Unbounded Simulation-Sound Subversion Resistant Quasi-Adaptive NIZK Proofs and Applications to Modular zk-SNARKs

Behzad Abdolmaleki[1] and Daniel Slamanig[2]

[1] University of Tartu, Estonia
behzad.abdolmaleki@ut.ee
[2] AIT Austrian Institute of Technology, Vienna, Austria
daniel.slamanig@ait.ac.at

Abstract. Quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proofs are NIZK proofs where the common reference string (CRS) is allowed to depend on the language and they can be very efficient for specific languages. Thus, they are for instance used within the LegoSNARK toolbox (Campanelli et. al ACM CCS'19) as SNARKs for linear subspace languages. Recently, there has been an increasing interest to reduce trust in the generator of the CRS, as a fully trusted party is usually hard to find for real-world applications. One important line of work in this direction is subversion zero-knowledge (Bellare et al. ASIACRYPT'16), where the zero-knowledge property even holds when the CRS is generated maliciously.
In this paper, we investigate QA-NIZKs in the aforementioned setting. First, we analyze the security of the most efficient QA-NIZK constructions of Kiltz and Wee (EUROCRYPT'15) and the asymmetric QA-NIZKs by González et al. (ASIACRYPT'15) when the CRS is subverted and propose subversion versions of them. Secondly, for the first time, we construct ℓ-time simulation sound and unbounded simulation sound subversion QA-NIZK. Thirdly, we show how to integrate our subversion QA-NIZKs into the LegoSNARK toolbox, where subversion resistance is not yet considered. Our results together with recent subversion zk-SNARKS (Abdolmaleki et al. ASIACRYPT'17; Fuchsbauer PKC'18, Lipmaa EPRINT'19), are an important step towards a subversion variant of the LegoSNARK toolbox. Finally, we believe that our (SS) subversion QA-NIZKs will be of interest beyond the aforementioned application.

1 Introduction

Zero-knowledge (ZK) proofs introduced by Goldwasser, Micali and Rackoff [GMR89] are cryptographic protocols between two parties called the prover and the verifier with the purpose that the prover can convince the verifier of the validity of a statement in any language in NP without revealing additional information. Besides this zero-knowledge property, such a system needs to provide soundness, i.e., it must be infeasible for the prover

to provide proofs for false statements. While ZK proofs, in general, may require many rounds of interaction, an interesting variant is non-interactive zero-knowledge (NIZK) proofs which require only a single round, i.e., the prover outputs a proof which can then be verified by anybody. A long line of research [Kil92, GOS06, GS08, Gro10, Lip12, GGPR13, Gro16] has led to efficient pairing-based zero-knowledge Succinct Non-interactive ARguments of Knowledge (zk-SNARKs), which are NIZK proofs with i) a stronger notion of soundness called knowledge soundness and, more importantly, ii) in which proofs, as well as the computation of the verifier, are succinct, i.e., ideally a small constant amount of space and computation respectively.[3] Due to these latter properties, zk-SNARKs are a suitable tool to preserve privacy within cryptocurrencies and distributed ledger technologies, most notably used within Zcash [SCG+14] and Ethereum [Buc17]. They attract a considerable and increasing practical interest.[4,5] In this paper, we are interested in quasi-adaptive non-interactive zero-knowledge (QA-NIZK) proofs [JR13a], which are NIZKs that are quasi-adaptive in the sense that the common reference string (CRS) depends on a language parameter. They have many applications and have been a subject of intensive study [JR13a, LPJY14, JR14, ABP15, KW15, LPJY15, GHR15, GHKW16, AJOR18, KS19, AJO+19].

For practical applications of (QA-)NIZKs and zk-SNARKs, an important question is the generation of the CRS. While in theory it is simply assumed that some trusted party will perform the CRS generation, such a party is hard and in certain settings (such as fully decentralized systems) impossible to find in the real-world. Recently, there has been an increasing interest to reduce trust in the generator of the CRS. One of these lines of work is subversion zero-knowledge initiated by Bellare et al. in [BFS16], where the zero-knowledge property even holds when the CRS is generated maliciously, i.e., the CRS generator is subverted. Following this initial work, Abdolmaleki et al. [ABLZ17] as well as Fuchsbauer [Fuc18] investigated subversion zk-SNARKs. More recently, Abdolmaleki et al. in [ALSZ18] initiated the study of subversion QA-NIZK (Sub-QA-NIZK for short).

Our Contribution. Our results can be summarized as follows.

Sub-QA-NIZKs. We investigate the most efficient QA-NIZK constructions of Kiltz and Wee (KW) [KW15] and the asymmetric QA-NIZKs by González et al. (GHR) [GHR15] in a subverted setup, i.e., when the CRS is subverted. We show that for KW we can construct Sub-QA-NIZK arguments for their arguments Π_{as} and Π'_{as} (where latter requires a witness samplable distribution [JR13a]) by extending the CRS suitably. Thereby, compared to the recent Sub-QA-NIZK based upon KW in [ALSZ18] (called ALSZ henceforth), we only consider a weaker vari-

[3] We might use the terms proofs and arguments interchangeably.

[4] ZKProof (`https://zkproof.org/`) being the most notable industry and academic initiative towards a common framework and standards has been founded in 2018.

[5] Zero-knowledge proofs are *on the rise* in Gartner's' Hype Cycle for Privacy 2019, cf. `https://www.gartner.com/en/documents/3947373/hype-cycle-for-privacy-2019`.

ant where the CRS is subverted, but the language parameter is chosen honestly. However, we can overcome many of their limitations: 1) ALSZ works only if matrix $\boldsymbol{A}$ is a square matrix and checking whether the matrix is invertible (in the Vcrs algorithm) firstly is more costly than our approach and secondly, the Vcrs algorithm in our paper is the same for different values of k (and thus is more general), whereas in ALSZ one needs to change the Vcrs for different k (and it gets increasingly inefficient with growing k). 2) ALSZ (for the comparable case when the language parameter is picked honestly) works only for the QA-NIZK Π'_{as} of KW, which requires a witness sampleable distribution (restricting the set of languages which can be handled), whereas we present Sub-QA-NIZKs for both Π_{as} and Π'_{as} constructions of KW. 3) in contrast to ALSZ, which relies on a new non-standard knowledge assumption for their statistical subversion zero-knowledge property, our Sub-QA-NIZK can be shown to have this property under the well known and often used BDH and KoE assumptions. Moreover, we also show that we can achieve a Sub-QA-NIZK version of GHR by relying on the same assumptions.

Simulation Soundness of Sub-QA-NIZKs. We investigate the construction of Sub-QA-NIZK that satisfies a stronger notion of soundness called simulation soundness [Sah99, Sah01, Gro06]. A QA-NIZK is called ℓ-times/unbounded simulation sound if an adversary even when allowed to see ℓ/an arbitrary number of simulated proofs (which she can query adaptively), she cannot come up with a new valid proof. This is an important notion as it removes malleability from the proofs and thus prevents man-in-the-middle type of attacks, i.e., where an adversary can re-randomize a given proof and submit it again. Our work is the first treatment of ℓ-time and unbounded simulation sound Sub-QA-NIZK (ALSZ do not consider this notion) and we present unbounded simulation sound Sub-QA-NIZKs based on KW and under the same assumptions as above.

Towards Subversion LegoSNARK. The LegoSNARK framework for commit-and-prove zk-SNARKs (CP-SNARKs) recently proposed by Campanelli et. al in [CFQ19] uses QA-NIZK proofs (with succinct proofs) as the zk-SNARKs for linear subspace languages and in particular, they use a knowledge-sound version of the KW QA-NIZK Π'_{as}. We discuss how to integrate subversion primitives into LegoSNARK. In particular, we show how to integrate the subversion QA-NIZKs Π'_{as} and Π'_{asy} instead of their non-subversion counterparts.[6] Together with the recent results on subversion zk-SNARKs in [Lip19, Bag19, ARS20] (Sub-SE-SNARKs), we thus make an important step towards a complete subversion variant of the LegoSNARK framework.

2 Preliminaries

Let PPT denote probabilistic polynomial-time. Let $\lambda \in \mathbb{N}$ be the security parameter. All adversaries will be stateful. By $y \leftarrow \mathcal{A}(x;\omega)$ we denote the fact that

[6] We also stress that LegoSNARK can easily be extended by the non-subversion asymmetric QA-NIZKs in [GHR15].

$\mathcal{A}$, given an input x and random coins ω, outputs y. By $x \leftarrow_{\$} \mathcal{D}$ we denote that x is sampled according to distribution $\mathcal{D}$ or uniformly randomly if $\mathcal{D}$ is a set. Let $\mathsf{RND}(\mathcal{A})$ denote the random tape of $\mathcal{A}$, and let $\omega \leftarrow_{\$} \mathsf{RND}(\mathcal{A})$ denote the random choice of the random coins ω from $\mathsf{RND}(\mathcal{A})$. We denote by $\mathsf{negl}(\lambda)$ an arbitrary negligible function. We write $a \approx_\lambda b$ if $|a-b| \leq \mathsf{negl}(\lambda)$. A bilinear group generator $\mathsf{Pgen}(1^\lambda)$ returns $\mathsf{BG} = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, \hat{e})$, where $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$ are three additive cyclic groups of prime order p, and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a non-degenerate efficiently computable bilinear map (pairing). We use the implicit bracket notation of [EHK+13], that is, we write $[a]_\iota$ to denote ag_ι where g_ι is a fixed generator of $\mathbb{G}_\iota$. We denote $\hat{e}([a]_1, [b]_2)$ as $[a]_1[b]_2$. Thus, $[a]_1[b]_2 = [ab]_T$. We denote $s[a]_\iota = [sa]_\iota$ for $s \in \mathbb{Z}_p$ and $S \cdot [a]_\iota = [Sa]_T$ for $S \in \mathbb{G}_{3-\iota}$ and $\iota \in \{1, 2, T\}$. We freely use the bracket notation together with matrix notation, for example, if $\boldsymbol{XY} = \boldsymbol{Z}$ then $[\boldsymbol{X}]_1[\boldsymbol{Y}]_2 = [\boldsymbol{Z}]_T$. Furthermore in the Figures describing our QA-NIZK arguments, we will not explicitly provide return statements for P and Sim, but the output are all $\boldsymbol{\pi}$ elements.

Computational Assumptions. We require the following assumptions.

Definition 1 (KoE Assumption [Dam92]). *We say that* Pgen *is* KoE secure *if for any* λ, $\mathsf{BG} \leftarrow \mathsf{Pgen}(1^\lambda)$, $\iota \in \{1, 2, T\}$, *and PPT adversary* $\mathcal{A}$ *there exists a PPT extractor* $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}$, *such that*

$$\Pr\left[\begin{matrix}\omega_{\mathcal{A}} \leftarrow_{\$} \mathsf{RND}(\mathcal{A}), x \leftarrow_{\$} \mathbb{Z}_p([\alpha_1]_\iota, [\alpha_2]_\iota || a) \leftarrow (\mathcal{A} || \mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}})(\mathsf{BG}, [x]_\iota; \omega_{\mathcal{A}}) : \\ [\alpha_1]_\iota = x[\alpha_2]_\iota \wedge [\alpha_2]_\iota \neq [a]_\iota\end{matrix}\right] \approx_\lambda 0 \ .$$

Intuitively, given only $[1]_\iota$ and $[x]_\iota$ it is assumed to be hard to generate a pair of the form $([a]_\iota, a[x]_\iota)$, unless one starts by simply choosing $a \in \mathbb{Z}_p$.

Definition 2 (BDH Assumption [ABLZ17]). *We say that* Pgen *is* BDH-KE secure *for* $\mathcal{R}$ *if for any* λ, $(\mathcal{R}, \mathsf{aux}_{\mathcal{R}}) \in \mathrm{range}(\mathcal{R}(1^\lambda))$, *and PPT adversary* $\mathcal{A}$ *there exists a PPT extractor* $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}$, *such that*

$$\Pr\left[\begin{matrix}\omega_{\mathcal{A}} \leftarrow_{\$} \mathsf{RND}(\mathcal{A}), ([\alpha_1]_1, [\alpha_2]_2 || a) \leftarrow (\mathcal{A} || \mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}})(\mathcal{R}, \mathsf{aux}_{\mathcal{R}}; \omega_{\mathcal{A}}) : \\ [\alpha_1]_1[1]_2 = [1]_1[\alpha_2]_2 \wedge a \neq \alpha_1\end{matrix}\right] \approx_\lambda 0 \ .$$

Note that the BDH assumption can be considered as a simple case of the PKE assumption of [DFGK14](where $\mathcal{A}$ is given as an input the tuple $\{([x^i]_1, [x^i]_2)\}_{i=0}^n$ for some $n \geq 0$, and assumed that if $\mathcal{A}$ outputs $([\alpha]_1, [\alpha]_2)$ then she knows $(a_0, a_1, \ldots, a_n)$, such that $\alpha = \sum_{i=0}^n a_i x^i$.) as used in the case of asymmetric pairings in [DFGK14]. Thus, BDH can be seen as an asymmetric-pairing version of the original KoE assumption [Dam92].

In the following let $\mathcal{D}_k$ be a matrix distribution in $\mathbb{Z}_p^{(k+1)\times k}$ and we set $\mathcal{D}_k$ be the commonly used uniform distribution, where $\boldsymbol{A} \leftarrow_{\$} \mathcal{D}_k$ means that all $a_{ij} \leftarrow_{\$} \mathbb{Z}_p^*$, which in [EHK+13] is denoted as $\mathcal{U}_k$, i.e., $\mathcal{D}_k := \mathcal{U}_k$.

Definition 3 ($\mathcal{D}_k$**-Matrix Diffie-Hellman** ($\mathcal{D}_k$-MDDH) **Assumption [MRV16]).** *The* $\mathcal{D}_k$-MDDH *assumption for* $\iota \in \{1,2\}$ *holds relative to* K_0*, if for any PPT adversary* $\mathcal{A}$*,* $|\mathsf{Exp}^{\mathsf{MDDH}}_{\mathcal{A}}(\mathsf{p}) - 1/2| \approx_\lambda 0$*, where* $\mathsf{Exp}^{\mathsf{MDDH}}_{\mathcal{A}}(\mathsf{p}) :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow_\$ \mathsf{K}_0(1^\lambda); \boldsymbol{A} \leftarrow_\$ \mathcal{D}_k; \boldsymbol{v} \leftarrow_\$ \mathbb{Z}_p^k; \boldsymbol{u} \leftarrow_\$ \mathbb{Z}_p^{k+1}; b \leftarrow_\$ \{0,1\}; \\ b^* \leftarrow \mathcal{A}(\mathsf{p}, [\boldsymbol{A}]_\iota, [\boldsymbol{u}]_\iota, [b \cdot \boldsymbol{A}\boldsymbol{v} + (1-b)\cdot \boldsymbol{u}]_\iota)\end{array} : b = b^*\right].$$

Definition 4 ($\mathcal{D}_k$**-KerMDH Assumption [MRV16]).** *The* $\mathcal{D}_k$-KerMDH *assumption for* $\iota \in \{1,2\}$ *holds relative to* K_0*, if for any PPT* $\mathcal{A}$*,*

$$\Pr\left[\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \boldsymbol{A} \leftarrow_\$ \mathcal{D}_k; [\boldsymbol{s}]_{3-\iota} \leftarrow \mathcal{A}(\mathsf{p}, [\boldsymbol{A}]_\iota) : \boldsymbol{s} \neq \boldsymbol{0} \wedge \boldsymbol{A}^\top \boldsymbol{s} = \boldsymbol{0}_k \right] \approx_\lambda 0 \ .$$

Note that as shown in [MRV16], if $\mathcal{D}_k$-MDDH holds then $\mathcal{D}_k$-KerMDH holds.

