mathcal{W}$ *be witness samplable distributions, and* CS *be the algebraic commitment scheme of Fig. 3 for which* $\mathsf{CS} \otimes \mathsf{CS}$ *is obliviously extractable. Then Construction QAHad of Fig. 7 is* h_{ns}*-strong oblivious where* $h_{ns} = ([\mathbf{G} \otimes \mathbf{H} - \mathbf{Z}]_1, [\mathbf{Z}]_2)$*. Furthermore,*

1. *For every PPT* $\mathcal{A}$ *against index set hiding of QAHad, there exist an adversary* $\mathcal{B}$ *against* h_{ns}*-strong index set hiding property of* CS *such that* $\mathsf{Adv}^{\mathsf{QAHad}}_{\mathsf{ISH}}(\mathcal{A}) \leq 3\mathsf{Adv}^{\mathsf{CS}}_{\mathsf{ISH}}(\mathcal{B})$*.*
2. *For every* $\mathcal{A}$ *against oblivious crs generation of QAHad, there exist an adversary* $\mathcal{B}$ *against oblivious crs generation of QASum such that* $\mathsf{Adv}^{\mathsf{QAHad}}_{\mathsf{oblv}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{QASum}}_{\mathsf{oblv}}(\mathcal{B})$*.*

Proof. It is enough to show that index set hiding holds and that we can sample a tuple (ρ, crs) indistinguishable from the one we are given, along with a valid trapdoor. This is the case because the commitment keys are perfectly binding in S', which means that the witnesses are unique and do not help the (unbounded) distinguisher who can compute them on its own.

h_{ns}-Strong Index Set Hidning. Assume there exist sets S, S' of size at most K and an adversary $\mathcal{A}$ which distinguishes $(\rho, \mathsf{crs}, h_{ns}(\theta))$ sampled for S from $(\rho, \mathsf{crs}, h_{ns}(\theta))$ sampled for S' with some probability α. We construct adversaries $\mathcal{B}$ distinguishing ck sampled for S from ck sampled for S with probability β such that $\alpha \leq 2\beta$.

$\mathcal{B}$ takes as input some ck sampled either for S or S' which is parsed as $[\mathbf{G}]_1$ and honestly computes the crs following K of Lemma 2 using the values $[\mathbf{G} \otimes \mathbf{H} - \mathbf{Z}]_1, [\mathbf{Z}]_2$ which are included in h_{ns} except that $[\mathbf{H}]_2, \mathbf{H}, \mathbf{T_H}, [\mathbf{F}]_1, \mathbf{F}, \mathbf{T_F}$ are computed as follows: it samples $b \leftarrow \{0, 1\}$ and if $b = 0$ it sets

$$([\mathbf{H}]_2, \mathbf{H}, \mathbf{T_H}) \leftarrow \mathsf{CS.KeyGen}(gk_2, d, K, S), \quad ([\mathbf{F}]_1, \mathbf{F}, \mathbf{T_F}) \leftarrow \mathsf{CS.KeyGen}(gk_1, d, K, S)$$

otherwise it sets

$$([\mathbf{H}]_2, \mathbf{H}, \mathbf{T_H}) \leftarrow \mathsf{CS.KeyGen}(gk_2, d, K, S'), \quad ([\mathbf{F}]_1, \mathbf{F}, \mathbf{T_F}) \leftarrow \mathsf{CS.KeyGen}(gk_1, d, K, S')$$

If the guess b is correct, by witness samplability of $\mathbf{U}, \mathbf{V}, \mathbf{W}$ the distribution of ρ is not changed, and since the crs is computed as an honest one conditioned on ρ, index set hiding follows holds with probability $\frac{\alpha}{2}$.

Oblivious trapdoor generation: Here, we can simply use the oblivious trapdoor generation of protocol QASum. The conditions of corollary 7 are satisfied since we include the values $[\mathbf{G} \otimes \mathbf{H} - \mathbf{Z}]_1, [\mathbf{Z}]_2$ in h_{ns} and by Thm 5 the commitment key for the sum has oblivious trapdoor generation. It is enough to show that we can compute the crs for the QAHad given a crs for QASum. But this is easy since when given a pair $(\rho_{\text{sum}}, \mathsf{crs}_{\text{sum}})$ we execute the oblivious crs algorithm $\mathsf{QASum.OblKeyGen}(\rho, \mathsf{crs}, \boldsymbol{S} = (S, S))$ as in Lemma 2.

□

Corollary 5. *If* CS *is the one from fig. 3, then QAHad from fig. 7 is* h_{ns}*-strong no-signaling where* $h_{ns} = ([\mathbf{G} \otimes \mathbf{H} - \mathbf{Z}]_1, [\mathbf{Z}]_2, \mathbf{U}, \mathbf{V}, \mathbf{W})$ *.*

Proof. The proof follows directly from Theorem 6 and the h_{ns}-strong oblivious trapdoor generation of QAHad which is shown in Thm. 10. □

6 Delegation for Arithmetic Circuit Evaluation

Formally, we define a delegation scheme as follows.

Definition 10. *A triplet of algorithms* $\mathsf{Del} = (\mathsf{Setup}, \mathsf{Prove}, \mathsf{Verify})$ *is a delegation scheme for circuit evaluation with preprocessing if for any circuit* $C : \mathbb{Z}_p^{n_0} \to \mathbb{Z}_p^{n_d}$*:*

Completeness: *For any* $\boldsymbol{x}, \boldsymbol{y}$ *such that* $\boldsymbol{y} = C(\boldsymbol{x})$ *it holds*

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)}[\mathsf{Verify}(\mathsf{crs}, \boldsymbol{x}, \boldsymbol{y}, \pi) = 1 | \mathsf{crs} \leftarrow \mathsf{Setup}(gk, C), \pi \leftarrow \mathsf{Prove}(\mathsf{crs}, \boldsymbol{x}, \boldsymbol{y})] \geq 1 - \mathsf{negl}(\kappa),$$

Soundness: *For any adversary* $\mathcal{A}$ *it holds that*

$$\Pr_{gk \leftarrow \mathcal{G}(1^\kappa)}[\mathsf{Verify}(\mathsf{crs}, \boldsymbol{x}, \boldsymbol{y}, \pi) = 1 \text{ and } \boldsymbol{y} \neq C(\boldsymbol{x}) | \mathsf{crs} \leftarrow \mathsf{Setup}(gk, C), (\boldsymbol{x}, \boldsymbol{y}, \pi) \leftarrow \mathcal{A}(\mathsf{crs})] \leq \mathsf{negl}(\kappa),$$

Efficiency: *The setup algorithm and the prover run in time* $\mathsf{poly}(|C|, \kappa)$*. The size of each proof is* $O(\kappa)$ *and verification time* $n\mathsf{poly}(\kappa) + \mathsf{poly}(\kappa)$*.*

6.1 The Scheme

In the delegation scheme from [GR19] the prover, gives $3d$ commitments $[L_1]_1, \ldots, [L_d]_1, [R_1]_2, \ldots, [R_d]_2$, $[O_1]_1, \ldots, [O_d]_1$ to, respectively, the left, right and output wires of each level of the circuit. Then, it gives a linear and quadratic knowledge transfer arguments to "transfer" knowledge of the opening from the input level, which is known to the verifier, to the next levels. Finally, the verifier checks that the commitment to the output opens to $\boldsymbol{y}$.

We give a "compressed" version of [GR19] where the $3d$ commitments are shrunken into 3 no-signaling SSB commitments, and the $2d$ knowledge transfer arguments are shrunk into 2 quasi arguments. From the SSB commitments we can extract $[L_i]_1[R_i]_2, [O_j]_1$ for $j = i-1$ or $j = i$. Local knowledge soundness of the quasi arguments imply that knowledge is "transferred" from $[O_{i-1}]_1$ to $[L_i]_1, [R_i]_2$ or from $[L_i]_1, [R_i]_2$ to $[O_i]_1$. One important technical problem with this approach is that the linear knowledge transfer argument is between the next level and all previous levels. That is, the knowledge is transferred from commitments to the output in all previous levels $[O_1]_1, \ldots, [O_i]_1$, to commitments to the left and right wires in the next level $[L_{i+1}]_1, [R_{i+1}]_2$. This means the quasi-argument must extract $O(d)$ values and hence is not succinct. We solve this issue by computing L_i, R_i, O_i as commitments also to the respective wires of all previous levels. Consider an arithmetic circuit $C : \mathbb{Z}_p^{n_0} \to \mathbb{Z}_p^{n_d}$. The circuit can be naturally sliced into $d+1$ levels, where level 0 contains the input and level i is formed by a set of n_i multiplication gates, the inputs of which depends on a linear transformation of outputs of previous levels.[15] Let $N_i = \sum_{j=0}^{i}$ and $N = N_d$. Denote by $\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i \in \mathbb{Z}_p^{N_i}$ the left, right and output wires of level $1, \ldots, i$ respectively. That is $\boldsymbol{a}_i = \left(\begin{smallmatrix} \boldsymbol{a}_{i-1} \\ \mathbf{D}_i \boldsymbol{c}_{i-1} \end{smallmatrix}\right)$ and $\boldsymbol{b}_i = \left(\begin{smallmatrix} \boldsymbol{b}_{i-1} \\ \mathbf{E}_i \boldsymbol{c}_{i-1} \end{smallmatrix}\right)$, where $\mathbf{D}_i, \mathbf{E}_i \in \mathbb{Z}_p^{n_i \times N_{i-1}}$ are defined by the circuit's linear gates, $\boldsymbol{a}_0, \boldsymbol{b}_0$ are of size 0 and $\boldsymbol{c}_0 = \boldsymbol{x}$ is the input. Let $\mathbf{D} \in \mathbb{Z}_p^{N - n_0 \times N}$ (resp. $\mathbf{E}$) be the matrix such that the i-th row of $\mathbf{D}$ is $(\mathbf{D}_i | \mathbf{0}_{n_i \times N - N_{i-1}})$. Note that matrices $\mathbf{D}, \mathbf{E}$ are lower triangular. For the outputs we have $\boldsymbol{c}_i = \boldsymbol{a}_i \circ \boldsymbol{b}_i$.

Denote $\boldsymbol{a} = \boldsymbol{a}_d$, $\boldsymbol{b} = \boldsymbol{b}_d$ and $\boldsymbol{c} = \boldsymbol{c}_{d-1}$. The evaluation of the circuit is correct if $\left(\begin{smallmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{smallmatrix}\right) = \left(\begin{smallmatrix} \mathbf{D} \\ \mathbf{E} \end{smallmatrix}\right) \boldsymbol{c}$ and $\boldsymbol{c} = \boldsymbol{a} \circ \boldsymbol{b}$. Next, consider Pedersen commitment keys $\mathbf{U}_i^* \leftarrow \mathbb{Z}_p^{1 \times n_i}$, $\mathbf{V}_i^* \leftarrow \mathbb{Z}_p^{1 \times n_i}$ and $\mathbf{W}_i^* \leftarrow \mathbb{Z}_p^{1 \times n_i}$ and define $\mathbf{U}_i = (\mathbf{U}_1^*, \ldots, \mathbf{U}_i^*), \mathbf{V}_i = (\mathbf{V}_1^*, \ldots, \mathbf{V}_i^*)$, for $i \in [d]$, $\mathbf{W}_i = (\mathbf{W}_1^*, \ldots, \mathbf{W}_i^*)$, for $i \in [d-1]$. Consider commitments (represented in $\mathbb{Z}_p$) to left, right and output wires as $O_i = \mathbf{W}_i \boldsymbol{c}_i, \boldsymbol{O} = \mathbf{W}\boldsymbol{c}, L_i = \mathbf{U}_i \boldsymbol{a}_i = \mathbf{U}\boldsymbol{a}, R_i = \mathbf{V}_i \boldsymbol{b}_i, \boldsymbol{R} = \mathbf{V}\boldsymbol{b}$, where

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_1^* & & \mathbf{0} \\ \vdots & \ddots & \\ \mathbf{U}_1^* & \ldots & \mathbf{U}_d^* \end{pmatrix}, \mathbf{V} = \begin{pmatrix} \mathbf{V}_1^* & & \mathbf{0} \\ \vdots & \ddots & \\ \mathbf{V}_1^* & \ldots & \mathbf{V}_d^* \end{pmatrix}, \mathbf{W} = \begin{pmatrix} \mathbf{W}_1^* & & \mathbf{0} \\ \vdots & \ddots & \\ \mathbf{W}_1^* & \ldots & \mathbf{W}_{d-1}^* \end{pmatrix}, \tag{3}$$

[15] We consider w.l.o.g. only linear transformations since if we can handle affine ones by including a wire with the value 1 in the input.

$\boldsymbol{O} = (O_1, \ldots, O_{d-1})^\top, \boldsymbol{L} = (L_1, \ldots, L_d)^\top, \boldsymbol{R} = (R_1, \ldots, R_d)^\top$.

We additionally pick $\mathbf{G}, \mathbf{H}, \mathbf{F}$ for computing SSB commitments to vectors of size d and publish $[\mathbf{GU}]_1$, $[\mathbf{HV}]_2$, $[\mathbf{FW}]_2$. The prover computes $[\hat{\boldsymbol{L}}]_1 = [\mathbf{GU}]_1\boldsymbol{a}, [\hat{\boldsymbol{R}}]_2 = [\mathbf{HV}]_2\boldsymbol{b}, [\hat{\boldsymbol{O}}]_1 = [\mathbf{FW}]_1\boldsymbol{c}$ and gives a quasi-argument of linear knowledge transfer from $\boldsymbol{x}, [\boldsymbol{O}]_1, \boldsymbol{y}$ to $[\boldsymbol{L}]_1, [\boldsymbol{R}]_2$ with the following structure

$$\begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{O} \\ \boldsymbol{y} \\ \boldsymbol{L} \\ \boldsymbol{R} \end{pmatrix} = \begin{pmatrix} \overbrace{\mathbf{I}_{n_0}}^{\text{input}} & \overbrace{\mathbf{0}}^{\text{mid-wires}} & \overbrace{\mathbf{0}}^{\text{output}} \\ \mathbf{0} & \boxed{\mathbf{W}} & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n_d} \\ \boxed{\mathbf{UD}} & & \mathbf{0} \\ \boxed{\mathbf{VE}} & & \mathbf{0} \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{c} \\ \boldsymbol{y} \end{pmatrix}. \tag{4}$$

That is, we can extract $[L_i]_1, [R_i]_2, [O_{i-1}]_1$ and, if we are additionally given $\boldsymbol{c}_{i-1}$ such that $O_{i-1} = \mathbf{W}_{i-1}\boldsymbol{c}_{i-1}$, then $L_i = \mathbf{U}_i\mathbf{D}_i\boldsymbol{c}_i, R_i = \mathbf{V}_i\mathbf{E}_i\boldsymbol{c}_i$. We also use a quasi-argument of knowledge transfer of the hadamard product from $[\boldsymbol{L}]_1, [\boldsymbol{R}]_2$ to $[\boldsymbol{O}]_1$. In this case we extract $[L_i]_1, [R_i]_2, [O_i]_1$ and, if we are additionally given $\boldsymbol{a}_i, \boldsymbol{b}_i$ such that $L_i = \mathbf{U}_i\boldsymbol{a}_i$ and $R_i = \mathbf{V}_i\boldsymbol{b}_i$, then $O_i = \mathbf{W}_i(\boldsymbol{a}_i \circ \boldsymbol{b}_i)$.

We need to make one last change that will allow us to take into account the input $\boldsymbol{x}$ and the claimed output $\boldsymbol{y}$. Essentially, we make the first and last commitment key (trivially) perfectly binding by using as a commitment key the identity matrix. The security properties still hold in a trivial way (the $\mathbf{I}_{n_0}$-MDDH assumption is perfectly secure). We change accordingly the SSB commitment key, that is we set $\mathbf{F}' = \begin{pmatrix} \mathbf{I}_{n_0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{n_d} \end{pmatrix}$. Note that the extraction trapdoor remains the same, but the extractor can trivially extract the values corresponding to $\boldsymbol{x}, \boldsymbol{y}$ regardless of $\mathbf{F}'$ distribution. In other words, our commitment key is always perfectly binding in the first n_0 and n_d coordinates. We denote with $\mathbf{W}'$ the modified matrix where we change the first and last rows with $(\mathbf{I}_{n_0} \mid \mathbf{0})$ and $(\mathbf{0} \mid \mathbf{I}_{n_d})$ respectively. Therefore, if $\boldsymbol{O} = \mathbf{W}'\boldsymbol{c}$, we get that $O_0 = \boldsymbol{x}$ and $O_d = \boldsymbol{y}$.

