Matrix Computational Assumptions in Multilinear Groups

Paz Morillo[1], Carla Ràfols[2], and Jorge L. Villar[1⋆]

[1] Universitat Politècnica de Catalunya, Spain
{paz,jvillar}@ma4.upc.edu
[2] Horst Görtz Institut für IT Sicherheit, Ruhr-Universität Bochum, Germany
carla.rafols@rub.de

Abstract. We put forward a new family of computational assumptions, the Kernel Matrix Diffie-Hellman Assumption. Given some matrix $\mathbf{A}$ sampled from some distribution $\mathcal{D}_{\ell,k}$, the kernel assumption says that it is hard to find "in the exponent" a nonzero vector in the kernel of $\mathbf{A}^\top$. This family is the natural computational analogue of the Matrix Decisional Diffie-Hellman Assumption (MDDH), proposed by Escala *et al.* As such it allows to extend the advantages of their algebraic framework to computational assumptions.
The k-Decisional Linear Assumption is an example of a family of decisional assumptions of strictly increasing hardness when k grows. We show that for any such family the corresponding Kernel Assumption family is also a strictly increasingly weaker family of computational assumptions. This requires ruling out the existence of some black-box reductions between flexible problems (*i.e.*, computational problems with a non unique solution).

1 Introduction

It is always desirable to base security of cryptographic protocols on the weakest possible assumptions, like discrete logarithm or factoring. Although this is possible in many scenarios, it usually limits either the efficiency or the functionality of the protocols. This is the main reason why stronger assumptions like DDH are broadly used. However, such a strong assumption is not always true, like in the case of symmetric bilinear groups. Therefore, it is important to have a thorough understanding of the hardness of computational assumptions and their relations.

This issue has been treated extensively in the cryptographic literature, *e.g.*, in [4,19,26,27,28,30,31] just to name a few. On the computational side, many of the proposed assumptions are often shown to be equivalent [4,19,30]. Typically, most computational problems related to prime order groups are equivalent or reducible to CDH. However, on the stronger side, it is difficult to find relations between decisional assumptions [4,10,11,30] which makes it hard to compare the security achieved by different cryptographic protocols based on different assumptions.

The security notions for cryptographic protocols can be classified mainly in hiding and unforgeability ones. The former typically appear in encryption schemes and commitments and the latter in signature schemes and soundness in zero knowledge proofs. Although it is theoretically possible to base the hiding property on computational problems, most of the practical schemes achieve this notion either information theoretically or based on decisional assumptions, at least in the standard model. Likewise, unforgeability naturally comes from computational assumptions (typically implied by stronger, decisional assumptions).

Most computational problems considered in the literature are search problems with a unique solution like discrete logarithm or CDH. But, unforgeability actually means the inability to produce one among many solutions to a given problem (*e.g.*, in many signature schemes or zero knowledge proofs). Thus, unforgeability is more naturally captured by a *flexible computational problem*, namely, a problem which admits several solutions[3]. Unfortunately, flexible problems have received less attention until recently [2,5,9,16,17,23,25]. This is probably due to the difficulty of finding reductions among them, and even worse, of fully grasping the meaning of a black-box reduction between them. A better understanding

⋆ This work has been partially supported by the Spanish research project MTM2013-41426-R.

[3] In the cryptographic literature we sometimes find the term "strong" as an alternative to "flexible", like the Strong RSA or the Strong DDH

of flexible problems, which are weaker than non-flexible computational ones and harder than decisional ones, would be a useful tool to design simpler or more efficient signature and zero knowledge protocols with extended functionalities (in fact our framework has already been used in [22] and [21]), similarly to what happened with decisional problems and encryption schemes with the work of Escala *et al.* [12].

The contribution of [12] is to put forward a new family of decisional assumptions in a prime order group $\mathbb{G}$, the *Matrix Diffie-Hellman* Assumption ($\mathcal{D}_{\ell,k}$-MDDH). It says that, given some matrix $\mathbf{A} \in \mathbb{Z}_q^{\ell \times k}$ sampled from some distribution $\mathcal{D}_{\ell,k}$, it is hard to decide membership in $\operatorname{Im} \mathbf{A}$ in "the exponent". Rather than as new assumption, it should be seen as an algebraic framework for decisional assumptions which includes as a special case the widely used k-Lin family. In line with the objective of improving on the understanding of cryptographic assumptions, a natural question is if one can find an interesting computational analogue of their MDDH Assumption.

1.1 Our results

In the following $\mathcal{G} = (\mathbb{G}, q, \mathcal{P})$, being $\mathbb{G}$ some group of order q generated by $\mathcal{P}$ where the discrete logarithm is hard, and its elements are denoted $[a] := a\mathcal{P}$.

Computational Matrix Assumptions. In our first attempt to design a computational analogue of the MDDH Assumption, we introduce the *Matrix Computational DH Assumption*, (MCDH) which says that, given a uniform vector $[\boldsymbol{v}] \in \mathbb{G}^k$ and some matrix $[\mathbf{A}]$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ for $\ell > k$, it is hard to extend $[\boldsymbol{v}]$ to a vector in $\mathbb{G}^\ell$ in $\operatorname{Im}[\mathbf{A}]$. Although this assumption is natural and is weaker than the MDDH one, we argue that it is equivalent to CDH.

We then propose the *Kernel Matrix DH Assumption* ($\mathcal{D}_{\ell,k}$-KerMDH). This new flexible assumption states that, given some matrix $[\mathbf{A}]$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ for some $\ell > k$, it is hard to find a vector $[\boldsymbol{v}] \in \mathbb{G}^\ell$ in the kernel of $\mathbf{A}^\top$. We observe that for some special instances of $\mathcal{D}_{\ell,k}$, these assumptions have appeared in the literature in [2,9,16,17,23,25] under different names, like *Simultaneous Pairing, Simultaneous Double Pairing (SDP in the following), Simultaneous Triple Pairing, 1-Flexible CDH, 1-Flexible Square CDH.* Thus, the new KerMDH Assumption allows us to organize and give a unified view on several useful assumptions. This suggests that the KerMDH Assumption (and not the MCDH one) is the right computational analogue of the MDDH framework. We define a generalization of the KerMDH Assumption in multilinear maps where the solution must be in one of the intermediate groups.

The criterions for generic hardness in k-linear maps for $\mathcal{D}_{\ell,k}$-MDDH also apply to the corresponding Kernel Assumption, because $\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow \mathcal{D}_{\ell,k}$-KerMDH. On the other hand, we argue that the known attacks against decisional MDDH Assumptions in the current candidate k-linear maps [14] do not seem to apply to the KerMDH Assumptions. We leave as an open question to further study its security in the present candidate instantiations of k-linear maps.

The power of Kernel Assumptions. At Eurocrypt 2015, the KerMDH Assumptions were applied to design simpler QA-NIZK proofs of membership in linear spaces [22]. They have also been used to give more efficient constructions of structure preserving signatures [21]. The power of a KerMDH Assumption is that it allows to guarantee uniqueness. This has been useful, for instance, to compile some secret key primitives to the public key setting, a line of work initiated in [7]. Indeed, Kiltz *et al.* [22] modify a hash proof system (which is only designated verifier) to allow public verification (a QA-NIZK proof of membership). In a hash proof system for membership in some linear subspace of $\mathbb{G}^n$ spanned by the columns of some matrix $[\mathbf{M}]$, the public information is $[\mathbf{M}^\top \mathbf{K}]$, for some secret matrix $\mathbf{K}$, and given the proof $[\boldsymbol{\pi}]$ that $[\boldsymbol{y}]$ is in the subspace, verification tests if $[\boldsymbol{\pi}] \stackrel{?}{=} [\mathbf{y}^\top \mathbf{K}]$.

The core argument to compile this to a public key primitive is that given $([\mathbf{A}], [\mathbf{KA}])$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ and any pair $[\boldsymbol{y}], [\boldsymbol{\pi}]$,

$$e([\boldsymbol{\pi}^\top], [\mathbf{A}]) = e([\boldsymbol{y}^\top], [\mathbf{KA}]) \iff e([\boldsymbol{\pi}^\top - \boldsymbol{y}^\top \mathbf{K}], [\mathbf{A}]) = [\mathbf{0}]_2 \stackrel{\mathcal{D}_{\ell,k}\text{-KerMDH}}{\Longrightarrow} [\boldsymbol{\pi}] = [\boldsymbol{y}^\top \mathbf{K}]. \tag{1}$$

That is, although potentially there are many possible proofs which satisfy the public verification equation (left hand side of Equation (1)), the $\mathcal{D}_{\ell,k}$-KerMDH Assumption guarantees that only one of them is

efficiently computable, so verification gives the same guarantees as in the private key setting (right hand side of Equation (1)). This property is also used in a very similar way in the context of structure preserving signatures [21]. In Section 5 we use it to argue that, of all the possible openings of a commitment, only one is efficiently computable, *i.e.* to prove computational soundness of a commitment scheme. This property is true of any KerMDH Assumption, and in particular it explains the great number of applications of the Simultaneous (Double) Pairing Assumption, most notably in the design of structure preserving cryptographic primitives [1,2,3,24] (to name a few). We expect that these constructions can be generalized to any KerMDH Assumption.

The advantages of the Kernel Abstraction. As it was also true for its decisional variant, the generalization to any Kernel assumption is useful in several ways. First, schemes based on any (decisional or computational) $\mathcal{D}_{\ell,k}$ Matrix Diffie-Hellman Assumption tend to highlight the algebraic structure of the construction. Further, many instantiations of the given scheme can be written in a compact way and this abstraction points out to a tradeoff between security and efficiency. Indeed, on the one hand, the uniform assumption is the weakest of all possible assumptions but has the worst representation size, while the symmetric cascade (defined in [12]) has optimal representation size but is a stronger assumption. The algebraic viewpoint however, is not new or unique to this paper. The specific gain of introducing the $\mathcal{D}_{\ell,k}$-KerMDH Assumption is that one can properly refer to the assumption on which security is based — rather than just saying "security is based on an assumption weaker than $\mathcal{D}_{\ell,k}$-MDDH" —. For instance, the SDP Assumption ($\mathcal{RL}_2$-KerMDH) is weaker than the $\mathcal{L}_2$-MDDH (or 2-Lin) Assumption and, when possible, it is more precise to say that security is based on the former than on the latter.

This lack of precision is always undesirable in a reduction argument, but in the present case it leads to some additional problems. One minor issue is that works which (implicitly) base the security on the SDP Assumption but claim to base it on the 2-Lin Assumption are typically making non-optimal choices which affect the efficiency of the protocol. This is because there are other computational assumptions which are also weaker than $\mathcal{L}_2$-MDDH but have better representation size (in particular, the Kernel 2-Lin Assumption), and which would typically reduce the size of the public parameters and the number of cryptographic operations.

Another problem is that it is not clear if there are increasingly weaker families of KerMDH Assumptions. That is, some decisional assumptions families parameterized by k like the k-Lin Assumption are known to strictly increasingly weaker. The proof of increasing hardness is more or less immediate and the term *strictly* follows from the fact that every two $\mathcal{D}_{\ell,k}$-MDDH and $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-MDDH problems with $\widetilde{k} < k$ are separated by an oracle computing a k-linear map. For the computational case, increasing hardness is also not too difficult, but nothing is known about *strictly* increasing hardness (see Fig. 1). This means that, as opposed to the decisional case, for protocols based on KerMDH Assumptions there is no-known tradeoff between larger k (less efficiency) and security.

Therefore, the claim that security is based on the MDDH decisional assumptions when only computational ones are necessary might give the impression that a certain tradeoff is in place when this is not known to be the case. For instance, Jutla and Roy [20] construct constant-size QA-NIZK arguments of membership in linear spaces under what they call the "Switching Lemma", which is proven under a certain $\mathcal{D}_{k+1,k}$-MDDH Assumption. However, a close look at the proof reveals that in fact it is based on the corresponding $\mathcal{D}_{k+1,k}$-KerMDH Assumption[4] This means in particular that it is unclear if the choice of larger k gives any additional guarantees, and the claim that the security relies on decisional assumptions might obscure this.

Main result: strictly increasing families of Kernel Assumptions. We first show that the families of matrix distributions in [12], $\mathcal{U}_{\ell,k}$, $\mathcal{L}_k$, $\mathcal{SC}_k$, $\mathcal{C}_k$ and $\mathcal{RL}_k$, as well as $\mathcal{CI}_{k,d}$, define families of kernel

[4] To see this, note that in the proof of their "Switching Lemma" on which soundness is based, they use the output of the adversary to decide if $\boldsymbol{f} \stackrel{?}{\in} \operatorname{Im} \mathbf{A}$, $\mathbf{A} \leftarrow \mathcal{RL}_k$, by checking whether $[\boldsymbol{f}]$ is orthogonal to the adversary's output (equation (1), proof of Lemma 1, [20], full version), and where $\mathcal{RL}_k$ is the matrix distribution of Section 2.3

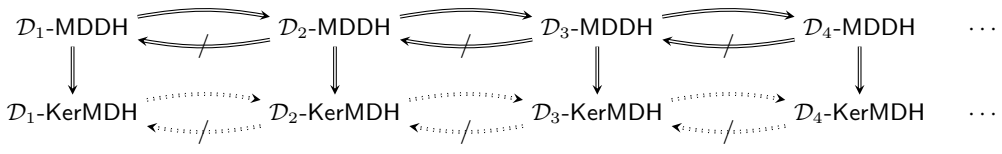

Fig. 1. If $(\mathcal{D}_k\text{-MDDH})_k$ are (strictly) increasingly weaker assumptions, the corresponding Kernel assumptions are not necessarily (strictly) increasingly weaker. That is, the (non-)implications in dots do not follow from the other (non-)implications.

problems with increasing hardness. For this we show show a rather straightforward reduction from the smaller to the larger problems in each family. Our main result (Theorem 1) is to prove that the hardness of these problems is *strictly* increasing. For this, we prove that there is no black-box reduction from the larger to the smaller problems in the multilinear generic group model. These two results together prove all the implications in dots in Fig. 1.

The last step requires dealing with the notion of black-box reduction between flexible problems. A black-box reduction must work for any possible behavior of the oracle, but, on the contrary to the normal (unique answer) black-box reductions, here the oracle has to choose among the set of valid answers in every call. Ruling out the existence of a reduction implies that for any reduction there is an oracle behavior for which the reduction fails. This is specially subtle when dealing with multiple oracle calls. We think that the proof technique we introduce to deal with these issues can be considered as a contribution in itself and it can potentially be used in future work.

Theorem 1 justifies the intuition that there is a tradeoff between the size of the matrix — which typically results in less efficiency — and the hardness of the KerMDH Problems, and justifies the generalization of several protocols to different choices of k given in of [20,22,21].

New Applications. The discussion of our results given so far should already highlight some of the advantages of using the new Kernel family of assumptions and the power of these new assumptions, which have already been used in [22,21]. To further illustrate the usefulness of the new framework, we apply it to the study of trapdoor commitments. First, we revisit the Pedersen commitment [29] to vectors of scalars and its generalization to vectors of group elements of Abe *et al.* [2] in bilinear maps. We generalize the construction to commit vectors of elements at each level $\mathbb{G}_r$, for any $0 \leq r \leq m$ under the extension of KerMDH Assumptions to the ideal m-graded encodings setting. In particular, when $m = 2$ we recover in a single construction as a special case both the original Pedersen commitment and its generalization to vectors of group elements.

The (generalized) Pedersen commitment maps vectors in $\mathbb{G}_r$ to vectors in $\mathbb{G}_{r+1}$, is perfectly hiding and computationally binding under some Kernel Assumption. In Sect. 5.2 we give a black-box construction from any such "shrinking" commitment to a "group-to-group" commitment, *i.e.* commitments which map vectors in $\mathbb{G}_r$ to vectors in the same group $\mathbb{G}_r$. These commitments were defined in [3] because they are a good match to Groth-Sahai proofs. In [3], two constructions were given, one in asymmetric and the other in symmetric bilinear groups. Both are optimal in terms of commitment size and number of verification equations. Rather surprisingly, we show that both constructions in [3], are special instances of our group-to-group commitment when the underlying "shrinking" commitment is the Pedersen commitment for some specific matrix distributions.

A new family of MDDH Assumptions of optimal representation size. We also propose a new interesting family of Matrix distributions, the circulant matrix distribution, $\mathcal{CI}_{k,d}$, which defines new MDDH and KerMDH assumptions. This family generalizes the Symmetric Cascade Distribution ($\mathcal{SC}_k$) defined in [12] to matrices of size $\ell \times k$, $\ell = k + d > k + 1$. We prove that the $\mathcal{CI}_{k,d}$-MDDH Assumption is generically hard in k-linear maps. We prove that it has optimal representation size d independent of k among all matrix distributions of the same size. The case $\ell > k + 1$ typically arises when one considers

commitments/encryption in which the message is a vector of group elements instead of a single group element and the representation size typically affects the size of the public parameters.

Analyzing the hardness of a family of decisional problems (depending on a parameter k) can be rather involved, specially when an efficient k-linear map is supposed to exist. This is why in [12], the authors gave a practical criterion for generic hardness when $\ell = k + 1$ in terms of irreducibility of some polynomials involved in the description of the problem. This criterion was used then to prove the generic hardness of several families of MDDH Problems. To analyze the generic hardness of the $\mathcal{CI}_{k,d}$-MDDH Problem for any d, the techniques in [12] are not practical enough, and we needed some extensions of these techniques for the case $\ell > k + 1$, recently introduced in [18]. However, we could not avoid the explicit computation of a large (but well-structured) Gröbner basis of an ideal associated to the matrix distribution. The new assumption can be used to instantiate the commitment schemes of Section 5 with shorter public parameters and improved efficiency.

2 Preliminaries

For $\lambda \in \mathbb{N}$, we write 1^λ for the string of λ ones. For a set S, $s \leftarrow S$ denotes the process of sampling an element s from S uniformly at random. For an algorithm $\mathcal{A}$, we write $z \leftarrow \mathcal{A}(x, y, \ldots)$ to indicate that $\mathcal{A}$ is a (probabilistic) algorithm that outputs z on input $(x, y, \ldots)$. For any two computational problems $\mathbb{P}_1$ and $\mathbb{P}_2$ we recall that $\mathbb{P}_1 \Rightarrow \mathbb{P}_2$ denotes the fact that $\mathbb{P}_1$ reduces to $\mathbb{P}_2$, and then '$\mathbb{P}_1$ is hard' $\Rightarrow$ '$\mathbb{P}_2$ is hard'. Thus, we will use '$\Rightarrow$' both for computational problems and for the corresponding hardness assumptions.

Let Gen denote a cyclic group instance generator, that is a probabilistic polynomial time (PPT) algorithm that on input 1^λ returns a description $\mathcal{G} = (\mathbb{G}, q, \mathcal{P})$ of a cyclic group $\mathbb{G}$ of order q for a λ-bit prime q and a generator $\mathcal{P}$ of $\mathbb{G}$.

2.1 Multilinear Maps

In the case of groups with a bilinear map, or more generally with a k-linear map for some $k \geq 2$, we consider a generator producing the tuple $(e_k, \mathbb{G}_1, \mathbb{G}_k, q, \mathcal{P}_1, \mathcal{P}_k)$, where $\mathbb{G}_1, \mathbb{G}_k$ are cyclic groups of prime-order q, $\mathcal{P}_i$ is a generator of $\mathbb{G}_i$ and e is a non-degenerate efficiently computable k-linear map $e_k : \mathbb{G}_1^k \to \mathbb{G}_k$, such that $e_k(\mathcal{P}_1, \ldots, \mathcal{P}_1) = \mathcal{P}_k$.

