Politecnico di Torino

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation
Doctoral Program in Pure and Applied Mathematics (36^{th}cycle)

Cryptographic Applications of Group Actions from Linear Algebra

By

Giuseppe D'Alconzo

******

Supervisors:
Prof. Danilo Bazzanella, Polytechnic of Turin, Italy
Dott. Guglielmo Morgari, Telsy S.p.A., Italy

Doctoral Examination Committee:
Dott. Chiara Marcolla, Referee, TII, United Arab Emirates
Prof. Riccardo Aragona, Referee, University of L'Aquila, Italy
Prof. Antonio Jose Di Scala, Polytechnic of Turin, Italy
Prof. Carlo Sanna, Polytechnic of Turin, Italy
Prof. Lea Terracini, University of Turin, Italy

Politecnico di Torino
2024

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and do not compromise in any way the rights of third parties, including those relating to the security of personal data.

Some results presented in this document are taken from the following articles and preprints.

- *Monomial Isomorphism for Tensors and Applications to Code Equivalence Problems*. G. D'Alconzo. 2024. Designs, Codes and Cryptography. doi.org/10.1007/s10623-024-01375-0.
- *Representations of Group Actions and their Applications in Cryptography*. G. D'Alconzo, A. J. Di Scala. 2023. Preprint ia.cr/2023/1247.
- *Non-interactive Commitment from Non-transitive Group Actions*. G. D'Alconzo, A. Flamini, A. Gangemi. 2023. Asiacrypt 2023. doi.org/10.1007/978-981-99-8739-9-8. (Open access authors' version ia.cr/2023/723)

Giuseppe D'Alconzo
2024

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

Acknowledgements

I would like to acknowledge CrypTO, the group of Cryptography and Number Theory of the Politecnico di Torino. Moreover, I am a member of GNSAGA of INdAM. I acknowledge support from TIM S.p.A. through the PhD scholarship.

Abstract

The need to diversify post-quantum cryptographic assumptions has become relevant after the disruptive attacks to two cryptosystems proposed to NIST's standardization call: the third round proposal Rainbow and the fourth round one SIKE. Those attacks left as a viable option only the lattice and code-based solutions. To provide more variety, the cryptographic community began to look for new computational hard problems on which to base new cryptosystems.
Group actions are becoming a viable option for post-quantum cryptography assumptions. The main assumption in group actions-based systems is the hardness of inverting the map behind the action. In this thesis, we explore the class of group actions derived from linear and multilinear algebra, concerning the general linear group and its subgroups. In particular, we focus on the ones behind the code equivalence and the tensor isomorphism problems.
The contribution of this work is three-fold:

1. we present a polynomial time reduction between code equivalence problems in three different metrics. As an intermediate result, we link these problems to a special variant of the tensor isomorphism problem, where one acting matrix has a special structure;

2. using some tools from representation theory, we investigate some cryptographic assumptions regarding the multiple use of the secret in some cryptographic schemes. We use this technique to cryptanalyze actions from linear and multilinear algebra and their use in some primitives. Moreover, we give some results about action derived by classic groups;

3. finally, we propose a bit commitment scheme based on a newly introduced framework on actions, the group actions with canonical element framework. In short, we use non-transitive actions and distinct orbits to commit to distinct

elements, improving the state of the art of group actions-based bit commitments.

This thesis shows how many aspects of group actions can still be explored, from theory to practice. The field is flourishing, and as a growing topic in recent years, it still presents many problems and open questions to be explored.

Contents

List of Figures

Chapter 1

Introduction

"*Og ég fæ blóðnasir*
En ég stend alltaf upp"
Hoppípolla - Sigur Rós

1.1 Post-Quantum Cryptography

In 1994, a breakthrough event changed cryptography: Peter Shor published an algorithm that, on a quantum computer, solves in polynomial-time all the problems on which cryptography was relying on at those times [81]. Even if a capable quantum computer is still not deployed yet, the cryptographic community has tried to find new hard problems for basing the security of the world's communications. This gave birth to the Post-Quantum Cryptography[1], a branch of the public-key cryptography that deals with problems that are assumed to be hard both for classical and quantum computers.

The two main problems attacked by Shor's algorithm are the factorization and the Discrete Logarithm Problem. The first one asks to find the prime numbers in the factorization of a given large integer n. The second one is a generalization of the computation of the logarithm, but, instead of being in the rational, real or complex field, it must be computed in finite fields of q elements, where q is a prime power.

[1] This term was coined by D. J. Bernstein in 2003 [9].

For instance, given $q = 47$, the logarithm of 42 to the base 2 is given by 32, since $2^{32} \equiv 42 \mod 47$.

Since the beginning of the 2000s, the cryptographic community has started looking for quantum-safe assumptions to build cryptographic systems. Various assumptions have been classified into 5 major groups, based on problems involving lattices, error-correcting codes, multivariate polynomials, hash functions and elliptic curves isogenies.

- **Lattices**. The systems in this category are based on the hardness of finding the shortest element of a lattice, or, given a vector, the element of the lattice nearest to it. Historically, the use of lattices in cryptography started as a cryptanalytic tool [81], to attack the Merkle-Hellman public key encryption system [61].

- **Error-correcting codes**. This can be seen as the *discrete* counterpart of lattices. A linear code is a subspace of a discrete vector space endowed with a metric. Given a code and a vector, the hard problem here is to find the nearest element of the code, and this procedure is called *decoding*. It is known that decoding random codes is a hard problem [19].

- **Multivariate polynomials**. While solving linear systems is a rather simple task, if the degree of the polynomials involved in the system is at least two and the polynomials *look* random, the problem becomes intractable. A large branch of commutative algebra studies the complexity of finding the roots of these objects and provides some tools for these estimates.

- **Hash functions**. It is well-known that hash functions, which are maps that take an (eventually large) string of bits and return a succinct output, are a fundamental tool in symmetric cryptography, but from the hardness of inverting them and to finding collisions, several digital signature schemes have been built since the late 70s [54].

- **Elliptic curves isogenies.** To build a link with classical elliptic curves cryptography, where the objects involved are the points on the curve, here we work with curves and maps between them. An isogeny is a map between two elliptic curves that has some algebraic properties. Given two *isogeneous* elliptic curves, the problem of finding the isogeny between them seems hard, even for a quantum computer.

Each of the above categories has its advantages and disadvantages. The lattices category seems to be one of the most promising since it is well-studied, its performances are good and it provides some advanced functionalities. On the other hand, error-correcting codes provide a strong baseline security, but the sizes of the keys are the worst among all the proposals. From multivariate assumptions, nowadays one can only build digital signatures. Even if they are promising, there is a delicate question about the structure of multivariate systems involved, since more structure guarantees better performance but weakens security. Digital signatures based on hash functions are reliable and versatile since one can use their favorite cryptographic hash. However, the huge amount of hash evaluation undermines their performance. The last and newest category is the isogenies one. Even if some disruptive attacks have been published [22, 75, 59], today the field is flourishing and many constructions can be found. If on one hand it provides the shortest keys sizes, it suffers from the fact that the assumptions are rather new and need some time to gain the cryptographic community's trust.

1.2 The Post-Quantum NIST's calls

In the last 20 years, the advancement of quantum technologies led in 2017 to the start of the first standardization process by the NIST (the USA National Institute of Standards and Technology) to select the next cryptographic algorithms that will replace the ones broken by Shor's algorithm [65]. This call is aimed at (assumed) quantum-resistant algorithms and has two categories: public key encryption (or key encapsulation mechanisms, KEM) and digital signatures. This process is still ongoing, but the first algorithms have been selected. In the KEM category, the lattice-based cryptosystem CRYSTALS-Kyber [17] is the new standard (the selection is still open for a code-based one), while, for digital signatures, there are three proposals: the two lattice-based ones CRYSTALS-Dilithium [35] and Falcon [39], and the hash-based SPHINCS$^+$ [8].
Since the majority of the systems standardized are based on lattices and codes[2], the NIST opened a second call for signature schemes diversification in 2023 [66]. We are still at the beginning of the process, but there are a lot of interesting proposals

[2]There are many relations between lattices and coding problems to the point that they are both friendly called "noisy linear algebra" in [21].

based on multivariate systems, isogenies, and new assumptions about equivalence problems.

1.3 Group Actions-based Cryptography

A group action is formed by a set and a group that permutes the elements of this set. The discrete logarithm problem can be generalized to fit in the framework of group actions cryptography. This branch concerns groups acting on sets where, given two set elements x and y linked by an element g of the group, it is hard to find g. Its introduction in public key cryptography came from the 1991 article of Brassard and Yung [18], but only in the mid-2000s the community started to exploit its potential. Mainly, the framework has been instantiated with group actions from isogenies, like in the work of Couveignes [26]. Recently, in [1], a systematic study of the framework and its assumptions has been proposed.
A subset of cryptographic group actions can be defined based on the commutativity of the underlying group. This algebraic property is crucial and determines the cryptographic design space of the action. For instance, from an Abelian action, one can build a Diffie-Hellman key exchange, following the same steps as the discrete logarithm-based one. Other primitives needing the commutativity property are Oblivious Transfer, Dual-mode public key encryption [1] and group signatures [10]. However, for a generic cryptographic group action, the design space is still non-trivial. Sigma protocols for the knowledge of the above g can be constructed [42] and converted into digital signature schemes via the Fiat-Shamir transform [38]. Recently, signatures with more functionality have been proposed, like (linkable) ring signatures [11], threshold signatures [6] and threshold ring signatures [71].

Apart from the commutativity of the action, other attributes that define the design space are the weak unpredictability, the weak pseudorandomness and pseudorandomness [1]. These properties are additional assumptions needed to build other protocols like Updatable Encryption [55] schemes and perfectly binding bit commitments [30].

There is a variety of actions available from a cryptographic point of view. They range from elliptic curves isogenies [23] to many actions from linear and multilinear algebra, like the ones modeling equivalence problems between linear codes [5, 74], tensors [50], polynomial systems, groups and algebras [45].

1.4 Group Actions from Linear Algebra

In this work, we will focus on cryptographic actions derived from linear and multilinear algebra. For instance, those on which the group acting is a subgroup of the general linear group or direct products of such objects. The first appearance of such group actions is the one related to *Graph Isomorphism*, from [42]. Subsequently, due to Babai's work [3], this has been shown to be insecure for use in cryptography. In 2019, with the digital signature LESS [13] the action linked to the linear code equivalence problem gained a lot of interest.
In the same year, a detailed study on a newly-introduced complexity class was published: in [44], Grochow and Qiao analyzed the relations between various equivalence problems, proving that they are particular instances of the *Tensor Isomorphism* problem, which asks to decide whether two tensors are equivalent under a change of basis. They defined the TI class, containing all the problems that can be reduced to Tensor Isomorphism. Surprisingly, the linear code equivalence problem also falls in this class. Here, we study other code equivalence problems, for instance, the one related to the rank metric (another view of the 3-Tensor Isomorphism) and the sum-rank metric, all modeled by certain group actions. Another problem in TI that has gained a lot of interest in cryptography is the *Alternate Trilinear Form Equivalence* (ATFE) [46], on which a signature has been built recently. Moreover, three proposals of the new NIST's call for digital signatures base their security on group actions-related problems: LESS [4], based on linear code equivalence, MEDS [24], a variant of LESS on the rank metric, and ALTEQ [83], which concerns the equivalence of trilinear forms. Even if two of them concern linear codes, they do not really belong to the error-correcting codes category or any other. However, the best attacks against MEDS and ALTEQ are via algebraic modeling, and, in some sense, we can include them in the multivariate category.

1.5 Organization and Original Results

This thesis studies some actions derived from equivalence problems concerning linear codes, matrix codes and tensors. Multiple aspects are analyzed, mainly from a theoretical point of view, but with some practical implications. The three original works presented concern three different branches: polynomial reductions from com-

putational complexity theory, cryptanalysis of problems from group actions and the design of new cryptographic primitives.
The preliminaries are given in Chapter 2, ranging from cryptographic definitions, group actions, tensors and linear codes in different metrics. Here we introduce the cryptographic assumptions needed in the subsequent chapters.
Chapter 3 shows a polynomial reduction from the code equivalence problem in the sum-rank metric to the same problem in the rank metric. This result is obtained by introducing an intermediate problem called Monomial Isomorphism, a variant of the well-known Tensor Isomorphism where one acting matrix is required to be monomial. The technique adopted is a generalization of the reduction shown in [27], tailored to the tensor setting, using projections and constrains to the rank in order to ensure the special structure of the monomial matrix.
Concerning cryptanalysis, Chapter 4 analyzes the weak pseudorandomness and the weak unpredictability properties for group actions using a newly introduced assumption, the multiple one-way assumption. With the study of this property, some results are given on some group actions that do not satisfy the former ones. In particular, a lot of actions from linear algebra are shown to not be weakly pseudorandom and weakly unpredictable. This analysis is carried out by the use of some tools from representation theory, the definition of a new concept, the representation of a group action, and some metrics on it that are used to obtain our results. In short, we "linearize" the group action embedding the set into a vector space. If the action is already enough linear, then we can use this linearized version to attack the above assumptions. Some actions related to the ones behind LESS, MEDS and ALTEQ are studied and shown to be not weakly psedorandom nor weakly unpredictable, and hence, not capable of certain cryptographic constructions.
In Chapter 5, it is presented a bit commitment scheme based on a particular class of group actions, the non-transitive ones. The previous ones from group actions were *interactive*, i.e. the party who commit needs a first message from the one who receive the commitment. Instead, our proposal is a non-interactive bit commitment scheme. We show its security reducing the hiding property to some known assumptions from the literature. As a concrete instantiation, we present an example based on tensors. However, recently, an attack on this construction has been published and we briefly describe it and its implications.

Chapter 2

Preliminaries

2.1 Notation and Cryptographic Definitions

For a prime power q, $\mathbb{F}_q$ is the finite field with q elements, and $\mathbb{F}_q^n$ is the n-dimensional vector space over $\mathbb{F}_q$. The vector space generated by the vectors $v_1, \ldots, v_k$ is denoted by $\langle v_1, \ldots, v_k \rangle$. We denote the vector space of $n \times m$ matrices with coefficients in $\mathbb{F}_q$ by $\mathbb{F}_q^{n \times m}$. Let $\mathrm{GL}(n, \mathbb{F}_q)$ be the group of invertible $n \times n$ matrices with coefficients in $\mathbb{F}_q$. When the field is implicit, we use $\mathrm{GL}(n)$ instead. A monomial $n \times n$ matrix is given by the product of an $n \times n$ diagonal matrix with non-zero entries on the diagonal, with an $n \times n$ permutation matrix. The $n \times n$ monomial matrices over the field $\mathbb{F}_q$ form a group which is denoted by $\mathrm{Mon}(n, \mathbb{F}_q)$ or $\mathrm{Mon}(n)$, and it is a subgroup of $\mathrm{GL}(n)$. We denote the direct sum of vector spaces $\mathbb{W}_1$ and $\mathbb{W}_2$ by $\mathbb{W}_1 \oplus \mathbb{W}_2$, and its elements are written as (w_1, w_2), where w_i is in $\mathbb{W}_i$. With $\mathscr{S}_t$, we denote the symmetric group over a set of t elements. The transpose of a matrix A is denoted with A^t, and I_ℓ denotes the $\ell \times \ell$ identity matrix.

In the course of this paper, with $\Pr[A]$ we denote the probability of the event A. The security parameter is denoted by λ, this means that the parameters of the cryptographic schemes instantiated with security parameter λ are chosen in such a way that the best-known attack would break the scheme using at least 2^λ operations. A function $\mu(\lambda)$ is *negligible* in λ if, for every positive integer c, there exists a λ_0 such that for each $\lambda > \lambda_0$ we get $\mu(\lambda) < \frac{1}{\lambda^c}$. With $f(n) = \mathsf{poly}(n)$, we denote the fact that there exist two positive integers c and n_0 such that, for every $n > n_0$ we have $f(n) < n^c$. The *big-o* notation is denoted with the capital letter O.

Finally, in the pseudocode "$\leftarrow_{\$}$" denotes the random sampling, "$\leftarrow$" is a variable assignment and "$=$" the equality check.

2.2 Group Actions

This section introduces the algebraic framework of group actions, which can be used to model many cryptographic assumptions from the literature. Throughout this thesis, we will always use groups with multiplicative notation.

Definition 2.2.1. *A group G is said to act on a set X if there is a map $\star : G \times X \to X$ that satisfies the following properties:*

- identity*: if e is the identity element of the group G, then $e \star x = x$ for every x in X;*
- compatibility*: given g and h in G and x in X, we have that $(gh) \star x = g \star (h \star x)$.*

In this case, we say that the triple $(G, X, \star)$ is a group action*.*

Sometimes, when the context is clear, we will denote $\star$ to be the action (of G on X). The *orbit* of the element x in X is the set $\mathcal{O}(x) = \{g \star x \mid g \in G\} \subset X$.

It can be proven that, if for any g in G we define the map

$$\pi_g : X \to X, x \mapsto g \star x,$$

we have that π_g is in $\mathcal{S}_X$, the group of permutations of the elements in X. The correspondence $g \mapsto \pi_g$ is a group homomorphism from G to $\mathcal{S}_X$. If the kernel of this homomorphism is trivial, the action is said *faithful*. If, given any two elements x, y in X there exists g in G such that $y = g \star x$, then the action is said *transitive*. If there exists an element x such that $g \star x = x$ implies $g = e$, then we say that the action is *free*. An action that is both free and transitive is denoted *regular*. For regular actions we have that for every x and y in X there is a *unique* g in G such that $y = g \star x$ and we denote it with $\delta(x, y)$.
Note that, if the action is regular and the group G is finite, then for every x in X the map $g \mapsto g \star x$ is a bijection and we have that $|G| = |X|$.

In the cryptographic settings, as one can see in the following, one property is more important than the others: the commutativity of the group G. We say that the action $(G,X,\star)$ is *abelian* if the group G is abelian.

2.3 Cryptographic Group Actions

Now, we explore the use of group action in cryptography. The first explicit appearance can be found in the 1991 work of Brassard and Yung [18], where the authors studied the use of what they called *one-way group actions*. With the introduction of elliptic curves isogenies, the interest in this framework is grown. Initially, it has been defined in 2006 by Couveignes with the name *hard homogenous spaces* [26], even if in this case the author requires the action to be abelian, as in the isogenies setting. A more formal cryptographic systematization is then presented in 2020 by Alamati, De Feo, Montgomery and Patranabis [1], and we will follow their exposition.

We start defining the properties that a group action should be suitable for computations, and hence, for cryptographic purposes. In other words, we want to easily manipulate the considered objects.

Definition 2.3.1. *A group action $(G,X,\star)$ is said* effective *if:*

- *the group G is finite and there exists a probabilistic polynomial time (*PPT*) algorithm for executing each of the following tasks:*
  1. membership testing*: decide if a bit-string represents a valid element of G;*
  2. equality testing*: given two bit-strings, decide whether they represent the same element of G;*
  3. sampling*: given a distribution $\mathscr{D}_G$ on G, sample an element with respect to $\mathscr{D}_G$.*
  4. operation*: for all $g_1, g_2 \in G$, compute $g_1 g_2$;*
  5. inversion*: compute g^{-1} for all $g \in G$.*
- *The set X is finite and there exists* PPT *algorithms for executing the following tasks:*

1. membership testing*: decide whether a bit-string represents an element in X;*
2. unique representation*: given an element in X, compute a bit-string that canonically represents it.*

- *there exists an efficient algorithm to compute $g \star x$, for each g in G and x in X.*

This definition can be extended to cover other tasks, like sampling from X or computing the unique representation of elements in G.

Informally, a group action is said effective if it can be manipulated easily and it can be computed in practical time. An example of non-effective group actions is the set of polynomials in m variables of bounded degree n over a finite field, with the symmetric group $\mathscr{S}_m$, permuting the variables. It can be seen that the unique representation is given by the algebraic normal form, but it cannot be computed in polynomial time in n and m.

In the rest of this work, even when not explicitly written, we will consider effective group actions.

In [1], it is also defined the concept of *restricted effective group action*, namely those actions where $g \star x$ can be performed efficiently only for a subset of elements g in G. In this thesis, we will not meet these group actions.

2.3.1 Assumptions

The above definitions lead to efficient group actions, which can be used to build cryptographic protocols. However, in order to use them in cryptography, we need to define some suitable computational assumptions. In [1], the authors report some computational assumptions on group actions, for example, the next definition embraces the fact that, given two random elements $x, y \in X$ in the same orbit, then it must be intractable to compute $\delta(x,y)$ if it exists. From now on, λ will be the security parameter and $(G,X,\star)$ will be a group action such that $\log(|G|) = O(\mathsf{poly}(\lambda))$ and $\log(|X|) = O(\mathsf{poly}(\lambda))$.

Definition 2.3.2. *Being D_G and D_X two distributions over G and X respectively, the group action $(G,X,\star)$ is* (D_G, D_X)-one-way *if, for all* PPT *adversaries $\mathscr{A}$, there*

exists a negligible function $\mu(\lambda)$ *such that*

$$\Pr[\mathscr{A}(x, g\star x)\star x = g\star x] \leq \mu(\lambda),$$

where x is sampled according to D_X *and g according to* D_G*.*

We will refer to this assumption as the *one-way group action assumption*.

Example 2.3.3. *The discrete logarithm problem can be seen as an instance of the above assumption. Let* $X = \langle g \rangle$ *be a cyclic group of prime cardinality p and let* $G = (\mathbb{Z}_p^*, \cdot)$*. Then, define* $a \star h = h^a$ *for every* $a \in G$ *and* $h \in X$*. The one-way assumption states that, given* h_1 *and* h_2*, finding a such that* $h_1^a = h_2$ *is intractable.*

Let Π_g be a randomized oracle that, when queried, samples x from D_X and returns $(x, g\star x)$.

Definition 2.3.4. *Being* D_G *and* D_X *two distributions over G and X respectively, the group action* $(G, X, \star)$ *is* (D_G, D_X)-weakly unpredictable *if, for all* PPT *adversaries* $\mathscr{A}$ *having access to the oracle* Π_g*, where g is sampled according to* D_G*, there exists a negligible function* μ *such that*

$$\Pr\left[\mathscr{A}^{\Pi_g}(1^\lambda, y) = g\star y\right] \leq \mu(\lambda).$$

In other words, an action is weakly unpredictable if it is hard to compute $g\star y$ given y and a polynomial number of pairs of the form $(x_i, g\star x_i)$.
Another assumption from [1] that makes use of the oracle Π_g is the following.

Definition 2.3.5. *Being* D_G *and* D_X *two distributions over G and X respectively, the group action* $(G, X, \star)$ *is* (D_G, D_X)-weakly pseudorandom *if, given the randomized oracle U such that, when queried samples x from* D_X*,* σ *uniformly at random from* $\mathscr{S}_X$ *and returns* $(x, \sigma(x))$*, for all* PPT *adversaries* $\mathscr{A}$*, there exists a negligible function* μ *such that*

$$\left|\Pr\left[\mathscr{A}^{\Pi_g}(1^\lambda) = 1\right] - \Pr\left[\mathscr{A}^{U}(1^\lambda) = 1\right]\right| \leq \mu(\lambda),$$

where g is sampled according to D_G*.*

In the above definition, the task of the adversary is to distinguish whether he has access to the oracle that uses the group element g or not.

Now, we introduce a slightly more general assumption that uses the oracle Π_g. It is a variant of the one-wayness where the adversary has access to Π_g and he must retrieve g.

Definition 2.3.6. *Being D_G and D_X two distributions over G and X respectively, the group action $(G,X,\star)$ is (D_G,D_X)-*multiple one-way *if, for all* PPT *adversaries $\mathscr{A}$ having access to the oracle Π_g, where g is sampled according to D_G, there exists a negligible function μ such that*

$$\Pr\left[\mathscr{A}^{\Pi_g}(1^\lambda) \in gN\right] \leq \mu(\lambda),$$

where $N = \{h \in G \mid \forall x \in X, h\star x = x\}$ is the kernel of the induced homomorphism from G to $\mathscr{S}_X$.

The request on the coset of the kernel gN in the above definition allows the adversary to find a different group element g' acting like g. This is needed in case the action is not faithful.

Observe that breaking the multiple one-wayness implies breaking both the weak unpredictability and the weak pseudorandomness. We will use this fact to attack such assumptions.

When we omit the distributions D_G and D_X from definitions 2.3.2, 2.3.4, 2.3.5 and 2.3.6, we use the uniform ones.

Remark 2.3.7. *A similar but stronger treatment of multiple one-way group actions is given in [73], under the name of* transparent security*. The adversary $\mathscr{A}$ has access to a more malleable oracle, called the* transparent oracle*: it acts as Π_g, but, instead of sampling the set element x from D_X, it is queried by $\mathscr{A}$. It can be seen that an adversary with access to a transparent oracle can trivially simulate Π_g sampling x from D_X and then querying it. Therefore, an attack regarding the oracle Π_g can be carried in the context of transparent security while the converse, in general, is not true.*

Observe that particular attention must be given to whether the action is Abelian or not. For actions that are commutative and transitive, seeing a single sample of the form $(x, g\star x)$ is equivalent to seeing a polynomial number of them. In fact, one can produce other random samples picking $h_1,\ldots,h_l$ from G and computing $(h_i\star x, h_i\star(g\star x)) = (y_i, g\star y_i)$, setting $y_i = h_i\star x$ for every i. For instance, an attacker

can simulate the oracle Π_g from a single sample $(x, g \star x)$, this means that breaking the multiple one-wayness directly implies breaking the one-wayness of the action. Since we want to investigate the case whether the latter holds, we set ourselves in the more general non-Abelian scenario.

2.3.2 Non-transitive group actions

Since in Chapter 5 we will focus on cryptographic constructions from non-transitive group actions, here we state some assumptions that can be useful in this setting. The first one is the *Group Action Pseudo Randomness* (GA-PR) problem, defined in [50]. It can be seen as a generalisation of the Decisional Diffie-Hellman assumption. For example, in [50], the authors state that it can be applied to the general linear group action on tensors. Let us now define more formally the problem on which the GA-PR assumption is based. As in the previous subsection, λ is the security parameter and $(G, X, \star)$ will be a group action such that $\log(|G|) = O(\mathsf{poly}(\lambda))$ and $\log(|X|) = O(\mathsf{poly}(\lambda))$.

Definition 2.3.8. *Denote with* pp *the group action* $(G, X, \star)$. *The* group action pseudo random game *(GA-PR) is given in Figure 2.1. We define the* advantage *of an adversary* $\mathscr{A}$ *of GA-PR as*

$$\mathsf{Adv}_{GA\text{-}PR}(\mathscr{A}) = \left| \Pr\left[\mathscr{A} \textit{ wins } \text{GA-PR}(\mathsf{pp})\right] - \frac{1}{2} \right|.$$

The GA-PR assumption *states that for all* PPT *adversaries* $\mathscr{A}$*, there is a negligible function* $\mu(\lambda)$*, with* λ *being the security parameter, such that*

$$\mathsf{Adv}_{GA\text{-}PR}(\mathscr{A}) \leq \mu(\lambda),$$

For the bit commitment scheme presented in Chapter 5, we will refer to the GA-PR assumption when the set X consists of only two orbits. We call this new assumption and the relative game 2GA-PR.