$\mathsf{Setup}(gk, C)$**:**

- From the linear gates of C compute matrices $\mathbf{D}, \mathbf{E}$.
- $([\mathbf{F}]_1, \mathbf{F}, \mathbf{T_F}) \leftarrow \mathsf{CS.KeyGen}(gk, d-1, 1, \emptyset)$,
  $([\mathbf{G}]_1, \mathbf{G}, \mathbf{T_G}) \leftarrow \mathsf{CS.KeyGen}(gk, d, 1, \emptyset)$, $([\mathbf{H}]_1, \mathbf{H}, \mathbf{T_H}) \leftarrow \mathsf{CS.KeyGen}(gk, d, 1, \emptyset)$;
- Sample $\mathbf{U}, \mathbf{V}, \mathbf{W}$ as in equation 3. Define $\mathbf{W}'$ as the matrix $\mathbf{W}$ augmented with $(\mathbf{I}_{n_0} \mid \mathbf{0})$ and $(\mathbf{0} \mid \mathbf{I}_{n_d})$ as its first and last row.
- Let $\rho_{\mathsf{blin}} = (gk, [\mathbf{F}']_1, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{W}']_1, [\mathbf{UD}]_1, [\mathbf{VE}]_2)$ and $\theta_{\mathsf{blin}} = (\mathbf{F}, \mathbf{G}, \mathbf{H}, \mathbf{T_F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{U}', \mathbf{UD}, \mathbf{VE})$, where $\mathbf{F}'$ contains rows $(\mathbf{I}_n \mid \mathbf{0} \mid \mathbf{0}), (\mathbf{0} \mid \mathbf{F} \mid \mathbf{0}), (\mathbf{0} \mid \mathbf{0} \mid \mathbf{I}_{n_d})$.
- Let $\rho_{\mathsf{had}} = (gk, [\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}'']_1, [\mathbf{U}]_1, [\mathbf{V}]_2, [\mathbf{U}]_1)$ and $\theta_{\mathsf{had}} = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{T_G}, \mathbf{T_H}, \mathbf{T_F}, \mathbf{U}, \mathbf{V}, \mathbf{W})$, where $\mathbf{F}''$ contains the rows $(\mathbf{F} \mid \mathbf{0}), (\mathbf{0} \mid \mathbf{I}_{n_d})$.
- Sample $\mathsf{crs}_{\mathsf{blin}} \leftarrow \mathsf{QABlin.K}(\rho_{\mathsf{blin}}, \theta_{\mathsf{blin}})$ and $\mathsf{crs}_{\mathsf{had}} \leftarrow \mathsf{QAHad.K}(\rho_{\mathsf{had}}, \theta_{\mathsf{had}})$
- output $\mathsf{crs} := ([\mathbf{GU}]_1, [\mathbf{HV}]_2, [\mathbf{FW}]_1, \mathsf{crs}_{\mathsf{lin}}, \mathsf{crs}_{\mathsf{had}})$

$\mathsf{Prove}(\mathsf{crs}, \boldsymbol{x}, \boldsymbol{y})$**:**

- Evaluate the circuit on input $\boldsymbol{x}$ to obtain values for the wires $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$.
- Compute $[\hat{\boldsymbol{L}}]_1 = [\mathbf{GU}]_1 \boldsymbol{a}, [\hat{\boldsymbol{R}}]_2 = [\mathbf{HV}]_2 \boldsymbol{b}, [\hat{\boldsymbol{O}}]_1 = [\mathbf{FW}]_1 \boldsymbol{c}$.
- $\pi_{\mathsf{blin}} \leftarrow \mathsf{QABlin.Prove}(\mathsf{crs}_{\mathsf{blin}}, \begin{pmatrix} \boldsymbol{x} \\ [\hat{\boldsymbol{O}}]_1 \\ \boldsymbol{y} \end{pmatrix}, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2), (\boldsymbol{x}, \boldsymbol{c}, \boldsymbol{y}))$.
- $\pi_{\mathsf{had}} \leftarrow \mathsf{QAHad.Prove}(\mathsf{crs}_{\mathsf{had}}, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \begin{pmatrix} [\hat{\boldsymbol{O}}]_1 \\ \boldsymbol{y} \end{pmatrix}, \boldsymbol{a}, \boldsymbol{b})$.
- Return $\pi = ([\hat{\boldsymbol{O}}]_1, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \pi_{\mathsf{blin}}, \pi_{\mathsf{had}})$.

$\mathsf{Verify}(\mathsf{crs}, (\boldsymbol{x}, \boldsymbol{y}), \pi)$**:**

- Parse $\pi := ([\hat{\boldsymbol{O}}]_1, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \pi_{\mathsf{blin}}, \pi_{\mathsf{had}})$.
- Output 1 if the following tests are successful and 0 otherwise:
  - $\mathsf{QABlin.Verify}(\mathsf{crs}_{\mathsf{blin}}, \begin{pmatrix} \boldsymbol{x} \\ [\hat{\boldsymbol{O}}]_1 \\ \boldsymbol{y} \end{pmatrix}, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2), \pi_{\mathsf{blin}}) = 1$ and
  - $\mathsf{QAHad.Verify}(\mathsf{crs}_{\mathsf{had}}, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \begin{pmatrix} [\hat{\boldsymbol{O}}]_1 \\ \boldsymbol{y} \end{pmatrix}, \pi_{\mathsf{had}}) = 0$

Figure 8: Delegation scheme for an arithmetic circuit.

6.2 Proof of Security

Theorem 11. *Let $\mathcal{A}$ be an adversary against Adaptive Soundness of the delegation scheme of Fig. 8, that outputs an input/output pair $\boldsymbol{x}, \boldsymbol{y}^*$ and a valid proof $\pi := \left([\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, [\hat{\boldsymbol{O}}]_1, \pi_{\mathsf{had}}, \pi_{\mathsf{blin}}\right)$ but $\boldsymbol{y}^* \neq C(\boldsymbol{x})$. Then there exists a distinguisher $\mathcal{D}_{\mathsf{blin}}, \mathcal{D}_{\mathsf{had}}$ and adversaries $\mathcal{B}_{\mathsf{blin}}, \mathcal{B}_{\mathsf{had}}$ against the no-signaling property of* QABlin *and* QAHad*, respectively, and adversaries $\mathcal{A}_{\mathsf{blin}}, \mathcal{A}_{\mathsf{had}}$ against local knowledge soundness of* QABlin *and local knowledge soundness* QAHad*, respectively, such that*

$$\mathsf{Adv}_{\mathsf{Del}}(\mathcal{A}) \leq 6d \left(\mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) + \mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}})\right) + d \left(\mathsf{Adv}_{\mathsf{snd}}^{\mathsf{QAHad}}(\mathcal{A}_{\mathsf{had}}) + \mathsf{Adv}_{\mathsf{snd}}^{\mathsf{QABLin}}(\mathcal{A}_{\mathsf{blin}})\right).$$

Proof. For $i \in [d]$, consider the following experiments

Game_0**:** This is the soundness game where the adversary wins if outputs $\boldsymbol{x}, \boldsymbol{y}^*$ and a valid π but $C(\boldsymbol{x}) \neq \boldsymbol{y}^*$.

$\mathsf{BadO}_{i,\boldsymbol{S}}$**:** As Game_0 but $\mathsf{crs}_{\mathsf{blin}}$ and $\mathsf{crs}_{\mathsf{had}}$ are perfectly binding in $\boldsymbol{S}$; if possible, $[O_i]_1$ is extracted from the adversary's proof; the game returns 1 if $[O_i]_1 \neq [\mathbf{W}_i^*]_1 \boldsymbol{c}_i$, where $\boldsymbol{c}_i$ is computed from $\boldsymbol{x}$.

$\mathsf{BadLR}_{i,\boldsymbol{S}}$**:** As Game_0 but $\mathsf{crs}_{\mathsf{blin}}$ and $\mathsf{crs}_{\mathsf{had}}$ are perfectly binding in $\boldsymbol{S}$; $[L_i]_1, [R_i]_2$ are extracted from the adversary's proof; the game returns 1 if $[L_i]_1 \neq [\mathbf{U}_i^*]_1 \boldsymbol{a}_i$ or $[R_i]_1 \neq [\mathbf{V}_i^*]_1 \boldsymbol{b}_i$, where $\boldsymbol{a}_i, \boldsymbol{b}_i$ are computed from $\boldsymbol{x}$.

We define $O_d = \boldsymbol{y}^*$ and $\mathbf{W}_d^* = (\mathbf{0}_{n_d \times N - n_d} | \mathbf{I}_{n_d})$ so that $\mathsf{Game}_0 = \mathsf{BadO}_{d,(\emptyset,\emptyset)}$. We also define $\mathsf{BadO}_i = \mathsf{BadO}_{i,(\emptyset,\emptyset)}$.

We show that

$$\begin{aligned}\Pr[\mathsf{BadO}_i = 1] \leq & 6i \left(\mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) + \mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) \right) + \\ & i \left(\mathsf{Adv}_{\mathsf{snd}}^{\mathsf{QAHad}}(\mathcal{A}_{\mathsf{had}}) + \mathsf{Adv}_{\mathsf{snd}}^{\mathsf{QABLin}}(\mathcal{A}_{\mathsf{blin}}) \right).\end{aligned}$$

The proof is by induction over i. In the inductive case we show that

$$\begin{aligned}\Pr[\mathsf{BadO}_i = 1] \approx \Pr[\mathsf{BadO}_{i,(\{i\},\{i\})} = 1] \approx \Pr[\mathsf{BadLR}_{i,(\{i\},\{i\})} = 1] \approx \Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})} = 1] \\ \text{and } \Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})} = 1] \approx \Pr[\mathsf{BadO}_{i-1,(\{i\},\{i-1\})} = 1] \approx \Pr[\mathsf{BadO}_{i-1} = 1]\end{aligned} \tag{5}$$

where $p_1 \approx p_2$ is defined as $|p_1 - p_2| \leq \mathsf{negl}(\kappa)$. Now we show that each $\approx$ is indeed negligible. Note that ρ_{had} can be computed from ρ_{blin} and vice-versa.

$\mathsf{BadO}_i, \mathsf{BadO}_{i,(\{i\},\{i\})}$**:** Consider the sets $\boldsymbol{S} = (\emptyset, \emptyset), \boldsymbol{S}' = (\emptyset, \{i\})$ and $\boldsymbol{S}'' = (\{i\}, \{i\})$. We build distinguishers for no-signaling extraction and use them twice, first for distinguishing $\boldsymbol{S}, \boldsymbol{S}'$ and then $\boldsymbol{S}', \boldsymbol{S}''$.

We construct adversaries $\mathcal{D}_{\mathsf{blin}}, \mathcal{B}_{\mathsf{blin}}$ against no-signaling extraction of QABlin. By Corollary 7, the no-signaling property holds even when $\mathcal{B}_{\mathsf{blin}}$ is given $\rho_{\mathsf{blin}}, \mathsf{crs}_{\mathsf{blin}}$ and additionally $h_{\mathsf{ns}}(\theta_{\mathsf{blin}}) = (\mathbf{U}, \mathbf{V}, \mathbf{W}, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [-\mathbf{Z}]_2)$. Using this additional help, $\mathcal{D}_{\mathsf{blin}}$ computes $\mathsf{crs}_{\mathsf{had}} \leftarrow \mathsf{QAHad.K}(\rho_{\mathsf{had}}, \theta' = h_{\mathsf{ns}}(\theta_{\mathsf{blin}}))$ as in Lemma 2. It then runs $\mathcal{A}(\mathsf{crs})$ until it outputs $(\boldsymbol{x}, \boldsymbol{y}^*, [\hat{\boldsymbol{O}}]_1, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \pi_{\mathsf{blin}}, \pi_{\mathsf{had}})$, and then $\mathcal{B}_{\mathsf{blin}}$ outputs $(\left[\begin{smallmatrix} \boldsymbol{x} \\ \hat{\boldsymbol{O}} \\ \boldsymbol{y}^* \end{smallmatrix}\right]_1, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2)$ and π_{blin}. Adversary $\mathcal{D}_{\mathsf{blin}}$ outputs 1 if and only if $[O_i]_1 \neq [\mathbf{W}_i^*]_1 \boldsymbol{c}_i$ and π_{blin} is accepting.

Similarly, we construct adversaries $\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}$ against the no-signaling property of QAHad. The adversary $\mathcal{B}_{\mathsf{had}}$ additionally receives $h_{\mathsf{ns}}(\theta_{\mathsf{had}}) = (\mathbf{U}, \mathbf{V}, \mathbf{W}, [\mathbf{G} \otimes \mathbf{H} + \mathbf{Z}]_1, [\mathbf{Z}]_2)$ (as shown in Corollary 5). Using this additional help, $\mathcal{B}_{\mathsf{had}}$ computes $\mathsf{crs}_{\mathsf{blin}} \leftarrow \mathsf{QABLin.K}(\rho_{\mathsf{blin}}, \theta' = h_{\mathsf{ns}}(\theta_{\mathsf{had}}))$ as in Lemma 7. It then runs $\mathcal{A}(\mathsf{crs})$ until it outputs $(\boldsymbol{x}, \boldsymbol{y}^*, [\hat{\boldsymbol{O}}]_1, [\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \pi_{\mathsf{blin}}, \pi_{\mathsf{had}})$, and then $\mathcal{B}_{\mathsf{had}}$ outputs $([\hat{\boldsymbol{L}}]_1, [\hat{\boldsymbol{R}}]_2, \left[\begin{smallmatrix} \hat{\boldsymbol{O}} \\ \boldsymbol{y}^* \end{smallmatrix}\right]_1)$ and π_{had}. Adversary $\mathcal{D}_{\mathsf{had}}$ outputs 1 if and only if $[O_i]_1 \neq [\mathbf{W}_i^*]_1 \boldsymbol{c}_i$ and π_{had} is accepting.

Note that $\mathsf{BadO}_i, \mathsf{BadO}_{i,(\{i\},\emptyset)}$ and $\mathsf{BadO}_{i,(\{i\},\{i\})}$ outputs 1 if and only if $\mathcal{D}_{\mathsf{blin}}$ and $\mathcal{D}_{\mathsf{had}}$ output 1 when the crs is sampled for $\boldsymbol{S}, \boldsymbol{S}'$ and $\boldsymbol{S}''$, respectively. Then we can bound $|\Pr[\mathsf{BadO}_i = 1] - \Pr[\mathsf{BadO}_{i,(\{i\},\{i\})} = 1]| \leq 2 \left(\mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QABlin}}(\mathcal{D}_{\mathsf{blin}}, \mathcal{B}_{\mathsf{blin}}) + \mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) \right)$.

$\mathsf{BadO}_{i,(\{i\},\{i\})}, \mathsf{BadLR}_{i,(\{i\},\{i\})}$**:** Note that $|\Pr[\mathsf{BadO}_{i,(\{i\},\{i\})} = 1] - \Pr[\mathsf{BadLR}_{i,(\{i\},\{i\})} = 1]| \leq \Pr[\mathsf{BadO}_{i,(\{i\},\{i\})} = 1 \text{ and } \mathsf{BadLR}_{i,(\{i\},\{i\})} \neq 1]$. The last value is the probability that $\mathcal{A}$'s proof locally opens to $[L_i]_1, [R_i]_2, [O_i]_1$ and $[L_i]_1 = [\mathbf{U}_i^*]_1 \boldsymbol{a}_i, [R_i]_2 = [\mathbf{V}_i^*]_2 \boldsymbol{b}_i$ but $[O_i] \neq [\mathbf{W}_i^*]_1 \boldsymbol{c}_i = [\mathbf{W}_i^*]_1 (\boldsymbol{a}_i \otimes \boldsymbol{b}_i)$. Then we can build an adversary $\mathcal{A}_{\mathsf{had}}$ against the h-strong knowledge soundness of QAHad. On input $\mathsf{crs}_{\mathsf{had}}$ and $h_{\mathsf{ls}}(\theta_{\mathsf{had}}) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{W})$ computes $\mathsf{crs}_{\mathsf{blin}} \leftarrow \mathsf{QABlin.K}(\rho_{\mathsf{blin}}, \theta' = h_{\mathsf{ls}}(\theta_{\mathsf{had}}))$ as in Lemma 7. Then runs $\mathcal{A}(\mathsf{crs})$ until it outputs $\boldsymbol{x}, \boldsymbol{y}^*, \pi$ from which $\mathcal{A}_{\mathsf{had}}$ outputs $([\boldsymbol{L}]_1, [\boldsymbol{R}]_2, \left[\begin{smallmatrix} \hat{\boldsymbol{O}} \\ \boldsymbol{y}^* \end{smallmatrix}\right]_1)$ and π_{had}. We conclude that $\Pr[\mathsf{BadO}_{i,(\{i\},\{i\})} = 1 \text{ and } \mathsf{BadLR}_{i,(\{i\},\{i\})} \neq 1] \leq \mathsf{Adv}_{\mathsf{snd}}^{\mathsf{QAHad}}(\mathcal{A}_{\mathsf{had}})$.

$\mathsf{BadLR}_{i,(\{i\},\{i\})}, \mathsf{BadLR}_{i,(\{i\},\{i-1\})}$**:** Similarly as the case $\mathsf{BadO}_i, \mathsf{BadO}_{i,(\{i\},\{i\})}$, but we need to transition between sets $(\{i\},\{i\}) \to (\{i\},\emptyset) \to (\{i\},\{i-1\})$. Therefore, $|\Pr[\mathsf{BadLR}_i = 1] - \Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})} = 1]| \leq 2 \left(\mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QAHad}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) + \mathsf{Adv}_{\mathsf{NS}}^{\mathsf{QABlin}}(\mathcal{D}_{\mathsf{blin}}, \mathcal{B}_{\mathsf{blin}}) \right)$.