Multilinear groups (*i.e.*, groups with a k-linear map for $k > 2$) have been considered in a number of works both for functionality and for security reasons. However, the only known constructions of multilinear maps actually offer a reacher functionality: the graded encodings, which allow the computation of 'intermediate' results in the evaluation of the k-linear map. As we will use multilinearity for security reasons, we prefer to consider the reacher possible structure in the security model, then giving more power to a potential adversary.

For any fixed $k \geq 1$, let MGen_k be a PPT algorithm that on input 1^λ returns a description of a graded encoding $\mathcal{MG}_k = (e, \mathbb{G}_1, \ldots, \mathbb{G}_k, q, \mathcal{P}_1, \ldots, \mathcal{P}_k)$, where $\mathbb{G}_1, \ldots, \mathbb{G}_k$ are cyclic groups of prime-order q, $\mathcal{P}_i$ is a generator of $\mathbb{G}_i$ and e is a collection of non-degenerate efficiently computable bilinear maps $e_{i,j} : \mathbb{G}_i \times \mathbb{G}_j \to \mathbb{G}_{i+j}$, for $i + j \leq k$, such that $e(\mathcal{P}_i, \mathcal{P}_j) = \mathcal{P}_{i+j}$. For simplicity we will omit the subindexes of e when they become clear from the context. Sometimes $\mathbb{G}_0$ is used to refer to $\mathbb{Z}_q$. For group elements we use the following implicit notation: for all $i = 1, \ldots, k$, $[a]_i := a\mathcal{P}_i$. The notation extends in a natural way to vectors and matrices and to linear algebra operations. We sometimes drop the index when referring to elements in $\mathbb{G}_1$, *i.e.*, $[a] := [a]_1 = a\mathcal{P}_1$. In particular, it holds that $e([a]_i, [b]_j) = [ab]_{i+j}$.

Additionally, for the asymmetric case, let AGen_2 be a PPT algorithm that on input 1^λ returns a description of an asymmetric bilinear group $\mathcal{AG}_2 = (e, \mathbb{G}, \mathbb{H}, \mathbb{T}, q, \mathcal{P}, \mathcal{Q})$, where $\mathbb{G}, \mathbb{H}, \mathbb{T}$ are cyclic groups of prime-order q, $\mathcal{P}$ is a generator of $\mathbb{G}$, $\mathcal{Q}$ is a generator of $\mathbb{H}$ and $e : \mathbb{G} \times \mathbb{H} \to \mathbb{T}$ is a non-degenerate, efficiently computable bilinear map. In this case we refer to group elements as: $[a]_G := a\mathcal{P}$, $[a]_H := a\mathcal{Q}$ and $[a]_T := e(\mathcal{P}, \mathcal{Q})^a$.

2.2 A Generic Model For Groups With Graded Encodings

In this section we describe a (purely algebraic) generic model for the graded encodings in order to obtain meaningful results about the hardness and separations of computational problems. The model is a natural

extension of Maurer's generic group model [26,27] including the k-graded encodings, but in a completely algebraic formulation that follows the ideas in [6,13,18].

As we will handle elements in different groups, we will use the convenient vector notation $[\boldsymbol{x}]_{\boldsymbol{i}}$ for the group elements $[x_1]_{i_1}, \ldots, [x_n]_{i_n}$. We will use a tilde for the variables containing non-group elements (*i.e.*, elements not in any $\mathbb{G}_i$, $i = 1, \ldots, k$). In a first approach we consider Maurer's model augmented with the graded encodings, but still not phrased in a purely algebraic language. In this model, an algorithm $\mathcal{A}$ does not deal with proper group elements in $\mathbb{G}_i$, but only with labels (Y, i), and it has access to an additional oracle internally performing the group operations. Namely, on start $\mathcal{A}$ receives (in addition to some non-group elements $\widetilde{x}$) the labels $(X_1, i_1), \ldots, (X_n, i_n)$, corresponding to the group elements in the input given to $\mathcal{A}$, $[\boldsymbol{x}]_{\boldsymbol{i}}$, along with two additional labels $(0, i), (1, i)$ for the neutral element and the generator of each group $\mathbb{G}_i$. We will assume that these additional elements are implicitly given to all algorithms. Then $\mathcal{A}$ can adaptively make the following queries to an oracle implementing the k-graded encodings:

- $\mathsf{GroupOp}((Y_1, i), (Y_2, i))$: group operation in $\mathbb{G}_i$ for two previously issued labels in $\mathbb{G}_i$ resulting in a new label (Y_3, i) in $\mathbb{G}_i$.
- $\mathsf{GroupInv}((Y_1, i))$: idem. for group inversion in $\mathbb{G}_i$.
- $\mathsf{GroupPair}((Y_1, i), (Y_2, j))$: bilinear map for two previously issued labels in $\mathbb{G}_i$ and $\mathbb{G}_j$, $i + j \leq k$, resulting in a new label $(Y_3, i + j)$ in $\mathbb{G}_{i+j}$.
- $\mathsf{GroupEqTest}((Y_1, i), (Y_2, i))$: test two previously issued labels in $\mathbb{G}_i$ for equality of the corresponding group elements, resulting in a bit (1 indicates equality).

Every badly formed query (for instance, containing an unknown label) is answered with a special rejection symbol $\perp$. Following the usual step in generic group model proofs (see for instance [6,12,18]), we use polynomials as labels to group elements. Namely, labels in $\mathbb{G}_i$ are polynomials of degree $\leq i$ in $\mathbb{Z}_q[\boldsymbol{X}]$, where the algebraic variables $\boldsymbol{X} = (X_1, \ldots, X_n)$ are just formal representations of the group elements in the input of $\mathcal{A}$. Now in the oracle side, group operations are replaced by polynomial operations in the labels. The group elements $[\boldsymbol{y}]_{\boldsymbol{j}}$ in the output of $\mathcal{A}$ are now replaced by labels $(Y_1, j_1), \ldots, (Y_m, j_m)$ given at some time by the generic group oracle. Therefore, for any fixed random tape of $\mathcal{A}$ and any choice of $\widetilde{x}$, there exist polynomials $Y_1, \ldots, Y_m \in \mathbb{Z}_q[\boldsymbol{X}]$ of degrees upper bounded by $j_1, \ldots, j_m$[5] respectively, with coefficients known to $\mathcal{A}$.

Notice that the algorithm itself can predict all answers given by the oracle except for some equality test queries. Indeed, some equality test queries trivially outputs 1, due to the group structure itself (*e.g.*, the labels of $[0]$, $[x][-x]$ and $[x]^q$ are all equivalent). Nontrivial test queries (*i.e.*, equality test queries resulting in equality for two labels that are not 'structurally' equal) depend on the *a priori* constraints in the input group elements, that is the definition of the problem instance solved by $\mathcal{A}$. All the information $\mathcal{A}$ can obtain from the generic group oracle is via the nontrivial equality test queries.

We now introduce a "purely algebraic" version of the generic model by replacing the test oracle with a trivial one, answering 1 if and only if it is queried with two identical labels (polynomials). With this replacement the behavior of $\mathcal{A}$ can only differ negligibly from the original, assuming that the distribution of $\boldsymbol{x}$ can be sampled by evaluating polynomial functions of constant degree at a random point.[6] As usually, the proposed generic model reduces the analysis of the hardness of some problems to solving a merely algebraic problem related to polynomials. In particular, consider a computational problem $\mathcal{P}$ which instances are entirely described by some group elements in the base group $\mathbb{G}_1$ $[\boldsymbol{x}] \leftarrow \mathcal{P}.\mathsf{InstGen}(1^\lambda)$ and its solutions are also described by some group elements $[\boldsymbol{y}]_{\boldsymbol{j}} \in \mathcal{P}.\mathsf{Sol}([\boldsymbol{x}])$. $\mathcal{P}$ is hard in the purely algebraic generic multilinear group model if and only if for all (randomized) polynomials $Y_1, \ldots, Y_m \in \mathbb{Z}_q[\boldsymbol{X}]$ of degrees upper bounded by $j_1, \ldots, j_m$ respectively,

$$\Pr([\boldsymbol{y}]_{\boldsymbol{j}} \in \mathcal{P}.\mathsf{Sol}([\boldsymbol{x}]) : \ [\boldsymbol{x}] \leftarrow \mathcal{P}.\mathsf{InstGen}(1^\lambda), \boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{x})) \in \mathit{negl}(\lambda)$$

[5] The upper bounds on the degrees come from the fact that all input group elements are in $\mathbb{G}_1$, and the only way to build elements in $\mathbb{G}_i$ is by using the bilinear maps or the "fixed" elements (the neutral elements and the generators).

[6] As a standard argument used in proofs in the generic group model, the difference between the original model and its purely algebraic reformulation amounts to a negligible probability, which is typically upper-bounded by using Schwartz-Zippel Lemma and the union bound, as shown for instance in [6,13,18].

where $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$ and the probability is computed with respect the random coins of the instance generator and the randomized polynomials.[7]

On the other hand, from the above discussion we can state the following lemma.

Lemma 1. *Let $\mathcal{A}$ be an algorithm in the (purely algebraic) generic multilinear group model. Let $([\boldsymbol{x}]_{\boldsymbol{i}}, \widetilde{x})$ and $([\boldsymbol{y}]_{\boldsymbol{j}}, \widetilde{y})$ respectively be the input and output of $\mathcal{A}$. Then, for every choice of $\widetilde{x}$ and any choice of the random tape of $\mathcal{A}$, there exist polynomials $Y_1, \ldots, Y_m \in \mathbb{Z}_q[\boldsymbol{X}]$ of degree upper bounded by $j_1, \ldots, j_m$ such that $\boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{x})$, for all possible $\boldsymbol{x} \in \mathbb{Z}_q^n$, where $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$. Moreover, $\widetilde{y}$ does not depend on $\boldsymbol{x}$.*

The previous model extends naturally to algorithms with oracle access (*e.g.*, black-box reductions) but only when the oracles fit well into the generic model. Let us consider the algorithm $\mathcal{A}^{\mathcal{O}}$, with oracle access to $\mathcal{O}$. A completely arbitrary oracle (specified in the plain model) could have access to the internal representation of the group elements, and then it could leak some information that is outside the generic group model. Thus, we will impose the very limiting constraint that the oracles are also "algebraic", meaning that the oracle's input/output behaviour respects the one-wayness of the graded encodings, it only performs polynomial operations on the input labels, but the constraint on the degrees of the polynomials is removed.

Definition 1. *Let $([\boldsymbol{u}]_{\boldsymbol{d}}, \widetilde{u})$ and $([\boldsymbol{v}]_{\boldsymbol{e}}, \widetilde{v})$ respectively be a query to an oracle $\mathcal{O}$ and its corresponding answer, where $\widetilde{u}$ and $\widetilde{v}$ contain the respective non-group elements. The oracle $\mathcal{O}$ is called algebraic if for any choice of $\widetilde{u}$ there exist polynomials $V_1, \ldots, V_\beta \in \mathbb{Z}_q[\boldsymbol{U}, \boldsymbol{R}]$, $\boldsymbol{R} = (R_1, \ldots, R_\tau)$, of constant degree (in the security parameter) such that*

- *for the specific choice of $\widetilde{u}$, $v_k = V_k(\boldsymbol{u}, \boldsymbol{r})$, $k = 1, \ldots, \beta$, for all $\boldsymbol{u} \in \mathbb{Z}_q^\alpha$ and $\boldsymbol{r} \in \mathbb{Z}_q^\tau$, where $\boldsymbol{r} = (r_1, \ldots, r_\tau)$ are parameters internally defined by the oracle,*
- *V_k does not depend on any U_l such that $e_k < d_l$ (in order to preserve the one-wayness of the graded encodings),*
- *$\widetilde{v}$ does not depend on $\boldsymbol{u}, \boldsymbol{r}$ (thus, $\boldsymbol{r}$ can only have influence in the group elements in the answer),*
- *the sets of parameters $\boldsymbol{r}$ corresponding to different oracle calls can be independent or not, depending on whether the oracle is stateless or stateful.*

Although this notion looks very limiting (*e.g.*, it excludes a Discrete Logarithm oracle, as it destroys the one-wayness property of the graded encodings, but oracles solving CDH or the Bilinear Computational Diffie-Hellman problem fit well in the definition), it is general enough for our purposes[8]. The parameters $\boldsymbol{r}$ capture the behaviour of an oracle solving a problem with many solutions (called here a "flexible" problem).

We will need the following generalization of the previous lemma. Observe that we loose the control of the degree of the polynomials due to the interaction with the algebraic oracle.

Lemma 2. *Let $\mathcal{A}^{\mathcal{O}}$ be an oracle algorithm in the (purely algebraic) generic multilinear group model, making a bounded number of calls Q to an algebraic oracle $\mathcal{O}$. Let $[\boldsymbol{x}]_{\boldsymbol{i}}$, $[\boldsymbol{y}]_{\boldsymbol{j}}$, $\widetilde{x}$ and $\widetilde{y}$ defined as in the previous lemma. Then, for every choice of $\widetilde{x}$ and the random tape, there exist polynomials of constant degree $Y_1, \ldots, Y_m \in \mathbb{Z}_q[\boldsymbol{X}, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_Q]$, such that $\boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{x}, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_Q)$, for all possible inputs, where $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$, and $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_Q$ are the parameters introduced in Definition 1 for the Q queries. Moreover, $\widetilde{y}$ does not depend on $\boldsymbol{x}$ or $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_Q$.*

Proof. We proceed by induction in Q. The first step, $Q = 0$, follows immediately from Lemma 1, because $\mathcal{A}^{\mathcal{O}}$ is just an algorithm (without oracle access). For $Q \geq 1$, we split $\mathcal{A}^{\mathcal{O}}$ into two sections $\mathcal{A}_0^{\mathcal{O}}$ and $\mathcal{A}_1$, separated exactly at the last query point (see Figure 2). Let $([\boldsymbol{z}]_{\boldsymbol{\gamma}}, \widetilde{z})$ be the state information (group and non-group elements) that $\mathcal{A}_0^{\mathcal{O}}$ passes to $\mathcal{A}_1$, $([\boldsymbol{u}]_{\boldsymbol{\alpha}}, \widetilde{u})$ be the Q-th query to $\mathcal{O}$, and $([\boldsymbol{v}]_{\boldsymbol{\beta}}, \widetilde{v})$ be its corresponding answer. We assume that $\mathcal{A}_0^{\mathcal{O}}$ and $\mathcal{A}_1$ receive the same random tape, \$, (perhaps introducing some redundant computations in $\mathcal{A}_1$). Observe that the output of $\mathcal{A}_0^{\mathcal{O}}$ consists of $([\boldsymbol{z}]_{\boldsymbol{\gamma}}, \widetilde{z})$ and $([\boldsymbol{u}]_{\boldsymbol{\alpha}}, \widetilde{u})$.

[7] We can similarly deal with problems with non-group elements both in the instance description and the solution, but this would require a more sophisticated formalization, in which both the polynomials and the non-group elements in the solution could depend on the non-group elements in the instance, but in an efficient way.

[8] This model can be easily extended in a number of ways, but it would unnecessarily obfuscate the exposition.

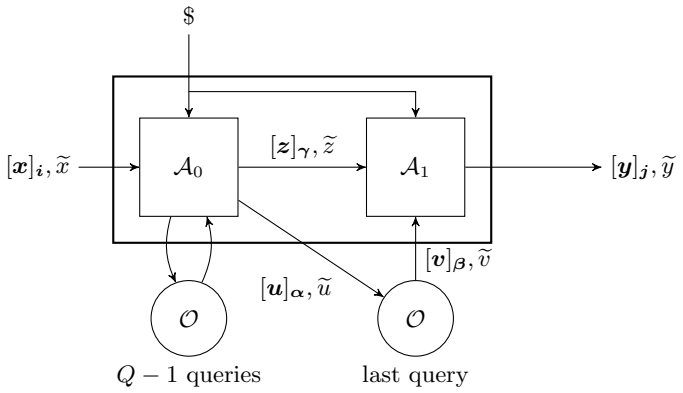

Fig. 2. Splitting of the oracle algorithm in Lemma 2.

By the induction assumption, for any choice of $\widetilde{x}$ and \$, there exist some polynomials of constant degree $Z_1, \ldots, Z_\gamma \in \mathbb{Z}_q[\boldsymbol{X}, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_{Q-1}]$ and $U_1, \ldots, U_\alpha \in \mathbb{Z}_q[\boldsymbol{X}, \boldsymbol{R}_1, \ldots, \boldsymbol{R}_{Q-1}]$ such that $\boldsymbol{z} = \boldsymbol{Z}(\boldsymbol{x}, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{Q-1})$, where $\boldsymbol{Z} = (Z_1, \ldots, Z_\gamma)$, and $\boldsymbol{u} = \boldsymbol{U}(\boldsymbol{x}, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{Q-1})$, where $\boldsymbol{U} = (U_1, \ldots, U_\alpha)$, for all possible $\boldsymbol{x} \in \mathbb{Z}_q^n$ and $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{Q-1} \in \mathbb{Z}_q^\tau$. Moreover, $\widetilde{z}$ and $\widetilde{u}$ only depend on $\widetilde{x}$ and \$.

Now, the algorithm $\mathcal{A}_1$ receives as input $([\boldsymbol{z}]_\gamma, \widetilde{z})$ and $([\boldsymbol{v}]_{\boldsymbol{\beta}}, \widetilde{v})$. By Definition 1, $[\boldsymbol{v}]_{\boldsymbol{e}}$ also depend polynomially on $\boldsymbol{u}$ and $\boldsymbol{r}_Q$. Namely, for every choice of $\widetilde{u}$, there exist polynomials of constant degree $V_1, \ldots, V_\beta \in \mathbb{Z}_q[\boldsymbol{U}, \boldsymbol{R}_Q]$ such that $\boldsymbol{v} = \boldsymbol{V}(\boldsymbol{u}, \boldsymbol{r}_Q)$, where $\boldsymbol{V} = (V_1, \ldots, V_\beta)$, while $\widetilde{v}$ only depends on $\widetilde{u}$.

Since $\mathcal{A}_1$ is just an algorithm without oracle access, by Lemma 1, for any choice of $\widetilde{v}$, $\widetilde{z}$ and \$, there exist polynomials of constant degree $Y_1, \ldots, Y_m \in \mathbb{Z}_q[\boldsymbol{V}, \boldsymbol{Z}]$ such that $\boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{v}, \boldsymbol{z})$, where $\boldsymbol{Y} = (Y_1, \ldots, Y_m)$, for all $\boldsymbol{v} \in \mathbb{Z}_q^\beta$ and $\boldsymbol{z} \in \mathbb{Z}_q^\gamma$, while $\widetilde{y}$ only depends on $\widetilde{v}$, $\widetilde{z}$ and \$.

By composition of all the previous polynomials, we show that $\boldsymbol{y}$ depend polynomially on $\boldsymbol{x}$ and $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_Q$, where the polynomials depend only on \$ and $\widetilde{x}$. Indeed

$$\boldsymbol{y} = \boldsymbol{Y}(\boldsymbol{V}(\boldsymbol{U}(\boldsymbol{x}, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{Q-1}), \boldsymbol{r}_Q), \boldsymbol{Z}(\boldsymbol{x}, \boldsymbol{r_1}, \ldots, \boldsymbol{r_{Q-1}}))$$

and all the polynomials involved depend only on $\widetilde{x}$, $\widetilde{z}$, $\widetilde{u}$, $\widetilde{v}$ and \$, but all in turn only depend on $\widetilde{x}$ and \$. In addition, for the same reason, $\widetilde{y}$ only can depend on $\widetilde{x}$ and \$, which concludes the proof.

2.3 The Matrix Decisional Diffie-Hellman Assumption

We recall here the definition of the decisional assumptions introduced in [12], which are the starting point of our flexible computational matrix problems.

