Shorter QA-NIZK and SPS with Tighter Security

Masayuki Abe[1], Charanjit S. Jutla[2], Miyako Ohkubo[3], Jiaxin Pan[⋆4], Arnab Roy[5], and Yuyu Wang[⋆⋆6]

[1] NTT Corporation, Tokyo, Japan
abe.masayuki@lab.ntt.co.jp
[2] IBM T. J. Watson Research Center, Yorktown Heights, USA
csjutla@us.ibm.com
[3] Security Fundamentals Laboratories, CSR, NICT, Tokyo, Japan
m.ohkubo@nict.go.jp
[4] Department of Mathematical Sciences, NTNU – Norwegian University of Science and Technology, Trondheim, Norway
jiaxin.pan@ntnu.no
[5] Fujitsu Laboratories of America, Sunnyvale, USA
aroy@us.fujitsu.com
[6] University of Electronic Science and Technology of China, Chengdu, China
wangyuyu@uestc.edu.cn

Abstract. Quasi-adaptive non-interactive zero-knowledge proof (QA-NIZK) systems and structure-preserving signature (SPS) schemes are two powerful tools for constructing practical pairing-based cryptographic schemes. Their efficiency directly affects the efficiency of the derived advanced protocols.

We construct more efficient QA-NIZK and SPS schemes with tight security reductions. Our QA-NIZK scheme is the *first* one that achieves both tight simulation soundness and constant proof size (in terms of number of group elements) at the same time, while the recent scheme from Abe et al. (ASIACRYPT 2018) achieved tight security with proof size linearly depending on the size of the language and the witness. Assuming the hardness of the Symmetric eXternal Diffie-Hellman (SXDH) problem, our scheme contains only 14 elements in the proof and remains independent of the size of the language and the witness. Moreover, our scheme has tighter simulation soundness than the previous schemes.

Technically, we refine and extend a partitioning technique from a recent SPS scheme (Gay et al., EUROCRYPT 2018). Furthermore, we improve the efficiency of the tightly secure SPS schemes by using a relaxation of NIZK proof system for OR languages, called designated-prover NIZK system. Under the SXDH assumption, our SPS scheme contains 11 group

⋆ Research was conducted at KIT, Germany under the DFG grant HO 4534/4-1.

⋆⋆ Research was conducted at Tokyo Institute of Technology. A part of this work was supported by the Sichuan Science and Technology Program under Grant 2017GZDZX0002 and 2018GZDZX0006, Input Output Cryptocurrency Collaborative Research Chair funded by IOHK, JST OPERA JPMJOP1612, JST CREST JPMJCR14D6, JSPS KAKENHI JP16H01705, JP17H01695.

elements in the signature, which is shortest among the tight schemes and is the same as an early non-tight scheme (Abe et al., ASIACRYPT 2012). Compared to the shortest known non-tight scheme (Jutla and Roy, PKC 2017), our scheme achieves tight security at the cost of 5 additional elements.

All the schemes in this paper are proven secure based on the Matrix Diffie-Hellman assumptions (Escala et al., CRYPTO 2013). These are a class of assumptions which include the well-known SXDH and DLIN assumptions and provide clean algebraic insights to our constructions. To the best of our knowledge, our schemes achieve the best efficiency among schemes with the same functionality and security properties. This naturally leads to improvement of the efficiency of cryptosystems based on simulation-sound QA-NIZK and SPS.

Keywords. Quasi-adaptive NIZK, simulation soundness, structure-preserving signature, tight reduction.

1 Introduction

Bilinear pairing groups have enabled the construction of a plethora of rich cryptographic primitives in the last two decades, starting from the seminal works on three-party key exchange [30] and identity-based encryption (IBE) [11]. In particular, the Groth-Sahai non-interactive zero-knowledge (NIZK) proof system [24] for proving algebraic statements over pairing groups has proven to be a powerful tool to construct more efficient advanced cryptographic protocols, such as group signatures [21], anonymous credentials [7], and UC-secure commitment [17] schemes.

Quasi-Adaptive NIZK for Linear Subspaces. There are many applications which require NIZK systems for proving membership in linear subspaces of group vectors. A couple of examples are CCA2-secure public-key encryption via the Naor-Yung paradigm [42], and publicly verifiable CCA2-secure IBE [29].

For proving linear subspace membership, the Groth-Sahai system has a proof size linear in the dimension of the language and the subspace, in terms of number of group elements. To achieve better efficiency, Jutla and Roy proposed a weaker notion [32] called quasi-adaptive NIZK arguments (QA-NIZK), where the common reference string (CRS) may depend on the linear subspace and the soundness is computationally adaptive. For computationally adaptive soundness, the adversary is allowed to submit a proof for its adaptively chosen invalid statement. Based on their work, further improvements [38,33,1] gave QA-NIZK systems with constant proof size. This directly led to KDM-CCA2-secure PKE and publicly verifiable CCA2-secure IBE with constant-size ciphertexts.

Structure-Preserving Signature. Structure-Preserving (SP) cryptography [3] has evolved as an important paradigm in designing modular protocols. In order to enable interoperability, it is required for SP primitives to support verification only by pairing product equations, which enable zero-knowledge proofs using Groth-Sahai NIZKs.

Structure-preserving signature (SPS) schemes are the most important building blocks in constructing anonymous credential [7], voting systems and mixnets [22], and privacy-preserving point collection [25]. In an SPS, all the public keys, messages, and signatures are group elements and verification is done by checking pairing-product equations. Constructing SPS is a very challenging task, as traditional group-based signatures use hash functions, which are not structure-preserving.

Tight Security. The security of a cryptographic scheme is proven by constructing a reduction $\mathcal{R}$ which uses a successful adversary $\mathcal{A}$ against the security of the scheme to solve some hard problem. Concretely, this argument establishes the relation between the success probability of $\mathcal{A}$ (denoted by $\varepsilon_{\mathcal{A}}$) and that of $\mathcal{R}$ (denoted by $\varepsilon_{\mathcal{R}}$) as $\varepsilon_{\mathcal{A}} \leq \ell \cdot \varepsilon_{\mathcal{R}} + \mathsf{negl}(\lambda)$, where $\mathsf{negl}(\lambda)$ is negligible in the security parameter λ. The reduction $\mathcal{R}$ is called *tight* if ℓ is a small constant and the running time of $\mathcal{R}$ is approximately the same as that of $\mathcal{A}$. Most of the recent works consider a variant notion of tight security, called *almost* tight security, where the only difference is that ℓ may linearly (or, even better, logarithmically) depend on the security parameter λ. It is worth mentioning that the security loss in all our schemes is $O(\log Q)$, where Q is the number of $\mathcal{A}$'s queries. We note that $Q \ll 2^{\lambda}$ and thus our security loss is much less than $O(\lambda)$. In this paper, we do not distinguish tight security and almost tight security, but we do provide the concrete security bounds.

Tightly secure schemes are more desirable than their non-tight counterparts, since tightly secure schemes do not need to compensate much for their security loss and allow universal key-length recommendations independent of the envisioned size of an application. In recent years, there have been significant efforts in developing schemes with tight security, such as PKEs [28,26,18,27,19], IBEs [13,9,29], and signatures [28,8,4,20].

As discussed above, QA-NIZK and SPS are important building blocks for advanced protocols which are embedded in larger scale settings. Designing efficient QA-NIZK and SPS with tight security is very important, since non-tight schemes can result in much larger security loss in the derived protocols.

QA-NIZK: Tight Security or Compact Proofs? Several of the aforementioned applications of QA-NIZK require a stronger security notion, called simulation soundness, where an adversary can adaptively query simulated proofs for vectors either inside or outside the linear subspace and in the end the adversary needs to forge a proof on a vector outside the subspace. We assume that the simulation oracle can be queried by the adversary up to Q times. If $Q > 1$, we call the QA-NIZK scheme unbounded simulation-sound and if $Q = 1$, we call it one-time simulation-sound. Many applications, such as multi-challenge (KDM-)CCA2-secure PKE and CCA2-secure IBE, require unbounded simulation soundness.

If we consider the tightness, CRS and proof sizes[7] of previous works, we have three different flavors of unbounded simulation-sound QA-NIZK schemes:

[7] We only count numbers of group elements.

(1) schemes with non-tight security, but compact CRS-es (which only depend on the dimension of the subspace) and constant-size proofs [37]; (2) schemes with tight security and constant-size proofs, but linear-size CRS-es (which are linearly in λ) [18,29]; and (3) schemes with tight security and compact CRS-es, but linear-size proofs (in the dimension of the language and the subspace) [5,6].

A few remarks are made for the tightly secure QA-NIZK scheme of Abe et al. [5,6]. Its proceedings version has a bug and the authors fix it in the ePrint version [6], but the proof size of the new scheme linearly depends on the dimension of the language and the subspace. To be more technical, the work of Abe et al. achieves tight simulation soundness via the (structure-preserving) adaptive partitioning of [4,31]. Due to its use of OR proofs (cf. Figure 1 in their full version [6]), the QA-NIZK proof size ends up being linear in the size of the language and the subspace (in particular, $|\pi| = O(n_1+n_2)$). Thus, it remained open and interesting to construct a tightly simulation-sound QA-NIZK with compact CRS-es and constant-size proofs.

SPS: Tightness with Shorter Signatures. In the past few years, substantial progress was made to improve the efficiency of SPS. So far the schemes with shortest signatures have 6 signature elements with non-tight reduction [34] by improving [36], or 12 elements with security loss $36\log(Q)$ [6], or 14 elements with security loss $6\log(Q)$ [20], where Q is the number of signing queries. Our goal is to construct tightly secure SPS with shorter signatures and less security loss.

1.1 Our Contributions

To make progress on the aforementioned two questions, we construct a QA-NIZK scheme with 14 proof elements and an SPS scheme with 11 signature elements, based on the Symmetric eXternal Diffie-Hellman (SXDH) assumption. The security of both schemes is proven with tight reduction to the Matrix Diffie-Hellman (MDDH) assumption [16], which is an algebraic generalization of Diffie-Hellman assumptions (including SXDH). The security proof gives us algebraic insights to our constructions and furthermore our constructions can be implemented by (possibly weaker) linear assumptions beyond SXDH.

Our QA-NIZK scheme is the *first* one that achieves tight simulation soundness, compact CRS-es and constant-size proofs at the same time. Even among the tightly simulation-sound schemes, our scheme has less security loss. Since it achieves better efficiency, using our scheme immediately improves the efficiency of the applications of QA-NIZK with unbounded simulation soundness, including publicly verifiable CCA2-secure PKE with multiple challenge ciphertexts.

In contrast to the Abe et al. framework [5], we use a simpler and elegant framework to achieve better efficiency. Technically, we make novel use of the recent core lemma from [20] to construct a designated-verifier QA-NIZK (DV-QA-NIZK) and then compile it to (publicly verifiable) QA-NIZK by using the bilinearity of pairings. As a by-product, we achieve a tightly secure DV-QA-NIZK, where the verifier holds a secret verification key.

Let

$$\mathcal{L}_{[\mathbf{M}]_1} := \{[\mathbf{y}]_1 \in \mathbb{G}_1^{n_1} : \exists \mathbf{w} \in \mathbb{Z}_p^{n_2} \text{ such that } \mathbf{y} = \mathbf{M}\mathbf{w}\}^{8} \tag{1}$$

be a linear subspace, where $\mathbf{M} \in \mathbb{Z}_p^{n_1 \times n_2}$ and $n_1 > n_2$. We compare the efficiency and security loss of QA-NIZK schemes in Table 1. Here we instantiate our schemes (in both Tables 1 and 2) based on the SXDH assumption for a fair comparison.

Scheme	**Type**	$\lvert \mathsf{crs} \rvert$	$\lvert\pi\rvert$	**Sec. los.**	**Ass.**
LPJY14 [38]	QA-NIZK	$2n_1 + 3(n_2 + \lambda) + 10$	20	$O(Q)$	DLIN
KW15 [37]	QA-NIZK	$(2n_2 + 6, n_1 + 6)$	$(4, 0)$	$O(Q)$	SXDH
LPJY15 [39]	QA-NIZK	$2n_1 + 3n_2 + 24\lambda + 55$	42	$3\lambda + 7$	DLIN
GHKW16 [18]	DV-QA-NIZK	$n_2 + \lambda$	4	$8\lambda + 2$	DDH
GHKW16 [18]	QA-NIZK	$(n_2 + 6\lambda + 1, n_1 + 2)$	$(3, 0)$	$4\lambda + 1$	SXDH
AJOR18 [5,6]	QA-NIZK	$(3n_2 + 15, n_1 + 12)$	$(n_1 + 16, 2(n_2 + 5))$	$36 \log(Q)$	SXDH
Ours (Section 3.1)	DV-QA-NIZK	$(2n_2 + 3, 4)$	$(7, 6)$	$6 \log(Q)$	SXDH
Ours (Section 3.2)	QA-NIZK	$(4n_2 + 4, 8 + 2n_1)$	$(8, 6)$	$6 \log(Q)$	SXDH

Table 1. Comparison of unbounded simulation-sound QA-NIZK schemes for proving membership in $\mathcal{L}_{[\mathbf{M}]_1}$. $\lvert \mathsf{crs} \rvert$ and $\lvert\pi\rvert$ denote the size of CRS-es and proofs in terms of numbers of group elements. For asymmetric pairings, notation (x, y) means x elements in $\mathbb{G}_1$ and y elements in $\mathbb{G}_2$. Q denotes the number of simulated proofs and λ is the security parameter.

Our second contribution is a more efficient tightly secure SPS. It contains 11 signature elements and $n_1 + 15$ public key elements, while the scheme from [5] contains 12 and $3n_1 + 23$ elements respectively, where n_1 denotes the number of group elements in a message vector. We give a comparison between our scheme and previous ones in Table 2. Compared with GHKP18, our construction has shorter signatures and less pairing-product equations (PPEs) with the same level of security loss. Compared with AJOR18, our construction has shorter signature and tighter security, but slightly more PPEs. We leave constructing an SPS with the same signature size and security loss but less PPEs as an interesting open problem. As an important building block of our SPS, we propose the notion of designated-prover OR proof systems for a unilateral language, where a prover holds a secret proving key and the language is defined in one single group. We believe that it is of independent interest.

1.2 Our QA-NIZK: Technical Overview

THE KILTZ-WEE FRAMEWORK. In contrast to the work of Abe et al. [5], our construction is motivated by the simple Kiltz-Wee framework [37], where they implicitly constructed a simulation-sound DV-QA-NIZK and then compiled it

[8] We follow the implicit notation of a group element. $[\cdot]_s$ ($s \in \{1, 2, T\}$) denotes the entry-wise exponentiation in $\mathbb{G}_s$.

Scheme	$\lvert\mathsf{m}\rvert$	$\lvert\sigma\rvert$	$\lvert\mathsf{vk}\rvert$	**Sec. loss**	**Assumption**	**# PPEs**			
						Total	NL	L1	L2
HJ12 [28]	1	$10\ell + 6$	13	$O(1)$	DLIN	$6\ell + 3$			
ACDKNO16 [2]	$(n_1, 0)$	$(7, 4)$	$(5, n_1 + 12)$	$O(Q)$	SXDH, XDLIN	5	1	2	2
LPY15 [40]	$(n_1, 0)$	$(10, 1)$	$(16, 2n_1 + 5)$	$O(Q)$	SXDH, XDLINX	5	3	2	
KPW15 [36]	$(n_1, 0)$	$(6, 1)$	$(0, n_1 + 6)$	$O(Q^2)$	SXDH	3	2	1	
JR17 [34]	$(n_1, 0)$	$(5, 1)$	$(0, n_1 + 6)$	$O(Q \log Q)$	SXDH	2	1	1	
AHNOP17 [4]	$(n_1, 0)$	$(13, 12)$	$(18, n_1 + 11)$	$O(\lambda)$	SXDH	15	4	3	8
JOR18 [31]	$(n_1, 0)$	$(11, 6)$	$(7, n_1 + 16)$	$O(\lambda)$	SXDH	8	4	2	2
GHKP18 [20]	$(n_1, 0)$	$(8, 6)$	$(2, n_1 + 9)$	$6 \log(Q)$	SXDH	9	8	1	
AJOR18 [5,6]	$(n_1, 0)$	$(6, 6)$	$(n_1 + 11, 2n_1 + 12)$	$36 \log(Q)$	SXDH	6	4	1	1
Ours(unilateral)	$(n_1, 0)$	$(7, 4)$	$(2, n_1 + 11)$	$6 \log(Q)$	SXDH	7	6	1	

Table 2. Comparison of structure-preserving signatures for message space $\mathbb{G}^{n_1}$ (in their most efficient variants). "$\lvert\mathsf{m}\rvert$", "$\lvert\sigma\rvert$", and "$\lvert\mathsf{vk}\rvert$" denote the size of messages, signatures, and public keys in terms of numbers of group elements. Q denotes the number of signing queries. "# PPEs" denotes the number of pairing-product equations. "NL" denotes the number of non-linear equations that includes signatures in both groups. "L1" denotes the number of linear equations in $\mathbb{G}_1$ group. "L2" denotes the number of linear equations in $\mathbb{G}_2$ group.

to a simulation-sound QA-NIZK with pairings. However, their simulation-sound DV-QA-NIZK is not tight. In the following, we focus on constructing a tightly simulation-sound DV-QA-NIZK. By a similar "DV-QA-NIZK $\rightarrow$ QA-NIZK transformation as in [37], we derive our QA-NIZK with shorter proofs and tighter simulation soundness in the end.

The DV-QA-NIZK in [37] is essentially a simple hash proof system [14] for the linear language $\mathcal{L}_{[\mathbf{M}]_1}$: to prove that $[\mathbf{y}]_1 = [\mathbf{Mx}]_1$ for some $\mathbf{x} \in \mathbb{Z}_p$, the prover outputs a proof as $\pi := [\mathbf{x}^\top \mathbf{p}]_1$, where the projection $[\mathbf{p}]_1 := [\mathbf{M}^\top \mathbf{k}]_1$ is published in the CRS. With the vector $\mathbf{k}$ as the secret verification key, a designated verifier can check whether $\pi = [\mathbf{y}^\top \mathbf{k}]_1$. By using $\mathbf{k}$ as a simulation trapdoor, a zero-knowledge simulator can return the simulated proof as $\pi := [\mathbf{y}^\top \mathbf{k}]_1$, due to the following equation:

$$\mathbf{x}^\top \mathbf{p} = \mathbf{x}^\top (\mathbf{M}^\top \mathbf{k}) = \mathbf{y}^\top \mathbf{k}.$$

Soundness is guaranteed by the fact that the value $\mathbf{y}^{*\top} \mathbf{k}$ is uniformly random, given $\mathbf{M}^\top \mathbf{k}$, if $\mathbf{y}^*$ is outside the span of $\mathbf{M}$.

Affine MACs and Unbounded Simulation Soundness. To achieve unbounded simulation soundness, we need to hide the information of $\mathbf{k}$ in all the Q_s-many simulation queries, in particular for the information outside the span of $\mathbf{M}^\top$. The Kiltz-Wee solution is to blind the term $\mathbf{y}^\top \mathbf{k}$ with a 2-universal hash proof system. Via a non-tight reduction the hash proof system can be proved to be a pseudorandom affine message authentication code (MAC) scheme proposed by [9]. Technically, unbounded simulation soundness requires the underlying affine MAC to be pseudorandom against multiple challenge queries. This notion has been formally considered in [29] later and it is stronger than the original security in [9]. Because of that, the affine MAC based on the Naor-Reingold PRF in [9] cannot be directly used in constructing tightly simulation-sound QA-NIZK.

Gay et al. [18] constructed a tightly secure unbounded simulation-sound QA-NIZK [9]. Essentially, their tight PCA-secure PKE against multiple challenge ciphertexts is a pseudorandom affine MAC against multiple challenge queries. Then they use this MAC to blind the term $\mathbf{y}^\top \mathbf{k}$. However, this tight solution has a large CRS, namely, the number of group elements in the CRS is linear in the security parameter. That is because the number of $\mathbb{Z}_p$ elements in the underlying affine MAC secret keys is also linear in the security parameter. These $\mathbb{Z}_p$ elements are later converted as group elements in the CRS of QA-NIZK. To the best of our knowledge, current pairing-based affine MACs enjoy either tight security and linear size secret keys or constant size secret keys but non-tight security. Therefore, it may be more promising to develop a new method, other than affine MACs, to hide $\mathbf{y}^\top \mathbf{k}$ with compact CRS and tight security.

Our Solution. We solve the above dilemma by a novel use of the core lemma from [20]. To give more details, we fix some matrices $\mathbf{A}_0, \mathbf{A}_1 \in \mathbb{Z}_p^{2k \times k}$, choose a random vector $\mathbf{k}'$ and consider $\mu := ([\mathbf{t}]_1, [u']_1, \pi')$ that has the distribution:

$$\begin{aligned} &\mathbf{t} \xleftarrow{\$} \mathsf{Span}(\mathbf{A}_0) \cup \mathsf{Span}(\mathbf{A}_1) \\ &u' = \mathbf{t}^\top \mathbf{k}' \in \mathbb{Z}_p \\ &\pi' : \text{ proves that } \mathbf{t} \in \mathsf{Span}(\mathbf{A}_0) \cup \mathsf{Span}(\mathbf{A}_1) \end{aligned} \quad . \tag{2}$$

In a nutshell, the NIZK proof π' guarantees that $\mathbf{t}$ is from the disjunction space and, by introducing randomness in the "right" space, the core lemma shows that $[u']_1$ is pseudorandom with tight reductions. The core lemma itself is not a MAC scheme, since it does not have message inputs, although it has been used to construct a tightly secure (non-affine) MAC in [20].

A "naive" attempt: Using the Core Lemma. To have unbounded simulation soundness, our first attempt is to use the pseudorandom value $[u']_1$ to directly blind the term $\mathbf{y}^\top \mathbf{k}$ from the DV-QA-NIZK with only adaptive soundness in a straightforward way. Then the resulting DV-QA-NIZK outputs the proof $([\mathbf{t}]_1, [u]_1, \pi')$, which has the following distribution:

$$\begin{aligned} &\mathbf{t} \xleftarrow{\$} \mathsf{Span}(\mathbf{A}_0) \cup \mathsf{Span}(\mathbf{A}_1) \\ &u = \mathbf{y}^\top \mathbf{k} + \boxed{\mathbf{t}^\top \mathbf{k}'} \in \mathbb{Z}_p \\ &\pi' : \text{ proves that } \mathbf{t} \in \mathsf{Span}(\mathbf{A}_0) \cup \mathsf{Span}(\mathbf{A}_1) \end{aligned} \quad . \tag{3}$$

In order to publicly generate a proof for a valid statement $[\mathbf{y}]_1 = [\mathbf{M}\mathbf{x}]_1$ with witness $\mathbf{x} \in \mathbb{Z}_p^{n_2}$, we publish $[\mathbf{M}^\top \mathbf{k}]_1, [\mathbf{A}_0^\top \mathbf{k}']_1$ and CRS for generating π' in the CRS of our DV-QA-NIZK. Verification is done with designated verification key $(\mathbf{k}, \mathbf{k}')$. Zero knowledge can be proven using $(\mathbf{k}, \mathbf{k}')$.

However, when we try to prove the unbounded simulation soundness, we run into a problem. The core lemma shows the following two distributions are tightly

[9] We note that the tight affine MAC in [29] can also be used to construct a DV-QA-NIZK and a QA-NIZK with tight unbounded simulation soundness. Their efficiency is slightly better than those in [18].

indistinguishable:

$$\mathsf{REAL} := \{([\mathbf{t}_i]_1, [\mathbf{t}_i^\top \mathbf{k}']_1, \pi_i')\} \approx_c \{([\mathbf{t}_i]_1, [\mathbf{t}_i^\top \mathbf{k}_i']_1, \pi_i')\} =: \mathsf{RAND},$$

where $\mathbf{k}', \mathbf{k}_i' \xleftarrow{\$} \mathbb{Z}_p^{2k}$ and $i = 1, ..., Q$. In the proof of unbounded simulation soundness, we switch from REAL to RAND and then we can argue that all our simulated proofs are random, since $\mathbf{y}^\top \mathbf{k}$ is blinded by the random value $\mathbf{t}_i^\top \mathbf{k}_i'$. Unfortunately, here we cannot use an information-theoretical argument to show that an adversary cannot compute a forgery for an invalid statement: An adversary can reuse the $\mathbf{k}_j$ in the j-th $(1 \leq j \leq Q)$ simulation query on $[\mathbf{y}_j]_1 \in \mathsf{Span}([\mathbf{M}']_1)$ and $\mathsf{Span}([\mathbf{M}']_1) \cap \mathsf{Span}([\mathbf{M}]_1) = \{[\mathbf{0}]_1\}$ and given the additional information $\mathbf{M}'^\top \mathbf{k}$ from the j-th query an adversary can compute a valid proof for another invalid statement $\mathbf{y}^* \in \mathsf{Span}(\mathbf{M}')$.

Moreover, this straightforward scheme has an attack: An adversary can ask for a simulated proof $\pi := ([\mathbf{t}]_1, [u]_1, \pi')$ on an invalid $[\mathbf{y}]_1$. Then it computes $([2\mathbf{t}]_1, [2u]_1)$ and adapts the OR proof π' accordingly to $\hat{\pi}$. The proof $\pi^* := ([2\mathbf{t}]_1, [2u]_1, \hat{\pi})$ is a valid proof for an invalid statement $[\mathbf{y}^*]_1 := [2\mathbf{y}]_1 \notin \mathsf{Span}([\mathbf{M}]_1)$.

From Failure to Success via Pairwise Independence. The above problem happens due to the malleability in the "naive" attempt. We introduce non-malleability by using a pairwise independent function in $\mathbf{k}$. More precisely, let $\tau \in \mathbb{Z}_p$ be a tag and our DV-QA-NIZK proof is still $([\mathbf{t}]_1, [u]_1, \pi')$ with $([\mathbf{t}]_1, \pi')$ as in Equation (3) but

$$u := \mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1) + \mathbf{t}^\top \mathbf{k}'.$$

We assume that all the tags in the simulated proofs and forgery are distinct, which can be achieved by using a collision-resistant hash as $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi') \in \mathbb{Z}_p$. Given $\mathbf{k}_j$ the adversary can only see $\mathbf{y}_j^\top(\mathbf{k}_0 + \tau_j \mathbf{k}_1)$ from the j-th query and for all the other queries the random values $\mathbf{t}_i^\top \mathbf{k}_i$ $(i \neq j)$ hide the information about $\mathbf{k}_0$ and $\mathbf{k}_1$. Given $\mathbf{k}_0 + \tau_j \mathbf{k}_1$ for a τ_j, the pairwise independence guarantees that even for a computationally unbounded adversary it is hard to compute $\mathbf{k}_0 + \tau^* \mathbf{k}_1$ for any $\tau^* \neq \tau_j$. Thus, the unbounded simulation soundness is concluded. Details are presented in Section 3.1. In a nutshell, we use the pseudorandom element $[u']_1$ from the core lemma to hide $[\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1)]_1$ from a one-time simulation sound DV-QA-NIZK.

From Designated to Public Verification. What is left to do is to convert our DV-QA-NIZK scheme into a QA-NIZK. Intuitively, we first make u publicly verifiable via the (tuned) Groth-Sahai proof technique, and then modify the QA-NIZK so that we can embed the secret key of our DV-QA-NIZK into it without changing the view of the adversary. Then we can extract a forgery for the USS experiment of the DV-QA-NIZK from the forgery by the adversary. Similar ideas have been used in many previous works [37,33,36,12,9,20].

1.3 Our SPS: Technical Overview

The recent SPS schemes exploit the adaptive partitioning paradigm [27,19,4] to achieve tight security. In this paradigm, NIZK for OR languages [23,43] plays an important role, while at the same time, it also incurs high cost. Our basic idea is to replace the full-fledged OR proof system proposed by Gay et al. [20] with one in the designated-prover setting, where a prover is allowed to use a secret proving key. Intuitively, it is easier to achieve an efficient scheme in such a setting since it suffers less restrictions. In fact, the previous SPS scheme in [5] has already exploited the designated-prover setting to reduce the proof size. However, it only works for bilateral OR language (i.e., one out of two words lies in the linear span of its corresponding space), while an OR-proof for unilateral language (i.e., a single word lies in the linear span of either one of two spaces) is required in the construction of [20]. Thus, some new technique is necessary for solving this problem.

For ease of exposition, we focus on the SXDH setting now, where the following OR-language is in consideration:

$$\mathcal{L}_1 := \{[\mathbf{y}]_1 \in \mathbb{G}_1^2 \mid \exists r \in \mathbb{Z}_p \colon [\mathbf{y}]_1 = [\mathbf{A}_0]_1 \cdot r \vee [\mathbf{y}]_1 = [\mathbf{A}_1]_1 \cdot r\}.$$

Let $\mathbf{A}_1 = (a,\ b)^\top$, we observe that it is equivalent to the following language.

$$\mathcal{L}_2 := \{[y_0, y_1]_1^\top \in \mathbb{G}_1^2 \mid \exists x, x' \in \mathbb{Z}_p \colon [y_1]_1 - [y_0]_1 \cdot \frac{b}{a} = [x]_1 \wedge [\mathbf{y}]_1 \cdot x = [\mathbf{A}_0]_1 \cdot x'\}.$$

Specifically, when $x = 0$, we have $[y_1]_1 - [y_0]_1 \cdot \frac{b}{a} = [0]_1$, i.e., $[y_0, y_1]_1^\top$ is in the span of $\mathbf{A}_1$. Otherwise, we have $[\mathbf{y}]_1 = [\mathbf{A}_0]_1 \cdot \frac{x'}{x}$, i.e., $[y_0, y_1]_1^\top$ is in the span of $\mathbf{A}_0$. Note that this language is an "AND-language" now. More importantly, a witness consists only of 2 scalars and a statement consists only of 3 equations. Hence, when applying the Groth-Sahai proof [24,15], the proof size will be only 7 (4 elements for committing the witness and 3 elements for equations), which is shorter than the well-known OR proof in [43] (10 elements). However, the statement contains $\frac{b}{a}$ now, which may leak information on a witness. To avoid this, we make $\frac{b}{a}$ part of the witness and store its commitment (which consists of 2 group elements) in the common reference string. By doing this, we can ensure that the information on $\frac{b}{a}$ will not be leaked and $\frac{b}{a}$ is always "fixed", due to the hiding and biding properties of commitments respectively. Also, notice that this does not increase the size of proofs at all. This scheme satisfies perfect soundness, and zero-knowledge can be tightly reduced to the SXDH assumption. Since the prover has to use $\frac{b}{a}$ to generate a witness for $\mathcal{L}_2$ given a witness for $\mathcal{L}_1$, this scheme only works in the designated-prover setting. However, notice that when simulating the proof, $\mathbf{A}_0$ and $\mathbf{A}_1$ are not necessary, which is a crucial property when applying to the partitioning paradigm.