We remark that the adversary of the GA-PR game must be able to distinguish whether the challenger has picked the element t uniformly at random from the orbit of s or from the whole set X. However, when t is picked inside X, it is still possible that

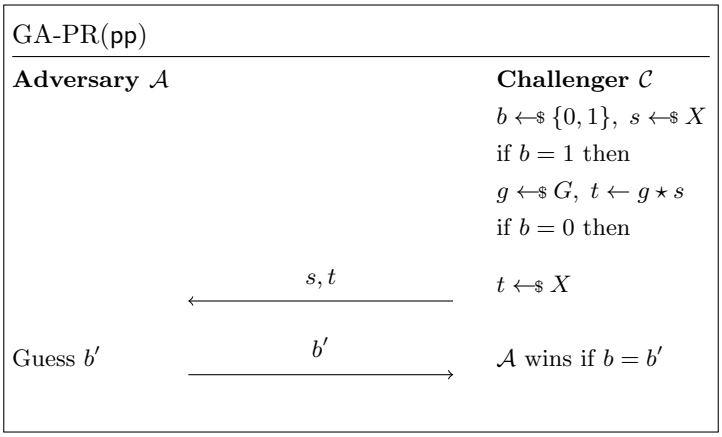

Fig. 2.1 Group Action Pseudo Random game.

t is picked inside the orbit of s as well; therefore, even a computationally unbounded adversary would not be able to win the game with probability 1.

In particular, if we consider the 2GA-PR game, and we suppose that the two orbits have the same cardinality, the event that t is picked uniformly at random inside the set X and t results to be an element in the orbit of s is $\frac{1}{4}$. Therefore, even an adversary with unbounded computational power, who can distinguish whether t lives in the same orbit of s or not, cannot win the game with a probability greater than $\frac{3}{4}$.

The observation above motivates the introduction of an assumption which we refer to as *decisional Group Action Inversion Problem* (dGA-IP). The dGA-IP problem, also known as Isomorphism Problem [50], is the decisional variant of the group action inversion problem presented in [82], applied to the case in which the set X is given by only two orbits. If the restriction on the two orbits is removed, a large number of similar problems can be found in literature [44, 45, 70].

Definition 2.3.9. *The* dGA-IP game *is presented in Figure 2.2, where* pp *is given by the tuple* $(G, X, \star, t_0, t_1)$*, with* t_0 *and* t_1 *elements that lie in distinct orbits under the action of* G*. We define the advantage of an adversary* $\mathscr{A}$ *of dGA-IP as*

$$\mathsf{Adv}_{\textit{dGA-IP}}(\mathscr{A}) = \left| \Pr\left[\mathscr{A} \textit{ wins dGA-IP}(\mathsf{pp})\right] - \frac{1}{2} \right|.$$

The dGA-IP assumption *states that for all* PPT *adversaries* $\mathscr{A}$*, there is a negligible function* $\mu(\lambda)$*, with* λ *being the security parameter, such that*

$$\mathsf{Adv}_{dGA\text{-}IP}(\mathscr{A}) \leq \mu(\lambda),$$

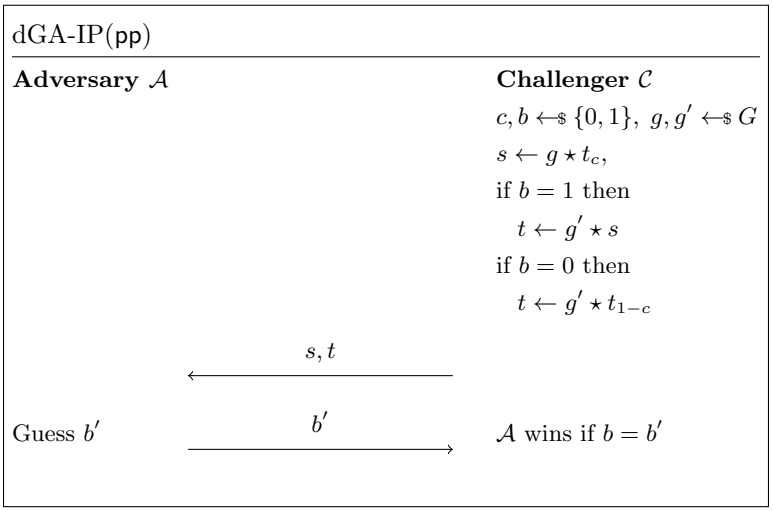

Fig. 2.2 decisional Group Action Inversion Problem game.

This game, compared to 2GA-PR, reflects more clearly the fact that it is hard to distinguish whether two elements in X lie in the same orbit or not, and an adversary with unbounded computational power would win this game with probability 1.

2.4 Coding Theory

In telecommunications, error-correcting codes are a very useful tool: they allow to correct random errors occurring in the transmission of a message over a noisy channel. Once a *code* is chosen, a message is encoded as a codeword and is sent over the channel. Once arrived, the receiver applies the *decoding* procedure to obtain the original message, even if some errors occurred. The number of errors that a code can detect and correct depends on the code itself and the decoding procedure.

Linear codes (i.e. codes that are linear subspaces) play a significant role in post-quantum cryptography: the hardness of decoding a random code is used to ensure the security of many cryptosystems presented to the first standardization call of the NIST. However, in this work, we will make large use of codes without talking about

decoding and errors. We are mainly interested in whether two codes are "essentially the same"; formally, this means that there is an equivalence between them.

2.4.1 Linear codes

A *linear code* $\mathscr{C}$ of dimension k is a linear space of dimension k. Linear codes can be embedded in different linear spaces $\mathbb{V}$ over $\mathbb{F}_q$, depending on their form. A code is endowed with a map *weight* w defined on $\mathbb{V}$

$$\mathrm{w} : \mathbb{V} \to \mathbb{N}$$

such that $\mathrm{w}(x) = 0$ if and only if $x = 0$, i.e it is the zero vector. We can define a metric d from a weight w

$$\mathrm{d} : \mathbb{V} \times \mathbb{V} \to \mathbb{N},\ (x,y) \mapsto \mathrm{w}(y-x).$$

Throughout this paper, we will consider three weights with their corresponding metrics. We highlight that, even if we can endow the same code with two or more different metrics, we always consider a code with just a metric.

We recall the general problem of deciding whether two linear codes are equivalent. Given a weight w and a metric d, we say that an invertible linear map f from the vector space $\mathbb{V}$ to itself preserves the metric (or, equivalently, the weight) if $\mathrm{w}(f(x)) = \mathrm{w}(x)$ for every x in $\mathbb{V}$. We call such maps *linear isometries*, and they form a group with the composition. Two linear codes are *linearly equivalent* if there exists a linear isometry between them. The task of checking if two codes are equivalent is called *Linear Code Equivalence Problem*. Since in the rest of the paper we will consider only linear isometries, sometimes we drop the word "linear" when we talk about isometries or equivalences, in particular, we refer to the problem above as *Code Equivalence* (CE). Its hardness depends on which codes and metric we consider. In the following, we define CE with respect to the three different metrics we saw in the next subsections.

2.4.2 Hamming metric

The weight we present is the *Hamming* weight. Here, we consider linear codes embedded in $\mathbb{F}_q^n$, and we say that the code $\mathscr{C}$ has length n. This weight is defined as the number of non-zero entries of a vector:

$$\mathrm{w_H} : \mathbb{F}_q^n \to \mathbb{N},\ (x_1,\ldots,x_n) \mapsto |\{i \mid x_i \neq 0\}|\,.$$

We refer to the distance induced by $\mathrm{w_H}$ as $\mathrm{d_H}$. A useful representation of a k-dimensional code $\mathscr{C}$ of length n in the Hamming metric is given by its *generator matrix*, a $k \times n$ matrix having a basis $\{v_1,\ldots,v_k\}$ of $\mathscr{C}$ as rows. Notice that the generator matrix is not unique since there are many bases for the same linear code.

We can characterize linear isometries in the Hamming metric, reporting a well-known result from [58].

Proposition 2.4.1. *If $f : \mathbb{F}_q^n \to \mathbb{F}_q^n$ is a linear isometry in the Hamming metric, then there exists an $n \times n$ monomial matrix Q such that $f(x) = xQ$ for all x in $\mathbb{F}_q^n$.*

Then two codes $\mathscr{C}$ and $\mathscr{D}$ are linearly equivalent if there exists a monomial matrix Q such that

$$\mathscr{C} = \left\{yQ \in \mathbb{F}_q^n : y \in \mathscr{D}\right\}.$$

The generator matrix G of a code $\mathscr{C}$ is not unique, hence, for every invertible matrix S, the matrix SG generates the same code $\mathscr{C}$. This must be considered since we state the equivalence problem in terms of generator matrices.

Definition 2.4.2. *The* Hamming Linear Code Equivalence *($\mathsf{CE_H}$) problem is given by*

- input*: two codes $\mathscr{C}$ and $\mathscr{D}$ represented by their $k \times n$ generator matrices G and G', respectively;*
- output*: YES if there exist a $k \times k$ invertible matrix S and an $n \times n$ monomial matrix Q such that $G = SG'Q$, and NO otherwise.*

The search *version is the problem of finding such matrices given two linearly equivalent codes.*

Observe that the matrix S in the above definition models a possible change of basis, while the monomial matrix Q is a permutation and a scaling of the coordinates of the code.

2.4.3 Rank metric

The second weight we consider is defined on matrices. This means that our code $\mathscr{C}$ is a space of matrices and usually we refer to it as a *matrix code*. If we consider $n \times m$ matrices, the code has *length* $n \times m$. The map

$$\mathrm{w}_{\mathrm{rk}} : \mathbb{F}_q^{n\times m} \to \mathbb{N},\ M \mapsto \mathrm{rk}(M)$$

is defined as the rank of the matrix M. Hence, the distance d_{rk} between M_1 and M_2 is given by the rank of the difference $M_2 - M_1$.

From [62], linear isometries for the rank metric can be characterized as follows.

Proposition 2.4.3. *If $f : \mathbb{F}_q^{n\times m} \to \mathbb{F}_q^{n\times m}$ is a linear isometry in the rank metric, then there exist an $n \times n$ invertible matrix A and an $m \times m$ invertible matrix B such that*

1. *$f(M) = AMB$ for all M in $\mathbb{F}_q^{n\times m}$, or*
2. *$f(M) = AM^tB$ for all M in $\mathbb{F}_q^{n\times m}$,*

where the latter case can occur only if $n = m$.

Usually, an isometry can be denoted with a pair of matrices (A,B).

In the literature, for example in [27, 74], the linear equivalence problem for matrix codes is defined taking into account only the first case given in Proposition 2.4.3, even when we have $n = m$. In terms of the computational effort to solve the problem, this is not an issue, since considering both cases requires at most twice the time of considering only the first one, and hence, just a polynomial overhead that we can ignore. For simplicity, we continue the approach from [27, 74] in the following definition.

Definition 2.4.4. *The* rank Linear Code Equivalence *($\mathsf{CE}_{\mathsf{rk}}$) problem is given by*

- input*: two $n \times m$ matrix codes $\mathscr{C}$ and $\mathscr{D}$ of dimension s represented by their bases;*

- output*: YES if there exist matrices A in* $\mathrm{GL}(n)$ *and B in* $\mathrm{GL}(m)$ *such that, for every M in* $\mathscr{D}$*, we have that AMB is in* $\mathscr{C}$*, and NO otherwise.*

The search *version is the problem of finding such matrices given two linearly equivalent codes.*

In the literature, this problem is also called *Matrix Code Equivalence* (MCE).

Given a matrix code $\mathscr{C}$, an *automorphism* of $\mathscr{C}$ is a linear isometry f such that $f(\mathscr{C}) = \mathscr{C}$. We say that $\mathscr{C}$ has *trivial automorphisms* if the only automorphisms of $\mathscr{C}$ are of the form $M \mapsto (\lambda I_n) M (\mu I_m)$ for some non-zero λ, μ in $\mathbb{F}_q$.

2.4.4 Sum-rank metric

The last class of codes we consider is embedded into the direct sum (or Cartesian product) of spaces of matrices. Given positive integers $d, n_1, \ldots, n_d, m_1, \ldots, m_d$, we have that the linear space $\mathbb{V}$ defined above is $\mathbb{F}_q^{n_1 \times m_1} \oplus \cdots \oplus \mathbb{F}_q^{n_d \times m_d}$. We can define the *Sum-rank* weight as the sum of the ranks

$$\begin{array}{rccl} \mathrm{w_{sr}}: & \mathbb{F}_q^{n_1 \times m_1} \oplus \cdots \oplus \mathbb{F}_q^{n_d \times m_d} & \to & \mathbb{N}, \\ & (M_1, \ldots, M_d) & \mapsto & \sum_{i=1}^{d} \mathrm{rk}\,(M_i)\,. \end{array}$$

The distance $\mathrm{d_{sr}}$ induced by $\mathrm{w_{sr}}$ is called *sum-rank metric* and we call a code endowed with this distance a *sum-rank code* of parameters $d, n_1, \ldots, n_d, m_1, \ldots, m_d$.

Observe that the sum-rank metric is both a generalization of the Hamming and the rank distance. For $n_1 = \cdots = n_d = m_1 = \cdots = m_d = 1$, the sum-rank metric coincides with the Hamming metric, and sum-rank codes can be seen as linear codes of length d in $\mathbb{F}_q^d$. If we have $d = 1$, then $\mathrm{d_{sr}}$ is the rank metric, and sum-rank codes are matrix codes of size $n_1 \times m_1$.

The equivalence problem between sum-rank codes was introduced in 2020 by Martínez-Peñas [60]. Before stating the problem, we characterize linear sum-rank isometries. This result is given in [20] and a slightly less general statement can be found in [64, Proposition 4.26].

Proposition 2.4.5. *Let* $f : \mathbb{F}_q^{n_1 \times m_1} \oplus \cdots \oplus \mathbb{F}_q^{n_d \times m_d} \to \mathbb{F}_q^{n_1 \times m_1} \oplus \cdots \oplus \mathbb{F}_q^{n_d \times m_d}$ *be a linear isometry in the sum-rank metric. Then there exists a permutation* σ *in* $\mathscr{S}_d$

such that $n_i = n_{\sigma(i)}$ and $m_i = m_{\sigma(i)}$ for every i, and there exist $\psi_i : \mathbb{F}_q^{n_i \times m_i} \to \mathbb{F}_q^{n_i \times m_i}$ isometries in the rank metric such that

$$f(M_1, \ldots, M_d) = \left(\psi_1(M_{\sigma(1)}), \ldots, \psi_d(M_{\sigma(d)})\right)$$

for each $M_i \in \mathbb{F}_q^{n_i \times m_i}$.

We are ready to state the linear equivalence problem for sum-rank codes. As in the case of $\mathsf{CE}_{\mathsf{rk}}$, we choose to not include the case of transposition of matrices.

Remark 2.4.6. *Observe that, even if for $\mathsf{CE}_{\mathsf{rk}}$ the inclusion of the transposition of matrices has only a polynomial blow-up, this is not the case for $\mathsf{CE}_{\mathsf{rs}}$. In fact, from [64] we can see that the transposition can be seen as the action of $\mathbb{F}_2^d$. This implies that there is an overhead of $O(2^d)$ (at least using a naive approach) between considering or not the transposition of matrices, for example, see [27, Remark 2] for the rank case.*

Recall that, as linear space, a sum-rank code $\mathscr{C}$ of parameters $d, n_1, \ldots, n_d$, $m_1, \ldots, m_d$ and dimension k admits a basis of the form $\{\mathbf{C}_1, \ldots, \mathbf{C}_k\}$ where $\mathbf{C}_i = \left(C_i^{(1)}, \ldots, C_i^{(d)}\right)$ is a tuple of matrices. In particular, $C_i^{(j)}$ is in $\mathbb{F}_q^{n_j \times m_j}$ for each i and j.

Definition 2.4.7. *The* sum-rank Linear Code Equivalence *($\mathsf{CE}_{\mathsf{rs}}$) problem is given by*

- input*: two sum-rank codes $\mathscr{C}$ and $\mathscr{D}$, of parameters $d, n_1, \ldots, n_d, m_1, \ldots, m_d$ and dimension k represented by their bases $\{\mathbf{C}_i\}$ and $\{\mathbf{D}_i\}$, respectively;*
- output*: YES if there exist matrices $A_1, \ldots, A_d, B_1, \ldots, B_d$, where A_i is in $\mathrm{GL}(n_i)$ and B_i is in $\mathrm{GL}(m_i)$, and a permutation σ in $\mathscr{S}_d$ such that*

$$\mathscr{C} = \mathrm{Span}\Big\{ \left(A_1 D_1^{(\sigma(1))} B_1, \ldots, A_d D_1^{(\sigma(d))} B_d\right), \ldots, \left(A_1 D_k^{(\sigma(1))} B_1, \ldots, A_d D_k^{(\sigma(d))} B_d\right) \Big\},$$

  and NO otherwise.

The search *version is the problem of finding such matrices given two linearly equivalent codes.*

This formulation embraces both the previous linear equivalence problems for Hamming and rank metric as special cases. Due to this, we can formulate the next result.

Proposition 2.4.8. *Both* CE_{H} *and* $\mathsf{CE}_{\mathsf{rk}}$ *polynomially reduce to* $\mathsf{CE}_{\mathsf{rs}}$.

A natural question is about the converse, whether problems in the Hamming or the sum-rank metric reduce to $\mathsf{CE}_{\mathsf{rk}}$. It has been shown independently in [27] and [43] that CE_{H} can be reduced to $\mathsf{CE}_{\mathsf{rk}}$, using two different approaches. In [43, Section 5], the reduction uses 3-tensors via an "individualization" argument to force a matrix to be monomial. In [27], given a linear code of dimension k in $\mathbb{F}_q^n$, the reduction defines a matrix code in $\mathbb{F}_q^{k\times(k+n)}$. This approach will be generalized in the setting of d-tensors in Chapter 3, and it will give us some reductions between tensors problem in dimensions higher than 3.

2.5 Tensor Isomorphism

In computational complexity theory, in particular when we consider problems from linear group actions, one problem seems "central" in the sense that many ones from the same field reduce to it. We are referring to the Tensor Isomorphism problem, which asks, given two 3-tensors, to decide if they are the same, apart from a change of basis. Other equivalence problems like group, algebra and graph isomorphism, d-linear form equivalence all reduce to the above one. The centrality of this problem has prompted the definition of a new complexity class in [45] called TI, containing all the decision problems reducible to Tensor Isomorphism.

2.5.1 Tensors

Given a positive integer d, a d-tensor over $\mathbb{F}_q$ is an element of the tensor space $\bigotimes_{i=1}^{d}\mathbb{F}_q^{n_i}$. If we fix the bases $\{e_1^{(i)},\dots,e_{n_i}^{(i)}\}$ for every vector space $\mathbb{F}_q^{n_i}$, we can represent a d-tensor T with respect to its coefficients $T(i_1,\dots,i_d)$ in $\mathbb{F}_q$

$$T=\sum_{i_1,\dots,i_d} T(i_1,\dots,i_d)e_{i_1}^{(1)}\otimes\cdots\otimes e_{i_d}^{(d)}.$$

We say that T has size $n_1 \times \cdots \times n_d$. For example, observe that 1-tensors and 2-tensors can be represented as vectors and matrices, respectively.

A *rank one* (or *decomposable*) tensor is an element of the form $a_1 \otimes \cdots \otimes a_d$, where a_i is in $\mathbb{F}_q^{n_i}$. Given a d-tensor T, its *rank* is the minimal non-negative integer r such that there exist $t_1, \ldots, t_r$ rank one tensors for which $T = \sum_{i=1}^{r} t_i$. In general, computing the rank of a d-tensor is a hard task for $d \geq 3$ [47, 77, 80].

For any a in $\mathbb{F}_q^{n_j}$, the projection to a can be defined. Since we are interested mainly in projections to an element of the basis $e_k^{(j)}$ of $\mathbb{F}_q^{n_j}$, we define

$$\begin{aligned}
&\operatorname{proj}_{e_k^{(j)}} : \mathbb{F}_q^{n_1} \otimes \cdots \otimes \mathbb{F}_q^{n_j} \otimes \cdots \otimes \mathbb{F}_q^{n_d} \to \mathbb{F}_q^{n_1} \otimes \cdots \otimes \mathbb{F}_q^{n_{j-1}} \otimes \mathbb{F}_q^{n_{j+1}} \otimes \cdots \otimes \mathbb{F}_q^{n_d}, \\
&\quad \sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) e_{i_1}^{(1)} \otimes \cdots \otimes e_{i_d}^{(d)} \\
&\quad \mapsto \sum_{\substack{i_1,\ldots,i_{j-1},\\ i_{j+1},\ldots,i_d}} T(i_1,\ldots,i_{j-1},k,i_{j+1},\ldots,i_d) e_{i_1}^{(1)} \otimes \cdots \otimes e_{i_{j-1}}^{(j-1)} \otimes e_{i_{j+1}}^{(j+1)} \otimes \cdots \otimes e_{i_d}^{(d)}.
\end{aligned} \tag{2.1}$$

In other words, we send to zero every component of $\sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) e_{i_1}^{(1)} \otimes \cdots \otimes e_{i_d}^{(d)}$ which does not contain $e_k^{(j)}$, obtaining a $(d-1)$-tensor.

A group action can be defined on the vector space $\mathscr{T} = \bigotimes_{i=1}^{d} \mathbb{F}_q^{n_i}$ of d-tensors of size $n_1 \times \cdots \times n_d$ from the Cartesian product of invertible matrices $G = \mathrm{GL}(n_1) \times \cdots \times \mathrm{GL}(n_d)$ as follows

$$\begin{aligned}
&\star : G \times \mathscr{T} \to \mathscr{T}, \\
&\quad \left((A_1,\ldots,A_d), \sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) e_{i_1}^{(1)} \otimes \cdots \otimes e_{i_d}^{(d)} \right) \\
&\quad \mapsto \sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) A_1 e_{i_1}^{(1)} \otimes \cdots \otimes A_d e_{i_d}^{(d)}.
\end{aligned}$$

It can be shown that the action defined above does not change the rank of a tensor[1]. In particular, this implies that the action of an element in $\mathrm{GL}(n_1) \times \cdots \times \mathrm{GL}(n_{i-1}) \times \mathrm{GL}(n_{i+1}) \times \cdots \times \mathrm{GL}(n_d)$ on the projection $\operatorname{proj}_{e_k^{(i)}}(T)$ of a tensor T has the same rank as $\operatorname{proj}_{e_k^{(i)}}(T)$. We summarize these properties in formulas

[1] However, if we extend the action to non-invertible matrices, this property does not hold: the zero matrix sends every tensor into the zero tensor (which has rank zero by definition).

1. $\mathrm{rk}\left((A_1,\ldots,A_d)\star T\right) = \mathrm{rk}\left(T\right)$,
2. $\mathrm{rk}\left((A_1,\ldots,A_{i-1},A_{i+1},\ldots,A_d)\star \mathrm{proj}_{e_k^{(i)}}(T)\right) = \mathrm{rk}\left(\mathrm{proj}_{e_k^{(i)}}(T)\right)$.

2.5.2 The TI class

The isomorphism problem between tensors has some interesting links and properties in computational complexity theory. Here we recall the formal definition of the problem.

Definition 2.5.1. *The* d-Tensor Isomorphism *(d-TI) problem is given by*

- input*: two d-tensors T_1 and T_2 in $\bigotimes_{i=1}^{d}\mathbb{F}_q^{n_i}$;*
- output*: YES if there exists an element g of $\mathrm{GL}(n_1)\times\cdots\times\mathrm{GL}(n_d)$ such that $T_2 = g\star T_1$ and NO otherwise.*

The search *version is the problem of finding such matrices, given two isomorphic d-tensors.*

If we recall the decision problems *d-Colourability* (d-COL) and d-SAT, it is known that the first integer for which these problems are NP-complete is $d = 3$. In particular, there are polynomial reductions from d-COL to 3-COL and from d-SAT to 3-SAT. The same happens for d-TI and 3-TI, as shown in the following astonishing result from [43].

Theorem 2.5.2. *d-TI and 3-TI are polynomially equivalent.*

Since a lot of different problems can be reduced to d-TI, in the same flavour of the complexity class GI (the set of problems reducible in polynomial time to Graph Isomorphism [52]), the authors of [45] define the TI class.

Definition 2.5.3. *The* Tensor Isomorphism *class (TI) contains decision problems that can be polynomially reduced to d-TI for a certain d. A problem D is said* TI-hard *if d-TI can be reduced to D, for any d. A problem is said* TI-complete *if it is in* TI *and is* TI*-hard.*

It is easy to see that TI is a subset of NP, and we can adapt the AM protocol for Graph Non-Isomorphism [42] and Code Non-Equivalence [70] to show that TI is in coAM. This means that no problem in TI cannot be NP-complete unless the polynomial hierarchy collapses at the second level [16].

2.6 Commitment Schemes

A commitment scheme is a cryptographic scheme that allows one party to commit to a value m by sending a commitment com, and then to reveal m by opening the commitment at a later point in time.

Definition 2.6.1. *A* commitment scheme *on a message space $\mathscr{M}$ is a triple of* PPT *algorithms* $(\mathsf{PGen}, \mathsf{Commit}, \mathsf{Open})$ *such that:*

1. $\mathsf{PGen}(1^\lambda)$ *takes as input a security parameter λ in unary and returns public parameters* pp*;*
2. $\mathsf{Commit}(\mathsf{pp}, m)$ *takes as input the public parameters* pp*, a message m in $\mathscr{M}$ and returns the commitment* com *and the* opening material *r;*
3. $\mathsf{Open}(\mathsf{pp}, m, \mathtt{com}, r)$ *takes as input the public parameters* pp*, the message m, the commitment* com *and the opening material r and returns* **accept** *if* com *is the commitment of m or* **reject** *otherwise.*

In the rest of this work, we omit the public parameters pp in the inputs of Commit and Open.

To be suitable in cryptography, commitment schemes must satisfy the *hiding* and *binding* properties. Hiding means that com reveals nothing about m and binding means that it is not possible to create a commitment com that can be opened in two different ways. These properties are formally defined.