Definition 5 ($\mathcal{D}_k$**-SKerMDH Assumption [GHR15]).** *The* $\mathcal{D}_k$-SKerMDH *assumption holds relative to* K_0*, if for any PPT* $\mathcal{A}$*,*

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \boldsymbol{A} \leftarrow_\$ \mathcal{D}_k; ([\boldsymbol{s}_1]_1, [\boldsymbol{s}_2]_2) \leftarrow \mathcal{A}(\mathsf{p}, [\boldsymbol{A}]_1, [\boldsymbol{A}]_2) : \\ \boldsymbol{s}_1 - \boldsymbol{s}_2 \neq \boldsymbol{0} \wedge \boldsymbol{A}^\top(\boldsymbol{s}_1 - \boldsymbol{s}_2) = \boldsymbol{0}_k\end{array}\right] \approx_\lambda 0 \ .$$

Quasi-Adaptive NIZK Arguments. We recall the definition of QA-NIZK arguments of Jutla and Roy [JR13a]. A QA-NIZK proof system provides a proof for membership of words x with according witnesses $\boldsymbol{w}$ in a language $\mathcal{L}_\varrho$ defined by a relation $\mathcal{R}_\varrho$ which is parametrized by some parameter ϱ chosen from a distribution $\mathcal{D}_\mathsf{p}$. The distribution $\mathcal{D}_\mathsf{p}$ is witness samplable if there exist an efficient algorithm that samples $(\varrho, \mathtt{tc}_\varrho)$ so that the parameter ϱ is distributed according to $\mathcal{D}_\mathsf{p}$. The membership of the language parameter ϱ can be efficiently verified with $\mathtt{tc}_\varrho$. The CRS of QA-NIZKs depends on a language parameter ϱ and as mentioned in [JR13a], it has to be chosen from a correct distribution $\mathcal{D}_\mathsf{p}$.

A tuple of PPT algorithms $\mathit{\Pi} = (\mathsf{Pgen}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ is a QA-NIZK argument in the CRS model for a set of witness-relations $\mathcal{R}_\mathsf{p} = \{\mathcal{R}_\varrho\}_{\varrho \in \mathrm{Supp}(\mathcal{D}_\mathsf{p})}$ with ϱ sampled from a distribution $\mathcal{D}_\mathsf{p}$ over associated parameter language $\mathcal{L}_\mathsf{p}$, if the following properties (i-iii) hold. Here, Pgen is the parameter and the CRS generation algorithm, more precisely, Pgen consists of two algorithms K_0 (generates the the parameter p) and K (generates the CRS), P is the prover, V is the verifier, and Sim is the simulator.

(i) **Completeness.** For any λ, and $(x, \boldsymbol{w}) \in \mathcal{R}_\varrho$,

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; \\ (\mathsf{crs}, \mathtt{tc}) \leftarrow \mathsf{K}(\varrho); \pi \leftarrow \mathsf{P}(\varrho, \mathsf{crs}, x, \boldsymbol{w}) : \\ \mathsf{V}(\varrho, \mathsf{crs}, x, \pi) = 1\end{array}\right] = 1 \ .$$

(ii) **Statistical Zero-Knowledge.** For any λ, and computationally unbounded adversary $\mathcal{A}$, $2 \cdot |\varepsilon^{zk} - 1/2| \approx_\lambda 0$, where $\varepsilon^{zk} :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; (\mathsf{crs}, \mathtt{tc}) \leftarrow \mathsf{K}(\varrho); b \leftarrow_\$ \{0,1\} : \\ \mathcal{A}^{\mathsf{O}_b(\cdot,\cdot)}(\varrho, \mathsf{crs}) = 1\end{array}\right].$$

The oracle $\mathsf{O}_0(x, \boldsymbol{w})$ returns $\perp$ (reject) if $(x, \boldsymbol{w}) \notin \mathcal{R}_\varrho$, and otherwise it returns $\mathsf{P}(\varrho, \mathsf{crs}, x, \boldsymbol{w})$. Similarly, $\mathsf{O}_1(x, \boldsymbol{w})$ returns $\perp$ (reject) if $(x, \boldsymbol{w}) \notin \mathcal{R}_\varrho$, and otherwise it returns $\mathsf{Sim}(\varrho, \mathsf{crs}, \mathsf{tc}, x)$.

(iii) **Computational Soundness.** For any λ, and PPT $\mathcal{A}$,

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; (\mathsf{crs}, \mathsf{tc}) \leftarrow \mathsf{K}(\varrho); \\ (x, \pi) \leftarrow \mathcal{A}(\varrho, \mathsf{crs}) : \\ \mathsf{V}(\varrho, \mathsf{crs}, x, \pi) = 1 \wedge \neg(\exists \boldsymbol{w} : (x, \boldsymbol{w}) \in \mathcal{R}_\varrho)\end{array}\right] \approx_\lambda 0 \ .$$

Also we consider a strengthening of adaptive soundness called simulation soundness [Sah99, Sah01, Gro06] which stipulates that even if an adversary $\mathcal{A}$ can see simulated proofs for words of its choice, it cannot provide valid proofs for words outside the language. For simulation soundness we implicitly use a tag-based variant of QA-NIZKs, i.e., every proof π is associated to a unique tag τ which is also input to the V algorithm (the adaptations to the above definitions are straightforward and we refer to [KW15] for details).

Simulation Soundness. For any λ, and PPT $\mathcal{A}$,

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; (\mathsf{crs}, \mathsf{tc}) \leftarrow \mathsf{K}(\varrho); \\ (\tau', x', \pi') \leftarrow \mathcal{A}^{\mathsf{O}(\cdot,\cdot)}(\varrho, \mathsf{crs}) : \\ x' \notin \mathcal{L}_\varrho \wedge \tau' \notin Q_\tau \wedge \mathsf{V}(\varrho, \mathsf{crs}, \tau', x', \pi') = 1\end{array}\right] \approx_\lambda 0 \ .$$

where $\mathsf{O}(\tau, x)$ outputs $\mathsf{Sim}(\varrho, \mathsf{crs}, \tau, x, \mathsf{tc})$ and adds τ to the set Q_τ keeping track of the queried tags. A QA-NIZK is called ℓ-time simulation-sound, if $\mathcal{A}$ is restricted to make at most ℓ queries to the oracle O and unbounded simulation-sound (USS) otherwise.

QA-NIZK Argument for Linear Spaces. Now we recall the two constructions of QA-NIZK arguments of membership in linear spaces given by Kiltz and Wee [KW15] for the language

$$\mathcal{L}_{[\boldsymbol{M}]_1} = \left\{ [\boldsymbol{y}]_1 \in \mathbb{G}_1^n : \exists \boldsymbol{w} \in \mathbb{Z}_p^m \text{ s.t. } \boldsymbol{y} = \boldsymbol{M}\boldsymbol{w} \right\} \ .$$

The corresponding relation is defined as $\mathcal{R}_{[\boldsymbol{M}]_1} = \{([\boldsymbol{y}]_1, \boldsymbol{w}) \in \mathbb{G}_1^n \times \mathbb{Z}_p^m : \boldsymbol{y} = \boldsymbol{M}\boldsymbol{w}\}$. This language is useful in many applications (cf. [JR13a] and follow up work). We recall the full construction of the Kiltz-Wee QA-NIZK arguments for linear subspaces in the CRS model in Fig. 1.

Theorem 1 (Theorem 1 of [KW15]). *If $\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$, Fig. 1 describes a QA-NIZK proof system Π_{as} with perfect completeness, computational adaptive soundness based on the $\mathcal{D}_k$-*KerMDH *assumption, perfect zero-knowledge, and proof size $k + 1$.*

Theorem 2 (Theorem 2 of [KW15]). *If $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$, $\hat{k} = k$, the, and $\mathcal{D}_\mathsf{p}$ is a witness samplable distribution, Fig. 1 describes a QA-NIZK proof system Π_{as} with perfect completeness, computational adaptive soundness based on the $\mathcal{D}_k$-*KerMDH *assumption, perfect zero-knowledge, and proof size k.*

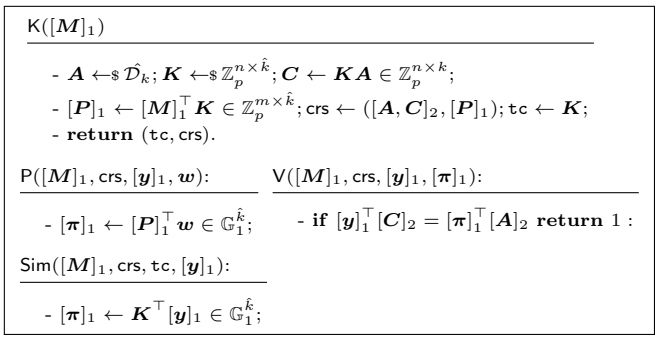

Fig. 1. Kiltz-Wee QA-NIZK Π_{as} ($\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k+1$) and Π'_{as} ($\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$).

In the following we recall the two core Lemmas 1 and 3 of [KW15], where Lemma 1 is used in the proof of Theorem 1 and together with Lemma 3 is used later in the proofs of Sections 3 and 4.

Lemma 1 (Lemma 2 of [KW15]). *Let n, m and k be integers. For any $\boldsymbol{M} \in \mathbb{Z}_p^{n\times m}$, $\boldsymbol{A} \in \mathbb{Z}_p^{(k+1)\times k}$ and any (possibly unbounded) adversary $\mathcal{A}$,*

- $\Pr\left[\begin{array}{l}\boldsymbol{K} \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}; (\boldsymbol{y},\boldsymbol{\pi}) \leftarrow \mathcal{A}(\boldsymbol{M}^\top\boldsymbol{K}, \boldsymbol{K}\boldsymbol{A}) : \\ \boldsymbol{y} \notin \mathit{span}(\boldsymbol{M}) \wedge \boldsymbol{\pi}^\top = \boldsymbol{y}^\top\boldsymbol{K}\end{array}\right] \leq 1/p.$
- $\Pr\left[\begin{array}{l}\boldsymbol{K}_0, \boldsymbol{K}_1 \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}; \\ (\tau,\boldsymbol{y},\boldsymbol{\pi}) \leftarrow \mathcal{A}^{\mathsf{O}(\cdot)}(\boldsymbol{M}^\top\boldsymbol{K}_0, \boldsymbol{M}^\top\boldsymbol{K}_1, \boldsymbol{K}_0\boldsymbol{A}, \boldsymbol{K}_1\boldsymbol{A}) : \\ \boldsymbol{y} \notin \mathit{span}(\boldsymbol{M}) \wedge \tau \neq \tau' \wedge \boldsymbol{\pi}^\top = \boldsymbol{y}^\top(\boldsymbol{K}_0 + \tau'\boldsymbol{K}_1)\end{array}\right] \leq 1/p.$

where $\mathsf{O}(\tau')$ can be called only one time and returns $\boldsymbol{K}_0 + \tau'\boldsymbol{K}_1$.

Lemma 2 (Lemma 3 of [KW15]). *For all adversaries $\mathcal{A}$, there is an adversary $\mathcal{B}$ against* MDDH *problem with*

$$\Pr\left[\begin{array}{l}\boldsymbol{B}, \boldsymbol{A} \leftarrow_\$ \mathcal{D}_k; \boldsymbol{K}_0, \boldsymbol{K}_1 \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}; \\ (\boldsymbol{P}_0, \boldsymbol{P}_1) \leftarrow (\boldsymbol{B}^\top\boldsymbol{K}_0, \boldsymbol{B}^\top\boldsymbol{K}_1); \\ \mathsf{pk} \leftarrow ([\boldsymbol{P}_0, \boldsymbol{P}_1, \boldsymbol{B}]_1, \boldsymbol{K}_0\boldsymbol{A}, \boldsymbol{K}_1\boldsymbol{A}); \\ b \leftarrow_\$ \{0,1\}; \\ (\tau,\boldsymbol{y},\boldsymbol{\pi}) \leftarrow \mathcal{A}^{\mathsf{O}_b(\cdot),\mathsf{O}^*(\cdot)}(\mathsf{pk}) : \\ \tau' \neq Q_\tau \wedge b' = b\end{array}\right] \leq 1/2 + 2Q \cdot \varepsilon_{\mathrm{MDDH}} + Q/p.$$

where Q is the number of $\mathcal{A}$'s queries to O_b and $\mathcal{A}$ only makes one query to the oracle $\mathsf{O}^*(\tau')$ and obtains $\boldsymbol{K}_0 + \tau'\boldsymbol{K}_1$. Also $\mathsf{O}_b(\tau)$ outputs $[br'\boldsymbol{A}^\perp + (\boldsymbol{P}_0^\top + \tau\boldsymbol{P}_1^\top)\boldsymbol{r}]_1, [\boldsymbol{B}\boldsymbol{r}]_1$ where $r' \leftarrow_\$ \mathbb{Z}_p$, $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^k$, and $\boldsymbol{A}^\perp\boldsymbol{A} = \boldsymbol{0}$, and $\boldsymbol{A}^\perp \neq \boldsymbol{0}$. Finally the tag τ is added to Q_τ.

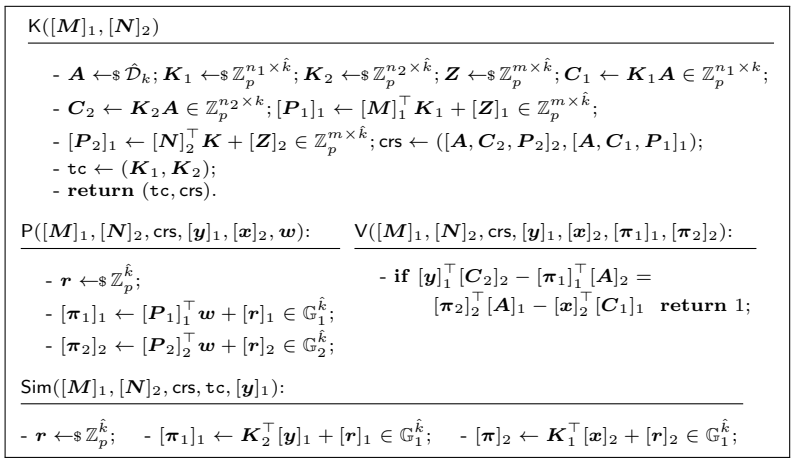

Fig. 2. Asymmetric QA-NIZK Π_{asy} ($\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$) and Π'_{asy} ($\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$).

Asymmetric QA-NIZK Argument Systems. We recall the constructions of asymmetric QA-NIZK arguments of membership in different subspace concatenation of $\mathbb{G}_1^{n_1} \times \mathbb{G}_2^{n_2}$ given by Gonzalez et al. [GHR15] for the language

$$\mathcal{L}_{[\boldsymbol{M}]_1,[\boldsymbol{N}]_2} = \left\{ ([\boldsymbol{y}]_1, [\boldsymbol{x}]_2) \in \mathbb{G}_1^{n_1} \times \mathbb{G}_2^{n_2} : \exists \boldsymbol{w} \in \mathbb{Z}_p^m \text{ s.t. } \boldsymbol{y} = \boldsymbol{M}\boldsymbol{w}, \boldsymbol{x} = \boldsymbol{N}\boldsymbol{w} \right\} .$$

This language is also known as the concatenation language, since one can define $\boldsymbol{R}$ as a concatenation of language parameters $[\boldsymbol{M}]_1$ and $[\boldsymbol{N}]_2$ so that $\boldsymbol{R} = \begin{pmatrix} [\boldsymbol{M}]_1 \\ [\boldsymbol{N}]_2 \end{pmatrix}$. In other words $([\boldsymbol{y}]_1, [\boldsymbol{x}]_2) \in \mathcal{L}_{[\boldsymbol{M}]_1,[\boldsymbol{N}]_2}$ iff $\begin{pmatrix} [\boldsymbol{y}]_1 \\ [\boldsymbol{x}]_2 \end{pmatrix}$ is in the span of $\boldsymbol{R}$. We recall the full construction of asymmetric QA-NIZK arguments in the CRS model in Fig. 2.

Notice that the QA-NIZK in Fig. 2 for $\mathcal{L}_{[\boldsymbol{M}]_1,[\boldsymbol{N}]_2}$ is a generalization of Π_{as} of [KW15] in two groups when we set $\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k+1$ (denoted as Π_{asy}). Also it is a generalization of Π'_{as} of [KW15] in two groups when we set $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$ (denoted as Π'_{asy}).