We further generalize this scheme to one under the $\mathcal{D}_k$-MDDH assumptions for a fixed k. The size of proof will become $O(k^3)$, and the zero-knowledge property can be reduced to the $\mathcal{D}_k$-MDDH assumption with almost no security loss.

Replacing the OR-proof system of [20] with our designated-prover ones immediately derives the most efficient SPS by now. We refer the reader to Table 2 for the comparison between our scheme and previous ones.

Additionally, we give another designated-prover OR proof scheme where the proof size is $O(k^2)$, which is smaller than the above scheme when $k > 1$. As a trade-off, it suffers a security loss of k. When $k = 1$, its efficiency is the same as that of our original designated-prover OR proof scheme described above. In symmetric groups, we adapt the designated-prover OR proof to provide the most efficient full NIZK (i.e., one with public prover and verifier algorithms) for OR languages based on the $\mathcal{D}_k$-MDDH assumptions by now.

2 Preliminaries

Notations. We denote an empty string as ϵ. We use $x \xleftarrow{\$} \mathcal{S}$ to denote the process of sampling an element x from set $\mathcal{S}$ uniformly at random. For positive integers $k > 1, \eta \in \mathbb{Z}^+$ and a matrix $\mathbf{A} \in \mathbb{Z}_p^{(k+\eta)\times k}$, we denote the upper square matrix of $\mathbf{A}$ by $\overline{\mathbf{A}} \in \mathbb{Z}_p^{k\times k}$ and the lower η rows of $\mathbf{A}$ by $\underline{\mathbf{A}} \in \mathbb{Z}_p^{\eta\times k}$. Similarly, for a column vector $\mathbf{v} \in \mathbb{Z}_p^{k+\eta}$, we denote the upper k elements by $\overline{\mathbf{v}} \in \mathbb{Z}_p^k$ and the lower η elements of $\mathbf{v}$ by $\underline{\mathbf{v}} \in \mathbb{Z}_p^\eta$. For a bit string $m \in \{0,1\}^n$, m_i denotes the ith bit of m $(i \leq n)$ and $m_{|i}$ denotes the first i bits of m.

All our algorithms are probabilistic polynomial time unless we stated otherwise. If $\mathcal{A}$ is a probabilistic polynomial time algorithm, then we write $a \xleftarrow{\$} \mathcal{A}(b)$ to denote the random variable that outputted by $\mathcal{A}$ on input b.

Games. We follow [9] to use code-based games for defining and proving security. A game G contains procedures Init and Finalize, and some additional procedures $\mathrm{P}_1, \ldots, \mathrm{P}_n$, which are defined in pseudo-code. All variables in a game are initialized as 0, and all sets are empty (denote by $\emptyset$). An adversary $\mathcal{A}$ is executed in game G (denote by $\mathsf{G}^{\mathcal{A}}$) if it first calls Init, obtaining its output. Next, it may make arbitrary queries to P_i (according to their specification) and obtain their output, where the total number of queries is denoted by Q. Finally, it makes one single call to Finalize$(\cdot)$ and stops. We use $\mathsf{G}^{\mathcal{A}} \Rightarrow d$ to denote that G outputs d after interacting with $\mathcal{A}$, and d is the output of Finalize.

2.1 Collision Resistant Hash Functions.

Let $\mathcal{H}$ be a family of hash functions $H : \{0,1\}^* \to \{0,1\}^\lambda$. We assume that it is efficient to sample a function from $\mathcal{H}$, which is denoted by $H \xleftarrow{\$} \mathcal{H}$.

Definition 1 (Collision resistance). *We say a family of hash functions $\mathcal{H}$ is collision-resistant (*CR*) if for all adversaries $\mathcal{A}$*

$$\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},\mathcal{A}}(\lambda) := \Pr[x \neq x' \wedge H(x) = H(x') \mid H \xleftarrow{\$} \mathcal{H}, (x, x') \xleftarrow{\$} \mathcal{A}(1^\lambda, H)]$$

is negligible.

2.2 Pairing Groups and Matrix Diffie-Hellman Assumptions

Let GGen be a probabilistic polynomial time (PPT) algorithm that on input 1^λ returns a description $\mathcal{G} := (\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, p, P_1, P_2, e)$ of asymmetric pairing groups where $\mathbb{G}_1$, $\mathbb{G}_2$, $\mathbb{G}_T$ are cyclic groups of order p for a λ-bit prime p, P_1 and P_2 are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively, and $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is an efficient computable (non-degenerated) bilinear map. Define $P_T := e(P_1, P_2)$, which is a generator in $\mathbb{G}_T$. In this paper, we only consider Type III pairings, where $\mathbb{G}_1 \neq \mathbb{G}_2$ and there is no efficient homomorphism between them.

We use implicit representation of group elements as in [16]. For $s \in \{1, 2, T\}$ and $a \in \mathbb{Z}_p$ define $[a]_s = aP_s \in \mathbb{G}_s$ as the implicit representation of a in $\mathbb{G}_s$. Similarly, for a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_p^{n\times m}$ we define $[\mathbf{A}]_s$ as the implicit representation of $\mathbf{A}$ in $\mathbb{G}_s$. $\mathsf{Span}(\mathbf{A}) := \{\mathbf{Ar}|\mathbf{r} \in \mathbb{Z}_p^m\} \subset \mathbb{Z}_p^n$ denotes the linear span of $\mathbf{A}$, and similarly $\mathsf{Span}([\mathbf{A}]_s) := \{[\mathbf{Ar}]_s|\mathbf{r} \in \mathbb{Z}_p^m\} \subset \mathbb{G}_s^n$. Note that it is efficient to compute $[\mathbf{AB}]_s$ given $([\mathbf{A}]_s, \mathbf{B})$ or $(\mathbf{A}, [\mathbf{B}]_s)$ with matching dimensions. We define $[\mathbf{A}]_1 \circ [\mathbf{B}]_2 := e([\mathbf{A}]_1, [\mathbf{B}]_2) = [\mathbf{AB}]_T$, which can be efficiently computed given $[\mathbf{A}]_1$ and $[\mathbf{B}]_2$.

Next we recall the definition of the Matrix Decisional Diffie-Hellman (MDDH) [16] and related assumptions [41].

Definition 2 (Matrix distribution). *Let $k, \ell \in \mathbb{N}$ with $\ell > k$. We call $\mathcal{D}_{\ell,k}$ a matrix distribution if it outputs matrices in $\mathbb{Z}_p^{\ell\times k}$ of full rank k in polynomial time. By $\mathcal{D}_k$ we denote $\mathcal{D}_{k+1,k}$.*

Without loss of generality, we assume the first k rows of $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$ form an invertible matrix. For a matrix $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$, we define the set of kernel matrices of $\mathbf{A}$ as

$$\mathsf{ker}(\mathbf{A}) := \{\mathbf{a}^\perp \in \mathbb{Z}_p^{(\ell-k)\times\ell} \mid \mathbf{a}^\perp \cdot \mathbf{A} = \mathbf{0} \in \mathbb{Z}_p^{(\ell-k)\times k} \text{ and } \mathbf{a}^\perp \text{ has rank } (\ell - k)\}.$$

Given a matrix $\mathbf{A}$ over $\mathbb{Z}_p^{\ell\times k}$, it is efficient to sample an $\mathbf{a}^\perp$ from $\mathsf{ker}(\mathbf{A})$.

The $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman problem is to distinguish the two distributions $([\mathbf{A}], [\mathbf{Aw}])$ and $([\mathbf{A}], [\mathbf{u}])$ where $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}$, $\mathbf{w} \xleftarrow{\$} \mathbb{Z}_p^k$ and $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_p^\ell$.

Definition 3 ($\mathcal{D}_{\ell,k}$-matrix decisional Diffie-Hellman assumption). *Let $\mathcal{D}_{\ell,k}$ be a matrix distribution and $s \in \{1, 2, T\}$. We say that the $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-MDDH) is hard relative to GGen in group $\mathbb{G}_s$ if for all PPT adversaries $\mathcal{A}$, it holds that*

$$\mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) := |\Pr[1 \xleftarrow{\$} \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{Aw}]_s)] - \Pr[1 \xleftarrow{\$} \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s, [\mathbf{u}]_s)]|$$

is negligible in the security parameter λ, where the probability is taken over $\mathcal{G} \xleftarrow{\$} \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_{\ell,k}, \mathbf{w} \xleftarrow{\$} \mathbb{Z}_p^k$ and $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_p^\ell$.

We define the Kernel Diffie-Hellman assumption $\mathcal{D}_k$-KerMDH [41] which is a natural search variant of the $\mathcal{D}_k$-MDDH assumption.

Definition 4 ($\mathcal{D}_k$-kernel Diffie-Hellman assumption, $\mathcal{D}_k$-KerMDH). *Let $\mathcal{D}_k$ be a matrix distribution and $s \in \{1,2\}$. We say that the $\mathcal{D}_k$-kernel Matrix Diffie-Hellman ($\mathcal{D}_k$-KerMDH) is hard relative to* GGen *in group $\mathbb{G}_s$ if for all PPT adversaries $\mathcal{A}$, it holds that*

$$\mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) := \Pr[\mathbf{c}^\top \mathbf{A} = \mathbf{0} \wedge \mathbf{c} \neq \mathbf{0} | [\mathbf{c}]_{3-s} \xleftarrow{\$} \mathcal{A}(\mathcal{G}, [\mathbf{A}]_s)]$$

is negligible in security parameter λ, where the probability is taken over $\mathcal{G} \xleftarrow{\$} \mathsf{GGen}(1^\lambda)$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$.

The following lemma shows that the $\mathcal{D}_k$-KerMDH assumption is a relaxation of the $\mathcal{D}_k$-MDDH assumption since one can use a non-zero vector in the kernel of $\mathbf{A}$ to test membership in the column space of $\mathbf{A}$.

Lemma 1 ($\mathcal{D}_k$-MDDH $\Rightarrow$ $\mathcal{D}_k$-KerMDH [41]). *For any matrix distribution $\mathcal{D}_k$, if $\mathcal{D}_k$-MDDH is hard relative to* GGen *in group $\mathbb{G}_s$, then $\mathcal{D}_k$-KerMDH is hard relative to* GGen *in group $\mathbb{G}_s$.*

For $Q > 1$, $\mathbf{W} \xleftarrow{\$} \mathbb{Z}_p^{k \times Q}, \mathbf{U} \xleftarrow{\$} \mathbb{Z}_p^{\ell \times Q}$, consider the Q-fold $\mathcal{D}_{\ell,k}$-MDDH problem which is distinguishing the distributions $([\mathbf{A}], [\mathbf{AW}])$ and $([\mathbf{A}], [\mathbf{U}])$. That is, the Q-fold $\mathcal{D}_{\ell,k}$-MDDH problem contains Q independent instances of the $\mathcal{D}_{\ell,k}$-MDDH problem (with the same $\mathbf{A}$ but different $\mathbf{w}_i$). The following lemma shows that the two problems are tightly equivalent and the reduction only loses a constant factor $\ell - k$.

Lemma 2 (Random self-reducibility [16]). *For $\ell > k$ and any matrix distribution $\mathcal{D}_{\ell,k}$, $\mathcal{D}_{\ell,k}$-MDDH is random self-reducible. In particular, for any $Q \geq 1$, if $\mathcal{D}_{\ell,k}$-MDDH is hard relative to* GGen *in group $\mathbb{G}_s$, then Q-fold $\mathcal{D}_{\ell,k}$-MDDH is hard relative to* GGen *in group $\mathbb{G}_s$, where $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and*

$$\mathsf{Adv}^{Q\text{-}\mathsf{mddh}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{A}}(\lambda) \leq (\ell - k)\mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_s,\mathcal{D}_{\ell,k},\mathcal{B}}(\lambda) + \frac{1}{p-1}.$$

The *boosting lemma* in [35] shows that the $\mathcal{D}_{2k,k}$-MDDH assumption reduces to the $\mathcal{D}_k$-MDDH assumption with a security loss of a factor of k.

2.3 Non-Interactive Zero-Knowledge Proof

In this section, we follow [24,37] to recall the notion of a non-interactive zero-knowledge proof [10] and then an instantiation for an OR-language.

Let par be the public parameter and $\mathcal{L} = \{\mathcal{L}_{\mathsf{par}}\}$ be a family of languages with efficiently computable witness relation $\mathcal{R}_{\mathcal{L}}$. This definition is as follows .

Definition 5 (Non-interactive zero-knowledge proof [24]). *A non-interactive zero-knowledge proof (*NIZK*) for $\mathcal{L}$ consists of five PPT algorithms $\Pi =$ (Gen, TGen, Prove, Ver, Sim) such that:*

- Gen(par) *returns a common reference string* crs*.*
- TGen(par) *returns* crs *and a trapdoor* td*.*

- $\mathsf{Prove}(\mathsf{crs}, x, w)$ *returns a proof* π*.*
- $\mathsf{Ver}(\mathsf{crs}, x, \pi)$ *returns* 1 *(accept) or* 0 *(reject). Here,* Ver *is deterministic.*
- $\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, x)$ *returns a proof* π*.*

Perfect completeness is satisfied if for all $\mathsf{crs} \in \mathsf{Gen}\ (1^\lambda, \mathsf{par})$*, all* $x \in \mathcal{L}$*, all witnesses* w *such that* $\mathcal{R}_\mathcal{L}(x, w) = 1$*, and all* $\pi \in \mathsf{Prove}(\mathsf{crs}, x, w)$*, we have*

$$\mathsf{Ver}(\mathsf{crs}, x, \pi) = 1.$$

Zero-knowledge is satisfied if for all PPT adversaries $\mathcal{A}$ *we have that*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{zk}}_{\Pi,\mathcal{A}}(\lambda) := \Big| &\Pr[\mathcal{A}^{\mathsf{Prove}(\mathsf{crs},\cdot,\cdot)}(1^\lambda, \mathsf{crs}) = 1 \mid \mathsf{crs} \xleftarrow{\$} \mathsf{Gen}(1^\lambda, \mathsf{par})] \\ &- \Pr[\mathcal{A}^{Sim(\mathsf{crs},\cdot,\cdot)}(1^\lambda, \mathsf{crs}) = 1 \mid (\mathsf{crs}, \mathsf{td}) \xleftarrow{\$} \mathsf{TGen}(1^\lambda, \mathsf{par})]\Big|\end{aligned}$$

is negligible, where $Sim(\mathsf{crs}, x, w)$ *returns* $\pi \xleftarrow{\$} \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, x)$ *if* $\mathcal{R}_\mathcal{L}(x, w) = 1$ *and aborts otherwise.*

Perfect soundness is satisified if for all $\mathsf{crs} \in \mathsf{Gen}(\mathsf{par})$*, for all words* $x \notin \mathcal{L}$ *and all proofs* π *it holds* $\mathsf{Ver}(\mathsf{crs}, x, \pi) = 0$*.*

Notice that Gay et al. [20] adopted a stronger notion of composable zero-knowledge. However, one can easily see that the standard we defined above is enough for their constructions, as well as ours introduced later. Also, we can define *perfect zero-knowledge*, which requires $\mathsf{Adv}^{\mathsf{zk}}_{\Pi,\mathcal{A}}(\lambda) = 0$, and *computational soundness*, which requires that for all for all words $x \notin \mathcal{L}$,

$$\mathsf{Adv}^{\mathsf{snd}}_{\Pi,\mathcal{A}} = \Big|\Pr[\mathsf{Ver}(\mathsf{crs}, x, \pi) = 1 \mid \mathsf{crs} \xleftarrow{\$} \mathsf{Gen}(1^\lambda, \mathsf{par}), \pi \xleftarrow{\$} \mathcal{A}(1^\lambda, \mathsf{crs})]\Big|$$

is negligible.

NIZK FOR AN OR-LANGUAGE. In Appendix A we recall a NIZK for an OR-language, which will be used as a building block of our QANIZK proof.

2.4 Quasi-Adaptive Zero-Knowledge Argument

The notion of quasi-adaptive zero-knowledge argument (QANIZK) was proposed by Jutla and Roy [32], where the common reference string CRS depends on the specific language for which proofs are generated. In the following, we recall the definition of QANIZK [37,18]. For simplicity, we only consider arguments for linear subspaces.

Let par be the public parameters for QANIZK and $\mathcal{D}_{\mathsf{par}}$ be a probability distribution over a collection of relations $R = \{R_{[\mathbf{M}]_1}\}$ parametrized by a matrix $[\mathbf{M}]_1 \in \mathbb{G}_1^{n_1 \times n_2}$ $(n_1 > n_2)$ with associated language $\mathcal{L}_{[\mathbf{M}]_1} = \{[\mathbf{t}]_1 : \exists \mathbf{w} \in \mathbb{Z}_q^t,\ \text{s.t.}\ [\mathbf{t}]_1 = [\mathbf{M}\mathbf{w}]_1\}$. We consider witness sampleable distributions [32] where there is an efficiently sampleable distribution $\mathcal{D}'_{\mathsf{par}}$ outputs $\mathbf{M}' \in \mathbb{Z}_q^{n_1 \times n_2}$ such that $[\mathbf{M}']_1$ distributes the same as $[\mathbf{M}]_1$. We note that the matrix distribution in Definition 2 is sampleable.

We define the notions of QANIZK, designated-prover QANIZK ($\mathsf{DPQANIZK}$), designated-verifier QANIZK ($\mathsf{DVQANIZK}$), designated-prover-verifier QANIZK ($\mathsf{DPVQANIZK}$) as follow.

Definition 6 (QANIZK). *Let* $\mathsf{X} \in \{\epsilon, \mathsf{DP}, \mathsf{DV}, \mathsf{DPV}\}$*. An* $\mathsf{XQANIZK}$ *for a language distribution* $\mathcal{D}_{\mathsf{par}}$ *consists of four PPT algorithms* $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$*.*

- $\mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1)$ *returns a common reference string* crs*, a prover key* prk*, a verifier key* vrk *and a simulation trapdoor* td*:*
  - $\mathsf{X} = \epsilon$ *iff* $\mathsf{prk} = \mathsf{vrk} = \epsilon$*.*
  - $\mathsf{X} = \mathsf{DP}$ *iff* $\mathsf{vrk} = \epsilon$*.*
  - $\mathsf{X} = \mathsf{DV}$ *iff* $\mathsf{prk} = \epsilon$*.*
  - $\mathsf{X} = \mathsf{DPV}$ *iff* $\mathsf{prk} \neq \epsilon$ *and* $\mathsf{vrk} \neq \epsilon$*.*
- $\mathsf{Prove}(\mathsf{crs}, \mathsf{prk}, [\mathbf{y}]_1, \mathbf{w})$ *returns a proof* π*.*
- $\mathsf{Ver}(\mathsf{crs}, \mathsf{vrk}, [\mathbf{y}]_1, \pi)$ *returns* 1 *(accept) or* 0 *(reject). Here,* Ver *is a deterministic algorithm.*
- $\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1)$ *returns a simulated proof* π*.*

Perfect completeness is satisfied if for all λ*, all* $[\mathbf{M}]_1$*, all* $([\mathbf{y}]_1, \mathbf{w})$ *with* $[\mathbf{y}]_1 = [\mathbf{M}\mathbf{w}]_1$*, all* $(\mathsf{crs}, \mathsf{prk}, \mathsf{vrk}, \mathsf{td}) \in \mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1)$*, and all* $\pi \in \mathsf{Prove}(\mathsf{crs}, \mathsf{prk}, [\mathbf{y}]_1, \mathbf{w})$*, we have*

$$\mathsf{Ver}(\mathsf{crs}, \mathsf{vrk}, [\mathbf{y}]_1, \pi) = 1.$$

Perfect zero knowledge is satisfied if for all λ*, all* $[\mathbf{M}]_1$*, all* $([\mathbf{y}]_1, \mathbf{w})$ *with* $[\mathbf{y}]_1 = [\mathbf{M}\mathbf{w}]_1$*, and all* $(\mathsf{crs}, \mathsf{prk}, \mathsf{vrk}, \mathsf{td}) \in \mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1)$*, the following two distributions are identical:*

$$\mathsf{Prove}(\mathsf{crs}, \mathsf{prk}, [\mathbf{y}]_1, \mathbf{w}) \quad \textit{and} \quad \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1).$$

We define the (unbounded) simulation soundness for all types of QANIZK.

Definition 7 (Unbounded simulation soundness). *Let* $\mathsf{X} \in \{\epsilon, \mathsf{DP}, \mathsf{DV}, \mathsf{DPV}\}$*. An* $\mathsf{XQANIZK}$ $\Pi := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ *is unbounded simulation sound (*USS*) if for any adversary* $\mathcal{A}$*,*

$$\mathsf{Adv}^{\mathsf{uss}}_{\Pi,\mathcal{A}}(\lambda) := \Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1]$$

is negligible, where Game USS *is defined in Figure 1.*

Weak USS. We can also consider a weak notion of simulation-soundness. in the sense that it is only required that $[\mathbf{y}^*]_1 \notin \mathcal{Q}_{\mathsf{sim}}$.[10]

Witness-Samplable Distribution. Here we define simulation soundness for witness-sampleable distributions, namely, Init gets $\mathbf{M} \in \mathbb{Z}_p^{n_1 \times n_2}$ as input, proofs of our $\mathsf{DVQANIZK}$ and QANIZK schemes do not require the explicit $\mathbf{M}$ over $\mathbb{Z}_p$.

[10] In [5], the defined security is this weak version. However, it is not sufficient for constructing a CCA2 secure encryption scheme, since it does not prevent an adversary from forging a new ciphertext for a challenge message and sending that it as a decryption query.

$\underline{\textsc{Init}(\mathbf{M})}$:	$\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)}$:
$(\mathsf{crs}, \mathsf{prk}, \mathsf{vrk}, \mathsf{td}) \xleftarrow{\$} \mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1)$	If $[\mathbf{y}^*]_1 \notin \mathcal{L}_{[\mathbf{M}]_1} \wedge ([\mathbf{y}^*]_1, \pi^*) \notin \mathcal{Q}_{\mathsf{sim}}$ then
Return crs.	return $\mathsf{Ver}(\mathsf{crs}, \mathsf{vrk}, [\mathbf{y}^*]_1, \pi^*)$
	Else return 0
$\underline{\textsc{Sim}([\mathbf{y}]_1)}$: // Q_{s} queries	
$\pi \xleftarrow{\$} \mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1)$	
$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup ([\mathbf{y}]_1, \pi)$	
Return π	

Fig. 1. USS security game for XQANIZK.

In all the standard definitions of (simulation) soundness of QANIZK for linear subspaces, the challenger needs information on $\mathbf{M}$ in $\mathbb{Z}_p$ (not necessary the whole matrix) to check whether the target word $[\mathbf{y}^*]_1$ is inside the language $\mathsf{Span}([\mathbf{M}]_1)$. This information can be a non-zero kernel vector of $\mathbf{M}$ (either in $\mathbb{Z}_p$ or in $\mathbb{G}_2$). We can also define USS with respect to non-witness sampleable distributions while our security proofs (with straightforward modifications) introduced later also hold. In this case, we have to allow the challenger to use super polynomial computational power to check whether $[\mathbf{y}^*]_1 \in \mathsf{Span}(\mathbf{M})$, i.e., then the USS game becomes non-falsifiable. Otherwise, we have to assume that the attacker always gives $[\mathbf{y}^*]_1 \notin \mathsf{Span}(\mathbf{M})$ in USS. In fact, we note that many constructions and applications of simulation-sound QANIZKs consider witness-sampleable distributions (c.f., [32,38,18,29]).

2.5 Structure-Preserving Signature

We now recall the notion of structure-preserving signature (SPS) [3] and unforgeability against chosen message attacks (UF-CMA).

Definition 8 (Signature). *A signature scheme is a tuple of PPT algorithms* $\mathsf{SIG} := (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ *such that:*
- $\mathsf{Gen}(\mathsf{par})$ *returns a verification/signing key pair* $(\mathsf{vk}, \mathsf{sk})$*.*
- $\mathsf{Sign}(\mathsf{sk}, \mathsf{m})$ *returns a signature* σ *for* $\mathsf{m} \in \mathcal{M}$*.*
- $\mathsf{Ver}(\mathsf{vk}, \mathsf{m}, \sigma)$ *returns* 1 *(accept) or* 0 *(reject). Here* Ver *is deterministic.*

Correctness is satisfied if for all $\lambda \in \mathbb{N}$*, all* $\mathsf{m} \in \mathcal{M}$*, and all* $(\mathsf{vk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})$*,*

$$\mathsf{Ver}(\mathsf{vk}, \mathsf{m}, \mathsf{Sign}(\mathsf{sk}, \mathsf{m})) = 1.$$

Definition 9 (Structure-preservation). *A signature scheme is said to be structure-preserving if its verification keys, signing messages, and signatures consist only of group elements and verification proceeds via only a set of pairing product equations.*

Definition 10 (UF-CMA security). *For a signature scheme* $\mathsf{SIG} := (\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ *and any adversary* $\mathcal{A}$*, we define the following experiment:*

$\underline{\textsc{Init}}$:	$\underline{\textsc{SignO}(\mathsf{m})}$:	$\underline{\textsc{Finalize}(\mathsf{m}^*, \sigma^*)}$:
$(\mathsf{vk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(\mathsf{par})$	$\mathcal{Q}_{\mathsf{sign}} := \mathcal{Q}_{\mathsf{sign}} \cup \{\mathsf{m}\}$	If $\mathsf{m}^* \notin \mathcal{Q}_{\mathsf{sign}}$ and $\mathsf{Ver}(\mathsf{vk}, \mathsf{m}^*, \sigma^*) = 1$
Return vk	$\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, \mathsf{m})$	Return 1
	Return σ	Else return 0

Fig. 2. UF-CMA security game for SIG.

A signature scheme SIG *is unforgeable against chosen message attacks* (UF-CMA)*, if for all PPT adversaries* $\mathcal{A}$*,*

$$\mathsf{Adv}^{\mathsf{uf\text{-}cma}}_{\mathsf{SIG},\mathcal{A}}(\lambda) := \Pr[\mathsf{UF\text{-}CMA}^{\mathcal{A}} \Rightarrow 1]$$

is negligible, where Game UF-CMA *is defined in Figure 2.*

3 Quasi-Adaptive NIZK

In this section, we construct a QANIZK with tight simulation soundness. As a stepping stone, we develop a DVQANIZK based on the Matrix Diffie-Hellman assumption. By using the Kernel Matrix Diffie-Hellman assumption and pairings, our DVQANIZK gives us a more efficient QANIZK. All the security reductions in this section are tight.

The Core Lemma. We recall the useful core lemma from [20], which can computationally introduce randomness. More precisely, it shows that moving from experiment Core_0 to Core_1 can (up to negligible terms) only increase the winning chances of an adversary.

$\underline{\textsc{Init}_{\mathsf{core}}}$:	$\underline{\textsc{Eval}_{\mathsf{core}}}$:	$\underline{\textsc{Finalize}_{\mathsf{core}}(\mu)}$:
$\mathsf{c} := 0$	$\mathsf{c} := \mathsf{c} + 1$	Parse $\mu =: ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$
$\mathbf{A}_0, \mathbf{A}_1 \overset{\$}{\leftarrow} \mathcal{D}_{2k,k}$	$\mathbf{s} \overset{\$}{\leftarrow} \mathbb{Z}_p^k, \mathbf{t} := \mathbf{A}_0\mathbf{s} \in \mathbb{Z}_p^{2k}$	If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$
$\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$	$u' := \mathbf{t}^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathsf{c})}) \in \mathbb{Z}_p$	then return 0
$\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$	$\pi_{\mathsf{or}} \overset{\$}{\leftarrow} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$	If $[u']_1 = \mathbf{t}^\top(\mathbf{k} + \boxed{\mathbf{RF}(\mathsf{c}')})$
$\mathbf{k} \overset{\$}{\leftarrow} \mathbb{Z}_p^{2k}$	$\mu := ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$	and $0 \leq \mathsf{c}' \leq \mathsf{c}$ then
$\mathbf{p} := \mathbf{A}_0^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathbf{0})})$	Return μ	return 1
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$		Else return 0
Return crs		

Fig. 3. Security games Core_0 and Core_1 for the core lemma. $\mathbf{RF} : \mathbb{Z}_p \to \mathbb{Z}_p^{2k}$ is a random function. All the codes are executed in both games, except the boxed codes which are only executed in Core_1.

Lemma 3 (Core lemma). *If the* $\mathcal{D}_k$-MDDH *assumption holds in the group* $\mathbb{G}_2$*, and* $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ *is a* NIZK *for* $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ *with perfect* *completeness, perfect soundness, and zero-knowledge, then for any adversary* $\mathcal{A}$ *against the core lemma, there exist adversaries* $\mathcal{B}$, $\mathcal{B}'$ *with running time* $T(\mathcal{B}) \approx T(\mathcal{B}') \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *such that*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{core}}_{\mathcal{A}}(\lambda) :=& \Pr[\mathsf{Core}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{Core}_1^{\mathcal{A}} \Rightarrow 1]\\ \leq& (4k\lceil \log Q\rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_2,\mathcal{D}_{2k,k},\mathcal{B}}(\lambda) + (2\lceil \log Q\rceil + 2) \cdot \mathsf{Adv}^{\mathsf{zk}}_{\mathsf{NIZK},\mathcal{B}'}(\lambda)\\ &+ \lceil \log Q\rceil \cdot \varDelta_{\mathcal{D}_{2k,k}} + \frac{4\lceil \log Q\rceil + 2}{p-1} + \frac{\lceil \log Q\rceil \cdot Q}{p},\end{aligned}$$

where $\varDelta_{\mathcal{D}_{2k,k}}$ *is a statistically small term for* $\mathcal{D}_{2k,k}$.

In a slight departure from [20], we include the term $[\mathbf{A}_0^\top \mathbf{k}]_1$ in crs. We argue that the core lemma still holds by the following reasons (for notation, our $\mathbf{k}$ is their $\mathbf{k}_0$):

– The main purpose of $\mathbf{k}$ is to introduce the constant random function $\mathbf{F}_0(\epsilon)$ in the transition from G_2 to $\mathsf{G}_{3.0}$ in Lemma 4 in [20]. The same argument still holds, given $[\mathbf{A}_0^\top \mathbf{k}]_1$.
– The randomization of Lemma 5 in [20] is done by switching $[\mathbf{t}]_1$ into the right span, and this can be done independent of $\mathbf{k}$. Additionally, we note that, given $[\mathbf{A}_0^\top \mathbf{k}]_1$, one cannot efficiently compute $[\mathbf{t}^\top \mathbf{k}]_1$ without knowing $\mathbf{s} \in \mathbb{Z}_p^k$ s.t. $\mathbf{t} = \mathbf{A}_0 \mathbf{s}$.