Definition 2.6.2. *Let* $\Pi_{\mathsf{Com}} = (\mathsf{PGen}, \mathsf{Commit}, \mathsf{Open})$ *be a commitment scheme and let* $\mathrm{Hiding}(\Pi_{\mathsf{Com}})$ *be the hiding game represented in Figure 2.3. We define the advantage of an adversary $\mathscr{A}$ of* $\mathrm{Hiding}(\Pi_{\mathsf{Com}})$ *as*

$$\mathsf{Adv}_{\mathrm{Hiding}(\Pi_{\mathsf{Com}})}(\mathscr{A}) = \left| \Pr\left[\mathscr{A} \textit{ wins } \mathrm{Hiding}(\Pi_{\mathsf{Com}})\right] - \frac{1}{2} \right|.$$

A commitment scheme Π_{Com} *is* computationally hiding *if for all* PPT *adversaries* $\mathscr{A}$ *there is a negligible function* $\mu(\lambda)$*, with* λ *being the security parameter, such that*

$$\mathsf{Adv}_{\mathrm{Hiding}(\Pi_{\mathsf{Com}})}(\mathscr{A}, \mathrm{Hiding}(\Pi_{\mathsf{Com}})) \leq \mu(\lambda),$$

If, for every pair m_0, m_1*, the commitments* $\mathtt{com}_0$ *and* $\mathtt{com}_1$ *have the same distribution, where* $(\mathtt{com}_i, r_i) = \mathsf{Commit}(m_i)$ *for* $i = 0, 1$*, we say that the commitment is* perfectly hiding.

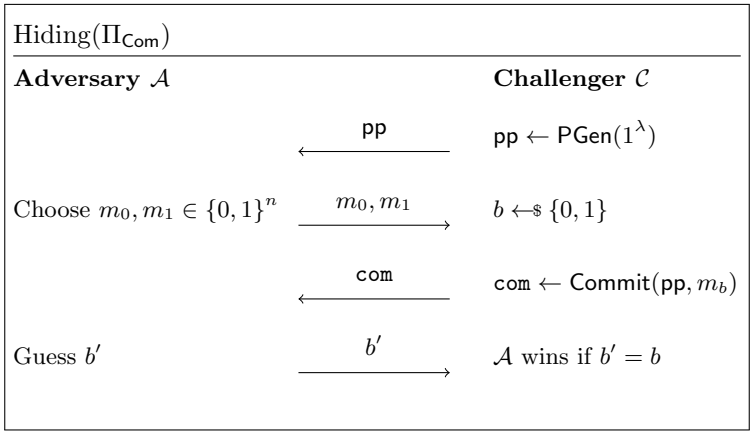

Fig. 2.3 Hiding game for commitment schemes.

Note that, in the case of a bit commitment, the adversary does not send m_0 and m_1, and the bit chosen by the challenger is the committed bit in $\mathtt{com}$.

Definition 2.6.3. *A commitment scheme* $\Pi_{\mathsf{Com}} = (\mathsf{PGen}, \mathsf{Commit}, \mathsf{Open})$ *is* computationally binding *if for all* PPT *adversaries* $\mathscr{A}$ *there is a negligible function* $\mu(\lambda)$*, with* λ *being the security parameter, such that*

$$\Pr\left[\begin{array}{c|c} & m_0 \neq m_1, \\ \mathsf{pp} \leftarrow \mathsf{PGen}(1^\lambda), & \mathsf{Open}(m_0, \mathtt{com}, r_0) = \textbf{accept}, \\ (\mathtt{com}, m_0, r_0, m_1, r_1) \leftarrow \mathscr{A}(\mathsf{pp}) & \mathsf{Open}(m_1, \mathtt{com}, r_1) = \textbf{accept} \end{array}\right] \leq \mu(\lambda).$$

If for every adversary $\mathscr{A}$ *it holds that* $\mu(\lambda) = 0$*, we say that the commitment scheme is* perfectly binding.

Chapter 3

Monomial Isomorphism for Tensors and Applications to Code Equivalence Problems

3.1 Introduction

In this chapter, we investigate the relations between equivalence problems in different metrics. In particular, we extend the knowledge of previously known polynomial reductions between the Hamming, the rank and the sum-rank case. As said in the introduction, this first result has a more theoretical flavor and it is necessary to understand the hardness of some problems in order to use them in cryptography. The whole chapter is based on the work [28] published in Designs, Codes and Cryptography (Springer). Many thanks to Antonio J. Di Scala and Joshua A. Grochow for discussions on this work. The author would like to thank the anonymous reviewers for the valuable comments, which helped to improve the overall quality of this work.

3.1.1 Equivalence problems

An *equivalence problem* is a computational problem where, given two objects A and B of the same nature, it asks whether there exists a map with some properties (an equivalence) sending A to B. Different problems can be stated, depending on the nature of the considered objects or the properties of the map. One of the most well-

known equivalence problems is *Graph Isomorphism*, but in the literature one can find problems concerning groups, quadratic forms, algebras, linear codes, tensors, and many other objects. We will focus on the latter, with the *Code Equivalence* and the *Tensor Isomorphism* problems. An interesting fact is that the isomorphism problem for tensors seems "central" among others. In particular, a large class of equivalence problems can be polynomially reduced to it. In other words, given a pair of objects (groups, algebras, graphs, etc.), a pair of tensors can be built such that they are isomorphic if and only if the starting objects are equivalent. This led to the definition of the complexity class TI in [45]. Different reductions among these problems can be found in [27, 43, 46, 70, 74]. In general, there are no known polynomial algorithms for most of the above problems. Because of this, many public key cryptosystems base their security on the hardness of solving these kinds of problems, for example, *Isomorphism of Polynomials* [68], Code Equivalence [4, 25], Tensor Isomorphism [50], *Lattice Isomorphism* [36], *Trilinear Forms Equivalence* [83], and problems from isogenies of elliptic curves [12, 32, 33].

3.1.2 Code equivalence

One of the most studied equivalence problems concerns linear codes. In the Hamming metric, the maps that generate an equivalence were classified in [58], leading to the *Monomial Equivalence Problem*, which was studied in [70] in the binary case and, in general, in [79]. Worth mentioning is the Support Splitting Algorithm [78], which solves the above problem in *average* polynomial time for a large class of codes over $\mathbb{F}_q$ for $q < 5$. For a detailed analysis, the interested reader can refer to [5]. Recently, the problem of equivalence in different metrics has been studied, and we will focus on the rank metric and the sum-rank one. Concerning the rank metric, the classification of equivalence maps is given in [62], while in [27], the authors analyze the *Matrix Code Equivalence*, and they reduce the Hamming case to it. The same result is given in an independent work [43], where the former problem is called *Matrix Space Equivalence*. In [74], it is shown that Matrix Code Equivalence is polynomially equivalent to problems on bilinear and quadratic maps. Moreover, the link between the rank and the sum-rank metric is studied, leading to a reduction from the latter to the former in a special case. In this chapter, we extend this analysis finding an unconditional reduction from the code equivalence in the sum-rank metric to the rank one.

3.1.3 Original contribution

We give two results of different nature. The first one concerns some relations between tensors problems. The *d-Tensor Isomorphism Problem* (d-TI) asks, given two d-tensors T_1 and T_2, if there are d invertible matrices $A_1, \dots, A_d$ sending T_1 to T_2. We introduce another problem called *d-Tensor Monomial Isomorphism Problem* (d-TI*), where, instead of having d invertible matrices, we require that one of them must be monomial. We show that d-TI* reduces to 3-TI for every $d \geq 4$. To show this, we use techniques from [27], where the authors exhibit a reduction from Monomial Code Equivalence to Matrix Code Equivalence. We reformulate this reduction in terms of tensors, and we generalize it in higher dimensions. In particular, we show that d-TI* is reducible to $(2d-1)$-TI (Theorem 3.2.7), and then, using a result from [43], we get as corollary that d-TI* reduces to 3-TI.
Our techniques are the following: given the reduction Ψ and the $(2d-1)$-tensors $\Psi(T_1)$ and $\Psi(T_2)$, we project to the vector space $\mathbb{W}$ where we expect the action of the monomial matrix. Then, we consider the projected tensor as a 2-tensor in order to compute its rank. We show that some constraints on the rank imply that the matrix acting on $\mathbb{W}$ is monomial.
Observe that the techniques from [43] can be adapted and used as well, but they are less efficient in terms of output dimension since the reduction is looser with respect to the one given in [27].
Another contribution is about the sum-rank code equivalence. Using the result from above, we reduce the problem of deciding whether two sum-rank codes are equivalent to the problem of deciding if two matrix codes are equivalent. Note that a similar result is given in [74] with the assumption that some automorphisms groups are of a given form. While such a hypothesis is mostly satisfied for randomly generated matrix codes (for example the ones used in cryptography [25]), here we give an unconditional reduction. Unfortunately, our reduction produces matrix codes with dimensions and sizes that are polynomially bigger than the starting parameters of the sum-rank codes. In particular, we get a $O(x^6)$ overhead. Due to this result, we can conclude that for the three considered metrics (Hamming, rank, sum-rank), Code Equivalence problems are in the class TI. Figure 5.5 summarizes new and known reductions between code equivalence and other problems, showing the path we used.

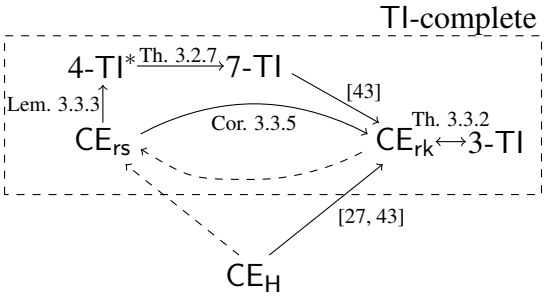

Fig. 3.1 Reduction between problems and TI-completeness. "A → B" indicates that A reduces to B. Dashed arrows denote trivial reductions.

3.2 Monomial Isomorphism Problems

In this section, we will examine the relationship between tensor isomorphism problems when a matrix acting on a specific space is required to be monomial instead of using the action from the entire group $\mathrm{GL}(n_1) \times \cdots \times \mathrm{GL}(n_d)$. Specifically, there exists a j such that the action on the j-th space is given by $\mathrm{Mon}(n_j)$. For simplicity, we will refer to this special space as the last one throughout the remainder of the article and in the problems statements. Since $\mathrm{Mon}(n_d)$ is a subgroup of $\mathrm{GL}(n_d)$, the action of the group $\mathrm{GL}(n_1) \times \cdots \times \mathrm{GL}(n_{d-1}) \times \mathrm{Mon}(n_d)$ on d-tensors is well-defined. When there exists an element g sending the d-tensor T_1 into T_2, we say that they are *monomially isomorphic*.

Definition 3.2.1. *The* Monomial d-Tensor Isomorphism *(d-TI*) problem is given by*

- input*: two d-tensors T_1 and T_2 in $\bigotimes_{i=1}^{d} \mathbb{F}_q^{n_i}$;*
- output*: YES if there exists an element g of $\mathrm{GL}(n_1) \times \cdots \times \mathrm{GL}(n_{d-1}) \times \mathrm{Mon}(n_d)$ such that $T_2 = g \star T_1$ and NO otherwise.*

The search *version is the problem of finding such matrices given two monomially isomorphic d-tensors.*

We recall that, if the action of the monomial matrix is not on the last vector space, we can permute the spaces to obtain the problem above. Observe that the problem 2-TI* is exactly CE$_{\mathsf{H}}$ and the proof that CE$_{\mathsf{H}}$ reduces to CE$_{\mathsf{rk}}$ from [27] can be viewed as a reduction from 2-TI* to 3-TI. In the following, we generalize this approach to reduce d-TI* to $(2d-1)$-TI.

Let $\mathbb{V}_1,\ldots,\mathbb{V}_d$ be vector spaces over $\mathbb{F}_q$ of dimension $n_1,\ldots,n_d$, respectively. Now let $\{v_1^{(j)},\ldots,v_{n_j}^{(j)}\}$ be a basis for the space $\mathbb{V}_j$. We recall that $\mathbb{W}_1 \oplus \mathbb{W}_2$ is the direct sum of vector spaces $\mathbb{W}_1$ and $\mathbb{W}_2$ and its elements are of the form (w_1,w_2). The action of an element of $\mathrm{GL}(\dim(\mathbb{W}_1)+\dim(\mathbb{W}_2))$ is block-by-block:

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} w_1 \\ w_2 \end{pmatrix} = \begin{pmatrix} A_{11}w_1 + A_{12}w_2 \\ A_{21}w_1 + A_{22}w_2 \end{pmatrix}.$$

The reduction we use is the following map, going from a space of d-tensors to a space of $(2d-1)$-tensors,

$$\begin{aligned}
\Psi : \bigotimes_{i=1}^{d} \mathbb{V}_i \to & \left(\bigotimes_{i=1}^{d-1} \mathbb{V}_i\right) \otimes \left(\bigotimes_{i=1}^{d-1} (\mathbb{V}_i \oplus \mathbb{V}_d)\right) \otimes \mathbb{V}_d, \\
& \sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_d}^{(d)} \mapsto \\
& \sum_{\substack{i_1,\ldots,i_d,\\ j_1,\ldots,j_{d-1}}} T(i_1,\ldots,i_d) T(j_1,\ldots,j_{d-1},i_d) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \\
& \qquad \otimes (v_{j_1}^{(1)},0) \otimes \cdots \otimes (v_{j_{d-1}}^{(d-1)},0) \otimes v_{i_d}^{(d)} \\
& + \sum_{i_1,\ldots,i_d} T(i_1,\ldots,i_d) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \otimes (0,v_{i_d}^{(d)}) \otimes \cdots \otimes (0,v_{i_d}^{(d)}) \otimes v_{i_d}^{(d)}.
\end{aligned} \tag{3.1}$$

Example 3.2.2 (Running example)**.** *As an example, consider $d=3$ and a tensors in $\mathbb{F}_2^2 \otimes \mathbb{F}_2^2 \otimes \mathbb{F}_2^3$. The map Ψ became*

$$\begin{aligned}
\Psi : \mathbb{F}_2^2 \otimes \mathbb{F}_2^2 \otimes \mathbb{F}_2^3 \to & \mathbb{F}_2^2 \otimes \mathbb{F}_2^2 \otimes \left(\mathbb{F}_2^2 \oplus \mathbb{F}_2^3\right) \otimes \left(\mathbb{F}_2^2 \oplus \mathbb{F}_2^3\right) \otimes \mathbb{F}_2^3, \\
& \sum_{i,j,k} T(i,j,k) e_i \otimes e_j \otimes e_k \mapsto \\
& \sum_{\substack{i,j,k,\\ i',j'}} T(i,j,k) T(i',j',k) e_i \otimes e_j \otimes (e_{i'},0) \otimes (e_{j'},0) \otimes e_k \\
& + \sum_{i,j,k} T(i,j,k) e_i \otimes e_j \otimes (0,e_k) \otimes (0,e_k) \otimes e_k.
\end{aligned}$$

Given the tensor

$$T_1 = e_1 \otimes e_1 \otimes e_1 + e_2 \otimes e_2 \otimes e_2 + e_1 \otimes e_2 \otimes e_3,$$

its image under Ψ *is given by*

$$\begin{aligned}\Psi(T_1) = {} & e_1\otimes e_1\otimes(e_1,0)\otimes(e_1,0)\otimes e_1 + e_2\otimes e_2\otimes(e_2,0)\otimes(e_2,0)\otimes e_2\\ & + e_1\otimes e_2\otimes(e_1,0)\otimes(e_2,0)\otimes e_3 + e_1\otimes e_1\otimes(0,e_1)\otimes(0,e_1)\otimes e_1\\ & + e_2\otimes e_2\otimes(0,e_2)\otimes(0,e_2)\otimes e_2 + e_1\otimes e_2\otimes(0,e_3)\otimes(0,e_3)\otimes e_3\end{aligned}$$

In the following, we show that two tensors T_1 and T_2 are monomially isomorphic if and only if $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic.

Proposition 3.2.3. *If* T_1 *and* T_2 *are two monomially isomorphic* d*-tensors, then* $\Psi(T_1)$ *and* $\Psi(T_2)$ *are isomorphic as* $(2d-1)$*-tensors.*

Proof. Suppose that T_1 and T_2 are in $\bigotimes_{i=1}^{d}\mathbb{V}_i$ as defined above. Now, since T_1 and T_2 are monomially isomorphic, there exist $d-1$ invertible matrices $A_1,\dots,A_{d-1}$ and a monomial matrix Q such that

$$(A_1,\dots,A_{d-1},Q)\star T_1 = T_2.$$

Let Q be the product of a permutation matrix P corresponding to the permutation σ in $\mathscr{S}_{n_d}$ and a diagonal matrix $D=\operatorname{diag}(\alpha_1,\dots,\alpha_{n_d})$. More explicitly

$$\begin{aligned}&\sum_{i_1,\dots,i_d} T_1(i_1,\dots,i_d)A_1v_{i_1}^{(1)}\otimes\cdots\otimes A_{d-1}v_{i_{d-1}}^{(d-1)}\otimes\alpha_{i_d}v_{\sigma(i_d)}^{(d)}\\ &=\sum_{i_1,\dots,i_d} T_2(i_1,\dots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_d}^{(d)}.\end{aligned}\tag{3.2}$$

Our claim to obtain the thesis is that

$$(A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1},\tilde{Q})\star\Psi(T_1)=\Psi(T_2),$$

where for every $i=1,\dots,d-2$

$$\tilde{A}_i=\begin{pmatrix}A_i & 0\\ 0 & P\end{pmatrix},$$

while

$$\tilde{A}_{d-1}=\begin{pmatrix}A_{d-1} & 0\\ 0 & PD^{-1}\end{pmatrix},\quad\text{and}\quad\tilde{Q}=PD^2$$

Consider T_2, and, for a k in $\{1,\dots,n_d\}$, we write its projection to $v_k^{(d)}$

$$\mathrm{proj}_{v_k^{(d)}}(T_2) = \sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_{d-1},k) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)}. \tag{3.3}$$

Combining Eq. (3.2) and Eq. (3.3), we have

$$\begin{aligned} &\sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_{d-1},k) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \\ &= \sum_{i_1,\dots,i_{d-1}} \alpha_{\sigma^{-1}(k)} T_1(i_1,\dots,i_{d-1},\sigma^{-1}(k)) A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_{d-1} v_{i_{d-1}}^{(d-1)} \end{aligned} \tag{3.4}$$

We define ι to be the canonic injection of $\bigotimes_{i=1}^{d-1} \mathbb{V}_i$ into $\bigotimes_{i=1}^{d-1} (\mathbb{V}_i \oplus \mathbb{V}_d)$, and we consider $\mathrm{proj}_{v_k^{(d)}}(T_2) \otimes \iota \left(\mathrm{proj}_{v_k^{(d)}}(T_2) \right)$, that is

$$\begin{aligned} &\sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_{d-1},k) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \\ &\otimes \sum_{j_1,\dots,j_{d-1}} T_2(j_1,\dots,j_{d-1},k) (v_{j_1}^{(1)},0) \otimes \cdots \otimes (v_{i_{d-1}}^{(d-1)},0) \end{aligned}$$

and, applying Eq. (3.4) two times, it is equal to

$$\begin{aligned} &\sum_{\substack{i_1,\dots,i_{d-1},\\ j_1,\dots,j_{d-1}}} \alpha^2_{\sigma^{-1}(k)} T_1(i_1,\dots,i_{d-1},\sigma^{-1}(k)) T_1(j_1,\dots,j_{d-1},\sigma^{-1}(k)) \\ &\qquad A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_{d-1} v_{i_{d-1}}^{(d-1)} \otimes (A_1 v_{i_1}^{(1)},0) \otimes \cdots \otimes (A_{d-1} v_{i_{d-1}}^{(d-1)},0). \end{aligned} \tag{3.5}$$

Observe that, if we tensorize this element with $v_k^{(d)}$ and we take the sum over $k = 1,\dots,n_d$, we have the first term of $(A_1,\dots,A_{d-1},\tilde{A_1},\dots,\tilde{A}_{d-1},\tilde{Q}) \star \Psi(T_1)$, that is equal to the first term of $\Psi(T_2)$.
To complete the proof we compute the second term of $(A_1,\dots,A_{d-1},\tilde{A_1},\dots,\tilde{A}_{d-1},\tilde{Q}) \star \Psi(T_1)$, and we show that it is equal to the second one of $\Psi(T_2)$. In fact, using Eq. (3.4), we have

$$\begin{aligned} &\sum_{i_d} \sum_{i_1,\dots,i_{d-1}} T_1(i_1,\dots,i_d) A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_1 v_{i_{d-1}}^{(d-1)} \\ &\qquad \otimes (0, v_{\sigma(i_d)}^{(d)}) \otimes (0, v_{\sigma(i_d)}^{(d)}) \otimes (0, \alpha_{i_d}^{-1} v_{\sigma(i_d)}^{(d)}) \otimes \alpha_{i_d}^2 v_{\sigma(i_d)}^{(d)} = \\ &\sum_{i_d} \sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_d) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \otimes (0, v_{i_d}^{(d)}) \otimes \cdots \otimes (0, v_{i_d}^{(d)}) \otimes v_{i_d}^{(d)}. \end{aligned} \tag{3.6}$$

The first and the second terms of $(A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1},\tilde{Q})\star\Psi(T_1)$ are equal to the ones of $\Psi(T_2)$, and we can conclude that

$$(A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1},\tilde{Q})\star\Psi(T_1)=\Psi(T_2).$$

To complete the proof we observe that matrices $A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1}$ and $\tilde{Q}$ are invertible by construction, hence $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic as $(2d-1)$-tensors. □

Example 3.2.4 (Running example)**.** *Consider the tensor T_1 from Example 3.2.2 under the action of matrices*

$$A=\begin{pmatrix}1&0\\0&1\end{pmatrix},\quad B=\begin{pmatrix}0&1\\1&0\end{pmatrix},\quad C=\begin{pmatrix}0&0&1\\0&1&0\\1&0&0\end{pmatrix}.$$

We obtain the monomially isomorphic tensor

$$T_2=(A,B,C)\star T_1=e_1\otimes e_2\otimes e_3+e_2\otimes e_1\otimes e_2+e_1\otimes e_1\otimes e_1$$

and it can be seen that $\Psi(T_1)$ is isomorphic to $\Psi(T_2)$ via the matrices $(A,B,\tilde{A},\tilde{B},\tilde{C})$, where

$$\tilde{A}=\begin{pmatrix}A&0\\0&C\end{pmatrix},\quad \tilde{B}=\begin{pmatrix}A&0\\0&C\end{pmatrix},\quad \tilde{C}=C$$

as in the proof of Proposition 3.2.3.

Now we show the converse.

Proposition 3.2.5. *If $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic, then T_1 and T_2 are monomially isomorphic.*

Proof. Since $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic, there exist invertible matrices

$$A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1},\tilde{Q}\in \mathrm{GL}(n_1)\times\cdots\times\mathrm{GL}(n_{d-1}+n_d)\times\mathrm{GL}(n_d)$$

such that

$$(A_1,\dots,A_{d-1},\tilde{A}_1,\dots,\tilde{A}_{d-1},\tilde{Q})\star\Psi(T_1)=\Psi(T_2).$$

We want to exhibit $d-1$ invertible matrices $A'_1,\dots,A'_{d-1}$ and a monomial matrix Q' such that $(A'_1,\dots,A'_{d-1},Q')\star T_1 = T_2$. In particular, we will show that $A'_i = A$ for every $i = 1,\dots,d-1$. First, we claim that $\tilde{Q}$ is a monomial matrix. Consider $(I_{n_1},\dots,I_{n_{d-1}},I_{n_1+n_d},\dots,I_{n_{d-1}+n_d},\tilde{Q})\star\Psi(T_1)$ and use $\tilde{Q}v_{i_d}^{(d)} = \sum_{j=1}^{n_d}\tilde{Q}_{j,i_d}v_j^{(d)}$

$$\begin{aligned}
&\sum_{\substack{i_1,\dots,i_d,\\ j_1,\dots,j_{d-1}}} T_1(i_1,\dots,i_d)T_1(j_1,\dots,j_{d-1},i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\\
&\qquad\otimes(v_{j_1}^{(1)},0)\otimes\cdots\otimes(v_{j_{d-1}}^{(d-1)},0)\otimes\sum_{k=1}^{n_d}\tilde{Q}_{k,i_d}v_k^{(d)}\\
&+\sum_{i_1,\dots,i_d} T_1(i_1,\dots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\otimes(0,v_{i_d}^{(d)})\otimes\cdots\otimes(0,v_{i_d}^{(d)})\otimes\sum_{k=1}^{n_d}\tilde{Q}_{k,i_d}v_k^{(d)}.
\end{aligned} \tag{3.7}$$

If we project it to $v_k^{(d)}$ along the last space $\mathbb{V}_d$ we obtain

$$\begin{aligned}
&\sum_{\substack{i_1,\dots,i_d,\\ j_1,\dots,j_{d-1}}} \tilde{Q}_{k,i_d}T_1(i_1,\dots,i_d)T_1(j_1,\dots,j_{d-1},i_d)v_{i_1}^{(1)}\\
&\qquad\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\otimes(v_{i_1}^{(1)},0)\otimes\cdots\otimes(v_{i_{d-1}}^{(d-1)},0)\\
&\qquad+\sum_{i_1,\dots,i_d}\tilde{Q}_{k,i_d}T_1(i_1,\dots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\otimes(0,v_{i_d}^{(d)})\otimes\cdots\otimes(0,v_{i_d}^{(d)}).
\end{aligned} \tag{3.8}$$

Now consider Eq. (3.8) as a 2-tensor in $\left(\bigotimes_{i=1}^{d-1}\mathbb{V}_i\right)\otimes\left(\bigoplus_{i=1}^{d-1}(\mathbb{V}_i\oplus\mathbb{V}_d)\right)$. With this new view, we obtain

$$\sum_{i_d}\tilde{Q}_{k,i_d}\left[\left(\sum_{i_1,\ldots,i_{d-1}} T_1(i_1,\ldots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\right)\right.$$
$$\left.\otimes\left(\sum_{j_1,\ldots,j_{d-1}} T_1(j_1,\ldots,j_{d-1},i_d)(v_{j_1}^{(1)},0)\otimes\cdots\otimes(v_{j_{d-1}}^{(d-1)},0)\right)\right]$$
$$+\sum_{i_d}\tilde{Q}_{k,i_d}\left(\sum_{i_1,\ldots,i_{d-1}} T_1(i_1,\ldots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\right)\otimes(0,v_{i_d}^{(d)})\otimes\cdots\otimes(0,v_{i_d}^{(d)})=$$
$$\sum_{i_d}\tilde{Q}_{k,i_d}\left[\left(\sum_{i_1,\ldots,i_{d-1}} T_1(i_1,\ldots,i_d)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\right)\otimes\right.$$
$$\left.\left(\sum_{j_1,\ldots,j_{d-1}} T_1(j_1,\ldots,j_{d-1},i_d)(v_{j_1}^{(1)},0)\otimes\cdots\otimes(v_{j_{d-1}}^{(d-1)},0)+(0,v_{i_d}^{(d)})\otimes\cdots\otimes(0,v_{i_d}^{(d)})\right)\right], \tag{3.9}$$

having rank at most the number of non-zero elements of $\tilde{Q}_{k,\cdot}$, the k-th row of the matrix $\tilde{Q}$, but at least 1 since $\tilde{Q}$ is invertible. Now consider the action of $(A_1,\ldots,A_{d-1},\tilde{A}_1,\ldots,\tilde{A}_{d-1})$ on this tensor: the rank remains the same. If we repeat this process for $\Psi(T_2)$, we obtain the following rank-1 tensor in $\left(\bigotimes_{i=1}^{d-1}\mathbb{V}_i\right)\otimes\left(\bigoplus_{i=1}^{d-1}(\mathbb{V}_i\oplus\mathbb{V}_d)\right)$

$$\left(\sum_{i_1,\ldots,i_{d-1}} T_2(i_1,\ldots,i_{d-1},k)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}\right)\otimes$$
$$\left(\sum_{j_1,\ldots,j_{d-1}} T_2(j_1,\ldots,j_d,k)(v_{i_1}^{(1)},0)\otimes\cdots\otimes(v_{i_{d-1}}^{(d-1)},0)+(0,v_k^{(d)})\otimes\cdots\otimes(0,v_k^{(d)})\right). \tag{3.10}$$

From the equality of the ranks, $\tilde{Q}_{k,\cdot}$ must have exactly a non-zero element for each k, and hence, $\tilde{Q}$ is a monomial matrix of the form PD, where $D=\operatorname{diag}(\alpha_1,\ldots,\alpha_{n_d})$ is a diagonal matrix and P is a permutation matrix corresponding to the permutation σ in $\mathscr{S}_{n_d}$.