$\mathsf{BadLR}_{i,(\{i\},\{i-1\})}, \mathsf{BadO}_{i-1,(\{i\},\{i-1\})}$**:** Note that $|\Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})}=1] - \Pr[\mathsf{BadO}_{i-1,(\{i\},\{i-1\})}=1]| \leq \Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})}=1 \text{ and } \mathsf{BadO}_{i-1,(\{i\},\{i-1\})} \neq 1]$. The last value is the probability that $\mathcal{A}$'s proof locally opens to $[O_{i-1}]_1, [L_i]_1, [R_i]_2$ and $[O_{i-1}]_1 = [\mathbf{W}^*_{i-1}]\boldsymbol{c}_{i-1}$ but $[L_i]_1 \neq [\mathbf{U}^*_i]_1\mathbf{D}_i\boldsymbol{c}_{i-1}$ or $[R_i]_2 \neq [\mathbf{V}^*_i]\mathbf{E}_i\boldsymbol{c}_{i-1}$. Then we can build an adversary $\mathcal{A}_{\mathsf{lin}}$ against the h-strong knowledge soundness of QABlin. On input $\mathsf{crs}_{\mathsf{blin}}$ and $h_{\mathsf{ls}}(\theta_{\mathsf{blin}}) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{U}, \mathbf{V})$ computes $\mathsf{crs}_{\mathsf{had}} \leftarrow \mathsf{QAHad.K}(\rho_{\mathsf{had}}, h_{\mathsf{ls}}(\theta_{\mathsf{blin}}))$, as in Lemma 2. Then runs $\mathcal{A}(\mathsf{crs})$ until it outputs $\boldsymbol{x}, \boldsymbol{y}^*, \pi$ and then $\mathcal{A}_{\mathsf{blin}}$ outputs $(\left[\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{O}\\ \boldsymbol{y}^*\end{smallmatrix}\right]_1, [\boldsymbol{L}]_1, [\boldsymbol{R}]_2)$ and π_{blin}. We conclude that $\Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})}=1 \text{ and } \mathsf{BadO}_{i-1,(\{i\},\{i-1\})} \neq 1] \leq \mathsf{Adv}^{\mathsf{QABLin}}_{\mathsf{snd}}(\mathcal{A}_{\mathsf{blin}})$.

$\mathsf{BadO}_{i-1,(\{i\},\{i-1\})}, \mathsf{BadO}_i$**:** Similarly as the case $\mathsf{BadO}_i, \mathsf{BadO}_{i,(\{i\},\{i\})}$, $|\Pr[\mathsf{BadLR}_i=1] - \Pr[\mathsf{BadLR}_{i,(\{i\},\{i-1\})}=1]| \leq 2\left(\mathsf{Adv}^{\mathsf{QAHad}}_{\mathsf{NS}}(\mathcal{D}_{\mathsf{had}}, \mathcal{B}_{\mathsf{had}}) + \mathsf{Adv}^{\mathsf{QABlin}}_{\mathsf{NS}}(\mathcal{D}_{\mathsf{blin}}, \mathcal{B}_{\mathsf{blin}})\right)$.

Then, assuming the lemma holds for BadO_{i-1} and adding the previous advantages, we get that it also holds for BadO_i.

In the base case $i = 1$, we show that

$$\Pr[\mathsf{BadO}_1 = 1] \approx \Pr[\mathsf{BadO}_{i,(\{1\},\{1\})} = 1] \approx \Pr[\mathsf{BadLR}_{1,\{1\},\{1\}} = 1] \approx 0$$

We can reuse equation 5 to get that $\Pr[\mathsf{BadO}_1 = 1] \leq \mathsf{negl}(\kappa) + \Pr[\mathsf{BadLR}_{1,\{1\},\{1\}} = 1]$. Now we show that $\Pr[\mathsf{BadLR}_{1,\{1\},\{1\}} = 1]$ is negligible. Note that $\mathsf{BadLR}_{1,\{1\},\{1\}} = 1$ implies that $[L_1]_1 \neq [\mathbf{U}^*_1]\boldsymbol{a}_1 = [\mathbf{U}^*_1]\mathbf{D}_1\boldsymbol{c}_0$ or $[R_1]_2 \neq [\mathbf{V}^*_1]_2\boldsymbol{b}_1 = [\mathbf{V}^*_1]_2\mathbf{E}_1\boldsymbol{c}_0$, but also $\boldsymbol{x} = \mathbf{I}_{n_0}\boldsymbol{c}_0$. Hence we can build an adversary $\mathcal{A}_{\mathsf{blin}}$ against the h-strong knowledge soundness of QABLin. On input $\mathsf{crs}_{\mathsf{bin}}$ and $h_{\mathsf{ls}}(\theta_{\mathsf{blin}}) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{U}, \mathbf{V})$ computes $\mathsf{crs}_{\mathsf{had}} \leftarrow \mathsf{QAHad.K}(\rho_{\mathsf{had}}, \theta' = h_{\mathsf{ls}}(\theta_{\mathsf{blin}}))$ as in Lemma 2. Then runs $\mathcal{A}(\mathsf{crs})$ until it outputs $\boldsymbol{x}, \boldsymbol{y}^*, \pi$ from which $\mathcal{A}_{\mathsf{blin}}$ outputs $(\left[\begin{smallmatrix}\boldsymbol{x}\\ \boldsymbol{O}\\ \boldsymbol{y}^*\end{smallmatrix}\right]_1, [\boldsymbol{L}]_1, [\boldsymbol{R}]_2)$ and π_{blin}. It holds that $\Pr[\mathsf{BadLR}_{1,(\{1\},\{1\})}=1] \leq \mathsf{Adv}^{\mathsf{QABLin}}_{\mathsf{snd}}(\mathcal{A}_{\mathsf{blin}})$. □

Efficiency. The size of the crs is $(6N^2 + 6N + 24)\mathbb{G}_1$ elements and $(6N^2 + 4N + 36)\mathbb{G}_2$ elements and computing it is dominated by the same number of group exponentiations in $\mathbb{G}_1, \mathbb{G}_2$ respectively; the prover is dominated by $6N^2 + 6N$ exponentiations in $\mathbb{G}_1$ and $6N^2 + 2N$ exponentiations in $\mathbb{G}_2$ and produces a proof of size $12\mathbb{G}_1 + 10\mathbb{G}_2$ group elements; verifying a proof requires 36 pairing operations. The size of the proof can be reduced to $10\mathbb{G}_1 + 8\mathbb{G}_2$ combining the linear argument with the one used by the hadamard quasi argument.

7 Applications

In this section we show how to use our delegation scheme to (1) get a NIZK argument for NP in the preprocessing model where the size of the proof is linear in the size of the NP witness and independent of the computation size, in spite of most NIZK constructions under standard assumptions; (2) a zk-SNARK with quantitatively weaker assumptions and (3) compact NIZK for NP with proof size proportional to the witness.

We will use Groth-Sahai proofs [GS08] and, for completeness, we give a high level overview.

7.1 Groth-Sahai Proofs

The Groth Sahai (GS) proof system is a non-interactive witness indistinguishable proof system (and in some cases also zero-knowledge) for the language of quadratic equations over a bilinear group. The admissible equation types must be in the following form:

$$\sum_{j=1}^{m_y} f(\alpha_j, \mathsf{y}_j) + \sum_{i=1}^{m_x} f(\mathsf{x}_i, \beta_i) + \sum_{i=1}^{m_x}\sum_{j=1}^{m_y} f(\mathsf{x}_i, \gamma_{i,j}\mathsf{y}_j) = t, \tag{6}$$

where $\boldsymbol{\alpha} \in \mathbb{M}_1^{m_y}$, $\boldsymbol{\beta} \in \mathbb{M}_2^{m_x}$, $\boldsymbol{\Gamma} = (\gamma_{i,j}) \in \mathbb{Z}_q^{m_x \times m_y}$, $t \in \mathbb{M}_T$, and $\mathbb{M}_1, \mathbb{M}_2, \mathbb{M}_T \in \{\mathbb{Z}_q, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T\}$ are equipped with some bilinear map $f : \mathbb{M}_1 \times \mathbb{M}_2 \to \mathbb{M}_T$. The proof system is also zero-knowledge whenever $\mathbb{M}_1 \neq \mathbb{G}_1$ or $\mathbb{M}_2 \neq \mathbb{G}_2$ or $t = 0$ [EG14]. We will use only equations for which $t = 0$.

The GS proof system is a *commit-and-prove* proof system. That is, the prover first commits to solutions of equation 6 using Groth-Sahai commitments[16], and then computes a proof that the committed values satisfies equation 6. We denote an instance of the Groth-Sahai proof system by $\mathsf{GS} = (\mathsf{Setup}_{\mathsf{pb}}, \mathsf{Setup}_{\mathsf{ph}}, \mathsf{P}, \mathsf{V})$.

GS proofs are perfectly sound when the CRS is sampled from the perfectly binding distribution, i.e $\mathsf{crs}_{\mathsf{GS}} \leftarrow \mathsf{GS.Setup}_{\mathsf{pb}}(gk)$. This means that any π such that $\mathsf{GS.V}(\mathsf{crs}_{\mathsf{GS}}, \text{equation } 6, \pi) = 1$ contains commitments from which one can extract solutions to equation 6 with probability 1. Proofs are perfectly witness-indistinguishable when sampled from the perfectly hiding distribution, i.e. $\mathsf{crs}_{\mathsf{GS}} \leftarrow \mathsf{GS.Setup}_{\mathsf{ph}}(gk)$. That is, for any two solution to equation 6 the proofs follow exactly the same distribution, Computational indistinguishability of $\mathsf{GS.Setup}_{\mathsf{pb}}$ and $\mathsf{GS.Setup}_{\mathsf{ph}}$ implies that either the proof system is perfectly sound and computationally witness indistinguishable or computationally sound and perfect witness-indistinguishable.

7.2 NIZK arguments for NP.

Let CS_E an be algebraic commitment scheme –namely compatible with the Groth-Sahai proof system [GS08]– which is hiding and extractable. Also note that we can express the verification algorithm $\mathsf{Del.Verify}$ as a set of pairing product equation. The idea to construct a NIZK is the following: let C be an arithmetic circuit that takes public input $\boldsymbol{x}$ and secret input $\boldsymbol{w}$ the secret input, and let $\mathsf{crs}_{\mathsf{Del}}$ be a crs for the delegation of computation of C. The prover commits to $\boldsymbol{w}$ and the group elements defining the proof of the delegation using the extractable commitment and gives a Groth-Sahai proof that the set of verification equations are satisfied w.r.t. the opening of the commitment. Now, if CS_E is extractable, we can extract the witness $\boldsymbol{w}$, and if the circuit is not satisfied w.r.t. $\boldsymbol{x}, \boldsymbol{w}$ we can break adaptive soundness of delegation scheme Del. We present the scheme.

[16] For elements of $\mathbb{Z}_p$, a Groth-Sahai commitment is just an SSB commitment wiht locality parameter 1.

$\mathsf{Setup}(gk, C)$**:** Let C an arithmetic circuit which on public input $\boldsymbol{x}$ size n_x and secret input $\boldsymbol{w}$ size n_w outputs $\boldsymbol{y}$ of size n_d.

- $ck_w \leftarrow \mathsf{CS}_E(gk, n_w)$; $\mathsf{crs}_{\mathsf{Del}} \leftarrow \mathsf{Del.Setup}(gk, C)$.
- $\mathsf{crs}_{\mathsf{GS}} \leftarrow \mathsf{GS.Setup}_{\mathsf{pb}}(gk)$.
- Output $\mathsf{crs} = (ck_w, \mathsf{crs}_{\mathsf{Del}}, \mathsf{crs}_{\mathsf{GS}})$.

$\mathsf{Prove}(\mathsf{crs}, \boldsymbol{w}, \boldsymbol{x}, \boldsymbol{y})$**:**

- Parse $\mathsf{crs} = (ck_w, \mathsf{crs}_{\mathsf{Del}}, \mathsf{crs}_{\mathsf{GS}})$.
- Compute $\pi \leftarrow \mathsf{Del}(\mathsf{crs}_{\mathsf{Del}}, (\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{y})$ and $c_w = \mathsf{CS}_E.\mathsf{Com}(\boldsymbol{w}; \boldsymbol{r})$.
- Denote ϕ_{GS} the system of pairing product equations that contain
  1. The equations defined by $\mathsf{Del.V}(\mathsf{crs}, (\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{y}, \pi) = 1$, where the unknowns are $\boldsymbol{w}$ and π.
  2. The equations defined by $c_w = \mathsf{CS}_E.\mathsf{Com}(ck_w, \boldsymbol{w}; \boldsymbol{r})$, where the unknowns are $\boldsymbol{w}$ and $\boldsymbol{r}$.
- $\pi_{\mathsf{GS}} \leftarrow \mathsf{GS.P}(\mathsf{crs}_{\mathsf{GS}}, \phi_{GS}, (\boldsymbol{w}, \boldsymbol{r}))$
- Output $\pi \leftarrow (c_w, \pi_{\mathsf{GS}})$.

$\mathsf{Verify}(\mathsf{crs}, (\boldsymbol{x}, \boldsymbol{y}), \pi)$**:**

- Parse $\mathsf{crs} = (ck_w, \mathsf{crs}_{\mathsf{Del}}, \mathsf{crs}_{\mathsf{GS}})$. and $\pi = (c_w, \pi_{\mathsf{GS}})$.
- Output 1 iff $\mathsf{GS.V}(\mathsf{crs}_{\mathsf{GS}}, \phi_{\mathsf{GS}}, \pi_{\mathsf{GS}}) = 1$

Figure 9: NIZK argument of NP. CS_E is an algebraic commitment, GS is the Groth-Sahai proof system of [GS08] and Del the delegation scheme of Fig. 8.

Theorem 12. *Let CS_E be an algebraic commitment scheme that is hiding and extractable, GS the Groth-Sahai proof system of [GS08] and Del the delegation scheme of Fig. 8. Then, construction of Fig. 9 is a NIZK argument of knowledge. Furthermore, for every adversary $\mathcal{A}$ against knowledge soundness there exist adversaries $\mathcal{B}_1, \mathcal{B}_2$ against extractability of CS_E and against soundness of Del respectively such that $\mathsf{Adv}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{CS}_E}_{ext}(\mathcal{B}_1) + \mathsf{Adv}^{Del}_{snd}(\mathcal{B}_2)$.*

Proof. Completeness follows by the correctness of CS_E, and completeness of GS, Del. Computational zero knowledge follows from the computational zero-knowledge of GS and the hiding property of CS_E. For knowledge soundness, we show how we can extract a valid witness given an accepting proof. In what follows, let $\mathcal{E}_{\mathsf{CS}}$ be the extractors for CS_E. The NIZK extractor $\mathcal{E}_{\mathcal{A}}(\mathsf{crs}, \boldsymbol{x}, \boldsymbol{y}, \pi = (c_w, \pi_{\mathsf{GS}}))$ simply outputs $(\boldsymbol{w}, \pi) \leftarrow \mathcal{E}_{\mathsf{CS}}(ck_w, c_w)$. Now, we claim that this a valid witness except with negligible probability. It is enough to note that if it is not, there are three possible cases:

1. The extractor $\mathcal{E}_{\mathsf{CS}}$ failed which contradicts extractability of CS_E.
2. The extracted solutions $\boldsymbol{w}, \pi, \boldsymbol{r}$ are not solutions to ϕ_{GS}, contradicting perfect soundness of GS since the proof verifies.
3. $\boldsymbol{y} \neq C(\boldsymbol{x}||\boldsymbol{w})$. We can extract the solution $\boldsymbol{w}, \pi, \boldsymbol{r}$ and it must hold that $\mathsf{Del.Verify}(\mathsf{crs}, (\boldsymbol{x}, \boldsymbol{w}), \boldsymbol{y}, \pi) = 1$ contradicting adaptive soundness of Del.

□

As for efficiency, and specifically proof size, noting that the Groth-Sahai proof gives only a constant, multiplicative overhead to the proof –which is constant –, its size is dominated by the size of CS_E. Depending on the choice of CS_E we can get qualitatively different constructions. We discuss the following cases:

(i) For a NIZK argument of knowledge under falsifiable assumptions, we can extend our result to apply to boolean circuits instead of arithmetic ones by arithmetizing the different types of gates e.g. as in[DFGK14]. We can then use commitments for boolean vectors that are extractable in the field under falsifiable assumptions such as Groth-Sahai commitments or using methods of [GHR15a]. The proof size in this case is $\mathcal{O}(\lambda|\boldsymbol{w}|)$ where $\boldsymbol{w}$ is the secret input. Since fully succinct algebraic extractable commitments that allow extraction in the field are unknown to exist under falsifiable assumptions, we cannot achieve a (concretely more efficient) NIZK AoK for arithmetic circuits.

(ii) We use succinct extractable commitments based on knowledge assumptions, yielding a SNARK of constant proof size. Additionally, since the committed value is the secret input and not the full wire assignment we get a quantitatively smaller assumption size. For example, in case of q-power knowledge of exponent assumption (q-KEA) used in [DFGK14], we use only the n_w-KEA while [DFGK14] requires the larger (and hence stronger) $|C|$-KEA.

(iii) To construct a compact NIZK where the proof size is $O(|\boldsymbol{w}|) + \mathsf{poly}(\kappa)$ we follow essentially the ideas of [KNYY19, KNYY20]. We use a secret key symmetric encryption scheme $\mathsf{SE} = (\mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec})$ with additive overhead in the cyphertexts. That is, $|\mathsf{SE.Enc}(sk, \boldsymbol{w})| = O(|\boldsymbol{w}|) + \mathsf{poly}(\kappa)$. We use the NIZK from figure 9, instantiated with the commitment scheme from (i), for showing knowledge of some $K \in \mathrm{Im}(\mathsf{SE.KGen})$ such that $C'(K, D) = 1$, where K is the secret input, D the public input, and $C'(K, D) = C(\mathsf{SE.Dec}(K, D))$. To prove that $C(\boldsymbol{w}) = 1$ the prover picks $K \leftarrow \mathsf{SE.KGen}(1^\kappa)$ and computes $D \leftarrow \mathsf{SE.Enc}(K, \boldsymbol{w})$ together with a proof π that $C'(K, D) = 1$. The verifier on input crs, D and π ouputs 1 if π is a valid proof for D. In spite of [KNYY19, KNYY20] and by the nature of the underlying non-compact NIZK scheme we use, we don't require $\mathsf{SE.Dec}$ to be in NC^1.