Definition 2. *[12], Let $\ell, k \in \mathbb{N}$ with $\ell > k$. We call $\mathcal{D}_{\ell,k}$ a matrix distribution if it outputs (in polynomial time, with overwhelming probability) matrices in $\mathbb{Z}_q^{\ell \times k}$ of full rank k. We denote $\mathcal{D}_k := \mathcal{D}_{k+1,k}$.*

Definition 3 ($\mathcal{D}_{\ell,k}$-MDDH Assumption). *[12] Let $\mathcal{D}_{\ell,k}$ be a matrix distribution. The $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-MDDH) Problem is telling apart the two probability distributions $(\mathbb{G}, q, \mathcal{P}, [\mathbf{A}], [\mathbf{A}\boldsymbol{w}])$ and $(\mathbb{G}, q, \mathcal{P}, [\mathbf{A}], [\boldsymbol{z}])$, where $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_q^k, \boldsymbol{z} \leftarrow \mathbb{Z}_q^\ell$.*

We say that the $\mathcal{D}_{\ell,k}$-Matrix Diffie-Hellman ($\mathcal{D}_{\ell,k}$-MDDH) Assumption holds relative to Gen the corresponding problem is hard, that is, if for all PPT adversaries $\mathcal{A}$,

$$\mathbf{Adv}_{\mathcal{D}_{\ell,k},\mathsf{Gen}}(\mathcal{A}) = \Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\mathbf{A}\boldsymbol{w}]) = 1] - \Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\boldsymbol{z}]) = 1] \in \mathit{negl}(\lambda),$$

where the probability is taken over $\mathcal{G} = (\mathbb{G}, q, \mathcal{P}) \leftarrow \mathsf{Gen}(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_q^k, \boldsymbol{z} \leftarrow \mathbb{Z}_q^\ell$ and the coin tosses of adversary $\mathcal{A}$.

In the case of symmetric k-linear groups, we similarly say that the $\mathcal{D}_{\ell,k}$-MDDH Assumption holds relative to MGen_k when

$$\mathbf{Adv}_{\mathcal{D}_{\ell,k},\mathsf{MGen}_k}(\mathcal{A}) = \Pr[\mathcal{A}(\mathcal{MG}_k, [\mathbf{A}]_1, [\mathbf{A}\boldsymbol{w}]_1) = 1] - \Pr[\mathcal{A}(\mathcal{MG}_k, [\mathbf{A}]_1, [\boldsymbol{z}]_1) = 1] \in \mathit{negl}(\lambda),$$

where the probability is taken over $\mathcal{MG}_k = (e, \mathbb{G}_1, \ldots, \mathbb{G}_k, q, \mathcal{P}_1, \ldots, \mathcal{P}_k) \leftarrow \mathsf{MGen}_k(1^\lambda)$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}, \boldsymbol{w} \leftarrow \mathbb{Z}_q^k, \boldsymbol{z} \leftarrow \mathbb{Z}_q^\ell$ and the coin tosses of adversary $\mathcal{A}$. The asymmetric case is dealt with in the same way. We will say that the $\mathcal{D}_{\ell,k}$-MDDH Assumption holds relative to AGen_2 in the left (resp. in the right) when $\mathcal{A}$ is given as input the output $\mathcal{AG}_2 = (e, \mathbb{G}, \mathbb{H}, \mathbb{T}, q, \mathcal{P}, \mathcal{Q})$ of AGen_2 along with the matrix $\mathbf{A}$ and the vector $\mathbf{A}\boldsymbol{w}$ or $\boldsymbol{z}$, both encoded in $\mathbb{G}$ (resp. in $\mathbb{H}$).

The following definition just aims to simplifying some of the statements in the paper.

Definition 4. *A matrix distribution $\mathcal{D}_{\ell,k}$ is hard if the corresponding $\mathcal{D}_{\ell,k}$-MDDH problem is hard in the generic k-linear group model.*

Some particular families of matrix distributions were presented in [12]. Namely,

$$\mathcal{SC}_k : \mathbf{A} = \begin{pmatrix} a & & 0 \\ 1 & \ddots & \\ & \ddots & a \\ 0 & & 1 \end{pmatrix} \quad \mathcal{C}_k : \mathbf{A} = \begin{pmatrix} a_1 & & 0 \\ 1 & \ddots & \\ & \ddots & a_k \\ 0 & & 1 \end{pmatrix} \quad \mathcal{L}_k : \mathbf{A} = \begin{pmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_k \\ 1 & \cdots & 1 \end{pmatrix},$$

where $a, a_i \leftarrow \mathbb{Z}_p$, and $\mathcal{U}_{\ell,k}$ which is simply the uniform distribution in $\mathbb{Z}_p^{\ell\times k}$. The $\mathcal{SC}_k$-MDDH Assumption is the Symmetric Cascade Assumption, the $\mathcal{C}_k$-MDDH Assumption is the Cascade Assumption, which were proposed for the first time. $\mathcal{U}_{\ell,k}$-MDDH and $\mathcal{L}_k$-MDDH were implicitly used in some previous works. Actually, $\mathcal{L}_k$-MDDH is the Decisional Linear Assumption in [8]. For instance, we can consider the case $k = 2$, in which the $\mathcal{L}_2$-MDDH problem is given $([1], [a_1], [a_2])$, tell apart the two distributions $([1], [a_1], [a_2], [w_1 a_1], [w_2 a_2], [w_1 + w_2])$ and $([1], [a_1], [a_2], [z_1], [z_2], [z_3])$, where $a_1, a_2, w_1, w_2, z_1, z_2, z_3$ are random. This is exactly the 2-Lin Problem, since we can always set $z_1 = w_1 a_1$ and $z_2 = w_2 a_2$.

We also give examples of matrix distributions which did not appear in [12] but that are implicitly used in 2 and 4. The Randomized Linear and the Square Polynomial distributions are respectively given by the matrices

$$\mathcal{RL}_k : \mathbf{A} = \begin{pmatrix} a_1 & & 0 \\ & \ddots & \\ 0 & & a_k \\ b_1 & \cdots & b_k \end{pmatrix} \qquad \mathcal{P}_{\ell,2} : \mathbf{A} = \begin{pmatrix} a_1 & a_1^2 \\ a_2 & a_2^2 \\ \vdots & \vdots \\ a_\ell & a_\ell^2 \end{pmatrix}$$

where $a_i \leftarrow \mathbb{Z}_q$ and $b_i \leftarrow \mathbb{Z}_q^\times$. Jutla and Roy [20] referred to $\mathcal{RL}_k$-MDDH Assumption as the k-lifted Assumption. From the results in Section 4.2 it is easy to see that $\mathcal{RL}_k$ is a hard matrix distribution.[9]

3 The Matrix Diffie-Hellman Computational Problems

In this section we introduce two families of search problems naturally related to the Matrix Decisional Diffie-Hellman problems. Given a matrix distribution, $\mathcal{D}_{\ell,k}$, the first family consists of the problems of given a matrix $[\mathbf{A}]$, where $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$, and the first k components of a vector $[\boldsymbol{z}]$, complete it so that $\boldsymbol{z} \in \operatorname{Im} \mathbf{A}$. The second family consists of the problems of finding $[\boldsymbol{x}]$ such that $\boldsymbol{x} \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$. It is noticeable that some computational problems in the literature are particular cases of this second family.

We next show that any solution of the new search problems is enough to solve the corresponding decisional MDDH problem. Finally, we study the existence of reductions between the kernel problems for the matrix distributions previously given: for different sizes within the same distribution, and also between different distributions with the same size.

[9] The hardness of the $\mathcal{P}_{\ell,2}$ matrix distribution is partially analyzed, under the name of Simultaneous Pairing Assumption, by Groth and Lu [17].

Definition 5 ($\mathcal{D}_{\ell,k}$-MCDH). *Given a matrix distribution $\mathcal{D}_{\ell,k}$ in a group $\mathbb{G}$, such that the upper $k \times k$ submatrix of $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$ has full rank with overwhelming probability, the computational matrix Diffie-Hellman Problem is given $([\mathbf{A}], [\boldsymbol{z}_0])$, with $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$, $\boldsymbol{z}_0 \leftarrow \mathbb{Z}_q^k$, compute $[\boldsymbol{z}_1] \in \mathbb{G}^{\ell-k}$ such that $(\boldsymbol{z}_0 \| \boldsymbol{z}_1) \in Im\, \mathbf{A}$.*

The full-rank condition ensures the existence of solutions to the $\mathcal{D}_{\ell,k}$-MCDH problem instance, and then we tolerate the existence of a negligible fraction of the problem instances that are unsolvable. Indeed, all known interesting matrix distributions fulfil this requirement with overwhelming probability.

Notice that CDH and the computational k-Lin problems are particular examples of MCDH problems. Namely, CDH is exactly $\mathcal{L}_1$-MCDH and the computational k-Lin problem is $\mathcal{L}_k$-MCDH. Indeed, the $\mathcal{L}_1$-MCDH problem is given $[1], [a], [z_1]$, compute $[z_2]$ such that (z_1, z_2) is collinear with $(1, a)$, or equivalently, $z_2 = z_1 a$, which is solving the CDH problem.

All MCDH problems have a unique solution and they appear naturally in some scenarios using MDDH problems. For instance, the one-wayness of the encryption scheme in [12] is equivalent to the corresponding MCDH assumption. However, any MCDH problem amounts to computing some polynomial on the elements of $\mathbf{A}$ and it is equivalent to CDH ([4,19]), although the tightness of the reduction depends on the degree of the polynomial.

The second family is more interesting. It is a family of flexible problems. Flexible computational problems are the natural way to model the adversarial capability in some scenarios like unforgeability, and finding reductions between flexible problems is not an obvious task. This new problem family is closely related to the various flavors of "simultaneous pairing" assumptions in the literature.

Definition 6 ($\mathcal{D}_{\ell,k}$-KerMDH). *Given a matrix distribution $\mathcal{D}_{\ell,k}$ in a group $\mathbb{G}$, the Kernel Diffie-Hellman Problem is given $[\mathbf{A}]$, with $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$, find a nonzero vector $[\boldsymbol{x}] \in \mathbb{G}^{\ell}$ such that $\boldsymbol{x}$ is orthogonal to $Im\, \mathbf{A}$, that is, $\boldsymbol{x} \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$.*

Note that one can efficiently test if a vector $[\boldsymbol{x}]$ is a solution to the problem KerMDH in a bilinear group, by checking whether $e([\boldsymbol{x}^\top], [\mathbf{A}]) = [\mathbf{0}]_2$.

Definition 6 naturally extends to asymmetric bilinear groups. There, given $[\mathbf{A}]_H$, the problem is to find $[\boldsymbol{x}]_G$ such that $\boldsymbol{x} \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$. A solution can be obviously verified by checking if $e([\boldsymbol{x}^\top]_G, [\mathbf{A}]_H) = [\mathbf{0}]_T$. We can also consider an extension of this problem in which the goal is to solve the same problem but giving the solution in a different group $\mathbb{G}_r$, in some ideal graded encoding $\mathcal{MG}_m$, for some $0 \leq r \leq \min(m, k-1)$. The case $r = 1$ corresponds to the previous problem defined in a m-linear group.

Definition 7 ($(r, m, \mathcal{D}_{\ell,k})$-KerMDH). *Given a matrix distribution $\mathcal{D}_{\ell,k}$ over a m-linear group $\mathcal{MG}_m$ and r an integer $0 \leq r \leq \min(m, k-1)$, the $(r, m, \mathcal{D}_{\ell,k})$-KerMDH Problem is to find $[\boldsymbol{x}]_r \in \mathbb{G}_r^{\ell}$ such that $\boldsymbol{x} \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$.*

When the precise degree of multilinearity m is not an issue, we will write $(r, \mathcal{D}_{\ell,k})$-KerMDH instead of $(r, m, \mathcal{D}_{\ell,k})$-KerMDH, for any $m \geq r$. Again, we note that if $m \geq r+1$ one can efficiently test if a vector $[\boldsymbol{x}]_r$ solves the $(r, m, \mathcal{D}_{\ell,k})$-KerMDH by checking whether $e([\boldsymbol{x}^\top]_r, [\mathbf{A}]) = [\mathbf{0}]_{r+1}$. However, if $m = r$ the solution of the problem cannot be checked as it would require the use of a $(m+1)$-linear map. Notice that we do not consider the case $r \geq k$ because it makes the problem easy.

Lemma 3. *The $(r, m, \mathcal{D}_{\ell,k})$-KerMDH Problem for $k \leq r \leq m$ is easy.*

Proof. If $\ell = k+1$, a solution to the problem is the vector $[(A_1, -A_2, \ldots, (-1)^k A_{k+1})]_r$ where A_i is the minor of $\mathbf{A}$ obtained by deleting the i-th row, computed by means of the m-linear map, as $m \geq r$. In the case $\ell > k+1$ the solution can be obtained with a similar trick applied to any full-rank $(k+1) \times k$ submatrix of $\mathbf{A}$.

3.1 Decisional vs. Computational Matrix Problems

In this section we detail the relation between the new search problems given in definitions 5 and 6 and the Matrix Decisional Diffie-Hellman Problems. Specifically, any solution to the computational problems allows to tell apart the real and the random instances of the corresponding MDDH problem.

The first lemma states the obvious relation between MCDH and MDDH.

Lemma 4. *In a k-linear group,* $\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow$ $\mathcal{D}_{\ell,k}$-MCDH.

The kernel problem is also harder than the corresponding decisional problem, in multilinear groups.

Lemma 5. *In a m-linear group with* $m \geq 2$, $\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow$ $\mathcal{D}_{\ell,k}$-KerMDH.

Proof. Given an instance of the $\mathcal{D}_{\ell,k}$-MDDH problem $([\mathbf{A}], [\boldsymbol{z}])$, a solution to test membership in $\operatorname{Im}\mathbf{A}$ is simply checking whether $e([\boldsymbol{x}^\top], [\boldsymbol{z}]) = [\boldsymbol{x}^\top \boldsymbol{z}]_2 \stackrel{?}{=} [0]_2$, where $[\boldsymbol{x}]$ is the output of the $\mathcal{D}_{\ell,k}$-KerMDH solver on input $[\mathbf{A}]$. For a real instance of $\mathcal{D}_{\ell,k}$-MDDH (*i.e*, $\boldsymbol{z} \in \operatorname{Im}\mathbf{A}$) the solver gives always the correct answer. Furthermore, if the instance is random (*i.e.*, $\boldsymbol{z}$ is a random vector) then $\boldsymbol{x}^\top \boldsymbol{z} = 0$ occurs only with a negligible probability $1/q$. Therefore, the reduction works fine with overwhelming probability.

Analogously, we have

Lemma 6. *In a m-linear group,* $\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow$ $(r, m, \mathcal{D}_{\ell,k})$-KerMDH *for any* $0 \leq r \leq m-1$.

3.2 The Kernel DH Assumptions in the Multilinear Maps Candidates

We have shown that for any hard matrix distribution $\mathcal{D}_{\ell,k}$ the $\mathcal{D}_{\ell,k}$-KerMDH problem is generically hard in m-linear groups. However, in the only candidate multilinear groups which have resisted cryptanalysis [14], every $\mathcal{D}_{\ell,k}$-MDDH Assumption is false (see [14], full version, Section 4.4). Indeed, every MDDH problem amounts to deciding whether a matrix has full rank or not.

Roughly speaking, in an m-linear group $\mathcal{MG}_m$, given a matrix $[\mathbf{A}]_r$, $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$, $0 \leq r \leq m-1$ and the zero-test element given in the multilinear group description (which allows to decide if two elements in $\mathbb{G}_m$ encode the same element of $\mathbb{Z}_q$), one can compute a matrix $\mathbf{A}'$ of "weak discrete logarithms", *i.e.*, a noisy encoding at level 0 of the matrix $\mathbf{A}$. The attack uses that $\mathbf{A}'$ has full rank if and only if $\mathbf{A}$ has. Observe that the rank of $\mathbf{A}'$ can be efficiently computed using standard linear algebra at level 0, without using multilinear maps (so independently of ℓ, k).

However, this attack does not apply in a straightforward way to break the KerMDH Assumption. With the matrix $\mathbf{A}'$ one can easily compute a nonzero vector $\boldsymbol{v}' \in \ker \mathbf{A}'^\top$, which is a "noisy version" of a vector in $\ker \mathbf{A}^\top$, but it is not in $\mathbb{G}_i$ for any $i = 1, \ldots, m$. Thus, this attack does not apply directly to break the $(r, m, \mathcal{D}_{\ell,k})$-KerMDH. We leave it as an open question to examine this issue further.

3.3 A Unifying View on Computational Matrix Problems

In this section we recall some computational problems in the cryptographic literature that we unify as particular instances of KerMDH problems. These problems are listed below, as they appear in the cited references. In the following, all parameters a_i and b_i are assumed to be randomly chosen in $\mathbb{Z}_q$.

1. Find-Rep [9]: Given $([a_1], \ldots, [a_\ell])$, find a nonzero tuple $(x_1, \ldots, x_\ell)$ such that $x_1 a_1 + \ldots + a_\ell x_\ell = 0$.
2. Simultaneous Double Pairing (SDP) [2]: Given the two tuples, $([a_1], [b_1])$ and $([a_2], [b_2])$, find a nonzero tuple $([x_1], [x_2], [x_3])$ such that $x_1 b_1 + x_2 a_1 = 0$, $x_1 b_2 + x_3 a_2 = 0$.
3. Simultaneous Triple Pairing [16]: Given the two tuples, $([a_1], [a_2], [a_3])$ and $([b_1], [b_2], [b_3])$, find a nonzero tuple $([x_1], [x_2], [x_3])$ such that $x_1 a_1 + x_2 a_2 + x_3 a_3 = 0$, $x_1 b_1 + x_2 b_2 + x_3 b_3 = 0$.
4. Simultaneous Pairing [17]: Given $([a_1], [a_2], \ldots, [a_\ell])$ and $([a_1^2], [a_2^2], \ldots, [a_\ell^2])$, find a nonzero tuple $([x_1], \ldots, [x_\ell])$ such that $\sum_{i=1}^{\ell} x_i a_i = 0$, $\sum_{i=1}^{\ell} x_i a_i^2 = 0$.
5. 1-Flexible Diffie-Hellman (1-FlexDH) [25]: Given $([1], [a], [b])$, find a triple $([r], [ra], [rab])$ with $r \neq 0$.
6. 1-Flexible Square Diffie-Hellman (1-FlexSDH) [23]: Given $([1], [a])$, find a triple $([r], [ra], [ra^2])$ with $r \neq 0$.
7. ℓ-Flexible Diffie-Hellman (ℓ-FlexDH) [25]: Given $([1], [a], [b])$, find a $(2\ell+1)$-tuple $([r_1], \ldots, [r_\ell], [r_1 a], [r_1 r_2 a], \ldots, [(\prod_{i=1}^{\ell} r_i)a], [(\prod_{i=1}^{\ell} r_i)ab])$ such that $r_j \neq 0$ for all $j = 1, \ldots, \ell$.
8. Double Pairing (DP) [16]: In an asymmetric group $(\mathbb{G}, \mathbb{H}, \mathbb{T})$, given a pair of random elements $([a_1]_H, [a_2]_H) \in \mathbb{H}^2$, find a nonzero tuple $([x_1]_G, [x_2]_G)$ such that $[x_1 a_1 + x_2 a_2]_T = [0]_T$.

Notice that Find-Rep is just $(0,\mathcal{U}_{\ell,1})$-KerMDH, SDP is $\mathcal{RL}_2$-KerMDH, the Simultaneous Triple Pairing problem is $\mathcal{U}_2$-KerMDH, the Simultaneous Pairing problem is $\mathcal{P}_{\ell,2}$-KerMDH. DP corresponds to $\mathcal{U}_1$-KerMDH in an asymmetric bilinear setting. On the other hand, 1-FlexDH is $\mathcal{C}_2$-KerMDH, 1-FlexSDH problem is $\mathcal{SC}_2$-KerMDH and ℓ-FlexDH for $\ell > 1$ is the only one which is not in the KerMDH problem family. However, ℓ-FlexDH $\Rightarrow \mathcal{C}_{\ell+1}$-KerMDH. Getting the last three results require a bit more work, as we show in the next two lemmas.