We give some brief intuition about the proof of the lemma here. Similar to [20], we re-randomize $\mathbf{k}$ via a sequence of hybrid games. In the i-th hybrid game, we set $u = \mathbf{t}^\top(\mathbf{k} + \mathbf{RF}_i(\mathsf{c}_{|i}))$ where $\mathbf{RF}_i$ is a random function and $\mathsf{c}_{|i}$ denotes the first i-bit prefix of the counter c for queries to $\textsc{Eval}_{\mathsf{core}}$. To proceed from the i-th game to the $(i+1)$-th, we choose $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_{\mathsf{c}_{i+1}})$ in $\textsc{Eval}_{\mathsf{core}}$ depending on the $(i+1)$-th bit of c. We note that the view of the adversary does not change due to the $\mathcal{D}_{2k,k}$-MDDH assumption. Then, as in [20], we can construct $\mathbf{RF}_i$ in the way that it satisfies $\mathbf{t}^\top \mathbf{RF}_{i+1}(\mathsf{c}_{|i+1}) = \mathbf{t}^\top \mathbf{RF}_i(\mathsf{c}_{|i})$. The main difference is that our $\mathbf{RF}_i$ additionally satisfies $\mathbf{A}_0^\top(\mathbf{k} + \mathbf{RF}_{i+1}(0^{i+1})) = \mathbf{A}_0^\top(\mathbf{k} + \mathbf{RF}_i(0^i))$, namely, it not only re-randomizes $\mathbf{k}$ but also ensures that the $\mathbf{A}_0^\top \mathbf{k}$ part in crs is always independent of all the u'-s generated by $\textsc{Eval}_{\mathsf{core}}$. We furthermore make consistent changes to $\textsc{Finalize}_{\mathsf{core}}$ as in [20]. For completeness, we give a detailed proof in Appendix B.

3.1 Stepping Stone: Designated-Verifier QA-NIZK

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, $k \in \mathbb{N}$, $\mathcal{H}$ be a collision-resistant hash function family, and $\varPi^{\mathsf{or}} := (\mathsf{Gen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}})$ be a NIZK system for language $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ (constructed as in Figure 12). Our $\mathsf{DVQANIZK}$ $\varPi^{\mathsf{dv}} := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined as in Figure 4. We note that our scheme can be easily extended to a tag-based scheme by putting the label ℓ inside the hash function. Thus, our scheme can be used in all the applications that require tag-based $\mathsf{DVQANIZK}$.

Theorem 1 (Security of $\varPi^{\mathsf{dv}}$). $\varPi^{\mathsf{dv}}$ *is a* $\mathsf{DVQANIZK}$ *with perfect zero-knowledge and (tightly) unbound simulation soundness. In particular, for any adversary*

$\underline{\mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1 \in \mathbb{G}_1^{n_1 \times n_2})\text{:}}$ $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k},\ H \xleftarrow{\$} \mathcal{H}$ $\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$ $\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$ $\mathbf{k}_0, \mathbf{k}_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1},\ \mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$ $[\mathbf{p}]_1 := [\mathbf{A}_0^\top \mathbf{k}]_1 \in \mathbb{G}_1^k$ $[\mathbf{p}_0]_1 := [\mathbf{M}^\top \mathbf{k}_0]_1 \in \mathbb{G}_1^{n_2},\ [\mathbf{p}_1]_1 := [\mathbf{M}^\top \mathbf{k}_1]_1 \in \mathbb{G}_1^{n_2}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1, [\mathbf{p}_0]_1, [\mathbf{p}_1]_1, H)$ $\mathsf{td} := (\mathbf{k}_0, \mathbf{k}_1)$ $\mathsf{vk} := (\mathbf{k}, \mathbf{k}_0, \mathbf{k}_1)$ Return $(\mathsf{crs}, \mathsf{vk}, \mathsf{td})$ $\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1)\text{:}}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k,\ \mathbf{t} := \mathbf{A}_0 \mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ $[u]_1 := [\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1)]_1 + [\mathbf{s}^\top \mathbf{p}]_1$ Return $\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$	$\underline{\mathsf{Prove}(\mathsf{crs}, [\mathbf{y}]_1, \mathbf{w})\text{:}}$ // $\mathbf{y} = \mathbf{M}\mathbf{w} \in \mathbb{Z}_p^{n_1}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k, [\mathbf{t}]_1 := [\mathbf{A}_0]_1 \mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ $[u]_1 := [\mathbf{w}^\top(\mathbf{p}_0 + \tau \mathbf{p}_1) + \mathbf{s}^\top \mathbf{p}]_1$ Return $\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$ $\underline{\mathsf{Ver}(\mathsf{crs}, \mathsf{vk}, [\mathbf{y}]_1, \pi)\text{:}}$ Parse $\pi = ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$ then return 0 If $[u]_1 = [\mathbf{y}^\top]_1(\mathbf{k}_0 + \tau \mathbf{k}_1) + [\mathbf{t}^\top]_1 \mathbf{k}$ then return 1 Else return 0

Fig. 4. Construction of $\Pi^{\mathsf{dv}} := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$.

$\mathcal{A}$, there exist adversaries $\mathcal{B}$ and $\mathcal{B}'$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A})$ and

$$\begin{aligned}\mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{A}}(\lambda) \leq &\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda) + (4k\lceil\log Q\rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}}(\lambda) \\ &+ (2\lceil\log Q\rceil + 2) \cdot \mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}'}(\lambda) + \lceil\log Q\rceil \cdot \varDelta_{\mathcal{D}_{2k,k}} \\ &+ \frac{4\lceil\log Q\rceil + 2}{p-1} + \frac{(\lceil\log Q\rceil + 1)\cdot Q + 1}{p}.\end{aligned}$$

Proof (of Theorem 1). Perfect completeness follows directly from the correctness of the OR proof system and the fact that for all $\mathbf{y} = \mathbf{M}\mathbf{w}$, $\mathbf{p} := \mathbf{A}_0^\top \mathbf{k}$, $\mathbf{p}_0 := \mathbf{M}^\top \mathbf{k}_0$, $\mathbf{p}_1 := \mathbf{M}^\top \mathbf{k}_1$, and $\mathbf{t} = \mathbf{A}_0 \mathbf{s}$, for any τ, we have

$$\begin{aligned}\mathbf{w}^\top(\mathbf{p}_0 + \tau\mathbf{p}_1) + \mathbf{s}^\top\mathbf{p} &= \mathbf{w}^\top(\mathbf{M}^\top\mathbf{k}_0 + \tau\mathbf{M}^\top\mathbf{k}_1) + \mathbf{s}^\top\mathbf{A}_0^\top\mathbf{k} \\ &= \mathbf{y}^\top(\mathbf{k}_0 + \tau\mathbf{k}_1) + \mathbf{t}^\top\mathbf{k}.\end{aligned}$$

Moreover, since

$$\begin{aligned}\mathbf{w}^\top(\mathbf{p}_0 + \tau\mathbf{p}_1) + \mathbf{s}^\top\mathbf{p} &= \mathbf{w}^\top(\mathbf{M}^\top\mathbf{k}_0 + \tau\mathbf{M}^\top\mathbf{k}_1) + \mathbf{s}^\top\mathbf{p} \\ &= \mathbf{y}^\top(\mathbf{k}_0 + \tau\mathbf{k}_1) + \mathbf{s}^\top\mathbf{p},\end{aligned}$$

proofs generated by Prove and Sim for the same $\mathbf{y} = \mathbf{M}\mathbf{w}$ are identical. Hence, perfect zero knowledge is also satisfied.

We now focus on the tight simulation soundness of Π^{dv}. Let $\mathcal{A}$ be an adversary against the unbounded simulation soundness of Π^{dv}. We bound the advantage of $\mathcal{A}$ via a sequence of games defined in Figure 5.

$\underline{\textsc{Init}(\mathbf{M}):}$ $\boxed{\mathsf{G}_2}$

$i := 0$

$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$, $H \xleftarrow{\$} \mathcal{H}$

$\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$

$\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$

$\mathbf{k}_0, \mathbf{k}_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1}$, $\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$

$[\mathbf{p}]_1 := [\mathbf{A}_0^\top(\mathbf{k}\boxed{+\mathbf{RF}(\mathbf{0})})]_1 \in \mathbb{G}_1^k$

$\mathbf{p}_0 := \mathbf{M}^\top \mathbf{k}_0 \in \mathbb{Z}_p^{n_2}$

$\mathbf{p}_1 := \mathbf{M}^\top \mathbf{k}_1 \in \mathbb{Z}_p^{n_2}$

$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1, [\mathbf{p}_0]_1, [\mathbf{p}_1]_1, H)$

Return crs

$\underline{\textsc{Sim}([\mathbf{y}]_1):}$ // $\boxed{\mathsf{G}_2}$

$\mathsf{c} := \mathsf{c} + 1$

$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{t}]_1 := [\mathbf{A}_0]_1 \mathbf{s}$

$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$

$\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$

$[u]_1 := [\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1) + \mathbf{t}^\top(\mathbf{k}\boxed{+\mathbf{RF}(\mathsf{c})})]_1$

$\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$

$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{([\mathbf{y}]_1, \pi)\}$, $\mathcal{Q}_{\text{tag}} := \mathcal{Q}_{\text{tag}} \cup \{\tau\}$

Return π

$\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*):}$ // G_1-G_2, $\boxed{\mathsf{G}_2}$

Parse $\pi^* =: ([\mathbf{t}^*]_1, [u^*]_1, \pi_{\mathsf{or}}^*)$

$\tau^* := H([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*) \in \mathbb{Z}_p$

If $\tau^* \in \mathcal{Q}_{\text{tag}}$ then return 0

If $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ then return 0

If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*) = 0$ then return 0

$\mathcal{S} := \{[\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* \mathbf{k}_1) + \mathbf{t}^{*\top}(\mathbf{k}\boxed{+\mathbf{RF}(j^*)})]_1 : 0 \le j^* \le \mathsf{c}\}$

If $[u^*]_1 \in \mathcal{S}$ then return 1

Else return 0

Fig. 5. Games G_0, G_1 and G_2 for the proof of Theorem 1. $\mathbf{RF} : \mathbb{Z}_p \to \mathbb{Z}_p^{2k}$ is a random function. Given $\mathbf{M}$ over $\mathbb{Z}_p$, it is efficient to check whether $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$.

G_0 is the real USS experiment for $\mathsf{DVQANIZK}$ as defined in Definition 7.

Lemma 4 (G_0). $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]$.

Lemma 5 (G_0 to G_1). *There is an adversary $\mathcal{B}$ breaking the collision resistance of $\mathcal{H}$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A})$ and $\mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},\mathcal{B}}(\lambda) \ge |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|$.*

Proof. We note that in G_0 and G_1 the value u is uniquely defined by $\mathbf{y}, \mathbf{t}$ and π_{or}. Thus, if $\mathcal{A}$ asks $\textsc{Finalize}$ with $([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*)$ that appears from one of the $\textsc{Sim}$ queries, then $\textsc{Finalize}$ will output 0, since $([\mathbf{y}^*]_1, \pi^* := ([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, [u^*]_1, \pi_{\mathsf{or}}^*)) \in \mathcal{Q}_{\mathsf{sim}}$. Now if $([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*)$ has never appeared from one of the $\textsc{Sim}$ queries, but $\tau^* = H([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*) \in \mathcal{Q}_{\text{tag}}$, the we can construct a straightforward reduction $\mathcal{B}$ to break the CR property of $\mathcal{H}$. □

Lemma 6 (G_1 to G_2). *There is an adversary $\mathcal{B}$ breaking the core lemma (cf. Lemma 3) with running time $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A})$ and $\mathsf{Adv}^{\mathsf{core}}_{\mathcal{B}}(\lambda) = \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]$.*

Proof. We construct the reduction $\mathcal{B}$ defined in Figure 6 to break the core lemma.

Clearly, if $\mathcal{B}$'s oracle access is from Core_0, then $\mathcal{B}$ simulates G_1; and if $\mathcal{B}$'s oracle access is from Core_1 (which uses a random function $\mathbf{RF}$), then $\mathcal{B}$ simulates

$\underline{\textsc{Init}([\mathbf{M}]_1):}$
$i := 0$
$\mathsf{crs}' \xleftarrow{\$} \textsc{Init}_{\mathsf{core}}$
Parse $\mathsf{crs}' =: (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$
$\mathbf{k}_0, \mathbf{k}_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1}$, $H \xleftarrow{\$} \mathcal{H}$
$[\mathbf{p}_0]_1 := [\mathbf{M}^\top \mathbf{k}_0]_1 \in \mathbb{G}_1^{n_2}$
$[\mathbf{p}_1]_1 := [\mathbf{M}^\top \mathbf{k}_1]_1 \in \mathbb{G}_1^{n_2}$
$\mathsf{crs} := (\mathsf{crs}', [\mathbf{p}_0]_1, [\mathbf{p}_1]_1, H)$
Return crs

$\underline{\textsc{Sim}([\mathbf{y}]_1):}$
$\mathsf{c} := \mathsf{c} + 1$
$([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}}) \xleftarrow{\$} \textsc{Eval}_{\mathsf{core}}$
$\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$
$[u]_1 := [\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1) + u']_1$
$\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$
$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{([\mathbf{y}]_1, \pi)\}$, $\mathcal{Q}_{\mathrm{tag}} := \mathcal{Q}_{\mathrm{tag}} \cup \{\tau\}$
Return π

$\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*):}$
Parse $\pi^* =: ([\mathbf{t}^*]_1, [u^*]_1, \pi^*_{\mathsf{or}})$
$\tau^* := H([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi^*_{\mathsf{or}}) \in \mathbb{Z}_p$
If $\tau^* \in \mathcal{Q}_{\mathrm{tag}}$ then return 0
If $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ then
  return 0
$[u'^*]_1 = [u^*]_1 - [\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* \mathbf{k}_1)]_1$
Return $\textsc{Finalize}_{\mathsf{core}}([\mathbf{t}^*]_1, [u'^*]_1, \pi^*_{\mathsf{or}})$

Fig. 6. Reduction $\mathcal{B}$ for the proof of Lemma 6 with oracle $\textsc{Init}_{\mathsf{core}}$, $\textsc{Eval}_{\mathsf{core}}$, $\textsc{Finalize}_{\mathsf{core}}$ defined in Figure 22. We highlight the oracle calls with grey.

G_2. Thus, $\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{Core}_0^{\mathcal{B}} \Rightarrow 1] - \Pr[\mathsf{Core}_1^{\mathcal{B}} \Rightarrow 1] = \mathsf{Adv}_{\mathcal{B}}^{\mathsf{core}}(\lambda)$, which concludes the lemma. □

Lemma 7 (G_2). $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] = \frac{Q}{p}$.

Proof. We apply the following information-theoretical arguments to show that even a computationally unbounded adversary $\mathcal{A}$ can win in G_2 only with negligible probability. If $\mathcal{A}$ wants to win in G_2, then $\mathcal{A}$ needs to output a fresh and valid $\pi^* := ([\mathbf{t}^*]_1, [u^*]_1, \pi^*_{\mathsf{or}})$. According to the additional rejection rule introduced in G_2, $u = \mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* \mathbf{k}_1) + \mathbf{t}^{*\top}(\mathbf{k} + \mathbf{RF}(j^*))$ must hold for some $0 \le j^* \le Q$. Fix a $j^* \le Q$, we show that $\mathcal{A}$ can compute such a u with probability at most $1/p$.

The argument is based on the information leak about $\mathbf{k}_0$ and $\mathbf{k}_1$:

- For the j-th Sim query ($j \neq j^*$), the term $\mathbf{t}^\top \mathbf{RF}(j)$ completely blinds the information about $\mathbf{k}_0$ and $\mathbf{k}_1$ as long as $\mathbf{t} \neq \mathbf{0}$.
- For the j^*-th Sim query, we cannot use the entropy from the term $(\mathbf{k} + \mathbf{RF}(j^*))$ to hide $\mathbf{k}_0$ and $\mathbf{k}_1$ anymore, but we make the following stronger argument. We assume that $\mathcal{A}$ learns the term $\mathbf{t}^\top(\mathbf{k} + \mathbf{RF}(j^*))$, and thus $\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1)$ is also leaked to $\mathcal{A}$. However, since $\tau^* \neq \tau$, the terms $(\mathbf{k}_0 + \tau^* \mathbf{k}_1)$ and $(\mathbf{k}_0 + \tau \mathbf{k}_1)$ are pairwise independent.

Now together with the information leaked from $\mathbf{M}^\top \mathbf{k}_0$ and $\mathbf{M}^\top \mathbf{k}_1$ in crs, from $\mathcal{A}$'s view, the term $\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* \mathbf{k}_1)$ is distributed uniformly at random, given $\mathbf{y}^\top(\mathbf{k}_0 + \tau \mathbf{k}_1)$ from the j^*-th Sim query ($[\mathbf{y}]_1$ may not be in $\mathcal{L}_{[\mathbf{M}]_1}$). Thus, $\mathcal{A}$ can compute the random term $\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* \mathbf{k}_1)$ and make Finalize output 1 with probability at most $1/p$. By the union bound, $\mathcal{A}$ can win in G_2 with probability at most $(Q+1)/p$. □

From Lemmata 4 to 7, we have $\mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{A}}(\lambda) := \Pr[\mathsf{USS}^{\mathcal{A}}] \le \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda) + \mathsf{Adv}^{\mathsf{core}}_{\mathcal{B}'}(\lambda) + \frac{(Q+1)}{p}$. By Lemma 3, we conclude Theorem 1 as

$$\begin{aligned}\mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{A}}(\lambda) \le & \mathsf{Adv}^{\mathsf{cr}}_{\mathcal{H},\hat{\mathcal{B}}}(\lambda) + (4k\lceil \log Q\rceil + 2)\cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}}(\lambda) \\ & + (2\lceil \log Q\rceil + 2)\cdot \mathsf{Adv}^{\mathsf{zk}}_{\mathsf{NIZK},\mathcal{B}'}(\lambda) + \lceil \log Q\rceil \cdot \Delta_{\mathcal{D}_{2k,k}} \\ & + \frac{4\lceil \log Q\rceil + 2}{p-1} + \frac{(\lceil \log Q\rceil + 1)\cdot Q + 1}{p}.\end{aligned}$$

□

3.2 QA-NIZK

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, $k \in \mathbb{N}$, $\mathcal{H}$ be a collision-resistant hash function family, and $\Pi^{\mathsf{or}} := (\mathsf{Gen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}})$ be a NIZK system for language $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$. Our (publicly verifiable) QANIZK $\Pi := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined as in Figure 7. The main idea behind our construction is to tightly compile the $\mathsf{DVQANIZK}$ Π^{dv} from Figure 4 by using pairings. Again we note that our scheme can be easily extended to a tag-based scheme by putting the label ℓ inside the hash function. Thus, our scheme can be used in all the applications that require tag-based QANIZK.

$\underline{\mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1 \in \mathbb{G}_1^{n_1\times n_2}):}$ $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$, $\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$, $H \xleftarrow{\$} \mathcal{H}$ $\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$ $\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$ $\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$ $\mathbf{K}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$, $\mathbf{K}_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$ $\mathbf{P} := \mathbf{A}_0^\top \mathbf{K} \in \mathbb{Z}_p^{k\times(k+1)}$ $[\mathbf{P}_0]_1 := [\mathbf{M}^\top \mathbf{K}_0]_1 \in \mathbb{G}_1^{n_2\times(k+1)}$ $[\mathbf{P}_1]_1 := [\mathbf{M}^\top \mathbf{K}_1]_1 \in \mathbb{G}_1^{n_2\times(k+1)}$ $\mathbf{C} := \mathbf{K}\mathbf{A} \in \mathbb{Z}_p^{2k\times k}$ $\mathbf{C}_0 := \mathbf{K}_0\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathbf{C}_1 := \mathbf{K}_1\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1, [\mathbf{P}_1]_1,$ $\quad [\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2, H)$ $\mathsf{td} := (\mathbf{K}_0, \mathbf{K}_1)$ Return $(\mathsf{crs}, \mathsf{td})$	$\underline{\mathsf{Prove}(\mathsf{crs}, [\mathbf{y}]_1, \mathbf{w}):}$ // $\mathbf{y} = \mathbf{M}\mathbf{w} \in \mathbb{Z}_p^{n_1}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{t}]_1 := [\mathbf{A}_0]_1\mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ $[\mathbf{u}]_1 := \mathbf{w}^\top([\mathbf{P}_0]_1 + \tau[\mathbf{P}_1]_1) + \mathbf{s}^\top[\mathbf{P}]_1 \in \mathbb{G}_1^{1\times(k+1)}$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\underline{\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}]_1, \pi):}$ Parse $\pi = ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$ then return 0 If $[\mathbf{u}]_1 \circ [\mathbf{A}]_2 = [\mathbf{y}^\top]_1 \circ [\mathbf{C}_0 + \tau\mathbf{C}_1]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{C}]_2$ then $\quad$ return 1 Else return 0 $\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1):}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $\mathbf{t} := \mathbf{A}_0\mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ $[\mathbf{u}]_1 := [\mathbf{y}^\top(\mathbf{K}_0 + \tau\mathbf{K}_1)]_1 + [\mathbf{s}^\top\mathbf{P}]_1$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$

Fig. 7. Construction of Π.

Theorem 2 (Security of Π). *Π defined in Figure 7 is a* QANIZK *with perfect zero-knowledge and (tight) unbounded simulation soundness if the $\mathcal{D}_k$-*KerMDH *assumption holds in $\mathbb{G}_2$ and the* DVQANIZK *Π^{dv} in Figure 4 is unbounded simulation sound. In particular, for any adversary $\mathcal{A}$, there exist adversaries $\mathcal{B}$ and $\mathcal{B}'$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{B}') \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$, where Q is the number of queries to* Sim*,* poly *is independent of Q and*

$$\mathsf{Adv}^{\mathsf{uss}}_{\Pi,\mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) + \mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{B}'}(\lambda).$$

Proof (of Theorem 2). Perfect completeness follows directly from the completeness of the OR proof system and the fact that for all $\mathbf{P} := \mathbf{A}_0^\top \mathbf{K}$, $\mathbf{P}_0 := \mathbf{M}^\top \mathbf{K}_0$, $\mathbf{P}_1 := \mathbf{M}^\top \mathbf{K}_1$, $\mathbf{C} := \mathbf{K}\mathbf{A}$, $\mathbf{C}_0 := \mathbf{K}_0\mathbf{A}$, $\mathbf{C}_1 := \mathbf{K}_1\mathbf{A}$, and any τ

$$\begin{aligned}
&[\mathbf{w}^\top(\mathbf{P}_0 + \tau\mathbf{P}_1) + \mathbf{s}^\top\mathbf{P}]_1 \circ [\mathbf{A}]_2 \\
=&[\mathbf{w}^\top(\mathbf{M}^\top\mathbf{K}_0 + \tau\mathbf{M}^\top\mathbf{K}_1) + \mathbf{s}^\top\mathbf{A}_0^\top\mathbf{K}]_1 \circ [\mathbf{A}]_2 \\
=&[\mathbf{w}^\top\mathbf{M}^\top]_1 \circ [\mathbf{K}_0\mathbf{A} + \tau\mathbf{K}_1\mathbf{A}]_2 + [\mathbf{s}^\top\mathbf{A}_0^\top]_1 \circ [\mathbf{K}\mathbf{A}]_2 \\
=&[\mathbf{y}^\top]_1 \circ [\mathbf{C}_0 + \tau\mathbf{C}_1]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{C}]_2.
\end{aligned}$$

Moreover, since

$$\begin{aligned}
\mathbf{w}^\top(\mathbf{P}_0 + \tau\mathbf{P}_1) + \mathbf{s}^\top\mathbf{P} &= \mathbf{w}^\top(\mathbf{M}^\top\mathbf{K}_0 + \tau\mathbf{M}^\top\mathbf{K}_1) + \mathbf{s}^\top\mathbf{P} \\
&= \mathbf{y}^\top(\mathbf{K}_0 + \tau\mathbf{K}_1) + \mathbf{s}^\top\mathbf{P},
\end{aligned}$$

the output of Prove is identical to that of Sim for the same $\mathbf{y} = \mathbf{M}\mathbf{w}$. Hence, perfect zero knowledge is also satisfied.

We now focus on the tight simulation soundness of Π. We prove it by a sequence of games: G_0 is defined as the real experiment, USS (we omit the description here), G_1 and G_2 are defined as in Figure 8.

Lemma 8 (G_0). $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]$.

In G_1, Finalize additionally verifies the adversarial forgery with secret keys $\mathbf{K}$, $\mathbf{K}_0$, and $\mathbf{K}_1$ as in Figure 8.

Lemma 9 (G_0 to G_1). *There is an adversary $\mathcal{B}$ breaking the $\mathcal{D}_k$-*KerMDH *assumption over $\mathbb{G}_2$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and $\mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) \ge |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|$.*

Proof. It is straightforward that a pair $([\mathbf{y}^*]_1, \pi^*)$ passing the Finalize in G_1 always passes the Finalize in G_0. We now bound the probability that $\mathcal{A}$ produces $([\mathbf{y}^*]_1, \pi^*)$ that passes the verification in G_0 but not that in G_1. For $\pi^* = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$, the verification equation in G_0 is:

$$\begin{aligned}
&[\mathbf{u}^*]_1 \circ [\mathbf{A}]_2 = [\mathbf{y}^{*\top}]_1 \circ [\mathbf{K}_0\mathbf{A} + \tau\mathbf{K}_1\mathbf{A}]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{K}\mathbf{A}]_2 \\
&\qquad \Leftrightarrow [\mathbf{u}^* - \mathbf{y}^{*\top}(\mathbf{K}_0 + \tau\mathbf{K}_1) - \mathbf{t}^\top\mathbf{K}]_1 \circ [\mathbf{A}]_2 = [\mathbf{0}]_T.
\end{aligned}$$

$\underline{\text{INIT}([\mathbf{M}]_1):}$ $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k},\ \mathbf{A} \xleftarrow{\$} \mathcal{D}_k,\ H \xleftarrow{\$} \mathcal{H}$ $\boxed{\mathbf{a}^\perp \xleftarrow{\$} \mathsf{ker}(\mathbf{A})}$ $\boxed{/\!/\ \mathbf{a}^\perp \in \mathbb{Z}_p^{1\times(k+1)} \text{ and } \mathbf{a}^\perp \cdot \mathbf{A} = \mathbf{0}}$ $\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$ $\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$ $\mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$ $\mathbf{K}_0' \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)},\ \mathbf{K}_1' \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$ $\mathbf{k}_0 = \mathbf{k}_1 := \mathbf{0} \in \mathbb{Z}_p^{n_1},\ \mathbf{k} := \mathbf{0} \in \mathbb{Z}_p^{2k}$ $\boxed{\mathbf{k}_0, \mathbf{k}_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1},\ \mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}}$ $\mathbf{K} := \mathbf{K}' + \mathbf{k} \cdot \mathbf{a}^\perp$ $\mathbf{K}_0 := \mathbf{K}_0' + \mathbf{k}_0' \cdot \mathbf{a}^\perp$ $\mathbf{K}_1 := \mathbf{K}_1' + \mathbf{k}_1' \cdot \mathbf{a}^\perp$ $\mathbf{P} := \mathbf{A}_0^\top \mathbf{K} \in \mathbb{Z}_p^{k\times(k+1)}$ $[\mathbf{P}_0]_1 := [\mathbf{M}^\top \mathbf{K}_0]_1 \in \mathbb{G}_1^{n_2\times(k+1)}$ $[\mathbf{P}_1]_1 := [\mathbf{M}^\top \mathbf{K}_1]_1 \in \mathbb{G}_1^{n_2\times(k+1)}$ $\mathbf{C} := \mathbf{K}\mathbf{A} \in \mathbb{Z}_p^{2k\times k}$ $\mathbf{C}_0 := \mathbf{K}_0\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathbf{C}_1 := \mathbf{K}_1\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1, [\mathbf{P}_1]_1$ $\quad [\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2, H)$ Return crs	$\underline{\text{SIM}([\mathbf{y}]_1):}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k,\ [\mathbf{t}]_1 := [\mathbf{A}_0]_1\mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $\tau := H([\mathbf{y}]_1, [\mathbf{t}]_1, \pi_{\mathsf{or}}) \in \mathbb{Z}_p$ $[\Delta]_1 := [(\mathbf{y}^\top(\mathbf{k}_0 + \tau\mathbf{k}_1) + \mathbf{t}^\top\mathbf{k}) \cdot \mathbf{a}^\perp]_1$ $[\mathbf{u}]_1 := [\mathbf{y}^\top(\mathbf{K}_0' + \tau\mathbf{K}_1') + \mathbf{t}^\top\mathbf{K}' + \Delta]_1$ $\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{([\mathbf{y}]_1, \pi)\}$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\underline{\text{FINALIZE}([\mathbf{y}^*]_1, \pi^*):}$ Parse $\pi = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi_{\mathsf{or}}^*)$ $\tau^* := H([\mathbf{y}^*]_1, [\mathbf{t}^*]_1, \pi_{\mathsf{or}}^*) \in \mathbb{Z}_p$ If $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ or $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 0$ then $\quad$ return 0 $[\Delta^*]_1 := [({\mathbf{y}^*}^\top(\mathbf{k}_0 + \tau^*\mathbf{k}_1) + {\mathbf{t}^*}^\top\mathbf{k}) \cdot \mathbf{a}^\perp]_1$ If $[\mathbf{u}^*]_1 = [{\mathbf{y}^*}^\top(\mathbf{K}_0' + \tau^*\mathbf{K}_1') + {\mathbf{t}^*}^\top\mathbf{K}' +$ $\Delta^*]_1$ then $\quad$ return 1 Else return 0

Fig. 8. Games G_1 and $\boxed{\mathsf{G}_2}$ for proving Theorem 2.

One can see that for any $([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi_{\mathsf{or}}^*)$ that passes the verification equation in G_0 but not that in G_1, $\mathbf{u}^* - \mathbf{y}^*(\mathbf{K}_0 + \tau\mathbf{K}_1) - \mathbf{t}^\top\mathbf{K}$ is a non-zero vector in the kernel of $\mathbf{A}$.