Theorem 3 (Theorem 3 of [GHR15]). *If $\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$, the QA-NIZK proof system in Fig. 2 is perfect complete, computational adaptive soundness based on the $\mathcal{D}_k$-*SKerMDH *assumption, perfect zero-knowledge.*

Theorem 4 (Theorem 4 of [GHR15]). *If $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$, $\hat{k} = k$ and $\mathcal{D}_{\mathsf{p}}$ is a witness samplable distribution, Fig. 2 describes a QA-NIZK proof system with perfect completeness, computational adaptive soundness based on the $\mathcal{D}_k$-*KerMDH *assumption, perfect zero-knowledge.*

3 QA-NIZK Arguments in the Subversion Setting

In this section, we investigate QA-NIZK arguments when the CRS is subverted and propose corresponding Sub-QA-NIZK arguments. First we discuss subversion security and then our focus will be on the two fundamental QA-NIZK constructions Π_{as} and Π'_{as} in [KW15] (cf. Section 2) and the asymmetric QA-NIZK constructions Π_{asy} and Π'_{asy} in [GHR15] (cf. Section 2) for linear subspaces languages

3.1 Security Definitions for Subversion QA-NIZK Arguments

The notion of subversion security for QA-NIZKs in the CRS model was first noted by Jutla and Roy in the full version of [JR13a] (cf. [JR13b]). They have shown that one can obtain both soundness and zero-knowledge (under falsifiable assumptions) when the language parameter ϱ is subverted but the CRS is *generated honestly*. They showed that such a setting can cover a large family of subspace languages[7]. Later Abdolmaleki et al. [ALSZ18] defined the security of QA-NIZKs in the bare public-key (BPK) model, when both ϱ and the CRS are subverted. More precisely, they obtain a version of the Kiltz-Wee QA-NIZK [KW15] when both ϱ and CRS are chosen maliciously, but under a new non-falsifiable KWKE knowledge assumption. They also obtain (knowledge) soundness when only ϱ is chosen maliciously under a new (non-falsifiable) interactive assumptions $\mathrm{KerMDH}^{\mathrm{dl}}$ and $\mathrm{SKerMDH}^{\mathrm{dl}}$ [ALSZ18].

In this paper, we investigate the missing direction, namely the security of QA-NIZKs in the CRS model when the CRS is subverted but with *honestly chosen* ϱ. This can be viewed as a dual version of Jutla and Roy's QA-NIZK in [JR13b, JR13a]. Concretely, we define Sub-QA-NIZKs security with some changes in the CRS model. The most important properties are subversion completeness (an honest prover convinces an honest verifier, and an honestly generated CRS passes the CRS checking), computational soundness (an honest prover convinces an honest verifier), and statistical subversion zero-knowledge (given a possibly subverted CRS, a proof generated by the honest prover reveals no information about the witness). A tuple of PPT algorithms $\Pi = (\mathsf{Pgen}, \mathsf{Vcrs}, \mathsf{P}, \mathsf{V}, \mathsf{Sim})$ is a Sub-QA-NIZK argument if the following properties (i-iii) hold. Here, Vcrs is a new algorithm that checks the well-formedness of the CRS. We note that since soundness is proved in the case crs is generated correctly (by the verifier or a trusted third party) and V does not need to run Vcrs, so the computational soundness and the simulation soundness are similar to the original QA-NIZK definitions. We note that similar to [ALSZ18] by a subversion QA-NIZK argument we mean a *no-auxiliary-string non-black-box zero knowledge subversion QA-NIZK argument*. In this paper for the sake of simplicity we just use subversion QA-NIZK or Sub-QA-NIZK for short.

[7] As here only ϱ is subverted (CRS is created honestly) and ϱ has a different role compared to the CRS, such result does not contradict the impossibility result in [BFS16].

(i) **Subversion Completeness.** For any λ, and $(x, \boldsymbol{w}) \in \mathcal{R}_\varrho$,

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; (\mathsf{crs}, \mathtt{tc}) \leftarrow \mathsf{K}(\varrho); \\ \pi \leftarrow \mathsf{P}(\varrho, \mathsf{crs}, x, \boldsymbol{w}) : \\ \mathsf{Vcrs}(\varrho, \mathsf{crs}) = 1 \wedge \mathsf{V}(\varrho, \mathsf{crs}, x, \pi) = 1\end{array}\right] = 1 \ .$$

(ii) **Statistical Subversion Zero-Knowledge.** For any PPT subverter Z there exists a PPT extractor Ext_Z, such that for any λ, and computationally unbounded adversary $\mathcal{A}$, $2 \cdot |\varepsilon^{zk} - 1/2| \approx_\lambda 0$, where $\varepsilon^{zk} :=$

$$\Pr\left[\begin{array}{l}\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \varrho \leftarrow_\$ \mathcal{D}_\mathsf{p}; \omega_\mathsf{Z} \leftarrow_\$ \mathsf{RND}(\mathsf{Z}); (\mathsf{crs}, \mathsf{aux}_\mathsf{Z}) \leftarrow \mathsf{Z}(\varrho; \omega_\mathsf{Z}); \\ \mathtt{tc} \leftarrow \mathsf{Ext}_\mathsf{Z}(\varrho; \omega_\mathsf{Z}); b \leftarrow_\$ \{0,1\} : \\ \mathsf{Vcrs}(\varrho, \mathsf{crs}) = 1 \wedge \mathcal{A}^{\mathsf{O}_b(\cdot,\cdot)}(\varrho, \mathsf{crs}, \mathsf{aux}_\mathsf{Z}) = 1\end{array}\right] .$$

The oracle $\mathsf{O}_0(x, \boldsymbol{w})$ returns $\perp$ (reject) if $(x, \boldsymbol{w}) \notin \mathcal{R}_\varrho$, and otherwise it returns $\mathsf{P}(\varrho, \mathsf{crs}, x, \boldsymbol{w})$. Similarly, $\mathsf{O}_1(x, \boldsymbol{w})$ returns $\perp$ (reject) if $(x, \boldsymbol{w}) \notin \mathcal{R}_\varrho$, and otherwise it returns $\mathsf{Sim}(\varrho, \mathsf{crs}, \mathtt{tc}, x)$.

(iii) **Computational Soundness.** Is similar to the original definition.
Simulation soundness. Is similar to the original definition.

3.2 QA-NIZK in the Subversion Setting

In this part, we construct a Sub-QA-NIZK based on [KW15]. Intuitively, for constructing such a system, one needs to make the CRS publicly verifiable, and also the trapdoor of the CRS should be extractable under some knowledge assumption (the latter is required to simulate proofs in the subversion zero-knowledge game).

For the QA-NIZKs Π_{as} and Π'_{as} from [KW15], we achieve the first property by defining a Vcrs algorithm which takes the CRS crs and the language parameter ϱ of the QA-NIZK's language and checks the well-formedness of crs. If the possibly maliciously generated crs (from the prover's point of view) passes the Vcrs algorithm, it is guaranteed that there exists a trapdoor $\mathtt{tc}$ for crs. Then, by using the well-known BDH and KoE knowledge assumptions, we can extract the trapdoor $\mathtt{tc}$ from crs which realizes the second property. Note that in some cases, similar to [Fuc18], one can achieve the public verifiability property of the CRS for free, i.e., without adding some extra elements to the CRS (cf. Section 3.3). Here, however, and similar to [ABLZ17], we need to add some extra elements $[\boldsymbol{A}]_1 \in \mathbb{G}_1^{(k+1)\times k}$ to the CRS. Then, we prove that the new construction is subversion complete, subversion zero-knowledge and sound in Theorem 5.

Fig. 3 describes Sub-QA-NIZK argument Π_{sub}, which is the subversion version of the Kiltz-Wee [KW15] QA-NIZK for linear subspaces in the CRS model. We note that when $\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$, Π_{sub} of Fig. 3 is the subversion version of Π_{as} in [KW15]. When $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$ then Π_{sub} in Fig. 3 is the subversion version of Π'_{as} in [KW15]. In Lemma 3, we show that from any adversary producing a valid CRS crs from scratch it is possible to extract the trapdoor $\boldsymbol{K}$ (simulation trapdoors). We will use it in the proof of subversion zero-knowledge in Theorem 5.

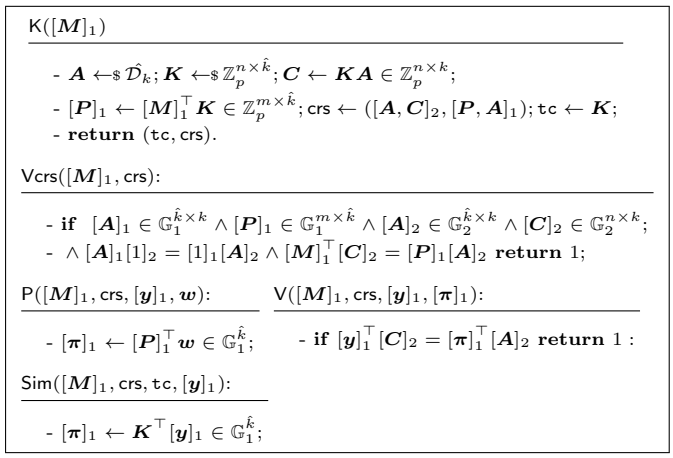

Fig. 3. Sub-QA-NIZK Π_{sub}: Sub Π_{as} ($\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k+1$) and Sub Π'_{as} ($\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$).

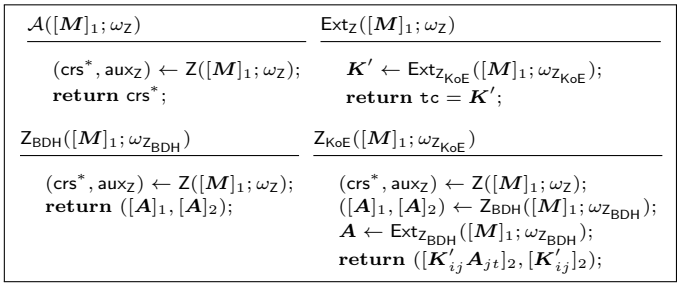

Fig. 4. The extractors and the constructed adversary $\mathcal{A}$ for Lemma 3.

Lemma 3. *For any PPT adversary $\mathcal{A}$ that outputs crs^*, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}})$ outputs $\mathtt{tc} = \boldsymbol{K}$.*

Proof. Let $\mathcal{A}$ be the adversary from Fig. 4. The subverter Z outputs crs^* such that $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$. Beside the main Z, we use some internal subverter $\mathsf{Z}_{\mathsf{BDH}}$ and $\mathsf{Z}_{\mathsf{KoE}}$ for extracting the trapdoor $\mathtt{tc}$. We note that all these subverters and the adversary are in connection and separating them is just for readability of the proof. Let $\mathsf{Z}_{\mathsf{BDH}}$ run Z and output $([\boldsymbol{A}]_1, [\boldsymbol{A}]_2)$. Then under the BDH assumption, there exists an extractor $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$ outputs $\boldsymbol{A}$. Let $\mathsf{Z}_{\mathsf{KoE}}$ run the subverter Z and $\mathsf{Z}_{\mathsf{BDH}}$ and the extractor $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$ and output $([\boldsymbol{K}'_{ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}']_2)_{ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$. More precisely, by running $\mathsf{Z}_{\mathsf{BDH}}$ and $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$, the subverter $\mathsf{Z}_{\mathsf{KoE}}$ obtains $\boldsymbol{A}$, then by solving a system of linear equations of $[\boldsymbol{C}]_2 = \boldsymbol{XA}$, she computes $[\boldsymbol{K}']_2 = \boldsymbol{X}$ such that $\boldsymbol{K}'\boldsymbol{A} = \boldsymbol{KA}$. Then she outputs the pair $([\boldsymbol{K}'_{ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}']_2)_{ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$. Finally under the KoE assumption,

there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{KoE}}})$ outputs $\mathtt{tc} = \boldsymbol{K}'$.

Theorem 5. *Let Π_{sub} be a Sub-QA-NIZK argument for linear subspaces from Fig. 3. (i) Π_{sub} is subversion complete, (ii) if the* BDH *and* KoE *assumptions hold then Π_{sub} is statistically subversion zero-knowledge, and (iii) if the $\mathcal{D}_k$-*SKerMDH *then Π_{sub} is computationally sound.*

Proof. **(i: Subversion Completeness):** This is straight forward.

(ii: Subversion Zero-Knowledge:) Let the BDH and KoE assumption hold. Let Z be a subverter that computes crs so as to break the subversion zero-knowledge property of the Sub-QA-NIZK in Fig. 3. That is, $\mathsf{Z}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}})$ outputs $(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}})$. Let $\mathcal{A}$ be the adversary from Fig. 4 of Lemma 3. Let $\mathsf{RND}(\mathcal{A}) = \mathsf{RND}(\mathsf{Z}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{BDH}}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{KoE}})$ in Lemma 3. Note that all these subverters and the adversary are in connection. Underlying Lemma 3, if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}})$ from Fig. 4 outputs $\boldsymbol{K}'$ such that $[\boldsymbol{K}']_2\boldsymbol{A} = [\boldsymbol{K}]_2\boldsymbol{A} = [\boldsymbol{C}]_2$. Due to the fact that $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ it returns $[\boldsymbol{M}]_1^\top \boldsymbol{K}' = [\boldsymbol{M}]_1^\top \boldsymbol{K}$. Finally it is clear that, $\mathsf{Ext}_{\mathsf{Z}}$ of Fig. 4 returns $\mathtt{tc} = \boldsymbol{K}'$, such that $\boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{K}'$ which is enough for simulating the proof.

Fix concrete values of λ, $\mathsf{p} \in \mathrm{im}(\mathsf{Pgen}(1^\lambda))$, $[\boldsymbol{M}]_1 \leftarrow_{\$} \mathcal{D}_{\mathsf{p}}$, $([\boldsymbol{y}]_1, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$, $\omega_{\mathsf{Z}} \in \mathsf{RND}(\mathsf{Z})$, and run $\mathsf{Ext}_{\mathsf{Z}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}})$ to obtain $\boldsymbol{K}'$. Thus, it suffices to show that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ and $([\boldsymbol{y}]_1, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ then

$$\begin{aligned}\mathsf{O}_0([\boldsymbol{y}]_1, \boldsymbol{w}) &= \mathsf{P}([\boldsymbol{M}]_1, \mathsf{crs}^*, [\boldsymbol{y}]_1, \boldsymbol{w}) = [\boldsymbol{P}]_1^\top \boldsymbol{w} \ , \\ \mathsf{O}_1([\boldsymbol{y}]_1, \boldsymbol{w}) &= \mathsf{Sim}([\boldsymbol{M}]_1, \mathsf{crs}^*, [\boldsymbol{y}]_1, \boldsymbol{K}') = \boldsymbol{K}'^\top [\boldsymbol{y}]_1\end{aligned}$$

have the same distribution. This holds since from $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ it follows that $\boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{K}'$ and from $([\boldsymbol{y}]_1; \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ it follows that $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{w}$. Thus,

$$\mathsf{O}_0([\boldsymbol{y}]_1, \boldsymbol{w}) = [\boldsymbol{P}]_1^\top \boldsymbol{w} = [\boldsymbol{K}'^\top \boldsymbol{M}\boldsymbol{w}]_1 = \boldsymbol{K}'^\top [\boldsymbol{y}]_1 = \mathsf{O}_1([\boldsymbol{y}]_1, \boldsymbol{w}) \ .$$

Hence, O_0 and O_1 have the same distribution and thus, Π_{sub} is Sub-ZK under the BDH and KoE assumptions.

(iii: Adaptive Soundness:) We prove the adaptive soundness for the two constructions subversion Π_{as} (when $\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$) and the subversion Π'_{as} (when $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ and $\hat{k} = k$, and in addition $\mathcal{D}_{\mathsf{p}}$ be witness sampleable).

Adaptive soundness proof of Π_{sub} for $\hat{\mathcal{D}}_k = \mathcal{D}_k$ (Subversion Π_{as}): The proof is similar to the soundness proof of Π_{as} in [KW15] but with some modifications. Since we added $[\boldsymbol{A}]_1$ to the CRS, we proceed to establish adaptive soundness based on the SKerMDH assumption of [GHR15] instead of the KerMDH assumption. This changes certain aspects of the proof. Assume that $\mathcal{A}$ breaks the soundness of subversion Π_{as} with probability ε. We will build an adversary $\mathcal{B}$, that breaks SKerMDH with probability $\geq \varepsilon - 1/p$.