Lemma 7. *1-FlexDH* $= \mathcal{C}_2$-KerMDH *and 1-FlexSDH* $= \mathcal{SC}_2$-KerMDH.

Proof. The proof of the first statement is obvious from the fact that the solutions $([r],[ra],[rab])$ of the 1-FlexDH problem instance $([1],[a],[b])$ correspond exactly to the nonzero vectors in $\ker \mathbf{A}^\top$ for $\mathbf{A} = \begin{pmatrix} -a & 0 \\ 1 & -b \\ 0 & 1 \end{pmatrix}$. The second statement is proven in a similar way.

Lemma 8. *ℓ-FlexDH* $\Rightarrow \mathcal{C}_{\ell+1}$-KerMDH.

Proof. Given a ℓ-FlexDH problem instance $([1],[a],[b])$, pick random $r_2,\ldots,r_\ell \in \mathbb{Z}_q^*$ and compute the matrix $[\mathbf{A}]$ where

$$\mathbf{A} = \begin{pmatrix} -a & & & & 0 \\ 1 & -r_2 & & & \\ & \ddots & \ddots & & \\ & & 1 & -r_\ell & \\ & & & 1 & -b \\ 0 & & & & 1 \end{pmatrix}$$

Then, run the $\mathcal{C}_{\ell+1}$-KerMDH solver on $[\mathbf{A}]$, obtaining the vector $([r_1],[r_1a],[r_1r_2a],\ldots,[r_1\cdots r_\ell a],[r_1\cdots r_\ell ab])$ for some $r_1 \in \mathbb{Z}_q^*$, which along with $([r_2],\ldots,[r_\ell])$ solves the ℓ-FlexDH problem.

4 Reduction and Separation of Kernel Diffie-Hellman Problems

In this section we prove the following result

Theorem 1. *$\mathcal{U}_{\ell,k}$, $\mathcal{L}_k$, $\mathcal{CI}_{k,d}$, $\mathcal{SC}_k$, $\mathcal{C}_k$ and $\mathcal{RL}_k$ define families of* KerMDH *problems with* ***strictly*** *increasing hardness.*

By 'strictly increasing' we mean that

1. there are known reductions of the smaller problems to the larger problems (in terms of k) within each family,
2. there are no black-box reductions in the other way in the multilinear generic group model.

This result means that it does make sense relying on $\mathcal{D}_{\ell,k}$-KerMDH Assumption for $k > 2$. A similar result is known for the corresponding $\mathcal{D}_{\ell,k}$-MDDH problems. Indeed, one can easily prove a separation between large and small problems. Observe that any efficient m-linear map can efficiently solve any $\mathcal{D}_{\ell,k}$-MDDH problem with $k \leq m-1$, and therefore every two $\mathcal{D}_{\ell,k}$-MDDH and $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-MDDH problems with $\widetilde{k} < k$ are separated by an oracle computing a k-linear map.

However, when dealing with the computational $\mathcal{D}_{\ell,k}$-KerMDH family, no such a trivial argument is known to exist. Actually, an m-linear map does not seem to help to solve any $\mathcal{D}_{\ell,k}$-KerMDH problem with $k > 1$. Furthermore, the m-linear map seems to be useless for any (reasonable) reduction between KerMDH problems defined over the same group. Indeed, all group elements involved in the problem instances and their solutions belong to the base group $\mathbb{G}$, and the result of computing any m-linear map is an element in $\mathbb{G}_m$, where no efficient map from $\mathbb{G}_m$ back to $\mathbb{G}$ is supposed to exist.

4.1 Separation

In this section we firstly show the negative part of Theorem 1. Namely, we show that there is no black-box reduction in the generic group model (described in Section 2.2) from $\mathcal{D}_{\ell,k}$-KerMDH to $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH for $k > \widetilde{k}$, assuming that the two matrix distributions $\mathcal{D}_{\ell,k}$ and $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$ are hard (see Definition 4). Before proving the main result we need some technical lemmas and also a new geometrical notion defined on a family of subspaces of a vector space, named *t-Elusiveness*.

In the first lemma we show that the natural (black-box, algebraic) reductions between KerMDH problems have a very special form. Observe that a black-box reduction to a flexible problem must work for any adversary solving it. In particular, the reduction should work for **any** solution given by this adversary, or for **any** probability distribution of the solutions given by it. Informally, the lemma states that the output of a successful reduction can always be computed in essentially two ways:

- by just applying a (randomized) linear map to the answer given by the adversary in the last call. Therefore, all possibly existing previous calls to the adversary are just used to prepare the last one.
- by just ignoring the last call to the adversary and using only the information gathered in the previous ones.

Let $\mathcal{R}^{\mathcal{O}}$ be a black-box reduction of $\mathcal{D}_{\ell,k}$-KerMDH to $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH, in the purely algebraic generic multilinear group model, discussed in Section 2.2, for some matrix distributions $\mathcal{D}_{\ell,k}$ and $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$. Namely, $\mathcal{R}^{\mathcal{O}}$ solves $\mathcal{D}_{\ell,k}$-KerMDH with a non-negligible probability by making $Q \geq 1$ queries to an oracle $\mathcal{O}$ solving $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH with probability one. As we aim at ruling out the existence of some reductions, we just consider the best possible case any black-box reduction must be able to handle. Now we consider the same splitting used in the proof of Lemma 2, $\mathcal{R}^{\mathcal{O}} = (\mathcal{R}_0^{\mathcal{O}}, \mathcal{R}_1)$, where the splitting point is the last oracle call, as shown in Figure 3. More formally, on the input of $[\mathbf{A}]$, for $\mathbf{A} \leftarrow \mathcal{D}_{\ell,k}$, and after making $Q-1$ oracle calls, $\mathcal{R}_0^{\mathcal{O}}$ stops by outputting the last query to $\mathcal{O}$, that is a matrix $[\widetilde{\mathbf{A}}]$, where $\widetilde{\mathbf{A}} \in \mathcal{D}_{\widetilde{\ell},\widetilde{k}}$, together with some state information s for $\mathcal{R}_1$. Next, $\mathcal{R}_1$ resumes the execution from the state information s and the answer $[\boldsymbol{w}] \in \mathbb{G}^{\widetilde{\ell}}$ given by the oracle, to finally output a vector $[\boldsymbol{v}] \in \mathbb{G}^{\ell}$. Without loss of generality, we assume that both stages $\mathcal{R}_0^{\mathcal{O}}$ and $\mathcal{R}_1$ receive the same random tape, \$ (and perhaps $\mathcal{R}_1$ will redo some of the computations performed by $\mathcal{R}_0^{\mathcal{O}}$).

In order to find some algebraic objects associated to the reduction, we first apply Lemma 2 to $\mathcal{R}_0^{\mathcal{O}}$. Notice that $\mathcal{O}$ is clearly an algebraic oracle (in the sense of Definition 1), because $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$ is described by a polynomial map, and there exists a polynomial map that parameterizes the subspace $\ker \widetilde{\mathbf{A}}^{\top}$ given $\widetilde{\mathbf{A}}$. Actually, one can use the $\widetilde{k}$-minors of $\widetilde{\mathbf{A}}$, which are just polynomials of degree $\widetilde{k}$, to obtain a basis of $\ker \widetilde{\mathbf{A}}^{\top}$. Then the oracle can use parameters $r_1, \ldots, r_{\widetilde{\ell}-\widetilde{k}}$ to build an arbitrary linear combination of the basis vectors. Lemma 2 implies that only the group elements in s can depend on $\mathbf{A}$. Indeed, the non-group elements in s can only depend on \$.

Next, from Lemma 1 applied to $\mathcal{R}_1$, we know that its output $[\boldsymbol{v}]$ is determined by a polynomial of degree at most one in the input group elements (*i.e.*, $\widetilde{\mathbf{A}}$ and the group elements in s), and the coefficients of this polynomial can only depend on \$, and the non-group elements in s, which in turn only depend on \$. Therefore, for every fixed \$, and every fixed oracle behaviour in the first $Q-1$ oracle calls, there exists a vector $u \in \mathbb{Z}_q^{\ell}$ and a linear map $\eta : \mathbb{Z}_q^{\widetilde{\ell}} \to \mathbb{Z}_q^{\ell}$ such that we can write $\boldsymbol{v} = \boldsymbol{u} + \eta(\boldsymbol{w})$, where $\boldsymbol{u}$ actually depends on the group elements in s. The important fact here is that η can only depend on \$, but not on $\mathbf{A}$.

Lemma 9. *Let $\mathcal{R}^{\mathcal{O}} = (\mathcal{R}_0^{\mathcal{O}}, \mathcal{R}_1)$ be a black-box reduction from $\mathcal{D}_{\ell,k}$-KerMDH to $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH, in the purely algebraic generic multilinear group model, making $Q \geq 1$ calls to an oracle $\mathcal{O}$ solving the latter with probability one. If $\mathcal{R}^{\mathcal{O}}$ succeeds with a non negligible probability then, for every possible behaviour of the oracle, either $\Pr(\eta(\boldsymbol{w}) \in S') > negl$ or $\Pr(\boldsymbol{u} \in S') > negl$, where $S' = \ker \mathbf{A}^{\top} \setminus \{\mathbf{0}\}$, $[\mathbf{A}]$ is the input of $\mathcal{R}^{\mathcal{O}}$, and its output is written as $[\boldsymbol{u} + \eta(\boldsymbol{w})]$, for some $\boldsymbol{u}$ only depending on the state output by $\mathcal{R}_0^{\mathcal{O}}$, $[\boldsymbol{w}]$ is the answer to the Q-th oracle query, and $\eta : \mathbb{Z}_q^{\widetilde{l}} \to \mathbb{Z}_q^{l}$ is a (randomized) linear map that only depends on the random tape of $\mathcal{R}^{\mathcal{O}}$.*

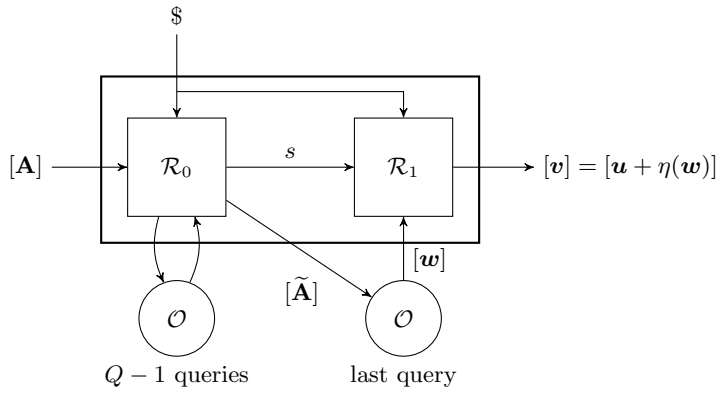

Fig. 3. Splitting of the black-box reduction.

Proof. Let us denote $S = \ker \mathbf{A}^\top$, where $[\mathbf{A}]$ is the input to $\mathcal{R}^{\mathcal{O}}$, and $S' = S \setminus \{\mathbf{0}\}$. Analogously, $\widetilde{S} = \ker \widetilde{\mathbf{A}}^\top$, where $[\widetilde{\mathbf{A}}]$ is the Q-th oracle query, and $\widetilde{S}' = \widetilde{S} \setminus \{\mathbf{0}\}$. From the discussion preceding the lemma, we know that $\boldsymbol{u}$ and η are well-defined and fulfil the required properties. In particular, η depends only on the random tape, \$, of $\mathcal{R}^{\mathcal{O}}$.

As a black-box reduction, $\mathcal{R}^{\mathcal{O}}$ is successful means that it is successful for every possible behaviour of the oracle in its Q queries. We arbitrarily fix its behaviour in the first $Q-1$ queries. Concerning the last one, for all $\boldsymbol{w} \in \widetilde{S}'$, $\Pr(\boldsymbol{u} + \eta(\boldsymbol{w}) \in S') > \mathit{negl}$, where the probability is computed with respect to \$ and the randomness of $[\mathbf{A}]$. Now, defining

$$p_{\boldsymbol{w}} = \Pr(\boldsymbol{u} \in S \wedge \boldsymbol{u} + \eta(\boldsymbol{w}) \in S')$$

$$r_{\boldsymbol{w}} = \Pr(\boldsymbol{u} \notin S \wedge \boldsymbol{u} + \eta(\boldsymbol{w}) \in S')$$

we have $p_{\boldsymbol{w}} + r_{\boldsymbol{w}} > \mathit{negl}$. But not all $r_{\boldsymbol{w}}$ can be non-negligible since the corresponding events are disjoint. Indeed, for every nonzero vector $\boldsymbol{w}$ and any different $\alpha_1, \alpha_2 \in \mathbb{Z}_q^\times$,

$$\boldsymbol{u} + \eta(\alpha_1 \boldsymbol{w}) \in S,\ \boldsymbol{u} + \eta(\alpha_2 \boldsymbol{w}) \in S \quad \Rightarrow \quad (\alpha_2 - \alpha_1)\boldsymbol{u} \in S \quad \Rightarrow \quad \boldsymbol{u} \in S$$

and then $\sum_{\alpha \in \mathbb{Z}_q^\times} r_{\alpha \boldsymbol{w}} \leq 1$. Therefore, there is some α_m such that $r_{\alpha_m \boldsymbol{w}} \leq \frac{1}{q-1}$, which in turn implies $p_{\alpha_m \boldsymbol{w}} > \mathit{negl}$. Now, we can split $p_{\alpha_m \boldsymbol{w}}$ depending on whether $\boldsymbol{u} \in S'$ or $\boldsymbol{u} = \mathbf{0}$, obtaining

$$\begin{aligned} p_{\alpha_m \boldsymbol{w}} &= \Pr(\boldsymbol{u} = \mathbf{0} \wedge \eta(\boldsymbol{w}) \in S') + \Pr(\boldsymbol{u} \in S' \wedge \boldsymbol{u} + \eta(\alpha_m \boldsymbol{w}) \in S') \leq \\ &\leq \Pr(\eta(\boldsymbol{w}) \in S') + \Pr(\boldsymbol{u} \in S') \end{aligned}$$

and concluding that either $\Pr(\boldsymbol{u} \in S') > \mathit{negl}$ or for all nonzero $\boldsymbol{w} \in \widetilde{S}'$, $\Pr(\eta(\boldsymbol{w}) \in S') > \mathit{negl}$. However, which one is true could depend on the particular behaviour of the oracle in the first $Q-1$ calls.

In some cases, one of the two strategies of the reduction mentioned above can be ruled out. Namely, we can prove that $\Pr(\eta(\boldsymbol{w}) \in S \setminus \{\mathbf{0}\}) \in \mathit{negl}$.

Lemma 10. *Consider integers $l = k+d$, $\widetilde{l} = \widetilde{k} + \widetilde{d}$ such that $k, d, \widetilde{k}, \widetilde{d} > 0$ and $k > \widetilde{k}$. Let $\eta : \mathbb{Z}_q^{\widetilde{l}} \to \mathbb{Z}_q^l$ be a linear map. Then, there exists a subspace F of $\mathrm{Im}\,\eta$ of dimension at most k such that for all $\widetilde{d}$-dimensional subspaces $\widetilde{S}$ of $\mathbb{Z}_q^{\widetilde{l}}$, either $\widetilde{S} \subset \ker \eta$ or $\dim F \cap \eta(\widetilde{S}) \geq 1$.*

Proof. If $\mathrm{rank}\,\eta \leq k$ it suffices to take $F = \mathrm{Im}\,\eta$. Indeed, if $\widetilde{S} \not\subset \ker \eta$, *i.e.*, $\eta(\widetilde{S}) \neq \{\mathbf{0}\}$, then $\dim F \cap \eta(\widetilde{S}) = \dim \eta(\widetilde{S}) \geq 1$.

Otherwise, $\operatorname{rank}\eta > k$, let F a subspace of $\operatorname{Im}\eta$ of dimension k, using the Grassman's formula,

$$\begin{aligned}\dim F \cap \eta(\widetilde{S}) &= \dim F + \dim \eta(\widetilde{S}) - \dim(F + \eta(\widetilde{S})) \geq \\ &\geq k + \dim \eta(\widetilde{S}) - \operatorname{rank}\eta \geq k + \dim \widetilde{S} - \dim\ker\eta - \operatorname{rank}\eta = \\ &= k + \widetilde{d} - \widetilde{l} = k - \widetilde{k} \geq 1\end{aligned}$$

Definition 8 (t-Elusiveness). *A family of subspaces $\mathcal{S}$ of a vector space X over the finite field $\mathbb{Z}_q$ is called t-elusive for some $t < \dim X$ if for all t-dimensional subspaces $F \subset X$, $\Pr(F \cap S \neq \{\mathbf{0}\}) \in negl$, where the probability is computed with respect to the choice of $S \in \mathcal{S}$.*

A matrix distribution $\mathcal{D}_{\ell,k}$ is called t-elusive if the family $\{\ker \mathbf{A}^\top\}_{\mathbf{A}\in\mathcal{D}_{\ell,k}}$ is t-elusive.

Lemma 11. *If a matrix distribution $\mathcal{D}_{\ell,k}$ is hard (as given in Definition 4) then $\mathcal{D}_{\ell,k}$ is k-elusive.*

Proof. By definition, given a non-k-elusive matrix distribution $\mathcal{D}_{\ell,k}$, there exists a k-dimensional vector subspace $F \subset \mathbb{Z}_q^\ell$ such that $\Pr_{\mathbf{A}\leftarrow\mathcal{D}_{\ell,k}}(F \cap \ker \mathbf{A}^\top \neq \{\mathbf{0}\}) > negl$. F can be efficiently computed from the description of $\mathcal{D}_{\ell,k}$ with standard tools from linear algebra.

Let $\mathbf{M} \in \mathbb{Z}_q^{k\times\ell}$ be a maximal rank matrix such that $\operatorname{Im}\mathbf{M}^\top = F$. Then, $\dim(F \cap \ker \mathbf{A}^\top) = \dim(\operatorname{Im}\mathbf{M}^\top \cap \ker \mathbf{A}^\top) \leq \dim\ker(\mathbf{A}^\top\mathbf{M}^\top) = \dim\ker(\mathbf{MA})^\top = \dim\ker(\mathbf{MA})$, as $\mathbf{MA}$ is a $k \times k$ square matrix. Thus, we know that

$$\Pr_{\mathbf{A}\leftarrow\mathcal{D}_{\ell,k}}(\operatorname{rank}(\mathbf{MA}) < k) > negl$$

Now we show how to solve the $\mathcal{D}_{\ell,k}$-MDDH problem on some k-linear group $\mathbb{G}$, by means of a k-linear map. Let $[(\mathbf{A}\|\boldsymbol{z})]$ be an instance of the $\mathcal{D}_{\ell,k}$-MDDH problem. In a 'real' instance $\boldsymbol{z} = \mathbf{A}\boldsymbol{x}$ for a uniformly distributed vector $\boldsymbol{x} \in \mathbb{Z}_q^k$, while in a 'random' instance, $\boldsymbol{z}$ is uniformly distributed $\mathbb{Z}_q^\ell$. A distinguisher can efficiently compute $[\mathbf{MA}]$ and $[\mathbf{M}\boldsymbol{z}]$. Observe that in a 'real' instance $\operatorname{rank}(\mathbf{MA}\|\mathbf{M}\boldsymbol{z}) = \operatorname{rank}(\mathbf{MA}\|\mathbf{MA}\boldsymbol{x}) = \operatorname{rank}(\mathbf{MA})$, while in a 'random' instance $\mathbf{M}\boldsymbol{z}$ is uniformly distributed in $\mathbb{Z}_q^k$. Therefore, for a 'random' instance there is a non-negligible probability that $\operatorname{rank}(\mathbf{MA}) < k$ and $\operatorname{rank}(\mathbf{MA}\|\mathbf{M}\boldsymbol{z}) = \operatorname{rank}(\mathbf{MA}) + 1$, because $\mathbf{M}\boldsymbol{z} \in \operatorname{Im}(\mathbf{MA})$ occurs only with a negligible probability $< \frac{1}{q}$. Then, the distinguisher can efficiently tell apart the two cases because with a k-linear map at hand computing the rank of a $k \times k$ or a $k \times k+1$ matrix can be done efficiently.