We now construct an adversary $\mathcal{B}$ as follows. On receiving $(\mathcal{G}, [\mathbf{A}]_1)$ from the $\mathcal{D}_k$-KerMDH experiment, $\mathcal{B}$ samples all other parameters by itself and simulates G_0 for $\mathcal{A}$. When $\mathcal{A}$ outputs a tuple $([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi_{\mathsf{or}}^*)$, $\mathcal{B}$ outputs $\mathbf{u}^* - {\mathbf{y}^*}^\top(\mathbf{K}_0 + \tau\mathbf{K}_1) - \mathbf{t}^\top\mathbf{K}$. Since $\mathcal{B}$ succeeds in its experiment when $\mathcal{A}$ outputs a tuple such that $\mathbf{u}^* - {\mathbf{y}^*}^\top(\mathbf{K}_0 + \tau\mathbf{K}_1) - \mathbf{t}^\top\mathbf{K}$ is a non-zero vector in the kernel of $\mathbf{A}$, we have $\mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) \geq |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|$, completing the proof of this lemma. $\square$

Lemma 10 (G_1 to G_2). $\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]$.

Proof. Now we finish the reduction to the KerMDH assumption and we can have $\mathbf{A}$ over $\mathbb{Z}_p$. In G_2, for $i \in \{0,1\}$ we replace $\mathbf{K}_i$ by $\mathbf{K}_i' + \mathbf{k}_i\mathbf{a}^\perp$ for $\mathbf{a}^\perp \in \mathsf{ker}(\mathbf{A})$, where $\mathbf{K}_i' \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$, and $\mathbf{k}_i \xleftarrow{\$} \mathbb{Z}_p^{n_1}$. Furthermore, we replace $\mathbf{K}$ by $\mathbf{K}' + \mathbf{k}\mathbf{a}^\perp$ for $\mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$ and $\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$. Since $\mathbf{K}'$ and $\mathbf{K}_i'$ are uniformly random, $\mathbf{K}$ and $\mathbf{K}_i$ in G_2 are distributed at random and the same as in G_1. Thus, G_2 is distributed the same as G_1. $\square$

Lemma 11 (G_2)**.** *There is an adversary $\mathcal{B}'$ breaking the* USS *security of Π^{dv} defined in Figure 4 with $\mathsf{T}(\mathcal{B}') \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] \leq \mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{B}'}(\lambda)$.*

Proof. We construct a reduction $\mathcal{B}'$ in Figure 9 to break the USS security of Π^{dv} defined in Figure 4.

$\underline{\textsc{Init}(\mathbf{M})\text{:}}$	$\underline{\textsc{Sim}([\mathbf{y}]_1)\text{:}}$
$\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$ $\mathbf{a}^\perp \xleftarrow{\$} \mathsf{ker}(\mathbf{A})$ // $\mathbf{a}^\perp \in \mathbb{Z}_p^{1\times(k+1)}$ and $\mathbf{a}^\perp\mathbf{A} = \mathbf{0}$ $\mathsf{crs}_{\mathsf{dv}} \xleftarrow{\$} \textsc{Init}_{\mathsf{dv}}(\mathbf{M})$ Parse $\mathsf{crs}_{\mathsf{dv}} =: (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1,$ $[\mathbf{p}_0]_1, [\mathbf{p}_1]_1, H)$ $\mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$ $\mathbf{K}'_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$, $\mathbf{K}'_1 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$ $[\mathbf{P}]_1 := [\mathbf{A}_0]_1^\top\mathbf{K}' + [\mathbf{p}]_1\mathbf{a}^\perp$ $[\mathbf{P}_0]_1 := [\mathbf{M}]_1^\top\mathbf{K}'_0 + [\mathbf{p}_0]_1\mathbf{a}^\perp$ $[\mathbf{P}_1]_1 := [\mathbf{M}]_1^\top\mathbf{K}'_1 + [\mathbf{p}_1]_1\mathbf{a}^\perp$ $\mathbf{C} := \mathbf{K}'\mathbf{A} \in \mathbb{Z}_p^{2k\times k}$ $\mathbf{C}_0 := \mathbf{K}'_0\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathbf{C}_1 := \mathbf{K}'_1\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1, [\mathbf{P}_1]_1$ $[\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2, H)$ Return crs	$([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}}) \xleftarrow{\$} \textsc{Sim}_{\mathsf{dv}}([\mathbf{y}]_1)$ $[\varDelta]_1 := [u]_1 \cdot \mathbf{a}^\perp$ $[\mathbf{u}]_1 := [\mathbf{y}^\top(\mathbf{K}'_0 + \tau\mathbf{K}'_1) + \mathbf{t}^\top\mathbf{K}' + \varDelta]_1$ $\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{([\mathbf{y}]_1, \pi)\}$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)\text{:}}$ Parse $\pi = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$ If $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ or $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 0$ then return 0 Compute $[v]_1$ such that $[v]_1\mathbf{a}^\perp = [\mathbf{u}^* - \mathbf{y}^{*\top}(\mathbf{K}'_0 + \tau^*\mathbf{K}'_1) - \mathbf{t}^{*\top}\mathbf{K}']_1$ Return $\textsc{Finalize}_{\mathsf{dv}}([\mathbf{y}^*]_1, ([\mathbf{t}^*]_1, [v]_1, \pi^*_{\mathsf{or}}))$

Fig. 9. Reduction $\mathcal{B}'$ for the proof of Lemma 11 with oracle access to $\textsc{Init}_{\mathsf{dv}}$, $\textsc{Sim}_{\mathsf{dv}}$ and $\textsc{Finalize}_{\mathsf{dv}}$ as defined in G_0 of Figure 5. We highlight the oracle calls with grey.

We note that the $[\mathbf{p}]_1, [\mathbf{p}_i]_1$ $(i = 0, 1)$ from $\textsc{Init}_{\mathsf{dv}}$ have the forms, $\mathbf{p} = \mathbf{A}_0^\top\mathbf{k}$ and $\mathbf{p}_i = \mathbf{M}^\top\mathbf{k}_i$ for some random $\mathbf{k} \in \mathbb{Z}_p^{2k}$ and $\mathbf{k}_i \in \mathbb{Z}_p^{n_1}$, and furthermore the value $[u]_1$ from $\textsc{Sim}_{\mathsf{dv}}$ has the form $u = \mathbf{y}^\top(\mathbf{k}_0 + \tau\mathbf{k}_1) + \mathbf{t}^\top\mathbf{k}$. Hence, essentially, $\mathcal{B}'$ simulate the security game with $\mathbf{K}$ and $\mathbf{K}_i$ that are implicitly defined as $\mathbf{K} := \mathbf{K}' + \mathbf{k} \cdot \mathbf{a}^\perp$ and $\mathbf{K}_i := \mathbf{K}'_i + \mathbf{k}_i \cdot \mathbf{a}^\perp$. The simulated $\textsc{Init}$ and $\textsc{Sim}$ are identical to those in G_2.

In G_2, $\textsc{Finalize}([\mathbf{y}^*]_1, \pi^* := ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}}))$ outputs 1 if

$$\mathbf{u}^* = \mathbf{y}^{*\top}(\mathbf{K}'_0 + \tau^*\mathbf{K}'_1) + \mathbf{t}^{*\top}\mathbf{K}' + (\underbrace{\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^*\mathbf{k}_1) + \mathbf{t}^{*\top}\mathbf{k}}_{=:v}) \cdot \mathbf{a}^\perp$$

and $([\mathbf{y}^*]_1, \pi^*) \notin \mathcal{Q}_{\mathsf{sim}}$ and $[\mathbf{y}^*]_1 \notin \mathcal{L}_{[\mathbf{M}]_1}$ and $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 1$. Thus, if $\mathcal{A}$ can make $\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)$ output 1 then $\mathcal{B}'$ can extract the corresponding $[v]_1$ to break the USS security. We conclude the lemma. □

To sum up, we have $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] \leq \mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) + \mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{B}'}(\lambda)$ with $\mathcal{B}$ and $\mathcal{B}'$ as defined above. □

3.3 Application: Tightly IND-mCCA-Secure PKE

By instantiating the labeled (enhanced) USS-QA-NIZK in the generic construction (see Figure 16) in [5] with our construction in Section 3.2, we immediately obtain a more efficient publicly verifiable labeled public-key encryption (PKE) with tight IND-CCA2 security in the multi-user, multi-challenge setting (IND-mCCA). The security reduction is independent of the number of decryption-oracle requests of the CCA2 adversary. We refer the reader to Appendix C for the definition of labeled IND-mCCA secure PKE and the construction.

4 Tightly Secure Structure-Preserving Signature

In this section, we present an SPS via a designated-prover NIZK for the OR-language, whose security can be tightly reduced to the $\mathcal{D}_{2k,k}$-MDDH and $\mathcal{D}_k$-MDDH assumptions.

4.1 Designated-Prover OR-Proof

In this section, we construct NIZKs in the designated-prover setting. In contrast to [5], we focus on the language $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ defined in Section 2.3, where a single word $\mathbf{y}$ is required to be in the linear span of either one of two spaces given by matrices $\mathbf{A}_0$ and $\mathbf{A}_1$.

While previous techniques [23,43] require ten group elements in a proof, our novel solution gives a QANIZK with only seven group elements under the SXDH hardness assumption, by leveraging the privacy of the prover CRS.

Definition. For $\mathbf{A}_0, \mathbf{A}_1 \overset{\$}{\leftarrow} \mathcal{D}_{2k,k}$, we define the notion of designated-prover OR-proof for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$.

Definition 11 (Designated-Prover OR-Proof). *A designated-prover proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ is the same as that of* NIZK *for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ (see Section 2.3), except that*

- Gen *takes* $(\mathsf{par}, \mathbf{A}_0, \mathbf{A}_1)$ *as input instead of* $(\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$ *and outputs an additional prover key* prk*.*
- Prove *takes* prk *as additional input.*
- *In the soundness definition, the Adversary is given oracle access to* Prove *with* prk *instantiated by the one output by* Gen*.*

Construction. Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, and $k \in \mathbb{N}$. In Figure 10 we present a Designated-Prover OR-proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$.

Lemma 12. *If the $\mathcal{D}_k$-MDDH assumption holds in the group $\mathbb{G}_2$, then the proof system $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ as defined in Figure 10 is a designated-prover or-proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, perfect soundness, and zero-knowledge. More precisely, for all adversaries $\mathcal{A}$ attacking the zero-knowledge property of Π^{or}, we obtain an adversary $\mathcal{B}$ with $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and $\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda)$.*

$\underline{\mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, \mathbf{A}_0 \in \mathbb{Z}_p^{2k\times k}, \mathbf{A}_1 \in \mathbb{Z}_p^{2k\times k})}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_k$ $\mathbf{u} \xleftarrow{\$} \mathbb{Z}_p^{k+1} \setminus \mathsf{Span}(\mathbf{V})$
$\begin{pmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_k \end{pmatrix} := \underline{\mathbf{A}_1}\overline{\mathbf{A}_1}^{-1} \in \mathbb{Z}_p^{k\times k}$
For $i = 1, \cdots, k$:
  $\mathbf{S}_i \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$, $\mathbf{D}_i := \mathbf{d}_i^\top \mathbf{u}^\top + \mathbf{S}_i \mathbf{V}^\top$
$\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{u}]_2, [\mathbf{V}]_2, ([\mathbf{D}_i]_2)_{1\le i\le k})$
$\mathsf{sk}_{\mathsf{or}} := (\mathbf{A}_0, \mathbf{A}_1, (\mathbf{S}_i)_{1\le i\le k})$
Return $(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}})$

$\underline{\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{r})}$:
Parse $\mathsf{sk}_{\mathsf{or}} =: (\mathbf{A}_0, \mathbf{A}_1, (\mathbf{S}_i)_{1\le i\le k})$
If $\neg(\exists j \in \{0,1\} : [\mathbf{y}]_1 = [\mathbf{A}_j \mathbf{r}]_1)$ then abort
$\mathbf{d} := \begin{pmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_k \end{pmatrix} := \underline{\mathbf{A}_1}\overline{\mathbf{A}_1}^{-1} \in \mathbb{Z}_p^{k\times k}$
$(x_1, \cdots, x_k) := \overline{\mathbf{y}}^\top \mathbf{d}^\top - \underline{\mathbf{y}}^\top \in \mathbb{Z}_p^{1\times k}$
$(\mathbf{x}_1, \cdots, \mathbf{x}_k) := \mathbf{r}(x_1, \cdots, x_k) \in \mathbb{Z}_p^{k\times k}$
For $i = 1, \cdots, k$:
  $\mathbf{R}_i \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$
  $[\mathbf{C}_i]_2 := \mathbf{x}_i[\mathbf{u}^\top]_2 + \mathbf{R}_i[\mathbf{V}^\top]_2$
  $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_p^{1\times k}$
  $[\mathbf{c}_i]_2 := x_i[\mathbf{u}^\top]_2 + \mathbf{r}_i[\mathbf{V}^\top]_2$
  $\mathbf{\Pi}_i := \mathbf{A}_0\mathbf{R}_i - \mathbf{y}\mathbf{r}_i$
  $\pi_i := \overline{\mathbf{y}}^\top \mathbf{S}_i - \mathbf{r}_i$
Return $(([\mathbf{C}_i, \mathbf{c}_i]_2, [\mathbf{\Pi}_i, \pi_i]_1)_{1\le i\le k})$

$\underline{\mathsf{TGen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_k$, $\mathbf{z} \leftarrow \mathbb{Z}_p^k$, $\mathbf{u} := \mathbf{V}\mathbf{z}$
For $i = 1, \cdots, k$:
  $\mathbf{S}_i \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$, $\mathbf{D}_i := \mathbf{S}_i\mathbf{V}^\top$
$\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{u}]_2, [\mathbf{V}]_2, ([\mathbf{D}_i]_2)_{1\le i\le k})$,
$\mathsf{td}_{\mathsf{or}} := (\mathbf{z}, (\mathbf{S}_i)_{1\le i\le k})$
Return $(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}})$

$\underline{\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{y}]_1, ([\mathbf{C}_i, \mathbf{c}_i]_2, [\mathbf{\Pi}_i, \pi_i]_1)_{1\le i\le k})}$:
Parse $\pi =: ([\mathbf{C}_i, \mathbf{c}_i]_2, [\mathbf{\Pi}_i, \pi_i]_1)_{1\le i\le k}$
$[\underline{\mathbf{y}}]_1 =: [(y_1, \cdots, y_k)^\top]_1$
For $i = 1, \cdots, k$:
  If $[\mathbf{A}_0]_1 \circ [\mathbf{C}_i]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}_i]_2 \neq [\mathbf{\Pi}_i]_1 \circ [\mathbf{V}^\top]_2$ then return 0
  If $[\overline{\mathbf{y}}^\top]_1 \circ [\mathbf{D}_i]_2 - [y_i]_1 \circ [\mathbf{u}^\top]_2 - [1]_1 \circ [\mathbf{c}_i]_2 \neq [\pi_i]_1 \circ [\mathbf{V}^\top]_2$ then return 0
Else return 1

$\underline{\mathsf{Sim}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}}, [\mathbf{y}]_1)}$:
Parse $\mathsf{td}_{\mathsf{or}} =: (\mathbf{z}, (\mathbf{S})_{1\le i\le k})$
Parse $[\underline{\mathbf{y}}]_1 =: [(y_1, \cdots, y_k)]_1$
For $i = 1, \cdots, k$,
  $\mathbf{R}_i \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$, $[\mathbf{C}_i]_2 := [\mathbf{R}_i\mathbf{V}^\top]_2$
  $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_p^{1\times k}$, $[\mathbf{c}_i]_2 := [\mathbf{r}_i\mathbf{V}^\top]_2$
  $[\mathbf{\Pi}_i]_1 := [\mathbf{A}_0\mathbf{R}_i - \mathbf{y}\mathbf{r}_i]_1$
  $[\pi_i]_1 := [\overline{\mathbf{y}}^\top \mathbf{S}_i - \mathbf{r}_i - y_i\mathbf{z}^\top]_1$
Return $([\mathbf{C}_i, \mathbf{c}_i]_2, [\mathbf{\Pi}_i, \pi_i]_1)_{1\le i\le k}$

Fig. 10. Designated-prover or-proof for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$.

We refer the reader to Introduction for the high-level idea of our construction. We postpone the detailed proof in Appendix D.

Extensions. For larger matrices $\mathbf{A}_0, \mathbf{A}_1$, and under $\mathcal{D}_k$-MDDH assumption for a fixed k, we improve our proof size so that it asymptotically approaches a factor of two. As a trade-off, it loses a factor of k.

Roughly, for some invertible matrix $\mathbf{U}$, we exploit the following language instead:

$$\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1} := \{[\mathbf{y}]_1 \in \mathbb{G}_1^{2k} \mid \exists \mathbf{x} \in \mathbb{Z}_p^{1\times k}, \mathbf{X} \in \mathbb{Z}_p^{k\times k} \colon \mathbf{A}_0\mathbf{X} = \mathbf{y}\mathbf{x} \vee \mathbf{y}^\top \mathbf{A}_1^\perp \mathbf{U} = \mathbf{x}\}.$$

One can see that it is also equal to $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$, since $\mathbf{y}$ is in the span of $\mathbf{A}_0$ if $\mathbf{x} \neq \mathbf{0}$ and in the span of $\mathbf{A}_1$ otherwise. Instead of directly applying the Groth-Sahai proof to it as before, we make careful adjustment on the proof for $[\mathbf{y}]_1^\top \mathbf{A}_1^\perp \mathbf{U} =$

$[\mathbf{x}]_1$ and commitment of the information on $\mathbf{A}_1^\perp$ in this case. We also extend it to an efficient OR-Proof in the symmetric pairing, which might be of independent interest. We refer the reader to Appendix E for the constructions and security proofs.

4.2 Structure-Preserving Signature

By replacing the underlying OR-proof in the SPS in [20] with our designated-prover one, we immediately obtain a more efficient SPS. A signature consists only of 11 elements, which is the shortest known for tightly secure SPS-es.

$\underline{\mathsf{Gen}(\mathsf{par})}$:	$\underline{\mathsf{Sign}(\mathsf{vk}, \mathsf{sk}, [\mathbf{m}]_1 \in \mathbb{G}_1^n)}$:
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$	$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{t}]_1 := [\mathbf{A}_0]_1 \mathbf{r}$
$(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}) \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, \mathbf{A}_0, \mathbf{A}_1)$	$\pi_{\mathsf{or}} \leftarrow \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{r})$
$\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$	$[\mathbf{u}]_1 := \mathbf{K}_0^\top [\mathbf{t}]_1 + \mathbf{K}^\top \begin{bmatrix} \mathbf{m} \\ 1 \end{bmatrix}_1$
$\mathbf{K}_0 \xleftarrow{\$} \mathbb{Z}_p^{2k \times (k+1)}$	
$\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{(n+1) \times (k+1)}$	Return $\sigma := ([\mathbf{t}]_1, \pi_{\mathsf{or}}, [\mathbf{u}]_1)$
$\mathbf{C}_0 = \mathbf{K}_0 \mathbf{A} \in \mathbb{Z}_p^{2k \times k}$	
$\mathbf{C} = \mathbf{K}\mathbf{A} \in \mathbb{Z}_p^{(n+1) \times k}$	$\underline{\mathsf{Ver}(\mathsf{vk}, \sigma, [\mathbf{m}]_1)}$:
$\mathsf{vk} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{A}]_2, [\mathbf{C}_0]_2, [\mathbf{C}]_2)$	Parse $\sigma := ([\mathbf{t}]_1, \pi_{\mathsf{or}}, [\mathbf{u}]_1)$
$\mathsf{sk} := (\mathbf{K}_0, \mathbf{K}, \mathsf{sk}_{\mathsf{or}})$	$b \leftarrow \mathsf{Ver}_{\mathsf{or}}(\mathsf{vk}, [\mathbf{t}]_1, \pi_{\mathsf{or}})$
Return $(\mathsf{vk}, \mathsf{sk})$	If $b = 1$ and
	$[\mathbf{u}^\top]_1 \circ [\mathbf{A}]_2 = [\mathbf{t}^\top]_1 \circ [\mathbf{C}_0]_2 + [\mathbf{m}^\top, 1]_1 \circ [\mathbf{C}]_2$
	return 1
	Else return 0

Fig. 11. Tightly UF-CMA secure structure-preserving signature scheme Σ with message space $\mathbb{G}_1^n$. $k \in \mathbb{N}$ and the public parameter is $\mathsf{par} = \mathcal{G}$ where $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$.

Theorem 3 (Security of Σ). *If $\Pi^{\mathsf{or}} := (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ is a non-interactive zero-knowledge proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$, the signature scheme Σ described in Figure 11 is* UF-CMA *secure under the $\mathcal{D}_{2k,k}$-*MDDH *and $\mathcal{D}_k$-*MDDH *assumptions. Namely, for any adversary $\mathcal{A}$, there exist adversaries $\mathcal{B}, \mathcal{B}'$ with running time $T(\mathcal{B}) \approx T(\mathcal{B}') \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$, where Q is the number of signing queries,* poly *is independent of Q, and*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{uf\text{-}cma}}_{\mathsf{SPS},\mathcal{A}} \leq &(4k\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1, \mathcal{D}_{2k,k}, \mathcal{B}} \\ &+ (2\lceil \log Q \rceil + 3) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_2, \mathcal{D}_k, \mathcal{B}'} + \lceil \log Q \rceil \cdot \Delta_{\mathcal{D}_{2k,k}} \\ &+ \frac{4\lceil \log Q \rceil + 2}{p-1} + \frac{(Q+1)\lceil \log Q \rceil + Q}{p} + \frac{Q}{p^k}.\end{aligned}$$

We omit the proof of the above theorem since it is exactly the same as the security proof of the SPS in [20] except that we adopt the notion of standard zero knowledge instead of the composable one and the OR-proof system is a designated-prover one now, which does not affect the validity of the proof at all. We refer the reader to [20] for the details. Notice that in the MDDH games of the security proof, the reduction algorithm is not allowed to see $\mathbf{A}_0$ and $\mathbf{A}_1$ so that it cannot run the honest generation algorithm $\mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, \mathbf{A}_0, \mathbf{A}_1)$. However, it does not have to, since in all the MDDH games, common reference strings are always switched to simulated ones, namely, the reduction algorithms only have to run $\mathsf{TGen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$.

4.3 DPQANIZK and Black-Box Construction

We can also use our designated-or-proof system to construct a structure-preserving DPQANIZK with weak USS, which might be of independent interest. We refer the reader to Appendix F for the construction and security proof of it.

On the other hand, as shown in [5,6], there is an alternative approach for constructing SPS directly from DPQANIZK. It is just mapping a message to an invalid instance out of the language and simulating a proof with a trapdoor behind a common reference string published as a public key. In the concrete construction in [5,6], $n_0 + 1$ extra elements are included in a public key so that they are used to make sure that messages consisting of n_0 elements are certainly mapped to invalid instances. We can take the same approach but with improved mapping that requires only one extra element assuming the hardness of the computational Diffie-Hellman problem. The resulting signature size is exactly the same as that of proofs of DPQANIZK and the public-key size is that of a common-reference string plus one element.

References

1. Abdalla, M., Benhamouda, F., Pointcheval, D.: Disjunctions for hash proof systems: New constructions and applications. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 69–100. Springer, Heidelberg (Apr 2015) 2
2. Abe, M., Chase, M., David, B., Kohlweiss, M., Nishimaki, R., Ohkubo, M.: Constant-size structure-preserving signatures: Generic constructions and simple assumptions. Journal of Cryptology 29(4), 833–878 (Oct 2016) 6
3. Abe, M., Fuchsbauer, G., Groth, J., Haralambiev, K., Ohkubo, M.: Structure-preserving signatures and commitments to group elements. Journal of Cryptology 29(2), 363–421 (Apr 2016) 2, 15
4. Abe, M., Hofheinz, D., Nishimaki, R., Ohkubo, M., Pan, J.: Compact structure-preserving signatures with almost tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part II. LNCS, vol. 10402, pp. 548–580. Springer, Heidelberg (Aug 2017) 3, 4, 6, 9
5. Abe, M., Jutla, C.S., Ohkubo, M., Roy, A.: Improved (almost) tightly-secure simulation-sound QA-NIZK with applications. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part I. LNCS, vol. 11272, pp. 627–656. Springer, Heidelberg (Dec 2018) 4, 5, 6, 9, 14, 25, 28, 32, 38, 39
6. Abe, M., Jutla, C.S., Ohkubo, M., Roy, A.: Improved (almost) tightly-secure simulation-sound QA-NIZK with applications. IACR Cryptology ePrint Archive 2018, 849 (2018) 4, 5, 6, 28, 39, 40

7. Belenkiy, M., Chase, M., Kohlweiss, M., Lysyanskaya, A.: P-signatures and noninteractive anonymous credentials. In: Canetti, R. (ed.) TCC 2008. LNCS, vol. 4948, pp. 356–374. Springer, Heidelberg (Mar 2008) 2, 3
8. Blazy, O., Kakvi, S.A., Kiltz, E., Pan, J.: Tightly-secure signatures from chameleon hash functions. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 256–279. Springer, Heidelberg (Mar / Apr 2015) 3
9. Blazy, O., Kiltz, E., Pan, J.: (Hierarchical) identity-based encryption from affine message authentication. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 408–425. Springer, Heidelberg (Aug 2014) 3, 6, 8, 10
10. Blum, M., Feldman, P., Micali, S.: Non-interactive zero-knowledge and its applications (extended abstract). In: 20th ACM STOC. pp. 103–112. ACM Press (May 1988) 12
11. Boneh, D., Franklin, M.K.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (Aug 2001) 2
12. Chen, J., Gay, R., Wee, H.: Improved dual system ABE in prime-order groups via predicate encodings. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 595–624. Springer, Heidelberg (Apr 2015) 8
13. Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 435–460. Springer, Heidelberg (Aug 2013) 3
14. Cramer, R., Shoup, V.: Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (Apr / May 2002) 6
15. Escala, A., Groth, J.: Fine-tuning Groth-Sahai proofs. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 630–649. Springer, Heidelberg (Mar 2014) 9
16. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.: An algebraic framework for Diffie-Hellman assumptions. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 129–147. Springer, Heidelberg (Aug 2013) 4, 11, 12
17. Fischlin, M., Libert, B., Manulis, M.: Non-interactive and re-usable universally composable string commitments with adaptive security. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 468–485. Springer, Heidelberg (Dec 2011) 2
18. Gay, R., Hofheinz, D., Kiltz, E., Wee, H.: Tightly CCA-secure encryption without pairings. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 1–27. Springer, Heidelberg (May 2016) 3, 4, 5, 7, 13, 15
19. Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 133–160. Springer, Heidelberg (Aug 2017) 3, 9
20. Gay, R., Hofheinz, D., Kohl, L., Pan, J.: More efficient (almost) tightly secure structure-preserving signatures. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018, Part II. LNCS, vol. 10821, pp. 230–258. Springer, Heidelberg (Apr / May 2018) 3, 4, 6, 7, 8, 9, 10, 13, 16, 17, 27, 28, 32
21. Groth, J.: Fully anonymous group signatures without random oracles. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 164–180. Springer, Heidelberg (Dec 2007) 2
22. Groth, J., Lu, S.: A non-interactive shuffle with pairing based verifiability. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 51–67. Springer, Heidelberg (Dec 2007) 3

23. Groth, J., Ostrovsky, R., Sahai, A.: New techniques for noninteractive zero-knowledge. J. ACM 59(3), 11:1–11:35 (Jun 2012), http://doi.acm.org/10.1145/2220357.2220358 9, 25, 32
24. Groth, J., Sahai, A.: Efficient non-interactive proof systems for bilinear groups. In: Smart, N.P. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 415–432. Springer, Heidelberg (Apr 2008) 2, 9, 12
25. Hartung, G., Hoffmann, M., Nagel, M., Rupp, A.: BBA+: Improving the security and applicability of privacy-preserving point collection. In: Thuraisingham, B.M., Evans, D., Malkin, T., Xu, D. (eds.) ACM CCS 2017. pp. 1925–1942. ACM Press (Oct / Nov 2017) 3
26. Hofheinz, D.: Algebraic partitioning: Fully compact and (almost) tightly secure cryptography. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016-A, Part I. LNCS, vol. 9562, pp. 251–281. Springer, Heidelberg (Jan 2016) 3
27. Hofheinz, D.: Adaptive partitioning. In: Coron, J., Nielsen, J.B. (eds.) EUROCRYPT 2017, Part III. LNCS, vol. 10212, pp. 489–518. Springer, Heidelberg (Apr / May 2017) 3, 9
28. Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 590–607. Springer, Heidelberg (Aug 2012) 3, 6
29. Hofheinz, D., Jia, D., Pan, J.: Identity-based encryption tightly secure under chosen-ciphertext attacks. In: Peyrin, T., Galbraith, S. (eds.) ASIACRYPT 2018, Part II. LNCS, vol. 11273, pp. 190–220. Springer, Heidelberg (Dec 2018) 2, 3, 4, 6, 7, 15
30. Joux, A.: A one round protocol for tripartite Diffie-Hellman. In: Algorithmic Number Theory, 4th International Symposium, ANTS-IV, Leiden, The Netherlands, July 2-7, 2000, Proceedings. pp. 385–394 (2000) 2
31. Jutla, C.S., Ohkubo, M., Roy, A.: Improved (almost) tightly-secure structure-preserving signatures. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part II. LNCS, vol. 10770, pp. 123–152. Springer, Heidelberg (Mar 2018) 4, 6
32. Jutla, C.S., Roy, A.: Shorter quasi-adaptive NIZK proofs for linear subspaces. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 1–20. Springer, Heidelberg (Dec 2013) 2, 13, 15
33. Jutla, C.S., Roy, A.: Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 295–312. Springer, Heidelberg (Aug 2014) 2, 8
34. Jutla, C.S., Roy, A.: Improved structure preserving signatures under standard bilinear assumptions. In: Fehr, S. (ed.) PKC 2017, Part II. LNCS, vol. 10175, pp. 183–209. Springer, Heidelberg (Mar 2017) 4, 6
35. Jutla, C.S., Roy, A.: Smooth NIZK arguments. In: Beimel, A., Dziembowski, S. (eds.) TCC 2018, Part I. LNCS, vol. 11239, pp. 235–262. Springer, Heidelberg (Nov 2018) 12, 43
36. Kiltz, E., Pan, J., Wee, H.: Structure-preserving signatures from standard assumptions, revisited. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part II. LNCS, vol. 9216, pp. 275–295. Springer, Heidelberg (Aug 2015) 4, 6, 8
37. Kiltz, E., Wee, H.: Quasi-adaptive NIZK for linear subspaces revisited. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015, Part II. LNCS, vol. 9057, pp. 101–128. Springer, Heidelberg (Apr 2015) 4, 5, 6, 8, 12, 13, 32
38. Libert, B., Peters, T., Joye, M., Yung, M.: Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 514–532. Springer, Heidelberg (May 2014) 2, 5, 15

39. Libert, B., Peters, T., Joye, M., Yung, M.: Compactly hiding linear spans - tightly secure constant-size simulation-sound QA-NIZK proofs and applications. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 681–707. Springer, Heidelberg (Nov / Dec 2015) 5
40. Libert, B., Peters, T., Yung, M.: Short group signatures via structure-preserving signatures: Standard model security from simple assumptions. In: Gennaro, R., Robshaw, M.J.B. (eds.) CRYPTO 2015, Part II. LNCS, vol. 9216, pp. 296–316. Springer, Heidelberg (Aug 2015) 6
41. Morillo, P., Ràfols, C., Villar, J.L.: The kernel matrix Diffie-Hellman assumption. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part I. LNCS, vol. 10031, pp. 729–758. Springer, Heidelberg (Dec 2016) 11, 12
42. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: 22nd ACM STOC. pp. 427–437. ACM Press (May 1990) 2
43. Ràfols, C.: Stretching groth-sahai: NIZK proofs of partial satisfiability. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part II. LNCS, vol. 9015, pp. 247–276. Springer, Heidelberg (Mar 2015) 9, 25, 32

Supplementary Material

A A NIZK system for OR-language

We now recall a NIZK for an OR-language, which will be used as a building block of our QANIZK proof. It was firstly given in [23] and later generalized in [43] for more general languages. Up until now, this instantiation has been used in several works [20,5] to achieve tight security. We use it as well.