References

[AFG+16] Masayuki Abe, Georg Fuchsbauer, Jens Groth, Kristiyan Haralambiev, and Miyako Ohkubo. Structure-preserving signatures and commitments to group elements. *Journal of Cryptology*, 29(2):363–421, April 2016. 4, 7, 22

[BCG+14] Eli Ben-Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy*, pages 459–474. IEEE Computer Society Press, May 2014. 4

[BDH+19] Michael Backes, Nico Döttling, Lucjan Hanzlik, Kamil Kluczniak, and Jonas Schneider. Ring signatures: Logarithmic-size, no setup - from standard assumptions. In Yuval Ishai and Vincent Rijmen, editors, *EUROCRYPT 2019, Part III*, volume 11478 of *LNCS*, pages 281–311. Springer, Heidelberg, May 2019. 8

[CCH+19] Ran Canetti, Yilei Chen, Justin Holmgren, Alex Lombardi, Guy N. Rothblum, Ron D. Rothblum, and Daniel Wichs. Fiat-Shamir: from practice to theory. In Moses Charikar and Edith Cohen, editors, *51st ACM STOC*, pages 1082–1090. ACM Press, June 2019. 3

[CFQ19] Matteo Campanelli, Dario Fiore, and Anaïs Querol. LegoSNARK: Modular design and composition of succinct zero-knowledge proofs. In Lorenzo Cavallaro, Johannes Kinder, XiaoFeng Wang, and Jonathan Katz, editors, *ACM CCS 2019*, pages 2075–2092. ACM Press, November 2019. 10

[CS02] Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, Heidelberg, April / May 2002. 10

[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. 13, 48

[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. 4, 5

[DNS98] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *30th ACM STOC*, pages 409–418. ACM Press, May 1998. 3

[EG14] Alex Escala and Jens Groth. Fine-tuning Groth-Sahai proofs. In Hugo Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 630–649. Springer, Heidelberg, March 2014. 46

[FLPS20] Prastudy Fauzi, Helger Lipmaa, Zaira Pindado, and Janno Siim. Somewhere statistically binding commitment schemes with applications. Cryptology ePrint Archive, Report 2020/652, 2020. `https://eprint.iacr.org/2020/652`. 4, 5, 8, 9, 16, 20, 21

[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. 3, 4, 5, 13

[GHR15a] 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. 4, 5, 6, 8, 9, 11, 12, 13, 15, 20, 22, 32, 34, 48

[GHR15b] Alonso González, Alejandro Hevia, and Carla Ràfols. QA-NIZK arguments in asymmetric groups: New tools and new constructions. Cryptology ePrint Archive, Report 2015/910, 2015. `http://eprint.iacr.org/2015/910`. 23

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. Delegating computation: interactive proofs for muggles. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 113–122. ACM Press, May 2008. 3

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Perfect non-interactive zero knowledge for NP. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 339–358. Springer, Heidelberg, May / June 2006. 5

[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. 5, 8

[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. 3, 4, 5, 7, 11, 12, 15, 25, 26, 31, 41, 53, 54

[Gro16] Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 305–326. Springer, Heidelberg, May 2016. 3, 4, 5, 13

[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. 13, 45, 46, 47

[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. 3

[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. 8, 16

[JKKZ20] Ruta Jawale, Yael Tauman Kalai, Dakshita Khurana, and Rachel Zhang. Snargs for bounded depth computations and ppad hardness from sub-exponential lwe. Cryptology ePrint Archive, Report 2020/980, 2020. `https://eprint.iacr.org/2020/980`. 3

[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. 4, 5, 10, 25, 26

[JR14] Charanjit S. Jutla and Arnab Roy. Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 295–312. Springer, Heidelberg, August 2014. 4, 5, 10

[KNYY19] Shuichi Katsumata, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. Exploring constructions of compact NIZKs from various assumptions. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part III*, volume 11694 of *LNCS*, pages 639–669. Springer, Heidelberg, August 2019. 5, 13, 48

[KNYY20] Shuichi Katsumata, Ryo Nishimaki, Shota Yamada, and Takashi Yamakawa. Compact NIZKs from standard assumptions on bilinear maps. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 379–409. Springer, Heidelberg, May 2020. 5, 13, 48

[KPY18] Yael Kalai, Omer Paneth, and Lisa Yang. On publicly verifiable delegation from standard assumptions. Cryptology ePrint Archive, Report 2018/776, 2018. `https://eprint.iacr.org/2018/776`. 3

[KPY19] Yael Tauman Kalai, Omer Paneth, and Lisa Yang. How to delegate computations publicly. In Moses Charikar and Edith Cohen, editors, *51st ACM STOC*, pages 1115–1124. ACM Press, June 2019. 2, 3, 4, 5, 6, 11, 16, 17, 25, 26

[KRR14] Yael Tauman Kalai, Ran Raz, and Ron D. Rothblum. How to delegate computations: the power of no-signaling proofs. In David B. Shmoys, editor, *46th ACM STOC*, pages 485–494. ACM Press, May / June 2014. 3

[KW15] Eike Kiltz and Hoeteck Wee. Quasi-adaptive NIZK for linear subspaces revisited. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 101–128. Springer, Heidelberg, April 2015. 4, 5, 10, 15, 29, 30, 31, 53

[LPJY13] Benoît Libert, Thomas Peters, Marc Joye, and Moti Yung. Linearly homomorphic structure-preserving signatures and their applications. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 289–307. Springer, Heidelberg, August 2013. 4, 5

[LPJY14] Benoît Libert, Thomas Peters, Marc Joye, and Moti Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 514–532. Springer, Heidelberg, May 2014. 53

[MRV16] Paz Morillo, Carla Ràfols, and Jorge Luis Villar. The kernel matrix Diffie-Hellman assumption. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part I*, volume 10031 of *LNCS*, pages 729–758. Springer, Heidelberg, December 2016. 10, 11, 14

[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. 8

[PR17] Omer Paneth and Guy N. Rothblum. On zero-testable homomorphic encryption and publicly verifiable non-interactive arguments. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part II*, volume 10678 of *LNCS*, pages 283–315. Springer, Heidelberg, November 2017. 4, 16, 17

[RRR16] Omer Reingold, Guy N. Rothblum, and Ron D. Rothblum. Constant-round interactive proofs for delegating computation. In Daniel Wichs and Yishay Mansour, editors, *48th ACM STOC*, pages 49–62. ACM Press, June 2016. 3

[RS20] Carla Ràfols and Javier Silva. Qa-nizk arguments of same opening for bilateral commitments. Cryptology ePrint Archive, Report 2020/569, 2020. `https://eprint.iacr.org/2020/569`. 18

[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. 21, 39

A Delayed proof from Section 3.3

We use the following lemmas.

Lemma 3. *For any adversary $\mathcal{A}$ and for any $\mathbf{P} \in \mathbb{Z}_p^{\ell_3 \times n}$, let*

$$\epsilon_{\mathcal{A}} = \Pr\left[\begin{array}{c|l} \boldsymbol{d} \neq 0 & (\mathbf{M},\mathbf{N}) \leftarrow (\mathcal{M},\mathcal{N}); \mathsf{crs} \leftarrow \mathsf{K}(gk,[\mathbf{M}]_1,[\mathbf{N}]_2,[\mathbf{P}]_1); \\ \boldsymbol{\pi}+\boldsymbol{\theta} = \boldsymbol{d}^\top \mathbf{K}_3 & ([\boldsymbol{d}]_1,[\boldsymbol{\pi}]_1,[\boldsymbol{\theta}]_2) \leftarrow \mathcal{A}(\mathsf{crs},[\mathbf{M}]_1,[\mathbf{N}]_2,h(\mathbf{M},\mathbf{N}),\mathbf{P}) \end{array}\right].$$

Then, there exists a PPT adversary $\mathcal{B}$ such that $\epsilon_{\mathcal{A}} \leq \mathsf{Adv}_{(\mathcal{M}^\top,h)\text{-}\mathsf{MDDH}}(\mathcal{B}) + 1/p$, where $\mathcal{M}^\top$ is the distribution which results from sampling matrices from $\mathcal{M}$ and transposing them.

Proof. (Lemma 3)
We show this by a sequence of games.

Game_0**:** This game runs the adversary as in Lemma 3.

Game_1**:** This game is exactly as Game_0 but the crs is computed using algorithm K^*, as defined in Fig. 10, and the winning condition is $\boldsymbol{d} \neq 0$ and $\boldsymbol{\pi} = (\boldsymbol{d}^\top(\mathbf{C}_3 - \mathbf{K}_{3,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}, \boldsymbol{d}^\top \mathbf{K}_{3,2})$,

Game_2**:** This game is exactly as Game_1 but $\boldsymbol{s},\boldsymbol{t} \leftarrow \mathbb{Z}_p^n$.

We now prove some Lemmas which show that the games are indistinguishable. Lemmas 4 and 5 show that the adversary has essentially the same advantage of winning in any game. Lemma 6 says that the adversary has negligible probability of winning in Game_2. Lemma 3 follows from the composition of lemmas 4, 5 and 6.

□

Lemma 4. *For any (unbounded) algorithm $\mathcal{A}$ we have* $\Pr[\mathsf{Game}_1(\mathcal{A}) = 1] = \Pr[\mathsf{Game}_0(\mathcal{A}) = 1]$.

$\mathsf{K}^*(gk, [\mathbf{M}]_1, [\mathbf{N}]_2, [\mathbf{P}]_1)$:

- $\mathbf{C}_1 \leftarrow \mathbb{Z}_p^{\ell_1 \times k}$; $\mathbf{C}_3 \leftarrow \mathbb{Z}_p^{\ell_3 \times k}$; $\boldsymbol{\gamma} \leftarrow \mathbb{Z}_p^n$.
- $\mathbf{K}_{1,2} \leftarrow \mathbb{Z}_p^{\ell_1 \times 1}$; $\mathbf{K}_{3,2} \leftarrow \mathbb{Z}_p^{\ell_3 \times 1}$.
- Sample $\mathbf{A} = \begin{pmatrix} \overline{\mathbf{A}} \\ \underline{\mathbf{A}} \end{pmatrix} \leftarrow \mathcal{D}_k$; $\boldsymbol{\Gamma} \leftarrow \mathbb{Z}_p^{n \times \overline{k}}$. Here $\overline{\mathbf{A}}$ denotes the first k rows for $\mathbf{A}$ and $\underline{\mathbf{A}}$ the last row.
- $\mathbf{K}_{1,1} = (\mathbf{C}_1 - \mathbf{K}_{1,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}$; $\mathbf{K}_{3,1} = (\mathbf{C}_3 - \mathbf{K}_{3,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}$;
- $[\boldsymbol{s}]_1 \leftarrow [\mathbf{M}^\top]_1\mathbf{K}_{1,2} - [\boldsymbol{\gamma}]_1$;
  $[\boldsymbol{t}]_2 \leftarrow [\mathbf{N}^\top]_1\mathbf{K}_{1,2} + [\boldsymbol{\gamma}]_1$.
- $[\mathbf{B}]_1 = \left[\left(\mathbf{M}^\top\mathbf{K}_{1,1} + \mathbf{P}^\top\mathbf{K}_{3,1}, \boldsymbol{s} + \mathbf{P}^\top\mathbf{K}_{3,2}\right) + \boldsymbol{\Gamma}\right]_1$;
  $[\mathbf{D}]_2 = \left[\left(\mathbf{N}^\top\mathbf{K}_{1,1}, \boldsymbol{t}\right) - \boldsymbol{\Gamma}\right]_2$;
- Output $\mathsf{crs} = (gk, [\mathbf{A}]_{1,2}, [\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{C}_1]_2, [\mathbf{C}_3]_2)$.

Figure 10: Modified crs generation algorithm used in Lemma 3.

Proof. If we define $\mathbf{K}_{1,1} = (\mathbf{C}_1 - \mathbf{K}_{1,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}$ and $\mathbf{K} = \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_3 \end{pmatrix} = \begin{pmatrix} \mathbf{K}_{1,1} & \mathbf{K}_{1,2} \\ \mathbf{K}_{3,1} & \mathbf{K}_{3,2} \end{pmatrix}$, we observe that the output of K^* is well formed and the winning condition is the same as in the previous game, since $\mathbf{B}, \mathbf{D}$ are uniform conditioned on their sum being equal to

$$\begin{aligned}
\mathbf{B} + \mathbf{D} &= \left((\mathbf{M}^\top + \mathbf{N}^\top)\mathbf{K}_{1,1} + \mathbf{P}^\top\mathbf{K}_{3,1}, \boldsymbol{s} + \mathbf{P}^\top\mathbf{K}_{3,2}\right) + \left(\mathbf{R}^\top\mathbf{K}_{1,1}, \boldsymbol{t}\right) + \boldsymbol{\Gamma} - \boldsymbol{\Gamma} \\
&= \left((\mathbf{M}^\top + \mathbf{N}^\top)\mathbf{K}_{1,1} + \mathbf{P}^\top\mathbf{K}_{3,1}, (\mathbf{M}^\top + \mathbf{N}^\top)\mathbf{K}_{1,2} + \mathbf{P}^\top\mathbf{K}_{3,2}\right) \\
&= (\mathbf{M}^\top + \mathbf{N}^\top)\begin{pmatrix} \mathbf{K}_{1,1} \\ \mathbf{K}_{1,2} \end{pmatrix} + \mathbf{P}^\top\begin{pmatrix} \mathbf{K}_{3,1} \\ \mathbf{K}_{3,2} \end{pmatrix} = (\mathbf{M}^\top + \mathbf{N}^\top \mid \mathbf{P}^\top)\mathbf{K},
\end{aligned}$$

$$\mathbf{K}\mathbf{A} = \begin{pmatrix} (\mathbf{C}_1 - \mathbf{K}_{1,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1} & \mathbf{K}_{1,2} \\ (\mathbf{C}_3 - \mathbf{K}_{3,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1} & \mathbf{K}_{3,2} \end{pmatrix}\begin{pmatrix} \overline{\mathbf{A}} \\ \underline{\mathbf{A}} \end{pmatrix} = \begin{pmatrix} \mathbf{C}_1 - \mathbf{K}_{1,2}\underline{\mathbf{A}} + \mathbf{K}_{1,2}\underline{\mathbf{A}} \\ \mathbf{C}_3 - \mathbf{K}_{3,1}\underline{\mathbf{A}} + \mathbf{K}_{3,2}\underline{\mathbf{A}} \end{pmatrix} = \mathbf{C},$$

and by definition $\boldsymbol{\pi} + \boldsymbol{\theta} = (\boldsymbol{d}^\top(\mathbf{C}_3 - \mathbf{K}_{3,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}, \boldsymbol{d}^\top\mathbf{K}_{3,2}) = (\boldsymbol{d}^\top\mathbf{K}_{3,1}, \boldsymbol{d}^\top\mathbf{K}_{3,2}) = \boldsymbol{d}^\top\mathbf{K}_3$.

Therefore we just need to argue that the distribution of $\mathbf{K}$ is the same in both games. But this is an immediate consequence of the fact that for every value of $(\mathbf{C}, \mathbf{K}_{1,1}, \mathbf{K}_{3,1})$ there exists a unique value of $(\mathbf{K}_{1,2}, \mathbf{K}_{3,2})$ which is compatible with $\mathbf{C} = \mathbf{K}\mathbf{A}$. Indeed, $\mathbf{C} = \mathbf{K}\mathbf{A} \iff \mathbf{C}_i = \mathbf{K}_{i,1}\overline{\mathbf{A}} + \mathbf{K}_{i,2}\underline{\mathbf{A}},\ i = 1, 3 \iff (\mathbf{C}_i - \mathbf{K}_{i,2}\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1} = \mathbf{K}_{i,1},\ i = 1, 3$. □

Lemma 5. *For any PPT algorithm $\mathcal{A}$ there exists a PPT algorithm $\mathcal{B}$ such that*

$$\mathsf{Adv}_{\Pi_{kt\text{-}sum,h'}}(\mathcal{A}) \leq \mathsf{Adv}_{(\mathcal{M}^\top,\mathcal{N}^\top,h)\text{-}\mathsf{MDDH}}(\mathcal{B}).$$

Proof. We construct an adversary $\mathcal{B}$ that receives the challenge $([\mathbf{M}^\top]_1, [\mathbf{N}^\top]_2, [\boldsymbol{s}^*]_1, [\boldsymbol{t}^*]_2, h([\mathbf{M}^\top, \mathbf{N}^\top])$, where $\boldsymbol{s}^* + \boldsymbol{t}^* = (\mathbf{M}^\top + \mathbf{N}^\top)\boldsymbol{w}$, $\boldsymbol{w} \leftarrow \mathbb{Z}_p^{\ell_1}$, or $\boldsymbol{s}^*, \boldsymbol{t}^* \leftarrow \mathbb{Z}_p^n$. $\mathcal{B}$ computes the crs running $\mathsf{K}^*(gk, [\mathbf{M}]_1, [\mathbf{N}]_2, [\mathbf{P}]_1)$ but replaces $[\boldsymbol{s}]_1, [\boldsymbol{t}]_2$ with $[\boldsymbol{s}^*]_1, [\boldsymbol{t}^*]_2$ respectively, and then runs $\mathcal{A}$ as in game Game_1. Since Game_1 corresponds to the first case and Game_2 to the second, the lemma follows.