Without loss of generality, suppose that the permutation σ of the monomial matrix $\tilde{Q}$ is the identity. This avoids the use of σ on the index of $v_{i_d}^{(d)}$. Consider again $\Psi(T_2)$ and its projection to $v_k^{(d)}$ along $\mathbb{V}_d$ as in Eq. (3.10). We project on elements

of the basis of $\bigoplus_{i=1}^{d-1}(\mathbb{V}_i \oplus \mathbb{V}_d)$. For elements of the form $(v_{\ell_1}^{(1)},0)\otimes\cdots\otimes(v_{\ell_{d-1}}^{(d-1)},0)$ we get

$$\begin{aligned}&\mathrm{proj}_{(v_{\ell_1}^{(1)},0)\otimes\cdots\otimes(v_{\ell_{d-1}}^{(d-1)},0)}\left(\mathrm{proj}_{v_k^{(d)}}(\Psi(T_2))\right)=\\&T_2(\ell_1,\ldots,\ell_{d-1},k)\sum_{i_1,\ldots,i_{d-1}}T_2(i_1,\ldots,i_{d-1},k)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}.\end{aligned}\tag{3.11}$$

In particular, it is a multiple of $\sum_{i_1,\ldots,i_{d-1}}T_2(i_1,\ldots,i_{d-1},k)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}$ for every choice of $\ell_1,\ldots,\ell_{d-1}$. When we consider elements different from $(v_{\ell_1}^{(1)},0)\otimes\cdots\otimes(v_{\ell_{d-1}}^{(d-1)},0)$, the projection is always zero, except for the case $(0,v_k^{(d)})\otimes\cdots\otimes(0,v_k^{(d)})$

$$\begin{aligned}&\mathrm{proj}_{(0,v_{i_k}^{(d)})\otimes\cdots\otimes(0,v_{i_k}^{(d)})}\left(\mathrm{proj}_{v_k^{(d)}}(\Psi(T_2))\right)=\\&\sum_{i_1,\ldots,i_{d-1}}T_2(i_1,\ldots,i_{d-1},k)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}.\end{aligned}\tag{3.12}$$

Hence, every projection of $\mathrm{proj}_{v_k^{(d)}}(\Psi(T_2))$ is a multiple of

$$\sum_{i_1,\ldots,i_{d-1}}T_2(i_1,\ldots,i_{d-1},k)v_{i_1}^{(1)}\otimes\cdots\otimes v_{i_{d-1}}^{(d-1)}$$

and the linear space $\mathscr{V}_k$ generated by all the projections is generated by the $(d-1)$-tensor in Eq. (3.12). Consider now the projection to $v_k^{(d)}$ of $(A_1,\ldots,A_{d-1},\tilde{A}_1,\ldots,\tilde{A}_{d-1},\tilde{Q})\star\Psi(T_1)$, that is the $(2d)$-tensor

$$\begin{aligned}&\alpha_k\left(\sum_{i_1,\ldots,i_{d-1}}T_1(i_1,\ldots,i_{d-1},k)A_1v_{i_1}^{(1)}\otimes\cdots\otimes A_{d-1}v_{i_{d-1}}^{(d-1)}\right)\otimes\\&\left(\sum_{j_1,\ldots,j_{d-1}}T_1(j_1,\ldots,j_{d-1},k)\tilde{A}_1(v_{j_1}^{(1)},0)\otimes\cdots\otimes\tilde{A}_{d-1}(v_{j_{d-1}}^{(d-1)},0)+\right.\\&\left.\left(\tilde{A}_1(0,v_k^{(d)})\otimes\cdots\otimes\tilde{A}_{d-1}(0,v_k^{(d)})\right)\right).\end{aligned}\tag{3.13}$$

Again, if we project to any element of the basis of $\bigotimes_{i=1}^{d-1}(\mathbb{V}_i \oplus \mathbb{V}_d)$, we obtain a multiple of the $(d-1)$-tensor

$$\alpha_k \sum_{i_1,\dots,i_{d-1}} T_1(i_1,\dots,i_{d-1},k) A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_{d-1} v_{i_{d-1}}^{d-1}. \tag{3.14}$$

By hypothesis, the space generated by these projections is equal to $\mathscr{V}_k$, the space generated by the same projections of $\Psi(T_2)$, that can be written as

$$\begin{aligned}\mathscr{V}_k &= \langle \sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_{d-1},k) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \rangle \\ &= \langle \alpha_k \sum_{i_1,\dots,i_{d-1}} T_1(i_1,\dots,i_{d-1},k) A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_{d-1} v_{i_{d-1}}^{(d-1)} \rangle.\end{aligned}$$

Hence there exists a non-zero λ_k in $\mathbb{F}_q$ such that

$$\begin{aligned}&\sum_{i_1,\dots,i_{d-1}} T_2(i_1,\dots,i_{d-1},k) v_{i_1}^{(1)} \otimes \cdots \otimes v_{i_{d-1}}^{(d-1)} \\ &= \lambda_k \alpha_k \sum_{i_1,\dots,i_{d-1}} T_1(i_1,\dots,i_{d-1},k) A_1 v_{i_1}^{(1)} \otimes \cdots \otimes A_{d-1} v_{i_{d-1}}^{(d-1)}.\end{aligned} \tag{3.15}$$

Tensorizing Eq. (3.15) with $v_k^{(d)}$ and taking the sum on k, we have that T_1 and T_2 are monomially isomorphic via $(A_1,\dots,A_{d-1},Q')$, where $Q' = D'P$ with $D' = \mathrm{diag}(\lambda_1\alpha_1,\dots,\lambda_{n_d}\alpha_{n_d})$, and hence we have the thesis. □

Example 3.2.6 (Running example)**.** *Recall the tensors $T_1, T_2, \Psi(T_1)$ from examples 3.2.2 and 3.2.4. The tensor*

$$\begin{aligned}\Psi(T_2) = {} & e_1 \otimes e_2 \otimes (e_1,0) \otimes (e_2,0) \otimes e_3 + e_2 \otimes e_1 \otimes (e_2,0) \otimes (e_1,0) \otimes e_2 \\ & + e_1 \otimes e_1 \otimes (e_1,0) \otimes (e_1,0) \otimes e_1 + e_1 \otimes e_2 \otimes (0,e_3) \otimes (0,e_3) \otimes e_3 \\ & + e_2 \otimes e_1 \otimes (0,e_2) \otimes (0,e_2) \otimes e_2 + e_1 \otimes e_1 \otimes (0,e_1) \otimes (0,e_1) \otimes e_1\end{aligned}$$

is isomoprhic to $\Psi(T_1)$ via the invertible matrices $(A,B,\tilde{A},\tilde{B},C)$. We want to prove that T_1 is monomially isomorphic to T_2 via matrices (A,B,C). In particular, we first show that C is monomial.

Let $C = (c_{ij})$ and consider $(I_2, I_2, I_5, I_5, C) \star \Psi(T_1)$

$$\begin{aligned}
&e_1 \otimes e_1 \otimes (e_1,0) \otimes (e_1,0) \otimes (c_{1,1}e_1 + c_{2,1}e_2 + c_{3,1}e_3) \\
&+ e_2 \otimes e_2 \otimes (e_2,0) \otimes (e_2,0) \otimes (c_{1,2}e_1 + c_{2,2}e_2 + c_{3,2}e_3) \\
&+ e_1 \otimes e_2 \otimes (e_1,0) \otimes (e_2,0) \otimes (c_{1,3}e_1 + c_{2,3}e_2 + c_{3,3}e_3) \\
&+ e_1 \otimes e_1 \otimes (0,e_1) \otimes (0,e_1) \otimes (c_{1,1}e_1 + c_{2,1}e_2 + c_{3,1}e_3) \\
&+ e_2 \otimes e_2 \otimes (0,e_2) \otimes (0,e_2) \otimes (c_{1,2}e_1 + c_{2,2}e_2 + c_{3,2}e_3) \\
&+ e_1 \otimes e_2 \otimes (0,e_3) \otimes (0,e_3) \otimes (c_{1,3}e_1 + c_{2,3}e_2 + c_{3,3}e_3).
\end{aligned}$$

Projecting this tensor to e_2 from the basis of the last space $\mathbb{F}_2^3$ gives

$$\begin{aligned}
&c_{2,1}e_1 \otimes e_1 \otimes (e_1,0) \otimes (e_1,0) + c_{2,2}e_2 \otimes e_2 \otimes (e_2,0) \otimes (e_2,0) \\
&+ c_{2,3}e_1 \otimes e_2 \otimes (e_1,0) \otimes (e_2,0) + c_{2,1}e_1 \otimes e_1 \otimes (0,e_1) \otimes (0,e_1) \\
&+ c_{2,2}e_2 \otimes e_2 \otimes (0,e_2) \otimes (0,e_2) + c_{2,3}e_1 \otimes e_2 \otimes (0,e_3) \otimes (0,e_3).
\end{aligned}$$

Now consider the above tensor as a 2-tensor in the space

$$\left(\mathbb{F}_2^2 \otimes \mathbb{F}_2^2\right) \otimes \left(\left(\mathbb{F}_2^2 \oplus \mathbb{F}_2^3\right) \otimes \left(\mathbb{F}_2^2 \oplus \mathbb{F}_2^3\right)\right).$$

We have

$$\begin{aligned}
&c_{2,1}\,(e_1 \otimes e_1) \otimes \left((e_1,0) \otimes (e_1,0) + (0,e_1) \otimes (0,e_1)\right) \\
&+ c_{2,2}\,(e_2 \otimes e_2) \otimes \left((e_2,0) \otimes (e_2,0) + (0,e_2) \otimes (0,e_2)\right) \\
&+ c_{2,3}\,(e_1 \otimes e_2) \otimes \left((e_1,0) \otimes (e_2,0) + (0,e_3) \otimes (0,e_3)\right).
\end{aligned} \tag{3.16}$$

This 2-tensor has rank at most the number of non-zero elements in the row $(c_{2,1}, c_{2,2}, c_{2,3})$. This rank does not change when we apply the remaining part of the action, that is the element $(A, B, \tilde{A}, \tilde{B}, I_3)$. If we take the same projection to e_2 of $\mathbb{F}_2^3$ and the same view as 2-tensor of $\Psi(T_2)$, we obtain the following rank-1 tensor

$$\begin{aligned}
&e_2 \otimes e_1 \otimes (e_2,0) \otimes (e_1,0) + e_2 \otimes e_1 \otimes (0,e_2) \otimes (0,e_2) \\
&= \Big(e_2 \otimes e_1\Big) \otimes \Big((e_2,0) \otimes (e_1,0) + (0,e_2) \otimes (0,e_2)\Big).
\end{aligned} \tag{3.17}$$

Since $(A, B, \tilde{A}, \tilde{B}, C) \star \Psi(T_1) = \Psi(T_2)$, we have that the rank of Eq. (3.16) *is equal to the rank of Eq.* (3.17)*, hence the row $(c_{2,1}, c_{2,2}, c_{2,3})$ has exactly one non-zero element Using the same argument, projecting on different elements of the basis of $\mathbb{F}_2^3$,*

we show that every row of C has one non-zero entry. This shows that C is monomial and we denote with σ be the permutation associated to C.
Now we deal with the last part of the proof, showing that T_1 and T_2 are monomial isomorphic. Consider again Eq. (3.17)*. We can project to elements of the basis of $(\mathbb{F}_2^2 \otimes \mathbb{F}_2^3) \otimes (\mathbb{F}_2^2 \otimes \mathbb{F}_2^3)$. For example, when we project to $(e_2,0) \otimes (e_1,0)$, we have $e_2 \otimes e_1$. Similarly, projecting to $(0,e_2) \otimes (0,e_2)$ produces again $e_2 \otimes e_1$. Other projections to $(0,e_i) \otimes (0,e_j)$ with $i \neq j$, or to mixed elements like $(e_i,0) \otimes (0,e_j)$ give us the zero tensor. In particular, the non-zero projections are multiples of $e_2 \otimes e_1$. We denote the vector space generated by all these projections with $\mathscr{V}_2$. This space must be equal to the span of all the same projections (up to σ) of $(A,B,\tilde{A},\tilde{B},C) \star \Psi(T_1)$. As an example, we first project to $e_{\sigma^{-1}(2)}$ of $\mathbb{F}_2^3$, and then to $(e_1,0) \otimes (e_2,0)$. We obtain a multiple of the 2-tensor*

$$\sum_{i,j} T_1(i,j,2) A e_i \otimes B e_j.$$

The vector space generated by these projections is exactly $\mathscr{V}_2$ since $(A,B,\tilde{A},\tilde{B},C) \star \Psi(T_1)$ is equal $\Psi(T_2)$. In other words,

$$\mathscr{V}_2 = \langle e_2 \otimes e_1 \rangle = \langle \sum_{i,j} T_1(i,j,2) A e_i \otimes B e_j \rangle.$$

Hence, there exists a non-zero scalar λ_2 (in this case equal to 1) such that

$$e_2 \otimes e_1 = \sum_{i,j} T_1(i,j,2) A e_i \otimes B e_j.$$

We repeat the process with other elements of the basis of $\mathbb{F}_2^3$, both for $\Psi(T_2)$ and for $(A,B,\tilde{A},\tilde{B},C) \star \Psi(T_1)$. Then, we tensorise the projections of $\Psi(T_2)$ with e_k and the ones of $(A,B,\tilde{A},\tilde{B},C) \star \Psi(T_1)$ with $e_{\sigma^{-1}(k)}$. Taking the sum on k gives us

$$T_2 = \sum_{k=1}^{3} \sum_{i,j} T_1(i,j,2) A e_i \otimes B e_j \otimes e_{\sigma^{-1}(k)} = (A,B,C) \star T_1.$$

Therefore, T_1 and T_2 are monomially equivalent.

The combination of the two results above gives us the main result of this section.

Theorem 3.2.7. *The problem d-TI* polynomially reduces to $(2d-1)$-TI. Moreover, d-TI* is* TI*-complete.*

Proof. Given an instance (T_1, T_2) of d-TI*, we can build an instance $(\Psi(T_1), \Psi(T_2))$ of $(2d-1)$-TI. If we call an oracle for $(2d-1)$-TI on the latter pair of tensors, then we can decide the original monomial isomorphism: Proposition 3.2.3 shows that $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic if T_1 and T_2 are monomially isomorphic. On the other hand, Proposition 3.2.5 shows that if $\Psi(T_1)$ and $\Psi(T_2)$ are isomorphic, then T_1 and T_2 are monomially isomorphic. Since the map Ψ is polynomially computable, this is a correct and polynomial-time reduction. □

Let us analyze the sizes of the reduction Ψ. It takes a d tensor of size $n_1 \times \cdots \times n_d$ and returns a $(2d-1)$-tensor of size $n_1 \times \cdots \times n_{d-1} \times (n_1+n_d) \times \cdots \times (n_{d-1}+n_d) \times n_d$. We will use this reduction to link Code Equivalence problems in the following section, but this result could be of independent interest and shows how powerful is the TI class [45]. In particular, Theorem 3.2.7 proves that for every d, d-TI* is in the class TI. Moreover, a trivial reduction can be found from d-TI to $(d+1)$-TI* (send T to $T \otimes 1$), hence for $d \geq 4$ we have that d-TI* is TI-complete.

3.3 Relations between Code Equivalence Problems

In this section, we show how to reduce the code equivalence problem for sum-rank codes to the one in the rank metric. A reduction is given in [74], but it assumes that the automorphism group of the obtained rank code is trivial in the sense of Subsection 2.4.3. We recall the technique from [74], and we observe how this kind of reduction (sending a tuple of elements of $\mathbb{F}_q^m$ to a block-diagonal matrix) does not work without the trivial automorphisms assumptions.

Let $\mathscr{C}$ be a sum-rank code with basis $\{\mathbf{C}_1, \ldots, \mathbf{C}_k\}$, where $\mathbf{C}_i = \left(C_i^{(1)}, \ldots, C_i^{(d)}\right)$ is a tuple of matrices. We denote with Φ the map from the set of sum-rank codes to the set of matrix codes used in [74]

$$\Phi(\langle \mathbf{C}_1, \ldots, \mathbf{C}_k \rangle) = \langle W_1, \ldots, W_k \rangle,$$

where W_i is the $(\sum_i n_i) \times (\sum_i n_i)$ block diagonal matrix with the elements of $\mathbf{C}_i$ on the diagonal. We recall that if the automorphisms group of the image of Φ is not

trivial, then, given an isometry in the rank metric, we cannot retrieve an isometry in the sum-rank setting since the two codes are not equivalent.

Example 3.3.1. *Consider the field $\mathbb{F}_2$ and the one-dimensional sum-rank codes $\mathscr{C}$ and $\mathscr{D}$ with parameters $d = 2, n_1 = 3, n_2 = 2, m_1 = m_2 = 2$ generated by*

$$C_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad C_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad D_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad D_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix},$$

respectively. It can be seen that $\mathscr{C}$ and $\mathscr{D}$ are not equivalent since there is not any sum-rank isometry between them: the permutation must be the identity since $n_1 \neq n_2$ and do not exist invertible matrices (A,B) in $\mathrm{GL}(3) \times \mathrm{GL}(2)$ such that AC_1B is in the space generated by D_1 (just look at their ranks). However, if we consider $\Phi(\mathscr{C})$ and $\Phi(\mathscr{D})$, we obtain the two one-dimensional matrix codes generated by

$$C' = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \quad \text{and} \quad D' = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right),$$

respectively. We can see that $\Phi(\mathscr{C})$ and $\Phi(\mathscr{D})$ are equivalent via the isometry given by permutation matrices P_σ and P_τ, where $\sigma = (2\,4)$ is in $\mathscr{S}_5$ and $\tau = (2\,3)$ is in $\mathscr{S}_4$. In fact, $P_\sigma C' P_\tau = D'$. This happens since the automorphisms groups of $\Phi(\mathscr{C})$ and $\Phi(\mathscr{D})$ are not trivial. For example, for $\Phi(\mathscr{C})$ it contains the isometry $(P_{(4\,5)}, P_{(3\,4)})$, where $(4\,5)$ and $(3\,4)$ are permutations in $\mathscr{S}_5$ and $\mathscr{S}_4$, respectively.

The 3-TI problem is equivalent to the Code Equivalence in the rank metric $\mathsf{CE}_{\mathsf{rk}}$ since the former can be stated in terms of matrix spaces, and the admissible maps between these spaces are exactly the isometries used for $\mathsf{CE}_{\mathsf{rk}}$ (see [43]). A sketch of the reduction is the following. To a matrix code $\mathscr{C}$ generated by $C_1, \ldots, C_k$ we associate the 3-tensor in the space $\mathbb{A} \otimes \mathbb{B} \otimes \mathbb{C}$

$$T_{\mathscr{C}} = \sum_{i_1, i_2, i_3} (C_{i_3})_{i_1, i_2} a_{i_1} \otimes b_{i_2} \otimes c_{i_3}.$$

In particular, $\mathbb{A}$ and $\mathbb{B}$ represent the spaces of rows and columns, respectively, while $\mathbb{C}$ is the space representing the dimension of the code (or the elements in the basis). Hence, a matrix can be represented as a 2-tensor in $\mathbb{A}\otimes\mathbb{B}$, and the action $(A,B)\star M$ is the matrix multiplication AMB^t. The action regarding $\mathbb{C}$ is the map sending a k-uple of matrices into another k-uple. Therefore, given two matrix codes $\mathscr{C}$ and $\mathscr{D}$, with bases $C_1,\dots,C_k$ and $D_1,\dots,D_k$, equivalent via (A,B) and such that the invertible matrix M sends the basis $AC_1B,\dots,AC_kB$ to $D_1,\dots,D_k$, the tensors $T_{\mathscr{C}}$ and $T_{\mathscr{D}}$ are isomorphic via (A,B^t,M). The vice versa is obtained similarly and we highlight that there is no overhead in the sizes of tensors and matrix spaces obtained in both directions.

Hence, we can resume the above observation in the following result.

Theorem 3.3.2. *The problem* $\mathsf{CE}_{\mathsf{rk}}$ *is* TI*-complete.*

By the TI-hardness of $\mathsf{CE}_{\mathsf{rk}}$ and since it can be reduced to $\mathsf{CE}_{\mathsf{rs}}$, we get that $\mathsf{CE}_{\mathsf{rs}}$ is TI-hard. If we want to show its TI-completeness, we need to prove that it is in TI, presenting a reduction from a TI-complete problem, for instance 4-TI^*.

Lemma 3.3.3. *The problem* $\mathsf{CE}_{\mathsf{rs}}$ *is polynomially reducible to* 4-TI^*.

Proof. We model a sum-rank code as a 4-tensor. Given a sum-rank code $\mathscr{C}$ with parameters $d,n_1,\dots,n_d,m_1,\dots,m_d$ and basis $\{\mathbf{C}_1,\dots,\mathbf{C}_k\}$, let N be the maximum among $n_1,\dots,n_d$ and M be the maximum among $m_1,\dots,m_d$. For each i from 1 to d, we can embed an $n_i\times m_i$ matrix into an $N\times M$ one, filling it with zeros. Hence, there are d embeddings g_i such that

$$g_i : \mathbb{F}_q^{n_i\times m_i} \to \mathbb{F}_q^{N\times M}.$$

In the rest of the proof, we consider sum-rank codes embedded via the functions g_i, this means that we work with codes having parameters $d,n_i=N,m_i=M$ for every $i=1,\dots,d$. Let $\mathfrak{SR}(d,N,M)$ be the set of sum-rank codes of parameters $d,n_i=N,m_i=M$ and let $\mathbb{A},\mathbb{B},\mathbb{C},\mathbb{D}$ be vector spaces of dimension N,M,k,d with bases $\{a_i\}_i$, $\{b_i\}_i$, $\{c_i\}_i$ and $\{d_i\}_i$, respectively. Here, $\mathbb{A}$ and $\mathbb{B}$ denotes the row and column spaces of the matrices, $\mathbb{C}$ denotes the dimension of the code, while $\mathbb{D}$ models the factors of the sum-rank code. Hence, the code generated by $\{\mathbf{C}_1,\dots,\mathbf{C}_k\}$ can be

seen as the 4-tensor

$$\sum_{i_1,\dots,i_4} \left(C_{i_3}^{(i_4)}\right)_{i_1,i_2} a_{i_1} \otimes b_{i_2} \otimes c_{i_3} \otimes d_{i_4}.$$

The projection to a factor $\mathbb{F}_q^{n_j \times m_j}$ is a matrix code, which can be seen as the 3-tensor

$$\sum_{i_1,i_2,i_3} \left(C_{i_3}^{(j)}\right)_{i_1,i_2} a_{i_1} \otimes b_{i_2} \otimes c_{i_3},$$

where the action of (A,B,M) is intended as the left-right multiplication for A and B^t, while M is a change of basis.

Let $\delta_{i,j}$ be the Kronecker's delta and define the map

$$\begin{aligned}
\Phi :& \mathfrak{SR}(d,N,M) \to \left(\bigoplus_{i=1}^{d} \mathbb{A}\right) \otimes \left(\bigoplus_{i=1}^{d} \mathbb{B}\right) \otimes \left(\bigoplus_{i=1}^{d} \mathbb{C}\right) \otimes \mathbb{D}, \\
& \langle \mathbf{C}_1, \dots, \mathbf{C}_k \rangle \\
& \mapsto \sum_{i_1,\dots,i_4} \left(C_{i_3}^{(i_4)}\right)_{i_1,i_2} (\delta_{i_4,1} a_{i_1}, \dots, \delta_{i_4,d} a_{i_1}) \\
& \otimes (\delta_{i_4,1} b_{i_2}, \dots, \delta_{i_4,d} b_{i_2}) \otimes (\delta_{i_4,1} c_{i_3}, \dots, \delta_{i_4,d} c_{i_3}) \otimes d_{i_4}.
\end{aligned} \tag{3.18}$$

Now we show that sum-rank codes $\mathscr{C}$ and $\mathscr{D}$, with bases $\{\mathbf{C}_1, \dots, \mathbf{C}_k\}$ and $\{\mathbf{D}_1, \dots, \mathbf{D}_k\}$, are equivalent if and only if $\Phi(\mathscr{C})$ and $\Phi(\mathscr{D})$ are monomially isomorphic.

"$\implies$". Suppose that $\mathscr{C}$ and $\mathscr{D}$ are linear equivalent via the matrices $A_1, \dots, A_d$, $B_1, \dots, B_d$ and the permutation σ in $\mathscr{S}_d$. Suppose that, for every i, M_i is the $k \times k$ invertible matrix sending the basis $\{A_i C_j^{(\sigma(i))} B_i\}_j$ to the basis $\{D_j^{(i)}\}_j$. Then we define the matrices

$$\tilde{L} = \begin{pmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_d \end{pmatrix}, \quad \tilde{R} = \begin{pmatrix} B_1^t & 0 & \dots & 0 \\ 0 & B_2^t & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & B_d^t \end{pmatrix},$$

$$\tilde{S} = \begin{pmatrix} M_1 & 0 & \dots & 0 \\ 0 & M_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & M_d \end{pmatrix}, \quad \text{and} \quad \tilde{Q} = P_\sigma.$$

We can see that $(\tilde{L},\tilde{R},\tilde{S},\tilde{Q}) \star \Phi(\mathscr{C}) = \Phi(\mathscr{D})$, in fact

$$\begin{aligned} &\sum_{i_1,\dots,i_4} \left(C_{i_3}^{(i_4)}\right)_{i_1,i_2} (0,\dots,A_{i_1}a_{i_1},\dots,0) \\ &\otimes (0,\dots,B_{i_2}b_{i_2},\dots,0) \otimes (0,\dots,M_{i_3}c_{i_3},\dots,0) \otimes d_{\sigma(i_4)}, \end{aligned} \tag{3.19}$$

and this, by construction, is exactly $\Phi(\mathscr{D})$.