$\underline{\mathsf{Gen}_{\mathsf{or}}(\mathsf{par})}$:
$\mathbf{D} \xleftarrow{\$} \mathcal{D}_k,\ \mathbf{z} \xleftarrow{\$} \mathbb{Z}_p^{k+1} \setminus \mathsf{Span}(\mathbf{D})$
$\mathsf{crs} := (\mathsf{par}, [\mathbf{D}]_2, [\mathbf{z}]_2)$
Return crs

$\underline{\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}, [\mathbf{x}]_1, \mathbf{r})}$:
let $j \in \{0,1\}$ s.t. $[\mathbf{x}]_1 = [\mathbf{A}_j]_1 \cdot \mathbf{r}$
$\mathbf{v} \xleftarrow{\$} \mathbb{Z}_p^k$
$[\mathbf{z}_{1-j}]_2 := [\mathbf{D}]_2 \cdot \mathbf{v}$
$[\mathbf{z}_j]_2 := [\mathbf{z}]_2 - [\mathbf{z}_{1-j}]_2$
$\mathbf{S}_0, \mathbf{S}_1 \xleftarrow{\$} \mathbb{Z}_p^{k \times k}$
$[\mathbf{C}_j]_2 := \mathbf{S}_j \cdot [\mathbf{D}]_2^\top + \mathbf{r} \cdot [\mathbf{z}_j]_2^\top$
$[\Pi_j]_1 := [\mathbf{A}_j]_1 \cdot \mathbf{S}_j$
$[\mathbf{C}_{1-j}]_2 := \mathbf{S}_{1-j} \cdot [\mathbf{D}]_2^\top$
$[\Pi_{1-j}]_1 := [\mathbf{A}_{1-j}]_1 \cdot \mathbf{S}_{1-j} - [\mathbf{x}]_1 \cdot \mathbf{v}^\top$
Return $([\mathbf{z}_0]_2, ([\mathbf{C}_i]_2, [\Pi_i]_1)_{i \in \{0,1\}})$

$\underline{\mathsf{TGen}_{\mathsf{or}}(\mathsf{par})}$:
$\mathbf{D} \xleftarrow{\$} \mathcal{D}_k,\ \mathbf{u} \xleftarrow{\$} \mathbb{Z}_p^k$
$\mathbf{z} := \mathbf{D} \cdot \mathbf{u}$
$\mathsf{crs} := (\mathsf{par}, [\mathbf{D}]_2, [\mathbf{z}]_2),\ \mathsf{td} := \mathbf{u}$
Return $(\mathsf{crs}, \mathsf{td})$

$\underline{\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}, [\mathbf{x}]_1, ([\mathbf{z}_0]_2, ([\mathbf{C}_i]_2, [\Pi_i]_1)_{i \in \{0,1\}}))}$:
$[\mathbf{z}_1]_2 := [\mathbf{z}]_2 - [\mathbf{z}_0]_2$
if for all $i \in \{0,1\}$ it holds
$[\mathbf{A}_i]_1 \circ [\mathbf{C}_i]_2$
$= [\Pi_i]_1 \circ [\mathbf{D}^\top]_2 + [\mathbf{x}]_1 \circ [\mathbf{z}_i^\top]_2$
return 1
Else return 0

$\underline{\mathsf{Sim}_{\mathsf{or}}(\mathsf{crs}, \mathsf{td}, [\mathbf{x}]_1)}$:
parse $\mathsf{td} =: \mathbf{u}$
$\mathbf{v} \xleftarrow{\$} \mathbb{Z}_p^k$
$[\mathbf{z}_0]_2 := [\mathbf{D}]_2 \cdot \mathbf{v}$
$[\mathbf{z}_1]_2 := [\mathbf{z}]_2 - [\mathbf{z}_0]_2$
$\mathbf{S}_0, \mathbf{S}_1 \xleftarrow{\$} \mathbb{Z}_p^{k \times k}$
$[\mathbf{C}_0]_2 := \mathbf{S}_0 \cdot [\mathbf{D}]_2^\top$
$[\Pi_0]_1 := [\mathbf{A}_0]_1 \cdot \mathbf{S}_0 - [\mathbf{x}]_1 \cdot \mathbf{v}^\top$
$[\mathbf{C}_1]_2 := \mathbf{S}_1 \cdot [\mathbf{D}]_2^\top$
$[\Pi_1]_1 := [\mathbf{A}_1]_1 \cdot \mathbf{S}_1 - [\mathbf{x}]_1 \cdot (\mathbf{u} - \mathbf{v})^\top$
Return $([\mathbf{z}_0]_2, ([\mathbf{C}_i]_2, [\Pi_i]_1)_{i \in \{0,1\}})$

Fig. 12. Construction of Π^{or} for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ ([23,43])

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $k \in \mathbb{N}$, $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$, $\mathsf{par} := (\mathcal{G}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$. In Figure 12 we give the NIZK proof scheme $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$, which was previously presented in [37] and also implicitly given in [23,43], for the OR-language

$$\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1} := \{[\mathbf{x}]_1 \in \mathbb{G}_1^{2k} \mid \exists \mathbf{r} \in \mathbb{Z}_p^k \colon [\mathbf{x}]_1 = [\mathbf{A}_0]_1 \cdot \mathbf{r} \vee [\mathbf{x}]_1 = [\mathbf{A}_1]_1 \cdot \mathbf{r}\}.$$

Lemma 13. *If the $\mathcal{D}_k$-MDDH assumption holds in the group $\mathbb{G}_2$, then the proof system $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ is a NIZK for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, perfect soundness, and zero-knowledge. More precisely, for* *any adversary $\mathcal{A}$ against the zero-knowledge of Π^{or}, there exists an adversary $\mathcal{B}$ with $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and*

$$\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda).$$

B The Proof of the Core Lemma

In this section, we give the proof of the core lemma (see Lemma 3).

Proof (of Lemma 3). Let $\mathcal{A}$ be any adversary. We proceed via a series of hybrid games $\mathsf{G}_0, \ldots, \mathsf{G}_{3.\lceil \log Q \rceil}$, described in Figure 13, and we denote by ε_i the advantage of $\mathcal{A}$ to win G_i.

G_0, $\boxed{\mathsf{G}_1, \mathsf{G}_2, \mathsf{G}_{3.i}}$:

<u>$\textsc{Init}_{\mathsf{core}}$:</u>
$\mathsf{c} := 0$
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$
$\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$
$\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^{\lambda})$
$\boxed{(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}) \leftarrow \mathsf{TGen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}})}$
$\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$
$\mathbf{p} := \mathbf{A}_0^{\top}(\mathbf{k} + \mathbf{RF}_i(0^i))$
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$
Return crs

<u>$\textsc{Eval}_{\mathsf{core}}$:</u>
$\mathsf{c} := \mathsf{c} + 1$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k, [\mathbf{t}]_1 := [\mathbf{A}_0\mathbf{s}]_1$
$\mathbf{t} \xleftarrow{\$} \mathbb{Z}_p^{2k}$
$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$
$\boxed{\pi_{\mathsf{or}} \leftarrow \mathsf{Sim}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}, [\mathbf{t}]_1)}$
$[u']_1 := [\mathbf{t}^{\top}(\mathbf{k} + \mathbf{RF}_i(\mathsf{c}_{|i}))]_1$
$\mu := ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$
Return μ

<u>$\textsc{Finalize}_{\mathsf{core}}(\mu)$:</u>
Parse $\mu = ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$
$b := \mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}})$
If $b = 1$ and $\exists 0 \leq \mathsf{c}' \leq \mathsf{c}$:
  $[u']_1 := [\mathbf{t}]_1^{\top}(\mathbf{k} + \mathbf{RF}_i(\mathsf{c}'_{|i}))$
  return 1
Else return 0

Fig. 13. Games $\mathsf{G}_0, \mathsf{G}_1, \mathsf{G}_2, \mathsf{G}_{3.i}$ for $i \in \{0, \ldots, \lceil \log Q \rceil - 1\}$, for the proof of the core lemma (Lemma 3). $\mathbf{RF}_i : \{0,1\}^i \to \mathbb{Z}_p^{2k}$ denotes a random function, and $\mathsf{c}_{|i}$ denotes the i-bit prefix of the counter c written in binary. In each procedure, the components inside a solid (dotted, gray) frame are only present in the games marked by a solid (dotted, gray) frame.

G_0*:* By definition, we have

$$\varepsilon_0 = \Pr[\mathsf{Core}_0^{\mathcal{A}} \Rightarrow 1].$$

G_0 *to* G_1*:* Game G_1 is the same as G_0, except that $\mathsf{crs}_{\mathsf{or}}$ is generated by $\mathsf{TGen}_{\mathsf{or}}$ instead of $\mathsf{Gen}_{\mathsf{or}}$ and proofs are generated by $\mathsf{Sim}_{\mathsf{or}}$. We now construct an adversary $\mathcal{B}$ against the zero-knowledge property of NIZK as follows.

$\mathcal{B}$ runs the game similar to G_0, except that it uses $\mathsf{crs}_{\mathsf{or}}$ from its own experiment instead of sampling it by itself. Then it answers $\textsc{Eval}_{\mathsf{core}}$ and $\textsc{Finalize}_{\mathsf{core}}$ queries by using $\mathbf{k}$ and its own oracle which returns honest or simulated proofs π_{or}. Since $\mathcal{B}$ simulates G_0 (respectively, G_1) when $\mathsf{crs}_{\mathsf{or}}$ and π_{or} are generated by $\mathsf{Gen}_{\mathsf{or}}$ and $\mathsf{Prove}_{\mathsf{or}}$ (respectively, $\mathsf{TGen}_{\mathsf{or}}$ and $\mathsf{Sim}_{\mathsf{or}}$), we have $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\varepsilon_0 - \varepsilon_1| \leq \mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}}(\lambda).$$

G_1 *to* G_2*:* Game G_1 is the same as G_0, except that we switch $[\mathbf{t}]_1$ to random over $\mathbb{G}_1$. We now construct an adversary $\mathcal{B}_1$ against the Q-fold $\mathcal{D}_{2k,k}$-MDDH assumption as follows.

On input $(\mathcal{G}, [\mathbf{A}_0]_1, [\mathbf{z}_1]_1, \ldots, [\mathbf{z}_Q]_1)$, $\mathcal{B}_1$ sets up the game similar to G_1, except that it uses $[\mathbf{A}_0]_1$ from its own experiment instead of sampling $\mathbf{A}_0 \xleftarrow{\$} \mathcal{D}_{2k,k}$ by itself. Furthermore, to answer $\textsc{Eval}_{\mathsf{core}}$ queries $\mathcal{B}_1$ sets $[\mathbf{t}_i]_1 := [\mathbf{z}_i]_1$ (where $\mathbf{t}_i$ denotes $\mathbf{t}$ generated in Figure 13 for the ith $\textsc{Eval}_{\mathsf{core}}$ query) and computes the rest accordingly. Since $\mathcal{B}_1$ simulates G_1 when given a real $\mathcal{D}_{2k,k}$-challenge and simulates G_2 otherwise, according to Lemma 2, we have $T(\mathcal{B}_1) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\varepsilon_1 - \varepsilon_2| \leq k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_1}(\lambda) + \frac{1}{p-1}.$$

G_2 *to* $\mathsf{G}_{3.0}$*:* We denote 0^0 as empty string ϵ. In $\mathsf{G}_{3.0}$, instead of sampling a random $\mathbf{k}$, we use $\mathbf{k} + \mathbf{RF}_0(\epsilon)$ to simulate the security game, where $\mathbf{RF}_0(\epsilon)$ is a fixed random vector. We have

$$\varepsilon_2 = \varepsilon_{3.0}.$$

$\mathsf{G}_{3.i}$ *to* $\mathsf{G}_{3.(i+1)}$*:* There exist adversaries $\mathcal{B}_i, \mathcal{B}'_i$ such that $T(\mathcal{B}_i) \approx T(\mathcal{B}'_i) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$, and

$$\begin{aligned}\varepsilon_{3.i} \leq &\varepsilon_{3.(i+1)} + 4k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_i}(\lambda) + 2\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}'_i}(\lambda)\\ &+ \Delta_{\mathcal{D}_{2k,k}} + \frac{4}{p-1} + \frac{Q}{p}.\end{aligned}$$

We refer the reader to Lemma 14 for this part of proof.

$\mathsf{G}_{3.\lceil \log Q \rceil}$ *to* Core_1*:* We now introduce an intermediary game G_4, where we set $[\mathbf{t}]_1 := [\mathbf{A}_0]_1\mathbf{r}$ for $\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^k$. This corresponds to reversing transition from G_1 to G_2, and by the similar reasoning we obtain an adversary $\mathcal{B}_{3.\lceil \log Q \rceil}$ with $T(\mathcal{B}_{3.\lceil \log Q \rceil}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ such that

$$|\varepsilon_{3.\lceil \log Q \rceil} - \varepsilon_4| \leq k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_{3.\lceil \log Q \rceil}}(\lambda) + \frac{1}{p-1}.$$

We now switch back to honest generation of $\mathsf{crs}_{\mathsf{or}}$ and π_{or}. This corresponds to reversing transition G_0 to G_1, and by the same reasoning we obtain an adversary $\mathcal{B}_4$ such that $T(\mathcal{B}_4) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\varepsilon_4 - \Pr[\mathsf{Core}_1^{\mathcal{A}} \Rightarrow 1]| \leq \mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}_4}(\lambda),$$

completing the proof. ◻

Lemma 14 **($\mathsf{G}_{3.i}$ to $\mathsf{G}_{3.(i+1)}$).** *If the $\mathcal{D}_{2k,k}$-MDDH assumption holds in $\mathbb{G}_1$, and $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ is a NIZK for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, perfect soundness, and composable zero-knowledge, then for all $i \in \{0, \dots, \lceil \log Q \rceil - 1\}$, there exist adversaries $\mathcal{B}_i$, $\mathcal{B}'_i$ such that $T(\mathcal{B}_i) \approx T(\mathcal{B}'_i) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and*

$$\begin{aligned}\varepsilon_{3.i} \leq & \varepsilon_{3.(i+1)} + 4k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_i}(\lambda) + 2\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}'_i}(\lambda) \\ & + 2\Delta_{\mathcal{D}_{2k,k}} + \frac{4}{p-1} + \frac{Q}{p}.\end{aligned}$$

Proof. We proceed via a series of hybrid games $\mathsf{H}_{i.j}$ for $i \in \{0, \dots, \lceil \log Q \rceil - 1\}$, $j \in \{1, \dots, 8\}$, described in Figure 14, and we denote by $\hat{\varepsilon}_{i.j}$ the advantage of $\mathcal{A}$ to win $\mathsf{H}_{i.j}$.

$\mathsf{G}_{3.i}$ to $\mathsf{H}_{i.1}$: For $\textsc{Eval}_{\mathsf{core}}$ queries, we choose $[\mathbf{t}]_1$ randomly from $\mathsf{Span}([\mathbf{A}_0]_1)$ instead of the whole $\mathbb{Z}_p^{2k}$, where c_{i+1} is the $(i+1)$-th bit of the binary representation of c. This difference is bounded by using the $\mathcal{D}_{2k,k}$-MDDH assumption twice. More precisely, we introduce an intermediary game $\mathsf{H}_{i.0}$, where we choose $[\mathbf{t}_i]_1$ as

$$[\mathbf{t}_i]_1 = \begin{cases} [\mathbf{A}_0\mathbf{r}_i]_1 \text{ for } \mathbf{r}_i \stackrel{\$}{\leftarrow} \mathbb{Z}_p^k & \text{if } \mathsf{c}_{i+1} = 0 \\ [\mathbf{u}_i]_1 \text{ for } \mathbf{u}_i \stackrel{\$}{\leftarrow} \mathbb{Z}_p^{2k} & \text{else} \end{cases}.$$

Let $\mathcal{B}_{i.0}$ be an adversary receiving against the Q-fold MDDH assumption. On input $(\mathcal{G}, [\mathbf{A}_0]_1, [\mathbf{z}_1]_1, \dots, [\mathbf{z}_Q]_1)$, it sets up the game for $\mathcal{A}$ similar to game $\mathsf{G}_{3.i}$, except that it uses $[\mathbf{A}_0]_1$ from the challenge instead of sampling $[\mathbf{A}_0]_1$ by itself. Further, whenever obtaining a simulation query c with $\mathsf{c}_{|i+1} = 0$, $\mathcal{B}_{i.0}$ sets $[\mathbf{t}_i]_1 := [\mathbf{z}_i]_1$ and follows $\mathsf{G}_{3.i}$ otherwise. Similar, we can reduce the transition from game $\mathsf{H}_{i.0}$ to $\mathsf{H}_{i.1}$ to the $\mathcal{D}_{2k,k}$-MDDH assumption. According to Lemma 2, we have $T(\mathcal{B}_{i.0}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\varepsilon_{3.i} - \hat{\varepsilon}_{i.1}| \leq \frac{2}{p-1} + 2k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_{i.0}}(\lambda).$$

$\mathsf{H}_{i.1}$ to $\mathsf{H}_{i.2}$: We now reverse the transition from game G_0 to G_1 in Lemma 3. Specifically, we generate $\mathsf{crs}_{\mathsf{or}}$ by using $\mathsf{Gen}_{\mathsf{or}}$ instead of $\mathsf{TGen}_{\mathsf{or}}$ and generate proofs honestly by $\mathsf{Prove}_{\mathsf{or}}$. By the similar reasoning as for the transition from game G_0 to G_1, there exists an adversary $\mathcal{B}_{i.1}$ such that $T(\mathcal{B}_{i.1}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\hat{\varepsilon}_{i.1} - \hat{\varepsilon}_{i.2}| \leq \mathsf{Adv}^{\mathsf{zk}}_{\mathsf{NIZK},\mathcal{B}_{i.1}}(\lambda).$$

$\underline{\textsc{Init}_{\mathsf{core}}}$: // $\boxed{\mathsf{H}_{i.2} - \mathsf{H}_{i.7}}$, $\mathsf{H}_{i.4} - \mathsf{H}_{i.8}$

$\mathsf{c} := 0$
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$
$\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$
$(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}) \leftarrow \mathsf{TGen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}})$
$\boxed{\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}})}$
$\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$
$\mathbf{p} := \mathbf{A}_0^\top (\mathbf{k} + \mathbf{RF}_i(0^i))$
$\mathbf{p} := \mathbf{A}_0^\top (\mathbf{k} + \mathbf{RF}_{i+1}(0^{i+1}))$
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$
Return crs

$\underline{\textsc{Eval}_{\mathsf{core}}}$: // $\boxed{\mathsf{H}_{i.2} - \mathsf{H}_{i.7}}$, $\mathsf{H}_{i.4} - \mathsf{H}_{i.8}$

$\mathsf{c} := \mathsf{c} + 1$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$
$[\mathbf{t}]_1 := [\mathbf{A}_{\mathsf{c}_{i+1}}]_1 \mathbf{s}$
$\pi_{\mathsf{or}} \leftarrow (\mathsf{crs}_{\mathsf{or}}, \mathsf{td}, [\mathbf{t}]_1)$
$\boxed{\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})}$
$[u']_1 := [\mathbf{t}^\top (\mathbf{k} + \mathbf{RF}_i(\mathsf{c}_{|i}))]_1$
$[u']_1 := [\mathbf{t}^\top (\mathbf{k} + \mathbf{RF}_{i+1}(\mathsf{c}_{|i+1}))]_1$
$\mu := ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$
Return μ

$\underline{\textsc{Finalize}_{\mathsf{core}}(\mu)}$: // $\mathsf{H}_{i.3} - \mathsf{H}_{i.6}$, $\mathsf{H}_{i.4} - \mathsf{H}_{i.8}$

Parse $\mu = ([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$
$\beta := \mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}})$
$\mathcal{S} := \{\mathbf{RF}_i(\mathsf{c}'_{|i}) : \mathsf{c}' \le \mathsf{c}\}$
Game $\mathsf{H}_{i.4}$:
$\mathcal{S} := \{ \mathbf{RF}_{i+1}(\mathsf{c}'_{|i} | d_{\mathbf{t}}) : 0 \le \mathsf{c}' \le \mathsf{c}\}$
Game $\mathsf{H}_{i.5}$:
$\mathcal{S} := \{\mathbf{RF}_{i+1}(\mathsf{c}'_{|i} | b) : 0 \le \mathsf{c}' \le \mathsf{c}, b \in \{0,1\} \}$
Game $\mathsf{H}_{i.6} - \mathsf{H}_{i.8}$:
$\mathcal{S} := \{\mathbf{RF}_{i+1}(\mathsf{c}'_{|i+1}) : 0 \le \mathsf{c}' \le \mathsf{c}\}$
If $[\mathbf{t}]_1 \in \mathsf{Span}([\mathbf{A}_0]) \cup \mathsf{Span}([\mathbf{A}_1])$
  and $\beta = 1$ and $\exists \mathbf{w} \in \mathcal{S} : [u']_1 = [\mathbf{t}]_1^\top (\mathbf{k} + \mathbf{w})$
    Return 1
Else return 0

Fig. 14. Games $\mathsf{H}_{i.j}$ for $i \in \{0, \ldots, \lceil \log Q \rceil - 1\}$, $j \in \{1, \ldots, 8\}$, for the proof of Lemma 14. Here, $\mathbf{RF}_i : \{0,1\}^i \to \mathbb{Z}_p^{2k}$ denotes a random function, $\mathsf{c}_{|i}$ denotes the i-bit string that is a prefix of c written in binary, and c_i is the i'th bit of c written in binary. We have $d_{\mathbf{t}} = 0$ if $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_0)$, and $d_{\mathbf{t}} = 1$ if $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_1)$. In each procedure, the components inside a solid (dotted, gray) frame are only present in the games marked by a solid (dotted, gray) frame. For the intermediate transitions from game $\mathsf{H}_{i.4}$ to game $\mathsf{H}_{i.6}$ we use light gray highlighting to emphasize the respective differences.

$\mathsf{H}_{i.2}$ *to* $\mathsf{H}_{i.3}$*:* $\mathsf{H}_{i.3}$ is the same as $\mathsf{H}_{i.2}$ except that $\textsc{Finalize}_{\mathsf{core}}$ additionally checks whether $[\mathbf{t}]_1 \in \mathsf{Span}([\mathbf{A}_0]_1) \cup \mathsf{Span}([\mathbf{A}_1]_1)$. The perfect soundness of NIZK guarantees that

$$\hat{\varepsilon}_{i.2} = \hat{\varepsilon}_{i.3}.$$

$\mathsf{H}_{i.3}$ *to* $\mathsf{H}_{i.4}$*:* We first rewrite $\mathbf{RF}_i \colon \{0,1\}^i \to \mathbb{Z}_p^{2k}$ as

$$\mathbf{RF}_i(\nu) := \left(\mathbf{A}_0^\perp | \mathbf{A}_1^\perp\right) \begin{pmatrix} \Gamma_i(\nu) \\ \Upsilon_i(\nu) \end{pmatrix}, \tag{4}$$

where $\mathbf{A}_0^\perp \in \mathsf{ker}(\mathbf{A}_0) \subsetneq \mathbb{Z}_p^{2k \times k}$ and $\mathbf{A}_1^\perp \in \mathsf{ker}(\mathbf{A}_1) \subsetneq \mathbb{Z}_p^{2k \times k}$, $\nu \in \{0,1\}^i$ is an i-bit string, and $\Gamma_i, \Upsilon_i : \{0,1\}^i \to \mathbb{Z}_p^k$ are two independent random functions. We note that after rewriting $\mathbf{RF}_i$ it is still a random function if $(\mathbf{A}_0^\perp | \mathbf{A}_1^\perp)$ has full rank, which happens with probability $1 - \Delta_{\mathcal{D}_{2k,k}}$.

Next we define $\mathbf{RF}_{i+1} \colon \{0,1\}^{i+1} \to \mathbb{Z}_p^{2k}$ as

$$\mathbf{RF}_{i+1}(\nu) := \begin{cases} \mathbf{RF}_i(\nu_{|i}) + \mathbf{A}_0^\perp \Gamma_i'(\nu_{|i}) & \text{if } \nu_{i+1} = 0 \\ \mathbf{RF}_i(\nu_{|i}) + \mathbf{A}_1^\perp \Upsilon_i'(\nu_{|i}) & \text{else} \end{cases}, \tag{5}$$

where $\nu \in \{0,1\}^{i+1}$ and $\Gamma_i', \Upsilon_i' : \{0,1\}^i \to \mathbb{Z}_p^k$ are two new independent random functions. $\mathbf{RF}_{i+1}$ is a random function, since $\begin{pmatrix} \Gamma_i(\nu_{|i}) + \Gamma_i'(\nu_{|i}) \\ \Upsilon_i(\nu_{|i}) \end{pmatrix}$ and $\begin{pmatrix} \Gamma_i(\nu_{|i}) \\ \Upsilon_i(\nu_{|i}) + \Upsilon_i'(\nu_{|i}) \end{pmatrix}$ are two independent random vector and $(\mathbf{A}_0^\perp | \mathbf{A}_1^\perp)$ has full rank.

In $\mathsf{H}_{i.4}$, we simulate the security game with $\mathbf{RF}_{i+1}$ instead of $\mathbf{RF}_i$. We show that this move does not change the view of $\mathcal{A}$.

$\textsc{Init}_{\mathsf{core}}$ generates $\mathbf{p}$ by computing $\mathbf{p} := \mathbf{A}_0^\top(\mathbf{k} + \mathbf{RF}_{i+1}(0^{i+1}))$ instead of $\mathbf{p} := \mathbf{A}_0^\top(\mathbf{k} + \mathbf{RF}_i(0^i))$. This change does not change the distribution of crs since

$$\mathbf{p} = \mathbf{A}_0^\top \mathbf{RF}_{i+1}(0^{i+1}) = \mathbf{A}_0^\top(\mathbf{RF}_i(0^i) + \mathbf{A}_0^\perp \Gamma_i(0^i)) = \mathbf{A}_0^\top \mathbf{RF}_i(0^i).$$

For queries to $\textsc{Eval}_{\mathsf{core}}$, we consider

$$\begin{aligned} \mathbf{t}^\top \mathbf{RF}_{i+1}(\mathsf{c}_{|i+1}) &= \begin{cases} \mathbf{r}^\top \mathbf{A}_0^\top \mathbf{RF}_i(\mathsf{c}_{|i}) + \mathbf{r}^\top \mathbf{A}_0^\top \mathbf{A}_0^\perp \Gamma_i'(\mathsf{c}_{|i}) & \text{if } \mathsf{c}_{i+1} = 0 \\ \mathbf{r}^\top \mathbf{A}_1^\top \mathbf{RF}_i(\mathsf{c}_{|i}) + \mathbf{r}^\top \mathbf{A}_1^\top \mathbf{A}_1^\perp \Upsilon_i'(\mathsf{c}_{|i}) & \text{else} \end{cases} \\ &= \mathbf{r}^\top \mathbf{A}_{\mathsf{c}_{i+1}}^\top \mathbf{RF}_i(\mathsf{c}_{|i}) = \mathbf{t}^\top \mathbf{RF}_i(\mathsf{c}_{|i}). \end{aligned}$$

Thus, this change does not affect the output distribution of $\textsc{Eval}_{\mathsf{core}}$ queries.

In the oracle $\textsc{Finalize}_{\mathsf{core}}$ we have $[\mathbf{t}]_1 \in \mathsf{Span}([\mathbf{A}_0]) \cup \mathsf{Span}([\mathbf{A}_1])$ and $d_{\mathbf{t}} = 0$ if $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_0)$ and $d_{\mathbf{t}} = 1$ if $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_1)$. After replacing $\mathbf{RF}_i(\mathsf{c}_{|i})$ with $\mathbf{RF}_{i+1}(\mathsf{c}_{|i}|d_{\mathbf{t}})$, the output distribution of $\textsc{Eval}_{\mathsf{core}}$ does not change, since $\mathbf{t}^\top \mathbf{RF}_{i+1}(\mathsf{c}_{|i}|d_{\mathbf{t}}) = \mathbf{t}^\top \mathbf{RF}_i(\mathsf{c}_{|i})$.

To sum up, we obtain

$$|\hat{\varepsilon}_{i.3} - \hat{\varepsilon}_{i.4}| \leq \Delta_{\mathcal{D}_{2k,k}}.$$

$\mathsf{H}_{i.4}$ *to* $\mathsf{H}_{i.5}$*:* $\mathsf{H}_{i.5}$ is the same as $\mathsf{H}_{i.4}$ except that we enlarge the set $\mathcal{S}$ in $\textsc{Finalize}_{\mathsf{core}}$ to $\{\mathbf{RF}_{i+1}(\mathsf{c}'_{|i}|b) : \mathsf{c}' \leq \mathsf{c}, b \in \{0,1\}\}$. Thus, $\textsc{Finalize}_{\mathsf{core}}([\mathbf{t}]_1, \Pi, [u']_1)$ will output 1, even if $[u']_1 = (\mathbf{k}_0 + \mathbf{RF}_{i+1}(\mathsf{c}'_{|i}|1 - d_{\mathbf{t}}))^\top [\mathbf{t}]_1$ for some $\mathsf{c}' \leq \mathsf{c}$.