Theorem 2. *Let $\mathcal{D}_{\ell,k}$ be k-elusive. If there exists a black-box reduction in the purely algebraic generic multilinear group model from $\mathcal{D}_{\ell,k}$-KerMDH to another problem $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH with $\widetilde{k} < k$, then $\mathcal{D}_{\ell,k}$-KerMDH is easy.*

Proof. Let us assume the existence of the claimed reduction, $\mathcal{R}^{\mathcal{O}} = (\mathcal{R}_0^{\mathcal{O}}, \mathcal{R}_1)$, making $Q \geq 1$ oracle queries, where Q is minimal. Then, by Lemma 9, its output can be written as $[\boldsymbol{u}+\eta(\boldsymbol{w})]$, where $\eta : \mathbb{Z}_q^{\widetilde{l}} \to \mathbb{Z}_q^l$ is a (randomized) linear map that does not depend on the particular choice of the matrix $\mathbf{A}$ in the $\mathcal{D}_{\ell,k}$-KerMDH input instance, but only on the random tape of the reduction. Let us denote as above $S = \ker \mathbf{A}^\top$, and $S' = S \setminus \{\mathbf{0}\}$. Analogously, $\widetilde{S} = \ker \widetilde{\mathbf{A}}^\top$, where $\widetilde{\mathbf{A}} \leftarrow \mathcal{D}_{\widetilde{\ell},\widetilde{k}}$ and $\widetilde{S}' = \widetilde{S} \setminus \{\mathbf{0}\}$.

We now prove that in Lemma 9, for any possible behaviour of the oracle in the first $Q-1$ calls, there exists a particular behaviour in the last call such that $\Pr(\eta(\boldsymbol{w}) \in S')$ is negligible. Namely, the Q-th query is answered by $\mathcal{O}$ by choosing a uniformly distributed $\boldsymbol{w} \in \widetilde{S}'$. Indeed,

$$\Pr(\eta(\boldsymbol{w}) \in S') = \Pr(\eta(\boldsymbol{w}) \in S) - \Pr(\eta(\boldsymbol{w}) = 0)$$

Now, developing the second term,

$$\begin{aligned}\Pr(\eta(\boldsymbol{w}) = 0) &= \Pr(\eta(\boldsymbol{w}) = 0 \mid \widetilde{S} \subset \ker\eta)\Pr(\widetilde{S} \subset \ker\eta) + \\ &\quad + \Pr(\eta(\boldsymbol{w}) = 0 \mid \widetilde{S} \not\subset \ker\eta)\Pr(\widetilde{S} \not\subset \ker\eta) = \\ &= \Pr(\widetilde{S} \subset \ker\eta) + \Pr(\boldsymbol{w} \in \widetilde{S} \cap \ker\eta \mid \widetilde{S} \not\subset \ker\eta)\Pr(\widetilde{S} \not\subset \ker\eta) = \\ &= \Pr(\widetilde{S} \subset \ker\eta) + negl\end{aligned}$$

where the last equality uses that the probability that a vector uniformly distributed in $\widetilde{S}'$ belongs to a proper subspace of $\widetilde{S}'$ is negligible. And, analogously for the first term,

$$\begin{aligned}\Pr(\eta(\boldsymbol{w}) \in S) &= \Pr(\eta(\boldsymbol{w}) \in S \mid \eta(\widetilde{S}) \subset S) \Pr(\eta(\widetilde{S}) \subset S) + \\ &\quad + \Pr(\eta(\boldsymbol{w}) \in S \mid \eta(\widetilde{S}) \not\subset S) \Pr(\eta(\widetilde{S}) \not\subset S) = \\ &= \Pr(\eta(\widetilde{S}) \subset S) + \Pr(\boldsymbol{w} \in \widetilde{S} \cap \eta^{-1}(S) \mid \eta(\widetilde{S}) \not\subset S) \Pr(\eta(\widetilde{S}) \not\subset S) = \\ &= \Pr(\eta(\widetilde{S}) \subset S) + \mathit{negl}\end{aligned}$$

Thus,

$$\Pr(\eta(\boldsymbol{w}) \in S') = \Pr(\eta(\widetilde{S}) \subset S) - \Pr(\widetilde{S} \subset \ker \eta) + \mathit{negl}$$

Now, using Lemma 10, we know that there exists a subspace F of dimension at most k such that if $\widetilde{S} \not\subset \ker \eta$, then $\dim F \cap \eta(\widetilde{S}) \geq 1$. Therefore $\Pr(\eta(\widetilde{S}) \subset S) - \Pr(\widetilde{S} \subset \ker \eta) \leq \Pr(\eta(\widetilde{S}) \subset S \wedge \dim F \cap \eta(\widetilde{S}) \geq 1) \leq \Pr(\dim F \cap S \geq 1)$. But the last probability is negligible due to the k-elusiveness of $\mathcal{D}_{\ell,k}$.

Now applying Lemma 9 we know that $\Pr(\boldsymbol{u} \in S \setminus \{\mathbf{0}\}) > \mathit{negl}$ for any possible behaviour of the oracle in the first $Q-1$ calls. Therefore, we can modify the reduction $\mathcal{R}$ to output $\boldsymbol{u}$, without making the Q-th oracle call. The modified reduction is also successful, with only $Q-1$ oracle calls, which contradicts the assumption that Q is minimal. In summary, if the claimed reduction exists then there also exists an algorithm (a "reduction with $Q = 0$") directly solving $\mathcal{D}_{\ell,k}$-KerMDH without the help of any oracle.

Corollary 1. *If a matrix distribution family $\{\mathcal{D}_{\ell,k}\}$ is hard then for any $\mathcal{D}_{\ell,k}$ and $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$ in the family with $k > \widetilde{k}$ there is no black-box reduction in the generic group model from $\mathcal{D}_{\ell,k}$-KerMDH to $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH.*

Proof. Since all $\mathcal{D}_{\ell,k}$-MDDH problems in the family are generically hard on a k-linear group, we know that $\mathcal{D}_{\ell,k}$ is k-elusive by Lemma 11, and also $\mathcal{D}_{\ell,k}$-KerMDH is hard in that group (otherwise, any solution to $\mathcal{D}_{\ell,k}$-KerMDH can be used to solve $\mathcal{D}_{\ell,k}$-MDDH in a straightforward way). By the above theorem, no black-box reduction in the generic group model from $\mathcal{D}_{\ell,k}$-KerMDH to $\mathcal{D}_{\widetilde{\ell},\widetilde{k}}$-KerMDH can exist for $k > \widetilde{k}$.

4.2 Algebraic Reductions

In contrast to the previous negative results we now show how to build some natural reductions among different families of KerMDH problems of the same size. Thanks to the algebraic nature of matrix distributions it is easy to find some generic reductions among the corresponding problems.

Definition 9. *We say that $\mathcal{D}^1_{\ell,k}$ is algebraically reducible to $\mathcal{D}^2_{\ell,k}$ if there exist two efficiently samplable matrix distributions, $\mathcal{L}$ which outputs a matrix $\mathbf{L} \in \mathbb{Z}_q^{\ell \times \ell}$ matrix and $\mathcal{R}$ which outputs a matrix $\mathbf{R} \in \mathbb{Z}_q^{k \times k}$, such that given $\mathbf{A} \leftarrow \mathcal{D}^1_{\ell,k}$ the distribution of the matrix $\mathbf{LAR}$ is negligibly close to $\mathcal{D}^2_{\ell,k}$. In this case we write $\mathcal{D}^1_{\ell,k} \overset{a}{\Rightarrow} \mathcal{D}^2_{\ell,k}$.*

We note that since we assume that the matrices output by either of the distributions $\mathcal{D}^1_{\ell,k}$, $\mathcal{D}^2_{\ell,k}$ have full rank with overwhelming probability, the distributions $\mathcal{L}, \mathcal{R}$ must output full rank matrices also with overwhelming probability. We provide two examples of algebraic reductions: Taking random $\mathbf{L}$ and $\mathbf{R}$ gives the reduction $\mathcal{D}_{\ell,k} \overset{a}{\Rightarrow} \mathcal{U}_{\ell,k}$ for any matrix distribution $\mathcal{D}_{\ell,k}$, and considering $\mathbf{L}$ the identity matrix and $\mathbf{R}$ a random invertible diagonal matrix, we obtain $\mathcal{L}_k \overset{a}{\Rightarrow} \mathcal{R}\mathcal{L}_k$.

The notion of algebraic reducibility is useful to find reductions among the MDDH problems and also the Kernel problems.

Lemma 12. *$\mathcal{D}^1_{\ell,k} \overset{a}{\Rightarrow} \mathcal{D}^2_{\ell,k}$ implies both $\mathcal{D}^1_{\ell,k}$-MDDH $\Rightarrow \mathcal{D}^2_{\ell,k}$-MDDH and $\mathcal{D}^1_{\ell,k}$-KerMDH $\Rightarrow \mathcal{D}^2_{\ell,k}$-KerMDH.*

Proof. Given instance of the $\mathcal{D}^1_{\ell,k}$-MDDH problem, $([\mathbf{A}], [\boldsymbol{z}])$, the tuple $([\mathbf{LAR}], [\mathbf{L}\boldsymbol{z}])$, with $\mathbf{L} \leftarrow \mathcal{L}$, $\mathbf{R} \leftarrow \mathcal{R}$ is a properly distributed instance of the $\mathcal{D}^2_{\ell,k}$-MDDH problem. Indeed, it is easy to see that 'real' instances are transformed into 'real' instances, and 'random' instances into 'random' ones.

On the other hand, we show that given an algorithm $\mathcal{A}$ which solves $\mathcal{D}^2_{\ell,k}$-KerMDH there exists another algorithm which solves $\mathcal{D}^1_{\ell,k}$-KerMDH with the same probability. Given $\mathbf{A} \leftarrow \mathcal{D}^1_{\ell,k}$-KerMDH problem,

sample two matrices $\mathbf{L} \leftarrow \mathcal{L}$, $\mathbf{R} \leftarrow \mathcal{R}$ and construct an instance $\mathbf{LAR}$ of $\mathcal{D}^2_{\ell,k}$. Let $[\boldsymbol{x}]$ be the output of $\mathcal{A}$ on input $[\mathbf{LAR}]$, that is, $\boldsymbol{x}$ is a nonzero vector such that $\boldsymbol{x}^\top \mathbf{LAR} = \mathbf{0}^\top$. Since $\mathbf{L}$ and $\mathbf{R}$ are invertible with overwhelming probability, $\boldsymbol{x}^\top \mathbf{LA} = \mathbf{0}^\top$ also holds. Then output the nonzero vector $[\mathbf{L}^\top \boldsymbol{x}]$ as a solution to the $\mathcal{D}^1_{\ell,k}$-KerMDH problem.

From the above results, it is straightforward that $\mathcal{D}_{\ell,k}$-KerMDH $\Rightarrow$ $\mathcal{U}_{\ell,k}$-KerMDH, for any matrix distribution $\mathcal{D}_{\ell,k}$, and $\mathcal{L}_k$-KerMDH $\Rightarrow$ $\mathcal{RL}_k$-KerMDH.

4.3 Increasing Families of KerMDH Problems

Most matrix distributions, like $\mathcal{U}_{\ell,k}$, $\mathcal{L}_k$, $\mathcal{CI}_{k,d}$, $\mathcal{SC}_k$, $\mathcal{C}_k$ and $\mathcal{RL}_k$, are indeed families parameterized by their size k. The negative results in Corollary 1 prevent us to find reductions from larger to smaller KerMDH problems. Nevertheless, we provide here some examples of reductions going in the other way, within each of the previous families.

There is no known generic way to do that, and we use different techniques to build the reductions for each separate family. Observe that we cannot use the previously defined algebraic reductions here, because the reductions we need must increase the rank of the matrices, and this can be never done by matrix multiplications.

Lemma 13. $\mathcal{U}_{\widetilde{\ell},\widetilde{k}}$-KerMDH $\Rightarrow \mathcal{U}_{\ell,k}$-KerMDH *for* $\widetilde{k} \leq k$ *and* $\widetilde{\ell} \leq \ell$.

Proof. Given an instance $[\widetilde{\mathbf{A}}]$, with $\widetilde{\mathbf{A}} \leftarrow \mathcal{U}_{\widetilde{\ell},\widetilde{k}}$, we choose random invertible matrices $\mathbf{L} \in \mathbb{Z}_q^{\ell \times \ell}$ and $\mathbf{R} \in \mathbb{Z}_q^{k \times k}$ and compute $[\mathbf{A}] = \mathbf{L}([\widetilde{\mathbf{A}}] \oplus [\mathbf{B}])\mathbf{R}$, where $\mathbf{B}$ is any full-rank matrix in $\mathbb{Z}_q^{(\ell-\widetilde{\ell})\times(k-\widetilde{k})}$ and $\oplus$ operation denotes diagonal block matrix concatenation. Clearly, the probability distribution of the new matrix is statistically close to the uniform distribution in $\mathbb{Z}_q^{\ell \times k}$.

Any vector $[\boldsymbol{x}]$, obtained from a solver of $\mathcal{U}_{\ell,k}$-KerMDH, such that $\boldsymbol{x} \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$ can be transformed into $[\widetilde{\boldsymbol{x}}]$ such that $\widetilde{\boldsymbol{x}} \in \ker \widetilde{\mathbf{A}}^\top \setminus \{\mathbf{0}\}$ with overwhelming probability, by just letting $[\widetilde{x}]$ to be the first $\widetilde{\ell}$ components of $\mathbf{L}^\top[\boldsymbol{x}]$. Thus, we have built a tight reduction.

Lemma 14. $\mathcal{L}_k$-KerMDH $\Rightarrow \mathcal{L}_{k+1}$-KerMDH.

Proof. Observe that given a matrix $\widetilde{\mathbf{A}} \leftarrow \mathcal{L}_k$, with parameters $a_1, \ldots, a_k$, we can build a matrix $\mathbf{A}$ following the distribution $\mathcal{L}_{k+1}$, by adding an extra row and column to $\widetilde{\mathbf{A}}$ corresponding to new random parameter $a_{k+1} \in \mathbb{Z}_q$. Moreover, given $\boldsymbol{x} = (x_1, \ldots, x_{k+2}) \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$, the vector $\widetilde{\boldsymbol{x}} = (x_1, \ldots, x_k, x_{k+2})$ is in $\ker \widetilde{\mathbf{A}}^\top \setminus \{\mathbf{0}\}$ (except for a negligible probability due to the possibility that $a_{k+1} = 0$ and $\widetilde{\boldsymbol{x}} = \mathbf{0}$, while $\boldsymbol{x} \neq \mathbf{0}$). The reduction consists of choosing a random a_{k+1}, then building $[\mathbf{A}]$ from $[\widetilde{\mathbf{A}}]$ as above, and finally obtaining $[\widetilde{\boldsymbol{x}}]$ from $[\boldsymbol{x}]$ by deleting the $(k+1)$-th coordinate.

Lemma 15. $\mathcal{SC}_k$-KerMDH $\Rightarrow \mathcal{SC}_{k+1}$-KerMDH.

Proof. Similarly, from a matrix $\widetilde{\mathbf{A}} \leftarrow \mathcal{SC}_k$, with parameter a, we can obtain a matrix $\mathbf{A}$ following $\mathcal{SC}_{k+1}$ by adding a new row and column to $\widetilde{\mathbf{A}}$. Now given $\boldsymbol{x} = (x_1, \ldots, x_{k+2}) \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$, it is easy to see that the vector $\widetilde{\boldsymbol{x}} = (x_1, \ldots, x_{k+1})$ is always in $\ker \widetilde{\mathbf{A}}^\top \setminus \{\mathbf{0}\}$.

The proofs of $\mathcal{C}_k$-KerMDH $\Rightarrow$ $\mathcal{C}_{k+1}$-KerMDH and $\mathcal{RL}_k$-KerMDH $\Rightarrow$ $\mathcal{RL}_{k+1}$-KerMDH directly follow from the same ideas.

By combining the negative results in Corollary 1 with the explicit reductions given above, we conclude the proof of Theorem 1.

5 Application to Trapdoor Commitments

As a concrete application, we study how to abstract two constructions of trapdoor commitments in the literature to any Kernel Assumption. We recall the definition of a trapdoor commitment scheme.

Definition 10. *A commitment scheme is a tuple of three algorithms* $(\mathsf{K}, \mathsf{Comm}, \mathsf{Vrfy})$ *such that:*

- K *is a randomized algorithm, which on input the security parameter* 1^λ *outputs a commitment key* ck*,*
- Comm *is a randomized algorithm which, on input the commitment key* ck *and a message* m *in the message space* $\mathcal{M}_{ck}$ *outputs a commitment* c *and an opening* Op*,*
- Vrfy *is a deterministic algorithm which, on input the commitment key* ck*, a message* m *in the message space* $\mathcal{M}_{ck}$ *and an opening* Op*, outputs* 1 *if* Op *is a valid opening of* c *to the message* m *and* 0 *otherwise.*

Correctness requires that

$$\Pr\left[1 \leftarrow \mathsf{Vrfy}(ck, c, m, Op) : ck \leftarrow \mathsf{K}(1^\lambda), m \leftarrow \mathcal{M}_{ck}, (c, Op) \leftarrow \mathsf{Comm}(ck, m)\right] = 1.$$

Definition 11. *A commitment scheme is binding if, for any polynomial-time adversary* $\mathcal{A}$*,*

$$\Pr\left[1 \leftarrow \mathsf{Vrfy}(ck, c, m, Op) \cap 1 \leftarrow \mathsf{Vrfy}(ck, c, m', Op') : ck \leftarrow \mathsf{K}(1^\lambda), (c, m, Op, m', Op') \leftarrow \mathcal{A}(ck)\right]$$

is negligible. It is hiding if, for any polynomial-time adversary $\mathcal{A}$*,*

$$|\Pr\left[b' = b : ck \leftarrow \mathsf{K}(1^\lambda), (m_0, m_1, st) \leftarrow \mathcal{A}(ck), b \leftarrow \{0,1\}, (c, Op) \leftarrow \mathsf{Comm}(ck, m_b), b' \leftarrow \mathcal{A}(st, c)\right] - \frac{1}{2}|$$

is negligible.

Definition 12. *A commitment scheme is trapdoor if* K *additionally outputs a trapdoor key* tk *and there is an efficient algorithm* $\mathsf{TrapdoorEquiv}$ *which, on input* (ck, tk, c, m, Op, m') *outputs* Op' *such that* $1 \leftarrow \mathsf{Vrfy}(ck, c, m', Op')$*. Further, for any pair of valid messages* m*,* m' *and legitimately generated* ck, tk*, it holds that the distributions* (ck, c, Op') *when* $(c, Op) \leftarrow \mathsf{Comm}(ck, m)$*,* $Op' \leftarrow \mathsf{TrapdoorEquiv}(ck, tk, c, m, Op, m')$ *or when* $(c, Op') \leftarrow \mathsf{Comm}(ck, m')$ *are indistinguishable.*

5.1 Generalized Pedersen Commments in Multilinear Groups

In a group $(\mathbb{G}, q, \mathcal{P})$ where the discrete logarithm is hard, the Pedersen commitment is a statistically hiding and computationally binding commitment to a scalar. It can be naturally generalized to several scalars. Abe *et al.* [2] show how to do similar Pedersen type commitments to group elements in bilinear asymmetric groups under the DP Assumption, which in our language is the $\mathcal{U}_{1+d,1}$-KerMDH Assumption. With our new assumption family we can write both the Pedersen commitment and the commitment of [2] as a single construction and generalize it to (ideal) graded encodings.