Let $\mathcal{B}([\boldsymbol{A}]_1 \in \mathbb{G}_1^{k+1\times k}, [\boldsymbol{A}]_2 \in \mathbb{G}_2^{k+1\times k})$ generate $[\boldsymbol{M}]_1 \leftarrow_{\$} \mathcal{D}_{\mathsf{p}}$, pick $\boldsymbol{K} \leftarrow_{\$} \mathbb{Z}_p^{n\times(k+1)}$ and compute $\mathsf{crs}' = ([\boldsymbol{A}, \boldsymbol{C} = \boldsymbol{K}\boldsymbol{A}]_2, [\boldsymbol{A}, \boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{K}]_1)$ which has the same distribution as the real crs. With probability ε, $([\boldsymbol{y}]_1, [\boldsymbol{\pi}]_1) \leftarrow \mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}')$ and $\boldsymbol{y} \notin \mathrm{span}(\boldsymbol{M})$ is successful, so,

1. $\boldsymbol{y}^\top \boldsymbol{C} = \boldsymbol{\pi}^\top \boldsymbol{A}$ (V accepts). Thus, $\mathbf{0} = \boldsymbol{\pi}^\top \boldsymbol{A} - \boldsymbol{y}^\top \boldsymbol{C} = \left((\boldsymbol{\pi}^\top - \boldsymbol{y}^\top \boldsymbol{K})\right) \boldsymbol{A} = \boldsymbol{c}^\top \boldsymbol{A}$.

Based on Lemma 1, $\Pr[\boldsymbol{c} = 0] \leq 1/p$. Then $\mathcal{B}$ sets $\boldsymbol{s}_2 \leftarrow_\$ \mathbb{Z}_p^{k+1}; [\boldsymbol{s}_1]_1 \leftarrow [\boldsymbol{c} + \boldsymbol{s}_2]_1$. Finally $\mathcal{B}$ returns $([\boldsymbol{s}_1]_1, [\boldsymbol{s}_2]_2)$ as the answer to the SKerMDH problem.

Adaptive soundness proof of Π_{sub} for $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$(Subversion Π'_{as}): The proof is similar to the soundness proof of Π'_{as} in [KW15, ALSZ18] but with some modifications in a way that instead of KerMDH, similar to [ALSZ18], the soundness proof of Π'_{as} is based on the SKerMDH assumption (due to adding $[\boldsymbol{A}]_1$ to the CRS). Assume that $\mathcal{A}$ breaks the soundness of subversion Π'_{as} with probability ε. We will build an adversary $\mathcal{B}$, that breaks SKerMDH with probability $\geq \varepsilon - 1/p$.

Let $\mathcal{B}([\boldsymbol{A}]_1 \in \mathbb{G}_1^{k+1\times k}, [\boldsymbol{A}]_2 \in \mathbb{G}_2^{k+1\times k})$ generate $\boldsymbol{M} \leftarrow_\$ \mathcal{D}'_{\mathsf{p}}$. Note that the $\mathcal{D}'_{\mathsf{p}}$ exists since $\mathcal{D}_{\mathsf{p}}$ is witness sampleable. Let $\boldsymbol{M}^\perp$ be the basis for the kernel of $\boldsymbol{M}^\top$ where $\boldsymbol{M}^\top \boldsymbol{M}^\perp = \mathbf{0}$. Then it computes $[\boldsymbol{A'}]_2 = \begin{pmatrix} [\boldsymbol{A}]_2 \\ \boldsymbol{R}\cdot[\boldsymbol{A}]_2 \end{pmatrix} \in \mathbb{Z}_p^{(n-m+k)\times k}$ where $\boldsymbol{K'} \leftarrow_\$ \mathbb{G}_2^{n\times k}; \boldsymbol{R} \leftarrow_\$ \mathbb{Z}_p^{(n-m-1)\times(k+1)}$.

Let $[\bar{\boldsymbol{A}}']_2 = [\bar{\boldsymbol{A}}]_2 \in \mathbb{G}_2^{k\times k}$. Define implicitly (we do not know this value) $\boldsymbol{K} \leftarrow \boldsymbol{K'} + \boldsymbol{M}^\perp \underline{\boldsymbol{A}}' \bar{\boldsymbol{A}}^{-1} \in \mathbb{Z}_p^{n\times k}$. Thus,

$$[\boldsymbol{C}]_2 = (\boldsymbol{K'}||\boldsymbol{M}^\perp)[\boldsymbol{A'}]_2 = [\boldsymbol{K'}\bar{\boldsymbol{A}}' + \boldsymbol{M}^\perp \underline{\boldsymbol{A}}']_2 = [(\boldsymbol{K'} + \boldsymbol{M}^\perp \underline{\boldsymbol{A}}' \bar{\boldsymbol{A}}^{-1})\bar{\boldsymbol{A}}]_2 = [\boldsymbol{K}\bar{\boldsymbol{A}}]_2$$

and

$$[\boldsymbol{P}]_1 = [\boldsymbol{M}^\top \boldsymbol{K'}]_1 = [\boldsymbol{M}^\top (\boldsymbol{K} - \boldsymbol{M}^\perp \underline{\boldsymbol{A}}' \bar{\boldsymbol{A}}^{-1})]_1 = [\boldsymbol{M}^\top \boldsymbol{K}]_1 \ .$$

Thus, $\mathsf{crs}' = ([\boldsymbol{A}, \boldsymbol{C}]_2, [\boldsymbol{A}, \boldsymbol{P}]_1)$ has the same distribution as the real crs.

With probability ε, $([\boldsymbol{y}]_1, [\boldsymbol{\pi}]_1) \leftarrow \mathcal{A}([\boldsymbol{C}]_1, \mathsf{crs}')$ is successful, so, for $\boldsymbol{y} \notin \mathrm{span}(\boldsymbol{M})$ we have that $\boldsymbol{y}^\top \boldsymbol{M}^\perp \neq \mathbf{0}_{1\times(n-m)}$ and thus also $\boldsymbol{c} \neq \mathbf{0}_{n-m+k}$. Since $\mathcal{A}$ wins, $\boldsymbol{y}^\top \boldsymbol{C} = \boldsymbol{\pi}^\top \bar{\boldsymbol{A}}$. Thus,

$$\begin{aligned} \boldsymbol{\pi}^\top \bar{\boldsymbol{A}} - \boldsymbol{y}^\top \boldsymbol{C} &= \left(\boldsymbol{\pi}^\top || \mathbf{0}_{n-m}^\top\right) \boldsymbol{A'} - \boldsymbol{y}^\top \left(\boldsymbol{K'}||\boldsymbol{M}^\perp\right) \boldsymbol{A'} \\ &= \left((\boldsymbol{\pi}^\top - \boldsymbol{y}^\top \boldsymbol{K'})|| - \boldsymbol{y}^\top \boldsymbol{M}^\perp\right) \boldsymbol{A'} = \boldsymbol{c}^\top \boldsymbol{A'} = \mathbf{0} \end{aligned}$$

where $[\boldsymbol{c}]_1^\top \leftarrow [(\boldsymbol{\pi}^\top - \boldsymbol{y}^\top \boldsymbol{K'})|| - \boldsymbol{y}^\top \boldsymbol{M}^\perp]_1$. Define $[\boldsymbol{c}]_1^\top$ as $[\boldsymbol{c}_1^\top || \boldsymbol{c}_2^\top]_1$ with $[\boldsymbol{c}_1]_1 \in \mathbb{G}_1^{k+1}$ and $[\boldsymbol{c}_2]_1 \in \mathbb{G}_1^{n-m-1}$. Set $\boldsymbol{s}_2 \leftarrow_\$ \mathbb{Z}_p^{k+1}; [\boldsymbol{s}_1]_1 \leftarrow [\boldsymbol{c}_1 + \boldsymbol{R}^\top \boldsymbol{c}_2 + \boldsymbol{s}_2]_1$.

Clearly, $\boldsymbol{s}_1 - \boldsymbol{s}_2 = \boldsymbol{c}_1 + \boldsymbol{R}^\top \boldsymbol{c}_2$ and

$$(\boldsymbol{s}_1^\top - \boldsymbol{s}_2^\top)\boldsymbol{A} = (\boldsymbol{c}_1^\top + \boldsymbol{c}_2^\top \boldsymbol{R})\boldsymbol{A} = \boldsymbol{c}^\top \boldsymbol{A'} = \mathbf{0}_{1\times k} \ .$$

Since $\boldsymbol{c} \neq \mathbf{0}_{n-m+k}$ and $\boldsymbol{R}$ leaks only through $\boldsymbol{A'}$ as $\boldsymbol{RA}$,

$$\Pr[\boldsymbol{c}_1 + \boldsymbol{R}^\top \boldsymbol{c}_2 = \mathbf{0} \mid \boldsymbol{RA}] \leq 1/p \ ,$$

where the probability is over $\boldsymbol{R} \leftarrow_\$ \mathbb{Z}_p^{(n-m-1)\times(k+1)}$. Finally $\mathcal{B}$ outputs the pair $([\boldsymbol{s}_1]_1, [\boldsymbol{s}_2]_2)$ as the answer to the SKerMDH problem.

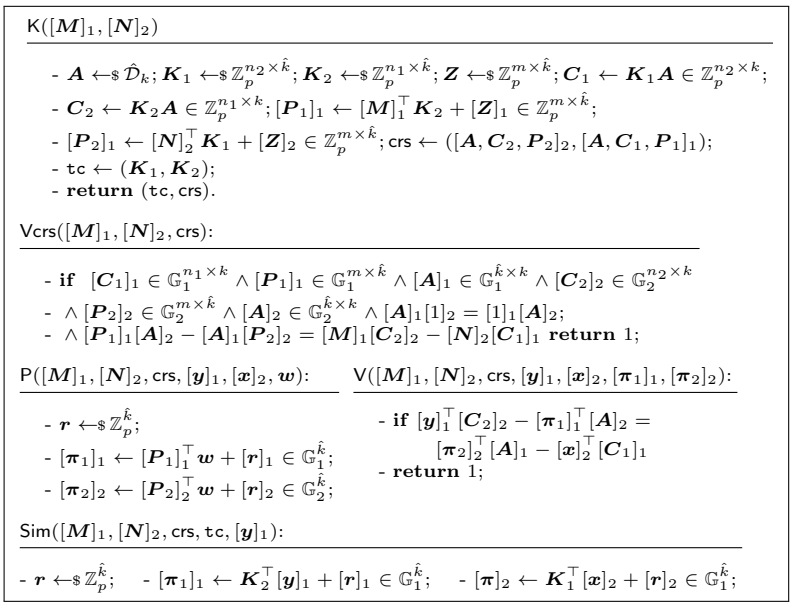

Fig. 5. Asymmetric Subversion QA-NIZK $\Pi_{\mathsf{asy\text{-}sub}}$: Sub Π_{asy} ($\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k + 1$) and Sub Π'_{asy} ($\hat{\mathcal{D}}_k = \mathcal{D}_k$ and $\hat{k} = k$).

3.3 Asymmetric QA-NIZK in the Subversion Setting

Now, we consider the asymmetric QA-NIZK argument in [GHR15] and show how one can achieve asymmetric Sub-QA-NIZK, i.e., subversion versions of Π_{asy} and Π'_{asy}. To this aim, similar to Sub-zk-SNARKs [Fuc18], we present a new Vcrs algorithm that does not require adding extra elements into the CRS. For extractability, we then again use the well-known BDH and KoE knowledge assumptions and show that if the possibly maliciously generated crs passes the Vcrs algorithm, then under the knowledge assumptions there exists an extractor that extracts the trapdoor $\mathtt{tc}$ of crs. In Theorem 6 we prove completeness and subversion zero-knowledge of the asymmetric Sub-QA-NIZKs. Since we do not add any new elements to the CRS, the soundness proof of the asymmetric Sub-QA-NIZKs will be the same as the one in [GHR15]. We depict the full construction of the asymmetric Sub-QA-NIZK arguments in Fig. 5.

We also want to stress that one can adapt the asymmetric Sub-QA-NIZKs construction in Fig. 5 to the *sum in subspace language* and obtain the subversion version of the *argument of sum in subspace* of [GHR15]. In Lemma 4, we show that from any adversary producing a valid CRS crs from scratch it is possible to extract the trapdoors $(\boldsymbol{K}_1, \boldsymbol{K}_2)$. We will use it in the proof of subversion zero-knowledge in Theorem 6.

Lemma 4. *For any PPT adversary $\mathcal{A}$ that outputs a CRS crs^*, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathcal{A}})$ outputs $\mathtt{tc} = (\boldsymbol{K}_1, \boldsymbol{K}_2)$.*

Proof. Let $\mathcal{A}$ be the adversary from Fig. 6. The subverter Z outputs

$\mathcal{A}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$

$(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}}) \leftarrow \mathsf{Z}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$;
return crs^*;

$\mathsf{Ext}_{\mathsf{Z}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$

$(\boldsymbol{K}'_1, \boldsymbol{K}'_2) \leftarrow \mathsf{Ext}_{\mathsf{Z}_{\mathsf{KoE}}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}_{\mathsf{KoE}}})$;
return $\mathtt{tc} = (\boldsymbol{K}'_1, \boldsymbol{K}'_2)$;

$\mathsf{Z}_{\mathsf{BDH}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$

$(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}}) \leftarrow \mathsf{Z}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$;
return $([\boldsymbol{A}]_1, [\boldsymbol{A}]_2)$;

$\mathsf{Z}_{\mathsf{KoE}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}_{\mathsf{KoE}}})$

$(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}}) \leftarrow \mathsf{Z}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$;
$([\boldsymbol{A}]_1, [\boldsymbol{A}]_2) \leftarrow \mathsf{Z}_{\mathsf{BDH}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$;
$\boldsymbol{A} \leftarrow \mathsf{Ext}_{\mathsf{Z}_{\mathsf{BDH}}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$;
return
$([\boldsymbol{K}'_{1,ij}\boldsymbol{A}_{jt}]_2, [\boldsymbol{K}'_{2,ij}]_2), ([\boldsymbol{K}'_{1,ij}\boldsymbol{A}_{jt}]_2, [\boldsymbol{K}'_{2,ij}]_2)$;

Fig. 6. The extractors and the constructed adversary $\mathcal{A}$ for Lemma 4.

crs^* such that $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$. For sake of simplicity, the same as Lemma 3 we assume there are some internal $\mathsf{Z}_{\mathsf{BDH}}$ and $\mathsf{Z}_{\mathsf{KoE}}$ which can compute some part of the CRS. The adversary $\mathcal{A}$ and all the subverters Z are in connection. Assume $\mathsf{Z}_{\mathsf{BDH}}$ runs Z and outputs $([\boldsymbol{A}]_1, [\boldsymbol{A}]_2)$. Then from the BDH assumption, there exists an extractor $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}([\boldsymbol{M}, [\boldsymbol{N}]_2]_1; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$ outputs $\boldsymbol{A}$. Let $\mathsf{Z}_{\mathsf{KoE}}$ runs the subverter Z and $\mathsf{Z}_{\mathsf{BDH}}$ and the extractor $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$, and outputs $([\boldsymbol{K}'_{1,ij}\boldsymbol{A}_{jt}]_1, ([\boldsymbol{K}']_1)_{1,ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$ and $([\boldsymbol{K}'_{2,ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}']_2)_{2,ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$. Roughly speaking, the subverter $\mathsf{Z}_{\mathsf{KoE}}$ runs $\mathsf{Z}_{\mathsf{BDH}}$ and $\mathsf{Ext}^{\mathsf{BDH}}_{\mathcal{A}}$, obtains $\boldsymbol{A}$. By having $\boldsymbol{A}$, and solving the system of linear equations of $([\boldsymbol{C}_1]_1, [\boldsymbol{C}_2]_2)$ (i.e $\left(\begin{smallmatrix}\boldsymbol{X}_1\\ \boldsymbol{X}_2\end{smallmatrix}\right)\left(\begin{smallmatrix}\boldsymbol{A}\end{smallmatrix}\right) = \left(\begin{smallmatrix}[\boldsymbol{C}_1]_1\\ [\boldsymbol{C}_2]_2\end{smallmatrix}\right)$), $\mathsf{Z}_{\mathsf{BDH}}$ computes $([\boldsymbol{K}'_1]_1 = \boldsymbol{X}_1, [\boldsymbol{K}_2]_1 = \boldsymbol{X}'_2)$ such that $[\boldsymbol{K}'_1]_1\boldsymbol{A} = [\boldsymbol{K}_1]_1\boldsymbol{A} = [\boldsymbol{C}_1]_1$ and $[\boldsymbol{K}'_2]_2\boldsymbol{A} = [\boldsymbol{K}_2]_2\boldsymbol{A} = [\boldsymbol{C}_2]_2$. The $\mathsf{Z}_{\mathsf{KoE}}$ finally outputs $([\boldsymbol{K}'_{1,ij}\boldsymbol{A}_{jt}]_1, ([\boldsymbol{K}']_1)_{1,ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$ and $([\boldsymbol{K}'_{2,ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}']_2)_{2,ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$. Based on KoE assumption, that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$, there exists an extractor $\mathsf{Ext}^{\mathsf{KoE}}_{\mathcal{A}}$ knowing the random coins of $\mathsf{Z}_{\mathsf{KoE}}$, outputs $(\boldsymbol{K}'_1, \boldsymbol{K}'_2)$.