The transition from $\mathsf{H}_{i.4}$ to $\mathsf{H}_{i.5}$ can only increase the chance of the adversary, since $\mathcal{S}_{i.4} \subseteq \mathcal{S}_{i.5}$. We thus have

$$\hat{\varepsilon}_{i.4} \leq \hat{\varepsilon}_{i.5}.$$

$\mathsf{H}_{i.5}$ *to* $\mathsf{H}_{i.6}$*:* Let $\hat{\mathsf{c}} := \mathsf{c}'_{|i}|(1-\mathsf{c}'_{i+1})$ for some $0 \leq \mathsf{c}' \leq \mathsf{c}$. The difference between $\mathsf{H}_{i.5}$ and $\mathsf{H}_{i.6}$ lies in whether $\textsc{Finalize}_{\mathsf{core}}$ rejects a query with $[u]_1 = [\mathbf{t}]_1^\top(\mathbf{k}_0 + \mathbf{RF}_{i+1}(\hat{\mathsf{c}}))$ and $\mathbf{t} \in \mathsf{Span}(\mathbf{A}_0) \cup \mathsf{Span}(\mathbf{A}_1)$:

- If $\hat{\mathsf{c}} \leq \mathsf{c}$, then such a query will be accepted in both $\mathsf{H}_{i.5}$ and $\mathsf{H}_{i.6}$. In this case, both $\mathsf{H}_{i.5}$ and $\mathsf{H}_{i.6}$ are distributed the same.
- If $\hat{\mathsf{c}} > \mathsf{c}$, then such a query will be accepted in $\mathsf{H}_{i.5}$, but rejected in $\mathsf{H}_{i.6}$. From the adversary $\mathcal{A}$'s view, the information of the random vector $\mathbf{RF}_{i+1}(\hat{\mathsf{c}})$ is perfectly hidden. Thus, for a fixed $\hat{\mathsf{c}} > \mathsf{c}$, the adversary $\mathcal{A}$ can only output the value $[\mathbf{t}^\top \mathbf{RF}_{i+1}(\hat{\mathsf{c}})]_1 \in \mathbb{G}_1$ for some $\mathbf{t} \neq \mathbf{0}$ with probability at most $1/p$.

By union bound we can sum up:

$$|\hat{\varepsilon}_{i.5} - \hat{\varepsilon}_{i.6}| \leq \frac{Q}{p}.$$

$\mathsf{H}_{i.6}$ *to* $\mathsf{H}_{i.7}$*:* $\mathsf{H}_{i.7}$ is the same as $\mathsf{H}_{i.6}$ except that $\textsc{Finalize}_{\mathsf{core}}$ does not perform the additional check $[\mathbf{t}]_1 \in \mathsf{Span}([\mathbf{A}_0]_1) \cup \mathsf{Span}([\mathbf{A}_1]_1)$ anymore. Due to the soundness of NIZK, this change will not affect the adversary's view at all. Moreover, in $\mathsf{H}_{i.7}$, we just use a truly random function to simulate instead of using Equations (4) and (5) and the kernel matrices of $\mathbf{A}_0$ and $\mathbf{A}_1$. Similar to "$\mathsf{H}_{i.3}$ to $\mathsf{H}_{i.4}$", this change will not affect the adversary's view as long as $(\mathbf{A}_0^\perp|\mathbf{A}_1^\perp)$ is full-rank, we have

$$|\hat{\varepsilon}_{i.6} - \hat{\varepsilon}_{i.7}| \leq \Delta_{\mathcal{D}_{2k,k}}.$$

$\mathsf{H}_{i.7}$ *to* $\mathsf{H}_{i.8}$*:* This transition is similar to the transition from G_0 to G_1 in Lemma 3. We can construct an adversary $\mathcal{B}_{i.7}$ such that $T(\mathcal{B}_{i.8}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$|\hat{\varepsilon}_{i.7} - \hat{\varepsilon}_{i.8}| \leq \mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{B}_{i.7}}(\lambda).$$

$\mathsf{H}_{i.8}$ *to* $\mathsf{H}_{3.(i+1)}$*:* We choose $[\mathbf{t}]_1$ in $\textsc{Eval}_{\mathsf{core}}$ uniformly at random from $\mathbb{G}_1^{2k}$. Similar to the transition from $\mathsf{G}_{3.i}$ *to* $\mathsf{H}_{i.1}$, this difference can be bounded by using the $\mathcal{D}_{2k,k}$-MDDH assumption twice, one with $[\mathbf{A}_0]_1$ as the challenge matrix and the other with $[\mathbf{A}_1]_1$. We obtain an adversary $\mathcal{B}_{i.8}$ with $T(\mathcal{B}_{i.8}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ such that

$$|\hat{\varepsilon}_{i.8} - \varepsilon_{3.(i+1)}| \leq \frac{2}{p-1} + 2k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}_{i.8}}(\lambda),$$

completing the proof. □

C Tightly IND-mCCA-Secure PKE

In this section, we recall the definition of labeled IND-mCCA secure PKE and the generic construction in [5]. By instantiating the underlying USS-QA-NIZK with our construction in Section 3.2, we immediately obtain a more efficient publicly verifiable labeled tightly IND-mCCA secure PKE.

Definition 12 (Labeled public-key encryption). *A public-key encryption (*PKE*) scheme consists of probabilistic polynomial-time algorithms* $\Pi_{\mathsf{PKE}} := (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$*:*

– $\mathsf{Gen}(\mathsf{par})$ *generates a pair of public and secret keys* $(\mathsf{pk}, \mathsf{sk})$. *Message space* $\mathcal{M}$ *is determined by* pk.
– $\mathsf{Enc}(\mathsf{pk}, \mathsf{m}, \ell)$ *returns a ciphertext* ct.
– $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}, \ell)$ *is deterministic and returns a message* m.

For correctness, it must hold that, for all $(\mathsf{pk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})$, *messages* $\mathsf{m} \in \mathcal{M}$, *and* $\mathsf{ct} \in \mathsf{Enc}(\mathsf{pk}, \mathsf{m}, \ell)$, $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}, \ell) = \mathsf{m}$.

Definition 13 (IND-mCCA security). *For a* PKE $\Pi_{\mathsf{PKE}} := (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *and any adversary* $\mathcal{A}$, *we define the following experiment:*

<u>Init</u>:	<u>EncO$(i, \mathsf{m}_0, \mathsf{m}_1, \ell)$</u>:	<u>DecO(i, ct, ℓ)</u>:	<u>Finalize(b^*)</u>:
$(\mathsf{pk}^i, \mathsf{sk}^i) \leftarrow \mathsf{Gen}(\mathsf{par})$ for $i = 1, \cdots, n$ $b \xleftarrow{\$} \{0,1\}$ Return pk.	$\mathsf{ct}^* := \mathsf{Enc}(\mathsf{pk}^i, \mathsf{m}_b, \ell)$ $\mathcal{Q}_{\mathsf{enc}} := \mathcal{Q}_{\mathsf{enc}} \cup \{i, \mathsf{ct}^*, \ell\}$ Return ct^*	If $(i, \mathsf{ct}, \ell) \in \mathcal{Q}_{\mathsf{enc}}$ then return $\perp$ $\mathsf{m} \leftarrow \mathsf{Dec}(\mathsf{sk}^i, \mathsf{ct}, \ell)$ Return m	If $b^* = b$ return 1 Else return 0

Fig. 15. IND-mCCA security game for Π.

A PKE Π_{PKE} *is indistinguishable against chosen ciphertext attacks* (IND-mCCA), *if for all PPT adversaries* $\mathcal{A}$,

$$\mathsf{Adv}_{\Pi,\mathcal{A}}^{\mathsf{IND\text{-}mCCA}}(\lambda) := |2\Pr[\mathsf{IND\text{-}mCCA}^{\mathcal{A}} \Rightarrow 1] - 1|$$

is negligible, where Game IND-mCCA *is defined in Figure 15.*

<u>$\mathsf{Gen}(\mathsf{par})$</u>:	<u>$\mathsf{Enc}(\mathsf{pk}, [m]_1 \in \mathbb{G}_1, \ell)$</u>:
$\mathbf{B} \xleftarrow{\$} \mathcal{D}_{2k,k}$, $\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^k$ $\mathsf{crs} \xleftarrow{\$} \mathsf{Gen}(\mathsf{par}, [\mathbf{B}]_1)$ $\mathbf{p} := \overline{\mathbf{B}}^\top \mathbf{k}$ $pk := (\mathsf{crs}, [\mathbf{B}]_1, [\mathbf{p}]_1)$ $\mathsf{sk} := \mathbf{k}$ Return $(\mathsf{pk}, \mathsf{sk})$.	$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{y}]_1 := [\overline{\mathbf{B}\mathbf{r}}]_1^\top$ $[\mathbf{c}]_1 := [m]_1 + \mathbf{r}^\top [\mathbf{p}]_1$ $\pi := \mathsf{Prove}(\mathsf{crs}, [\mathbf{y}]_1, \mathbf{r}, ([\mathbf{c}]_1, \ell))$ Return $\mathtt{ctxt} := ([\mathbf{y}]_1, [\mathbf{c}]_1, \pi)$ <u>$\mathsf{Dec}(\mathsf{sk}, [m]_1, \mathtt{ctxt}, \ell)$</u>: Parse $\mathtt{ctxt} =: ([\mathbf{y}]_1, [\mathbf{c}]_1, \pi)$ If $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}]_1, ([\mathbf{c}]_1, \ell), \pi) = 1$ return $[\mathbf{c}]_1 - [\overline{\mathbf{y}}]_1 \mathbf{k}$ Else abort

Fig. 16. IND-mCCA secure PKE using labeled QANIZK $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$.

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, $k \in \mathbb{N}$, and $\Pi = (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ be our labeled QANIZK in Figure 7. We adopt the generic construction in [5,6] to propose a IND-mCCA secure PKE in Figure 16, which is more efficient than the original instantiation in [5,6].

Theorem 4. *If the* $\mathcal{D}_{2k,k}$-MDDH *assumption holds, and* Π *is a labeled* QANIZK *with perfect completeness, perfect zero knowledge, and unbounded simulation soundness,* $\Pi_{\mathsf{PKE}} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is a* PKE *with tight* IND-mCCA *security. In particular, for any adversary* $\mathcal{A}$*, there exist adversaries* $\mathcal{B}$ *and* $\mathcal{B}'$ *with* $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{B}') \approx \mathsf{T}(\mathcal{A})$ *and*

$$\mathsf{Adv}^{\mathsf{IND\text{-}mCCA}}_{\Pi,\mathcal{A}}(\lambda) \leq 4 \cdot \mathsf{Adv}^{\mathsf{uss}}_{\Pi,\mathcal{B}}(\lambda) + 6k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}'}(\lambda) + O(\frac{1}{p}).$$

We refer the reader to [6] for the proof of the above theorem.

D Proof of Lemma 12

We now show the proofs of completeness, zero-knowledge, and soundness of Π^{or}.

Completeness. Let $j \in \{0,1\}$ such that $[\mathbf{y}]_1 = [\mathbf{A}_j]_1\mathbf{r}$. For $(x_1, \cdots, x_k)$ (computed as $\overline{\mathbf{y}}^\top \mathbf{d}^\top - \underline{\mathbf{y}}^\top$), we have $[\mathbf{A}_j]_1(\mathbf{x}_1, \cdots, \mathbf{x}_k) = [\mathbf{A}_j]_1\mathbf{r}(x_1, \cdots, x_k) = [\mathbf{y}]_1(x_1, \cdots, x_k)$. Let $([\mathbf{C}_i]_2, [\mathbf{c}_i]_2, [\mathbf{\Pi}_i]_1, [\pi_i]_1)_{1\leq i\leq k}$ be returned by $\mathsf{Prove}_{\mathsf{or}}$ on input $\mathsf{crs}_{\mathsf{or}}$, $[\mathbf{y}]_1$ and $\mathbf{r}$. For $i = 1, \cdots, k$, we have

$$\begin{aligned}
&[\mathbf{A}_0]_1 \circ [\mathbf{C}_i]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}_i]_2\\
=&[\mathbf{A}_0]_1 \circ [\mathbf{x}_i\mathbf{u}^\top + \mathbf{R}_i\mathbf{V}^\top]_2 - [\mathbf{y}]_1 \circ [x_i\mathbf{u}^\top + \mathbf{r}_i\mathbf{V}^\top]_2\\
=&[\mathbf{A}_0\mathbf{x}_i - \mathbf{y}x_i]_1 \circ [\mathbf{u}^\top]_2 + [\mathbf{A}_0]_1 \circ [\mathbf{R}_i\mathbf{V}^\top]_2 - [\mathbf{y}]_1 \circ [\mathbf{r}_i\mathbf{V}^\top]_2\\
=&[\mathbf{A}_0]_1 \circ [\mathbf{R}_i\mathbf{V}^\top]_2 - [\mathbf{y}]_1 \circ [\mathbf{r}_i\mathbf{V}^\top]_2\\
=&[\mathbf{A}_0\mathbf{R}_i - \mathbf{y}\mathbf{r}_i]_1 \circ [\mathbf{V}^\top]_2 = [\mathbf{\Pi}_i]_1 \circ [\mathbf{V}^\top]_2
\end{aligned}$$

and

$$\begin{aligned}
&[\overline{\mathbf{y}}^\top]_1 \circ [\mathbf{D}_i]_2 - [y_i]_1 \circ [\mathbf{u}^\top]_2 - [1]_1 \circ [\mathbf{c}_i]_2\\
=&[\overline{\mathbf{y}}^\top]_1 \circ [\mathbf{d}_i^\top\mathbf{u}^\top + \mathbf{S}_i\mathbf{V}^\top]_2 - [y_i]_1 \circ [\mathbf{u}^\top]_2 - [1]_1 \circ [x_i\mathbf{u}^\top + \mathbf{r}_i\mathbf{V}^\top]_2\\
=&[\overline{\mathbf{y}}^\top\mathbf{d}_i^\top - y_i - x_i]_1 \circ [\mathbf{u}^\top]_2 + [\overline{\mathbf{y}}^\top\mathbf{S}_i]_1 \circ [\mathbf{V}^\top]_2 - [\mathbf{r}_i]_1 \circ [\mathbf{V}^\top]_2\\
=&[\overline{\mathbf{y}}^\top\mathbf{S}_i - \mathbf{r}_i]_1 \circ [\mathbf{V}^\top]_2,
\end{aligned}$$

which prove the completeness.

Zero-knowledge. Let $\mathcal{A}$ be a PPT adversary attacking the zero-knowledge property. We build a PPT adversary $\mathcal{B}$, which is defined in Figure 17, such that $T(\mathcal{B}) \approx T(\mathcal{A})$ and

$$\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) + \frac{1}{p}.$$

Upon receiving its MDDH challenge $(\mathcal{G}, [\mathbf{V}]_2, [\mathbf{u}]_2)$, $\mathcal{B}$ computes $([\mathbf{A}_0]_1, [\mathbf{A}_1]_1, ([\mathbf{D}_i]_2)_{1\leq i\leq k})$ and $\mathsf{sk}_{\mathsf{or}}$ in the same way as in $\mathsf{Gen}_{\mathsf{or}}$ and forwards $\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{u}]_2, [\mathbf{V}]_2, ([\mathbf{D}_i]_2)_{1\leq i\leq k})$ to $\mathcal{A}$. Every time when receiving a query

$\underline{\text{Init}}$: $\boxed{\mathsf{MDDH\ challenge}: (\mathcal{G}, [\mathbf{V}]_2, [\mathbf{u}]_2)}$ $\mathsf{par} := \mathcal{G}\ \mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$ $\begin{pmatrix} \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_k \end{pmatrix} := \underline{\mathbf{A}_1}\overline{\mathbf{A}_1}^{-1} \in \mathbb{Z}_p^{k \times k}$ For $i = 1, \cdots, k$: $\quad \mathbf{S}_i \xleftarrow{\$} \mathbb{Z}_p^{k \times k}$ $\quad [\mathbf{D}_i]_2 := \mathbf{d}_i^\top [\mathbf{u}^\top]_2 + \mathbf{S}_i [\mathbf{V}^\top]_2$ $\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{u}]_2, [\mathbf{V}]_2,$ $([\mathbf{D}_i]_2)_{1 \le i \le k})$ $\mathsf{sk}_{\mathsf{or}} := (\mathbf{A}_0, \mathbf{A}_1, (\mathbf{S}_i)_{1 \le i \le k})$ Return $\mathsf{crs}_{\mathsf{or}}$	$\underline{\text{Prv}([\mathbf{y}]_1, \mathbf{r})}$: $\pi \leftarrow \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{r})$ Return π $\underline{\text{Finalize}(b)}$: Return b.

Fig. 17. Reduction $\mathcal{B}$ from the zero knowledge property of Π^{or} to $\mathcal{D}_k$-MDDH assumption.

$([\mathbf{y}]_1, \mathbf{r})$ from $\mathcal{A}$, $\mathcal{B}$ sends $\pi \leftarrow \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{r})$ back. When $\mathcal{A}$ outputs a bit b, $\mathcal{B}$ outputs b.

When $\mathcal{B}$ receives $[\mathbf{u}]_2 \xleftarrow{\$} \mathbb{G}_2^{k+1}$, due to fact that the uniformly random distribution over $\mathbb{Z}_p^{k+1}$ and the uniformly random distribution over $\mathbb{Z}_p^{k+1} \setminus \mathsf{Span}(\mathbf{V})$ are $1/p$-statistically close, it simulates the oracle $\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, \cdot, \cdot)$ in the zero knowledge game (see Definition 5) within a $1/p$ statistical distance. Now we argue that when $\mathcal{B}$ receives a real MDDH tuple, that is, when there exists $\mathbf{z} \in \mathbb{Z}_p^k$ such that $[\mathbf{u}]_2 := [\mathbf{V}\mathbf{z}]_2$, it simulates the simulation oracle in the zero knowledge game perfectly.

Since $[\mathbf{u}]_2 = [\mathbf{V}\mathbf{z}]_2$, for $i = 1, \cdots, k$, we can re-write $[\mathbf{D}_i]_2$, $[\mathbf{C}_i]_2$, and $[\mathbf{c}_i]_2$ as $[\mathbf{D}_i]_2 = (\mathbf{d}_i^\top \mathbf{z}^\top + \mathbf{S}_i)[\mathbf{V}^\top]_2$, $[\mathbf{C}_i]_2 = (\mathbf{x}_i \mathbf{z}^\top + \mathbf{R}_i)[\mathbf{V}^\top]_2$, and $[\mathbf{c}_i]_2 = (x_i \mathbf{z}^\top + \mathbf{r}_i)[\mathbf{V}^\top]_2$ respectively. Furthermore, for $i = 1, \cdots, k$, we can re-write $\mathbf{\Pi}_i$ and π_i as $\mathbf{A}_0(\mathbf{x}_i \mathbf{z}^\top + \mathbf{R}_i) - \mathbf{y}(x_i \mathbf{z}^\top + \mathbf{r}_i)$ (since $\mathbf{A}_0 \mathbf{x}_i - \mathbf{y} x_i = \mathbf{0}$) and $\overline{\mathbf{y}}^\top (\mathbf{d}_i^\top \mathbf{z}^\top + \mathbf{S}_i) - (x_i \mathbf{z}^\top + \mathbf{r}_i) - y_i \mathbf{z}^\top$ (since $\overline{\mathbf{y}}^\top \mathbf{d}_i^\top - x_i - y_i = \mathbf{0}$ where y_i denotes the ith element in $\mathbf{y}$) respectively. One can see that the distributions of $\mathbf{x}_i \mathbf{z}^\top + \mathbf{R}_i$, $x_i \mathbf{z}^\top + \mathbf{r}_i$, and $\mathbf{d}_i^\top \mathbf{z}^\top + \mathbf{S}_i$ are respectively identical to $\mathbf{R}_i$, $\mathbf{r}_i$, $\mathbf{S}_i$, where $\mathbf{S}_i, \mathbf{R}_i \xleftarrow{\$} \mathbb{Z}_p^{k \times k}$, and $\mathbf{r}_i \xleftarrow{\$} \mathbb{Z}_p^{1 \times k}$, for all i. Thus, $\mathcal{A}'s$ view in this case is identical to its view in the zero knowledge game where the oracle runs the simulator, which proves that $\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}}, \mathcal{A}}(\lambda) \le \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G}, \mathbb{G}_2, \mathcal{D}_k, \mathcal{B}}(\lambda) + \frac{1}{p}$.

Perfect soundness. Let $\mathbf{v} \in \mathbb{Z}_p^{k+1}$ be a vector satisfying $\mathbf{V}^\top \mathbf{v} = \mathbf{0}$ and $\mathbf{u}^\top \mathbf{v} = 1$, which must exists since $\mathbf{u} \notin \mathsf{Span}(\mathbf{V})$. Let $([\mathbf{C}_i]_2, [\mathbf{c}_i]_2, [\mathbf{\Pi}_i]_1, [\pi_i]_1)_{1 \le i \le k}$, be a valid proof. It must satisfy

$$[\mathbf{A}_0]_1 \circ [\mathbf{C}_i]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}_i]_2 = [\mathbf{\Pi}_i]_1 \circ [\mathbf{V}^\top]_2$$

and

$$[\overline{\mathbf{y}}^\top]_1 \circ [\mathbf{D}_i]_2 - [y_i]_1 \circ [\mathbf{u}^\top]_2 - [1]_1 \circ [\mathbf{c}_i]_2 = [\pi_i]_1 \circ [\mathbf{V}^\top]_2,$$

$\underline{\mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, \mathbf{A}_0 \in \mathbb{Z}_p^{2k\times k}, \mathbf{A}_1 \in \mathbb{Z}_p^{2k\times k})}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{S}_{2k,k}$, $\mathbf{K}, \mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times k}$, $\mathbf{T} \xleftarrow{\$} \mathcal{S}_{\mathbf{V}}$
$\mathbf{D} := \mathbf{K}\mathbf{V}^\top + \mathbf{A}_1^\perp \mathbf{T}^\top$
$\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{V}]_2, [\mathbf{D}]_2)$,
$\mathsf{sk}_{\mathsf{or}} := (\mathbf{A}_0, \mathbf{A}_1, \mathbf{K})$
Return $(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}})$

$\underline{\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{w})}$:
Parse $\mathsf{sk}_{\mathsf{or}} =: (\mathbf{A}_0, \mathbf{A}_1, \mathbf{K})$
If $\neg(\exists j \in \{0,1\} : [\mathbf{y}]_1 = [\mathbf{A}_j \mathbf{w}]_1)$
  then abort
Else
  $\Delta := \mathbf{y} - \mathbf{A}_1 \mathbf{w}$
  $\mathbf{R} \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$
  $[\mathbf{C}]_2 := \mathbf{R}[\mathbf{V}^\top]_2 + \mathbf{w}\Delta^\top[\mathbf{D}]_2$
  $\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^{1\times k}$
  $[\mathbf{c}]_2 := \mathbf{r}[\mathbf{V}^\top]_2 + \Delta^\top[\mathbf{D}]_2$
  $\mathbf{\Pi} := \mathbf{A}_0\mathbf{R} - \mathbf{y}\mathbf{r}$
  $\pi := (\mathbf{A}_1\mathbf{w})^\top \mathbf{K} - \mathbf{r}$
Return $([\mathbf{C}]_2, [\mathbf{c}]_2, [\mathbf{\Pi}]_1, [\pi]_1)$

$\underline{\mathsf{TGen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_{2k,k}$, $\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{2k\times k}$, $\mathbf{D} := \mathbf{K}\mathbf{V}^\top$
$\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{V}]_2, [\mathbf{D}]_2)$,
$\mathsf{td}_{\mathsf{or}} := \mathbf{K}$
Return $(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}})$

$\underline{\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{y}]_1, \pi)}$:
Parse $\pi =: ([\mathbf{C}]_2, [\mathbf{c}]_2, [\mathbf{\Pi}]_1, [\pi]_1)$.
If $[\mathbf{A}_0]_1 \circ [\mathbf{C}]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}]_2 \neq [\mathbf{\Pi}]_1 \circ [\mathbf{V}^\top]_2$
  then return 0
If $[\mathbf{y}^\top]_1 \circ [\mathbf{D}]_2 - [1]_1 \circ [\mathbf{c}]_2 \neq [\pi]_1 \circ [\mathbf{V}^\top]_2$
  then return 0
Else return 1

$\underline{\mathsf{Sim}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}}, [\mathbf{y}]_1)}$:
Parse $\mathsf{td}_{\mathsf{or}} =: \mathbf{K}$
$\mathbf{R} \xleftarrow{\$} \mathbb{Z}_p^{k\times k}$, $[\mathbf{C}]_2 := \mathbf{R}[\mathbf{V}^\top]_2$
$\mathbf{r} \xleftarrow{\$} \mathbb{Z}_p^{1\times k}$, $[\mathbf{c}]_2 := \mathbf{r}[\mathbf{V}^\top]_2$
$[\mathbf{\Pi}]_1 := [\mathbf{A}_0]_1\mathbf{R} - [\mathbf{y}]_1\mathbf{r}$
$[\pi]_1 := [\mathbf{y}^\top]_1\mathbf{K} - [\mathbf{r}]_1$
Return $([\mathbf{C}]_2, [\mathbf{c}]_2, [\mathbf{\Pi}]_1, [\pi]_1)$

Fig. 18. Designated-Prover or-proof for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$.

for $i = 1, \cdots, k$. Multiplying above verification equation by $\mathbf{v}$ we have

$$[\mathbf{A}_0]_1(\mathbf{C}_1\mathbf{v}, \cdots, \mathbf{C}_k\mathbf{v}) = [\mathbf{y}]_1(\mathbf{c}_1\mathbf{v}, \cdots, \mathbf{c}_k\mathbf{v}) \tag{6}$$

and

$$[\overline{\mathbf{y}}^\top]_1(\mathbf{d}_1^\top, \cdots, \mathbf{d}_k^\top) - [\underline{\mathbf{y}}^\top]_1 = [\mathbf{c}_1\mathbf{v}, \cdots, \mathbf{c}_k\mathbf{v}]_1. \tag{7}$$

When there exists some i such that $\mathbf{c}_i\mathbf{v} \neq \mathbf{0}$, we have $[\mathbf{A}_0]_1(\frac{\mathbf{C}_i\mathbf{v}}{\mathbf{c}_i\mathbf{v}}) = \mathbf{y}$, i.e., $\mathbf{y} \in \mathsf{Span}(\mathbf{A}_0)$, due to Equation 6. Otherwise, we have $[\overline{\mathbf{y}}^\top]_1(\underline{\mathbf{A}}_1\overline{\mathbf{A}}_1^{-1})^\top - [\underline{\mathbf{y}}^\top]_1 = \mathbf{0}$, i.e., $\mathbf{y} \in \mathsf{Span}(\mathbf{A}_1)$, due to Equation 7. Therefore, we must have $\mathbf{y} \in \mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ when there exists a valid proof, completing the proof of perfect soundness.

E Extensions of Designated-Prover OR-Proof

Designated-Prover OR-Proof with Improved Proof Size. Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, and $k \in \mathbb{N}$. Let $\mathcal{S}_{2k,k}$ be the set of all the elements $\mathbf{V}$ in $\mathcal{D}_{2k,k}$ such that $\underline{\mathbf{W}} = -\underline{\mathbf{V}}^{-\top}\overline{\mathbf{V}}^\top$ is well-defined, and $\mathcal{S}_{\mathbf{V}}$ be the set of all the elements $\mathbf{T}$ in $\mathbb{Z}_p^{k\times 2k}$ such that $\mathbf{TW}$ is invertible. We present the designated-prover OR-proof in Figure 18.

Lemma 15. *If the* $\mathcal{D}_k$-MDDH *assumption holds in the group* $\mathbb{G}_2$*, then the proof system* $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ *as defined in Figure 18 is a*

<u>INIT</u>: $\boxed{\text{MDDH challenge: } (\mathcal{G}, [\mathbf{V}]_2, [\mathbf{T}]_2)}$ $\mathsf{par} := \mathcal{G}$ $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$ $\mathbf{K}, \mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k \times k}$ $\mathbf{D} := \mathbf{K}\mathbf{V}^\top + \mathbf{A}_1^\perp \mathbf{T}^\top$ $\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{V}]_2, [\mathbf{D}]_2)$, $\mathsf{sk}_{\mathsf{or}} := (\mathbf{A}_0, \mathbf{A}_1, \mathbf{K})$ Return $\mathsf{crs}_{\mathsf{or}}$	<u>PRV($[\mathbf{y}]_1, \mathbf{r}$)</u>: $\pi \leftarrow \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{r})$ Return π <u>FINALIZE(b)</u>: Return b. Else return 1

Fig. 19. Reduction $\mathcal{B}$ from the zero knowledge property of Π^{or} to $\mathcal{D}_k$-MDDH assumption.

designated-prover or-proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, perfect soundness, and zero-knowledge. More precisely, for all adversaries $\mathcal{A}$ attacking the zero-knowledge of Π^{or}, we obtain an adversary $\mathcal{B}$ with $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ and

$$\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) + \frac{k}{p} + \Delta'_{2k,k},$$

where $\Delta'_{2k,k}$ is the probability that $\mathbf{V} \leftarrow \mathcal{D}_{2k,k}$ has full rank.

Proof. We now show the proofs of completeness, zero-knowledge, and soundness of Π^{or}.

Completeness. Completeness follows easily from the fact that

$$\begin{aligned}[\mathbf{A}_0]_1 \circ [\mathbf{C}]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}]_2 &= [\mathbf{A}_0]_1 \circ [\mathbf{R}\mathbf{V}^\top + \mathbf{w}\Delta^\top \mathbf{D}]_2 - [\mathbf{y}]_1 \circ [\mathbf{r}\mathbf{V}^\top + \Delta^\top \mathbf{D}]_2 \\ &= [\mathbf{A}_0\mathbf{R} - \mathbf{y}\mathbf{r}]_1 \circ [\mathbf{V}^\top]_2 = [\mathbf{\Pi}]_1 \circ [\mathbf{V}^\top]_2\end{aligned}$$

and

$$\begin{aligned}[\mathbf{y}^\top]_1 \circ [\mathbf{D}]_2 - [1]_1 \circ [\mathbf{c}]_2 &= [\mathbf{y}^\top]_1 \circ [\mathbf{D}]_2 - [1]_1 \circ [\mathbf{r}\mathbf{V}^\top + \Delta^\top \mathbf{D}]_2 \\ = [(\mathbf{A}_1\mathbf{w})^\top]_1 \circ [\mathbf{K}\mathbf{V}^\top + \mathbf{A}_1^\perp \mathbf{T}]_2 &- [\mathbf{r}]_1 \circ [\mathbf{V}^\top]_2 = [\pi]_1 \circ [\mathbf{V}^\top]_2.\end{aligned}$$

Computational zero-knowledge. Let $\mathcal{A}$ be a PPT adversary attacking the zero-knowledge property. We build a PPT adversary $\mathcal{B}$, which is defined in Figure 19, such that $T(\mathcal{B}) \approx T(\mathcal{A})$ and

$$\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) + \frac{k}{p} + \Delta'_{2k,k}.$$

Upon receiving its k-fold $\mathcal{D}_{2k,k}$-MDDH challenge[11] $(\mathcal{G}, [\mathbf{V}]_2, [\mathbf{T}]_2)$, $\mathcal{B}$ computes $(\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{D}]_2)$ and $\mathsf{sk}_{\mathsf{or}}$ in the same way as in $\mathsf{Gen}_{\mathsf{or}}$ and forwards

[11] It is known that the $\mathcal{D}_{2k,k}$-MDDH hardness assumption reduces to $\mathcal{D}_k$-MDDH with a security loss of factor k[35].