"$\Longleftarrow$". Suppose that $\Phi(\mathscr{C})$ and $\Phi(\mathscr{D})$ are monomially isomorphic via invertible matrices L, R, S and the monomial matrix $Q = DP$. We can see matrices L, R and S as block matrices, for example, we have

$$L = \begin{pmatrix} L_{11} & \dots & L_{1d} \\ L_{21} & \dots & L_{2d} \\ \vdots & \ddots & \vdots \\ L_{d1} & \dots & L_{dd} \end{pmatrix},$$

where L_{ij} is an $N \times N$ matrix for every i and j. Analogously, R and S have the same structure, with blocks of dimension $M \times M$ and $k \times k$, respectively. Now, for simplicity, we will focus on the action of L on $\Phi(\mathscr{C})$, but the same argument can be used for R and S. As in the proof of Proposition 3.2.5, we assume that the matrix Q is the identity matrix, otherwise we need to take care of the permutation σ in the indexes and the scalars of D. We write $\text{proj}_{d_k}\left((L,R,S,Q) \star \Phi(\mathscr{C})\right)$

$$\begin{aligned} &\sum_{i_1,i_2,i_3} \left(C_{i_3}^{(k)}\right)_{i_1,i_2} (L_{1k}a_{i_1},\dots,L_{dk}a_{i_1}) \\ &\otimes (R_{1k}b_{i_2},\dots,R_{dk}b_{i_2}) \otimes (S_{1k}c_{i_3},\dots,S_{dk}c_{i_3}). \end{aligned} \tag{3.20}$$

Consider the same projection of $\Phi(\mathscr{D})$

$$\sum_{i_1,i_2,i_3} \left(D_{i_3}^{(k)}\right)_{i_1,i_2} (0,\dots,a_{i_1},\dots,0) \otimes (0,\dots,b_{i_2},\dots,0) \otimes (0,\dots,c_{i_3},\dots,0), \tag{3.21}$$

this tensor is equal to the one of Eq. (3.20), and this holds for every k. Now consider the tensor

$$v^{(k)}_{\ell_2,\ell_3} = (0,\ldots,\underbrace{b_{\ell_2}}_{k\text{-th}},\ldots,0)\otimes(0,\ldots,\underbrace{c_{\ell_3}}_{k\text{-th}},\ldots,0).$$

The projection to $v^{(k)}_{\ell_2,\ell_3}$ of $\mathrm{proj}_{d_k}(\Phi(\mathscr{D}))$ is given by

$$\sum_{i_1}\left(D^{(k)}_{\ell_3}\right)_{i_1,\ell_2}(0,\ldots,a_{i_1},\ldots,0), \tag{3.22}$$

while, for $(L,R,S,Q)\star\Phi(\mathscr{C})$, we have

$$\sum_{i_1,i_2,i_3}(R_{kk})_{\ell_2,i_2}(S_{kk})_{\ell_3,i_3}\left(C^{(k)}_{i_3}\right)_{i_1,i_2}(L_{1k}a_{i_1},\ldots,L_{dk}a_{i_1}). \tag{3.23}$$

By hypothesis, Eq. (3.22) and Eq. (3.23) are equal. Then, for $\bar{k}\neq k$, we have that $L_{\bar{k}k}=0$. We can use the same argument for R and S, using the following tensors and the projections to them

$$(0,\ldots,\underbrace{a_{\ell_1}}_{k\text{-th}},\ldots,0)\otimes(0,\ldots,\underbrace{c_{\ell_3}}_{k\text{-th}},\ldots,0)\,;$$
$$(0,\ldots,\underbrace{a_{\ell_1}}_{k\text{-th}},\ldots,0)\otimes(0,\ldots,\underbrace{b_{\ell_2}}_{k\text{-th}},\ldots,0).$$

Finally, we obtain that L, R and S are block diagonal of the form

$$L=\begin{pmatrix}L_{11}&0&\ldots&0\\0&L_{22}&\ldots&0\\\vdots&&\ddots&\vdots\\0&\ldots&0&L_{dd}\end{pmatrix},\qquad R=\begin{pmatrix}R_{11}&0&\ldots&0\\0&R_{22}&\ldots&0\\\vdots&&\ddots&\vdots\\0&\ldots&0&R_{dd}\end{pmatrix},$$
$$\text{and}\quad S=\begin{pmatrix}S_{11}&0&\ldots&0\\0&S_{22}&\ldots&0\\\vdots&&\ddots&\vdots\\0&\ldots&0&S_{dd}\end{pmatrix}.$$

Since the matrices L, R and S are invertible, so are the matrices on their diagonal. We can conclude that codes $\mathscr{C}$ and $\mathscr{D}$ are equivalent via matrices $L_{11},\ldots,L_{dd}$, $R^t_{11},\ldots,R^t_{dd}$ and the permutation σ. □

Example 3.3.4. *Let $\mathscr{C}$ be the sum-rank code with parameters $d=2, n_1=3, n_2=m_1=m_2=2$ generated by $\{\mathbf{C}_1,\mathbf{C}_2\}$, where*

$$C_1^{(1)}=\begin{pmatrix}1&1\\0&0\\1&0\end{pmatrix},\quad C_1^{(2)}=\begin{pmatrix}0&0\\0&1\end{pmatrix},\ \textit{and}\ C_2^{(1)}=\begin{pmatrix}1&0\\0&0\\0&0\end{pmatrix},\quad C_2^{(2)}=\begin{pmatrix}0&0\\1&0\end{pmatrix}.$$

After applying the embeddings g_i from above, we can see $\mathscr{C}$ as a sum-rank code with parameters $d=2, n_1=n_2=3, m_1=m_2=2$ and we have

$$C_1^{(1)}=\begin{pmatrix}1&1\\0&0\\1&0\end{pmatrix},\quad C_1^{(2)}=\begin{pmatrix}0&0\\0&1\\0&0\end{pmatrix},\ \textit{and}\ C_2^{(1)}=\begin{pmatrix}1&0\\0&0\\0&0\end{pmatrix},\quad C_2^{(2)}=\begin{pmatrix}0&0\\1&0\\0&0\end{pmatrix}.$$

Using the notation from the previous proof, define $\mathbb{A}=\mathbb{F}_2^3$, $\mathbb{B}=\mathbb{F}_2^2$, $\mathbb{C}=\mathbb{F}_2^2$ and $\mathbb{D}=\mathbb{F}_2^2$. The image of $\mathscr{C}$ under Φ is the following 4-tensor in $(\mathbb{A}\oplus\mathbb{A})\otimes(\mathbb{B}\oplus\mathbb{B})\otimes(\mathbb{C}\oplus\mathbb{C})\otimes\mathbb{D}$

$$\begin{aligned}\Phi(\mathscr{C}) &= (e_1,0)\otimes(e_1,0)\otimes(e_1,0)\otimes e_1 \\ &\quad +(e_1,0)\otimes(e_2,0)\otimes(e_1,0)\otimes e_1 \\ &\quad +(e_3,0)\otimes(e_1,0)\otimes(e_1,0)\otimes e_1 \quad \Big\} C_1^{(1)} \\ &\quad +(e_1,0)\otimes(e_1,0)\otimes(e_2,0)\otimes e_1 \quad \Big\} C_2^{(1)} \\ &\quad +(0,e_2)\otimes(0,e_2)\otimes(0,e_1)\otimes e_2 \quad \Big\} C_1^{(2)} \\ &\quad +(0,e_2)\otimes(0,e_1)\otimes(0,e_2)\otimes e_2. \quad \Big\} C_2^{(2)}\end{aligned}$$

Using the same strategy adopted in the proof of Theorem 3.2.7, and since the map Φ is polynomial-time computable, the above result implies that $\mathsf{CE}_{\mathsf{rk}}$ reduces to 4-TI^*. This fact, combined with Theorem 2.5.2 and Theorem 3.3.2 leads to the following corollary.

Corollary 3.3.5. *The problem $\mathsf{CE}_{\mathsf{rs}}$ is TI-complete. In particular, it is polynomially reducible to $\mathsf{CE}_{\mathsf{rk}}$.*

A “proof” of the above result can be seen in Figure 5.5, showing the path of the reduction from $\mathsf{CE}_{\mathsf{rs}}$ to $\mathsf{CE}_{\mathsf{rk}}$.

Chapter 4

Representations of Group Actions and their Applications in Cryptography

4.1 Introduction

Now, we want to answer the following question, tailored to the group actions-based cryptography.

How many times can I use my secret key?

In the group actions scenario, the secret key is an element g, and the problem of retrieving g from x and $g \star x$ is assumed hard for certain actions. However, some constructions need to exhibit more than one pair of the form $(x, g \star x)$, for the same g. Does the security still hold? We show that some well-known actions linked to cryptography are not safe in this setting, and we provide some tools to analyze this fact. This chapter is based on a joint work with Antonio J. Di Scala and it is currently under review [29]. However, the authors would like to thank the anonymous reviewers for their comments, which helped to improve the overall quality of this work.

4.1.1 Group actions in cryptography

In recent years, the topic of cryptographic group actions has received a lot of attention. One of the main motivations of its study is the fact that this framework provides post-quantum assumptions. The topic was introduced by the seminal articles of Brassard and Yung [18] and Couveignes [26]. Moreover, the work of Couveignes had a focus on elliptic curves isogenies, on which more recent works rely [23, 1]. In the last years, many other cryptographic group actions have been proposed, concerning the general linear group [50, 74, 83], multivariate polynomials [68], lattices [37] and linear codes [5]. This framework enables the design of a lot of primitives; the most famous ones are key exchanges [76, 26, 23] and digital signatures [26, 82, 33]. Notably, the 2023 NIST's call for digital signatures [66] lists three candidates based on group actions in round 1 (MEDS [24], LESS [4] and ALTEQ [83]). The design space provided by these objects is huge, and it depends on the features of the employed action: for general actions in literature, we can find PRFs [1], ring signatures [11], updatable encryption schemes [55] and commitments [18]; with the additional requirement of having a commutative action, we can also build oblivious transfers [1], oblivious PRFs [48], group signatures [10] and verifiable random functions [53].

4.1.2 Original contribution

Given a group action $(G, X, \star)$, it is called *one-way* if the map $\star$ is *non-invertible*: given y and $x = g \star y$, it is hard to find g. This is the main assumption at the core of the majority of the cryptographic constructions. However, many primitives require stronger assumptions than the previous one to prove their security. For example, the *weak unpredictability* (Definition 2.3.4) and the *weak pseudorandomness* (Definition 2.3.5) properties are introduced in [1]. The former can be seen as the impossibility, for a probabilistic and polynomial time (PPT) adversary, to compute a set element x such that $g \star y$ is equal to x for a given y, whenever he sees a polynomial number of pairs $(x_i, g \star x_i)$, for random x_i. On the other hand, an action is weakly pseudorandom if an adversary cannot distinguish whether its input contains a polynomial number of pairs $(x_i, g \star x_i)$ or (x_i, y_i), for random x_i and y_i.

In this work, we analyze when the above properties hold introducing a more general assumption called *multiple one-wayness* (Definition 2.3.6), and we give some tools to estimate their validity. This assumption is a relaxation of the one-way

one, where a polynomial number of pairs of the form $(x, g \star x)$ are given to the adversary, whose goal is to find g. We recall that in this setting, the commutativity of the action is crucial. For actions that are commutative and transitive, seeing a single sample of the form $(x, g \star x)$ is equivalent to seeing a polynomial number of them. In fact, one can produce other random samples picking $h_1, \ldots, h_l$ from G and computing $(h_i \star x, h_i \star (g \star x)) = (y_i, g \star y_i)$, setting $y_i = h_i \star x$ for every i. This means that breaking multiple one-wayness directly implies breaking one-wayness of the action. Since we want to investigate the case whether the latter holds, we set ourselves in the non-Abelian scenario.

To study this new assumption, the main idea is that, if we *linearize* the group action, with non-negligible probability the set $\{x_i\}_i$ forms a basis of a certain linear space. Using the knowledge of elements $\{g \star x_i\}_i$, we can retrieve the secret g. With tools from representation theory, we introduce the concept of *group action representation*, which is given by a classical representation $\rho : G \to \mathrm{GL}(\mathbb{F}_q^n)$ endowed with an injective map $\iota : X \to \mathbb{F}_q^n$ such that they are compatible with the group action, i.e. it must hold that $\rho(g)(\iota(x)) = \iota(g \star x)$. The integer n is called the *dimension* of the representation. Then, we report some theoretical results on representations of group actions and we introduce the *q-linear dimension* of a group action, denoted with $\mathrm{LinDim}_{\mathbb{F}_q}$, given by the minimal integer such that there exists a representation of such dimension

$$\mathrm{LinDim}_{\mathbb{F}_q}(G, X, \star) = \min\left\{\dim_{\mathbb{F}_q}(\rho, \iota) \mid (\rho, \iota) \text{ is a representation of } (G, X, \star)\right\}.$$

We show that, under some hypothesis on the representation and if the q-linear dimension of the group action is polynomial in the security parameter, multiple one-wayness, and hence the weak unpredictability and the weak pseudorandomness assumptions, do not hold. In the Abelian case, this implies that, if this attack is doable, an action that has small linear dimension is not even one-way.
One can see that the requirements of our attack are satisfied by a group action where X is a vector space and $\star$ acts linearly. This implies that a large class of well-known cryptographic group actions are not weakly unpredictable nor weakly pseudorandom. In particular, we present some attacks to the above assumptions for the group actions on linear codes related to the ones underlying the LESS and the MEDS signature schemes, even if this does not impact their security since they rely only on the (non-multiple) one-wayness of the actions. In particular, the actions used in those

schemes involve a systematic form SF. These variants are equivalent to the ones without SF in the case of just one oracle call, while, for more calls, they are not. More generally, since we show that the action on d-tensors does not satisfy the above assumptions, all the actions linked to isomorphism problems in the class TI introduced in [43] are not weakly unpredictable nor weakly pseudorandom. As a practical result, such non-commutative group actions cannot be used in the design of Naor-Reingold PRFs [1], updatable key encryption schemes [55] and primitives that expose an oracle that returns samples of the form $(x, g \star x)$, with a secret g.

As a strictly mathematical result, we provide some bounds on the action of classical groups like the permutation group, the general linear group acting on a vector space, and the cyclic group $\mathbb{Z}_n$ acting on itself. The latter leads to an interesting closed formula that can be of independent interest.

Concurrent works. In [7], the authors model the lattice isomorphism problem as a group action and study its properties. Their approach is similar to ours, even if it is less general and they focus on a particular action. For instance, they define that a distribution on the set X *induces linear independence* whenever the sampled elements, under a certain function, are linearly independent with high probability. We generalize this property in the setting of group actions representations in Definition 4.3.1. Moreover, it is shown that the lattice isomorphism action is not weakly unpredictable nor weakly pseudorandom like we do with the code equivalence and other actions.

4.2 Representations and the Linear Dimension of a group action

In this section, we explore the concept of representations of finite groups when we endow them with an injection of the set X into a vector space. Such injection must be "compatible" with the map $\star$, as we see in the following definition.

Definition 4.2.1. *The pair* (ρ, ι) *is a* representation of the group action $(G, X, \star)$ over $\mathbb{F}$ *if* $\rho : G \to \mathrm{GL}(\mathbb{F}^n)$ *is a homomorphism of groups,* $\iota : X \to \mathbb{F}^n$ *is injective and* $\rho(g)(\iota(x)) = \iota(g \star x)$ *for every g in G and x in X. The integer n is said* dimension *of the representation and is denoted with* $\dim_{\mathbb{F}}(\rho, \iota)$.

Given a group action $(G,X,\star)$ and a representation of G, it is natural to ask whether a compatible injection ι is admitted. In the following, we look for necessary and sufficient conditions for the existence of an injection ι given a representation ρ of G.

Proposition 4.2.2. *Let $(G,X,\star)$ be a group action, let N be the kernel of the homomorphism $G \to \mathscr{S}_X$ and let $\mathscr{O} = X/G$ be the space of orbits of the action of G on X i.e. the quotient of X by the action of G. Let $\rho : G \to \mathrm{GL}(\mathbb{F}_q^n)$ be a linear representation. The following are equivalent*

(i) there is an injection $\iota : X \to \mathbb{F}_q^n$ such that $\rho(g)(\iota(x)) = \iota(g \star x)$ for every g in G and x in X,

(ii) there is a ρ-invariant subspace $\mathrm{V} \subset \mathbb{F}_q^n$ such that

$$\{g \in G : \rho(g)|_{\mathrm{V}} = \mathrm{Id}\} = N$$

and maps $\tau : \mathscr{O} \to X$, $\upsilon : \mathscr{O} \to \mathrm{V}$ such that for all $o \in \mathscr{O}$:

$$\begin{cases} \tau(o) \in o, \\ \rho(G)_{\upsilon(o)} = \rho(G_{\tau(o)}), \\ \text{if } o \neq o' \in \mathscr{O} \text{ then } \rho(G)\upsilon(o) \bigcap \rho(G)\upsilon(o') = \emptyset. \end{cases}$$

Proof. $(i) \implies (ii)$. Let $\mathrm{V} = \mathrm{span}_{\mathbb{F}_q}(\iota(X))$ be the linear subspace generated by the image of ι. If $g \in N$ then $\rho(g)(\iota(x)) = \iota(x)$ for all $x \in X$. So

$$N \subset \{g \in G : \rho(g)|_{\mathrm{V}} = \mathrm{Id}\},$$

and hence $N = \{g \in G : \rho(g)|_{\mathrm{V}} = \mathrm{Id}\}$ because ι is injective.

For each $o \in \mathscr{O}$ choose any element $\tau(o) \in o$ and define υ as follows:

$$\upsilon(o) = \iota(\tau(o)).$$

By construction, we have that $\tau(o)$ is in o. The second condition is as follows:

$$\begin{aligned}\rho(G)_{\upsilon(o)} &= \{\rho(g) : \rho(g)(\upsilon(o)) = \upsilon(o)\} = \\ &= \{\rho(g) : \iota(g \star \tau(o)) = \iota(\tau(o))\} = \\ &= \{\rho(g) : g \star \tau(o) = \tau(o)\} = \\ &= \rho(G_{\tau(o)})\end{aligned}$$

The third condition follows from the injectivity of ι since

$$\rho(G)\upsilon(o) \bigcap \rho(G)\upsilon(o') = \iota(G \star \tau(o)) \bigcap \iota(G \star \tau(o'))\,.$$

$(ii) \implies (i)$. Here we show how to define the injection $\iota : X \to \mathbb{F}_q^n$. Let $\pi : X \to X/G = \mathscr{O}$ be the projection to the space of orbits. Let $x \in X$ be any point and let $o = \pi(x)$ its projection. Let $g \in G$ such that $g \star \tau(o) = x$ and define

$$\iota(x) = \rho(g)(\upsilon(o))\,.$$

First of all notice that $\iota(x)$ is well defined. Indeed if for another $g' \in G$ we have $g' \star \tau(o) = x$ then $g' = g \cdot h$ with $h \in G_{\tau(o)}$. So

$$\begin{aligned}\rho(g')(\upsilon(o)) &= \rho(g \cdot h)(\upsilon(o)) = \\ &= \rho(g)(\rho(h)(\upsilon(o))) = \\ &= \rho(g)(\upsilon(o))\end{aligned}$$

since $\rho(h) \in \rho(G)_{\upsilon(o)}$. Notice that ι is injective by the third condition. Indeed, assume $\iota(x) = \iota(y)$ where

$$x = g_x \star \tau(o) \text{ and } y = g_y \star \tau(o')\,.$$

So $\iota(x) = \iota(y)$ means

$$\rho(g_x)(\upsilon(o)) = \rho(g_y)(\upsilon(o'))\,,$$

and then by the third condition, we get $o = o'$. Moreover, $\rho(g_y^{-1} g_x)$ is in $\rho(G)_{\upsilon(o)}$ and hence $\rho(g_y^{-1} g_x)$ is in $\rho(G_{\tau(o)})$. Then there is $h \in G_{\tau(o)}$ such that $\rho(g_y^{-1} g_x) = \rho(h)$ and so $\rho(h^{-1} g_y^{-1} g_x) = \mathrm{Id}$. Thus $h^{-1} g_y^{-1} g_x$ is in N, which gives $g_x \star \tau(o) = g_y \star \tau(o)$ hence $x = y$ and our ι is indeed injective. Finally, we check that $\rho(g)(\iota(x)) = \iota(g \star x)$

holds for every g in G and x in X. Let $x = g_x \star \tau(o)$ and let g be arbitrary in G, then

$$\begin{aligned}\rho(g)(\iota(x)) &= \rho(g)(\rho(g_x)(\upsilon(o))) \\ &= \rho(gg_x)(\upsilon(o)) \\ &= \iota(gg_x \star \tau(o)) \\ &= \iota(g \star (g_x \star \tau(o))) \\ &= \iota(g \star x).\end{aligned}$$

This completes the proof of the proposition. □

For our analysis, the following metric gives a useful tool in the study of cryptographic assumptions based on group actions.

Definition 4.2.3. *Let $(G,X,\star)$ be a group action. For every finite field $\mathbb{F}_q$, the* q-linear dimension *of $(G,X,\star)$ is the integer*

$$\mathrm{LinDim}_{\mathbb{F}_q}(G,X,\star) = \min\left\{\dim_{\mathbb{F}_q}(\rho,\iota) \mid (\rho,\iota) \text{ is a representation of } (G,X,\star)\right\}.$$

Remark 4.2.4. *Observe that the q-linear dimension is well-defined since the set*

$$S_{\mathbb{F}_q,(G,X,\star)} = \left\{\dim_{\mathbb{F}_q}(\rho,\iota) \mid (\rho,\iota) \text{ is a representation of } (G,X,\star)\right\}$$

is non-empty for every finite field $\mathbb{F}_q$ and every group action $(G,X,\star)$.
Indeed, let $X = \{x_1,\ldots,x_{|X|}\}$ and define $\mathbb{F}_q[X]$ as the vector space of linear combinations of the elements of X

$$\mathbb{F}_q[X] = \left\{\sum_j c_j x_j \;:\; c_j \in \mathbb{F}_q\right\}.$$

It can be shown that the dimension of $\mathbb{F}_q[X]$ over $\mathbb{F}_q$ is $|X|$. Let ι be the map that sends $x_j \in X$ to $x_j \in \mathbb{F}_q[X]$. Moreover, let ρ be the map from G to $\mathrm{GL}(\mathbb{F}_q[X])$ such that $\rho(g)$ is the permutation matrix associated to the invertible map

$$x \mapsto g \star x.$$

Hence, $\rho(g)(\iota(x)) = \rho(g \star x)$ and since $\mathbb{F}_q[X] \cong \mathbb{F}_q^{|X|}$, we have that $|X|$ is in $S_{\mathbb{F}_q,(G,X,\star)}$.

The above remark tells us that the cardinality of $|X|$ is an upper bound for the linear dimension of a group action. Moreover, we can prove the following lower bound.

Proposition 4.2.5. *Let $(G,X,\star)$ be a group action and N the kernel of the homomorphism $G \to \mathscr{S}_X$. For every finite field $\mathbb{F}_q$ it holds that*

$$\mathrm{LinDim}_{\mathbb{F}_q}(G,X,\star) \geq \sqrt{\log_q\left(\frac{|G|}{|N|}\right)}.$$

In particular, when the action is faithful, $\mathrm{LinDim}_{\mathbb{F}_q}(G,X,\star) \geq \sqrt{\log_q(|G|)}$.

Proof. Consider the action of the quotient G/N on X

$$\star_{/N} : (gN,x) \mapsto g \star x.$$

It can be shown that it is indeed a group action and it is faithful. Moreover, if ρ is a representation of G to $\mathbb{F}_q^n$ and ι an injection of X to $\mathbb{F}_q^n$, then ρ can be extended to

$$\tilde{\rho} : G/N \to \mathrm{GL}(\mathbb{F}_q^n),\ gN \mapsto \tilde{\rho}(gN) = \rho(g).$$

It holds that $\tilde{\rho}(gN)(\iota(x)) = \iota(gN \star_{/N} x)$ holds for every gN in G/N and x in X. Since the action of G/N is faithful, $\tilde{\rho}$ is injective. Now we have that $|G/N| = |\tilde{\rho}(G/N)| \leq \left|\mathrm{GL}(\mathbb{F}_q^n)\right|$. The cardinality of $\mathrm{GL}(\mathbb{F}_q^n)$ is given by $\prod_{j=0}^{n-1} q^n - q^j$ and it is upper bounded by q^{n^2}. This implies $|G/N| \leq q^{n^2}$ and hence $n \geq \sqrt{\log_q(|G/N|)}$, leading to the thesis. □

Moreover, whenever the set X is a vector space of dimension n on the field $\mathbb{F}_q$ and the action of G is linear, i.e. $g \star (\lambda_1 x_1 + \lambda_2 x_2) = \lambda_1 (g \star x_1) + \lambda_2 (g \star x_2)$, we have that

$$\mathrm{LinDim}_{\mathbb{F}_q}(G,X,\star) \leq n.$$

As we will see in the next sections, many group actions used in cryptography follow the above structure, and hence, a practical upper bound of the linear dimension is known.

4.3 On Multiple One-Way Group Actions

Here we propose an attack on the assumptions presented in Subsection 2.3.1, and a relation to the linear dimension. In particular, we will attack the multiple one-wayness, and, as a direct consequence, this leads to an attack on both the weak unpredictability and the weak pseudorandomness.

We need the following known combinatorial fact. Given $v_1,\dots,v_k$ uniformly sampled from $\mathbb{F}_q^k$, it is known that they form a basis with probability

$$\prod_{i=1}^{k}(1-q^{-i}) = O(1-q^{-1}).$$

This means that, for the uniform distribution on $\mathbb{F}_q^k$, we have that the sampled elements are linearly independent with non-negligible probability (with respect to k). We need to generalize this fact for a group action $(G,X,\star)$ and a representation (ρ,ι).