Theorem 6. *Let $\Pi_{\mathsf{asy\text{-}sub}}$ be a asymmetric Sub-QA-NIZK argument for linear subspaces from Fig. 5. (i) $\Pi_{\mathsf{asy\text{-}sub}}$ is subversion complete, (ii) if the BDH and KoE assumptions hold, then $\Pi_{\mathsf{asy\text{-}sub}}$ is statistically subversion zero-knowledge, and (iii) if the $\mathcal{D}_k$-SKerMDH, (for the case $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$, the distribution $\mathcal{D}_{\mathsf{p}}$ should be WS) then $\Pi_{\mathsf{asy\text{-}sub}}$ is computationally sound.*

Proof. **(i: Subversion Completeness):** This is straight forward from the construction.

(ii: Subversion Zero-Knowledge:) Let BDH and KoE assumptions hold. Let Z be a subverter that computes crs^* so as to break the subversion zero-knowledge of Fig. 5. That is, $\mathsf{Z}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$ outputs $(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}})$. Let $\mathcal{A}$ be the same adversary as in Lemma 4. Note that $\mathsf{RND}(\mathcal{A}) = \mathsf{RND}(\mathsf{Z})$. Underlying Lemma 4, if $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$ from Fig. 6 outputs $((\boldsymbol{K}'_1, \boldsymbol{K}'_2))$ such that $[\boldsymbol{K}'_1]_1\boldsymbol{A} = [\boldsymbol{K}_1]_1\boldsymbol{A} = [\boldsymbol{C}_1]_1$ and $[\boldsymbol{K}'_2]_2\boldsymbol{A} = [\boldsymbol{K}_2]_2\boldsymbol{A} = [\boldsymbol{C}_2]_2$. Since $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$, one concludes that $[\boldsymbol{M}]_1^\top \boldsymbol{K}'_2 = [\boldsymbol{M}]_1^\top \boldsymbol{K}_2$ and $[\boldsymbol{N}]_2^\top \boldsymbol{K}'_1 = [\boldsymbol{N}]_2^\top \boldsymbol{K}_1$ which these properties are enough for simulating the proof.

Fix concrete values of λ, $\mathsf{p} \in \mathrm{im}(\mathsf{Pgen}(1^\lambda))$, $([\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1, [\boldsymbol{N}]_2}$, $\omega_{\mathsf{Z}} \in \mathsf{RND}(\mathsf{Z})$, and run $\mathsf{Ext}_{\mathsf{Z}}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2; \omega_{\mathsf{Z}})$ to obtain $(\boldsymbol{K}'_1, \boldsymbol{K}'_1)$. Thus, it suffices to show that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$ and $([\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1, [\boldsymbol{N}]_2}$ then

$$\begin{aligned}
\mathsf{O}_0([\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{w}) &= \mathsf{P}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}, [\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{w}) \ , \\
\mathsf{O}_1([\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{w}) &= \mathsf{Sim}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}, [\boldsymbol{y}]_1, [\boldsymbol{x}]_2, \boldsymbol{K}'_1, \boldsymbol{K}'_2)
\end{aligned}$$

have the same distribution. This holds since from $\mathsf{Vcrs}([\boldsymbol{M}]_1, [\boldsymbol{N}]_2, \mathsf{crs}^*) = 1$. Hence, O_0 and O_1 have the same distribution and thus, $\Pi_{\mathsf{asy\text{-}sub}}$ is Sub-ZK under BDH and KoE assumptions.

(iii: Adaptive Soundness:) If $\hat{\mathcal{D}}_k = \mathcal{D}_k$, it follows directly from the adaptive soundness proof in [GHR15]. If $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$, and $\mathcal{D}_{\mathsf{p}}$ is WS, it follows directly from the adaptive soundness proof in [GHR15].

4 Simulation Sound QA-NIZK in the Subversion Setting

In this section, we present the simulation sound Sub-QA-NIZK (SS Sub-QA-NIZK) version of the Sub-QA-NIZK from Section 3.2. Recall, that in a simulation sound QA-NIZK, even if the adversary has seen an arbitrary number of simulated proofs, she cannot come up with a new valid proof.

4.1 ℓ-time Simulation Sound Subversion QA-NIZK

We start with a construction of a ℓ-time simulation sound Sub-QA-NIZK $\Pi_{\mathsf{ls\text{-}sub}}$ from Fig. 3 with the hash key $\boldsymbol{K}$ replaced by the ℓ-wise independent hash function $\boldsymbol{H}(\tau) = \sum_{i=0}^{\ell} \tau^i \boldsymbol{K}_i$ without increasing the size of the proof. This allows arguing for ℓ-time simulation soundness similar to [KW15, ABP15] but in a subverted setting. We present the full construction of the ℓ-time SS Sub-QA-NIZK $\Pi_{\mathsf{ls\text{-}sub}}$ in Fig. 7. In Lemma 5, we show that from any adversary producing a valid CRS crs from scratch it is possible to extract the trapdoor key $\mathtt{tc} = (\boldsymbol{K}_i)_{i=0}^{i=\ell}$, which will then be used in the core idea of the proof of subversion zero-knowledge in Theorem 7.

$\mathsf{K}([\boldsymbol{M}]_1)$

- $\boldsymbol{A} \leftarrow_\$ \mathcal{D}_k$; $(\boldsymbol{K}_i)_{i=0}^{i=\ell} \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}$; $(\boldsymbol{C}_i)_{i=0}^{i=\ell} \leftarrow \boldsymbol{K}_i\boldsymbol{A} \in \mathbb{Z}_p^{n\times k}$;
- $[(\boldsymbol{P}_i)_{i=0}^{i=\ell}]_1 \leftarrow [\boldsymbol{M}]_1^\top \boldsymbol{K}_i \in \mathbb{Z}_p^{m\times(k+1)}$; $\mathsf{crs} \leftarrow ([\boldsymbol{A}, (\boldsymbol{C}_i)_{i=0}^{i=\ell}]_2, [\boldsymbol{A}, (\boldsymbol{P}_i)_{i=0}^{i=\ell}]_1)$;
- $\mathtt{tc} \leftarrow (\boldsymbol{K}_i)_{i=0}^{i=\ell}$;
- **return** $(\mathtt{tc}, \mathsf{crs})$

$\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs})$:

- **if** $[\boldsymbol{A}]_1 \in \mathbb{G}_1^{(k+1)\times k} \wedge [(\boldsymbol{P}_i)_{i=0}^{i=\ell}]_1 \in \mathbb{G}_1^{m\times k} \wedge [\boldsymbol{A}]_2 \in \mathbb{G}_2^{(k+1)\times k} \wedge [(\boldsymbol{C}_i)_{i=0}^{i=\ell}]_2 \in \mathbb{G}_2^{n\times k}$
- $\wedge\, [\boldsymbol{A}]_1[1]_2 = [1]_1[\boldsymbol{A}]_2 \wedge [\boldsymbol{M}]_1^\top [(\boldsymbol{C}_i)_{i=0}^{i=\ell}]_2 = [(\boldsymbol{P}_i)_{i=0}^{i=\ell}]_1[\boldsymbol{A}]_2$ **return** 1;

$\mathsf{P}(\tau, [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}]_1, \boldsymbol{w})$:

- $[\boldsymbol{\pi}]_1 \leftarrow (\sum_{i=0}^{i=\ell} \tau^i [\boldsymbol{P}_i]_1^\top)\boldsymbol{w} \in \mathbb{G}_1^{k+1}$;

$\mathsf{V}(\tau, [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}]_1, [\boldsymbol{\pi}]_1)$:

- **if** $[\boldsymbol{y}]_1^\top \sum_{i=0}^{i=\ell} \tau^i [\boldsymbol{C}_i]_2 = [\boldsymbol{\pi}]_1^\top [\boldsymbol{A}]_2$ **return** 1 :

$\mathsf{Sim}([\boldsymbol{M}]_1, \mathsf{crs}, \mathtt{tc}, [\boldsymbol{y}]_1)$:

- $[\boldsymbol{\pi}]_1 \leftarrow \sum_{i=0}^{i=\ell} \tau^i \boldsymbol{K_i}^\top [\boldsymbol{y}]_1 \in \mathbb{G}_1^{k+1}$;

Fig. 7. ℓ-time simulation sound Sub-QA-NIZK argument $\Pi_{\mathsf{ls\text{-}sub}}$.

Lemma 5. *For any PPT adversary A that outputs a CRS crs^*, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}(\mathsf{crs}^*; \omega)$ outputs $\mathtt{tc} = (\boldsymbol{K}_i)_{i=0}^{i=\ell}$.*

Proof. For sake of simplicity, let $i = 1$ so that $\mathtt{tc} = (\boldsymbol{K}_0, \boldsymbol{K}_1)$. The proof is similar to the proof of Lemma 3 with slightly modifications in a way that the subverters Z and $\mathsf{Z}_{\mathsf{KoE}}$ need to generate more elements. More precisely, let the adversary $\mathcal{A}$ from Fig. 4. The subverter Z outputs a valid crs^*, then $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$. Let $\mathsf{Z}_{\mathsf{BDH}}$ runs Z and outputs $([\boldsymbol{A}]_1, [\boldsymbol{A}]_2)$. Then under the BDH assumption, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$ outputs $\boldsymbol{A}$. Let $\mathsf{Z}_{\mathsf{KoE}}$ runs the subverter Z and $\mathsf{Z}_{\mathsf{BDH}}$ and the extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}$, and using $[(\boldsymbol{C}_i)_{i=0}^{i=\ell}]_2$ (with the same the technique of Lemma 3) outputs $([\boldsymbol{K}'_{0,ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}'_0]_2)_{ij})_{i\in[1,n],j\in[1,k+1],t\in[1,k]}$ and $([\boldsymbol{K}'_{1,ij}\boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}'_1]_2)_{ij})_{i\in[1,n],j\in[1,k+1],t\in[1,k]}$. Under the KoE assumption, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{KoE}}})$ outputs $(\boldsymbol{K}'_0, \boldsymbol{K}'_1)$ such that $\boldsymbol{K}'_0\boldsymbol{A} = \boldsymbol{K}_0\boldsymbol{A}$ and $\boldsymbol{K}'_1\boldsymbol{A} = \boldsymbol{K}_1\boldsymbol{A}$. Similarly one can extend it for $i \in [0, \ell]$ and extract $\mathtt{tc} = (\boldsymbol{K}_i)_{i=0}^{i=\ell}$.

Theorem 7. *Let $\Pi_{\mathsf{ls\text{-}sub}}$ be the ℓ-time SS Sub-QA-NIZK argument for linear subspaces from Fig. 7. (i) $\Pi_{\mathsf{ls\text{-}sub}}$ is subversion complete, (ii) if the KoE assumption holds, and $\mathcal{D}_{\mathsf{p}}$ is a witness samplable distribution then $\Pi_{\mathsf{ls\text{-}sub}}$ is statistically*

subversion zero-knowledge, and (iii) if the SKerMDH *then* $\Pi_{\mathsf{ls\text{-}sub}}$ *is adaptive* ℓ*-time simulation sound.*

Proof. **(i: Subversion Completeness):** This is straight forward from the construction.

(ii: Subversion Zero-Knowledge:) Let BDH and KoE assumptions hold. Let Z be a subverter that computes crs so as to break the subversion zero-knowledge of Fig. 7. That is, $\mathsf{Z}([\boldsymbol{M}]_1;\omega_{\mathsf{Z}})$ outputs $(\mathsf{crs}^*,\mathsf{aux}_{\mathsf{Z}})$. Let $\mathcal{A}$ be the adversary from Lemma 5. Let $\mathsf{RND}(\mathcal{A}) = \mathsf{RND}(\mathsf{Z}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{BDH}}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{KoE}})$ in Lemma 5. Note that all these subverters and the adversary are in connection.

Underlying Lemma 3, if $\mathsf{Vcrs}([\boldsymbol{M}]_1,\mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1;\omega_{\mathsf{Z}})$ from Lemma 5 outputs $\mathtt{tc} = (\boldsymbol{K}'_s)_{s=0}^{s=\ell}$ such that $[(\boldsymbol{K}'_s)_{s=0}^{s=\ell}]_2\boldsymbol{A} = [(\boldsymbol{K}_s)_{s=0}^{s=\ell}]_2\boldsymbol{A} = [(\boldsymbol{C}_s)_{s=0}^{s=\ell}]_2$. Since $\mathsf{Vcrs}([\boldsymbol{M}]_1,\mathsf{crs}^*) = 1$, or more precisely $[\boldsymbol{M}]_1^\top[(\boldsymbol{C}_s)_{s=0}^{s=\ell}]_2 = [(\boldsymbol{P}_s)_{s=0}^{s=\ell}]_1[\boldsymbol{A}]_2$, then we have $(\boldsymbol{M}^\top\boldsymbol{K}')_{s=0}^{s=\ell} = (\boldsymbol{M}^\top\boldsymbol{K})_{s=0}^{s=\ell} = (\boldsymbol{P}_s)_{s=0}^{s=\ell}$ which is enough for simulating the proof.

Fix concrete values of λ, $\mathsf{p} \in \mathrm{im}(\mathsf{Pgen}(1^\lambda))$, $[\boldsymbol{M}]_1 \leftarrow\!\!\$\ \mathcal{D}_{\mathsf{p}}$, $([\boldsymbol{y}]_1,\boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$, $\omega_{\mathsf{Z}} \in \mathsf{RND}(\mathsf{Z})$, and run $\mathsf{Ext}_{\mathsf{Z}}([\boldsymbol{M}]_1;\omega_{\mathsf{Z}})$ to obtain $(\boldsymbol{K})_{s=0}^{s=\ell}$. Thus, it suffices to show that if $\mathsf{Vcrs}([\boldsymbol{M}]_1,\mathsf{crs}^*) = 1$ and $([\boldsymbol{y}]_1,\boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ then

$$\mathsf{O}_0([\boldsymbol{y}]_1,\boldsymbol{w}) = \mathsf{P}([\boldsymbol{M}]_1,\mathsf{crs}^*,[\boldsymbol{y}]_1,\boldsymbol{w}) = (\sum_{s=1}^{s=\ell}\tau^s[\boldsymbol{P}_s]_1^\top)\boldsymbol{w}\ ,$$

$$\mathsf{O}_1([\boldsymbol{y}]_1,\boldsymbol{w}) = \mathsf{Sim}([\boldsymbol{M}]_1,\mathsf{crs}^*,[\boldsymbol{y}]_1,(\boldsymbol{K}'_i)_{s=0}^{s=\ell}) = \sum_{s=1}^{s=\ell}\tau^s\boldsymbol{K'_s}^\top[\boldsymbol{y}]_1$$

have the same distribution. This holds since from $\mathsf{Vcrs}([\boldsymbol{M}]_1,\mathsf{crs}) = 1$ it follows that $(\boldsymbol{P}_i)_{s=0}^{s=\ell} = \boldsymbol{M}^\top\boldsymbol{K}'_s$ and from $([\boldsymbol{y}]_1;\boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ it follows that $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{w}$. Thus,

$$\mathsf{O}_0([\boldsymbol{y}]_1,\boldsymbol{w}) = (\sum_{s=1}^{s=\ell}\tau^s[\boldsymbol{P}_s]_1^\top)\boldsymbol{w} = \sum_{s=1}^{s=\ell}\tau^s[\boldsymbol{K}'^\top_s\boldsymbol{M}]_1\boldsymbol{w} = \sum_{s=1}^{s=\ell}\tau^s\boldsymbol{K'_s}^\top[\boldsymbol{y}]_1 = \mathsf{O}_1([\boldsymbol{y}]_1,\boldsymbol{w})\ .$$

Hence, O_0 and O_1 have the same distribution and thus, $\Pi_{\mathsf{ls\text{-}sub}}$ is Sub-ZK under BDH and KoE assumptions.