$\mathsf{crs}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1, [\mathbf{V}]_2, [\mathbf{D}]_2)$ to $\mathcal{A}$. Every time when receiving a query $([\mathbf{y}]_1, \mathbf{r})$ from $\mathcal{A}$, $\mathcal{B}$ sends $\pi \leftarrow \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{y}]_1, \mathbf{r})$ back. When $\mathcal{A}$ outputs a bit b, $\mathcal{B}$ outputs b.

When $\mathcal{B}$ receives $[\mathbf{T}]_2 \xleftarrow{\$} \mathbb{G}_2^{k \times 2k}$, we consider the $(2k \times k)$-matrix $\mathbf{W}$ such that $\overline{\mathbf{W}} = I^{k \times k}$ and $\underline{\mathbf{W}} = -\underline{\mathbf{V}}^{-\top}\overline{\mathbf{V}}^{\top}$. With probability $\Delta'_{2k,k}$(which is over the choice of $\mathbf{V}$), $\mathbf{W}$ is well-defined. Note that $\mathbf{V}^{\top}\mathbf{W} = 0$ and $\mathbf{D}\mathbf{W} = \mathbf{A}_1^{\perp}\mathbf{T}\mathbf{W}$. The claim that $\mathbf{T}\mathbf{W}$ is invertible with probability at least $1 - k/p$. The reason is that the top part of $\mathbf{W}$ is just the identity matrix then $\mathbf{T}\mathbf{W}$ is just a random and independent square matrix, which has determinant zero with probability at most k/p due to the Schwartz-Zippel lemma (the determinant being a multi-variate degree k polynomial in the entries of its matrix).

Now we argue that when $\mathcal{B}$ receives a real MDDH tuple, that is, when there exists $\mathbf{k}' \in \mathbb{Z}_p^k$ such that $[\mathbf{T}]_2 := [\mathbf{k}'\mathbf{V}^{\top}]_2$, it simulates the simulation oracle in the zero knowledge game perfectly.

Note that in the proving procedure $(\mathbf{A}_0\mathbf{w} - \mathbf{y})\Delta^{\top} = 0$ and $\mathbf{y} - \Delta = \mathbf{A}_1\mathbf{w}$. Hence, we have

$$\mathbf{\Pi} = \mathbf{A}_0\mathbf{R} - \mathbf{y}\mathbf{r} = \mathbf{A}_0(\mathbf{R} + \mathbf{w}\Delta^{\top}\mathbf{K}) - \mathbf{y}(\mathbf{r} + \Delta^{\top}\mathbf{K})$$

and

$$\pi = (\mathbf{A}_1\mathbf{w})^{\top}\mathbf{K} - \mathbf{r} = \mathbf{y}^{\top}\mathbf{K} - (\mathbf{r} + \Delta^{\top}\mathbf{K}).$$

Moreover, the terms $(\mathbf{R} + \mathbf{w}\Delta^{\top}\mathbf{K})$, $(\mathbf{r} + \Delta^{\top}\mathbf{K})$, and $(\mathbf{K} + \mathbf{A}_1^{\perp}\mathbf{K}')$ are uniformly distributed. Thus, $\mathcal{A}'s$ view in this case is identical to its view in the zero knowledge game where the oracle runs the simulator, which proves that $\mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}},\mathcal{A}}(\lambda) \leq k \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) + \frac{k}{p} + \Delta'_{2k,k}$.

Soundness. Let $([\mathbf{C}]_2, [\mathbf{c}]_2, [\mathbf{\Pi}]_1, [\pi]_1)$ be a valid proof. It must satisfy

$$[\mathbf{A}_0]_1 \circ [\mathbf{C}]_2 - [\mathbf{y}]_1 \circ [\mathbf{c}]_2 = [\mathbf{\Pi}]_1 \circ [\mathbf{V}^{\top}]_2$$

and

$$[\mathbf{y}^{\top}]_1 \circ [\mathbf{D}]_2 - [1]_1 \circ [\mathbf{c}]_2 = [\pi]_1 \circ [\mathbf{V}^{\top}]_2.$$

Consider the $(2k \times k)$-matrix $\mathbf{W}$ such that $\overline{\mathbf{W}} = I^{k \times k}$ and $\underline{\mathbf{W}} = -\underline{\mathbf{V}}^{-\top}\overline{\mathbf{V}}^{\top}$. With high probability (over the choice of $\mathbf{V}$) the matrix $\mathbf{W}$ is well-defined. Note that $\mathbf{V}^{\top}\mathbf{W} = 0$, $\mathbf{D}\mathbf{W} = \mathbf{A}_1^{\perp}\mathbf{T}\mathbf{W}$, and $\mathbf{T}\mathbf{W}$ is invertible. Now, multiplying above verification equations by $\mathbf{W}$ on the right, we have

$$\mathbf{A}_0(\mathbf{C}\mathbf{W}) = \mathbf{y}(\mathbf{c}\mathbf{W})$$

and

$$\mathbf{y}^{\top}(\mathbf{A}_1^{\perp}\mathbf{T}\mathbf{W}) = \mathbf{c}\mathbf{W}.$$

Since the probability that $\mathbf{T}\mathbf{W}$ is invertible is at least $1 - \frac{k}{p}$ as discussed in the proof of computational zero-knowledge, if $\mathbf{c}\mathbf{W}$ is zero, $\mathbf{y}$ is in the span of $\mathbf{A}_1$ with probability at least $1 - \frac{k}{p}$. Otherwise, some element of the $(1 \times k)$-matrix $\mathbf{c}\mathbf{W}$ is non-zero. W.l.o.g. let this be the first element, i.e. $(\mathbf{c}\mathbf{W})_1$ is non-zero. Then

$\underline{\mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0] \in \mathbb{G}^{n\times t}, [\mathbf{A}_1] \in \mathbb{G}^{n\times t})}$:
We assume $l \geq n - t + k$.
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_{l,k}$, $\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{n\times k}$, $\mathbf{D} := \mathbf{K}\mathbf{V}^\top$,
$\mathbf{E} = (\mathbf{A}_0 - \mathbf{A}_1)^\top \mathbf{K}\mathbf{V}^\top$, $\mathbf{F} = \mathbf{A}_1^\top \mathbf{K}$
$\mathsf{crs}_{\mathsf{or}} := \begin{pmatrix} \mathsf{par}, [\mathbf{A}_0], [\mathbf{A}_1], [\mathbf{V}], [\mathbf{D}], \\ [\mathbf{E}], [\mathbf{F}] \end{pmatrix}$
Return $\mathsf{crs}_{\mathsf{or}}$

$\underline{\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{y}], \mathbf{w})}$:
If $\neg(\exists j \in \{0,1\} : [\mathbf{y}] = [\mathbf{A}_j \mathbf{w}])$ then abort
Let $[\mathbf{y}]_1 = [\mathbf{A}_j \mathbf{w}]_1$
$(\mathbf{R}, \mathbf{r}) \xleftarrow{\$} \mathbb{Z}_p^{t\times k} \times \mathbb{Z}_p^{1\times k}$
$[\mathbf{C}] := \mathbf{R}[\mathbf{V}^\top] + (1-j)\mathbf{w}\mathbf{w}^\top[\mathbf{E}]$
$[\mathbf{c}] := \mathbf{r}[\mathbf{V}^\top] + (1-j)\mathbf{w}^\top[\mathbf{E}]$
$[\mathbf{\Pi}] := [\mathbf{A}_0]\mathbf{R} - [\mathbf{y}]\mathbf{r}$
$[\pi] := \mathbf{w}^\top[\mathbf{F}] - [\mathbf{r}]$
Return $([\mathbf{C}], [\mathbf{c}], [\mathbf{\Pi}], [\pi])$

$\underline{\mathsf{TGen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0], [\mathbf{A}_1])}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_{l,k}$, $\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{n\times k}$, $\mathbf{D} := \mathbf{K}\mathbf{V}^\top$,
$\mathbf{E} = (\mathbf{A}_0 - \mathbf{A}_1)^\top \mathbf{K}\mathbf{V}^\top$, $\mathbf{F} = \mathbf{A}_1^\top \mathbf{K}$
$\mathsf{crs}_{\mathsf{or}} := \begin{pmatrix} \mathsf{par}, [\mathbf{A}_0], [\mathbf{A}_1], [\mathbf{V}], [\mathbf{D}], \\ [\mathbf{E}], [\mathbf{F}] \end{pmatrix}$
$\mathsf{td}_{\mathsf{or}} := \mathbf{K}$
Return $(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}})$

$\underline{\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{y}], \pi)}$:
If $[\mathbf{A}_0] \circ [\mathbf{C}] - [\mathbf{y}] \circ [\mathbf{c}] \neq [\mathbf{\Pi}] \circ [\mathbf{V}^\top]$
  then return 0
If $[\mathbf{y}^\top] \circ [\mathbf{D}] - [1] \circ [\mathbf{c}] \neq [\pi] \circ [\mathbf{V}^\top]$
  then return 0
Else return 1

$\underline{\mathsf{Sim}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{td}_{\mathsf{or}}, [\mathbf{y}])}$:
$(\mathbf{R}, \mathbf{r}) \xleftarrow{\$} \mathbb{Z}_p^{t\times k} \times \mathbb{Z}_p^{1\times k}$
$[\mathbf{C}] := \mathbf{R}[\mathbf{V}^\top]$
$[\mathbf{c}] := \mathbf{r}[\mathbf{V}^\top]$
$[\mathbf{\Pi}] := [\mathbf{A}_0]\mathbf{R} - [\mathbf{y}]\mathbf{r}$
$[\pi] := [\mathbf{y}^\top]\mathbf{K} - [\mathbf{r}]$
Return $([\mathbf{C}], [\mathbf{c}], [\mathbf{\Pi}], [\pi])$

Fig. 20. Full or-proof for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ in symmetric groups.

$\mathbf{y} = \mathbf{A}_0(\mathbf{C}\mathbf{W})_1/(\mathbf{c}\mathbf{W})_1$, and hence $\mathbf{y}$ is in span of $\mathbf{A}_0$ (here, $(\mathbf{C}\mathbf{W})_1$ is the first column of $\mathbf{C}\mathbf{W}$). □

Efficient Full OR-Proof in Symmetric Groups. We now adapt the designated-prover OR-proof to provide public prover and verifier algorithms, in other words, a full NIZK. To enable this, the CRS now exposes additional elements $[\mathbf{E}]$ and $[\mathbf{F}]$, which allows public proofs, while still preserving zero-knowledge and soundness. On the other hand we need representation of the language matrices additionally in the second coordinate to compute $\mathbf{E}$. This is why the construction is only for the symmetric setting. In addition, we also generalize the dimensions of the language and assumptions, with a mild constraint. We show it satisfies computational soundness and perfect zero-knowledge, while it is easy to show that its dual case satisfies perfect soundness and computational zero-knowledge. The construction is given in Figure 20.

Lemma 16. *If the $\mathcal{D}_{l,k}$-*MDDH *assumption holds in the group $\mathbb{G}$ with $l \geq n-t+k$, then the proof system $\Pi^{\mathsf{or}}_{sym} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ as defined in Figure 20 is a designated-prover or-proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, computational soundness, and perfect zero-knowledge. More precisely, for all adversaries $\mathcal{A}$ attacking the soundness property of Π^{or}_{sym}, we obtain an*

$\underline{\mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, [\mathbf{A}_0] \in \mathbb{G}^{n \times t}, [\mathbf{A}_1] \in \mathbb{G}^{n \times t})}$:
$\mathbf{V} \xleftarrow{\$} \mathcal{D}_{l,k}$
$(\mathbf{K}', \mathbf{L}, \mathbf{T}) \xleftarrow{\$} \mathbb{Z}_p^{n \times k} \times \mathbb{Z}_p^{(n-t) \times k} \times \mathbb{Z}_p^{(n-t) \times l}$
$\mathbf{K} := \mathbf{K}' \; + \mathbf{A}_1^{\perp}\mathbf{L}$
$\mathbf{D} := \mathbf{K}\mathbf{V}^{\top}$
$\mathbf{D} := \mathbf{K}'\mathbf{V}^{\top} + \mathbf{A}_1^{\perp}\mathbf{L}\mathbf{V}^{\top}$
$\boxed{\mathbf{D} := \mathbf{K}'\mathbf{V}^{\top} + \mathbf{A}_1^{\perp}\mathbf{T}}$
$\mathbf{E} := (\mathbf{A}_0 - \mathbf{A}_1)^{\top}\mathbf{D}$
$\mathbf{F} := \mathbf{A}_1^{\top}\mathbf{K}$
$\mathbf{F} := \mathbf{A}_1^{\top}\mathbf{K}'$
$\mathsf{crs}_{\mathsf{or}} := \begin{pmatrix} \mathsf{par}, [\mathbf{A}_0], [\mathbf{A}_1], [\mathbf{V}], [\mathbf{D}], \\ [\mathbf{E}], [\mathbf{F}] \end{pmatrix}$
Return $\mathsf{crs}_{\mathsf{or}}$

$\underline{\mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{y}], j, \mathbf{w})}$:
$(\mathbf{R}, \mathbf{r}) \xleftarrow{\$} \mathbb{Z}_p^{t \times k} \times \mathbb{Z}_p^{1 \times k}$
$[\mathbf{C}] := \mathbf{R}[\mathbf{V}^{\top}] + (1-j)\mathbf{w}\mathbf{w}^{\top}[\mathbf{E}]$
$[\mathbf{c}] := \mathbf{r}[\mathbf{V}^{\top}] + (1-j)\mathbf{w}^{\top}[\mathbf{E}]$
$[\mathbf{\Pi}] := [\mathbf{A}_0]\mathbf{R} - [\mathbf{y}]\mathbf{r}$
$[\pi] := \mathbf{w}^{\top}[\mathbf{F}] - [\mathbf{r}]$
Return $([\mathbf{C}], [\mathbf{c}], [\mathbf{\Pi}], [\pi])$

Fig. 21. Symmetric or-proof for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$: Soundness Games G_1, G_2 and $\boxed{\mathsf{G}_3}$.

adversary $\mathcal{B}$ *with* $T(\mathcal{B}) \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *and*

$$\mathsf{Adv}^{\mathsf{snd}}_{\Pi^{\mathsf{or}}_{sym},\mathcal{A}} \le (n-t) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathcal{G},\mathbb{G},\mathcal{D}_{l,k},\mathcal{B}}(\lambda) + \frac{n-t}{p} + \Delta'_{l,k},$$

where $\Delta'_{l,k}$ *is the probability that* $\mathbf{V} \leftarrow \mathcal{D}_{l,k}$ *has full rank.*

Proof. Completeness follows easily by inspection, and we now show the proofs of zero-knowledge and soundness of Π^{or}.

Perfect zero-knowledge. Observe that the CRS is generated identically in the real and simulation worlds. Perfect zero-knowledge now follows by sampling $\mathbf{R}$ and $\mathbf{r}$ differently, but from an identical statistical distribution:

$$\mathbf{R} \to \mathbf{R} + (1-j)\mathbf{w}\mathbf{w}^{\top}(\mathbf{A}_0 - \mathbf{A}_1)^{\top}\mathbf{K}$$

$$\mathbf{r} \to \mathbf{r} + (1-j)\mathbf{w}^{\top}(\mathbf{A}_0 - \mathbf{A}_1)^{\top}\mathbf{K}$$

Computational soundness. In Figure 21 we describe three games. The first game G_1 is identical to the real world. In the second game $\mathbf{K}$ is generated in a different way, but from an statistically identical distribution. In the third game

G_3, the matrix $\mathbf{D}$ is generated differently, but the games are computationaly indistinguishable by $(n-t)$-fold $\mathcal{D}_{l,k}$-MDDH.

Now, we show that the probability of the Adversary winning the soundness requirement is negligible. Let $([\mathbf{C}], [\mathbf{c}], [\mathbf{\Pi}], [\pi])$, be a valid proof. It must satisfy

$$[\mathbf{A}_0] \circ [\mathbf{C}] - [\mathbf{y}] \circ [\mathbf{c}] = [\mathbf{\Pi}] \circ [\mathbf{V}^\top]$$

and

$$[\mathbf{y}^\top] \circ [\mathbf{D}] - [1] \circ [\mathbf{c}] = [\pi] \circ [\mathbf{V}^\top].$$

Randomly generate matrix $\mathbf{U} \leftarrow \mathbb{Z}_p^{(l-k)\times(n-t)}$ and let $\mathbf{V}^\perp \in \mathbb{Z}_p^{l\times(l-k)}$ be a full rank matrix such that $\mathbf{V}^\top\mathbf{V}^\perp = 0$. Compute $\mathbf{W} = \mathbf{V}^\perp\mathbf{U}$. $\mathbf{TW} = \mathbf{TV}^\perp\mathbf{U}$ is invertible with overwhelming probability at least $1-(n-t)/p$. Since $\mathbf{T}$ and $\mathbf{U}$ are randomly generated, they are full rank with overwhelming probability. Now, since $(n-t) \le (l-k)$, we have that $\mathbf{TV}^\perp$ is a full rank matrix of dimension $(n-t)\times(l-k)$ and rank $(n-t)$. Similarly, $\mathbf{TV}^\perp\mathbf{U}$ is a square matrix of dimension $(n-t)\times(n-t)$ and is full rank. Hence, $\mathbf{TV}^\perp\mathbf{U}$ is invertible.

Now, multiplying above verification equations by $\mathbf{W}$ on the right, we have

$$\mathbf{A}_0(\mathbf{CW}) = \mathbf{y}(\mathbf{cW}) \tag{8}$$

and

$$\mathbf{y}^\top(\mathbf{A}_1^\perp\mathbf{TW}) = \mathbf{cW}. \tag{9}$$

If $\mathbf{cW}$ is zero, then by Equation (9), $\mathbf{y}$ is in span $\mathbf{A}_1$ with overwhelming probability at least $1-(n-t)/p$, i.e., the probability that $\mathbf{TW}$ is invertible. Otherwise, there is a vector $\mathbf{e} \in \mathbb{Z}_p^{n-t}$, such that $\mathbf{cWe} = 1$. Therefore, $\mathbf{y} = \mathbf{A}_0(\mathbf{CWe})$, and hence $\mathbf{y}$ is in span of $\mathbf{A}_0$. □

F Designated-Prover QA-NIZK

In this section, we give a designated-prover QANIZK based on our designated-prover OR-proof system, which only satisfies weak USS but is structure-preserving. Note that all the USS security notions mentioned in this section mean weak ones.

A VARIANT OF THE CORE LEMMA. We now give a variant of the core lemma (i.e., Lemma 3). By slightly changing the proof of Lemma 3 as follows, we immediately obtain the proof of a variant of the core lemma (Lemma 17).

- $\mathsf{Gen}_{\mathsf{or}}$ additionally generates $\mathsf{sk}_{\mathsf{or}}$ and takes as input $\mathbf{A}_0$ and $\mathbf{A}_1$, and
- $\mathsf{Prove}_{\mathsf{or}}$ additionally takes $\mathsf{sk}_{\mathsf{or}}$ as input.

Lemma 17 (A variant of core lemma). *If the $\mathcal{D}_k$-MDDH assumption holds in the group $\mathbb{G}_2$, and $\Pi^{\mathsf{or}} = (\mathsf{Gen}_{\mathsf{or}}, \mathsf{TGen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}}, \mathsf{Sim}_{\mathsf{or}})$ is an or-proof system for $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$ with perfect completeness, perfect soundness, and composable zero-knowledge as defined in Definition 11, then for any adversary $\mathcal{A}$ against the*

$\underline{\textsc{Init}_{\mathsf{core}}}$:	$\underline{\textsc{Eval}_{\mathsf{core}}}$:	$\underline{\textsc{Finalize}_{\mathsf{core}}([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})}$:
$\mathsf{c} := 0$ $\mathbf{A}_0, \mathbf{A}_1 \stackrel{\$}{\leftarrow} \mathcal{D}_{2k,k}$ $\mathsf{par}_{\mathsf{or}} := \mathsf{par}$ $(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}) \stackrel{\$}{\leftarrow} \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, \mathbf{A}_0, \mathbf{A}_1)$ $\mathbf{k} \stackrel{\$}{\leftarrow} \mathbb{Z}_p^{2k}$ $\mathbf{p} := \mathbf{A}_0^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathbf{0})})$ Return $(\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$	$\mathsf{c} := \mathsf{c} + 1$ $\mathbf{s} \stackrel{\$}{\leftarrow} \mathbb{Z}_p^k, [\mathbf{t}]_1 := [\mathbf{A}_0]_1\mathbf{s}$ $u' := \mathbf{t}^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathsf{c})}) \in \mathbb{Z}_p$ $\pi_{\mathsf{or}} \stackrel{\$}{\leftarrow} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ Return $([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}})$	If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$ then return 0 If $[u']_1 = [\mathbf{t}^\top(\mathbf{k} + \boxed{\mathbf{RF}(\mathsf{c}')})]_1$ and $0 \le \mathsf{c}' \le \mathsf{c}$ then return 1 Else return 0

Fig. 22. Security games Core_0 and Core_1 for the variant of the core lemma. $\mathbf{RF} : \mathbb{Z}_p \rightarrow \mathbb{Z}_p^{2k}$ is a random function. All the codes are executed in both games, except the boxed codes which are only executed in Core_1.

core lemma, there exist adversaries $\mathcal{B}$, $\mathcal{B}'$ with running time $T(\mathcal{B}) \approx T(\mathcal{B}') \approx T(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ such that

$$\begin{aligned}\mathsf{Adv}_{\mathcal{A}}^{\mathsf{core}}(\lambda) :=& \Pr[\mathsf{Core}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{Core}_1^{\mathcal{A}} \Rightarrow 1]\\ \le& (4k\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_2, \mathcal{D}_{2k,k}, \mathcal{B}}(\lambda) + (2\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{zk}}_{\mathsf{NIZK}, \mathcal{B}'}(\lambda)\\ &+ \lceil \log Q \rceil \cdot \Delta_{\mathcal{D}_{2k,k}} + \frac{4\lceil \log Q \rceil + 2}{p-1} + \frac{\lceil \log Q \rceil \cdot Q}{p}.\end{aligned}$$

where $\Delta_{\mathcal{D}_{2k,k}}$ is a statistically small term for $\mathcal{D}_{2k,k}$.

F.1 Stepping Stone: Designated-Prover-Verifier QA-NIZK

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, $k \in \mathbb{N}$, and $\Pi^{\mathsf{or}} := (\mathsf{Gen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}})$ be a designated-prover NIZK for language $\mathcal{L}^{\vee}_{\mathbf{A}_0, \mathbf{A}_1}$ (constructed as in Figure 10). Our DPVQANIZK $\Pi^{\mathsf{dpv}} := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined as in Figure 23.

Theorem 5 (Security of Π^{dpv}). *Π^{dpv} is a DPVQANIZK with perfect zero-knowledge and (tightly) unbound simulation soundness. In particular, for any adversary $\mathcal{A}$, there exist adversaries $\mathcal{B}$ and $\mathcal{B}'$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A})$ and*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{dpvss}}_{\Pi^{\mathsf{dpv}}, \mathcal{A}}(\lambda) \le& (4k\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1, \mathcal{D}_{2k,k}, \mathcal{B}}(\lambda)\\ &+ (2\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{zk}}_{\Pi^{\mathsf{or}}, \mathcal{B}'}(\lambda) + \lceil \log Q \rceil \cdot \Delta_{\mathcal{D}_{2k,k}}\\ &+ \frac{4\lceil \log Q \rceil + 2}{p-1} + \frac{(\lceil \log Q \rceil + 1) \cdot Q + 1}{p}.\end{aligned}$$

Proof (of Theorem 5). Perfect completeness follows directly from the correctness of the OR proof system and the fact that for all $\mathbf{y} = \mathbf{M}\mathbf{w}$, $\mathbf{p} := \mathbf{A}_0^\top \mathbf{k}$, $\mathbf{p}_0 := \mathbf{M}^\top \mathbf{k}_0$, and $\mathbf{t} = \mathbf{A}_0 \mathbf{s}$, we have

$$\begin{aligned}\mathbf{w}^\top \mathbf{p}_0 + k_1 + \mathbf{s}^\top \mathbf{p} &= \mathbf{w}^\top \mathbf{M}^\top \mathbf{k}_0 + k_1 + \mathbf{s}^\top \mathbf{A}_0^\top \mathbf{k}\\ &= \mathbf{y}^\top \mathbf{k}_0 + k_1 + \mathbf{t}^\top \mathbf{k}.\end{aligned}$$

$\underline{\mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1 \in \mathbb{G}_1^{n_1 \times n_2}):}$
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$
$(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}) \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}, \mathbf{A}_0, \mathbf{A}_1)$
$\mathbf{k}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1}$, $k_1 \xleftarrow{\$} \mathbb{Z}_p$, $\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$
$\mathbf{p} := \mathbf{A}_0^\top \mathbf{k} \in \mathbb{Z}_p^k$
$[\mathbf{p}_0]_1 := [\mathbf{M}^\top \mathbf{k}_0]_1 \in \mathbb{G}_1^{n_2}$
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1, [\mathbf{p}_0]_1)$
$\mathsf{td} := (\mathsf{sk}_{\mathsf{or}}, \mathbf{k}_0, k_1)$
$\mathsf{vk} := (\mathbf{k}, \mathbf{k}_0, k_1)$
$\mathsf{sk} := (\mathsf{sk}_{\mathsf{or}}, k_1)$
Return $(\mathsf{crs}, \mathsf{vk}, \mathsf{sk}, \mathsf{td})$

$\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1):}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $\mathbf{t} := \mathbf{A}_0 \mathbf{s}$
$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$
$[u]_1 := [\mathbf{y}^\top \mathbf{k}_0]_1 + [k_1]_1 + [\mathbf{s}^\top \mathbf{p}]_1$
Return $\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$

$\underline{\mathsf{Prove}(\mathsf{crs}, \mathsf{sk}, [\mathbf{y}]_1, \mathbf{w}):}$ // $\mathbf{y} = \mathbf{M}\mathbf{w} \in \mathbb{Z}_p^{n_1}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{t}]_1 := [\mathbf{A}_0]_1 \mathbf{s}$
$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$
$[u]_1 := \mathbf{w}^\top [\mathbf{p}_0]_1 + [k_1]_1 + \mathbf{s}^\top [\mathbf{p}]_1$
Return $\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$

$\underline{\mathsf{Ver}(\mathsf{crs}, \mathsf{vk}, [\mathbf{y}]_1, \pi):}$
Parse $\pi = ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$
If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$ then return 0
If $[u]_1 = [\mathbf{y}^\top]_1 \mathbf{k}_0 + k_1 + [\mathbf{t}^\top]_1 \mathbf{k}$ then
  return 1
Else return 0

Fig. 23. Construction of $\Pi^{\mathsf{dpv}} := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$.

Moreover, since

$$\begin{aligned}\mathbf{w}^\top \mathbf{p}_0 + k_1 + \mathbf{s}^\top \mathbf{p} &= \mathbf{w}^\top \mathbf{M}^\top \mathbf{k}_0 + k_1 + \mathbf{s}^\top \mathbf{p} \\ &= \mathbf{y}^\top \mathbf{k}_0 + k_1 + \mathbf{s}^\top \mathbf{p},\end{aligned}$$

proofs generated by Prove and Sim for the same $\mathbf{y} = \mathbf{M}\mathbf{w}$ are identical. Hence, perfect zero knowledge is also satisfied.

We now focus on the tight simulation soundness of Π^{dpv}. Let $\mathcal{A}$ be an adversary against the unbounded simulation soundness of Π^{dpv}. We bound the advantage of $\mathcal{A}$ via a sequence of games defined in Figure 5.

G_0 is the real experiment, DPVSS.

Lemma 18 **(G_0).** $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]$.

Lemma 19 **(G_0 to G_1).** *There is an adversary $\mathcal{B}$ breaking the core lemma (cf. Lemma 19) with running time $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A})$ and*

$$\mathsf{Adv}_{\mathcal{B}}^{\mathsf{core}}(\lambda) = \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1].$$

Proof. We construct the reduction $\mathcal{B}$ defined in Figure 25 to break the core lemma.

Clearly, if the oracles are simulated as in Core_0, then the distribution simulated by $\mathcal{B}$ is the same as in G_1; and if the oracles are simulated as in Core_1 (with a random function $\mathbf{RF}$), then the distribution simulated by $\mathcal{B}$ is the same as in G_2. Thus,

$$\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{Core}_0^{\mathcal{B}} \Rightarrow 1] - \Pr[\mathsf{Core}_1^{\mathcal{B}} \Rightarrow 1] = \mathsf{Adv}_{\mathcal{B}}^{\mathsf{core}}(\lambda),$$

$\underline{\textsc{Init}(\mathbf{M})\text{:}}$ $\boxed{\mathsf{G}_1}$	$\underline{\textsc{Sim}([\mathbf{y}]_1)\text{:}}$ $/\!/ \boxed{\mathsf{G}_1}$
$i := 0$	$\mathsf{c} := \mathsf{c} + 1$
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k}$	$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k$, $[\mathbf{t}]_1 := [\mathbf{A}_0\mathbf{s}]_1$
$\mathsf{par}_{\mathsf{or}} := (\mathsf{par})$	$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$
$\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, \mathbf{A}_0, \mathbf{A}_1)$	$[u]_1 := [\mathbf{y}^\top \mathbf{k}_0 + k_1 + \mathbf{t}^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathsf{c})})]_1$
$\mathbf{k}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1}$, $k_1 \xleftarrow{\$} \mathbb{Z}_p$, $\mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}$	$\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$
$\mathbf{p} := \mathbf{A}_0^\top(\mathbf{k} \boxed{+\mathbf{RF}(\mathbf{0})}) \in \mathbb{Z}_p^k$	$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{[\mathbf{y}]_1\}$
$\mathbf{p}_0 := \mathbf{M}^\top \mathbf{k}_0 \in \mathbb{Z}_p^{n_2}$	Return π
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1, [\mathbf{p}_0]_1)$	
Return crs	$\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)\text{:}}$ $/\!/ \boxed{\mathsf{G}_1}$
	Parse $\pi^* =: ([\mathbf{t}^*]_1, [u^*]_1, \pi^*_{\mathsf{or}})$
	If $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ then return 0
	If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}^*]_1, \pi^*_{\mathsf{or}}) = 0$ then return 0
	$\mathcal{S} := \{[\mathbf{y}^{*\top} \mathbf{k}_0 + k_1 + \mathbf{t}^{*\top}(\mathbf{k} \boxed{+\mathbf{RF}(j^*)})]_1 : 0 \le j^* \le \mathsf{c}\}$
	If $[u^*]_1 \in \mathcal{S}$ then return 1
	Else return 0

Fig. 24. Games G_0, G_1 and G_2 for the proof of Theorem 1. $\mathbf{RF} : \mathbb{Z}_p \to \mathbb{Z}_p^{2k}$ is a random function. Given $\mathbf{M}$ over $\mathbb{Z}_p$, it is efficient to check whether $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$.

which concludes the lemma. □

Lemma 20 **(G_2).** $\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] = \frac{Q}{p}$.