Definition 4.3.1. *Given a group action $(G,X,\star)$, a distribution D_X on X and a representation (ρ,ι) of dimension n over $\mathbb{F}_q$, we say that* (ρ,ι) induces linear independence *with respect to D_X if, given $\{x_1,\dots,x_Q\}$ sampled according to D_X, with $Q=\mathsf{poly}(n)$, then there exists a negligible function $\mu(n)$ such that*

$$\Pr\left[\langle \iota(x_1),\dots,\iota(x_Q)\rangle \neq \mathbb{F}_q^n\right] \leq \mu(n).$$

In particular, if X is a vector space, the uniform distribution over X induces a linear independence. Due to the above definition, we can analyze whenever an attacker can retrieve the secret g from a tuple of the form $\{(x_i, g\star x_i)\}_i$.

Definition 4.3.2. *Given the group action $(G,X,\star)$, the representation (ρ,ι) is* admissible *if the following hold*

1. *ι is polynomial time computable;*
2. *a preimage of $\rho(g)$ can be found in polynomial time for every g in G.*

Example 4.3.3. *Let $X=\langle g\rangle$ be a cyclic group of prime cardinality p and let $G=(\mathbb{Z}_p^*,\cdot)$. Then, define $a\star h = h^a$ for every $a\in G$ and $h\in X$. We can define a group*

action representation of p-linear dimension equal to 1 as follows.

$$\rho : \mathbb{Z}_p^* \to \mathbb{Z}_p^* \cong \mathrm{GL}_p(1),\ a \mapsto a$$
$$\iota : \langle g \rangle \to \mathbb{F}_p^1,\ g^a \mapsto a.$$

We can see that, even if finding a preimage of ρ is easy, computing ι means solving the discrete logarithm problem, and hence, this is not an admissible representation for the above action.

Now we are ready to state the attack to the multiple one-way assumption.

Proposition 4.3.4. *Let λ be the security parameter. Given the group action $(G,X,\star)$ and two distributions D_G and D_X over G and X respectively, if there exists a field $\mathbb{F}_q$ and an admissible representation (ρ,ι) which induces linear independence with respect to D_X with $\dim_{\mathbb{F}_q}(\rho,\iota) = \mathsf{poly}(\lambda)$, then the group action is not (D_G,D_X)-multiple one-way.*

Proof. Let $\mathscr{A}$ be the adversary having access to the oracle Π_g. If $n = \mathrm{LinDim}_{\mathbb{F}}(G,X,\star)$, then there exist $\rho : G \to \mathrm{GL}(\mathbb{F}^n)$ and $\iota : X \to \mathbb{F}^n$ such that (ρ,ι) is admissible by hypothesis. The strategy of the adversary is the following.

1. $\mathscr{A}$ performs a number of queries Q to the oracle Π_g until he obtains the set $Y = \{(x_i, g\star x_i)\}_{i=1,\dots,n}$ such that $\{\iota(x_1),\dots,\iota(x_n)\}$ is a basis of $\mathbb{F}^n$.

2. $\mathscr{A}$ evaluates ι on the set Y

$$\{(\iota(x_i),\iota(g\star x_i))\}_i = \{(\iota(x_i),\rho(g)(\iota(x_i)))\}_i.$$

3. Since $\{\iota(x_1),\dots,\iota(x_n)\}$ is a basis of $\mathbb{F}^n$, $\mathscr{A}$ can find the invertible matrix $\rho(g)$ and then inverting ρ, obtaining an element h in G such that $\rho(h) = \rho(g)$.

Let us analyse this strategy. Since $n = \mathsf{poly}(\lambda)$ and the representation induces linear independence, $\mathscr{A}$ requires a polynomial number of queries to retrieve a set Y with non-negligible probability in step 1. Step 2 is polynomial-time since the representation is admissible and ι is evaluated at most $2Q$ times. Moreover, since finding a preimage of $\rho(g)$ is a polynomial-time task, the adversary $\mathscr{A}$ finds an element h of G such that $\rho(g) = \rho(h)$. This implies that the action of h on all the

elements of X coincides with the one of g and h is in the coset gN. Therefore, the action cannot be multiple one-way. □

As a corollary, we easily get the following result.

Corollary 4.3.5. *Let λ be the security parameter. Given the group action $(G,X,\star)$ and two distributions D_G and D_X over G and X respectively, if there exists a field $\mathbb{F}_q$ and an admissible representation (ρ,ι) which induces linear independence with respect to D_X with $\dim_{\mathbb{F}_q}(\rho,\iota) = \mathsf{poly}(\lambda)$, then the group action is not (D_G,D_X)-weakly unpredictable nor (D_G,D_X)-weakly pseudorandom.*

Even if the requirements of the previous propositions are non-trivial, in the next section we show how a large class of group action used in cryptography satisfy them.

4.3.1 Analysis of some group actions from cryptography

Here we propose some representations of known cryptographic group actions, starting from the one concerning linear codes.

The hardness of the code equivalence problem has been used to build different primitives [4, 24]. However, in practice, a slightly different action from the one we define in the following is used, involving the systematic form of matrices. In the rest of the section, we will always refer to the *non*-systematic form variant. We refer to *(Linear) Code Equivalence Problem* as the following one: given two linearly equivalent linear codes $\mathscr{C}$ and $\mathscr{C}'$, find an isometry between them. This problem can be rephrased in the setting of group actions.

Definition 4.3.6. *Let $G = \mathrm{GL}(\mathbb{F}_q^k) \times \mathrm{Mon}(\mathbb{F}_q^n)$, where* Mon *is the group of monomial matrices, and let $X = \mathbb{F}_q^{k\times m}$ be the set of $k \times m$ matrices with coefficients in $\mathbb{F}_q$. The* (Linear) Code Equivalence Problem *asks, on inputs M, M' in X, to find (S,R) in G such that $M' = SMR$.*
The action underlying this problem is given by $(G,X,\star)$, where

$$\star : G \times X \to X,\ ((S,R),M) \mapsto SMR.$$

The map $\star$ for the above definition is given by the left-right multiplication of the two matrices S and R.

Remark 4.3.7. *Observe that, even if for one sample (M, SMR) the code equivalence problems with and without the systematic form are equivalent, the scenario changes when more samples are involved and it is not known if this equivalence still holds. In practice, the version with the systematic form is adopted for efficiency reasons: the group that acts on the set is only* $\mathrm{Mon}(\mathbb{F}_q^n)$*, and hence, it has a shorter bit representation.*

Corollary 4.3.8. *The group action of the Code Equivalence Problem is not weak unpredictable nor weak pseudorandom.*

Proof. We will show that this action is not multiple one-way and consequently, we get the thesis.
Since the space of $k \times n$ generator matrices is a vector space of dimension kn, we can see it as $\mathbb{F}^{kn}$ and ι is the natural bijection. Since G is the product $\mathrm{GL}(\mathbb{F}^k) \times \mathrm{Mon}(\mathbb{F}^n)$, we define the representation ρ as follows

$$\rho : G \to \mathrm{GL}(\mathbb{F}^{kn}),\ (S,R) \mapsto S \otimes R^T,$$

where $\otimes$ denotes the Kronecker product. It can be seen that $\rho(g)(\iota(x)) = \iota(g \star x)$ for every g in G and x in X. Moreover, the computation of ι is polynomial time and such is finding a preimage of $\rho(S,R)$. Indeed, let $A = S \otimes R^T$ and divide A in $n \times n$ blocks. Let (i,j) be such that the block $A_{(i,j)}$ is non-zero and set $R' = A^T_{(i,j)}$. Now compute S' as follows. Let u and v be two indexes such that R'_{uv} is non-zero. Then, for every $i, j = 1, \ldots, k$

$$S'_{ij} = \frac{A_{(i,j)uv}}{R'_{uv}}.$$

In this way, we found a pair (S', R') such that the image through ρ is the same as $\rho(S,R)$ and, observing that computing S' and R' is a polynomial time task, we can apply Proposition 4.3.4 and Corollary 4.3.5 to get the thesis. □

Another problem having a linked group action that raised interest is the Tensor Isomorphism Problem. It received a lot of attention both from a theoretical point of view [43] and from a cryptographic point of view [50, 30].

Definition 4.3.9. *Let d be a positive integer. Let $G = \Pi_{i=1}^d \mathrm{GL}(\mathbb{F}_q^{n_i})$ and let $X = \bigotimes_{i=1}^d \mathbb{F}_q^{n_i}$ be the set of d-tensors with coefficients in $\mathbb{F}_q$. The map $\star : G \times X \to X$ is*

defined as

$$\star : \left((A_1, \ldots, A_d), \sum_{i_1,\ldots,i_d} T_{i_1,\ldots,i_d} e_1 \otimes \cdots \otimes e_d \right) \mapsto \sum_{i_1,\ldots,i_d} T_{i_1,\ldots,i_d} A_1 e_1 \otimes \cdots \otimes A_d e_d.$$

The d-Tensor Isomorphism Problem *asks, on inputs* T, T' *in* X*, to find* $(A_1, \ldots, A_d)$ *in* G *such that* $T' = (A_1, \ldots, A_d) \star T$.

Corollary 4.3.10. *The action of the* d*-Tensor Isomorphism is not weak unpredictable nor weak pseudorandom.*

Proof. The set of d-tensors in $\mathbb{F}^{n_1} \otimes \cdots \otimes \mathbb{F}^{n_d}$ is a vector space of dimension $N = n_1 \cdots n_d$. Therefore, ι is the natural bijection. The representation ρ is the Kronecker product of matrices

$$\rho : G \to \mathrm{GL}(\mathbb{F}^N),\ (A_1, \ldots, A_d) \mapsto A_1 \otimes \cdots \otimes A_d$$

and it can be inverted iteratively with the computation from the proof of Corollary 4.3.8; consider $A_1 \otimes (A_2 \otimes \cdots \otimes A_d)$ and find matrices A_1' in $\mathrm{GL}(\mathbb{F}^{n_1})$ and B_1 $\mathrm{GL}(\mathbb{F}^{N/n_1})$ such that

$$A_1' \otimes B_1 = A_1 \otimes \cdots \otimes A_d.$$

Then, we find A_2' in $\mathrm{GL}\left(\mathbb{F}^{n_2}\right)$ and B_2 $\mathrm{GL}\left(\mathbb{F}^{\frac{N}{n_1 n_2}}\right)$ for which the following holds

$$A_2' \otimes B_2 = B_1.$$

Proceeding in this way, we find $A_1', \ldots, A_d'$ such that

$$A_1' \otimes \cdots \otimes A_d' = A_1 \otimes \cdots \otimes A_d.$$

Hence, we have the thesis using Proposition 4.3.4 and Corollary 4.3.5. □

Due to the TI-completeness of d-Tensors Isomorphism [43], all the group actions derived from problems in TI cannot be multiple one-way. In particular, the action on matrix codes from [24] and the one on trilinear forms from [83]. This is easy to see and we analyze the reductions between equivalence problems arising from group actions.

Suppose we have two group actions $(G,X,\star)$ and $(G',X',\star')$ and a polynomial time reduction $\Phi : X \to X'$ such that, for every x,y in X

$$\exists g \in G \text{ such that } g \star x = y \iff \exists g' \in G' \text{ such that } g' \star' \Phi(x) = \Phi(y). \quad (4.1)$$

Even if these kinds of reductions concern decision problems, most of the time they can be viewed as reductions between search problems, for instance like the ones in [43, 41]. If so, we define

$$\mathscr{R}_\Phi = \{(g,g') \in G \times G' \mid g \star x = y \iff g' \star' \Phi(x) = \Phi(y)\,,\ \forall x,y \in X\}$$

and we denote with G'_Φ the projection of $\mathscr{R}_\Phi$ to the second coordinate. Then, there is a pair of maps

$$f_\Phi : G \to G'_\Phi,\ g \mapsto f_\Phi(g)$$

and

$$f'_\Phi : G'_\Phi \to G,\ g' \mapsto f'_\Phi(g')$$

such that both $(g, f_\Phi(g))$ and $(f_\Phi(g'), g')$ are in $\mathscr{R}_\Phi$. With this notation, we can conclude that the reduction Φ induces the following equation

$$\Phi(g \star x) = f_\Phi(g) \star' \Phi(x).$$

Let us go back to group actions representations. Given a polynomial reduction Φ between $(G,X,\star)$ and $(G',X',\star')$ as in Eq. (4.1) and given a representation (ρ',ι') for $(G',X',\star')$, we have that the tuple $\{x_i, g \star x_i\}$ is sent to $\{\Phi(x_i), f_\Phi(g) \star' \Phi(x)\}$. Using Proposition 4.3.4, we retrieve $f_\Phi(g)$ in G', and this implies the following result.

Theorem 4.3.11. *Let $(G,X,\star)$ and $(G',X',\star')$ be two group actions. Suppose that there exist two polynomial-time computable maps $\Phi : X \to X'$ and $f'_\Phi : G'_\Phi \to G$, with $G'_\Phi \subseteq G'$, such that $g' \star \Phi(x) = \Phi(y)$ if and only if $f'_\Phi(g') \star x = y$. Then if $(G',X',\star')$ is not multiple one-way then neither $(G,X,\star)$ is multiple one-way. As an application, group actions derived from equivalence problems in the class* TI *for which there exists a polynomial reduction Φ to the d-Tensors Isomorphism Problem having a polynomial-time f'_Φ cannot be weakly unpredictable nor weakly pseudorandom.*

Proof. Assuming that $(G',X',\star')$ is not multiple one-way, we show that the action $(G,X,\star)$ is not multiple one-way. Calling the oracle Π_g for $(G,X,\star)$ multiple times, we can apply the map Φ to the samples $\{x_i, g\star x_i\}$ to obtain $\{\Phi(x_i), g'\star'\Phi(x)\}$, for a certain g' in G'. In this way, we can retrieve g' and, after applying f'_Φ, we can recover $h = f'_\Phi(g')$ the coset gN of the kernel N. This breaks the multiple one-way assumption for $(G,X,\star)$.

Since the d-Tensor Isomorphism problem is TI-complete, Corollary 4.3.10 implies that any group actions derived from equivalence problems in the class TI for which there exists a reduction Φ to the d-Tensors Isomorphism Problem having a polynomial-time f'_Φ cannot be weakly unpredictable nor weakly pseudorandom. □

Observe that many reductions from [43, 41] satisfy the hypotheses of Theorem 4.3.11, hence, it is safe to avoid any of these group actions in the design of primitives requiring weak unpredictability or weak pseudorandomness.

4.4 On the Linear Dimension of some Classical Groups

4.4.1 The symmetric group $\mathscr{S}_n$

Let $\mathscr{S}_n$ be the symmetric group in n letters $x_1,\cdots,x_n$, i.e. it is the group of all bijections of the set $\mathrm{X}_n = \{x_1,\cdots,x_n\}$. The action is the trivial one, let τ be in $\mathscr{S}_n$ and x_j be in X_n. We define $\tau\star x_j = x_{\tau(j)}$.

Surprisingly, the $n-2$ dimensional representation $\rho : S_n \to \mathrm{GL}(\mathbb{F}_p^{n-2})$ of the symmetric group S_n, when p divides n, stated by L.E. Dickson in [34, Theorem, page 123] does not admit a compatible injection ι. We show that, in general, the linear dimension of the symmetric group is $n-1$.

Proposition 4.4.1. *For $n > 2$ we have*

$$\mathrm{LinDim}_{\mathbb{F}_q}(\mathscr{S}_n, \mathrm{X}_n) = n-1\,.$$

For $n = 2$:

$$\mathrm{LinDim}_{\mathbb{F}_q}(\mathscr{S}_2, \mathrm{X}_2) = \begin{cases} 2 & \textit{if } 2 \,|\, q, \\ 1 & \textit{otherwise}. \end{cases}$$

Proof. First we show that $\mathrm{LinDim}_{\mathbb{F}_q}(\mathscr{S}_n, \mathrm{X}_n) \geq n-1$. Indeed, assume that $d = \mathrm{LinDim}_{\mathbb{F}_q}(\mathscr{S}_n, \mathrm{X}_n) \leq n-2$. Let ρ be a representation $\rho : \mathscr{S}_n \to \mathrm{GL}(\mathbb{F}_q^d)$ and let $\imath : X \to \mathbb{F}_q^d$ be an injective map such that

$$\rho(\tau)(\imath(x_j)) = \imath(\tau \star x_j)$$

for all $\tau \in \mathscr{S}_n, x_j \in \mathrm{X}_n$.

We have that the vectors of the set $B = \{\imath(x_1), \cdots, \imath(x_d)\}$ are either linearly independent or one of them is a linear combination of the others. Assume that $\imath(x_j)$ is a linear combination of the other vectors of B. Namely,

$$\imath(x_j) = \sum_{s \neq j, 1 \leq s \leq d} c_s \imath(x_s),$$

where the coefficients c_s are in $\mathbb{F}_q$.

Let τ from $\mathscr{S}_n$ be the transposition between x_j and x_n. Then

$$\begin{aligned} \rho(\tau)(\imath(x_j)) &= \rho(\tau)\left(\sum_{s \neq j, 1 \leq s \leq d} c_s \imath(x_s)\right) \\ &= \sum_{s \neq j, 1 \leq s \leq d} c_s \rho(\tau)\imath(x_s) \\ &= \sum_{s \neq j, 1 \leq s \leq d} c_s \imath(\tau \star x_s) \\ &= \sum_{s \neq j, 1 \leq s \leq d} c_s \imath(x_s) \\ &= \imath(x_j). \end{aligned}$$

So $\rho(\tau)(\imath(x_j)) = \imath(\tau \star x_j) = \imath(x_n) = \imath(x_j)$ which is a contradiction. Then, the vectors of B are linearly independent and they form a basis of $\mathbb{F}_q^d$. But then $\imath(x_{n-1})$ is a linear combination of vectors of B and we can use a transposition between x_{n-1} and x_n to get a contradiction as above. So $\mathrm{LinDim}_{\mathbb{F}_q}(\mathscr{S}_n, \mathrm{X}_n) \geq n-1$.

Now let $\rho_n : \mathscr{S}_n \to \mathrm{GL}(\mathbb{F}_q^n)$ be the standard representation. Namely,

$$\rho_n(\sigma)(e_i) = e_{\sigma(i)}$$

where $\{e_1, \ldots, e_n\}$ is the canonical basis of $\mathbb{F}_q^n$. Observe that the vector $u = \sum_{j=1}^n e_j$ is invariant by ρ_n, so we get a representation

$$\tilde{\rho}_n : \mathscr{S}_n \to \mathrm{GL}(\mathbb{F}_q^n/\mathbb{F}_q u)$$

on the quotient linear space $\mathbb{F}_q^n/\mathbb{F}_q u \cong \mathbb{F}_q^{n-1}$:

$$\tilde{\rho}_n(\sigma)(\pi(v)) := \pi(\rho_n(\sigma)(v))$$

where $\pi : \mathbb{F}_q^n \to \mathbb{F}_q^n/\mathbb{F}_q u$ is the projection to the quotient. Let us define $\iota : \mathrm{X}_n \to \mathbb{F}_q^n/\mathbb{F}_q u$ as

$$\iota(x_j) := \pi(e_j).$$

Then $\iota(x_j) = \iota(x_s)$ if and only if $e_j = e_s + \lambda u$, with λ in $\mathbb{F}_q$. Thus, for $n \geq 3$ the map ι is injective. Let us check that

$$\tilde{\rho}_n(\tau)(\iota(x_j)) = \iota(\tau \star x_j)$$

for all τ in $\mathscr{S}_n$ and x_j in X_n. We have

$$\begin{aligned}
\tilde{\rho}_n(\tau)(\iota(x_j)) &= \pi(\rho_n(\tau)(\iota(x_j))) \\
&= \pi(\rho_n(\tau)(e_j)) \\
&= \pi(e_{\tau(j)}) \\
&= \iota(x_{\tau(j)}) \\
&= \iota(\tau \star x_j)
\end{aligned}$$

Finally, for $n = 2$ the map ι is still injective for $p \neq 2$. For $p = 2$ our map ι fails to be injective. Actually, any 1-dimensional representation of $\mathscr{S}_2$ is trivial in characteristic $p = 2$. So $\mathrm{LinDim}_{\mathbb{F}_{2^k}}(\mathscr{S}_2, \mathrm{X}_2) = 2$ since the standard representation and the inclusion $\iota(x_1) = e_1, \iota(x_2) = e_2$ satisfies

$$\rho_2(\tau)(\iota(x_j)) = \iota(\tau \star x_j)$$

for all τ in $\mathscr{S}_2$ and x_j in X_2. □

An application to n-bit permutations. It is well-known that any 2-bit permutation is given by an affine map. Namely, that the boolean functions components of any bijection $f : \mathbb{F}_2^2 \to \mathbb{F}_2^2$ are affine:

$$f(x,y) = (ax+by+c, a'x+b'y+c')$$

where $a,b,c,a',b',c' \in \mathbb{F}_2$.

Here we give a proof of this fact together with a generalization to permutations of n-bit.

Let $\mathrm{P}(\mathbb{F}_2^n)$ be the group of bijections of $\mathbb{F}_2^n$ and let $\mathrm{aff}(\mathbb{F}_2^n)$ be the subgroup of affine maps i.e. $g \in \mathrm{aff}(\mathbb{F}_2^n)$ if and only if $g(x) = ax+b$ where $b \in \mathbb{F}_2^n$, $a \in \mathrm{GL}(\mathbb{F}_2^n)$.

Proposition 4.4.2. *There is a group monomorphism $\alpha : \mathrm{P}(\mathbb{F}_2^n) \to \mathrm{aff}(\mathbb{F}_2^{2^n-2})$ and an injection $\iota : \mathbb{F}_2^n \to \mathbb{F}_2^{2^n-2}$ such that*

$$\rho(g)(\iota(x)) = \iota(g(x))$$

for all $g \in \mathrm{P}(\mathbb{F}_2^n)$, $x \in \mathbb{F}_2^n$.

Proof. This is a consequence of Proposition 4.4.1. To see why, notice that we can identify the symmetric group $\mathscr{S}_{2^n}$ with the group of permutations $\mathrm{P}(\mathbb{F}_2^n)$ of $\mathbb{F}_2^n$. That is to say,

$$\mathscr{S}_{2^n} \cong \mathrm{P}(\mathbb{F}_2^n)\,.$$

Such identification can be done by using the binary representation of the subindex j of the letter $x_j \in \mathrm{X}_{2^n}$. Namely,

$$x_j \longleftrightarrow (d_{n-1}, d_{n-2}, \ldots, d_1, d_0) \in \mathbb{F}_2^n$$

where $j = \sum_{i=0}^{n-1} d_j 2^j$.

Now by Proposition 4.4.1 there is a representation $\rho : \mathscr{S}_{2^n} \to \mathrm{GL}(\mathbb{F}_2^{2^n-1})$ and map $\iota : \mathrm{X}_{2^n} \to \mathbb{F}_2^{2^n-1}$ such that

$$\rho(g)(\iota(x)) = \iota(g(x))$$

for all $x \in \mathrm{X}_{2^n}$, $g \in \mathscr{S}_{2^n}$.

Now let $\mathrm{H} \subset \mathbb{F}_2^{2^n-1}$ be the affine hyperplane generated as follows

$$\mathrm{H} = \{c_0 \cdot \imath(x_0) + \cdots + c_{2^n-1} \cdot \imath(x_{2^n-2}) : \sum_{i=0}^{2^n-2} c_i = 1\}.$$

It is clear that $\imath(x_j)$ is in H for $j = 0, \cdots, 2^n - 2$. Notice that, for $j = 2^n - 1$, $\imath(x_{2^n-1}) = \imath(x_0) + \cdots + \imath(x_{2^n-2})$ and $\sum_{i=0}^{2^n-2} 1 = 1$; hence, also $\imath(x_{2^n-1})$ is in H. So $\imath(\mathrm{X}_{2^n}) \subset \mathrm{H}$. Now, since the linear maps of $\rho(\mathscr{S}_{2^n})$ permute $\imath(\mathrm{X}_{2^n})$, they preserve the affine hyperplane H and hence, they act on H as affine maps. Keeping in mind the above identification of $\mathscr{S}_{2^n} \cong \mathrm{P}(\mathbb{F}_2^n)$, we get a monomorphism $\alpha : \mathrm{P}(\mathbb{F}_2^n) \to \mathrm{aff}(\mathrm{H})$ such that

$$\alpha(g)(\imath(x)) = \imath(g(x))$$

for all g in $\mathrm{P}(\mathbb{F}_2^n)$ and x in $\mathbb{F}_2^n$. Finally, being H an affine hyperplane of $\mathbb{F}_2^{2^n-1}$, it has dimension $2^n - 2$, hence $\mathrm{H} \cong \mathbb{F}_2^{2^n-2}$ and we are done. □

This shows that 2-bit permutations are affine 2-bit maps. The 3-bit permutations can be regarded as 6-bit affine maps and so on.

4.4.2 The general linear group $\mathrm{GL}(\mathbb{F}_q^n)$

For g in $\mathrm{GL}(\mathbb{F}_q^n)$ and v in $\mathbb{F}_q^n$, let us define $\star$ as $g \star v = g(v)$. Set $\mathrm{Y}_n := \mathbb{F}_q^n$.

Proposition 4.4.3. *We have that* $\mathrm{LinDim}_{\mathbb{F}_{p^k}}(\mathrm{GL}(\mathbb{F}_q^n), \mathrm{Y}_n) \geq n$.

Proof. Since the action of the symmetric group $\mathscr{S}_n$ on X_n is equal to the action of $\rho_n(\mathscr{S}_n) \subset \mathrm{GL}(\mathbb{F}_q^n)$ on $\imath(\mathrm{X}_n) \subset \mathbb{F}_q^n$ we have

$$\mathrm{LinDim}_{\mathbb{F}_{p^k}}(\mathrm{GL}(\mathbb{F}_q^n), \mathrm{Y}_n) \geq n - 1.$$

Assume that there is a representation $\rho : \mathrm{GL}(\mathbb{F}_q^n) \to \mathrm{GL}(\mathbb{F}_{p^k}^{n-1})$ and an injective map $\imath : \mathbb{F}_q^n \to \mathbb{F}_{p^k}^{n-1}$ such that

$$\rho(g)(\imath(v)) = \imath(g \star v)$$

for all g in $\mathrm{GL}(\mathbb{F}_q^n)$ and v in Y_n. One of the vectors $\iota(e_j)$, for $j = 1,\ldots,n$, must be a linear combination of the others. Namely, there is a j such that

$$\iota(e_j) = \sum_{s\neq j, 1\leq s\leq n} c_s \iota(e_s),$$

where the coefficients c_s are in $\mathbb{F}_{p^k}$. From the action of the permutations, it follows that all coefficients c_s are equal. Then, swapping e_j with any of the other vectors implies $c_s = -1$. Hence, we get

$$\sum_{j=1}^{n} \iota(e_j) = 0.$$

Now let g be an element of $\mathrm{GL}(\mathbb{F}_q^n)$ such that $g(e_1) = \lambda e_1$, $\lambda \neq 1$, and $g(e_j) = e_j$ for $1 < j \leq n$. Then

$$\begin{aligned} 0 &= \rho(g)\left(\sum_{j=1}^{n} \iota(e_j)\right) \\ &= \sum_{j=1}^{n} \rho(g)\iota(e_j) \\ &= \sum_{j=1}^{n} \iota(g\star e_j) = \iota(\lambda e_1) + \sum_{j=2}^{n} \iota(e_j). \end{aligned}$$

So $\iota(\lambda e_1) = \iota(e_1)$ which contradicts the fact that ι is injective. □

4.4.3 The cyclic group $(\mathbb{Z}_n, +)$ acting on itself

In this subsection, we compute the linear dimension for the action of the additive group $\mathbb{Z}_n$ acting on itself. For instance, let $G = \mathbb{Z}_n$, $X = \mathbb{Z}_n$ and $\star = +$.