(iii: Adaptive ℓ-time Simulation Soundness:) The proof is essentially a generalization of the proof of the soundness proof of subversion Π_{as} in Theorem 5. We proceed to establish adaptive ℓ-time simulation soundness based on SKerMDH assumption. Let that $\mathcal{A}$ breaks the adaptive ℓ-time simulation soundness of subversion Π_{as} in Fig. 7 with probability ε. We Construct an adversary $\mathcal{B}$, against SKerMDH with probability $\geq \varepsilon - 1/p$.

Let $\mathcal{B}([\boldsymbol{A}]_1 \in \mathbb{G}_1^{(k+1)\times k}, [\boldsymbol{A}]_2 \in \mathbb{G}_2^{(k+1)\times k})$, generates $[\boldsymbol{M}]_1 \leftarrow\!\!\$\ \mathcal{D}_{\mathsf{p}}$ and picks $(\boldsymbol{K}_i)_{i=0}^{i=\ell} \leftarrow\!\!\$\ \mathbb{Z}_p^{n\times(k+1)}$ and compute $\mathsf{crs}' = ([\boldsymbol{A},(\boldsymbol{C}_i)_{i=0}^{i=\ell} = \boldsymbol{K}_i\boldsymbol{A}]_2, [\boldsymbol{A},(\boldsymbol{P}_i)_{i=0}^{i=\ell} =$

$\boldsymbol{M}^\top \boldsymbol{K}_i]_1$) which has the same distribution as the real crs. With probability ε, $([\boldsymbol{y}]_1, [\boldsymbol{\pi}]_1) \leftarrow \mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}')$ and $\boldsymbol{y} \notin \mathrm{im}(\boldsymbol{M})$ is successful, so,

1. $\boldsymbol{y}^\top \boldsymbol{C} = \boldsymbol{\pi}^\top \boldsymbol{A}$. Thus, $\mathbf{0} = \boldsymbol{\pi}^\top \boldsymbol{A} - \boldsymbol{y}^\top \sum_{i=1}^{i=\ell} \tau^i \boldsymbol{C}_i = \left(\boldsymbol{\pi}^\top - \boldsymbol{y}^\top \sum_{i=1}^{i=\ell} \tau^i \boldsymbol{K}_i\right) \boldsymbol{A} = \boldsymbol{c}^\top \boldsymbol{A}$.

Based on Lemma 1, $\Pr[\boldsymbol{c} = 0] \leq 1/p$. Then $\mathcal{B}$ sets $\boldsymbol{s}_2 \leftarrow_\$ \mathbb{Z}_p^{k+1}; [\boldsymbol{s}_1]_1 \leftarrow [\boldsymbol{c} + \boldsymbol{s}_2]_1$. Finally $\mathcal{B}$ returns the pair $([\boldsymbol{s}_1]_1, [\boldsymbol{s}_2]_2)$ as the answer to the SKerMDH problem.

4.2 Unbounded Simulation Sound Subversion QA-NIZK

Finally, we present a SS Sub-QA-NIZK with unbounded simulation soundness denoted as $\Pi_{\mathsf{us\text{-}sub}}$. In contrast with the ℓ-time SS Sub-QA-NIZK construction, we can not use the information-theoretic techniques to have unbounded SS Sub-QA-NIZK. To this aim, we rely on the computational variant of the core Lemma 2 (cf. Section 2) which is based on the $\mathcal{D}_k$-MDDH assumption. Roughly speaking, we show how one can make the unbounded SS QA-NIZK of [KW15] subversion resistant by slightly changing the CRS to be publicly verifiable defining a new Vcrs algorithm to check whether the CRS is well-formed. Then by applying the technique from Lemma 3, we show the extractability of the CRS. We present the full construction of unbounded SS Sub-QA-NIZK in Fig. 8.

Theorem 8. *Let $\Pi_{\mathsf{us\text{-}sub}}$ be the unbounded SS Sub-QA-NIZK argument for linear subspaces from Fig. 8. (i) $\Pi_{\mathsf{us\text{-}sub}}$ is subversion complete, (ii) if the* BDH *and* KoE *assumption hold, then $\Pi_{\mathsf{us\text{-}sub}}$ is statistically subversion zero-knowledge, and (iii) if the* SKerMDH *assumption and the $\mathcal{D}_k$-*MDDH *assumption holds in $\mathbb{G}_1$ then $\Pi_{\mathsf{us\text{-}sub}}$ is adaptive unbounded simulation sound.*

Proof. **(i: Subversion Completeness):** This is straight forward.

(ii: Subversion Zero-Knowledge:) Let the BDH and KoE assumptions hold. Let Z be a subverter that computes crs^* so as to break the subversion zero-knowledge property of the construction of Fig. 8. That is, $\mathsf{Z}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}})$ outputs $(\mathsf{crs}^*, \mathsf{aux}_{\mathsf{Z}})$. Let $\mathcal{A}$ be the same adversary as in Lemma 3, Fig. 4. Note that $\mathsf{RND}(\mathcal{A}) = \mathsf{RND}(\mathsf{Z})$. More precisely $\mathsf{RND}(\mathcal{A}) = \mathsf{RND}(\mathsf{Z}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{BDH}}) = \mathsf{RND}(\mathsf{Z}_{\mathsf{KoE}})$. To be more precise, similar to the proof of Theorem 5, beside the main Z, we use some internal subverter $\mathsf{Z}_{\mathsf{BDH}}$ and $\mathsf{Z}_{\mathsf{KoE}}$ for extracting the trapdoor $\mathtt{tc}$. The subverter $\mathsf{Z}_{\mathsf{BDH}}$ runs Z and outputs $([\boldsymbol{A}]_1, [\boldsymbol{A}]_2)$. Then under the BDH assumption, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{BDH}}})$ outputs $\boldsymbol{A}$. Also $\mathsf{Z}_{\mathsf{KoE}}$ runs the subverter Z and $\mathsf{Z}_{\mathsf{BDH}}$ and the extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{BDH}}$, and outputs $([\boldsymbol{K}'_{ij} \boldsymbol{A}_{jt}]_2, ([\boldsymbol{K}']_2)_{ij})_{i\in[1,n],j\in[1,\hat{k}],t\in[1,k]}$. Under the KoE assumption, there exists an extractor $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}$, such that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\mathsf{Ext}_{\mathcal{A}}^{\mathsf{KoE}}([\boldsymbol{M}]_1; \omega_{\mathsf{Z}_{\mathsf{KoE}}})$ outputs $\boldsymbol{K}'$ such that $\boldsymbol{K}'\boldsymbol{A} = \boldsymbol{K}\boldsymbol{A} = \boldsymbol{C}$. Since $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ then $\boldsymbol{M}^\top \boldsymbol{K}' = \boldsymbol{M}^\top \boldsymbol{K}' = \boldsymbol{P}$. Finally it is clear that, $\mathsf{Ext}_{\mathsf{Z}}$ of Fig. 4 returns $\mathtt{tc} = \boldsymbol{K}'$, such that $\boldsymbol{P} = \boldsymbol{M}^\top \boldsymbol{K}'$ which is enough for simulating the proof.

$\mathsf{K}([\boldsymbol{M}]_1)$

- $\boldsymbol{A}, \boldsymbol{B} \leftarrow_\$ \mathcal{D}_k$; $\boldsymbol{K} \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}$; $\boldsymbol{K}_0, \boldsymbol{K}_1 \leftarrow_\$ \mathbb{Z}_p^{(k+1)\times(k+1)}$; $\boldsymbol{C} \leftarrow \boldsymbol{K}\boldsymbol{A} \in \mathbb{Z}_p^{n\times k}$;
- $[\boldsymbol{P}]_1 \leftarrow [\boldsymbol{M}]_1^\top \boldsymbol{K} \in \mathbb{G}_1^{m\times(k+1)}$; $(\boldsymbol{C}_i)_{i=0}^{i=1} \leftarrow \boldsymbol{K}_i\boldsymbol{A} \in \mathbb{Z}_p^{(k+1)\times k}$;
- $[(\boldsymbol{P}_i)_{i=0}^{i=1}]_1 \leftarrow [\boldsymbol{B}]_1^\top \boldsymbol{K}_i \in \mathbb{G}_1^{k\times(k+1)}$;
- $\mathsf{crs} \leftarrow ([\boldsymbol{A}, \boldsymbol{C}, (\boldsymbol{C}_i)_{i=0}^{i=1}]_2, [\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{P}, (\boldsymbol{P}_i)_{i=0}^{i=1}]_1; \mathtt{tc} \leftarrow \boldsymbol{K}$;
- **return** $(\mathtt{tc}, \mathsf{crs})$.

$\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs})$:

- **if** $[\boldsymbol{A}]_1 \in \mathbb{G}_1^{(k+1)\times k} \wedge [\boldsymbol{B}]_2 \in \mathbb{G}_2^{(k+1)\times k} \wedge [\boldsymbol{P}]_1 \in \mathbb{G}_1^{m\times k} \wedge [\boldsymbol{A}]_2 \in \mathbb{G}_2^{(k+1)\times k}$
- $\wedge\, [\boldsymbol{C}]_2 \in \mathbb{G}_2^{n\times k} \wedge\ [(\boldsymbol{P}_i)_{i=0}^{i=1}]_1 \in \mathbb{G}_1^{k\times(k+1)} \wedge [(\boldsymbol{C}_i)_{i=0}^{i=1}]_2 \in \mathbb{G}_2^{(k+1)\times k}$
- $\wedge\, [\boldsymbol{A}]_1[1]_2 = [1]_1[\boldsymbol{A}]_2 \wedge [\boldsymbol{M}]_1^\top[\boldsymbol{C}]_2 = [\boldsymbol{P}]_1[\boldsymbol{A}]_2$
- $\wedge\ [\boldsymbol{B}]_1^\top[(\boldsymbol{C}_i)_{i=0}^{i=1}]_2 = [(\boldsymbol{P}_i)_{i=0}^{i=1}]_1[\boldsymbol{A}]_2$ **return** 1;

$\mathsf{P}(\tau, [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}]_1, \boldsymbol{w})$:

- $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^k$;
- $[\boldsymbol{\pi}]_1 \leftarrow ([\boldsymbol{\pi}_1]_1, [\boldsymbol{\pi}_2]_1) \leftarrow ([\boldsymbol{P}]_1^\top \boldsymbol{w} + (\sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top)\boldsymbol{r}, [\boldsymbol{B}]_1\boldsymbol{r}) \in (\mathbb{G}_1^{k+1})^2$;

$\mathsf{V}(\tau, [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}]_1, [\boldsymbol{\pi}]_1)$:

- Parse $\boldsymbol{\pi} = (\boldsymbol{\pi}_1, \boldsymbol{\pi}_2)$;
- **if** $[\boldsymbol{y}]_1^\top[\boldsymbol{C}]_2 + [\boldsymbol{\pi}_2]_1^\top \sum_{i=0}^{i=1} \tau^i [\boldsymbol{C}_i]_2 = [\boldsymbol{\pi}_1]_1^\top[\boldsymbol{A}]_2$ **return** 1.

$\mathsf{Sim}(\tau, [\boldsymbol{M}]_1, \mathsf{crs}, \mathtt{tc}, [\boldsymbol{y}]_1)$:

- $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^k$;
- $[\boldsymbol{\pi}]_1 = ([\boldsymbol{\pi}_1]_1, [\boldsymbol{\pi}_2]_1) \leftarrow (\boldsymbol{K}^\top[\boldsymbol{y}]_1 + \sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top \boldsymbol{r}, [\boldsymbol{B}]_1\boldsymbol{r}) \in (\mathbb{G}_1^{k+1})^2$;

Fig. 8. Unbounded simulation sound Sub-QA-NIZK argument $\Pi_{\mathsf{us\text{-}sub}}$.

Fix concrete values of λ, $\mathsf{p} \in \mathrm{im}(\mathsf{Pgen}(1^\lambda))$, $[\boldsymbol{M}]_1 \leftarrow_\$ \mathcal{D}_\mathsf{p}$, $([\boldsymbol{y}]_1, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$, $\omega_\mathsf{Z} \in \mathsf{RND}(\mathsf{Z})$, and run $\mathsf{Ext}_\mathsf{Z}([\boldsymbol{M}]_1; \omega_\mathsf{Z})$ to obtain $\boldsymbol{P} = \boldsymbol{M}^\top\boldsymbol{K}'$. Thus, it suffices to show that if $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ and $([\boldsymbol{y}]_1, \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ then for any $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^k$,

$$\mathsf{O}_0([\boldsymbol{y}]_1, \boldsymbol{w}) = \mathsf{P}([\boldsymbol{M}]_1, \mathsf{crs}^*, [\boldsymbol{y}]_1, \boldsymbol{w}) = ([\boldsymbol{P}]_1^\top \boldsymbol{w} + (\sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top)\boldsymbol{r}, [\boldsymbol{B}]_1\boldsymbol{r}) \ ,$$

$$\mathsf{O}_1([\boldsymbol{y}]_1, \boldsymbol{w}) = \mathsf{Sim}(\boldsymbol{M}, \mathsf{crs}^*, [\boldsymbol{y}]_1, \boldsymbol{K}') = (\boldsymbol{K}'^\top[\boldsymbol{y}]_1 + \sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top \boldsymbol{r}, [\boldsymbol{B}]_1\boldsymbol{r})$$

have the same distribution. This holds since from $\mathsf{Vcrs}([\boldsymbol{M}]_1, \mathsf{crs}^*) = 1$ it follows that $\boldsymbol{P} = \boldsymbol{M}^\top\boldsymbol{K}'$ and from $([\boldsymbol{y}]_1; \boldsymbol{w}) \in \mathcal{R}_{[\boldsymbol{M}]_1}$ it follows that $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{w}$. It is easy to see that the second part $[\boldsymbol{B}]_1\boldsymbol{r}$ of the output of O_0 and O_1 are equal. Thus we need to argue about the first part of the proof by O_0 and O_1, and it follows from the fact $\boldsymbol{M}^\top\boldsymbol{K}' = \boldsymbol{M}^\top\boldsymbol{K} = \boldsymbol{P}$, which one concludes,

$$[\boldsymbol{P}]_1^\top \boldsymbol{w} + \sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top \boldsymbol{r} = [\boldsymbol{K}^\top \boldsymbol{M}]_1 \boldsymbol{w} + (\sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top) \boldsymbol{r} =$$

$$\boldsymbol{K'}^\top [\boldsymbol{y}]_1 + \sum_{i=0}^{i=1} \tau^i [\boldsymbol{P}_i]_1^\top \boldsymbol{r}$$

Therefore, O_0 and O_1 have the same distribution and thus, $\Pi_{\mathsf{us\text{-}sub}}$ is Sub-ZK under BDH and KoE assumptions.

(iii: Adaptive Unbounded Simulation Soundness:) The proof follows the lines to prove the unbounded simulation soundness proof in [KW15], but since we add new elements $[\boldsymbol{A}]_1$ to have publicly verifiable CRS, the proof contains some modifications that instead of KerMDH, some part of this proof is based on the SKerMDH assumption.

Proof. Assume that $\mathcal{A}$ makes at most Q queries to Sim and breaks the unbounded simulation soundness of subversion $\Pi_{\mathsf{us\text{-}sub}}$ in Fig. 8 with probability ε. We prove it by building a sequence of games transitioning from the real game Game_0 to the Game_3.

$\underline{\mathsf{Game}_0}$**.** In this game, the adversary $\mathcal{A}$ sees a pair $([\boldsymbol{M}]_1, \mathsf{crs})$ and it follows the definition of simulation soundness in Section 2. In particular:

- $\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda)$; $\boldsymbol{M} \in \mathbb{Z}_p^{n \times m} \leftarrow_\$ \mathcal{D}_\mathsf{p}$; $(\mathsf{crs}, \mathsf{tc}) \leftarrow \mathsf{K}([\boldsymbol{M}]_1)$;
- $([\pi']_1, [\boldsymbol{y}']_1, \tau') \leftarrow \mathcal{A}^{\mathsf{Sim}}([\boldsymbol{M}]_1, \mathsf{crs})$;
- If $[\boldsymbol{y}']_1 \notin \mathcal{L}_{[\boldsymbol{M}]_1} \wedge \tau' \notin Q_\tau, \wedge\ \mathsf{V}(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi'}]_1)\ :\ [\boldsymbol{y}']_1^\top [\boldsymbol{C}]_2 + [\boldsymbol{\pi}'_2]_1^\top \sum_{i=0}^{i=1} \tau'^i [\boldsymbol{C}_i]_2 = [\boldsymbol{\pi}'_1]_1^\top [\boldsymbol{A}]_2$ **return** 1;
- **Return** $\mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}, [\pi']_1, [\boldsymbol{y}']_1, \tau')$.