Proof. We apply the following information-theoretical arguments to show that even a computationally unbounded adversary $\mathcal{A}$ can win in G_2 only with negligible probability. If $\mathcal{A}$ wants to win in G_2, then $\mathcal{A}$ needs to output a fresh and valid $\pi^* := ([\mathbf{t}^*]_1, [u^*]_1, \pi^*_{\mathsf{or}})$. According to the additional rejection rule introduced in G_2, $u = \mathbf{y}^{*\top}\mathbf{k}_0 + k_1 + \mathbf{t}^{*\top}(\mathbf{k} + \mathbf{RF}(j^*))$ must hold for some $0 \le j^* \le Q$. Fix a $j^* \le Q$, we show that $\mathcal{A}$ can compute such a u with probability at most $1/p$.

The argument is based on the information leak about $\mathbf{k}_0$ and k_1:

– For the j-th Sim query ($j \neq j^*$), the term $\mathbf{t}^\top\mathbf{RF}(j)$ completely blinds the information about $\mathbf{k}_0$ and k_1 as long as $\mathbf{t} \neq \mathbf{0}$.
– For the j^*-th Sim query, we cannot use the entropy from the term $(\mathbf{k} + \mathbf{RF}(j^*))$ to hide $\mathbf{k}_0$ and k_1 anymore, but we make the following stronger argument. We assume that $\mathcal{A}$ learns the term $\mathbf{t}^\top(\mathbf{k} + \mathbf{RF}(j^*))$, and thus $\mathbf{y}^\top\mathbf{k}_0 + k_1$ is also leaked to $\mathcal{A}$. However, since $\mathbf{y}^* \neq \mathbf{y}$, the terms $\mathbf{p}_0 = \mathbf{M}^\top\mathbf{k}_0 + 0 \cdot k_1$, $\mathbf{y}^\top\mathbf{k}_0 + k_1$ and $\mathbf{y}^{*\top}\mathbf{k}_0 + k_1$ are pairwise independent.

As a result, from $\mathcal{A}$'s view, the term $\mathbf{y}^{*\top}(\mathbf{k}_0 + \tau^* k_1)$ is distributed uniformly at random, given $\mathbf{y}^\top(\mathbf{k}_0 + \tau k_1)$ from the j^*-th Sim query ($[\mathbf{y}]_1$ may not be in $\mathcal{L}_{[\mathbf{M}]_1}$). Thus, $\mathcal{A}$ can compute the random term $\mathbf{y}^{*\top}\mathbf{k}_0$ and make Finalize output 1 with probability at most $1/p$. By the union bound, $\mathcal{A}$ can win in G_2 with probability at most $(Q+1)/p$. □

$\underline{\text{INIT}([\mathbf{M}]_1):}$	$\underline{\text{SIM}([\mathbf{y}]_1):}$
$i := 0$	$\mathsf{c} := \mathsf{c} + 1$
$\mathsf{crs}' \xleftarrow{\$} \text{INIT}_{\mathsf{core}}$	$([\mathbf{t}]_1, [u']_1, \pi_{\mathsf{or}}) \xleftarrow{\$} \text{EVAL}_{\mathsf{core}}$
Parse $\mathsf{crs}' =: (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1)$	$[u]_1 := [\mathbf{y}^\top \mathbf{k}_0 + k_1 + u']_1$
$\mathbf{k}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1}$, $k_1 \xleftarrow{\$} \mathbb{Z}_p$	$\pi := ([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}})$
$[\mathbf{p}_0]_1 := [\mathbf{M}^\top \mathbf{k}_0]_1 \in \mathbb{G}_1^{n_2}$	$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{[\mathbf{y}]_1\}$, Return π
$\mathsf{crs} := (\mathsf{crs}', [\mathbf{p}_0]_1, [\mathbf{p}_1]_1)$	
Return crs	$\underline{\text{FINALIZE}([\mathbf{y}^*]_1, \pi^*):}$
	Parse $\pi^* =: ([\mathbf{t}^*]_1, [u^*]_1, \pi^*_{\mathsf{or}})$
	If $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $([\mathbf{y}^*]_1) \in \mathcal{Q}_{\mathsf{sim}}$ then return 0
	$[u'^*]_1 = [u^*]_1 - [\mathbf{y}^{*\top} \mathbf{k}_0 + k_1]_1$
	Return $\text{FINALIZE}_{\mathsf{core}}([\mathbf{t}^*]_1, [u'^*]_1, \pi^*_{\mathsf{or}})$

Fig. 25. Reduction $\mathcal{B}$ for the proof of Lemma 19 with oracle $\text{INIT}_{\mathsf{core}}$, $\text{EVAL}_{\mathsf{core}}$, $\text{FINALIZE}_{\mathsf{core}}$ defined in Figure 22. We highlight the oracle calls with grey.

From Lemmata 18 to 20, we have

$$\mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{A}}(\lambda) := \Pr[\mathsf{USS}^{\mathcal{A}}] \leq \mathsf{Adv}^{\mathsf{core}}_{\mathcal{B}'}(\lambda) + \frac{(Q+1)}{p}.$$

By Lemma 3, we conclude Theorem 5 as

$$\begin{aligned}\mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}},\mathcal{A}}(\lambda) \leq & (4k\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{mddh}}_{\mathbb{G}_1,\mathcal{D}_{2k,k},\mathcal{B}}(\lambda) \\ & + (2\lceil \log Q \rceil + 2) \cdot \mathsf{Adv}^{\mathsf{zk}}_{\mathsf{NIZK},\mathcal{B}'}(\lambda) + \lceil \log Q \rceil \cdot \Delta_{\mathcal{D}_{2k,k}} \\ & + \frac{4\lceil \log Q \rceil + 2}{p-1} + \frac{(\lceil \log Q \rceil + 1) \cdot Q + 1}{p}.\end{aligned}$$

□

F.2 Designated-Prover QA-NIZK

Let $\mathcal{G} \leftarrow \mathsf{GGen}(1^\lambda)$, $\mathsf{par} := \mathcal{G}$, $k \in \mathbb{N}$, and $\Pi^{\mathsf{or}} := (\mathsf{Gen}_{\mathsf{or}}, \mathsf{Prove}_{\mathsf{or}}, \mathsf{Ver}_{\mathsf{or}})$ be a NIZK system for language $\mathcal{L}^{\vee}_{\mathbf{A}_0,\mathbf{A}_1}$. Our (publicly verifiable) DPQANIZK $\Pi^{\mathsf{dp}} := (\mathsf{Gen}, \mathsf{Prove}, \mathsf{Ver}, \mathsf{Sim})$ is defined as in Figure 26.

Theorem 6 (Security of Π^{dp}). *Π^{dp} defined in Figure 26 is a DPQANIZK with perfect zero-knowledge and (tightly) unbounded simulation soundness if the $\mathcal{D}_k$-KerMDH assumption holds in $\mathbb{G}_2$ and the DPVQANIZK Π^{dpv} in Figure 23 is unbounded simulation sound. In particular, for any adversary $\mathcal{A}$, there exist adversaries $\mathcal{B}$ and $\mathcal{B}'$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{B}') \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$, where Q is the number of queries to SIM, poly is independent of Q and*

$$\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] \leq \mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) + \mathsf{Adv}^{\mathsf{dpvss}}_{\Pi^{\mathsf{dpv}},\mathcal{B}'}(\lambda).$$

$\underline{\mathsf{Gen}(\mathsf{par}, [\mathbf{M}]_1 \in \mathbb{G}_1^{n_1 \times n_2})}$: $\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k},\ \mathbf{A} \xleftarrow{\$} \mathcal{D}_k$ $\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$ $(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}) \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, \mathbf{A}_0, \mathbf{A}_1)$ $\mathbf{K} \xleftarrow{\$} \mathbb{Z}_p^{2k \times (k+1)}$ $\mathbf{K}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1 \times (k+1)}$ $\mathbf{K}_1 \xleftarrow{\$} \mathbb{Z}_p^{1 \times (k+1)}$ $\mathbf{P} := \mathbf{A}_0^\top \mathbf{K} \in \mathbb{Z}_p^{k \times (k+1)}$ $[\mathbf{P}_0]_1 := [\mathbf{M}^\top \mathbf{K}_0]_1 \in \mathbb{G}_1^{n_2 \times (k+1)}$ $\mathbf{C} := \mathbf{K}\mathbf{A} \in \mathbb{Z}_p^{2k \times k}$ $\mathbf{C}_0 := \mathbf{K}_0\mathbf{A} \in \mathbb{Z}_p^{n_1 \times k}$ $\mathbf{C}_1 := \mathbf{K}_1\mathbf{A} \in \mathbb{Z}_p^{1 \times k}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1$ $\quad [\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2)$ $\mathsf{td} := (\mathsf{sk}_{\mathsf{or}}, \mathbf{K}_0, \mathbf{K}_1)$ $\mathsf{sk} := (\mathsf{sk}_{\mathsf{or}}, \mathbf{K}_1)$ Return $(\mathsf{crs}, \mathsf{td})$	$\underline{\mathsf{Prove}(\mathsf{crs}, \mathsf{sk}, [\mathbf{y}]_1, \mathbf{w})}$: $\quad /\!/\ \mathbf{y} = \mathbf{M}\mathbf{w} \in \mathbb{Z}_p^{n_1}$ $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k, [\mathbf{t}]_1 := [\mathbf{A}_0]_1 \mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $[\mathbf{u}]_1 := \mathbf{w}^\top [\mathbf{P}_0]_1 + [\mathbf{K}_1]_1 + \mathbf{s}^\top [\mathbf{P}]_1$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\underline{\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}]_1, \pi)}$: Parse $\pi = ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ If $\mathsf{Ver}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, [\mathbf{t}]_1, \pi_{\mathsf{or}}) = 0$ then return 0 If $[\mathbf{u}]_1 \circ [\mathbf{A}]_2 = [\mathbf{y}^\top]_1 \circ [\mathbf{C}_0]_2 + [1]_1 \circ [\mathbf{C}_1]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{C}]_2$ then $\quad$ return 1 Else return 0 $\underline{\mathsf{Sim}(\mathsf{crs}, \mathsf{td}, [\mathbf{y}]_1)}$: $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k,\ \mathbf{t} := \mathbf{A}_0 \mathbf{s}$ $\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$ $[\mathbf{u}]_1 := [\mathbf{y}^\top \mathbf{K}_0]_1 + [\mathbf{K}_1]_1 + [\mathbf{s}^\top \mathbf{P}]_1$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$

Fig. 26. Construction of Π^{dp}.

Proof (of Theorem 6). Perfect completeness follows directly from the completeness of the OR proof system and the fact that for all $\mathbf{P} := \mathbf{A}_0^\top \mathbf{K}$, $\mathbf{P}_0 := \mathbf{M}^\top \mathbf{K}_0$ $\mathbf{C} := \mathbf{K}\mathbf{A}$, $\mathbf{C}_0 := \mathbf{K}_0\mathbf{A}$, $\mathbf{C}_1 := \mathbf{K}_1\mathbf{A}$,

$$\begin{aligned}
&[\mathbf{w}^\top \mathbf{P}_0 + \mathbf{K}_1 + \mathbf{s}^\top \mathbf{P}]_1 \circ [\mathbf{A}]_2 \\
=&[\mathbf{w}^\top \mathbf{M}^\top \mathbf{K}_0 + \mathbf{K}_1 + \mathbf{s}^\top \mathbf{A}_0^\top \mathbf{K}]_1 \circ [\mathbf{A}]_2 \\
=&[\mathbf{w}^\top \mathbf{M}^\top]_1 \circ [\mathbf{K}_0\mathbf{A}]_2 + [1]_1 \circ [\mathbf{K}_1\mathbf{A}]_2 + [\mathbf{s}^\top \mathbf{A}_0^\top]_1 \circ [\mathbf{K}\mathbf{A}]_2 \\
=&[\mathbf{y}^\top]_1 \circ [\mathbf{C}_0]_2 + [1]_1 \circ [\mathbf{C}_1]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{C}]_2.
\end{aligned}$$

Moreover, since

$$\begin{aligned}
\mathbf{w}^\top \mathbf{P}_0 + \mathbf{K}_1 + \mathbf{s}^\top \mathbf{P} &= \mathbf{w}^\top \mathbf{M}^\top \mathbf{K}_0 + \mathbf{K}_1 + \mathbf{s}^\top \mathbf{P} \\
&= \mathbf{y}^\top \mathbf{K}_0 + \mathbf{K}_1 + \mathbf{s}^\top \mathbf{P},
\end{aligned}$$

the output of Prove is identical to that of Sim for the same $\mathbf{y} = \mathbf{M}\mathbf{w}$. Hence, perfect zero knowledge is also satisfied.

We now focus on the tight simulation soundness of Π^{dp}. We prove it by a sequence of games: G_0 is defined as the real experiment, USS (we omit the description here), G_1 and G_2 are defined as in Figure 27.

Lemma 21 (G_0)**.** $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1]$.

In G_1, Finalize additionally verifies the adversarial forgery with secret keys $\mathbf{K}$, $\mathbf{K}_0$, and $\mathbf{K}_1$ as in Figure 27.

$\underline{\textsc{Init}([\mathbf{M}]_1)\text{:}}$
$\mathbf{A}_0, \mathbf{A}_1 \xleftarrow{\$} \mathcal{D}_{2k,k},\ \mathbf{A} \xleftarrow{\$} \mathcal{D}_k$
$\boxed{\mathbf{a}^\perp \xleftarrow{\$} \mathsf{ker}(\mathbf{A})}$
$\boxed{/\!/\ \mathbf{a}^\perp \in \mathbb{Z}_p^{1\times(k+1)} \text{ and } \mathbf{a}^\perp \cdot \mathbf{A} = \mathbf{0}}$
$\mathsf{par}_{\mathsf{or}} := (\mathsf{par}, [\mathbf{A}_0]_1, [\mathbf{A}_1]_1)$
$\mathsf{crs}_{\mathsf{or}} \leftarrow \mathsf{Gen}_{\mathsf{or}}(\mathsf{par}_{\mathsf{or}}, 1^\lambda)$
$\mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$
$\mathbf{K}'_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$
$\mathbf{K}'_1 \xleftarrow{\$} \mathbb{Z}_p^{1\times(k+1)}$
$\mathbf{k}_0 := \mathbf{0} \in \mathbb{Z}_p^{n_1},\ k_1 := 0 \in \mathbb{Z}_p,\ \mathbf{k} := \mathbf{0} \in \mathbb{Z}_p^{2k}$
$\boxed{\mathbf{k}_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1}, k_1 \xleftarrow{\$} \mathbb{Z}_p^{1\times k},\ \mathbf{k} \xleftarrow{\$} \mathbb{Z}_p^{2k}}$
$\mathbf{K} := \mathbf{K}' + \mathbf{k}\cdot\mathbf{a}^\perp$
$\mathbf{K}_0 := \mathbf{K}'_0 + \mathbf{k}_0\cdot\mathbf{a}^\perp$
$\mathbf{K}_1 := \mathbf{K}'_1 + k_1\cdot\mathbf{a}^\perp$
$\mathbf{P} := \mathbf{A}_0^\top\mathbf{K} \in \mathbb{Z}_p^{k\times(k+1)}$
$[\mathbf{P}_0]_1 := [\mathbf{M}^\top\mathbf{K}_0]_1 \in \mathbb{G}_1^{n_2\times(k+1)}$
$\mathbf{C} := \mathbf{K}\mathbf{A} \in \mathbb{Z}_p^{2k\times k}$
$\mathbf{C}_0 := \mathbf{K}_0\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$
$\mathbf{C}_1 := \mathbf{K}_1\mathbf{A} \in \mathbb{Z}_p^{1\times k}$
$\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1,$
$\quad [\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2)$
Return crs

$\underline{\textsc{Sim}([\mathbf{y}]_1)\text{:}}$
$\mathbf{s} \xleftarrow{\$} \mathbb{Z}_p^k,\ [\mathbf{t}]_1 := [\mathbf{A}_0]_1\mathbf{s}$
$\pi_{\mathsf{or}} \xleftarrow{\$} \mathsf{Prove}_{\mathsf{or}}(\mathsf{crs}_{\mathsf{or}}, \mathsf{sk}_{\mathsf{or}}, [\mathbf{t}]_1, \mathbf{s})$
$[\varDelta]_1 := [(\mathbf{y}^\top\mathbf{k}_0 + k_1 + \mathbf{t}^\top\mathbf{k})\cdot\mathbf{a}^\perp]_1$
$[\mathbf{u}]_1 := [\mathbf{y}^\top\mathbf{K}'_0 + \mathbf{K}'_1 + \mathbf{t}^\top\mathbf{K}' + \varDelta]_1$
$\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{[\mathbf{y}]_1\}$
Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$

$\underline{\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)\text{:}}$
Parse $\pi = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$
If $[\mathbf{y}^*]_1 \in \mathcal{Q}_{\mathsf{sim}}$ or $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 0$ then
  return 0
$[\varDelta^*]_1 := [(\mathbf{y}^{*\top}\mathbf{k}_0 + k_1 + \mathbf{t}^{*\top}\mathbf{k})\cdot\mathbf{a}^\perp]_1$
If $[\mathbf{u}^*]_1 = [\mathbf{y}^{*\top}(\mathbf{K}'_0 + \tau^*\mathbf{K}'_1) + \mathbf{t}^{*\top}\mathbf{K}' + \varDelta^*]_1$ then
  return 1
Else return 0

Fig. 27. Games G_1 and $\boxed{\mathsf{G}_2}$ for proving Theorem 6.

Lemma 22 (G_0 **to** G_1)**.** *There is an adversary $\mathcal{B}$ breaking the $\mathcal{D}_k$-*KerMDH *assumption over $\mathbb{G}_2$ with $\mathsf{T}(\mathcal{B}) \approx \mathsf{T}(\mathcal{A}) + Q\cdot\mathsf{poly}(\lambda)$ and*

$$\mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_2,\mathcal{D}_k,\mathcal{B}}(\lambda) \geq |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|.$$

Proof. It is straightforward that a pair $([\mathbf{y}^*]_1, \pi^*)$ passing the Finalize in G_1 always passes the Finalize in G_0. We now bound the probability that $\mathcal{A}$ produces $([\mathbf{y}^*]_1, \pi^*)$ that passes the verification in G_0 but not that in G_1. For $\pi^* = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$, the verification equation in G_0 is:

$$\begin{aligned}
[\mathbf{u}^*]_1 \circ [\mathbf{A}]_2 &= [\mathbf{y}^{*\top}]_1 \circ [\mathbf{K}_0\mathbf{A}]_2 + [1]_1 \circ [\mathbf{K}_1\mathbf{A}]_2 + [\mathbf{t}^\top]_1 \circ [\mathbf{K}\mathbf{A}]_2 \\
&\Leftrightarrow [\mathbf{u}^* - \mathbf{y}^{*\top}\mathbf{K}_0 - \mathbf{K}_1 - \mathbf{t}^\top\mathbf{K}]_1 \circ [\mathbf{A}]_2 = [\mathbf{0}]_T.
\end{aligned}$$

One can see that for any $([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$ that passes the verification equation in G_0 but not that in G_1, $\mathbf{u}^* - \mathbf{y}^*\mathbf{K}_0 - \mathbf{K}_1 - \mathbf{t}^\top\mathbf{K}$ is a non-zero vector in the kernel of $\mathbf{A}$.

We now construct an adversary $\mathcal{B}$ as follows. On receiving $(\mathcal{G}, [\mathbf{A}]_1)$ from the $\mathcal{D}_k$-KerMDH experiment, $\mathcal{B}$ samples all other parameters by itself and simulates

G_0 for $\mathcal{A}$. When $\mathcal{A}$ outputs a tuple $([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$, $\mathcal{B}$ outputs $\mathbf{u}^* - \mathbf{y}^{*\top}\mathbf{K}_0 - \mathbf{K}_1 - \mathbf{t}^\top\mathbf{K}$. Since $\mathcal{B}$ succeeds in its experiment when $\mathcal{A}$ outputs a tuple such that $\mathbf{u}^* - \mathbf{y}^{*\top}\mathbf{K}_0 - \mathbf{K}_1 - \mathbf{t}^\top\mathbf{K}$ is a non-zero vector in the kernel of $\mathbf{A}$, we have $\mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1,\mathcal{D}_k,\mathcal{B}}(\lambda) \geq |\Pr[\mathsf{G}_0^{\mathcal{A}} \Rightarrow 1] - \Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1]|$, completing the proof of this lemma. □

Lemma 23 (G_1 to G_2). $\Pr[\mathsf{G}_1^{\mathcal{A}} \Rightarrow 1] = \Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1]$.

Proof. Now we finish the reduction to the KerMDH assumption and we can have $\mathbf{A}$ over $\mathbb{Z}_p$. In G_2, for $i \in \{0,1\}$ we replace $\mathbf{K}_i$ by $\mathbf{K}'_i + \mathbf{k}_i\mathbf{a}^\perp$ for $\mathbf{a}^\perp \in \mathsf{ker}(\mathbf{A})$. Furthermore, we replace $\mathbf{K}$ by $\mathbf{K}' + \mathbf{k}\mathbf{a}^\perp$. Since $\mathbf{K}'$ and $\mathbf{K}'_i$ are uniformly random, $\mathbf{K}$ and $\mathbf{K}_i$ in G_2 are distributed at random and the same as in G_1. Thus, G_2 is distributed the same as G_1. □

Lemma 24 (G_2). *There is an adversary $\mathcal{B}'$ breaking the* DPVSS *security of Π^{dpv} defined in Figure 23 with* $\mathsf{T}(\mathcal{B}') \approx \mathsf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ *and*

$$\Pr[\mathsf{G}_2^{\mathcal{A}} \Rightarrow 1] \leq \mathsf{Adv}^{\mathsf{dpvss}}_{\Pi^{\mathsf{dpv}},\mathcal{B}'}(\lambda).$$

Proof. We construct a reduction $\mathcal{B}'$ in Figure 28 to break the DPVSS security of Π^{dpv} defined in Figure 23.

$\underline{\text{INIT}(\mathbf{M})\text{:}}$	$\underline{\text{SIM}([\mathbf{y}]_1)\text{:}}$
$\mathbf{A} \xleftarrow{\$} \mathcal{D}_k$ $\mathbf{a}^\perp \xleftarrow{\$} \mathsf{ker}(\mathbf{A})$ // $\mathbf{a}^\perp \in \mathbb{Z}_p^{1\times(k+1)}$ and $\mathbf{a}^\perp\mathbf{A} = \mathbf{0}$ $\mathsf{crs}_{\mathsf{dpv}} \xleftarrow{\$} \text{INIT}_{\mathsf{dpv}}(\mathbf{M})$ Parse $\mathsf{crs}_{\mathsf{dpv}} =: (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{p}]_1,$ $[\mathbf{p}_0]_1)$ $\mathbf{K}' \xleftarrow{\$} \mathbb{Z}_p^{2k\times(k+1)}$ $\mathbf{K}'_0 \xleftarrow{\$} \mathbb{Z}_p^{n_1\times(k+1)}$ $\mathbf{K}'_1 \xleftarrow{\$} \mathbb{Z}_p^{1\times(k+1)}$ $[\mathbf{P}]_1 := [\mathbf{A}_0]_1^\top\mathbf{K}' + [\mathbf{p}]_1\mathbf{a}^\perp$ $[\mathbf{P}_0]_1 := [\mathbf{M}]_1^\top\mathbf{K}'_0 + [\mathbf{p}_0]_1\mathbf{a}^\perp$ $\mathbf{C} := \mathbf{K}'\mathbf{A} \in \mathbb{Z}_p^{2k\times k}$ $\mathbf{C}_0 := \mathbf{K}'_0\mathbf{A} \in \mathbb{Z}_p^{n_1\times k}$ $\mathbf{C}_1 := \mathbf{K}'_1\mathbf{A} \in \mathbb{Z}_p^{1\times k}$ $\mathsf{crs} := (\mathsf{crs}_{\mathsf{or}}, [\mathbf{A}_0]_1, [\mathbf{P}]_1, [\mathbf{P}_0]_1, [\mathbf{P}_1]_1$ $[\mathbf{A}]_2, [\mathbf{C}]_2, [\mathbf{C}_0]_2, [\mathbf{C}_1]_2, H)$ Return crs	$([\mathbf{t}]_1, [u]_1, \pi_{\mathsf{or}}) \xleftarrow{\$} \text{SIM}_{\mathsf{dpv}}([\mathbf{y}]_1)$ $[\Delta]_1 := [u]_1 \cdot \mathbf{a}^\perp$ $[\mathbf{u}]_1 := [\mathbf{y}^\top\mathbf{K}'_0 + \mathbf{K}'_1 + \mathbf{t}^\top\mathbf{K}' + \Delta]_1$ $\mathcal{Q}_{\mathsf{sim}} := \mathcal{Q}_{\mathsf{sim}} \cup \{([\mathbf{y}]_1, \pi)\}$ Return $\pi := ([\mathbf{t}]_1, [\mathbf{u}]_1, \pi_{\mathsf{or}})$ $\underline{\text{FINALIZE}([\mathbf{y}^*]_1, \pi^*)\text{:}}$ Parse $\pi = ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}})$ If $([\mathbf{y}^*]_1, \pi^*) \in \mathcal{Q}_{\mathsf{sim}}$ or $[\mathbf{y}^*]_1 \in \mathcal{L}_{[\mathbf{M}]_1}$ or $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 0$ then return 0 Compute $[v]_1$ such that $[v]_1\mathbf{a}^\perp = [\mathbf{u}^* - \mathbf{y}^{*\top}\mathbf{K}'_0 - \mathbf{K}'_1 - \mathbf{t}^{*\top}\mathbf{K}']_1$ Return $\text{FINALIZE}_{\mathsf{dv}}([\mathbf{y}^*]_1, ([\mathbf{t}^*]_1, [v]_1, \pi^*_{\mathsf{or}}))$

Fig. 28. Reduction $\mathcal{B}'$ for the proof of Lemma 24 with oracle access to $\text{INIT}_{\mathsf{dpv}}$, $\text{SIM}_{\mathsf{dpv}}$ and $\text{FINALIZE}_{\mathsf{dpv}}$ as defined in G_0 of Figure 24. We highlight the oracle calls with grey.

We note that the $[\mathbf{p}]_1, [\mathbf{p}_0]_1$ from $\text{INIT}_{\mathsf{dpv}}$ have the forms, $\mathbf{p} = \mathbf{A}_0^\top\mathbf{k}$ and $\mathbf{p}_0 = \mathbf{M}^\top\mathbf{k}_0$ for some random $\mathbf{k} \in \mathbb{Z}_p^{2k}$ and $\mathbf{k}_0 \in \mathbb{Z}_p^{n_1}$, and furthermore the value $[u]_1$ from $\textsc{Sim}_{\mathsf{dpv}}$ has the form $u = \mathbf{y}^\top \mathbf{k}_0 + k_1 + \mathbf{t}^\top \mathbf{k}$. Hence, essentially, $\mathcal{B}'$ simulate the security game with $\mathbf{K}$ and $\mathbf{K}_i$ that are implicitly defined as $\mathbf{K} := \mathbf{K}' + \mathbf{k} \cdot \mathbf{a}^\perp$, $\mathbf{K}_0 := \mathbf{K}'_0 + \mathbf{k}_0 \cdot \mathbf{a}^\perp$, and $\mathbf{K}_1 := \mathbf{K}'_1 + k_1 \cdot \mathbf{a}^\perp$. The simulated $\textsc{Init}$ and $\textsc{Sim}$ are identical to those in G_2.

In G_2, $\textsc{Finalize}([\mathbf{y}^*]_1, \pi^* := ([\mathbf{t}^*]_1, [\mathbf{u}^*]_1, \pi^*_{\mathsf{or}}))$ outputs 1 if

$$\mathbf{u}^* = {\mathbf{y}^*}^\top \mathbf{K}'_0 + \mathbf{K}'_1 + {\mathbf{t}^*}^\top \mathbf{K}' + (\underbrace{{\mathbf{y}^*}^\top \mathbf{k}_0 + k_1 + {\mathbf{t}^*}^\top \mathbf{k}}_{=:v}) \cdot \mathbf{a}^\perp$$

and $([\mathbf{y}^*]_1) \notin \mathcal{Q}_{\mathsf{sim}}$ and $[\mathbf{y}^*]_1 \notin \mathcal{L}_{[\mathbf{M}]_1}$ and $\mathsf{Ver}(\mathsf{crs}, [\mathbf{y}^*]_1, \pi^*) = 1$. Thus, if $\mathcal{A}$ can make $\textsc{Finalize}([\mathbf{y}^*]_1, \pi^*)$ output 1 then $\mathcal{B}'$ can extract the corresponding $[v]_1$ to break the USS security. We conclude the lemma. $\square$

To sum up, we have $\Pr[\mathsf{USS}^{\mathcal{A}} \Rightarrow 1] \leq \mathsf{Adv}^{\mathsf{kmdh}}_{\mathbb{G}_1, \mathcal{D}_k, \mathcal{B}}(\lambda) + \mathsf{Adv}^{\mathsf{uss}}_{\Pi^{\mathsf{dv}}, \mathcal{B}'}(\lambda)$ with $\mathcal{B}$ and $\mathcal{B}'$ as defined above. $\square$