To state our main theorem we need the following definitions.
Let q be a prime power and n a positive integer such that $\gcd(q,n) = 1$, the order of q modulo n is denoted by $\mathrm{ord}_n(q)$. For $n = 1$ we set $\mathrm{ord}_1(q) = 0$.
Let $\mathrm{LD}(n,q)$ be defined as

$$\mathrm{LD}(n,q) = \min\left\{\left(\sum_{j=1}^{\ell} \mathrm{ord}_{n_j}(q)\right) : n = \prod_{j=1}^{\ell} n_j\,,\, \gcd(n_i,n_j) = 1\,,\, i\neq j\right\}$$

For example $\mathrm{LD}(15,2) = 4 = \mathrm{ord}_{15}(2)$ and $\mathrm{LD}(21,2) = 5 < \mathrm{ord}_{21}(2) = 6$. Notice that $\mathrm{LD}(1,q) = 0$ for every q.

Theorem 4.4.4. *Fix a prime power p^k and let $n = p^m r$, with $\gcd(p,r) = 1$. Then*

$$\mathrm{LinDim}_{\mathbb{F}_{p^k}}(\mathbb{Z}_n, \mathbb{Z}_n) = \begin{cases} \mathrm{LD}(r, p^k) & \text{if } m = 0, \\ \mathrm{LD}(r, p^k) + (p^{m-1} + 1) & \text{if } m > 0\,. \end{cases}$$

For the proof of the theorem, we need the following facts from linear algebra.

Let $w = \mathrm{LinDim}_{\mathbb{F}_{p^k}}(\mathbb{Z}_n, \mathbb{Z}_n)$ and let A be a matrix in $\mathrm{GL}(\mathbb{F}_{p^k}^w)$. Denote with n the order of A, i.e. the order of the cyclic subgroup of $\mathrm{GL}(\mathbb{F}_{p^k}^w)$ generated by A, and write $n = p^m r$ with $\gcd(p,r) = 1$.

Set $q = p^k$ and let $f(X) \in \mathbb{F}_q[X]$ be the minimal polynomial of A^{p^m} and let $f(X) = \prod_{i=1}^{l} f_i(X)$ be its factorization in irreducibles $f_i(X)$'s. Since $P(X) = X^r - 1$ has simple roots and $P(A^{p^m}) = 0$, we get that $f_i(X) \neq f_j(X)$ for $i \neq j$. Then A^{p^m} decomposes in s blocks $A_1, \ldots, A_s$ as follows

$$A^{p^m} = \begin{bmatrix} A_1 & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A_s \end{bmatrix}, \tag{4.2}$$

where the minimal polynomial of the block A_j is $f_j(X)$. Let r_i be the order of the block A_i. Then $r = \mathrm{LCM}(r_1, r_2, \ldots, r_s)$ i.e. r is the least common multiple of the r_i's.

The characteristic polynomial χ_i of each block A_i is the d_i-th power of f_i, i.e. $\chi_i(X) = f_i^{d_i}(X)$. Moreover, each block A_i is itself a matrix block of size d_i associated with the multiplication for α in the vector space $\mathbb{F}_q(\alpha)^{d_i}$. In particular, α has order r_i in the multiplicative group $\mathbb{F}_q(\alpha)^*$.

Now let $N = A^r - \mathrm{Id}$. Since

$$(N + \mathrm{Id})^{p^m} = N^{p^m} + \mathrm{Id} = (A^r)^{p^m} = \mathrm{Id},$$

we have that $N^{p^m} = 0$ and hence, N is nilpotent. Now observe that N commutes with A^{p^m}, so also N decompose in nilpotent blocks as

$$N = \begin{bmatrix} N_1 & 0 & 0 & 0 \\ 0 & N_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & N_s \end{bmatrix}.$$

The following lemma is a direct consequence of the above decompositions.

Lemma 4.4.5. *Let $w = \mathrm{LinDim}_{\mathbb{F}_q}(\mathbb{Z}_n, \mathbb{Z}_n)$ and let $n = p^m r$. Let $\rho : (\mathbb{Z}_n, +) \to \mathrm{GL}(\mathbb{F}_q^w)$ and $\iota : (\mathbb{Z}_n, +) \to \mathbb{F}_q^w$ such that*

$$\rho(g)(\iota(x)) = \iota(g \star x)$$

for all g, x in $\mathbb{Z}_n$. Then the matrix $A = \rho(1)$ has order n and w.r.t the above decomposition (4.2):

- $f_i \neq X - 1 \implies d_i = 1$,
- $f_i \neq X - 1 \implies N_i = 0$,
- *for $f_j = X - 1$, the block $A_j = \mathrm{Id}$.*

Proof. (of Theorem 4.4.4) By the above Lemma 4.4.5 we see that just one block of N_j is different from zero. Assume that it is N_1, and so $A_1 = \mathrm{Id}$. Then, the minimum size for N_1 to be nilpotent of order p^m but not of order p^{m-1} is $p^{m-1} + 1$.

For $i > 1$, let n_i be the order of each block A_i. To obtain the minimum size for A_i, we have to minimize over $\deg(f_i)$ where $f_i \in \mathbb{F}_q[X]$ is irreducible such that

$$n_i = \mathrm{ord}(\alpha) | q^{\deg(f_i)} - 1 \,.$$

where $\mathrm{ord}(\alpha)$ is the order of α in the multiplicative group $\mathbb{F}_q(\alpha)^*$. Thus

$$\deg(f_i) = \mathrm{ord}_{n_i}(q)\,,$$

since there is an irreducible $f_i \in \mathbb{F}_q[X]$ with $\deg(f_i) = \mathrm{ord}_{n_i}(q)$. By the Chinese Remainder Theorem, we can assume $\gcd(n_i, n_j) = 1$ and so

$$r = \mathrm{lcm}(n_2, \cdots, n_s) = \prod_{j=2}^{s} n_j .$$

We have shown the inequality

$$\mathrm{LinDeg}_{\mathbb{F}_q}(\mathbb{Z}_n, (\mathbb{Z}_n, +)) \geq \begin{cases} \mathrm{LD}(r,q) & \text{if } m = 0, \\ \mathrm{LD}(r,q) + (p^{m-1}+1) & \text{if } m > 0 . \end{cases}.$$

To show the equality, we need to construct the injective function

$$\iota : (\mathbb{Z}_n, +) \to \mathbb{F}_q^w$$

and the representation

$$\rho : (\mathbb{Z}_n, +) \to \mathrm{GL}(\mathbb{F}_q^w),$$

where

$$w = \begin{cases} \mathrm{LD}(r,q) & \text{if } m = 0, \\ \mathrm{LD}(r,q) + (p^{m-1}+1) & \text{if } m > 0 . \end{cases}.$$

We will assume $m > 0$ since for the case $m = 0$, it is enough to avoid the nilpotent block.

The previous proof shows us how to construct a matrix A in $\mathrm{GL}(\mathbb{F}_q^w)$ of order n by using blocks. Let A be in $\mathrm{GL}(\mathbb{F}_q^w)$ defined as

$$A = \begin{bmatrix} N + \mathrm{Id} & 0 & 0 & 0 \\ 0 & A_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A_s \end{bmatrix},$$

where Id is the $(p^{m-1}+1) \times (p^{m-1}+1)$ identity and N is the well-known $(p^{m-1}+1) \times (p^{m-1}+1)$ lower diagonal nilpotent matrix. Then

$$(N + \mathrm{Id})^{p^m} = \mathrm{Id},$$

but $(N+\mathrm{Id})^{p^{m-1}} \neq \mathrm{Id}$.

For each $j>1$ let $\mathbb{F}_q(\alpha_j)$ be the extension of degree $\mathrm{ord}_{n_j}(q)$ such that α_j has order n_j. The existence of such α_j is well-known, see e.g. [56, Theorem 2.46, page 65]. The extension $\mathbb{F}_q(\alpha_j)$ is a vector space over $\mathbb{F}_q$ isomorphic to $\mathbb{F}_q^{\mathrm{ord}_{n_j}(q)}$. So let A_j be the $\mathrm{ord}_{n_j}(q) \times \mathrm{ord}_{n_j}(q)$ matrix corresponding to the multiplication by α_j in $\mathbb{F}_q(\alpha_j)$. Moreover, let $v_j \in \mathbb{F}_q^{\mathrm{ord}_{n_j}(q)}$ be a vector corresponding to $1 \in \mathbb{F}_q(\alpha_j)$ w.r.t. the isomorphism $\mathbb{F}_q(\alpha_j) \cong_{\mathbb{F}_q} \mathbb{F}_q^{\mathrm{ord}_{n_j}}$. Finally, let $v_1 = [1,0,\cdots,0] \in \mathbb{F}_q^{(p^{m-1}+1)}$ and let $v = v_1 + v_2 + \cdots + v_s \in \mathbb{F}_q^w$.
Define $\rho : (\mathbb{Z}_n, +) \to \mathrm{GL}(\mathbb{F}_q^w)$ as

$$\rho(j) := A^j$$

and $\iota : \mathbb{Z}_n \to \mathbb{F}_q^w$ as

$$\iota(j) = A^j \cdot v\,.$$

We have that $\rho(g)(\iota(j)) = \iota(g \star j)$ holds for all g, j in $\mathbb{Z}_n$ and so, to complete the proof, we need to check that ι is injective.
Assume that for $0 \leq a < b \leq n-1$ we have $i(a) = i(b)$. Then $A^h \cdot v = v$ for $0 < h = b-a < n$. Then

$$\begin{cases} (N+\mathrm{Id})^h \cdot v_1 = v_1 \\ A_2^h \cdot v_2 = v_2 \\ \vdots \\ A_s^h \cdot v_s = v_s \end{cases},$$

then the equalities $A_j^h \cdot v_j = v_j$ for $j = 2, \cdots, s$ imply that $r|h$. Moreover, the first equality implies that $(N+\mathrm{Id})^h = \mathrm{Id}$ since the vectors $\{N^0 \cdot v_1, N^1 \cdot v_1, \ldots, N^{p^{m-1}} \cdot v_1\}$ form a basis of $\mathbb{F}_q^{(p^{m-1}+1)}$, and

$$(N+\mathrm{Id})^h \cdot N^j \cdot v_1 = N^j \cdot v_1$$

for all $j = 0, \ldots, p^{m-1}$. So $p^m \,|\, h$ and $n = p^m r \,|\, h$. This is a contradiction with $0 < h = b-a < n$. This completes the proof. □

Chapter 5

Non-interactive Commitment from Non-transitive Group Actions

5.1 Introduction

Using ideas from Tensor Isomorphism, this chapter tries to model the relations between the orbits of a non-transitive group action. In the case of tensors, two elements in the same orbit share the same rank (while it is not true the converse). Hence, we wonder if such a property can be used for cryptographic applications, i.e. the existence of an invariant function f that gives information on the orbits, and hence, when two set elements are linked by a group element. One of the basic cryptographic primitives is the bit commitment: we build such a functionality using the fact that the bit b is hidden into an element x having $f(x) = v_b$. However, the computation of f is intractable in general (e.g. the tensor rank), and this ensures the security of the commitment scheme. This chapter is based on a joint work with Andrea Flamini and Andrea Gangemi [30], presented at Asiacrypt 2023.

5.1.1 Commitment schemes

A commitment scheme is a cryptographic protocol between two parties, a sender and a receiver. The sender wants to commit to a value b without revealing it to the other party. To do this, he binds b to a commitment C that is sent to the receiver. In a second moment, the sender wants to reveal b and the receiver must be able to

verify that it was the committed value behind C. A commitment must satisfy two security properties: it must not reveal any information about the committed value (hiding property), and the sender cannot reveal a different $b' \neq b$ that opens to the same commitment (binding property).
Commitment schemes are widely used, both as stand-alone protocols and as atomic parts of more involved mechanisms. For example, they are used in Zero-Knowledge proofs [57], digital auctions [67], signature schemes [51], multi-party computation [40], e-voting [31] and confidential transactions [72]. In this work, we will mainly focus on *bit* commitments, where the committed value b can be 1 or 0.

5.1.2 Commitment schemes from group actions

Previous commitments were known from cryptographic group actions. Brassard and Young [18] propose two kinds of bit commitments from what they call *certified* and *uncertified* group actions. A certified group action is an action from the group G over the set X such that checking that two elements are in the same orbit is an easy task. On the contrary, the same verification could not be polynomial-time for an uncertified group action. Since the problem of deciding whether two elements of X are in the same orbit is assumed to be hard in this work, we will focus on the latter case. Given a group action from G on X, the computationally binding and perfectly hiding bit commitment presented in [18] is as follows.

- The receiver randomly generates x_0 from X and g from G. Then sets x_1 as $g \star x_0$. He sends to the sender the pair (x_0, x_1) and a proof π that they are in the same orbit.
- The sender wants to commit to the bit b. First, he checks that the proof π is valid, then he picks h from G and sends $\texttt{com} = h \star x_b$ to the receiver, keeping secret h.
- To open the committed bit b, the sender reveals b and h to the receiver, which checks that com is equal to $h \star x_b$.

The first thing to notice is that this is an *interactive* bit commitment since the sender needs the receiver's cooperation for the creation of the commitment. Secondly, the communication cost is at least as big as the proof of the statement that x_0 and x_1 are in

the same orbit. This is an NP-statement (the witness is given by g) and hence admits an interactive proof (even a non-interactive one, using the Fiat-Shamir heuristic and the Random Oracle Model), but it can be very large in communication.

In [50], Ji, Qiao, Song and Yun propose two bit commitment protocols. The first is a slight generalization of the protocol from [18], using non-abelian group actions. The obtained protocol has the same drawbacks noticed above: it is interactive and has a large communication cost. The second proposal concerns the use of the following pseudorandom function

$$f : X \times G \to X \times X, \quad (x, g) \mapsto (x, g \star x)$$

and, after applying the Blum-Micali amplification [15], the authors build an interactive bit commitment scheme using the construction from [63]. In this construction, it is needed that $|X| \geq |G|$, and the obtained bit commitment is statistically binding and computationally hiding.

5.1.3 Original contribution

We present a bit commitment scheme that is non-interactive, perfectly binding and computationally hiding in the standard model. This scheme is based on a group action framework that makes use of certain invariant functions. One of the innovative aspects of our proposal is that it concerns *non-transitive* group actions, while known cryptographic applications use transitive actions or they restrict to one orbit. The non-transitivity of the action used in this paper is crucial and necessary; in fact, we need to be able to exhibit two elements that are in two different orbits. Such elements are generated with the aid of the new group action framework, in which we endow the group action with a function that is constant inside the orbits. Given the group G acting on the set X via the action $\star$, an *invariant function* $f : X \to T$, with T be a set, has the following property

$$f(g \star x) = f(x), \quad \forall x \in X, g \in G.$$

The key point is that evaluating this function on a randomly chosen element is hard, while, for a particular subset of elements that we call *canonical elements*, it is easy to compute. Also, the fact that the function is constant inside the orbits

guarantees that, if we consider two elements with distinct images, they must live in (and generate under the action of G) distinct orbits. This observation is crucial to prove our commitment scheme is perfectly binding. We call *Group Action with Canonical Elements* (GACE) a group action with the above properties. Moreover, the existence of decision problems about whether an element is randomly picked from a specific orbit or not enables us to prove that our commitment scheme is computationally hiding.
The structure of our construction enables an additional property that is shared with the Pedersen commitment. An honest sender generating two commitments of the same value b can prove to the receiver that they are in fact linked to the same message, without revealing it. We call this scheme a *linkable commitment* and we formally define the security properties that enable the adoption of such a primitive in cryptography. However, using some techniques from ring signature schemes [11], we show how to extend this property to the case of a possibly malicious sender in the Random Oracle Model.

Finally, we propose an example of GACE based on tensors: the action is the usual one from $\mathrm{GL}(n)$, while the invariant function f is the rank, which is invariant under the proposed action. However, there are no known constructions of high-order tensors, hence, we instantiate the bit commitment with ranks n and $n-1$.

Update. After the publication of [30] to Asiacrypt 2023, the proposed instantiation based on tensors of this framework has been attacked in [69]. The authors show that the choice of low-rank tensors is not safe since there is an efficient way to distinguish tensors of rank n and $n-1$. This means breaking the hiding property of the commitment. Moreover, they repair the scheme obtaining a slightly lower security level, even if this has no impact on many practical constructions. However, finding a practical GACE remains an open problem.

5.2 Our Framework

The goal of this section is to design a non-interactive commitment scheme using assumptions from cryptographic group actions. We will focus on non-abelian and non-transitive actions. To develop such a commitment scheme, we first analyze the

issues arising from an initial construction, then we define a framework that we use to circumvent these problems.

5.2.1 A first attempt

Based on the non-transitivity of the group action $(G,X,\star)$, we can make a first attempt at building a *non-interactive* bit commitment scheme. We give its description using a trusted third party (TTP), and then we analyze how to remove it.
Given the action $(G,X,\star)$, the TTP chooses and publishes two elements x_0 and x_1 of X lying in different orbits. The sender, to commit a bit b, generates a random g in G and sets as the commitment of b the value $\texttt{com} = g \star x_b$. The opening material is g. In other words, the sender picks a random element in the orbit of x_b. In the opening phase, given b, com and g, the receiver accepts if com is equal to $g \star x_b$ and rejects otherwise. Informally, the hiding property is given by the fact that checking whether com is in the orbit of x_0 or x_1 is hard, while the binding property follows from the impossibility of going from an orbit to another via the action of G.
In the following, we try to remove the TTP and analyze some possible scenarios.

1. **The sender generates and publishes x_0 and x_1.** In this case, we can see that a malicious sender can generate x_0 and x_1 in the same orbit via $x_1 = h \star x_0$. He commits to $g \star x_0$ and, during the opening phase, he could open to both 0 and 1 using g or gh^{-1}. In this case, the binding property does not hold.

2. **The sender generates and publishes x_0, x_1 together with a proof π that they are in different orbits.** Given a proof π that x_0 and x_1 are not in the same orbit, we obtain that the protocol is hiding and binding, under the assumption that deciding whenever two elements share the orbit is hard. In this scenario, the hard task is the generation of the proof π. In fact, the language

$$L = \{(y_0, y_1) \in X \times X \mid y_0 \text{ and } y_1 \text{ are in different orbits}\}$$

   is in coNP. Unless we have a computationally unbounded prover [42] (and this is not the case), it means that known techniques fail to generate a short non-interactive proof for L which would enable the design of a non-interactive commitment scheme. Since interactive bit commitments based on group actions are known [18, 50], we do not further study this case.

3. **The receiver generates and publishes x_0 and x_1.** We are again in the case of interactive bit commitments, and we remand to the known schemes based on group actions.

With such techniques, we have seen that there are some tricky aspects that are hard to deal with. For example, we need to build a proof for a language in coNP, and the absence of a witness (as we are used to, when we work in NP) is the first obstacle. To overcome such difficulties, we introduce a general framework on group actions that eases the design of the non-interactive bit commitment sketched above. The trick is the definition of an invariant function that is constant inside the orbits and hard to compute for a randomly chosen element. However, we assume that there is a set of representative elements for which the computation of such a function is easy. This avoids the need for a proof for the above language L. These concepts will be formalized in the next subsection.

5.2.2 Group Actions with Canonical Elements (GACE)

In this section, we introduce the concepts of invariant functions and canonical elements, and we present the cryptographic assumptions linked to them.

Definition 5.2.1. *Given a group action $(G,X,\star)$ and a function $f : X \to T$, we say that f is* invariant under the action of G *if $f(g \star x) = f(x)$ for every g in G and every x in X. We say that f is* fully invariant *if $f(x) = f(y)$ if and only if there exists g in G such that $y = g \star x$.*

In the following, we can assume that f is surjective, restricting the set T to the image $f(X)$. To exploit the properties of invariant functions while keeping the dGA-IP hard, we want the function f to be hard to compute on a large class of elements of X. At the same time, we want to define particular elements of X on which the computation of f is feasible.

Definition 5.2.2. *Let $f : X \to T$ be a surjective invariant function for the action $(G,X,\star)$ and let $T' \subset T$. Suppose that there exists a polynomial-computable map*

$$\langle\cdot\rangle : T' \to X, \;\; t \mapsto \langle t \rangle$$

such that the function $f \circ \langle\cdot\rangle$ is the identity on the subset T' of T. We call $\langle\cdot\rangle$ the canonical representation of T' in X *and $\langle t \rangle$ the* canonical t-element *(with respect to f*

and $\langle\cdot\rangle$). If $T' = T$, we say that $\langle\cdot\rangle$ is complete*. Moreover, we say that $(G, X, \star, f, \langle\cdot\rangle)$ is a* Group Action with Canonical Element *(GACE) if the following hold:*

1. *if $\mathcal{O}(z)$ is the orbit of z in X, then for any* PPT *adversary $\mathcal{A}$ there is a negligible function μ such that*

$$\Pr[\mathcal{A}(x) = f(x)] \leq \frac{1}{|T'|} + \mu(|x|),$$

   where x is sampled uniformly random from $\bigsqcup_{t\in T'} \mathcal{O}(\langle t\rangle)$;

2. *there is a* PPT *algorithm that for any t in T' computes $f(\langle t\rangle)$.*

In other words, the definition above says that, for every t in T', we have $f(\langle t\rangle) = t$ and the function f is hard to compute in general, but is instead easy to calculate on canonical elements. Moreover, the construction of such $\langle t\rangle$ is a polynomial-time task.

In the following constructions, whenever a random element of X is needed, we pick a random canonical element $\langle t\rangle$, a random g from G and compute $g \star \langle t\rangle$. In this way, instead of using the whole X, we always work with the union of the orbits of the canonical elements. In other words, the set on which the group G acts becomes

$$X' = \bigsqcup_{t\in T'} \mathcal{O}(\langle t\rangle).$$

This implies that the GACE $(G, X', \star, f, \langle\cdot\rangle)$ has a fully invariant function f and the canonical representation $\langle\cdot\rangle$ is complete. Given a fully invariant function f, the problem of determining whether two elements have the same image under f is equivalent to deciding whether they lie in the same orbit (dGA-IP).

5.3 The Commitment Scheme

5.3.1 Bit commitment scheme from a GACE

The first application of our framework is a bit commitment scheme. Given a Group Action with Canonical Elements, we design the commitment scheme described in Figure 5.1, following the attempts shown in Subsection 5.2.1. The bit commitment

$\mathsf{PGen}\left(1^{\lambda}\right)$

1 : choose $(G, X, \star, f, \langle\cdot\rangle)$
2 : $t_0 \leftarrow_{\$} T'$
3 : $t_1 \leftarrow_{\$} T' \setminus \{t_0\}$
4 : **return** $(G, X, \star, f, \langle\cdot\rangle, t_0, t_1)$

$\mathsf{Commit}(b)$

1 : $g \leftarrow_{\$} G$
2 : $c \leftarrow g \star \langle t_b \rangle$
3 : **return** (c, g)

$\mathsf{Open}(b, c, g)$

1 : **if** $g^{-1} \star c = \langle t_b \rangle$
2 : **return accept**
3 : **else return reject**

Fig. 5.1 Bit commitment scheme from a GACE.

is proven secure under both the dGA-IP assumption that we have introduced in this paper and the 2GA-PR assumption.

Theorem 5.3.1. *The bit commitment scheme in Figure 5.1 is perfectly binding.*

Proof. Without loss of generality, we can assume $m_0 = 0$ and $m_1 = 1$. Suppose there exists an adversary $\mathscr{A}$ that on input $\mathsf{pp} = (G, X, \star, f, \langle\cdot\rangle, t_0, t_1)$ returns the tuple $\mathtt{com}, r_0, r_1$ such that

$$\mathsf{Open}(0, \mathtt{com}, r_0) = \mathsf{Open}(1, \mathtt{com}, r_1) = \textbf{accept}$$

with positive probability. This means that $r_0 \star \langle t_0 \rangle = \mathtt{com} = r_1 \star \langle t_1 \rangle$, and then $r_1^{-1} r_0 \star \langle t_0 \rangle = \langle t_1 \rangle$. Therefore, $\langle t_0 \rangle$ and $\langle t_1 \rangle$ are in the same orbit, but this is a contradiction and such an adversary $\mathscr{A}$ cannot exist.

□

Theorem 5.3.2. *The bit commitment scheme in Figure 5.1 is computationally hiding under the decisional Group Action Inversion Problem assumption.*

Proof. The dGA-IP assumption states that every adversary of the dGA-IP game has at most negligible advantage. We prove that the existence of an adversary of the game $\mathrm{Hiding}(\Pi_{\mathsf{Com}})$ with advantage at least $\varepsilon(\lambda)$, where $\varepsilon(\lambda)$ is a non-negligible function, implies the existence of an adversary $\mathscr{A}$ of the dGA-IP game with advantage $2\varepsilon^2(\lambda)$,

which is non-negligible.
The proof is divided into 3 parts: firstly, we describe our adversary $\mathscr{A}$ of the dGA-IP game. It will exploit two instances of an adversary of the Hiding(Π_{Com}) game, therefore we must show that it correctly simulates the challenger of such a game. Finally, we quantify a lower bound to the advantage of the adversary $\mathscr{A}$.

1. *Reduction description*.
   The adversary $\mathscr{A}$ of the dGA-IP game (see Figure 5.2) receives from the challenger two set elements s and t, generated according to the dGA-IP game. $\mathscr{A}$ creates two instances of the adversary of Hiding(Π_{Com}) game having non-negligible advantage, namely $\mathscr{A}_1$ and $\mathscr{A}_2$. Then, the adversary $\mathscr{A}$ provides $\mathscr{A}_1$ with s and $\mathscr{A}_2$ with t separately. The two hiding commitment adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$ return respectively the bits b_0 and b_1 as outputs of their internal routine. Finally, the dGA-IP adversary $\mathscr{A}$ returns to the challenger the bit b' which is set to 1 if $b_0 = b_1$, otherwise it is set to 0.