$\underline{\mathsf{Game}_1}$**.** This game is the same as Game_0, the only difference is that we replace the $\mathsf{V}^*(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi'}]_1)$ instead of $\mathsf{V}(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi'}]_1)$. Thus, Game_1 is as follows:

- $\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda)$; $\boldsymbol{M} \in \mathbb{Z}_p^{n \times m} \leftarrow_\$ \mathcal{D}_\mathsf{p}$; $(\mathsf{crs}, \mathsf{tc}) \leftarrow \mathsf{K}([\boldsymbol{M}]_1)$;
- $([\pi']_1, [\boldsymbol{y}']_1, \tau') \leftarrow \mathcal{A}^{\mathsf{Sim}}([\boldsymbol{M}]_1, \mathsf{crs})$;
- If $[\boldsymbol{y}']_1 \notin \mathcal{L}_{[\boldsymbol{M}]_1} \wedge \tau' \notin Q_\tau$,
- $\wedge \mathsf{V}^*(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi'}]_1)$:
  $\boxed{[\boldsymbol{\pi}'_1]_1^\top = [\boldsymbol{y}']_1^\top \boldsymbol{K} + [\boldsymbol{\pi}'_2]_1^\top \sum_{i=0}^{i=1} \tau'^i \boldsymbol{K}_i}$ **return** 1;
- **Return** $\mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}, [\pi']_1, [\boldsymbol{y}']_1, \tau')$.

$\underline{\mathsf{Game}_0 \to \mathsf{Game}_1}$**.** We observe that the verification equation V in Game_0 is:

$$[\boldsymbol{y}']_1^\top [\boldsymbol{C}]_2 + [\boldsymbol{\pi}'_2]_1^\top \sum_{i=0}^{i=1} \tau'^i [\boldsymbol{C}_i]_2 = [\boldsymbol{\pi}'_1]_1^\top [\boldsymbol{A}]_2;$$

$$([\boldsymbol{y}']_1^\top \boldsymbol{K} + [\boldsymbol{\pi}'_2]_1^\top \sum_{i=0}^{i=1} \tau'^i \boldsymbol{K}_i - [\boldsymbol{\pi}'_1]_1^\top)[\boldsymbol{A}]_2 = [c]_1 [\boldsymbol{A}]_2 = \boldsymbol{0},$$

which for any pair $([\boldsymbol{y}']_1, [\pi]'_1)$ passes V but not V^*, then the values $([s_1]_1, [s_2]_2)$ where $[s_1]_1 = [c + s_2]_1$ and $s_2 \leftarrow_\$ \mathbb{Z}_p^{1\times(k+1)}$ are non-zero vectors and the answer to the SKerMDH problem. Thus, it concludes that Game_0 and Game_1 are indistinguishable if the SKerMDH problem is hard.

Game_2**.** This game is as Game_1, but we slightly change the structure of $\boldsymbol{\pi}'$. Let $\boldsymbol{A}^\perp$ be an element from the kernel of $\boldsymbol{A}$. In this game we replace the $[\boldsymbol{\pi}']_1$ with $[\boldsymbol{\pi}']_1 = ([\boldsymbol{y}'^\top \boldsymbol{K} + r'\boldsymbol{A}^\perp + (\sum_{i=0}^{i=1} \tau'^i \boldsymbol{P}_i^\top)\boldsymbol{r}]_1, [\boldsymbol{B}]_1\boldsymbol{r}) \in (\mathbb{G}_1^{k+1})^2$ where $r' \leftarrow_\$ \mathbb{Z}_p$. Thus, Game_2 is as follows:

- $\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \boldsymbol{M} \in \mathbb{Z}_p^{n\times m} \leftarrow_\$ \mathcal{D}_\mathsf{p}$; $(\mathsf{crs}, \mathtt{tc}) \leftarrow \mathsf{K}([\boldsymbol{M}]_1)$;
- $(\boxed{[\pi']_1}, [\boldsymbol{y}']_1, \tau') \leftarrow \mathcal{A}^{\mathsf{Sim}}([\boldsymbol{M}]_1, \mathsf{crs})$;
- If $[\boldsymbol{y}']_1 \notin \mathcal{L}_{[\boldsymbol{M}]_1} \wedge \tau' \notin Q_\tau$,
- $\wedge \mathsf{V}^*(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi}']_1)$:
  $[\boldsymbol{\pi}'_1]_1^\top = [\boldsymbol{y}']_1^\top \boldsymbol{K} + r'\boldsymbol{A}^\perp + [\boldsymbol{\pi}'_2]_1^\top \sum_{i=0}^{i=1} \tau'^i \boldsymbol{K}_i$ **return** 1;
- **Return** $\mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}, [\pi']_1, [\boldsymbol{y}']_1, \tau')$.

$\underline{\mathsf{Game}_1 \to \mathsf{Game}_2}$**.** In this case, we can choose $\boldsymbol{K} \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}$ and when $\mathcal{A}$ queries $([\boldsymbol{y}']_1, \tau')$ and $\tau \neq \tau'$, The oracle O_b works either the same as the Sim in Game_1 (for $b = 0$) or the Sim in Game_2 for $b = 1$. When $\mathcal{A}$ queries $([\boldsymbol{y}']_1, \tau)$ and $\tau = \tau'$, it picks $\boldsymbol{r} \leftarrow_\$ \mathbb{Z}_p^k$ and outputs $([\boldsymbol{y}']_1^\top \boldsymbol{K} + \sum_{i=0}^{i=1} \tau'^i [\boldsymbol{P}_i]_1^\top \boldsymbol{r}, [\boldsymbol{B}]_1\boldsymbol{r})$. Since $\mathcal{D}_\mathsf{p}$ is a witness sampleable distribution, the condition of $\mathcal{A}$'s winning can be efficiently verified for a given $[\boldsymbol{y}']_1$ and $\boldsymbol{M} \in \mathbb{Z}_p^{n\times m}$, by checking $[\boldsymbol{y}']_1^\top \boldsymbol{M}^\perp \neq \boldsymbol{0}$ iff $[\boldsymbol{y}']_1 \in \mathcal{L}$. Thus from Lemma 2, we heave that the probability of distinguishing Game_1 and Game_2 is $\leq 2Q \cdot \varepsilon_{\mathrm{MDDH}} + Q/p$.

$\underline{\mathsf{Game}_3}$**.** The difference with Game_2 is that we now choose $\boldsymbol{\zeta} \leftarrow_\$ \mathbb{Z}_p^n$ and $\boldsymbol{K}' \leftarrow_\$ \mathbb{Z}_p^{n\times(k+1)}$ and replace the original $\boldsymbol{K}$ used in crs with $\boldsymbol{K} = \boldsymbol{K}' + \boldsymbol{\zeta}\boldsymbol{A}^\perp$. Thus, Game_2 is as follows:

- $\mathsf{p} \leftarrow \mathsf{K}_0(1^\lambda); \boldsymbol{M} \in \mathbb{Z}_p^{n\times m} \leftarrow_\$ \mathcal{D}_\mathsf{p}$; $(\boxed{\mathsf{crs}}, \mathtt{tc}) \leftarrow \mathsf{K}([\boldsymbol{M}]_1)$;
- $([\pi']_1, [\boldsymbol{y}']_1, \tau') \leftarrow \mathcal{A}^{\mathsf{Sim}}([\boldsymbol{M}]_1, \mathsf{crs})$;
- If $[\boldsymbol{y}']_1 \notin \mathcal{L}_{[\boldsymbol{M}]_1} \wedge \tau' \notin Q_\tau$,
- $\wedge \mathsf{V}^*(\tau', [\boldsymbol{M}]_1, \mathsf{crs}, [\boldsymbol{y}']_1, [\boldsymbol{\pi}']_1)$:
  $[\boldsymbol{\pi}_1]_1^\top = [\boldsymbol{y}]_1^\top \boldsymbol{K} + r'\boldsymbol{A}^\perp + [\boldsymbol{\pi}_2]_1^\top \sum_{i=0}^{i=1} \tau'^i \boldsymbol{K}_i$ **return** 1;
- **Return** $\mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}, [\pi']_1, [\boldsymbol{y}']_1, \tau')$.

$\underline{\mathsf{Game}_2 \to \mathsf{Game}_3}$**.** We show that these games are indistinguishable via an information-theoretic argument. Indeed for the crs in Game_3, the value $\boldsymbol{P} = \boldsymbol{M}^\top(\boldsymbol{K}' + \boldsymbol{\zeta}\boldsymbol{A}^\perp)$ leaks $\boldsymbol{M}^\top\boldsymbol{\zeta}\boldsymbol{A}^\perp$, but $\boldsymbol{C} = (\boldsymbol{K}' + \zeta\boldsymbol{A}^\perp)\boldsymbol{A} = \boldsymbol{K}'\boldsymbol{A}$ which completely hides $\boldsymbol{\zeta}$. Also in the first part of proof $\boldsymbol{\pi}'_1$, this is $(\boldsymbol{y}^\top\boldsymbol{K}' + \boldsymbol{y}\boldsymbol{\zeta}\boldsymbol{A}^\perp) + r'\boldsymbol{A}^\perp$ and since $\boldsymbol{y}\boldsymbol{\zeta}\boldsymbol{A}^\perp$ is masked by $r' \leftarrow_\$ \mathbb{Z}_p$, thus which this part is identically distributed to $\boldsymbol{y}^\top\boldsymbol{K}' + r'\boldsymbol{A}^\perp$. Then in order to pass V^* one needs to compute this part of the proof $\boldsymbol{\pi}$ with $(\boldsymbol{y}'^\top(\boldsymbol{K}' + \boldsymbol{\zeta}\boldsymbol{A}^\perp))$ for $\boldsymbol{y}'$ and so $\boldsymbol{y}'^\top\boldsymbol{\zeta} \in \mathbb{Z}_p$. Thus, given $\boldsymbol{M}^\top\boldsymbol{\zeta}\boldsymbol{A}^\perp$ for any adaptively chosen $\boldsymbol{y}'$ not in the span of $\boldsymbol{M}$, from $\mathcal{A}$'s point of view, $\boldsymbol{y}'^\top\boldsymbol{\zeta}$ is uniformly random over $\mathbb{Z}_p$. Therefore, probability of distinguishing Game_2 and Game_3 is $\leq 1/p$.

5 Integrating Sub-QA-NIZK into LegoSNARK

In this section, we show how to integrate our subversion QA-NIZK into the LeogSNARK framework of Commit-Prove zk-SNARKs(CP-SNARKs) [CFQ19], which so far does not consider subverted CRS generation. LegoSNARK uses a knowledge-sound version of the Kiltz-Wee QA-NIZK Π'_{as}. They, then show how to use this QA-NIZK to construct CP-SNARKs that work for any commitment scheme whose verification algorithm is the same as the generalized Pedersen commitment and present two schemes. The first scheme $\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{link}}$ allows proving that commitments under different keys open to the same vector and the second more general scheme $\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$ allows proving the correctness of a linear function of a committed vector. We will demonstrate how our Sub-QA-NIZK that we transform into Sub-CP-SNARKs can be used to construct a subversion variant of the more general $\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$ (called Sub-$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$), but we note that our results can be applied equivalently to the more specific first scheme. Technically, we, therefore, need to show that our Π_{sub} based on Π'_{as} is knowledge-sound. With regard to the potentially malicious generation of the respective commitment keys, as mentioned in [CFQ19] for Pedersen commitments they can easily be sampled in a transparent way such that no trusted setup is needed, e.g., by deriving them using a suitable hash function modelled as a random oracle. Consequently, we obtain a subversion variant of LegoSNARK for the QA-NIZK part and stress that using other recent results on subversion zk-SNARKs in [Lip19, Bag19, ARS20], one can further extend the toolbox of a subversion variant of the LegoSNARK framework.

We now demonstrate how to construct a Sub-CP-SNARK for the linear relation $\mathcal{R}^{Lin}$, which checks linear properties of some committed vectors: for a fixed public matrix $\boldsymbol{M} \in \mathbb{Z}_p^{n\times m}$, relation $\mathcal{R}^{Lin}_{\boldsymbol{M}}$ over public input $[\boldsymbol{y}]_1 \in \mathbb{G}_1^n$ and witness $\boldsymbol{w} \in \mathbb{Z}_p^m$, with $\boldsymbol{w} := (\boldsymbol{w}_j)_{j\in[\ell]}$ and $\boldsymbol{w}_j \in \mathbb{Z}_p^{n_j}$, holds iff $[\boldsymbol{y}]_1 = [\boldsymbol{M}]_1\boldsymbol{w}$.

For simplicity, we mostly use the notations used in [CFQ19]. Let Com be a commitment scheme such that $\mathsf{Com.VerCommit} = \mathsf{Ped.VerCommit}$. Let $\mathsf{pk} = [\boldsymbol{h}]_1 \in \mathbb{G}_1^{n+1}$ be the key of the global commitment Com. In our subversion $\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$, the public inputs of the prover are ℓ commitments $(c_j)_{j\in[\ell]}$ and another commitment $\boldsymbol{c}'$; the witness is a set of openings $((\boldsymbol{w}_j)_{j\in[\ell]}; (o_j)_{j\in[\ell]})$ for commitments $(c_j)_{j\in[\ell]}$, and an opening $\boldsymbol{o}'$ for $\boldsymbol{c}'$. In particular, the prover must prove the following relation,

$$\mathcal{R}^{\mathsf{lin}}_{\mathsf{ped}}(\boldsymbol{c}', (c_j)_{j=1}^{\ell}, (\boldsymbol{w}_j)_{j=1}^{\ell}, (o_j)_{j=1}^{\ell}, o') = 1 \iff$$
$$\bigwedge_{j=1}^{\ell} c_j = (o_j, \boldsymbol{w}_j^\top) \cdot [\boldsymbol{h}_{[0\,..\,n_j]}]_\iota \wedge \boldsymbol{c}' = (o', \boldsymbol{w}_1^\top, \ldots, \boldsymbol{w}_\ell^\top) \cdot [\boldsymbol{M}]_\iota \ .$$

Our scheme, called subversion Commit-Prove (Sub-$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$), is quite similar to $\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$ of [CFQ19] but it uses a Sub QA-NIZK in the prove phase. The Sub-$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$ essentially consists of the following algorithms:

$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}.\mathsf{K}(\mathcal{R}^{\mathsf{Lin}}_{\boldsymbol{M}}, \mathsf{pk})$**:** parse $\mathsf{pk} = [\boldsymbol{h}]_1 \in \mathbb{G}_1^{m+1}$. Use $[\boldsymbol{h}]_1$ and $\mathcal{R}^{Lin}_{\boldsymbol{M}}$ to construct $[\boldsymbol{M}^*]_1$ as in Eq. (1). Run $(\mathsf{crs}, \mathsf{tc}) \leftarrow \Pi_{\mathsf{sub}}.\mathsf{K}([\boldsymbol{M}^*]_1)$. Return $(\mathsf{crs}, \mathsf{tc})$.

$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}.\mathsf{Vcrs}([\boldsymbol{M}^*]_1, \mathsf{crs})$**:** return $\Pi_{\mathsf{sub}}.\mathsf{Vcrs}([\boldsymbol{M}^*]_1, \mathsf{crs})$.
$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}.\mathsf{P}([\boldsymbol{M}^*]_1, \mathsf{crs}, [\boldsymbol{y}^*]_1, \boldsymbol{w}^*)$**:** return $\pi \leftarrow \Pi_{\mathsf{sub}}.\mathsf{P}(\boldsymbol{M}^*, \mathsf{crs}, [\boldsymbol{y}^*]_1, \boldsymbol{w}^*)$.
$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}.\mathsf{V}([\boldsymbol{M}^*]_1, \mathsf{crs}, [\boldsymbol{y}^*]_1, \pi)$**:** return $\Pi_{\mathsf{sub}}.\mathsf{V}([\boldsymbol{M}^*]_1, \mathsf{crs}, \boldsymbol{y}^*, \pi)$.