- $\mathsf{K}(1^\lambda, d, m)$: Let $\mathcal{MG}_m = (e, \mathbb{G}_1, \mathbb{G}_2, \ldots, \mathbb{G}_m, q, \mathcal{P}_1, \ldots, \mathcal{P}_m) \leftarrow \mathsf{MGen}_m(1^\lambda)$. Sample $\mathbf{A} \leftarrow \mathcal{D}_{k+d,k}$. Let $\overline{\mathbf{A}}$ be the first k rows of $\mathbf{A}$ and $\underline{\mathbf{A}}$ the remaining d rows and $\mathbf{T} := \underline{\mathbf{A}}\overline{\mathbf{A}}^{-1}$ (w.l.o.g. we can assume $\overline{\mathbf{A}}$ is invertible). Output $ck := (\mathcal{MG}_m, [\mathbf{A}]_1)$, $tk := (\mathbf{T})$.
- $\mathsf{Comm}(ck, [\boldsymbol{c}], [\boldsymbol{v}]_r)$: To commit to a vector $[\boldsymbol{v}]_r \in \mathbb{G}_r^d$, $r < m$, pick $\boldsymbol{s} \leftarrow \mathbb{Z}_q^k$, and output

$$[\boldsymbol{c}]_{r+1} := e([(\boldsymbol{s}^\top \mid\mid \boldsymbol{v}^\top)]_r, [\mathbf{A}]_1) = [(\boldsymbol{s}^\top \mid\mid \boldsymbol{v}^\top)\,\mathbf{A}]_{r+1} \in \mathbb{G}_{r+1}^k,$$

  and the opening $Op = ([\boldsymbol{s}]_r)$.
- $\mathsf{Vrfy}(ck, [\boldsymbol{v}]_r, Op)$: This algorithm outputs 1 if

$$[\boldsymbol{c}]_{r+1} = e([\boldsymbol{s}^\top]_r, [\mathbf{A}]_1).$$

– $\mathsf{TrapdoorEquiv}(ck, tk, [\boldsymbol{c}]_{r+1}, [\boldsymbol{v}]_r, Op, [\boldsymbol{v}']_r)$: On a commitment $[\boldsymbol{c}]_{r+1} \in \mathbb{G}_{r+1}^k$ to message $[\boldsymbol{v}]_r$ with opening $Op = ([\boldsymbol{s}]_r)$, compute:

$$[\boldsymbol{s}']_r := [\boldsymbol{s}]_r + \mathbf{T}^\top [(\boldsymbol{v} - \boldsymbol{v}')]_r \in \mathbb{G}_r^k.$$

Output $Op' = ([\boldsymbol{s}']_r)$ as the opening of $[\boldsymbol{c}]_{r+1}$ to $[\boldsymbol{v}']_r$.

The analysis is almost identical to [2]. The correctness of the trapdoor opening is straightforward. The hiding property of the commitment is unconditional, while the soundness (at level r) is based on the $(r, m, \mathcal{D}_{\ell,k})$-$\mathsf{KerMDH}$ Assumption. Indeed, given two messages $[\boldsymbol{v}]_r, [\boldsymbol{v}']_r$ with respective openings $[\boldsymbol{s}]_r, [\boldsymbol{s}']_r$, it obviously follows that $[\boldsymbol{w}] := [\big((\boldsymbol{s} - \boldsymbol{s}')^\top \,||\, (\boldsymbol{v} - \boldsymbol{v}')^\top\big)]_r$ is a nonzero element in the kernel (in $\mathbb{G}_r$) of $\mathbf{A}^\top$, *i.e.* $e([\boldsymbol{w}^\top]_r, [\mathbf{A}]_1) = [\mathbf{0}_k]_{r+1}$.

Existing constructions. The Pedersen commitment (to multiple elements) is for messages in $\mathbb{G}_0$ and $\mathbf{A} \leftarrow \mathcal{U}_{d+1,1}$ and soundness is based on the $(0, m, \mathcal{U}_{d+1,1})$-$\mathsf{KerMDH}$. The construction proposed in [2] is for an asymmetric bilinear group $\mathcal{AG}_2 = (e, \mathbb{G}, \mathbb{H}, \mathbb{T}, q, \mathcal{P}, \mathcal{Q})$, and in this case messages are vectors in the group $\mathbb{H}$ and the commitment key consists of elements in $\mathbb{G}$, *i.e.* $ck = (\mathcal{AG}_2, [\mathbf{A}]_G)$, $\mathbf{A} \leftarrow \mathcal{U}_{d+1,1}$. Further, a previous version of the commitment scheme of [2] in symmetric bilinear roups (in [15]) corresponds to our construction with $\mathbf{A} \leftarrow \mathcal{U}_{2+d,2}$.

5.2 Group-to-Group Commitments

The commitments of the previous section are "shrinking" because they map a vector of length d in the group $\mathbb{G}_r$ to a vector of length k, for some k independent of and typically smaller than d. Abe *et al.* [3] noted that in some applications it is useful to have "group-to-group" commitments, *i.e.* commitments which are defined in the same group as the vector message. The motivation for doing so in the bilinear case is that these commitments are better compatible with Groth-Sahai proofs.

There is a natural generic construction of group-to-group commitments which uses as a black-box any trapdoor commitment $\widetilde{C} = (\widetilde{\mathsf{K}}, \widetilde{\mathsf{Comm}}, \widetilde{\mathsf{Vrfy}})$ mapping vectors of $\mathbb{G}_r$ to vectors of $\mathbb{G}_{r+1}$.

– $\mathsf{K}(1^\lambda, d, m)$: Run $(\widetilde{ck}, \widetilde{tk}) \leftarrow \widetilde{\mathsf{K}}(1^\lambda, m, d)$, output $ck = \widetilde{ck}$ and $tk = \widetilde{tk}$.
– $\mathsf{Comm}(ck, [\boldsymbol{v}]_r)$: To commit to a vector $[\boldsymbol{v}]_r \in \mathbb{G}_r^d$, $0 < r < m$, pick $[\boldsymbol{s}]_{r-1} \leftarrow [\mathbb{G}]_r^k$. Let $([\tilde{\boldsymbol{c}}]_r, \widetilde{Op}) \leftarrow \widetilde{\mathsf{Comm}}(ck, [\boldsymbol{s}]_{r-1})$ and output

$$c := ([\boldsymbol{s} + \boldsymbol{v}]_r, [\tilde{\boldsymbol{c}}]_r)$$

and the opening $Op = (\widetilde{Op})$.
– $\mathsf{Vrfy}(ck, c, [\boldsymbol{v}]_r, Op)$: On input $c = ([\boldsymbol{y}]_r, [\widetilde{\boldsymbol{c}}]_r)$ this algorithm outputs 1 if $[\boldsymbol{s}]_r := [\boldsymbol{y} - \boldsymbol{v}]_r$ satisfies that $1 \leftarrow \widetilde{\mathsf{Vrfy}}(ck, [\widetilde{\boldsymbol{c}}]_r, [\boldsymbol{s}]_r, Op)$, else it outputs 0.
– $\mathsf{TrapdoorEquiv}(ck, tk, c, [\boldsymbol{v}]_r, Op, [\boldsymbol{v}']_r)$: On a commitment $c = ([\boldsymbol{y}]_r, [\widetilde{\boldsymbol{c}}]_r)$ with opening $Op = \widetilde{Op}$, let $[\boldsymbol{s}]_r := [\boldsymbol{y} - \boldsymbol{v}]_r$ and $[\boldsymbol{s}']_r := [\boldsymbol{y} - \boldsymbol{v}']_r$. Run $\widetilde{Op}' \leftarrow \widetilde{\mathsf{TrapdoorEquiv}}(ck, tk, [\widetilde{\boldsymbol{c}}]_r, [\boldsymbol{s}]_r, \widetilde{Op}, [\boldsymbol{s}']_r)$.

Theorem 3. *If $\widetilde{C}$ is a perfectly hiding, computationally binding commitment to $[\boldsymbol{c}]_r$, then so is C.*

Proof. If $\widetilde{C}$ is perfectly hiding, then $([\boldsymbol{s} + \boldsymbol{v}]_r, \widetilde{Comm}(\widetilde{ck}, \boldsymbol{s}))$ perfectly hides $[\boldsymbol{v}]_r$ because $[\boldsymbol{s}]_r$ acts as a one-time pad. Similarly, it is straightforward to see that if $\widetilde{C}$ is computationally binding, so is $\widetilde{C}$.

In particular, this generic construction can be instantiated with $\widetilde{C}$ as the Pedersen commitment described in last section to obtain group-to-group commitments of size $k + d$, and where the opening is a vector of size k.

Existing constructions. Interestingly, this construction explains the two instantiations of "group-to-group" commitments given in [3]. Indeed, the generalization of the scheme described above to the asymmetric bilinear case (under the $\mathcal{U}_{1+d,1}$-KerMDH Assumption) matches the construction in [3], Sect. 4.1. When $\mathbf{A}$ is sampled from the distribution which results from sampling a matrix from the $\mathcal{RL}_2$ distribution and appending $d-1$ additional random rows, it matches the construction in [3], Sect. 4.2. For illustration, we discuss this last example in some more detail.

In [3], Sect. 4.2., (see Fig.2), the commitment to a message $(M_1, \ldots, M_d) \in \mathbb{G}$ is a tuple $(C_1, \ldots, C_{d+2}) \in \mathbb{G}^{d+2}$ where the elements in each of the groups are written in multiplicative notation, $G, H, \{G_j, F_j\}_{j=0,\ldots,d}$ are random elements in $\mathbb{G}$ and is defined by the equations:

$$C_i = M_i H^{\tau_i} \qquad C_{d+1} = G_0^{\tau_0} \prod_{j=1}^{d} G_j^{\tau_j} \qquad C_{d+2} = F_0^{\mu_0} \prod_{j=1}^{d} F_j^{\tau_j}.$$

Let $\begin{pmatrix} G_0 & 1_{\mathbb{G}} \\ 1_{\mathbb{G}} & F_0 \\ G_1 & F_1 \\ \vdots & \vdots \\ G_d & F_d \end{pmatrix} \in \mathbb{G}^{(d+2)\times 2}$ and define $\mathbf{A}$ as the corresponding matrix of discrete logarithms in base H. To see that this construction if of the appropriate form, it suffices to note that the pair (C_{d+1}, C_{d+2}) is the Pedersen commitment to $(\tau_1, \ldots, \tau_n)$ with randomness τ_0, μ_0 and commitment key $\mathbf{A}$.

6 A New Matrix Distribution and Its Applications

Both of our commitment schemes of section 5 base security on some $\mathcal{D}_{k+d,k}$-KerMDH assumptions, where d is the length of the committed vector. When $d > 1$, the only example of $\mathcal{D}_{k+d,k}$-MDDH Assumption considered in [12] is the one corresponding to the uniform matrix distribution $\mathcal{U}_{k+d,k}$, which is the weakest MDDH Assumption of size $(k+d) \times k$. Another natural assumption for $d > 1$ is the one associated to the matrix distribution resulting from sampling from an arbitrary distribution $\mathcal{D}_{k+1,k}$ (*e.g.*, $\mathcal{L}_k$) defining a hard $\mathcal{D}_{k+d,k}$-MDDH problem and adding $\ell - k - 1$ new random rows. The resulting $\mathcal{D}_{\ell,k}$-MDDH assumption is equivalent to the original $\mathcal{D}_{k+1,k}$-MDDH assumption. However, for efficiency reasons, we would like to have a matrix distributions with an even smaller representation size. This motivates us to introduce a new family of matrix distributions, the $\mathcal{CI}_{k,d}$ family.

Definition 13 (Circulant Matrix Distribution). *We define the distribution $\mathcal{CI}_{k,d}$ as follows*

$$\mathbf{A} = \begin{pmatrix} a_1 & & & 0 \\ \vdots & a_1 & & \\ a_d & \vdots & \ddots & \\ 1 & a_d & & a_1 \\ & 1 & \ddots & \vdots \\ & & \ddots & a_d \\ 0 & & & 1 \end{pmatrix} \in \mathbb{Z}_q^{(k+d)\times k}, \qquad \textit{where } a_i \leftarrow \mathbb{Z}_q$$

Matrix $\mathbf{A}$ is such that each column can be obtained by rotating one position the previous column, which explains the name. Notice that when $d = 1$, $\mathcal{CI}_{k,d}$ is exactly the symmetric cascade distribution $\mathcal{SC}_k$, introduced in [12]. We prove that $\mathcal{CI}_{k,d}$-MDDH Assumption holds generically in k-linear groups, which implies the hardness of the corresponding KerMDH problem.

On the other hand, Appendix A.1 shows that the representation size of $\mathcal{CI}_{k,d}$, which is the number of parameters d, is the optimal among all hard matrix distributions $\mathcal{D}_{k,k+d}$ defined by linear polynomials in the parameters. A similar argument shows that the circulant assumption is also optimal in the sense that it has a minimal numbers of nonzero entries among all hard matrix distributions $\mathcal{D}_{k,k+d}$.

The new assumption gives new instantiations of the commitment schemes of Section 5 with public parameters of size d, independent of k. Further, because the matrix $\mathbf{A} \leftarrow \mathcal{CI}_{k,d}$ has a many zero entries, the number of exponentiations computed by the Commit algorithm, and the number of pairings of the verification algorithm is kd — as opposed to $k(k+d)$ for the uniform assumption. This seems to be optimal — but we do not prove this formally.

To prove the generic hardness of the assumption, we turn to a result of Herold [18, Thm. 5.15 and corollaries]. It states that if all matrices produced by the matrix distribution are full-rank, $\mathcal{CI}_{k,d}$ is a hard matrix distribution. Indeed, an algorithm solving the $\mathcal{CI}_{k,d}$-MDDH problem in the generic k-linear group model must be able to compute a polynomial in the ideal $\mathfrak{H} \subset \mathbb{Z}_q[a_1, \ldots, a_d, z_1, \ldots, z_{k+d}]$ generated by all the $(k+1)$-minors of $\mathbf{A}\|\boldsymbol{z}$ as polynomials in $a_1, \ldots, a_d, z_1, \ldots, z_{k+d}$. Although this ideal can actually be generated using only a few of the minors, we need to build a Gröbner basis of $\mathfrak{H}$ to reason about the minimum degree a nonzero polynomial in $\mathfrak{H}$ can have. We show that, carefully selecting a monomial order, the set of all $(k+1)$-minors of $\mathbf{A}\|\boldsymbol{z}$ form a Gröbner basis, and all these minors have total degree exactly $k+1$. Therefore, all nonzero polynomials in $\mathfrak{H}$ have degree at least $k+1$, and then they cannot be evaluated by any algorithm in the generic k-linear group model.

As for other matrix distribution families, we can apply Corollary 1 and the following lemma to see that for any fixed $d \geq 1$ the hardness of $\mathcal{CI}_{k,d}$-KerMDH is strictly increasing.

Lemma 16. $\mathcal{CI}_{k,d}$-$\mathsf{KerMDH} \Rightarrow \mathcal{CI}_{k+1,d}$-$\mathsf{KerMDH}$.

Proof. The proof is very similar to the one of Lemma 15. From a matrix $\widetilde{\mathbf{A}} \leftarrow \mathcal{CI}_{k,d}$, with parameters $a_1, \ldots, a_d$, we also build $\mathbf{A} \in \mathcal{CI}_{k+1,d}$ by adding an extra row and column. Now given $\boldsymbol{x} = (x_1, \ldots, x_{k+d+1}) \in \ker \mathbf{A}^\top \setminus \{\mathbf{0}\}$, it is easy to see that the vector $\widetilde{\boldsymbol{x}} = (x_1, \ldots, x_{k+d})$ is always in $\ker \widetilde{\mathbf{A}}^\top \setminus \{\mathbf{0}\}$.

7 Acknowledgements

The authors thank E. Kiltz and G. Herold for improving this work through very fruitful discussions. Additionally, G. Herold gave us the insight and guidelines to prove the hardness of the circulant matrix distribution.

References

1. M. Abe, M. Chase, B. David, M. Kohlweiss, R. Nishimaki, and M. Ohkubo. Constant-size structure-preserving signatures: Generic constructions and simple assumptions. In X. Wang and K. Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 4–24, Beijing, China, Dec. 2–6, 2012. Springer, Berlin, Germany. 3
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo. Structure-preserving signatures and commitments to group elements. In T. Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 209–236, Santa Barbara, CA, USA, Aug. 15–19, 2010. Springer, Berlin, Germany. 1, 2, 3, 4, 11, 18, 19
3. M. Abe, K. Haralambiev, and M. Ohkubo. Group to group commitments do not shrink. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 301–317, Cambridge, UK, Apr. 15–19, 2012. Springer, Berlin, Germany. 3, 4, 19, 20
4. F. Bao, R. H. Deng, and H. Zhu. Variations of Diffie-Hellman problem. In S. Qing, D. Gollmann, and J. Zhou, editors, *ICICS 03*, volume 2836 of *LNCS*, pages 301–312, Huhehaote, China, Oct. 10–13, 2003. Springer, Berlin, Germany. 1, 10
5. N. Bari and B. Pfitzmann. Collision-free accumulators and fail-stop signature schemes without trees. In W. Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 480–494, Konstanz, Germany, May 11–15, 1997. Springer, Berlin, Germany. 1
6. G. Barthe, E. Fagerholm, D. Fiore, J. C. Mitchell, A. Scedrov, and B. Schmidt. Automated analysis of cryptographic assumptions in generic group models. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 95–112, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Berlin, Germany. 6