□

Lemma 6. *For any (unbounded) algorithm $\mathcal{A}$, $\Pr[\mathsf{Game}_2(\mathcal{A}) = 1] \leq 1/p$.*

Proof. We will show that, conditioned on $\mathbf{A}, \mathbf{C}, \mathbf{B} + \mathbf{D}, \mathbf{M} + \mathbf{N}, \mathbf{P}$, the matrix $\mathbf{K}_{3,2}$ is uniformly distributed. Since it holds that $(\mathbf{B} + \mathbf{D})\mathbf{A} = (\mathbf{M}^\top + \mathbf{N}^\top \mid \mathbf{P}^\top)\mathbf{C}$, we get that the first k columns of $\mathbf{B} + \mathbf{D}$, namely $\mathbf{B}_1 + \mathbf{D}_1$, are completely determined by the last columns $\mathbf{B}_2 + \mathbf{D}_2$. Indeed

$$(\mathbf{B}_1 + \mathbf{D}_1, \mathbf{B}_2 + \mathbf{D}_2)\mathbf{A} = (\mathbf{M}^\top + \mathbf{N}^\top \mid \mathbf{P}^\top)\mathbf{C} \iff \mathbf{B}_1 + \mathbf{D}_1 = ((\mathbf{M}^\top + \mathbf{N}^\top \mid \mathbf{P}^\top)\mathbf{C} - (\mathbf{B}_2 + \mathbf{D}_2)\underline{\mathbf{A}})\overline{\mathbf{A}}^{-1}.$$

Hence, conditioning in $\mathbf{A}, \mathbf{C}, \mathbf{B}_1 + \mathbf{D}_1, \mathbf{M} + \mathbf{N}, \mathbf{P}$ doesn't alter the probability. We have that $\mathbf{B}_2 + \mathbf{D}_2 = (\boldsymbol{s}+\boldsymbol{t}) + \mathbf{P}^\top \mathbf{K}_{3,2}$, which consists of n equations on $n+\ell_2$ variables. It follows that there are ℓ_2 free variables. Then $\mathbf{K}_{3,2}$ is uniformly distributed and hence completely hidden to the adversary.

Note that

$$\boldsymbol{\pi} + \boldsymbol{\theta} = \boldsymbol{d}^\top \mathbf{K}_3 \Longrightarrow \boldsymbol{\pi}_2 + \boldsymbol{\theta}_2 = \boldsymbol{d}^\top \mathbf{K}_{3,2},$$

where $\boldsymbol{\pi}_2, \boldsymbol{\theta}_2$ are the last element of $\boldsymbol{\pi}$, $\boldsymbol{\theta}$ respectively. Given that $\boldsymbol{d} \neq 0$, the last equation only holds with probability $1/p$ and so $\mathcal{A}$'s probability of winning.

□

The knowledge transfer property is a direct consequence of Lemma 3. We present the proof next.

Theorem 13. *For any adversary $\mathcal{A}$ against the soundness of $\Pi_{\mathsf{kt-sum}}$ with respect to $\mathcal{L}^{\mathsf{no}}_{\mathsf{sum}}$, there exist adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that*

$$\mathsf{Adv}_{\mathsf{kt\text{-}sum}}(\mathcal{A}) \leq \mathsf{Adv}_{\mathcal{D}_k\text{-}\mathsf{SKerMDH}}(\mathcal{B}_1) + \mathsf{Adv}_{(\mathcal{M}^\top, \mathcal{N}^\top, h)\text{-}\mathsf{MDDH}} + 1/p.$$

Proof. Given an adversary that produces a valid proof for a statement in $\mathcal{L}^{\mathsf{no}}_{\mathsf{sum}}$, successful attacks can be divided in two categories.

Type I: In this attack $\boldsymbol{\pi} + \boldsymbol{\theta} \neq (\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top)\mathbf{K}_1 + \boldsymbol{d}^\top \mathbf{K}_3$.

Type II: In this type of attack $\boldsymbol{\pi} + \boldsymbol{\theta} = (\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top)\mathbf{K}_1 + \boldsymbol{d}^\top \mathbf{K}_3$.

Type I attacks are computationally infeasible when $\overline{k} = k+1$, as they can be used to construct an adversary $\mathcal{B}_1$ against the $\mathcal{D}_k$-$\mathsf{SKerMDH}$ assumption.[17] Adversary $\mathcal{B}_1$ receives a challenge $[\mathbf{A}]_{1,2}$ and then runs the soundness experiment for $\mathcal{A}$. When $\mathcal{A}$ outputs $([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1, [\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2)$, $\mathcal{B}_1$ outputs $[\boldsymbol{\pi}^\dagger]_1 = [\boldsymbol{\pi}]_1 - [\boldsymbol{c}_1^\top]_1\mathbf{K}_1 - [\boldsymbol{d}^\top]_1\mathbf{K}_3$, $[\boldsymbol{\theta}^\dagger]_2 = [\boldsymbol{\theta}]_1 - [\boldsymbol{c}_2^\top]_1\mathbf{K}_1$ where it holds that $\boldsymbol{\pi} + \boldsymbol{\theta} \neq (\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top)\mathbf{K}_1 + \boldsymbol{d}^\top \mathbf{K}_3$. Since $[\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2$ is accepted by the verifier we get that $e([\boldsymbol{\pi}]_1, [\mathbf{A}]_2) + e([\boldsymbol{\theta}]_2, [\mathbf{A}]_1) = e([\boldsymbol{c}_1^\top]_1, [\mathbf{C}_1]_2) + e([\boldsymbol{c}_2^\top]_2, [\mathbf{C}_1]_1) + e([\boldsymbol{d}^\top]_1, [\mathbf{C}_3]_2)$ and then $(\boldsymbol{\pi}^\dagger + \boldsymbol{\theta}^\dagger)\mathbf{A} = (\boldsymbol{\pi} + \boldsymbol{\theta})\mathbf{A} - (\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top)\mathbf{K}_1\mathbf{A} - \boldsymbol{d}^\top \mathbf{K}_3\mathbf{A} = (\boldsymbol{\pi} + \boldsymbol{\theta})\mathbf{A} - (\boldsymbol{c}_1 + \boldsymbol{c}_2)^\top \mathbf{C}_1 - \boldsymbol{d}^\top \mathbf{C}_3 = 0$. We conclude that the success probability of a type I attack is bounded by $\mathsf{Adv}_{\mathcal{D}_k\text{-}\mathsf{SKerMDH}}(\mathcal{B}_1)$.

For type II attacks, since $[\boldsymbol{\pi}]_1 = [\boldsymbol{c}_1^\top]_1\mathbf{K}_1 + [\boldsymbol{d}^\top]_1\mathbf{K}_3, [\boldsymbol{\theta}]_2 = [\boldsymbol{c}_2^\top]_2\mathbf{K}_1$ is a valid proof for $\begin{pmatrix} [\boldsymbol{c}_1]_1 \\ [\boldsymbol{c}_2]_2 \\ [\boldsymbol{d}]_1 \end{pmatrix}$, then, by linearity of the verification equations $\boldsymbol{\pi}^\dagger = \boldsymbol{\pi} - \boldsymbol{w}^\top \mathbf{B}$ and $\boldsymbol{\theta}^\dagger = \boldsymbol{\theta} - \boldsymbol{w}^\top \mathbf{B}$ is a valid proof for $\begin{pmatrix} 0 \\ 0 \\ [\boldsymbol{d}^\dagger]_1 \end{pmatrix} = \begin{pmatrix} [\boldsymbol{c}_1]_1 - [\mathbf{M}]_1\boldsymbol{w} \\ [\boldsymbol{c}_2]_2 - [\mathbf{N}]_2\boldsymbol{w} \\ [\boldsymbol{d}]_1 - [\mathbf{P}]_1\boldsymbol{w} \end{pmatrix}$. Since $\boldsymbol{d} \neq \mathbf{N}\boldsymbol{w}$, we conclude that an attacker of type II can be turned into an attacker $\mathcal{B}_2$ for Lemma 3.

□

We next note that the argument of knowledge transfer remains secure even for matrix distribution that also include some zero columns.

Theorem 14. *Let $\mathcal{M}', \mathcal{N}', \mathcal{P}', \mathcal{Q}'$ be matrix distributions that sample $(\mathbf{M} \mid \mathbf{0}_{\ell_1 \times n'})$, $(\mathbf{N} \mid \mathbf{0}_{\ell_2 \times n'})$, $(\mathbf{P} \mid \mathbf{0}_{\ell_3 \times n'})$, $(\mathbf{Q} \mid \mathbf{0}_{\ell_4 \times n'})$ where $\mathbf{M} \leftarrow \mathcal{M}$, $\mathbf{N} \leftarrow \mathcal{N}$, $\mathbf{P} \leftarrow \mathcal{P}$, $\mathbf{Q} \leftarrow \mathcal{Q}$.*

1. *For any adversary $\mathcal{A}$ against the h-strong soundness of $\Pi_{\mathsf{kt\text{-}lin}}$ there exist adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that $\mathsf{Adv}_{\Pi_{\mathsf{kt\text{-}lin}, h'}}(\mathcal{A}) \leq \mathsf{Adv}_{\mathcal{D}_k\text{-}\mathsf{SKerMDH}}(\mathcal{B}_1) + \mathsf{Adv}_{(\mathcal{M}^\top, h)\text{-}\mathsf{MDDH}}(\mathcal{B}_2) + 1/p$, where $h'([\mathbf{M}]_1, [\mathbf{N}]_2, [\mathbf{P}]_1, [\mathbf{Q}]_2) = (h(\mathbf{M}), \mathbf{N}, \mathbf{P}, \mathbf{Q})$.*
2. *When $\ell_1 = \ell_2$, for any adversary $\mathcal{A}$ against the h-strong soundness of $\Pi_{\mathsf{kt\text{-}sum}}$ there exist adversaries $\mathcal{B}_1$ and $\mathcal{B}_2$ such that $\mathsf{Adv}_{\Pi_{\mathsf{kt\text{-}sum}, h'}}(\mathcal{A}) \leq \mathsf{Adv}_{\mathcal{D}_k\text{-}\mathsf{SKerMDH}}(\mathcal{B}_1) + \mathsf{Adv}_{(\mathcal{M}^\top, \mathcal{N}^\top, h)\text{-}\mathsf{MDDH}}(\mathcal{B}_2) + 1/p$, where $h'([\mathbf{M}]_1, [\mathbf{N}]_2, [\mathbf{P}]_1, [\mathbf{Q}]_2) = (h(\mathbf{M}, \mathbf{N}), \mathbf{P}, \mathbf{Q})$.*

The proof is implicitly shown in [GR19, Lemma 15]. Essentially one can reduce to the knowledge transfer argument where we delete the zero columns of the matrix and rely on the linearity properties of the proofs of construction of Fig. 1.

[17] This part of the proof follows essentially the same lines of the first constant-size QA-NIZK arguments for linear spaces of Libert et al.[LPJY14] which were later simplified and generalized by Kiltz and Wee [KW15].

B Delayed proofs from Section 5.2.1

B.1 Proof of security for the bilateral knowledge transfer quasi arguement

Theorem 15. *Let $\mathcal{M}, \mathcal{N}_1, \mathcal{N}_2$ be witness samplable distributions, $\mathcal{D}_k$ be a matrix distribution and* CS *an algebraic SSB commitment with perfect completeness. Also, let $\mathcal{A}$ be an adversary against h_{ls}-strong local knowledge soundness of construction* QABlin *of Fig. 5, where the index $h_{\mathsf{ls}}(\theta) = (\mathbf{G}, \mathbf{H}, \mathbf{F}, h(\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2))$. Then, completeness holds with probability 1 and h_{ls}-strong local knowledge soundness holds with probability at least $1 - \mathsf{Adv}^{\Pi_{\mathsf{kt\text{-}lin}}}_{\mathsf{snd}}(\mathcal{B}_{\boldsymbol{S}})$, where $\mathcal{B}_{\boldsymbol{S}}$ is any PPT adversary against h-strong soundness of $\Pi_{\mathsf{kt\text{-}lin}}$ and h giving the discrete logarithms of the last two matrices.*

Proof. For completeness, we have that

$$\begin{aligned}
(\boldsymbol{c}^\top \mid \boldsymbol{d}_1^\top)\mathbf{C}_1 + \boldsymbol{d}_2^\top \mathbf{C}_2 &= (\boldsymbol{c}^\top \mid \boldsymbol{d}_1^\top) \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \end{pmatrix} \mathbf{A} + \boldsymbol{d}_2^\top \mathbf{K}_2 \mathbf{A} \\
&= (\boldsymbol{c}^\top \mathbf{K}_1 + \boldsymbol{d}_1^\top \mathbf{K}_2 + \boldsymbol{d}_2^\top \mathbf{K}_2)\mathbf{A} \\
&= \left(\boldsymbol{w}^\top \mathbf{M}^\top \mathbf{G}^\top \mathbf{K}_1 + \boldsymbol{w}^\top \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{K}_2 + \boldsymbol{w}^\top \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{K}_2\right) \mathbf{A} \\
&= \left(\boldsymbol{w}^\top (\mathbf{M}^\top \mathbf{G}^\top \mathbf{K}_1 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{K}_2) + \boldsymbol{w}^\top \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{K}_2\right) \mathbf{A} \\
&= \boldsymbol{w}^\top \mathbf{B}\mathbf{A} + \boldsymbol{w}^\top \mathbf{D}\mathbf{A} \\
&= \boldsymbol{\pi}\mathbf{A} + \boldsymbol{\theta}\mathbf{A}
\end{aligned}$$

Local Extractability follows using almost an identical argument to Thm. 7 and reducing to knowledge transfer of linear KTA Argument of [GR19] presented in Fig. 1. Given an adversary $\mathcal{A}$ breaking h_{ls}-Strong local knowledge soundness of QABlin we construct another adversary $\mathcal{B}_{\boldsymbol{S}}$ that breaks h-strong soundness of the argument $\Pi_{\mathsf{kt\text{-}lin}}$ for matrices $[\mathbf{M}_{S_1}]_1$, $[\mathbf{N}_{1,S_2}]_1$ and $[\mathbf{N}_{2,S_2}]_2$. $\mathcal{B}_{\boldsymbol{S}}$ works as follows: it takes input $(\rho^\dagger, h(\theta^\dagger), \mathsf{crs}^\dagger)$ where

$$\rho^\dagger := (gk, [\mathbf{M}_{S_1}]_1, [\mathbf{N}_{1,S_2}]_1, [\mathbf{N}_{2,S_2}]_2), \quad h(\theta^\dagger) := (\mathbf{N}_{1,S_2}, \mathbf{N}_{2,S_2}), \quad \mathsf{crs}^\dagger := ([\mathbf{B}^\dagger]_1, [\mathbf{D}^\dagger]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1^\dagger]_2, [\mathbf{C}_2^\dagger]_1)$$

and does the following:

- $([\mathbf{G}]_1, \mathbf{G}, \mathbf{T}_{\mathbf{G}}) \leftarrow \mathsf{CS.KGen}(gk_1, d, K_1, S_1)$.
- $([\mathbf{H}]_1, \mathbf{H}, \mathbf{T}_{\mathbf{H}}) \leftarrow \mathsf{CS.KGen}(gk_1, d, K_2, S_2)$.
- $([\mathbf{F}]_2, \mathbf{F}, \mathbf{T}_{\mathbf{F}}) \leftarrow \mathsf{CS.KGen}(gk_2, d, K_2, S_2)$.
- It samples $\mathbf{M}_{\overline{S}_1}, \mathbf{N}_{1,\overline{S}_2}, \mathbf{N}_{2,\overline{S}_2}$, such that $\mathbf{M} = \mathbf{P}_{S_1} \begin{pmatrix} \mathbf{M}_{S_1} \\ \mathbf{M}_{\overline{S}_1} \end{pmatrix}$, $\mathbf{N}_1 = \mathbf{P}_{S_2} \begin{pmatrix} \mathbf{N}_{1,S_2} \\ \mathbf{N}_{1,\overline{S}_2} \end{pmatrix}$, $\mathbf{N}_2 = \mathbf{P}_{S_2} \begin{pmatrix} \mathbf{N}_{2,S_2} \\ \mathbf{N}_{2,\overline{S}_2} \end{pmatrix}$.
- $\mathbf{R}_0 \leftarrow \mathbb{Z}_p^{\overline{K}_0 \times k}$; $\mathbf{R}_1 \leftarrow \mathbb{Z}_p^{\overline{K}_1 \times k}$; $\mathbf{R}_2 \leftarrow \mathbb{Z}_p^{\overline{K}_2 \times k}$.
- It computes $[\mathbf{B}]_1 := [\mathbf{B}^\dagger]_1 + [\mathbf{M}]_1^\top \mathbf{G}^\top \mathbf{R}_0 + [\mathbf{N}_1]_1^\top \mathbf{H}^\top \mathbf{R}_1$ and $[\mathbf{D}]_2 := [\mathbf{D}^\dagger]_2 + [\mathbf{N}_2]_2^\top \mathbf{F}^\top \mathbf{R}_2$
- It computes $[\mathbf{C}_1]_2 := \begin{pmatrix} \mathbf{T}_{\mathbf{G}} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{\mathbf{H}} \end{pmatrix} [\mathbf{C}_1^\dagger]_2 + \begin{pmatrix} \mathbf{R}_0 \\ \mathbf{R}_1 \end{pmatrix} [\mathbf{A}]_2$ and $[\mathbf{C}_2]_1 := \mathbf{T}_{\mathbf{F}} [\mathbf{C}_2^\dagger]_1 + \mathbf{R}_2 [\mathbf{A}]_1$.
- It sets

$$\rho := ([\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2, [\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2), \quad h_{\mathsf{ls}}(\theta) := (\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N}_1, \mathbf{N}_2)$$
$$\mathsf{crs} := ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1)$$

It then executes $\mathcal{A}(\rho, h_{\mathsf{ls}}(\theta), \mathsf{crs})$ until it outputs a statement $([\boldsymbol{c}]_1, [\boldsymbol{d}_1]_1, [\boldsymbol{d}_2]_2, \boldsymbol{w})$ together with an accepting proof $[\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2$. Given an accepting proof $\mathcal{B}_{\boldsymbol{S}}$ sets $[\boldsymbol{x}^\dagger]_1 = \mathbf{T}_{\mathbf{G}}[\boldsymbol{c}]_1, [\boldsymbol{y}_1^\dagger]_1 = \mathbf{T}_{\mathbf{H}}[\boldsymbol{d}_1]_1, [\boldsymbol{y}_2^\dagger]_2 = \mathbf{T}_{\mathbf{F}}[\boldsymbol{d}_2]_2$, $[\boldsymbol{\pi}^\dagger]_1 = [\boldsymbol{\pi}]_1 - [\boldsymbol{c}]_1^\top \mathbf{R}_0 - [\boldsymbol{d}_1]_1^\top \mathbf{R}_1$ and $[\boldsymbol{\theta}^\dagger]_2 = [\boldsymbol{\theta}]_1 - [\boldsymbol{d}_2]_2^\top \mathbf{R}_2$. It outputs $\Big(([\boldsymbol{x}^\dagger]_1, [\boldsymbol{y}_1^\dagger]_1, [\boldsymbol{y}_2^\dagger]_2), \boldsymbol{w}, ([\boldsymbol{\pi}^\dagger]_1, [\boldsymbol{\theta}^\dagger]_2)\Big)$.