Notice that the scheme Sub-$\mathsf{CP}^{\mathsf{Ped}}_{\mathsf{lin}}$ considers each $\boldsymbol{w}_j$ to be committed using a Pedersen commitment scheme whose key is $\mathsf{pk} = [\boldsymbol{h}]_1 \in \mathbb{G}_1^{m+1}$. The general idea is to express such a commit-and-prove relation with the linear subspace relation $\mathcal{R}_{[\boldsymbol{M}^*]_1}(x^*, \boldsymbol{w}^*)$ that holds iff $[\boldsymbol{y}^*]_1 = [\boldsymbol{M}^*]_1 \boldsymbol{w^*}$, where $[\boldsymbol{y}^*]_1 \in \mathbb{G}_1^l$, $[\boldsymbol{M}^*]_1 \in \mathbb{G}_1^{l \times t}$, and $\boldsymbol{w}^* \in \mathbb{Z}_p^t$ can be built from the inputs of $\mathcal{R}_{\boldsymbol{F}}^{Lin}$ for $l = \ell + n$ and $t = m + \ell$, as follows:

$$\overbrace{\begin{pmatrix} c_1 \\ \vdots \\ c_\ell \\ \boldsymbol{y} \end{pmatrix}}^{\boldsymbol{y}^*} = \overbrace{\left(\begin{array}{cccc|ccc} h_0 & 0 & \cdots & 0 & h_{[1,n_1]} & \cdots & 0 \\ 0 & h_0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_0 & 0 & \cdots & h_{[1,n_\ell]} \\ \hline 0 & 0 & \cdots & 0 & \boldsymbol{M} & \cdots & \boldsymbol{M} \end{array}\right)}^{\boldsymbol{M}^*} \overbrace{\begin{pmatrix} o_1 \\ \vdots \\ o_\ell \\ \boldsymbol{w} \end{pmatrix}}^{\boldsymbol{w}^*} \tag{1}$$

Subsequently, we show that we can obtain a Sub-CP-SNARK suitable for LegoSNARK when using a suitable knowledge-sound Sub-QA-NIZK Π_{sub}.

Theorem 9. *Let $\boldsymbol{M} \in \mathbb{Z}_p^{n \times m}$ be a matrix from a distribution $\mathcal{D}_{\mathsf{p}}$, and* aux *be an auxiliary input distribution. If Π_{sub} is subversion zero-knowledge and knowledge sound, then the Sub-*$\mathsf{CP} - \mathsf{SNARK}$ *construction Sub-CP_{lin}^{Ped} given above is (i) subversion zero-knowledge and (ii) knowledge sound.*

Proof. **(i: Subversion Zero-knowledge):** This is straight forward from subversion zero-knowledge of Fig. 3 in Theorem 5.

(ii: Knowledge Soundness): The proof is given in Appendix A.

Remarks. LegoSNARK does not consider the integration of the asymmetric QA-NIZK (Π'_{asy}) by González et al. [GHR15]. We note, however, that this can be done analogously to the integration of Π'_{as}, which further helps to increase the expressiveness for languages supported by QA-NIZKs in LegoSNARK. Furthermore, we want to remark that our subversion version of Π'_{asy} can be integrated into LegoSNARK analogously to the integration of the subversion version of Π'_{as}.

Acknowledgements. This work was supported by the EUs Horizon 2020 ECSEL Joint Undertaking under grant agreement n°783119 (SECREDAS), by the Austrian Science Fund (FWF) and netidee SCIENCE under grant agreement P31621-N38 (PROFET) and the Estonian Research Council grant PRG49.

References

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

ABP15. Michel Abdalla, Fabrice Benhamouda, and David Pointcheval. Disjunctions for hash proof systems: New constructions and applications. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 69–100. Springer, Heidelberg, April 2015.

AJO+19. Masayuki Abe, Charanjit S. Jutla, Miyako Ohkubo, Jiaxin Pan, Arnab Roy, and Yuyu Wang. Shorter QA-NIZK and SPS with tighter security. In Steven D. Galbraith and Shiho Moriai, editors, *ASIACRYPT 2019, Part III*, volume 11923 of *LNCS*, pages 669–699. Springer, Heidelberg, December 2019.

AJOR18. Masayuki Abe, Charanjit S. Jutla, Miyako Ohkubo, and Arnab Roy. Improved (almost) tightly-secure simulation-sound QA-NIZK with applications. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part I*, volume 11272 of *LNCS*, pages 627–656. Springer, Heidelberg, December 2018.

ALSZ18. Behzad Abdolmaleki, Helger Lipmaa, Janno Siim, and Michal Zajac. On qa-nizk in the bpk model. Cryptology ePrint Archive, Report 2018/877, 2018. https://eprint.iacr.org/2018/877.

ARS20. Behzad Abdolmaleki, Sebastian Ramacher, and Daniel Slamanig. Lift-and-shift: Obtaining simulation extractable subversion and updatable snarks generically. Cryptology ePrint Archive, Report 2020/062, 2020. https://eprint.iacr.org/2020/062.

Bag19. Karim Baghery. Subversion-resistant simulation (knowledge) sound nizks. In *IMA International Conference on Cryptography and Coding*, pages 42–63. Springer, 2019.

BFS16. Mihir Bellare, Georg Fuchsbauer, and Alessandra Scafuro. NIZKs with an untrusted CRS: Security in the face of parameter subversion. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 777–804. Springer, Heidelberg, December 2016.

Buc17. Jon Buck. Ethereum upgrade byzantium is live, verifes'first zk-snark proof, 2017. https://cointelegraph.com/news/ethereum-upgrade-byzantium-is-live-verifies-first-zk-snark-proof.

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.

Dam92. Ivan Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 445–456. Springer, Heidelberg, August 1992.

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.

EHK+13. Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Heidelberg, August 2013.

FKL18. Georg Fuchsbauer, Eike Kiltz, and Julian Loss. The algebraic group model and its applications. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 33–62. Springer, Heidelberg, August 2018.

Fuc18. Georg Fuchsbauer. Subversion-zero-knowledge SNARKs. In Michel Abdalla and Ricardo Dahab, editors, *PKC 2018, Part I*, volume 10769 of *LNCS*, pages 315–347. Springer, Heidelberg, March 2018.

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.

GHKW16. Romain Gay, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Tightly CCA-secure encryption without pairings. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 1–27. Springer, Heidelberg, May 2016.

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

GMR89. Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on computing*, 18(1):186–208, 1989.

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.

Gro06. Jens Groth. Simulation-sound NIZK proofs for a practical language and constant size group signatures. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT 2006*, volume 4284 of *LNCS*, pages 444–459. Springer, Heidelberg, December 2006.

Gro10. Jens Groth. Short pairing-based non-interactive zero-knowledge arguments. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 321–340. Springer, Heidelberg, December 2010.

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.

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.

JR13a. 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.

JR13b. Charanjit S. Jutla and Arnab Roy. Shorter quasi-adaptive NIZK proofs for linear subspaces. Cryptology ePrint Archive, Report 2013/109, 2013. http://eprint.iacr.org/2013/109.

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.

Kil92. Joe Kilian. A note on efficient zero-knowledge proofs and arguments (extended abstract). In *24th ACM STOC*, pages 723–732. ACM Press, May 1992.

KS19. Mojtaba Khalili and Daniel Slamanig. Efficient tightly-secure structure-preserving signatures and unbounded simulation-sound qa-nizk proofs. Cryptology ePrint Archive, Report 2019/1026, 2019. https://eprint.iacr.org/2019/1026.

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.

Lip12. Helger Lipmaa. Progression-free sets and sublinear pairing-based non-interactive zero-knowledge arguments. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 169–189. Springer, Heidelberg, March 2012.

Lip19. Helger Lipmaa. Simulation-extractable snarks revisited. Cryptology ePrint Archive, Report 2019/612, 2019. https://eprint.iacr.org/2019/612.

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.

LPJY15. Benoît Libert, Thomas Peters, Marc Joye, and Moti Yung. Compactly hiding linear spans - tightly secure constant-size simulation-sound QA-NIZK proofs and applications. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part I*, volume 9452 of *LNCS*, pages 681–707. Springer, Heidelberg, November / December 2015.

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.

Sah99. Amit Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *40th FOCS*, pages 543–553. IEEE Computer Society Press, October 1999.

Sah01. Amit Sahai. Simulation-sound non-interactive zero knowledge. Technical report, IBM RESEARCH REPORT RZ 3076, 2001.

SCG+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, 2014.

A Proof of Knowledge Soundness of Π_{sub}

We show the theorem under the discrete logarithm assumption in asymmetric bilinear groups in the algebraic group model of [FKL18]. Without lose of generality, we consider the Sub QA-NIZK scheme for linear subspaces Π_{sub} for $\hat{\mathcal{D}}_k = \bar{\mathcal{D}}_k$ (Subversion Π'_{as}), in the MDDH setting where $k = 1$. The proof follows the argumentation in [CFQ19].

Proof. Assume an algebraic adversary $\mathcal{A}([\boldsymbol{M}]_1, \mathsf{crs}, \mathsf{aux})$ against the knowledge soundness of Π_{sub} where aux is an associated auxiliary input and $\mathsf{crs} = \{[\boldsymbol{A}, \boldsymbol{P}]_1, [\boldsymbol{A}, \boldsymbol{C}]_2\}$. Let $[\boldsymbol{\zeta}]_1$ be a vector that contains $\boldsymbol{M}$ and the portion of

aux that has elements from the group $\mathbb{G}_1$. Assume $[\boldsymbol{\zeta}]_1$ includes $[1]_1$. $\mathcal{A}$ returns a pair $([\boldsymbol{y}]_1, [\pi]_1)$ along with coefficients that explain these elements as linear combinations of its input in the group $\mathbb{G}_1$. Let these coefficients be:

$$[\boldsymbol{y}]_1 = \boldsymbol{Y}_0[\boldsymbol{P}]_1 + \boldsymbol{Y}_1[\boldsymbol{\zeta}]_1 = \boldsymbol{Y}_0[\boldsymbol{M}^\top \boldsymbol{K}]_1 + \boldsymbol{Y}_1[\boldsymbol{\zeta}]_1$$
$$[\boldsymbol{\pi}]_1 = \boldsymbol{Z}_0[\boldsymbol{P}]_1 + \boldsymbol{Z}_1[\boldsymbol{\zeta}]_1 = \boldsymbol{Z}_0[\boldsymbol{M}^\top \boldsymbol{K}]_1 + \boldsymbol{Z}_1[\boldsymbol{\zeta}]_1$$

Let the extractor $\mathsf{Ext}_{\mathcal{A}}([\boldsymbol{M}]_1, \mathsf{crs}, \mathsf{aux})$ be the algorithm that runs $\mathcal{A}$ and returns $\boldsymbol{w} = \boldsymbol{Z}_0$. Then, we have to show that the probability that the output of $(\mathcal{A}, \mathsf{Ext}_{\mathcal{A}})$ satisfies verification while $\boldsymbol{y} \neq \boldsymbol{M}\boldsymbol{w}$ is negligible. In other words, assume that the output of $\mathcal{A}$ is such that $[\boldsymbol{y}]_1 \neq [\boldsymbol{M}]_1 \boldsymbol{Z}_0$ and,

$$[\boldsymbol{y}]_1^\top [\boldsymbol{A}\boldsymbol{K}]_2 = [\boldsymbol{\pi}]_1 [\boldsymbol{A}]_T.$$

If it happens with non-negligible probability, we can construct an algorithm $\mathcal{B}$ that on input $([\boldsymbol{K}]_1, [\boldsymbol{K}]_2)$ outputs nonzero elements $\boldsymbol{\alpha} \in \mathbb{Z}_p^{\ell \times \ell}$, $\boldsymbol{\beta} \in \mathbb{Z}_p^{\ell}$, and $\gamma \in \mathbb{Z}_p$ such that

$$\boldsymbol{K}^\top \boldsymbol{\alpha} \boldsymbol{K} + \boldsymbol{K}^\top \boldsymbol{\beta} + \gamma = 0$$

then we can construct an algorithm $\mathcal{C}$ against the discrete logarithm assumption in asymmetric bilinear groups such that given elements $([t]_1, [t]_2)$ it returns the exponent $t \in \mathbb{Z}_p$. More precisely the algorithm $\mathcal{B}([\boldsymbol{K}]_1, [\boldsymbol{K}]_2)$ proceeds as follows:

- Choose $([\boldsymbol{M}]_1, \mathsf{aux})$ from $\mathcal{D}_{\mathsf{p}}$ along with its $\mathbb{G}_1$ (i.e., a vector $\boldsymbol{z}$ of entries in $\mathbb{Z}_p$).
- Sample $a \leftarrow_{\$} \mathbb{Z}_p$ and run $\mathcal{A}([\boldsymbol{\zeta}, \boldsymbol{P}, a]_1, [a, a\boldsymbol{K}]_2)$. We note that $\mathcal{A}$'s input can be efficiently simulated.
- Once received the output of $\mathcal{A}$, it sets $\boldsymbol{\alpha} := \boldsymbol{Y}_0 \boldsymbol{M}^\top$, $\boldsymbol{\beta} := \boldsymbol{Y}_1 \boldsymbol{\zeta} - \boldsymbol{M}\boldsymbol{Z}_0$ and $\gamma := -\boldsymbol{Z}_1 \boldsymbol{\zeta}$

Notice that

$$\begin{aligned}
\boldsymbol{K}^\top \boldsymbol{\alpha} \boldsymbol{K} + \boldsymbol{k}^\top \boldsymbol{\beta} + \gamma &= \boldsymbol{k}^\top \boldsymbol{Y}_0 \boldsymbol{M}^\top \boldsymbol{k} + \boldsymbol{k}^\top \boldsymbol{Y}_1 \boldsymbol{\zeta} - \boldsymbol{K}^\top \boldsymbol{M} \boldsymbol{Z}_0 - \boldsymbol{Z}_1^\top \boldsymbol{\zeta} \\
&= \boldsymbol{K}^\top \boldsymbol{Y}_0 \boldsymbol{M}^\top \boldsymbol{K} + \boldsymbol{K}^\top \boldsymbol{Y}_1 \boldsymbol{\zeta} - \boldsymbol{\pi} \\
&= \boldsymbol{K}^\top \boldsymbol{y} - \boldsymbol{\pi} = 0.
\end{aligned}$$

Note that, one among $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, and γ must be nonzero. Indeed, if they are all zero then $\boldsymbol{Y}_1 \boldsymbol{\zeta} - \boldsymbol{M}\boldsymbol{Z}_0 = 0$, that is $\boldsymbol{y} = \boldsymbol{M}\boldsymbol{Z}_0$, which contradicts our assumption on $\mathcal{A}$'s output.

Finally we show how the above problem can be reduced to discrete logarithm problem in asymmetric groups, i.e., the adversary $\mathcal{C}$ on input $([t]_1, [t]_2)$ returns t. Indeed $\mathcal{C}$ samples $\boldsymbol{r}, \boldsymbol{s} \in \mathbb{Z}_p^{\ell}$ and implicitly sets $\boldsymbol{K} = t\boldsymbol{r} + \boldsymbol{s}$. We see that $([\boldsymbol{K}]_1, [\boldsymbol{K}]_2)$ can be efficiently simulated with a distribution identical to the one expected by $\mathcal{B}$. Next, given a solution $(\boldsymbol{\alpha}, \boldsymbol{\beta}, \gamma)$ such that $\boldsymbol{K}^\top \boldsymbol{\alpha} + \boldsymbol{K}^\top \boldsymbol{\beta} + \gamma = 0$, one can find $e_1, e_2, e_3 \in \mathbb{Z}_p$ such that:

$$0 = (t\boldsymbol{r} + \boldsymbol{s})^\top \boldsymbol{\alpha} (t\boldsymbol{r} + \boldsymbol{s}) + (t\boldsymbol{r} + \boldsymbol{s})^\top \boldsymbol{\beta} + \gamma$$

$$= t^2(\boldsymbol{r}^\top \boldsymbol{\alpha} \boldsymbol{r}) + t(\boldsymbol{r}^\top \boldsymbol{\aleph} \boldsymbol{s} + \boldsymbol{s}^\top \boldsymbol{\alpha} \boldsymbol{r} + \boldsymbol{r}^\top \boldsymbol{\beta}) + (\boldsymbol{s}^\top \boldsymbol{\alpha} \boldsymbol{s} + \boldsymbol{s}^\top \boldsymbol{\beta} + \gamma)$$
$$= e_1 t^2 + e_2 t + e_3$$

In particular, with overwhelming probability (over the choice of $\boldsymbol{s}$ that is information theoretically hidden from $\mathcal{B}$'s view) $e_3 \neq 0$. From this solution, $\mathcal{C}$ can solve the system and extract t.