7. O. Blazy, E. Kiltz, and J. Pan. (hierarchical) identity-based encryption from affine message authentication. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 408–425, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Berlin, Germany. 2
8. D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In M. Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 41–55, Santa Barbara, CA, USA, Aug. 15–19, 2004. Springer, Berlin, Germany. 9
9. S. Brands. Untraceable off-line cash in wallets with observers (extended abstract). In D. R. Stinson, editor, *CRYPTO'93*, volume 773 of *LNCS*, pages 302–318, Santa Barbara, CA, USA, Aug. 22–26, 1994. Springer, Berlin, Germany. 1, 2, 11
10. E. Bresson, Y. Lakhnech, L. Mazaré, and B. Warinschi. A generalization of DDH with applications to protocol analysis and computational soundness. In A. Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 482–499, Santa Barbara, CA, USA, Aug. 19–23, 2007. Springer, Berlin, Germany. 1
11. M. Chase and S. Meiklejohn. Déjà Q: Using dual systems to revisit q-type assumptions. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 622–639, Copenhagen, Denmark, May 11–15, 2014. Springer, Berlin, Germany. 1
12. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar. An algebraic framework for Diffie-Hellman assumptions. In R. Canetti and J. A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147, Santa Barbara, CA, USA, Aug. 18–22, 2013. Springer, Berlin, Germany. 2, 3, 4, 5, 6, 8, 9, 10, 20, 24
13. A. Escala, G. Herold, E. Kiltz, C. Ràfols, and J. Villar. An algebraic framework for Diffie-Hellman assumptions. Cryptology ePrint Archive, Report 2013/377, 2013. `http://eprint.iacr.org/2013/377`. 6
14. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In T. Johansson and P. Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 1–17, Athens, Greece, May 26–30, 2013. Springer, Berlin, Germany. 2, 11
15. J. Groth. Homomorphic trapdoor commitments to group elements. Cryptology ePrint Archive, Report 2009/007, 2009. `http://eprint.iacr.org/2009/007`. 19
16. J. Groth. Homomorphic trapdoor commitments to group elements. *Manuscript.*, 2010. 1, 2, 11
17. J. Groth and S. Lu. A non-interactive shuffle with pairing based verifiability. In K. Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 51–67, Kuching, Malaysia, Dec. 2–6, 2007. Springer, Berlin, Germany. 1, 2, 9, 11
18. G. Herold. Applications of classical algebraic geometry to cryptography. *PhD Thesis, Ruhr-Universität Bochum*, 2014. 5, 6, 21, 23, 24
19. A. Joux and A. Rojat. Security ranking among assumptions within the uber assumption framework. Cryptology ePrint Archive, Report 2013/291, 2013. `http://eprint.iacr.org/`. 1, 10
20. C. S. Jutla and A. Roy. Switching lemma for bilinear tests and constant-size NIZK proofs for linear subspaces. In J. A. Garay and R. Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 295–312, Santa Barbara, CA, USA, Aug. 17–21, 2014. Springer, Berlin, Germany. 3, 4, 9
21. E. Kiltz, J. Pan, and H. Wee. Structure-preserving signatures from standard assumptions, revisited. *To appear in Crypto 2015*, 2015. 2, 3, 4
22. E. Kiltz and H. Wee. Quasi-adaptive NIZK for linear subspaces revisited. In E. Oswald and M. Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 101–128, Sofia, Bulgaria, Apr. 26–30, 2015. Springer, Berlin, Germany. 2, 4
23. F. Laguillaumie, P. Paillier, and D. Vergnaud. Universally convertible directed signatures. In B. K. Roy, editor, *ASIACRYPT 2005*, volume 3788 of *LNCS*, pages 682–701, Chennai, India, Dec. 4–8, 2005. Springer, Berlin, Germany. 1, 2, 11
24. B. Libert, T. Peters, M. Joye, and M. Yung. Non-malleability from malleability: Simulation-sound quasi-adaptive NIZK proofs and CCA2-secure encryption from homomorphic signatures. In P. Q. Nguyen and E. Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 514–532, Copenhagen, Denmark, May 11–15, 2014. Springer, Berlin, Germany. 3
25. B. Libert and D. Vergnaud. Multi-use unidirectional proxy re-signatures. In P. Ning, P. F. Syverson, and S. Jha, editors, *ACM CCS 08*, pages 511–520, Alexandria, Virginia, USA, Oct. 27–31, 2008. ACM Press. 1, 2, 11
26. U. M. Maurer. Towards the equivalence of breaking the Diffie-Hellman protocol and computing discrete algorithms. In Y. Desmedt, editor, *CRYPTO'94*, volume 839 of *LNCS*, pages 271–281, Santa Barbara, CA, USA, Aug. 21–25, 1994. Springer, Berlin, Germany. 1, 6
27. U. M. Maurer. Abstract models of computation in cryptography (invited paper). In N. P. Smart, editor, *10th IMA International Conference on Cryptography and Coding*, volume 3796 of *LNCS*, pages 1–12, Cirencester, UK, Dec. 19–21, 2005. Springer, Berlin, Germany. 1, 6
28. U. M. Maurer and S. Wolf. Lower bounds on generic algorithms in groups. In K. Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 72–84, Espoo, Finland, May 31 – June 4, 1998. Springer, Berlin, Germany. 1

29. T. P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In J. Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 129–140, Santa Barbara, CA, USA, Aug. 11–15, 1992. Springer, Berlin, Germany. 4
30. A.-R. Sadeghi and M. Steiner. Assumptions related to discrete logarithms: Why subtleties make a real difference. In B. Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 244–261, Innsbruck, Austria, May 6–10, 2001. Springer, Berlin, Germany. 1
31. V. Shoup. Lower bounds for discrete logarithms and related problems. In W. Fumy, editor, *EUROCRYPT'97*, volume 1233 of *LNCS*, pages 256–266, Konstanz, Germany, May 11–15, 1997. Springer, Berlin, Germany. 1

A More Details About the Circulant Matrix Distribution

In this appendix we give more details about both the hardness and the optimality of the representation size of the circulant matrix distribution.

A.1 Optimality of the Representation Size

Lemma 17. *A matrix distribution $\mathcal{D}_{\ell,k}$ defined by linear polynomials in the parameters, $\mathbf{A}(t_1,\ldots,t_d) = \mathbf{A}_0 + \mathbf{A}_1 t_1 + \ldots + \mathbf{A}_d t_d$, where $\mathbf{A}_1,\ldots,\mathbf{A}_d$ are linearly independent matrices, can only be hard if the number of parameters d is at least $\ell - k$.*

Proof. Assume for contradiction that $d < \ell - k$. Then, by gaussian elimination in $\mathbf{A}$ we can transform the matrix into another one, $\mathbf{B}(t_1,\ldots,t_d) = \mathbf{L}\mathbf{A}(t_1,\ldots,t_d)$ for a constant invertible matrix $\mathbf{L}$, such that the first column of $\mathbf{B}$ has zeroes in the lower $\ell - d - 1 \geq k$ positions (as at most $d+1$ entries can be linearly independent as polynomials in $t_1,\ldots,t_d$). Now, it is straightforward to see that this can be used to solve the associated $\mathcal{D}_{\ell,k}$-MDDH problem. Indeed, if $\widehat{\mathbf{L}}$ denotes the lowest k rows of $\mathbf{L}$, given an instance $([\mathbf{A}],[\boldsymbol{z}])$, we transform it into $([\widehat{\mathbf{L}}\mathbf{A}],[\widehat{\mathbf{L}}\boldsymbol{z}])$ and then compare the ranks of $\widehat{\mathbf{L}}\mathbf{A}$ and $\widehat{\mathbf{L}}\mathbf{A}\|\widehat{\mathbf{L}}\boldsymbol{z}$ (computed by means of the k-linear map). The ranks are equal for 'real' instances, while they are different with overwhelming probability for 'random' instances of the $\mathcal{D}_{\ell,k}$-MDDH problem, which contradicts the hardness of $\mathcal{D}_{\ell,k}$.

The linear independency requirement in the lemma just means that there is no redundancy among the d parameters (that is, the map $(t_1,\ldots,t_d) \mapsto \mathbf{A}(t_1,\ldots,t_d)$ is injective). The representation of $\mathbf{A}$ must contain at least d group elements, due to the previous injectivity. Therefore, $\mathcal{CI}_{k,d}$ has optimal representation size.

A.2 Hardness

Here we prove that $\mathcal{CI}_{k,d}$ is a hard matrix distribution (*i.e.*, the $\mathcal{CI}_{k,d}$-MDDH problem is generically hard in k-linear groups), using Theorem 5.15 and specially its Corollary 5.16 in [18] in the linear polynomial case, and Gröbner basis computations in some polynomial ideal.

Intuitively, an algorithm solving $\mathcal{CI}_{k,d}$-MDDH problem in the generic k-linear group model must know some nonzero polynomial in $t_1,\ldots,t_d,z_1,\ldots,z_{k+d}$ vanishing whenever $\boldsymbol{z} \in \operatorname{Im}\mathbf{A}(\boldsymbol{t})$. But this can only happen if such polynomial belongs to the ideal $\mathfrak{H} \in \overline{\mathbb{Z}}_q[t_1,\ldots,t_d,z_1,\ldots,z_{k+d}]$[10] generated by the relations between $t_1,\ldots,t_d,z_1,\ldots,z_{k+d}$ obtained by elimination of the variables $w_1,\ldots,w_k$ in the equation $\boldsymbol{z} = \mathbf{A}(\boldsymbol{t})\boldsymbol{w}$.

$\mathcal{CI}_{k,d}$ has some interesting properties that makes possible the generic hardness proof. Namely, it is defined by linear polynomials (*i.e.*, $\mathbf{A}(\boldsymbol{t})$ is made of polynomials of degree one in the parameters $t_1,\ldots,t_d$), and $\operatorname{rank}\mathbf{A}(\boldsymbol{t}) = k$ for all possible choices of $t_1,\ldots,t_d \in \overline{\mathbb{Z}}_q$ (*i.e.*, in the algebraic closure of $\mathbb{Z}_q$). This second property comes from the fact that the lowest k-minor of $\mathbf{A}(\boldsymbol{t})$ is constant and equal to 1. With these two properties, Theorem 5.15 and its Corollary 5.16 in [18] essentially state that $\mathfrak{H}$ is precisely the ideal generated by all the $(k+1)$-minors of $\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}$ as polynomials in $t_1,\ldots,t_d,z_1,\ldots,z_{k+d}$. More precisely,

[10] $\overline{\mathbb{Z}}_q$ denotes the algebraic closure of the field $\mathbb{Z}_q$. We define the ideal in the algebraic closure for technical reasons, although the polynomial used by the algorithm will necessarily have its coefficients in $\mathbb{Z}_q$.

Theorem 4 (from Theorem 5.15 and its Corollary 5.16 in [18]). *Let $\mathcal{D}_{\ell,k} = \{\mathbf{A}(\boldsymbol{t}) \mid \boldsymbol{t} \leftarrow \mathbb{Z}_q^d\}$ be a polynomial matrix distribution of degree one such that the matrices $\mathbf{A}(\boldsymbol{t})$ in the distribution have always full rank, for all choices of the parameters $t_1, \ldots, t_d$ in the algebraic closure of $\mathbb{Z}_q$. Let*

$$\mathfrak{H} = I(\{(\boldsymbol{t}, \mathbf{A}(\boldsymbol{t})\boldsymbol{w}) \mid \boldsymbol{t} \in \mathbb{Z}_q^d,\, \boldsymbol{w} \in \mathbb{Z}_q^k\})$$

that is the ideal of the polynomials in $\overline{\mathbb{Z}}_q[\boldsymbol{t}, \boldsymbol{z}]$ vanishing at all (rational) points such that $\boldsymbol{z} = \mathbf{A}(\boldsymbol{t})\boldsymbol{w}$, for some $\boldsymbol{w} \in \mathbb{Z}_q^k$, and

$$\mathfrak{D} = (\{\det{}_{i_1,\ldots,i_{k+1}}(\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}) \mid 1 \leq i_1 < i_2 < \cdots < i_{k+1} \leq \ell\})$$

the ideal generated by all $(k+1)$-minors of $\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}$. Then $\mathfrak{H} = \mathfrak{D}$ and it is a prime ideal.

At this point, proving the generic hardness of $\mathcal{CI}_{k,d}$ amounts to proving that there is no nonzero polynomial of total degree less that $k+1$ in $\mathfrak{H}$. Indeed, this means that the only way to generically solve the $\mathcal{CI}_{k,d}$-MDDH problem is computing a polynomial of degree strictly greater than k, which is not feasible in k-linear groups. Notice that in the general case $d \geq 1$, finding a lower bound for the total degree in $\mathfrak{H}$ is a nontrivial task, while in the case $\ell = k+1$ or $d = 1$, as seen in [12], it is as easy as computing the degree of the determinant polynomial $\det(\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z})$. The main reason for that difficulty is the fact that the ideal $\mathfrak{H}$ is not principal. Therefore, we need to compute a Gröbner basis of $\mathfrak{H}$, and show that all the polynomials in it have total degree at least $k+1$.

Gröbner bases can be computed quite easily for specific ideals by means of a computer, but here we will build bases for an infinite collection of ideals, that is for arbitrary values of the size parameters k and d. Thus, we have to compute them by hand. Fortunately, we manage to show that the set of all $(k+1)$-minors of $\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}$ as polynomials in $t_1, \ldots, t_d, z_1, \ldots, z_{k+d}$, where $\mathbf{A} \leftarrow \mathcal{CI}_{k,d}$, is a Gröbner basis of $\mathfrak{H}$.

We recall some basic notions related to ideals and Gröbner basis. An admissible monomial order $\prec$ in the polynomial ring $\mathbb{Z}_q[t_1, \ldots, t_d, z_1, \ldots, z_{k+d}]$ is a total order among the monomials in it such that for any monomials $\mathfrak{m}_1, \mathfrak{m}_2, \mathfrak{m}_3$

1. $\mathfrak{m}_1 \neq 1 \;\Rightarrow\; 1 \prec \mathfrak{m}_1$
2. $\mathfrak{m}_1 \prec \mathfrak{m}_2 \;\Rightarrow\; \mathfrak{m}_1\mathfrak{m}_3 \prec \mathfrak{m}_2\mathfrak{m}_3$

The leading monomial of a polynomial $\mathfrak{p}$, denoted by $\mathrm{LM}(\mathfrak{p})$ is defined as the greatest of its monomials (without the coefficient)[11] with respect of the monomial order. We recall that a Gröbner basis of $\mathfrak{H} \subset \mathbb{Z}_q[t_1, \ldots, t_d, z_1, \ldots, z_{k+d}]$ with respect of an admissible monomial order is a set of nonzero polynomials $G = \{\mathfrak{g}_1, \ldots, \mathfrak{g}_s\} \subset \mathfrak{H}$ with the following properties

1. for every $\mathfrak{f} \in \mathfrak{H}$ there exist $\mathfrak{c}_1, \ldots, \mathfrak{c}_s \in \mathbb{Z}_q[t_1, \ldots, t_d, z_1, \ldots, z_{k+d}]$ such that $\mathfrak{f} = \mathfrak{c}_1\mathfrak{g}_1 + \ldots + \mathfrak{c}_s\mathfrak{g}_s$,
2. for every $\mathfrak{f} \in \mathfrak{H}$, $\mathrm{LM}(\mathfrak{f})$ is divisible by $\mathrm{LM}(\mathfrak{g})$ for some $\mathfrak{g} \in G$.

The following lemma comes in a straightforward way from the previous definition

Lemma 18. *The minimal degree of nonzero polynomials in an ideal $\mathfrak{H}$ is the minimal degree of the polynomials in any Gröbner basis of $\mathfrak{H}$ with respect to any admissible monomial order compatible with the total degree.*

From now on we fix the following admissible monomial order:

1. $\deg \mathfrak{m}_1 < \deg \mathfrak{m}_2 \;\Rightarrow\; \mathfrak{m}_1 \prec \mathfrak{m}_2$, where deg denotes the total degree of the monomial,
2. $z_1 \prec \ldots \prec z_{k+d} \prec t_1 \prec \ldots \prec t_d$,
3. if $\deg \mathfrak{m}_1 = \deg \mathfrak{m}_2$, $\prec$ is the lexicographical order. That is, we write $\mathfrak{m}_1 = z_1^{\alpha_1} \cdots z_{k+d}^{\alpha_{k+d}} t_1^{\alpha_{k+d+1}} \cdots t_d^{\alpha_{k+2d}}$ and $\mathfrak{m}_2 = z_1^{\beta_1} \cdots z_{k+d}^{\beta_{k+d}} t_1^{\beta_{k+d+1}} \cdots t_d^{\beta_{k+2d}}$. Then, for the same total degree, $\mathfrak{m}_1 \prec \mathfrak{m}_2$ if and only if the first nonzero difference $\beta_i - \alpha_i$ is positive.

[11] We call *leading term* to the leading monomial multiplied by the corresponding coefficient.

Given $\mathbf{A} \leftarrow \mathcal{CI}_{k,d}$ we denote by $\Delta(\boldsymbol{i}) = \Delta(i_1, \ldots, i_{k+1}) = \det_{i_1,\ldots,i_{k+1}}(\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z})$ the $(k+1)$-minor of $\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}$ defined by the rows $1 \leq i_1 < i_2 < \cdots < i_{k+1} \leq k+d$. From now on we will use $\boldsymbol{i} \in \binom{k+d}{k+1}$ as a shorthand for the previous inequalities. These determinants have very special properties. Indeed, we show that with respect to the previous monomial order the main diagonal defines their leading monomial.

Lemma 19. *For all* $\boldsymbol{i} \in \binom{k+d}{k+1}$, $LM(\Delta(\boldsymbol{i})) = z_{i_{k+1}} t_{i_1} t_{i_2-1} \cdots t_{i_k-k+1}$.

Proof. In order to simplify the notation, we can always write the $(k+1)$-minors of $\mathbf{A}(\boldsymbol{t})\|\boldsymbol{z}$ as

$$\Delta(\boldsymbol{i}) = \begin{vmatrix} t_{i_1} & t_{i_1-1} & \cdots & t_{i_1-k+1} & z_{i_1} \\ t_{i_2} & t_{i_2-1} & \cdots & t_{i_2-k+1} & z_{i_2} \\ \vdots & \vdots & \vdots\vdots & \vdots & \vdots \\ t_{i_k} & t_{i_k-1} & \cdots & t_{i_k-k+1} & z_{i_k} \\ t_{i_{k+1}} & t_{i_{k+1}-1} & \cdots & t_{i_{k+1}-k+1} & z_{i_{k+1}} \end{vmatrix}$$

assuming that $t_{d+1} = 1$, and $t_i = 0$ for any i outside the range $i = 1, \ldots, d+1$. Then we show that in the development of the determinant, the monomial corresponding to the main diagonal $\mathfrak{m}_{\text{diag}} = z_{i_{k+1}} t_{i_1} t_{i_2-1} \cdots t_{i_k-k+1}$ is the leading monomial. Firstly, notice that all the terms occurring in the main diagonal are proper terms (*i.e.*, neither 0 nor 1). Indeed, $1 \leq i_1 \leq i_2 - 1 \leq \cdots \leq i_k - k + 1 \leq i_{k+1} - k \leq d$, which is something that deeply depends on the circulant structure of the matrix $\mathbf{A}$.

Now, assume by contradiction that there is another term in the development of the determinant, $\mathfrak{m} = z_{i_{\sigma(k+1)}} t_{i_{\sigma(1)}} t_{i_{\sigma(2)}-1} \cdots t_{i_{\sigma(k)}-k+1}$ (written in a possibly unsorted way), for a permutation $\sigma \in S_{k+1}$, such that $\mathfrak{m}_{\text{diag}} \prec \mathfrak{m}$. Due to the monomial order, the last column must contribute to this term with the variable z_{k+1} (that is $\sigma(k+1) = k+1$). Otherwise, $z_{i_{\sigma(k+1)}} \prec z_{i_{k+1}}$, which contradicts $\mathfrak{m}_{\text{diag}} \prec \mathfrak{m}$.

We can rewrite $\mathfrak{m}$ in terms of the inverse permutation $\pi = \sigma^{-1}$ as

$$\mathfrak{m} = z_{i_{k+1}} t_{i_1-\pi(1)+1} t_{i_2-\pi(2)+1} \cdots t_{i_k-\pi(k)+1}$$

Then, $\pi(1) \neq 1$ would imply $a_{i_1-\pi(1)+1} \prec a_{i_1}$, which is also in contradiction with $\mathfrak{m}_{\text{diag}} \prec \mathfrak{m}$. Therefore, $\sigma(1) = 1$. Observe that it could happen that $i_1 - \pi(1) + 1 \leq 0$, which means that $t_{i_1-\pi(1)+1}$ is actually 0, which also contradicts $\mathfrak{m}_{\text{diag}} \prec \mathfrak{m}$.

Proceeding similarly with subsequent indexes (rows) in increasing order, we easily show that σ can only be the identity permutation, which concludes the proof.

Now we prove that the set of all $(k+1)$-minors of $\mathbf{A}\|\boldsymbol{z}$ is a Gröbner basis. The proof of this result is rather technical and deeply relies on the properties of determinants.

Theorem 5. *The set* $G = \{\Delta(\boldsymbol{i}) \mid \boldsymbol{i} \in \binom{k+d}{k+1}\}$ *is a Gröbner basis of* $\mathfrak{H}$ *with respect to the monomial order* $\prec$.