Note that the commitment keys are perfectly binding at S. First, we claim that in this case the values $\rho, h_{\mathsf{ls}}(\theta), \mathsf{crs}$ output by $\mathcal{B}_{\boldsymbol{S}}$ are identically distributed to honestly computed ones and thus do not skew the probability that $\mathcal{A}$ outputs a valid proof. For $\rho, h_{\mathsf{ls}}(\theta)$, this is immediate by the witness samplability of the distributions $\mathcal{M}$, $\mathcal{N}_1$, $\mathcal{N}_2$. We show that this holds for crs as well.

Let $\mathbf{K}_0^\dagger \in \mathbb{Z}_p^{|S_1|\times k}, \mathbf{K}_1^\dagger \in \mathbb{Z}_p^{|S_2|\times k}, \mathbf{K}_2^\dagger \in \mathbb{Z}_p^{|S_2|\times k}$ be the implicit values used to compute $\mathsf{crs}^\dagger$, that is, they satisfy

$$\mathbf{B}^\dagger = \mathbf{M}_S^\top \mathbf{K}_0^\dagger + \mathbf{N}_{1,S}^\top \mathbf{K}_1^\dagger,\ \mathbf{D}^\dagger = \mathbf{N}_{2,S}^\top \mathbf{K}_2^\dagger,\ \mathbf{C}_1^\dagger = \begin{pmatrix} \mathbf{K}_0^\dagger \\ \mathbf{K}_1^\dagger \end{pmatrix} \mathbf{A} \text{ and } \mathbf{C}_2^\dagger = \mathbf{K}_2^\dagger \mathbf{A}.$$

Now $\mathcal{B}_{\boldsymbol{S}}$ implicitly defines $\mathbf{K}_2 = \mathbf{T}_{\mathbf{G}}\mathbf{K}_0^\dagger + \mathbf{R}_0$, $\mathbf{K}_2 = \mathbf{T}_{\mathbf{H}}\mathbf{K}_1^\dagger + \mathbf{R}_1$, $\mathbf{K}_2 = \mathbf{T}_{\mathbf{F}}\mathbf{K}_2^\dagger + \mathbf{R}_2$. First, note that these matrices are uniformly distributed since $\mathbf{R}_0, \mathbf{R}_1, \mathbf{R}_2$ are uniformly distributed. Thus $\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_2$ are distributed identically to honestly generated values for generating a crs. We claim that the crs output by $\mathcal{A}$ is identically distributed to sampling this matrix and computing the other values honestly. Indeed we have that

$$\begin{aligned}
\mathbf{B} &= \mathbf{B}^\dagger + \mathbf{M}^\top \mathbf{G}^\top \mathbf{R}_0 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}_{S_1}^\top \mathbf{K}_1^\dagger + \mathbf{N}_{1,S_2}^\top \mathbf{K}_2^\dagger + \mathbf{M}^\top \mathbf{G}^\top \mathbf{R}_0 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}^\top \mathbf{G}^\top \mathbf{T}_{\mathbf{G}} \mathbf{K}_1^\dagger + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{T}_{\mathbf{H}}^\top \mathbf{K}_2^\dagger + \mathbf{M}^\top \mathbf{G}^\top \mathbf{R}_0 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}^\top \mathbf{G}^\top (\mathbf{T}_{\mathbf{G}} \mathbf{K}_1^\dagger + \mathbf{R}_0) + \mathbf{N}_1^\top \mathbf{H}^\top (\mathbf{T}_{\mathbf{H}}^\top \mathbf{K}_2^\dagger + \mathbf{R}_1) \\
&= \mathbf{M}^\top \mathbf{G}^\top \mathbf{K}_1 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{K}_2
\end{aligned}$$

where the third equality follows since by the local extractability of the SSBs we have that $\mathbf{T}_{\mathbf{G}}^\top \mathbf{G}\mathbf{M} = \mathbf{M}_{S_1}$, $\mathbf{T}_{\mathbf{H}}^\top \mathbf{H}\mathbf{N}_1 = \mathbf{N}_{1,S_2}$. Similarly, we have

$$\begin{aligned}
\mathbf{D} &= \mathbf{D}^\dagger + \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{R}_2 \\
&= \mathbf{N}_{2,S_2}^\top \mathbf{K}_3^\dagger + \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{R}_2 \\
&= \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{T}_{\mathbf{F}} \mathbf{K}_2^\dagger + \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{R}_2 \\
&= \mathbf{N}_2^\top \mathbf{F}^\top (\mathbf{T}_{\mathbf{F}} \mathbf{K}_2^\dagger + \mathbf{R}_2) \\
&= \mathbf{N}_2^\top \mathbf{F}^\top \mathbf{K}_2
\end{aligned}$$

Also, we have that

$$\mathbf{C}_1 = \begin{pmatrix} \mathbf{T}_{\mathbf{G}} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{\mathbf{H}} \end{pmatrix} \mathbf{C}_1^\dagger + \begin{pmatrix} \mathbf{R}_0 \\ \mathbf{R}_1 \end{pmatrix} \mathbf{A} = \begin{pmatrix} \mathbf{T}_{\mathbf{G}} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_{\mathbf{H}} \end{pmatrix} \begin{pmatrix} \mathbf{K}_1^\dagger \\ \mathbf{K}_2^\dagger \end{pmatrix} \mathbf{A} + \begin{pmatrix} \mathbf{R}_0 \\ \mathbf{R}_1 \end{pmatrix} \mathbf{A} = \begin{pmatrix} \mathbf{T}_{\mathbf{G}} \mathbf{K}_1^\dagger + \mathbf{R}_0 \\ \mathbf{T}_{\mathbf{H}} \mathbf{K}_2^\dagger + \mathbf{R}_1 \end{pmatrix} \mathbf{A} = \begin{pmatrix} \mathbf{K}_1 \\ \mathbf{K}_2 \end{pmatrix} \mathbf{A}$$

$$\mathbf{C}_2 = \mathbf{T}_{\mathbf{F}} \mathbf{C}_2^\dagger + \mathbf{R}_2 \mathbf{A} = \mathbf{T}_{\mathbf{F}} \mathbf{K}_2^\dagger \mathbf{A} + \mathbf{R}_2 \mathbf{A} = (\mathbf{T}_{\mathbf{F}} \mathbf{K}_2^\dagger + \mathbf{R}_2)\mathbf{A} = \mathbf{K}_2 \mathbf{A}$$

so the outputted crs is indeed identically distributed to an honest one.

Then, we show that $\mathcal{B}$ outputs a valid statement-proof pair w.r.t. to $\mathsf{crs}^\dagger$. Since the commitment keys are extractable and perfectly binding at S, we have that $\boldsymbol{x}^\dagger$, $\boldsymbol{y}_1^\dagger$ and $\boldsymbol{y}_2^\dagger$ are valid openings for the commitments given. Assuming $\mathcal{A}$ produces a valid statement for $\mathcal{R}_{\rho,\boldsymbol{S}}^{\mathsf{no}}$, for the extracted values it holds that $\boldsymbol{x}^\dagger = \mathbf{M}_{S_1}\boldsymbol{w}$ and either $\boldsymbol{y}_1^\dagger \neq \mathbf{N}_{1,S_2}\boldsymbol{w}$ or $\boldsymbol{y}_2^\dagger \neq \mathbf{N}_{2,S_2}\boldsymbol{w}$. Thus, $\mathcal{B}_{\boldsymbol{S}}$ outputs a valid statement and it suffices to show that

$[\boldsymbol{\pi}^\dagger]_1, [\boldsymbol{\theta}^\dagger]_2$ is a valid proof. Indeed, we have that

$$\begin{aligned}
\mathbf{0} &= \boldsymbol{\pi}\mathbf{A} + \boldsymbol{\theta}\mathbf{A} - (\boldsymbol{c}^\top \mid \boldsymbol{d}_1^\top)\mathbf{C}_1 - \boldsymbol{d}_2^\top \mathbf{C}_2 \\
&= (\boldsymbol{\pi}^\dagger + \boldsymbol{c}^\top \mathbf{R}_0 + \boldsymbol{d}_1^\top \mathbf{R}_1)\mathbf{A} + (\boldsymbol{\theta}^\dagger + \boldsymbol{d}_2^\top \mathbf{R}_2)\mathbf{A} - (\boldsymbol{c}^\top \mid \boldsymbol{d}_1^\top)\left(\begin{pmatrix} \mathbf{T_G} & \mathbf{0} \\ \mathbf{0} & \mathbf{T_H} \end{pmatrix}\mathbf{C}_1^\dagger + \begin{pmatrix} \mathbf{R}_0 \\ \mathbf{R}_1 \end{pmatrix}\mathbf{A}\right) - \boldsymbol{d}_2^\top\left(\mathbf{T_F}\mathbf{C}_2^\dagger + \mathbf{R}_2\mathbf{A}\right) \\
&= (\boldsymbol{\pi}^\dagger + \boldsymbol{c}^\top \mathbf{R}_0 + \boldsymbol{d}_1^\top \mathbf{R}_1)\mathbf{A} + (\boldsymbol{\theta}^\dagger + \boldsymbol{d}_2^\top \mathbf{R}_2)\mathbf{A} - (\boldsymbol{c}^\top \mathbf{T_G} \mid \boldsymbol{d}_1^\top \mathbf{T_H})\mathbf{C}_1^\dagger - (\boldsymbol{c}^\top \mathbf{R}_0 - \boldsymbol{d}_1^\top \mathbf{R}_1)\mathbf{A} - \boldsymbol{d}_2^\top \mathbf{T_F}\mathbf{C}_2^\dagger - \boldsymbol{d}_2^\top \mathbf{R}_2\mathbf{A} \\
&= \boldsymbol{\pi}^\dagger\mathbf{A} + \boldsymbol{\theta}^\dagger\mathbf{A} - (\boldsymbol{c}^\top \mathbf{T_G} \mid \boldsymbol{d}_1^\top \mathbf{T_H})\mathbf{C}_1^\dagger - \boldsymbol{d}_2^\top \mathbf{T_F}\mathbf{C}_2^\dagger \\
&= \boldsymbol{\pi}^\dagger\mathbf{A} + \boldsymbol{\theta}^\dagger\mathbf{A} - ({\boldsymbol{x}^\dagger}^\top \mid {\boldsymbol{y}_1^\dagger}^\top)\mathbf{C}_1^\dagger - {\boldsymbol{y}_2^\dagger}^\top \mathbf{C}_2^\dagger
\end{aligned}$$

and the last equation is the verification equation for the knowledge transfer argument for $\mathsf{crs}^\dagger$. □

We next show that when the distribution $\mathcal{M}, \mathcal{N}_1, \mathcal{N}_2$ guarantee that the linear knowledge transfer argument is secure w.r.t. all possible sets $\boldsymbol{S}$, construction QABlin has h_{ls}-strong local knowledge soundness where h_{ls} includes $\mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N}_1, \mathbf{N}_2$, and some extra information about the matrix $\mathbf{M}$.

Corollary 6. *Let $\mathcal{D}_k$ be a matrix distribution for which $\mathcal{D}_k$-$\mathsf{SKerMDH}$. Denote $\mathcal{M}_S$ (resp. $\mathcal{N}_{1,S}$, $\mathcal{N}_{2,S}$) the distributions that sample matrices from $\mathcal{M}$ (res. $\mathcal{N}_1$, $\mathcal{N}_1$), and restricts them to rows corresponding to S. Then*

1. *If for all $S_1 \subseteq [d]$ with $S_1 \leq K_1$, $(\mathcal{M}_{S_1}^\top, h)$-$\mathsf{MDDH}$ holds, QABlin is an h_{ls}-strong local knowledge sound proof system, where $h_{\mathsf{ls}}(\theta) = (h(\mathbf{M}_S), \mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N}_1, \mathbf{N}_2)$.*
2. *If for all $S_1, S_2 \subseteq [d]$ with $S_1 \leq K_1$, $S_2 \leq K_2$ the distributions $\mathcal{M}_{S_1}, \mathcal{N}_{S_2}, \mathcal{N}_{S_2}$ output matrices with the last n' columns being $\mathbf{0}$, and $(\mathcal{M'}_{S_1}^\top, h)$-$\mathsf{MDDH}$ holds, with $\mathcal{M'}_{S_1}$ being $\mathcal{M}_{S_1}$ where we delete the trailing zero columns, then QABlin is an h_{ls}-strong local knowledge sound proof system, where $h_{\mathsf{ls}}(\theta) = (h(\mathbf{M}_S), \mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N}_1, \mathbf{N}_2)$.*

Proof. The proof is an immediate consequence of Thm. 15 and Thm. 14.1 for case 1 and Thm. 14.2 for case 2. □

The proof of oblivious trapdoor generation follows from the oblivious trapdoor generation and index set hiding of SSB commitments. We follow essentially the same proof as in the unilateral case.

First we show that we construct an indistinguishable crs given only the commitment keys and the matrices $\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2$.

Lemma 7. *There exists a modified crs generation algorithm K' that on input (ρ, θ'), where θ' contains only either $\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2$ or $\mathbf{G}, \mathbf{H}, \mathbf{F}$ outputs a crs such that (ρ, crs) are identically distributed to the honest algorithm.*

The lemma follows directly by noting that $[\mathbf{B}]_1, [\mathbf{D}]_2$ are efficiently computable given the commitment keys and the discrete logarithms of matrices $\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2$ (equivalently $\mathbf{G}, \mathbf{H}, \mathbf{F}$). As in the unilateral case, we abuse notation and refer to $\mathsf{K}'(\rho, \theta')$ as $\mathsf{K}(\rho, \theta')$.

In the next theorem we consider the three keys issued as a single key. It is easy to verify that the properties of the commitment keys still hold. Essentially, we want to capture the condition that the keys preserve oblivious key generation even if we consider a function h that outputs information that depends on all commitment keys. In our delegation construction this will correspond to $h(\mathbf{G}, \mathbf{H}, \mathbf{F}) = ([\mathbf{G}]_1, [\mathbf{H}]_2, [\mathbf{F}]_2, [\mathbf{H} \otimes \mathbf{F} - \mathbf{Z}]_1, [\mathbf{Z}]_2)$, for a uniform $\mathbf{Z}$, namely the information needed to obliviously create a crs for the kronecker composition of the last two keys.

Theorem 16. *Let $\mathcal{M}$, $\mathcal{N}_1$, $\mathcal{N}_2$ be witness samplable distributions, and CS be an algebraic SSB commitment scheme and let CS' be the concatenation of three instances of CS, that is it outputs $\mathbf{G}' = \begin{pmatrix} [\mathbf{G}0]_1 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & [\mathbf{G}_1]_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & [\mathbf{G}_2]_2 \end{pmatrix}$ with $\mathbf{G}_i \leftarrow \mathsf{CS.KeyGen}(gk, n, d, K_i, S_i)$. If CS' has h-strong oblivious trapdoor generation, then construction QABlin of Fig. 5 is h_{ns}-strong oblivious where $h_{ns} = (h(sk), \mathbf{M}_1, \mathbf{N}_1, \mathbf{N}_2)$. Furthermore,*

1. *For every PPT $\mathcal{A}$ against h_{ns}-strong index set hiding of* QABlin*, there exists an adversary $\mathcal{B}$ against h-index set hiding property of* CS*, such that* $\mathsf{Adv}^{\mathit{QABlin}}_{\mathsf{ISH}}(\mathcal{A}) \leq 3\mathsf{Adv}^{\mathsf{CS}}_{\mathsf{ISH}}(\mathcal{B})$.

2. *For every $\mathcal{A}$ against oblivious trapdoor generation of* QABlin*, there exists an adversary $\mathcal{B}$ against oblivious trapdoor of* CS*, such that* $\mathsf{Adv}^{\mathit{QABlin}}_{\mathsf{oblv}}(\mathcal{A}) \leq 3\mathsf{Adv}^{\mathsf{CS}}_{\mathsf{oblv}}(\mathcal{B})$.

Proof. Since the commitment key is perfectly binding at the extraction set, it is enough to show that h_{ns}-strong index set hiding holds and that we can sample a tuple (ρ, crs) indistinguishable from the one we are given, along with a valid trapdoor.