Proof. The usual way to prove that a set $G = \{\mathfrak{g}_1, \ldots, \mathfrak{g}_s\}$ is a Gröbner basis is by means of the so-called S-polynomials. The S-polynomial of a pair $\mathfrak{g}_i, \mathfrak{g}_j \in G$ for $i \neq j$ is defined by

$$\mathfrak{s}_{i,j} = \text{SPOL}(\mathfrak{g}_i, \mathfrak{g}_j) = \frac{\mathfrak{m}_{i,j}}{\mathfrak{m}_i}\mathfrak{g}_i - \frac{\mathfrak{m}_{i,j}}{\mathfrak{m}_j}\mathfrak{g}_j \tag{2}$$

where $\mathfrak{m}_i = \text{LM}(\mathfrak{g}_i)$, $\mathfrak{m}_j = \text{LM}(\mathfrak{g}_j)$, and $\mathfrak{m}_{i,j}$ denotes the least common multiple of $\mathfrak{m}_i$ and $\mathfrak{m}_j$. Then, G is a Gröbner basis if and only if for all $i \neq j$, $\text{RED}_G(\mathfrak{s}_{i,j}) = 0$, where $\text{RED}_G(\mathfrak{p})$ denotes the reduction of a polynomial $\mathfrak{p}$ by repeatedly taking the remainder of the division by elements in G until no further division can be properly performed[12]. Indeed, the famous Buchberger's algorithm for computing Gröbner

[12] The reduction algorithm repeatedly takes a polynomial $\mathfrak{p}$ and checks whether some $\text{LM}(\mathfrak{g}_i)$ divides $\text{LM}(\mathfrak{p})$. If so, the algorithm cancels out the leading term of $\mathfrak{p}$ by subtracting from it the appropriate multiple of $\mathfrak{g}_i$, and repeats the procedure with the resulting polynomial. Otherwise, the algorithm adds the leading term of $\mathfrak{p}$ to the output polynomial and proceeds with the remaining terms of $\mathfrak{p}$, until they are exhausted. The output has the property that none of its monomials is divisible by any $\text{LM}(\mathfrak{g}_i)$.

basis iteratively uses the previous computation to either verify that a pair $\mathfrak{g}_i, \mathfrak{g}_j \in G$ passes the check, or to add its reduced S-polynomial to G.

Observe that any expression of the form $\mathfrak{p} = \mathfrak{c}_1\mathfrak{g}_1 + \ldots + \mathfrak{c}_s\mathfrak{g}_s$, where all the leading monomials $\mathfrak{n}_j = \mathrm{LM}(\mathfrak{c}_j\mathfrak{g}_j)$, $j = 1, \ldots, s$, are different, shows that $\mathrm{RED}_G(\mathfrak{p}) = 0$. Indeed, without loss off generality we can sort the previous terms to ensure that $\mathfrak{n}_1 \prec \ldots \prec \mathfrak{n}_s$. Clearly, $\mathrm{LM}(\mathfrak{p}) = \mathfrak{n}_s$ and reducing $\mathfrak{p}$ by $\mathfrak{g}_s$ gives the remainder $\mathfrak{c}_1\mathfrak{g}_1 + \ldots + \mathfrak{c}_{s-1}\mathfrak{g}_{s-1}$. Therefore, by induction, we get $\mathrm{RED}_G(\mathfrak{p}) = 0$.

Buchberger also provided two optimization rules to speed up the algorithm. In particular, the second one (Buchberger's chain criterion) says that if for some indexes i, j, v, $\mathrm{RED}_G(\mathfrak{s}_{i,v}) = \mathrm{RED}_G(\mathfrak{s}_{j,v}) = 0$ and $\mathfrak{m}_v$ divides $\mathfrak{m}_{i,j}$, then also $\mathrm{RED}_G(\mathfrak{s}_{i,j}) = 0$. We use this criterion to inductively show that in our case, we only need to deal with pairs of $(k+1)$-minors differing only in one row.

We now split the proof into several technical claims. The idea is drawing a path between any two $(k+1)$-minors in which at each step we only change one row of the minor.

The first claim says that from any two $(k+1)$-minors with the same upper $\alpha - 1$ rows but that differ in the α-th row, we can build a third "hybrid" minor by moving the α-th row from one determinant to the other, and the corresponding three leading monomials are related in a suitable way for the chain criterion described above.

Claim 1. For any two sequences $\boldsymbol{i}, \boldsymbol{i^*} \in \binom{k+d}{k+1}$ such that the first difference occurs at position α, for $1 \leq \alpha \leq k$, that is $i_j = i^*_j$ if $j < \alpha$ and $i_\alpha < i^*_\alpha$,

$$\mathrm{LM}(\Delta(i_1, \ldots, i_\alpha, i^*_{\alpha+1} \ldots, i^*_{k+1})) \quad \text{divides} \quad \mathrm{lcm}(\mathrm{LM}(\Delta(\boldsymbol{i})), \mathrm{LM}(\Delta(\boldsymbol{i^*})))$$

Proof (of Claim 1). According to Lemma 19,

$$\mathrm{LM}(\Delta(i_1, \ldots, i_\alpha, i^*_{\alpha+1} \ldots, i^*_{k+1})) = \begin{cases} z_{i^*_{k+1}} t_{i_1} \cdots t_{i_\alpha - \alpha + 1} t_{i^*_{\alpha+1} - \alpha} \cdots t_{i^*_k - k + 1} & \text{if } \alpha < k \\ z_{i^*_{k+1}} t_{i_1} \cdots t_{i_k - k + 1} & \text{if } \alpha = k \end{cases}$$

Then the claim directly comes from the fact that this monomial can be split into two coprime factors $t_{i_1} \cdots t_{i_\alpha - \alpha + 1}$ and $z_{i^*_{k+1}} t_{i^*_{\alpha+1} - \alpha} \cdots t_{i^*_k - k + 1}$, each dividing $\mathrm{LM}(\Delta(\boldsymbol{i}))$ and $\mathrm{LM}(\Delta(\boldsymbol{i^*}))$, respectively. Indeed, both factors are coprime because $i_1 \leq \cdots \leq i_\alpha - \alpha + 1 < i^*_\alpha - \alpha + 1 \leq i^*_{\alpha+1} - \alpha \leq \cdots \leq i^*_{k+1} - k$. □

We call *adjacent pair* to any pair of minors $\Delta(\boldsymbol{i})$ and $\Delta(\boldsymbol{i^*})$ such that $\boldsymbol{i}$ and $\boldsymbol{i^*}$ differ only at position α, for some $1 \leq \alpha \leq k+1$, that is $i_j = i^*_j$ if $j \neq \alpha$ and $i_\alpha < i^*_\alpha$. This pair can be actually described by an increasing sequence of length $k+2$, $1 \leq i_1 < \cdots < i_\alpha < i^*_\alpha < \cdots < i_{k+1} \leq k + d \in \binom{k+d}{k+2}$ and the index α. The previous claim allows us to build a path connecting any two $(k+1)$-minors such that every consecutive pairs of minors in the path is an adjacent pair. An example for $k = 5$ is depicted below. The numbers in every column in the table correspond to the indices of the rows in every minor in the path connecting $\Delta(1, 5, 6, 9, 12, 13)$ and $\Delta(2, 4, 7, 8, 11, 14)$.

i_1	1	1	1	1	1	**1**	**2**
i_2	**5**	**4**	4	4	4	4	4
i_3	6	6	6	6	**6**	**7**	7
i_4	9	**9**	**8**	8	8	8	8
i_5	12	12	**12**	**11**	11	11	11
i_6	13	13	13	**13**	**14**	14	14

Using the above path we show that, according to Buchberger's chain criterion, to prove that G is a Gröbner basis it suffices to check only the S-polynomials of adjacent pairs.

Claim 2. If the S-polynomial of every adjacent pair of $(k+1)$-minors is reducible to 0, then so are all the other S-polynomials (and therefore G is a Gröbner basis of $\mathfrak{H}$).

Proof (of Claim 2). We prove the claim by (descending) induction in α, the index of the first row-difference between two $(k+1)$-minors. For $\alpha = k+1$ (*i.e.*, the minors only differ in the last row) the statement is obviously true, as the two minors form an adjacent pair. Now, let us assume that the statement is true for $\alpha = \alpha_0$, where $1 < \alpha_0 \leq k+1$. Then for any two minors with the first row-difference occurring at row $\alpha_0 - 1$, say $\mathfrak{g} = \Delta(i_1, \ldots, i_{\alpha_0-2}, i_{\alpha_0-1}, \ldots, i_{k+1})$ and $\mathfrak{g}^* = \Delta(i_1, \ldots, i_{\alpha_0-2}, i^*_{\alpha_0-1}, \ldots, i^*_{k+1})$ with $i_{\alpha_0-1} < i^*_{\alpha_0-1}$, we define as in the first claim the "hybrid" minor $\mathfrak{h} = \Delta(i_1, \ldots, i_{\alpha_0-1}, i^*_{\alpha_0}, \ldots, i^*_{k+1})$. Then, by Claim 1 we know that $\mathrm{LM}(\mathfrak{h})$ divides $\mathrm{lcm}(\mathrm{LM}(\mathfrak{g}), \mathrm{LM}(\mathfrak{g}^*))$. On the other hand, the induction assumption implies $\mathrm{RED}_G(\mathrm{SPOL}(\mathfrak{h}, \mathfrak{g})) = 0$, since the first row-difference between $\mathfrak{g}$ and $\mathfrak{h}$ occurs at row α_0. Moreover, $\mathrm{RED}_G(\mathrm{SPOL}(\mathfrak{h}, \mathfrak{g}^*)) = 0$ because $(\mathfrak{h}, \mathfrak{g}^*)$ is an adjacent pair. Thus, by Buchberger's chain criterion, $\mathrm{RED}_G(\mathrm{SPOL}(\mathfrak{g}, \mathfrak{g}^*)) = 0$, concluding the proof of the second claim. □

The last step in the proof of the theorem is showing that the S-polynomial of every adjacent pair of $(k+1)$-minors is reducible to 0. Actually, the S-polynomial of an adjacent pair of minors can be embedded into the determinant of a $(k+2) \times (k+2)$ matrix. This matrix gives us a syzygy (*i.e.*, a linear relation with polynomial coefficients among elements in G) that allows to manually reduce the S-polynomial to 0.

Claim 3. For any adjacent pair of $(k+1)$-minors given by the sequence $\boldsymbol{i} \in \binom{k+d}{k+2}$ and the index $1 \leq \alpha \leq k+1$, that is $\mathfrak{g} = \Delta(i_1, \ldots, i_\alpha, i_{\alpha+2} \ldots, i_{k+2})$ and $\mathfrak{g}^* = \Delta(i_1, \ldots, i_{\alpha-1}, i_{\alpha+1} \ldots, i_{k+2})$, $\mathrm{RED}_G(\mathrm{SPOL}(\mathfrak{g}, \mathfrak{g}^*)) = 0$.

Proof (of Claim 3). Let us consider for the case $\alpha \leq k$ the extended matrix

$$\mathbf{B} = \begin{pmatrix} t_{i_1} & t_{i_1-1} & \cdots & t_{i_1-k+1} & z_{i_1} & t_{i_1-\alpha+1} \\ \vdots & \vdots & \vdots\vdots\vdots & \vdots & \vdots & \vdots \\ t_{i_\alpha} & t_{i_\alpha-1} & \cdots & t_{i_\alpha-k+1} & z_{i_\alpha} & t_{i_\alpha-\alpha+1} \\ t_{i_{\alpha+1}} & t_{i_{\alpha+1}-1} & \cdots & t_{i_{\alpha+1}-k+1} & z_{i_{\alpha+1}} & t_{i_{\alpha-1}-\alpha+1} \\ \vdots & \vdots & \vdots\vdots\vdots & \vdots & \vdots & \vdots \\ t_{i_{k+2}} & t_{i_{k+2}-1} & \cdots & t_{i_{k+2}-k+1} & z_{i_{k+2}} & t_{i_{k+2}-\alpha+1} \end{pmatrix}$$

which is a $(k+2) \times (k+2)$ matrix[13] with two repeated columns: the α-th and the last columns. Thus, $\det \mathbf{B} = 0$, and using Laplace expansion of the determinant along the last column we obtain the following syzygy:

$$\sum_{\substack{j=1 \\ j \neq \alpha, \alpha+1}}^{k+2} (-1)^{k+j} t_{i_j-\alpha+1} \mathfrak{h}_j + (-1)^{k+\alpha} \left(t_{i_\alpha-\alpha+1} \mathfrak{g}^* - t_{i_{\alpha+1}-\alpha+1} \mathfrak{g} \right) = 0 \tag{3}$$

where $\mathfrak{h}_j = \Delta(i_1, \ldots, i_{j-1}, i_{j+1}, \ldots, i_{k+2})$. Actually, $\mathfrak{g} = \mathfrak{h}_\alpha$ and $\mathfrak{g}^* = \mathfrak{h}_{\alpha+1}$.

Notice that the S-polynomial $\mathfrak{s} = \mathrm{SPOL}(\mathfrak{g}, \mathfrak{g}^*)$, as given by Equation 2, is exactly $t_{i_{\alpha+1}-\alpha+1}\mathfrak{g} - t_{i_\alpha-\alpha+1}\mathfrak{g}^*$, since

$$\mathrm{LM}(\mathfrak{g}) = z_{i_{k+2}} t_{i_1} \cdots t_{i_\alpha-\alpha+1} t_{i_{\alpha+2}-\alpha} \cdots t_{i_{k+1}-k+1}$$

and

$$\mathrm{LM}(\mathfrak{g}^*) = z_{i_{k+2}} t_{i_1} \cdots t_{i_{\alpha-1}-\alpha+2} t_{i_{\alpha+1}-\alpha+1} \cdots t_{i_{k+1}-k+1}$$

As a consequence, Equation 3 gives an explicit reduction of $\mathfrak{s}$ to 0. Namely,

$$\mathfrak{s} = \sum_{\substack{j=1 \\ j \neq \alpha, \alpha+1}}^{k+2} (-1)^{\alpha+j} t_{i_j-\alpha+1} \mathfrak{h}_j \tag{4}$$

Actually, to see that Equation 4 implies $\mathrm{RED}_G(\mathfrak{s}) = 0$, we only need to show that all leading monomials $\mathfrak{n}_{\alpha,j} = \mathrm{LM}(t_{i_j-\alpha+1}\mathfrak{h}_j) = t_{i_j-\alpha+1}\mathrm{LM}(\mathfrak{h}_j)$, for $j = 1, \ldots, k+2$, $j \neq \alpha, \alpha+1$, corresponding to nonzero

[13] Recall that we are using the notational convention introduced in Lemma 19. Thus, some entries in the matrix can be equal to 0 or 1.

terms[14] are different. We now compute all leading monomials $\mathfrak{n}_{\alpha,j}$ for any $j \neq \alpha, \alpha+1$ (written as possibly unsorted products):

$$\mathfrak{n}_{\alpha,j} = \begin{cases} z_{i_{k+2}} t_{i_1-\alpha+1} t_{i_2} \cdots t_{i_{k+1}-k+1} & \text{for } j = 1 \\ z_{i_{k+2}} t_{i_1} \cdots t_{i_{j-1}-j+2} t_{i_j-\alpha+1} t_{i_{j+1}-j+1} \cdots t_{i_{k+1}-k+1} & \text{for } 1 < j < k+1 \\ z_{i_{k+2}} t_{i_1} \cdots t_{i_k-k+1} t_{i_{k+1}-\alpha+1} & \text{for } j = k+1 \\ z_{i_{k+1}} t_{i_1} \cdots t_{i_k-k+1} t_{i_{k+2}-\alpha+1} & \text{for } j = k+2 \end{cases}$$

Clearly, $\mathfrak{n}_{\alpha,k+2}$ is different to the others, and the only coincidence can be $\mathfrak{n}_{\alpha,j} = \mathfrak{n}_{\alpha,j^*}$ for some $1 \leq j < j^* \leq k+1$. But this would imply (removing all common terms)

$$t_{i_j-\alpha+1} t_{i_{j+1}-j+1} \cdots t_{i_{j^*-1}-j^*+3} t_{i_j^*-j^*+2} = t_{i_j-j+1} t_{i_{j+1}-j} \cdots t_{i_{j^*-1}-j^*+2} t_{i_j^*-\alpha+1}$$

But due to the inequalities of the indices, the least index in the righthand side, $i_j - j + 1$, can only be canceled out with the term $i_j - \alpha + 1$ in the left hand side, thus implying $j = \alpha$. Similarly, $i_j^* - j^* + 2$ on the left must be the same as $i_j^* - \alpha + 1$ on the right, and then $j^* = \alpha + 1$. But these values of j, j^* are out of the correct range, what shows that no collision among the $\mathfrak{n}_{\alpha,j}$ is actually possible.

For the remaining case, $\alpha = k + 1$, we proceed similarly, defining

$$\mathbf{B} = \begin{pmatrix} t_{i_1} & t_{i_1-1} & \cdots & t_{i_1-k+1} & z_{i_1} & z_{i_1} \\ \vdots & \vdots & \vdots\vdots\vdots & \vdots & \vdots & \vdots \\ t_{i_{k+1}} & t_{i_{k+1}-1} & \cdots & t_{i_{k+1}-k+1} & z_{i_{k+1}} & z_{i_{k+1}} \\ t_{i_{k+2}} & t_{i_{k+2}-1} & \cdots & t_{i_{k+2}-k+1} & z_{i_{k+2}} & z_{i_{k+2}} \end{pmatrix}$$

Now, the syzygy is

$$\sum_{j=1}^{k} (-1)^{k+j} z_{i_j} \mathfrak{h}_j - z_{i_{k+1}} \mathfrak{g}^* + z_{i_{k+2}} \mathfrak{g} = 0$$

where $\mathfrak{h}_j = \Delta(i_1, \ldots, i_{j-1}, i_{j+1}, \ldots, i_{k+2})$. Then

$$\mathfrak{s} = \mathrm{SPOL}(\mathfrak{g}, \mathfrak{g}^*) = z_{i_{k+2}} \mathfrak{g} - z_{i_{k+1}} \mathfrak{g}^* = -\sum_{j=1}^{k} (-1)^{k+j} z_{i_j} \mathfrak{h}_j \tag{5}$$

and for any $j < k + 1$

$$\mathfrak{n}_{k+1,j} = \mathrm{LM}(z_{i_j} \mathfrak{h}_j) = z_{i_j} \mathrm{LM}(\mathfrak{h}_j) = \begin{cases} z_{i_1} z_{i_{k+2}} t_{i_2} t_{i_3-1} \cdots t_{i_{k+1}-k+1} & \text{for } j = 1 \\ z_{i_j} z_{i_{k+2}} t_{i_1} \cdots t_{i_{j-1}-j+2} t_{i_{j+1}-j+1} \cdots t_{i_{k+1}-k+1} & \text{for } 1 < j < k+1 \end{cases}$$

which are clearly different. This shows that again Equation 5 is a reduction to 0 of $\mathrm{SPOL}(\mathfrak{g}, \mathfrak{g}^*)$, which covers all the remaining adjacent pairs of $(k+1)$-minors. □

Now, the theorem statement is a direct consequence of Claims 2 and 3.

The next corollary finally proves that $\mathcal{CI}_{k,d}$ is a hard matrix distribution, that is, the $\mathcal{CI}_{k,d}$-MDDH problem is generically hard in k-linear groups.

Corollary 2. *All nonzero polynomials in $\mathfrak{H}$ have degree at least $k + 1$.*

Proof. According to Lemma 19 the degree of all polynomials $\Delta(\boldsymbol{i})$ for $\boldsymbol{i} \in \binom{k+d}{k+1}$ is exactly $k + 1$. Thus, by Lemma 18 and Theorem 5 the statement follows directly.

[14] Some terms in Equation 4 can be zero due to $t_{i_j-\alpha+1} = 0$, what happens exactly when $i_j - \alpha + 1 \leq 0$ or $i_j - \alpha + 1 \geq d + 2$