For index set hiding, it is enough to notice that the crs of QABlin can be efficiently computed given only $[\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2$. Indeed by sampling $[\mathbf{M}]_1, \mathbf{M} \leftarrow \mathcal{M}, [\mathbf{N}_1]_1, \mathbf{N}_1 \leftarrow \mathcal{N}_1, [\mathbf{N}_2]_2, \mathbf{N}_2 \leftarrow \mathcal{N}_2$ all values of crs are efficiently computable as noted in Lemma 7. Thus, a distinguishing advantage in index set hiding of QABlin immediately implies equal advantage on the respective property of CS.

For oblivious crs generation we first describe the OblSetup algorithm. Let $\boldsymbol{S}' \subseteq \boldsymbol{S}$.

$\mathsf{OblSetup}(\rho := ([\mathbf{G}]_1, [\mathbf{H}]_1, [\mathbf{F}]_2, [\mathbf{M}]_1, [\mathbf{N}_1]_1, [\mathbf{N}_2]_2), \mathsf{crs})$:

- $([\mathbf{G}']_1, \mathbf{T}'_{\mathbf{G}}) \leftarrow \mathsf{CS.OblSetup}(gk, d, K_0, S_0, [\mathbf{G}]_1)$.
- $([\mathbf{H}']_1, \mathbf{T}'_{\mathbf{H}}) \leftarrow \mathsf{CS.OblSetup}(gk, d, K_1, S_1, [\mathbf{H}]_1)$.
- $([\mathbf{F}']_2, \mathbf{T}'_{\mathbf{F}}) \leftarrow \mathsf{CS.OblSetup}(gk, d, K_2, S_2, [\mathbf{F}]_2)$.
- Sample $([\mathbf{M}']_1, \mathbf{M}') \leftarrow \mathcal{M}$; $([\mathbf{N}'_1]_1, \mathbf{N}'_1) \leftarrow \mathcal{N}_2$; $([\mathbf{N}'_2]_2, \mathbf{N}'_2) \leftarrow \mathcal{N}_2$;
- Set $\tau' = (\mathbf{T}'_{\mathbf{G}}, \mathbf{T}'_{\mathbf{H}}, \mathbf{T}'_{\mathbf{F}})$ and compute $\mathsf{crs} \leftarrow \mathsf{QABlin.K}(\rho, \theta' = (\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2))$.

Note that the only difference in sampling with $\boldsymbol{S}$ and with $\boldsymbol{S}'$ is how we sample the commitment keys $\mathbf{G}, \mathbf{H}, \mathbf{F}$; crs is identically distributed to an honest one since we sample $\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2$ in the same way that $\mathcal{D}_{\mathsf{par}}$ does. Also, by oblivious key generation of CS, the trapdoor τ' is a valid one w.r.t. $\mathbf{G}', \mathbf{H}', \mathbf{F}'$ and set $\boldsymbol{S}'$, so it extracts valid witnesses which, by perfect binding in S' are unique and do not assist the distinguisher which can compute them itself.

□

Finally, we get the following corollary.

Corollary 7. *When* CS *is the one from fig. 3 and* CS′ *is the concatenation of the three keys as described in Thm. 16 for and $h(\mathbf{G}, \mathbf{H}, \mathbf{F}) = ([\mathbf{H} \otimes \mathbf{F} - \mathbf{Z}]_1, [\mathbf{Z}]_2)$ for uniform $\mathbf{Z}$, then* QASum *from fig. 6 is h_{ns}-strong no-signaling where $h_{ns}(\theta) = (h(\mathbf{G}, \mathbf{H}, \mathbf{F}), \mathbf{M}, \mathbf{N}_1, \mathbf{N}_2)$.*

Proof. Follows directly from Thm. 6, the h_{ns}-strong ISH of the QALin which we show on Thm. 16 and the properties of the kronecker key operator (Thm. 5). □

B.2 Proof of security for the sum knowledge transfer quasi arguement

Theorem 17. *Let $\mathcal{M}_1, \mathcal{M}_2$ be (possibly correlated) witness samplable distribution, $\mathcal{N}$ be a witness samplable distribution, $\mathcal{D}_k$ a matrix distribution and* CS, CS′ *an algebraic and split algebraic SSB commitment respectively with perfect completeness. Also, let $\mathcal{A}$ be an adversary against h_{ls}-strong local soundness of construction* QASum *where $h_{ls} = (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}, \mathbf{N})$. Then,* QAsum *has perfect completeness and h_{ls}-strong local knowledge soundness holds with probability at least $1 - \mathsf{Adv}^{\Pi_{\mathit{kt\text{-}sum}}}_{\mathsf{snd}}(\mathcal{B}_{\boldsymbol{S}})$, where $\mathcal{B}_{\boldsymbol{S}}$ is any PPT adversary against soundness of $\Pi_{\mathit{kt\text{-}sum}}$.*

Proof. For completeness, we have that

$$\begin{aligned}
(\boldsymbol{c}_1^\top \mid \boldsymbol{d}^\top)\mathbf{C}_1 + \boldsymbol{c}_2^\top \mathbf{C}_2 &= (\boldsymbol{c}_1^\top \mid \boldsymbol{d}^\top)\begin{pmatrix}\mathbf{K}_0\\ \mathbf{K}_1\end{pmatrix}\mathbf{A} + \boldsymbol{c}_2^\top \mathbf{K}_0\mathbf{A}\\
&= (\boldsymbol{c}_1^\top \mathbf{K}_0 + \boldsymbol{d}^\top \mathbf{K}_1 + \boldsymbol{c}_2^\top \mathbf{K}_0)\mathbf{A}\\
&= \left((\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top)\mathbf{K}_0 + \boldsymbol{d}^\top \mathbf{K}_1\right)\mathbf{A}\\
&= \left(\boldsymbol{w}^\top(\mathbf{M}_1^\top + \mathbf{M}_2^\top)(\mathbf{Q}_1^\top + \mathbf{Q}_2^\top)\mathbf{K}_0 + \boldsymbol{w}^\top \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_1\right)\mathbf{A}\\
&= \boldsymbol{w}^\top\left((\mathbf{M}_1^\top + \mathbf{M}_2^\top)\mathbf{Q}^\top \mathbf{K}_0 + \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_1\right)\mathbf{A}\\
&= \boldsymbol{w}^\top\left((\mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{K}_0 + \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_1 + \mathbf{Z}) + (\mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{K}_0 - \mathbf{Z})\right)\mathbf{A}\\
&= \boldsymbol{w}^\top(\mathbf{B} + \mathbf{D})\,\mathbf{A}\\
&= \boldsymbol{w}^\top \mathbf{B}\mathbf{A} + \boldsymbol{w}^\top \mathbf{D}\mathbf{A}\\
&= \boldsymbol{\pi}\mathbf{A} + \boldsymbol{\theta}\mathbf{A}
\end{aligned}$$

Local knowledge soundness follows using almost an identical argument to Thm. 15 and reducing to knowledge transfer of KTA Sum Argument $\Pi_{\mathsf{kt\text{-}sum}}$ of Fig. 1. Given an adversary $\mathcal{A}$ breaking Knowledge Transfer of the quasi-argument of Fig. 6, we construct another adversary $\mathcal{B}_{\boldsymbol{S}}$ that breaks Knowledge Transfer of the argument $\Pi_{\mathsf{kt\text{-}sum}}$ for matrices $[\mathbf{M}_{1,S_0}]_1$, $[\mathbf{M}_{2,S_0}]_2$ and $[\mathbf{N}_{S_1}]_1$. $\mathcal{B}_{\boldsymbol{S}}$ works as follows: it takes input $(\rho^\dagger, h_{kt}(\theta^\dagger), \mathsf{crs}^\dagger)$ where

$$\rho^\dagger := (gk, [\mathbf{M}_{1,S_0}]_1, [\mathbf{M}_{2,S_0}]_2, [\mathbf{N}_{S_1}]_1), \quad h_{kt}(\theta^\dagger) := \mathbf{N}_{S_1}, \quad \mathsf{crs}^\dagger := ([\mathbf{B}^\dagger]_1, [\mathbf{D}^\dagger]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1^\dagger]_2, [\mathbf{C}_2^\dagger]_1)$$

and does the following:

- It samples $([\mathbf{Q}_1]_1, [\mathbf{Q}_2]_2, \mathbf{Q}_1, \mathbf{Q}_2, \mathbf{T}_{\mathbf{Q}}) \leftarrow \mathsf{CS}'.\mathsf{KGen}(gk, d, K, S_1)$ and sets $\mathbf{Q} := \mathbf{Q}_1 + \mathbf{Q}_2$.
- It samples $([\mathbf{F}]_1, \mathbf{F}, \mathbf{T}_{\mathbf{F}}) \leftarrow \mathsf{CS}.\mathsf{KGen}(gk, d, K, S_2)$.
- It samples $\mathbf{M}_{1,\overline{S}_1}, \mathbf{M}_{2,\overline{S}_1}, \mathbf{N}_{\overline{S}_2}$, such that $\mathbf{M}_1 = \mathbf{P}_{S_1}\begin{pmatrix}\mathbf{M}_{1,S_1}\\ \mathbf{M}_{1,\overline{S}_1}\end{pmatrix}$, $\mathbf{M}_2 = \mathbf{P}_{S_1}\begin{pmatrix}\mathbf{M}_{2,S_1}\\ \mathbf{M}_{2,\overline{S}_1}\end{pmatrix}$, $\mathbf{N} = \mathbf{P}_{S_2}\begin{pmatrix}\mathbf{N}_{S_2}\\ \mathbf{N}_{\overline{S}_2}\end{pmatrix}$.
- It samples $\mathbf{R}_0 \leftarrow \mathbb{Z}_p^{\overline{K}_0 \times k}$; $\mathbf{R}_1 \leftarrow \mathbb{Z}_p^{\overline{K}_1 \times k}$
- It computes $[\mathbf{B}]_1 := [\mathbf{B}^\dagger]_1 + [\mathbf{M}_1]_1^\top \mathbf{Q}^\top \mathbf{R}_0 + [\mathbf{N}]_1^\top \mathbf{F}^\top \mathbf{R}_1$ and $[\mathbf{D}]_2 := [\mathbf{D}^\dagger]_2 + [\mathbf{M}_2]_2^\top \mathbf{Q}^\top \mathbf{R}_0$
- It computes $[\mathbf{C}_1]_2 := \begin{pmatrix}\mathbf{T}_{\mathbf{Q}} & \mathbf{0}\\ \mathbf{0} & \mathbf{T}_{\mathbf{F}}\end{pmatrix}[\mathbf{C}_1^\dagger]_2 + \begin{pmatrix}\mathbf{R}_0\\ \mathbf{R}_1\end{pmatrix}[\mathbf{A}]_2$ and $[\mathbf{C}_2]_1 := \mathbf{T}_{\mathbf{Q}}[\mathbf{C}_2^\dagger]_1 + \mathbf{R}_0[\mathbf{A}]_1$.
- It sets

$$\rho := ([\mathbf{Q}_1]_1, [\mathbf{Q}_2]_1, [\mathbf{F}]_2, [\mathbf{M}_1]_1, [\mathbf{M}_2]_2, [\mathbf{N}]_1), \quad h_{ls}(\theta) := (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}, \mathbf{N})$$
$$\mathsf{crs} := ([\mathbf{B}]_1, [\mathbf{D}]_2, [\mathbf{A}]_{1,2}, [\mathbf{C}_1]_2, [\mathbf{C}_2]_1)$$

It then executes $\mathcal{A}(\rho, h_{ls}(\theta), \mathsf{crs})$ until it outputs a statement $([\boldsymbol{c}_1]_1, [\boldsymbol{c}_2]_2, [\boldsymbol{d}]_1, \boldsymbol{w})$ together with an accepting proof $[\boldsymbol{\pi}]_1, [\boldsymbol{\theta}]_2$. Given an accepting proof $\mathcal{B}$ sets $[\boldsymbol{x}_1^\dagger]_1 = \mathbf{T}_{\mathbf{Q}}[\boldsymbol{c}_1]_1, [\boldsymbol{x}_2^\dagger]_2 = \mathbf{T}_{\mathbf{Q}}[\boldsymbol{c}_2]_2, [\boldsymbol{y}^\dagger]_1 = \mathbf{T}_{\mathbf{F}}[\boldsymbol{d}]_1$, $[\boldsymbol{\pi}^\dagger]_1 = [\boldsymbol{\pi}]_1 - [\boldsymbol{c}_1]_1^\top \mathbf{R}_1 - [\boldsymbol{d}]_1^\top \mathbf{R}_2$ and $[\boldsymbol{\theta}^\dagger]_2 = [\boldsymbol{\theta}]_1 - [\boldsymbol{c}_2]_2^\top \mathbf{R}_1$. It outputs $\left(([\boldsymbol{x}_1^\dagger]_1, [\boldsymbol{x}_2^\dagger]_2, [\boldsymbol{y}^\dagger]_1), \boldsymbol{w}, ([\boldsymbol{\pi}^\dagger]_1, [\boldsymbol{\theta}^\dagger]_2)\right)$.

Note that by perfect completeness of the commitment scheme, the commitment keys are extractable and perfectly binding at S.

First, we claim that in this case the values $\rho, h_{ls}(\theta), \mathsf{crs}$ output by $\mathcal{B}_{\boldsymbol{S}}$ are identically distributed to honestly computed ones and thus do not skew the probability that $\mathcal{A}$ outputs a valid proof. For $\rho, h_{ls}(\theta)$,

this is immediate by the witness samplability of the distributions $\mathcal{M}_1$, $\mathcal{M}_2$, $\mathcal{N}$. We show that this holds for crs as well. Let $\mathbf{K}_0^\dagger \in \mathbb{Z}_p^{|S_1|\times k}, \mathbf{K}_1^\dagger \in \mathbb{Z}_p^{|S_2|\times k}$, $\mathbf{Z}^\dagger \in \mathbb{Z}_p^{n\times k}$ matrices satisfying:

$$\mathbf{B}^\dagger = \mathbf{M}_{1,S_1}^\top \mathbf{K}_0^\dagger + \mathbf{N}_S^\top \mathbf{K}_1^\dagger + \mathbf{Z}^\dagger,\ \mathbf{D}^\dagger = \mathbf{M}_{2,S_1}^\top \mathbf{K}_0^\dagger - \mathbf{Z}^\dagger,\ \mathbf{C}_1^\dagger = \begin{pmatrix}\mathbf{K}_0^\dagger \\ \mathbf{K}_1^\dagger\end{pmatrix}\mathbf{A} \text{ and } \mathbf{C}_2^\dagger = \mathbf{K}_0^\dagger \mathbf{A}.$$

Now $\mathcal{B}_S$ implicitly defines $\mathbf{K}_0 = \mathbf{T}_\mathbf{Q}\mathbf{K}_0^\dagger + \mathbf{R}_0$, $\mathbf{K}_1 = \mathbf{T}_\mathbf{F}\mathbf{K}_1^\dagger + \mathbf{R}_1$, and note that these matrices are uniformly distributed since $\mathbf{R}_0, \mathbf{R}_1$ are uniformly distributed. Thus $\mathbf{K}_0, \mathbf{K}_1$ are distributed identically to honestly generated values for generating a crs. We claim that the crs output by $\mathcal{A}$ is identically distributed to sampling this matrix and computing the other values honestly. Indeed we have that

$$\begin{aligned}
\mathbf{B} &= \mathbf{B}^\dagger + \mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{R}_1 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}_{1,S_1}^\top \mathbf{K}_0^\dagger + \mathbf{N}_{S_2}^\top \mathbf{K}_1 + \mathbf{Z}^\dagger + \mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{R}_0 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{T}_\mathbf{Q} \mathbf{K}_0^\dagger + \mathbf{N}^\top \mathbf{F}^\top \mathbf{T}_\mathbf{F}^\top \mathbf{K}_1 + \mathbf{Z}^\dagger + \mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{R}_0 + \mathbf{N}_1^\top \mathbf{H}^\top \mathbf{R}_1 \\
&= \mathbf{M}_1^\top \mathbf{Q}^\top (\mathbf{T}_\mathbf{Q} \mathbf{K}_0^\dagger + \mathbf{R}_0) + \mathbf{N}^\top \mathbf{F}^\top (\mathbf{T}_\mathbf{F}^\top \mathbf{K}_1 + \mathbf{R}_1) + \mathbf{Z}^\dagger \\
&= \mathbf{M}_1^\top \mathbf{Q}^\top \mathbf{K}_0 + \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_1 + \mathbf{Z}^\dagger
\end{aligned}$$

where the third equality follows since by the local extractability of the SSBs (1) $\mathbf{T}_\mathbf{Q}^\top \mathbf{Q} \mathbf{M}_1 = \mathbf{M}_{1,S}$ and (2) $\mathbf{T}_\mathbf{F}^\top \mathbf{F} \mathbf{N} = \mathbf{N}_S$. Similarly, we have

$$\begin{aligned}
\mathbf{D} &= \mathbf{D}^\dagger + \mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{R}_0 \\
&= \mathbf{M}_{2,S_1}^\top \mathbf{K}_0^\dagger - \mathbf{Z}^\dagger + \mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{R}_0 \\
&= \mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{T}_\mathbf{Q} \mathbf{K}_0^\dagger - \mathbf{Z}^\dagger + \mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{R}_0 \\
&= \mathbf{M}_2^\top \mathbf{Q}^\top (\mathbf{T}_\mathbf{Q} \mathbf{K}_0^\dagger + \mathbf{R}_0) - \mathbf{Z}^\dagger \\
&= \mathbf{M}_2^\top \mathbf{Q}^\top \mathbf{K}_0 - \mathbf{Z}^\dagger
\end{aligned}$$

Also, we have that

$$\mathbf{C}_1 = \begin{pmatrix}\mathbf{T}_\mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_\mathbf{F}\end{pmatrix}\mathbf{C}_1^\dagger + \begin{pmatrix}\mathbf{R}_0 \\ \mathbf{R}_1\end{pmatrix}\mathbf{A} = \begin{pmatrix}\mathbf{T}_\mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_\mathbf{F}\end{pmatrix}\begin{pmatrix}\mathbf{K}_0^\dagger \\ \mathbf{K}_1^\dagger\end{pmatrix}\mathbf{A} + \begin{pmatrix}\mathbf{R}_0 \\ \mathbf{R}_1\end{pmatrix}\mathbf{A} = i\begin{pmatrix}\mathbf{T}_\mathbf{Q}\mathbf{K}_0^\dagger + \mathbf{R}_0 \\ \mathbf{T}_\mathbf{F}\mathbf{K}_1^\dagger + \mathbf{R}_1\end{pmatrix}\mathbf{A} = \begin{pmatrix}\mathbf{K}_0 \\ \mathbf{K}_1\end{pmatrix}\mathbf{A}$$
$$\mathbf{C}_2 = \mathbf{T}_\mathbf{Q}\mathbf{C}_2^\dagger + \mathbf{R}_0\mathbf{A} = \mathbf{T}_\mathbf{Q}\mathbf{K}_0^\dagger\mathbf{A} + \mathbf{R}_0\mathbf{A} = (\mathbf{T}_\mathbf{Q}\mathbf{K}_0^\dagger + \mathbf{R}_0)\mathbf{A} = \mathbf{K}_0\mathbf{A}$$

so the outputted crs is indeed identically distributed to an honest one.

Then, we show that $\mathcal{B}$ outputs a valid statement-proof pair w.r.t. to $\mathsf{crs}^\dagger$. Since the commitment keys are extractable and perfectly binding, we have that $(\boldsymbol{x}_1^\dagger, \boldsymbol{x}_2^\dagger)$ and $\boldsymbol{y}^\dagger$ are valid openings for the commitments $(\boldsymbol{c}_1, \boldsymbol{c}_2)$ and $\boldsymbol{d}$ respectively. Assuming $\mathcal{A}$ produces a valid statement for $\mathcal{R}_{\rho,\boldsymbol{S}}^{\mathsf{no}}$, for the extracted values it holds that $\boldsymbol{x}_1^\dagger + \boldsymbol{x}_2^\dagger = (\mathbf{M}_{1,S_1} + \mathbf{M}_{2,S_1})\boldsymbol{w}$ and $\boldsymbol{y}^\dagger \neq \mathbf{N}_{S_2}\boldsymbol{w}$. Thus $\mathcal{B}_{\boldsymbol{S}}$ outputs a valid statement and it suffices to show that $(\boldsymbol{\pi}^\dagger, \boldsymbol{\theta}^\dagger)$ is a valid proof. Indeed, we have

$$\begin{aligned}
\mathbf{0} &= \boldsymbol{\pi}\mathbf{A} + \boldsymbol{\theta}\mathbf{A} - (\boldsymbol{c}_1^\top \mid \boldsymbol{d}^\top)\mathbf{C}_1 - \boldsymbol{c}_2^\top \mathbf{C}_2 \\
&= (\boldsymbol{\pi}^\dagger + \boldsymbol{c}_1^\top \mathbf{R}_0 + \boldsymbol{d}^\top \mathbf{R}_1)\mathbf{A} + (\boldsymbol{\theta}^\dagger + \boldsymbol{c}_2^\top \mathbf{R}_0)\mathbf{A} - (\boldsymbol{c}_1^\top \mid \boldsymbol{d}^\top)\left(\begin{pmatrix}\mathbf{T}_\mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}_\mathbf{F}\end{pmatrix}\mathbf{C}_1^\dagger + \begin{pmatrix}\mathbf{R}_0 \\ \mathbf{R}_1\end{pmatrix}\mathbf{A}\right) - \boldsymbol{c}_2^\top\left(\mathbf{T}_\mathbf{Q}\mathbf{C}_2^\dagger + \mathbf{R}_0\mathbf{A}\right) \\
&= (\boldsymbol{\pi}^\dagger + \boldsymbol{c}_1^\top \mathbf{R}_0 + \boldsymbol{d}^\top \mathbf{R}_1)\mathbf{A} + (\boldsymbol{\theta}^\dagger + \boldsymbol{c}_2^\top \mathbf{R}_0)\mathbf{A} - (\boldsymbol{c}_1^\top \mathbf{T}_\mathbf{Q} \mid \boldsymbol{d}^\top \mathbf{T}_\mathbf{F})\mathbf{C}_1^\dagger - (\boldsymbol{c}_1^\top \mathbf{R}_0 - \boldsymbol{d}^\top \mathbf{R}_1)\mathbf{A} - \boldsymbol{c}_2^\top \mathbf{T}_\mathbf{Q}\mathbf{C}_2^\dagger - \boldsymbol{c}_2^\top \mathbf{R}_0\mathbf{A} \\
&= \boldsymbol{\pi}^\dagger\mathbf{A} + \boldsymbol{\theta}^\dagger\mathbf{A} - (\boldsymbol{c}_1^\top \mathbf{T}_\mathbf{Q} \mid \boldsymbol{d}^\top \mathbf{T}_\mathbf{F})\mathbf{C}_1^\dagger - \boldsymbol{c}_2^\top \mathbf{T}_\mathbf{Q}\mathbf{C}_2^\dagger \\
&= \boldsymbol{\pi}^\dagger\mathbf{A} + \boldsymbol{\theta}^\dagger\mathbf{A} - (\boldsymbol{x}_1^{\dagger\top} \mid \boldsymbol{y}^{\dagger\top})\mathbf{C}_1^\dagger - \boldsymbol{x}_2^{\dagger\top}\mathbf{C}_2^\dagger
\end{aligned}$$

and the last equation is the verifying equation for the knowledge transfer argument for $\mathsf{crs}^\dagger$.

□

We next show that when the distributions $(\mathcal{M}_1\mathcal{M}_2), \mathcal{N}$ guarantee that the sum knowledge transfer argument is secure w.r.t. all possible sets $\boldsymbol{S}$, construction QASum has h_{ls}-strong local knowledge soundness where h_{ls} includes $\mathbf{G}, \mathbf{H}, \mathbf{F}, [\mathbf{M}]_2, [\mathbf{N}_1 \otimes \mathbf{N}_2 - \mathbf{R}]_1, [-\mathbf{R}]_2)$, for a uniform $\mathbf{R}$ and some extra information about the matrix $\mathbf{M}$.

Corollary 8. *Let $\mathcal{D}_k$ be a matrix distribution for which $\mathcal{D}_k$-SKerMDH. Denote $\mathcal{M}_{1,S}$ (resp. $\mathcal{M}_{2,S}$, $\mathcal{N}_S$) the distributions that sample matrices from $\mathcal{M}_1$ (res. $\mathcal{M}_2$, $\mathcal{N}$), and restricts them to rows corresponding to S. Then*

1. *If for all $S_0 \subseteq [d]$ with $S_0 \leq K_0$, $(\mathcal{M}_{1,S_0}^\top, \mathcal{M}_{2,S_0}^\top, h)$-MDDH holds, QASum is an h_{ls}-strong local knowledge sound proof system, where $h_{ls}(\theta) = (h(\mathbf{M}_{1,S}, \mathbf{M}_{2,S}), \mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N})$.*
2. *If for all $S_0, S_1 \subseteq [d]$ with $S_0 \leq K_0$, $S_1 \leq K_1$ the distributions $\mathcal{M}_{1,S_0}, \mathcal{M}_{2,S_0}, \mathcal{N}_{S_1}$ output matrices with the last n' columns being $\mathbf{0}$, and $(\mathcal{M'}_{1,S_0}^\top, \mathcal{M'}_{2,S_0}^\top, h)$-MDDH holds, with $\mathcal{M'}_{b,S_0}$ being $\mathcal{M}_{b,S_0}$ where we delete the trailing zero columns, then QASum is an h_{ls}-strong local knowledge sound proof system, where $h_{ls}(\theta) = (h(\mathbf{M}_{1,S_0}, \mathbf{M}_{2,S_0}), \mathbf{G}, \mathbf{H}, \mathbf{F}, \mathbf{N})$.*

Proof. The proof is an immediate consequence of Thm. 17 and Thm. 14.1 for case 1 and Thm. 14.2 for case 2. □

The proof that QASum is oblivious follows from the oblivious trapdoor generation and index set hiding of SSB commitments. We follow essentially the same proof as in the QABlin case.

First we show the corresponding lemma to Lemma 7, that is, we construct an indistinguishable crs given only the commitment keys and the matrices $\mathbf{M}_1, \mathbf{M}_2, \mathbf{N}$.

Lemma 8. *There exists a modified* crs *generation algorithm K' that on input (ρ, θ'), where θ' contains only either $\mathbf{M}_1, \mathbf{M}_2, \mathbf{N}$ or $\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F}$ and outputs a* crs *such that (ρ, crs) are identically distributed to the honest algorithm.*

Proof. Given these values we can compute the crs using a simple trick. Instead of computing

$$[\mathbf{B}]_1 = [\mathbf{M}_1^\top]_1 \mathbf{Q}^\top \mathbf{K}_0 + [\mathbf{N}^\top]_1 \mathbf{F}^\top \mathbf{K}_1 + [\mathbf{Z}]_1$$

$$[\mathbf{D}]_2 = [\mathbf{M}_2^\top]_2 \mathbf{Q}^\top \mathbf{K}_0 - [\mathbf{Z}]_2,$$

we compute

$$[\mathbf{B}]_1 = (\mathbf{M}_1^\top + \mathbf{M}_2^\top)[\mathbf{Q}_1^\top]_1 \mathbf{K}_0 + [\mathbf{N}^\top]_1 \mathbf{F}^\top \mathbf{K}_1 + [\mathbf{Z}]_1$$

$$[\mathbf{D}]_2 = (\mathbf{M}_2^\top + \mathbf{M}_2^\top)[\mathbf{Q}_2^\top]_2 \mathbf{K}_0 - [\mathbf{Z}]_2,$$

Noting that in both cases the elements computed are uniformly distributed conditioned on $\mathbf{B} + \mathbf{D} = (\mathbf{M}_1^\top + \mathbf{M}_2^\top)(\mathbf{Q}_1^\top + \mathbf{Q}_2^\top)\mathbf{K}_0 + \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_1$ we see that these values are computed as in the honest setup.

In the case where $\theta = (\mathbf{Q}_1, \mathbf{Q}_2, \mathbf{F})$ we can directly compute the crs by noting that $\mathbf{Q} = \mathbf{Q}_1 + \mathbf{Q}_2$ and the group elements in ρ are enough to compute all values of crs. □

As in the previous cases, we abuse notation and refer to $\mathsf{K}'(\rho, \theta')$ as $\mathsf{K}(\rho, \theta')$.

The proof of oblivious extraction essentially follows from the oblivious key generation and index set hiding of the SSB commitments and is similar to the proof of Thm. 16.

Theorem 18. *Let $\mathcal{M}_1, \mathcal{M}_2$ be (possibly correlated) witness samplable distribution, $\mathcal{N}$ be a witness samplable distribution, and $\mathsf{CS}, \mathsf{CS}'$ be an algebraic and a split algebraic SSB commitment scheme respectively with perfect completeness, oblivious trapdoor generation and h, h'-index set hiding respectively. Then Construction QASum of Fig. 6 is h_{ns}-strong oblivious, where $h_{ns}(\theta) = (h(sk), h'(sk'), \mathbf{M}_1, \mathbf{M}_2, \mathbf{N})$. Furthermore,*

1. *For every PPT $\mathcal{A}$ against index set hiding of* QASum*, there exist adversaries $\mathcal{B}_0, \mathcal{B}_1$ against index set hiding property of* CS', CS *respectively, such that* $\mathsf{Adv}^{QASum}_{\mathsf{ISH}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{CS}'}_{\mathsf{ISH}}(\mathcal{B}_0) + \mathsf{Adv}^{\mathsf{CS}}_{\mathsf{ISH}}(\mathcal{B}_1)$.

2. *For every $\mathcal{A}$ against oblivious crs generation of* QASum*, there exist an adversaries $\mathcal{B}_0, \mathcal{B}_1$ against oblivious key generation of* $\mathsf{CS}', \mathsf{CS}$ *respectively, such that* $\mathsf{Adv}^{QASum}_{\mathsf{oblv}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{CS}'}_{\mathsf{oblv}}(\mathcal{B}_0) + \mathsf{Adv}^{\mathsf{CS}}_{\mathsf{oblv}}(\mathcal{B}_1)$.

Proof. It is enough to show that h_{ns}-strong index set hiding holds and that we can sample a tuple (ρ, crs) indistinguishable from the one we are given, along with a valid trapdoor. This is the case because the commitment keys are perfectly binding in S', which means that the witnesses are unique and do not help the (unbounded) distinguisher who can compute them on its own.

Index Set Hidning. Assume there exist sets $\boldsymbol{S}, \boldsymbol{S'}$ of size at most $\boldsymbol{K}$ and an adversary $\mathcal{A}$ which distinguishes (ρ, crs) sampled for $\boldsymbol{S}$ from (ρ, crs) sampled for $\boldsymbol{S'}$ with some probability α. We construct adversaries $\mathcal{B}_0$ distinguishing ck_0 sampled for S_1 from ck_0 sampled for S'_1 with probability α_0 and an adversary $\mathcal{B}_1$ distinguishing ck_1 sampled for S_2 from ck_1 sampled for S'_2 with probability α_1 such that $\alpha \leq \frac{\alpha_0+\alpha_1}{2}$.

$\mathcal{B}_0$ takes as input some ck_0 and $h'(\mathsf{sk}_0)$ sampled either for S_0 or S'_0 and parses ck_0 as $[\mathbf{Q}]_1, [\mathbf{Q}]_2, \mathsf{aux}$. It then honestly computes the crs by sampling $\mathbf{M}_1, \mathbf{M}_2, \mathbf{N}$ and following the K described in Lemma 8 except that ck_1 is computed as follows: it samples $b \leftarrow \{0,1\}$ and if $b = 0$ it sets $(ck_1, sk_1) \leftarrow \mathsf{CS.KeyGen}(gk_1, d, K, S_1)$ otherwise it sets $(ck_1, sk_1) \leftarrow \mathsf{CS.KeyGen}(gk_1, d, K, S'_1)$.Note that, with probability $1/2$, the crs computed by $\mathcal{B}$ follows exactly the original distribution. This is the case since $\mathbf{B}, \mathbf{D}$ are uniform matrices conditioned on their sum being equal to $(\mathbf{M}_1^\top + \mathbf{M}_2^\top)(\mathbf{Q}_1^\top + \mathbf{Q}_2^\top)\mathbf{K}_1 + \mathbf{N}^\top \mathbf{F}^\top \mathbf{K}_2$ for uniform $\mathbf{K}_1, \mathbf{K}_2$, exactly as in the honest crs generation. Finally $\mathcal{B}_0$ runs $\mathcal{A}(\rho, \mathsf{crs}, h_{ns}(\theta) = (h'(\mathsf{sk}_0), h'(\mathsf{sk}_1), \mathbf{M}_1, \mathbf{M}_2, \mathbf{N}))$ and output whatever it outputs.

Similarly, on input $ck_1, h(\mathsf{sk}_1)$ sampled either for S_1 or S'_1, $\mathcal{B}_1$ samples $b \leftarrow \{0,1\}$ and if $b = 0$ it sets $(ck_0, sk_0) \leftarrow \mathsf{CS.KeyGen}(gk, d, K, S_0)$ otherwise it sets $(ck_0, sk_0) \leftarrow \mathsf{CS.KeyGen}(gk, d, K, S'_0)$ and honestly computes the crs as in the previous case. A simple case analysis shows that $\rho \leq \frac{\rho_1+\rho_2}{2}$.

Oblivious trapdoor generation: We show how to obliviously sample a trapdoor given black box access to $\mathsf{CS.OblKeyGen}$ and $\mathsf{CS}'.\mathsf{OblKeyGen}$. For oblivious trapdoor generation, given a pair ρ, crs for the quasi argument and set $\boldsymbol{S'}$ the oblivious setup $\mathsf{QASum.OblKeyGen}$ does the following:

- $(ck'_0, \tau'_0) \leftarrow \mathsf{CS.OblKeyGen}(ck_0, S'_0)$ and $(ck'_1, \tau'_1) \leftarrow \mathsf{CS.OblKeyGen}(ck_1, S'_1)$.
- Sample $([\mathbf{M}_1]_1, [\mathbf{M}_2]_2, \mathbf{M}_1, \mathbf{M}_2) \leftarrow \mathcal{M}$, $([\mathbf{N}]_1, \mathbf{N}) \leftarrow \mathcal{N}$.
- Compute the rest of the crs by $\mathsf{K}(ck'_0, ck'_1, \mathbf{M}_1, \mathbf{M}_2, \mathbf{N})$.

Arguing as in the index set hiding proof, the only difference in the oblivious and an honest crs is how the commitment keys are sampled. We can thus use a standard hybrid argument to reduce the property to the oblivious trapdoor generation of the commitment schemes $\mathsf{CS}, \mathsf{CS}'$.

□

Corollary 9. *If* CS *is the one from fig. 3, and* CS *is the construction of* kCS *of Thm. 5, then* QASum *from fig. 6 is h_{ns}-strong no-signaling where $h_{ns}(\theta) = (\mathbf{M}, \mathbf{N}_1, \mathbf{N}_2)$.*

Proof. The proof follows directly from Theorem 6 and the h_{ns}-strong oblivious property of QASum, which in turn follows from applying Theorems 1, 5 to Theorem 18. □