2. $\mathscr{A}$ *correctly simulates the* Hiding(Π_{Com}) *challenger*.

   We show that $\mathscr{A}$ correctly simulates the challenger of the Hiding(Π_{Com}) game so that it is possible to quantify the probability of success of the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$. The elements s and t which $\mathscr{A}$ uses as input to $\mathscr{A}_1$ and $\mathscr{A}_2$ are generated as follows:

   - s is a random element in the orbit generated by $\langle t_c \rangle$, with c chosen uniformly at random in $\{0,1\}$;
   - when $b = 1$, t is chosen uniformly at random in the same orbit of s (note that $g' \star s = g'g \star \langle t_c \rangle$ is random as long as $g' \leftarrow_{\$} G$), otherwise, if $b = 0$, t is chosen at random in the orbit of $\langle t_{1-c} \rangle$.

   In particular, the orbit of s is chosen uniformly at random via the selection of c; then, given c, the orbit of t is chosen uniformly at random via b. This guarantees that $\mathscr{A}$ correctly simulates the challenger of the Hiding(Π_{Com}) game, who must choose, in the first step, whether to create a commitment to 0 or 1. Therefore, the adversaries $\mathscr{A}_1, \mathscr{A}_2$ win their games with probability greater than $\frac{1}{2} + \varepsilon(\lambda)$.

3. *Measurement of* $\mathscr{A}$*'s advantage*.

Finally, we compute a lower bound to the probability of success of $\mathscr{A}$ that we have described in the dGA-IP game.

We observe that the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$ do not interact, so the events that they win their games can be considered independent as long as their inputs are also independent.

It is possible to show that the selection of the inputs is independent since the selection process of s and t is performed by picking at random the orbit $\mathscr{O}(s)$ of s by sampling the bit c, and the orbit $\mathscr{O}(t)$ of t by sampling the bit b (actually the bit that determines the orbit of t is interpreted according to the value of s, but this is not relevant as long as the bit b is chosen at random).
Then, the canonical elements of the sampled orbits are randomized by sampling two random group elements $g, g' \in G$ and computing the action of such elements (or of the element $g'g$ instead of g', if $b = 1$, which is a random element as long as g' is random) on the canonical elements.

Given that the inputs to $\mathscr{A}_1$ and $\mathscr{A}_2$ are independent and that the two adversaries perform their operations regardless of the existence of each other, the events that $\mathscr{A}_1$ wins its game and $\mathscr{A}_2$ wins its game are independent.

For the sake of brevity, we refer to the event that $\mathscr{A}_1$ wins or loses its game as $(\mathscr{A}_1 \texttt{ wins})$ or $(\mathscr{A}_1 \texttt{ loses})$ and we do the same for $\mathscr{A}_2$ and $\mathscr{A}$: the game they are playing will be clear from the context.

Finally, we compute the lower bound of the probability of advantage of $\mathscr{A}$. To do that, we observe that $\mathscr{A}$ wins the game when $b' = b$ and this happens either when both $\mathscr{A}_1$ and $\mathscr{A}_2$ win, or when they both lose.
In fact, when $b = 0$ then $\mathscr{O}(t) \neq \mathscr{O}(s)$; therefore, $b_0 \neq b_1$ happens if and only if both $\mathscr{A}_1$ and $\mathscr{A}_2$ win or when they both lose. The same holds when $b = 1$. Therefore,

$$
\begin{aligned}
&\Pr[\mathscr{A} \texttt{ wins}] = \\
&\quad \Pr[(\mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ wins}) \vee (\mathscr{A}_1 \texttt{ loses} \wedge \mathscr{A}_2 \texttt{ loses})] = \\
&\quad \Pr[(\mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ wins})] + \Pr[(\mathscr{A}_1 \texttt{ loses} \wedge \mathscr{A}_2 \texttt{ loses})] = \\
&\quad \Pr[(\mathscr{A}_1 \texttt{ wins})] \Pr[(\mathscr{A}_2 \texttt{ wins})] + \Pr[(\mathscr{A}_1 \texttt{ loses})] \Pr[\mathscr{A}_2 \texttt{ loses})] \geq \\
&\quad \left(\frac{1}{2} + \varepsilon(\lambda)\right)^2 + \left(\frac{1}{2} - \varepsilon(\lambda)\right)^2 = \frac{1}{2} + 2\varepsilon(\lambda)^2.
\end{aligned}
$$

Since $\varepsilon(\lambda)$ is a non-negligible function, we have defined an adversary $\mathscr{A}$ of the dGA-IP game that has a non-negligible advantage. This contradicts the dGA-IP assumption, therefore the adversary of Hiding(Π_{Com}) with non-negligible advantage does not exist and the commitment scheme Π_{Com} satisfies the hiding property. □

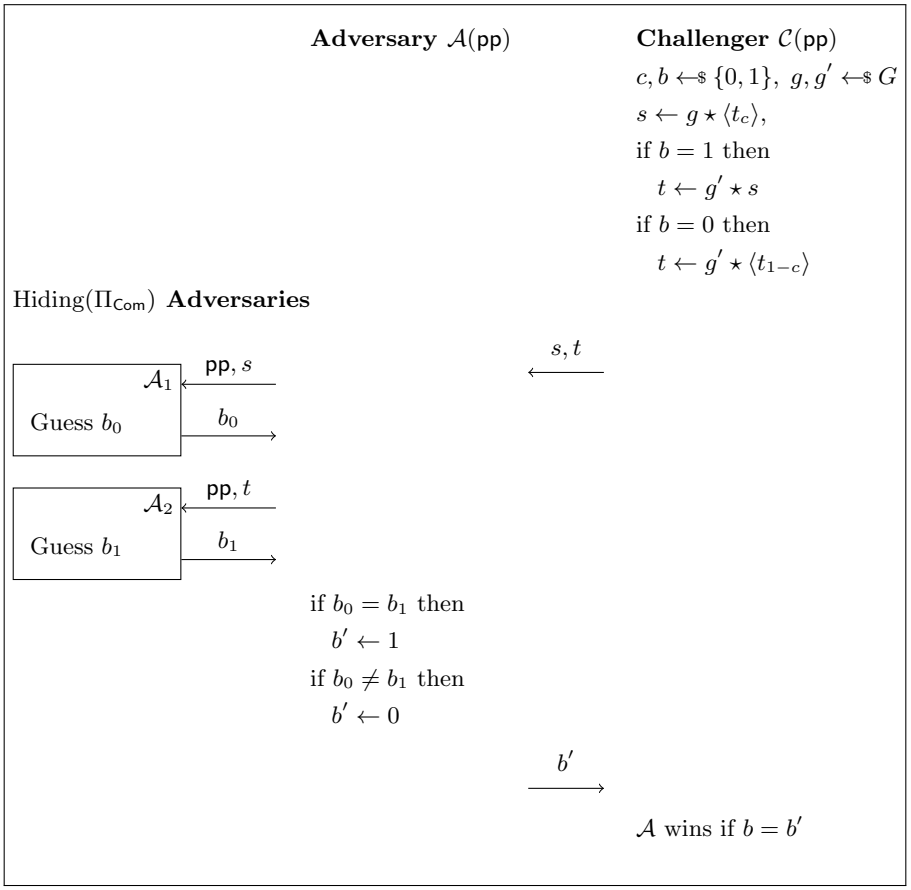

Fig. 5.2 Reduction from dGA-IP(pp) to the hiding game for the bit commitment scheme.

The two previous results can be summarized in the following corollary.

Corollary 5.3.3. *The bit commitment scheme in Figure 5.1 is secure under the decisional Group Action Inversion Problem assumption.*

We also have expanded the security analysis of the hiding property of the commitment scheme under the 2GA-PR assumption requiring that the two orbits $\mathscr{O}_0$ and $\mathscr{O}_1$ used to instantiate the bit commitment have *similar size*, i.e.

$$|\Pr[x \in \mathscr{O}_0] - \Pr[x \in \mathscr{O}_1]| = \nu(\lambda)$$

for a randomly chosen x in $\mathscr{O}_0 \cup \mathscr{O}_1$ and a negligible function $\nu(\lambda)$.

We obtain the following theorem.

Theorem 5.3.4. *If the bit commitment scheme in Figure 5.1 is instantiated using two orbits of similar size, it is secure under the 2GA-PR assumption.*

Proof. The commitment scheme satisfies the property of perfect binding, as shown in Theorem 5.3.1.
Now, we show the computationally hiding property. For simplicity, we assume that the cardinality of the two orbits is the same, that is, the probability of picking an element at random inside any orbit is $\frac{1}{2}$. The proof can be easily generalized to the case where the probability of falling into one orbit is negligibly greater than the probability of falling into the other. In other words, the proof holds whenever there exists a negligible function $\nu(\lambda)$ such that, given the two orbits $\mathscr{O}_0$ and $\mathscr{O}_1$,

$$|\Pr[x \in \mathscr{O}_0] - \Pr[x \in \mathscr{O}_1]| = \nu(\lambda)$$

for a randomly chosen x in $\mathscr{O}_0 \cup \mathscr{O}_1$. This assumption seems admissible and not too strict for cryptographic purposes.

We show that, given an adversary of the Hiding(Π_{Com}) game with non-negligible advantage, we can build an adversary of the 2GA-PR game with non-negligible advantage (recall that the advantage of $\mathscr{A}$ is defined as $\mathbf{Adv}(\mathscr{A}, \text{2GA-PR}(\mathtt{pp})) = \Pr[\mathscr{A} \text{ wins 2GA-PR}(\mathtt{pp})] - \frac{1}{2}$).

1. *Reduction description.*
   To define $\mathscr{A}$, we use two independent instances of the same adversary $\mathscr{A}_1, \mathscr{A}_2$ of the hiding game as we did in the proof of Theorem 5.3.2; then, we perform the same reduction, as it is presented in Figure 5.3.

2. *$\mathscr{A}$ correctly simulates the* Hiding(Π_{Com}) *challenger.*
   The adversary $\mathscr{A}$ correctly simulates the challenger of Hiding(Π_{Com}) with respect to the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$ separately, in fact, both s and t are uniformly sampled from the set of commitment to 0 and 1. Therefore, $\mathscr{A}_1$ and $\mathscr{A}_2$ will output the right bit with advantage $\varepsilon(\lambda)$.

3. *Measurement of $\mathscr{A}$'s advantage.*
   From now on, when we consider the orbits $\mathscr{O}(s)$ and $\mathscr{O}(t)$ of s and t respec-

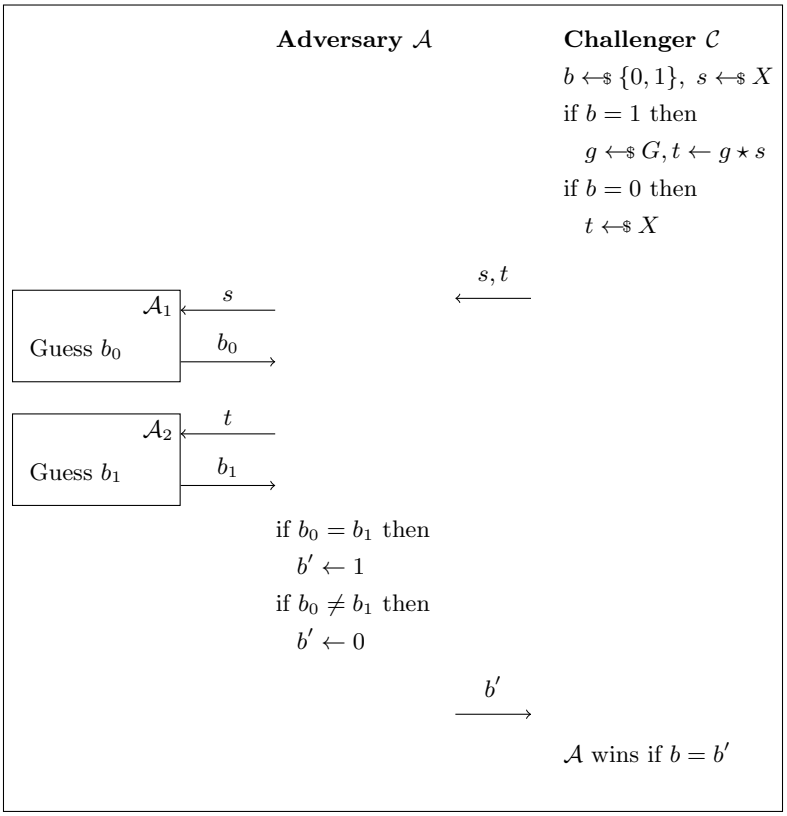

Fig. 5.3 Reduction from 2GA-PR to the hiding game for the bit commitment scheme.

tively, with abuse of notation, they will assume binary values according to the relation used in the bit commitment scheme Π_{Com}: $\mathscr{O}(s) = 1$ if s lies in the orbit of commitments to 1, and $\mathscr{O}(s) = 0$ if s lives in the orbit of commitments to 0. The same holds for $\mathscr{O}(t)$.

Before computing the lower bound of the advantage of the adversary $\mathscr{A}$, we state the following remark.

Remark 5.3.5. *The outcomes of the games performed by $\mathscr{A}_1$ and $\mathscr{A}_2$ in the reduction of Figure 5.3 are not independent since the values given as inputs to them are dependent (note that t is in the same orbit of s with probability $\frac{3}{4}$). However, it is still true that the outcomes of the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$ are independent if conditioned to fixed input values.*

For the sake of generality, we need to consider the case in which the advantage of the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$ in playing Hiding(Π_{Com}) game is not uniformly

distributed on the possible outputs. That is, it is possible that

$$\Pr[\mathscr{A}_1 \ \texttt{wins} \mid \mathscr{O}(s) = 1] = \frac{1}{2} + \varepsilon(\lambda) + \Delta,$$

$$\Pr[\mathscr{A}_1 \ \texttt{wins} \mid \mathscr{O}(s) = 0] = \frac{1}{2} + \varepsilon(\lambda) - \Delta,$$

with Δ possibly a negative value. The same holds for $\Pr[\mathscr{A}_2 \ \texttt{wins} \mid \mathscr{O}(t) = b]$, with $b \in \{0,1\}$.

Now, we can start with the computation of the lower bound of the advantage of $\mathscr{A}$ in winning the 2GA-PR game.

The probability that $\mathscr{A}$ wins the 2GA-PR game can be computed as follows, partitioning the event into three disjoint events:

$$\begin{aligned}\Pr[\mathscr{A} \ \texttt{wins}] = \Pr\left[b' = b\right] = \\ \Pr\left[\underbrace{(b = 0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t)) \wedge b' = b}_{\text{Event A}}\right] + \\ \Pr\left[\underbrace{(b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t)) \wedge b' = b}_{\text{Event B}}\right] + \\ \Pr\left[\underbrace{b = 1 \wedge b' = b}_{\text{Event C}}\right].\end{aligned}$$

We now separately quantify the three probabilities as follows. We recall that according to the event we are considering, the event $b = b'$ can be translated in terms of the success of the adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$

- **Event A**: when $b = 0$ and $\mathscr{O}(s) \neq \mathscr{O}(t)$, then $b = b'$ when both $\mathscr{A}_1$ and $\mathscr{A}_2$ win or when both of them lose. Therefore, it holds that

$$\begin{aligned}\Pr\left[b = 0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge b' = b\right] = \\ \Pr[b = 0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge \mathscr{A}_1 \ \texttt{wins} \wedge \mathscr{A}_2 \ \texttt{wins}] + \\ \Pr[b = 0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge \mathscr{A}_1 \ \texttt{loses} \wedge \mathscr{A}_2 \ \texttt{loses}].\end{aligned} \tag{5.1}$$

We can compute this probability by considering the general case $\Pr[b=0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}]$ and then substituting `outcome` with `wins` or `loses` accordingly with the formula above.

It holds that

$$\Pr[b=0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}] =$$
$$\sum_{c=0}^{1} \Pr[b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}] =$$
$$\sum_{c=0}^{1} \Big(\Pr[\mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome} \mid b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c] \cdot$$
$$\cdot \Pr[b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c] \Big).$$

Since the outcomes of $\mathscr{A}_1$ and $\mathscr{A}_2$ are independent once their input values are fixed, we have that

$$\Pr[\mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome} \mid b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c] =$$
$$\prod_{i=1}^{2} \Pr[\mathscr{A}_i \ \texttt{outcome} \mid b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c],$$

with $c \in \{0,1\}$.

Since the outcome of $\mathscr{A}_1$ only depends on the value of $\mathscr{O}(s)$ and the outcome of $\mathscr{A}_2$ depends only on $\mathscr{O}(t)$, then

$$\Pr[\mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome} \mid b=0 \wedge \mathscr{O}(s)=c \wedge \mathscr{O}(t)=1-c] =$$
$$\Pr[\mathscr{A}_1 \ \texttt{outcome} \mid \mathscr{O}(s)=c] \Pr[\mathscr{A}_2 \ \texttt{outcome} \mid \mathscr{O}(t)=1-c]$$

Therefore, since $\Pr\left[b=0 \wedge \mathscr{O}(s)=\bar{b} \wedge \mathscr{O}(t)=1-\bar{b}\right]=\frac{1}{8}$ with $\bar{b} \in \{0,1\}$ then

$$\Pr[b=0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}] =$$
$$\frac{1}{8}\Big(\Pr[\mathscr{A}_1 \ \texttt{outcome} \mid \mathscr{O}(s)=1] \cdot \Pr[\mathscr{A}_2 \ \texttt{outcome} \mid \mathscr{O}(t)=0] +$$
$$\Pr[\mathscr{A}_1 \ \texttt{outcome} \mid \mathscr{O}(s)=0] \cdot \Pr[\mathscr{A}_2 \ \texttt{outcome} \mid \mathscr{O}(t)=1] \Big).$$

We can finally compute the initial probability given in Eq. (5.1), by substituting `outcome` with `wins` and `loses` and obtaining

$$\Pr\left[b = 0 \wedge \mathscr{O}(s) \neq \mathscr{O}(t) \wedge b' = b\right] = \frac{1}{8} + \frac{1}{2}\varepsilon^2(\lambda) - \frac{1}{2}\Delta^2. \tag{5.2}$$

- **Event B**: when $b = 0$ and $\mathscr{O}(s) = \mathscr{O}(t)$, then $b = b'$ when either $\mathscr{A}_1$ wins and $\mathscr{A}_2$ loses or when $\mathscr{A}_1$ loses and $\mathscr{A}_2$ wins. Therefore, it holds that

$$\begin{aligned}\Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge b' = b\right] = \\ \Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge \mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ loses}\right] + \\ \Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge \mathscr{A}_1 \texttt{ loses} \wedge \mathscr{A}_2 \texttt{ wins}\right].\end{aligned} \tag{5.3}$$

Since in this case the input of $\mathscr{A}_1$ and $\mathscr{A}_2$ are in the same orbit, then we can state

$$\begin{aligned}&\Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge b' = b\right] = \\ &2\Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge \mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ loses}\right] = \\ &2\sum_{c=0}^{1} \Pr\left[b = 0 \wedge \mathscr{O}(s) = c \wedge \mathscr{O}(t) = c \wedge \mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ loses}\right].\end{aligned}$$

Using arguments similar to the ones used for **Event A**, that is the conditional independence of the outcomes of the adversaries once the inputs are fixed, the fact that the output of $\mathscr{A}_1$ (resp. $\mathscr{A}_2$) depends only on $\mathscr{O}(s)$ (resp. on $O(t)$) and finally that $\Pr\left[b = 0 \wedge O(s) = c \wedge \mathscr{O}(t) = c\right] = \frac{1}{8}$, for $c \in \{0,1\}$, we can write the Eq. (5.3) as follows

$$\Pr\left[b = 0 \wedge \mathscr{O}(s) = \mathscr{O}(t) \wedge b' = b\right] = \frac{1}{8} - \frac{1}{2}\varepsilon^2(\lambda) - \frac{1}{2}\Delta^2. \tag{5.4}$$

- **Event C**: when $b = 1$, $\mathscr{O}(s) = \mathscr{O}(t)$, then $b = b'$ when both $\mathscr{A}_1$ and $\mathscr{A}_2$ win or when both of them lose. Therefore, it holds that

$$\begin{aligned}\Pr\left[b = 1 \wedge b' = b\right] = \\ \Pr\left[b = 1 \wedge \mathscr{A}_1 \texttt{ wins} \wedge \mathscr{A}_2 \texttt{ wins}\right] + \\ \Pr\left[b = 1 \wedge \mathscr{A}_1 \texttt{ loses} \wedge \mathscr{A}_2 \texttt{ loses}\right].\end{aligned} \tag{5.5}$$

As in the computation of the probability of **Event A**, we must compute $\Pr[b = 1 \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}]$. Using similar arguments as before, and noticing that $\Pr[b = 1 \wedge \mathscr{O}(s) = c \wedge \mathscr{O}(t) = c] = \frac{1}{4}$ with $c \in \{0,1\}$, it can be shown that

$$\Pr[b = 1 \wedge \mathscr{A}_1 \ \texttt{outcome} \wedge \mathscr{A}_2 \ \texttt{outcome}] = \frac{1}{4}\sum_{c=0}^{1} \Pr[\mathscr{A}_1 \ \texttt{outcome} \mid \mathscr{O}(s) = c] \Pr[\mathscr{A}_2 \ \texttt{outcome} \mid \mathscr{O}(t) = c]$$

Therefore, substituting `outcome` with `loses` and `wins`, and using the probabilities of success of adversaries $\mathscr{A}_1$ and $\mathscr{A}_2$, from Eq. (5.5) we obtain

$$\Pr\left[b = 1 \wedge b' = b\right] = \frac{1}{4} + \varepsilon^2(\lambda) + \Delta^2. \tag{5.6}$$

Combining the partial results derived analysing **Event A**, **Event B** and **Event C** from Equations (5.2),(5.4) and (5.6) respectively, we obtain the final result

$$\Pr[\mathscr{A} \ \texttt{wins}] = \frac{1}{2} + \varepsilon^2(\lambda),$$

which proves that we have built an adversary for the 2GA-PR game which wins with non-negligible advantage. Therefore, an adversary who wins the hiding game with non-negligible advantage does not exist due to the 2GA-PR assumption. This means that the binary commitment scheme we have described results to be perfectly binding and computationally hiding.

□

Moreover, we can show that $\text{Hiding}(\Pi_{\mathsf{Com}})$ reduces to dGA-IP. This allows us to describe the relation between the dGA-IP and 2GA-PR assumptions.

Corollary 5.3.6. *The 2GA-PR problem reduces to dGA-IP when it is instantiated with two orbits of similar size.*

Proof. From previous results, we only need to show that the $\text{Hiding}(\Pi_{\mathsf{Com}})$ game reduces to the dGA-IP game.
We show how the existence of an adversary of the dGA-IP problem with non-negligible advantage allows the creation of an adversary of the $\text{Hiding}(\Pi_{\mathsf{Com}})$ game with non-negligible advantage.

1. *Reduction description.*

   The adversary $\mathscr{A}$ of the Hiding(Π_{Com}) game (see Figure 5.4) receives from the challenger a commitment c to a randomly generated bit b. $\mathscr{A}$ generates a commitment c' to a random bit b' and sends c, c' to $\mathscr{A}'$, the adversary to the dGA-IP game with non-negligible advantage. $\mathscr{A}$ receives a response b_0 from $\mathscr{A}'$ and returns to the Hiding(Π_{Com}) challenger the bit b' if $b_0 = 1$ (i.e. $\mathscr{A}'$ has guessed that c and c' are in the same orbit), otherwise $\mathscr{A}$ returns $1 - b'$.

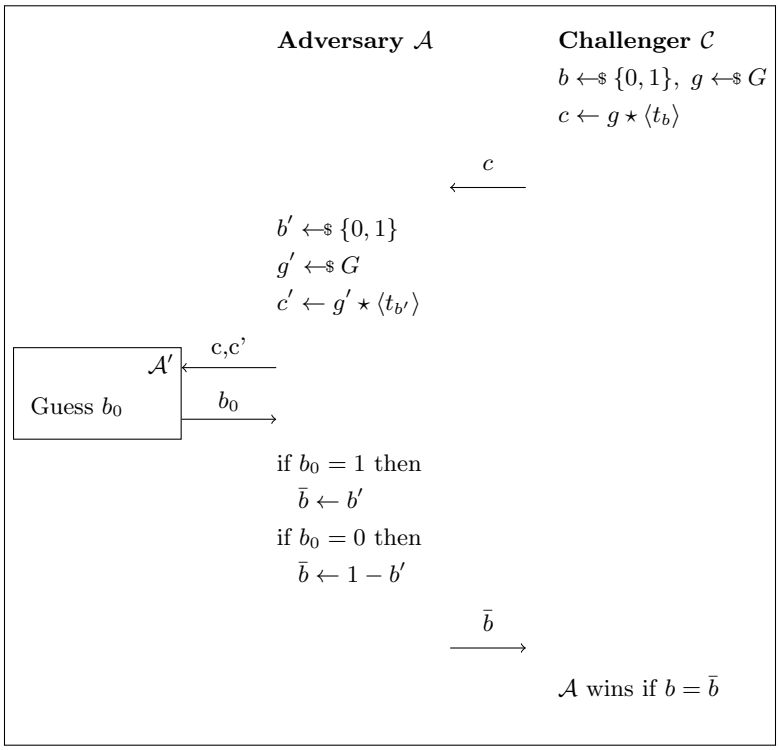

Fig. 5.4 Reduction from the hiding game for the bit commitment scheme to dGA-IP.

2. *$\mathscr{A}$ correctly simulates the dGA-IP challenger.*

   The adversary $\mathscr{A}$ receives a commitment to a random unknown bit b. Therefore, in order to simulate the dGA-IP challenger, it generates a random bit b' and a commitment to such bit. In this way, $\mathscr{A}$ generates couples of elements in X that live in the same orbit with probability $\frac{1}{2}$ as it does the dGA-IP challenger.

3. *Measurement of $\mathscr{A}$'s advantage.*

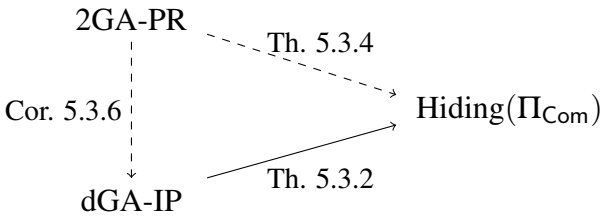

Fig. 5.5 Reductions between games and problems. "$A \to B$" means that solving B implies solving A. The reductions represented by a dashed line require the extra hypothesis about the similarity of the orbits.

The adversary $\mathscr{A}$ wins exactly with the same probability of $\mathscr{A}'$, since every time $\mathscr{A}'$ guesses the right answer to the dGA-IP game, $\mathscr{A}$ learns the orbit in which the element c lies since it knows the orbit of c'. Therefore, if $\mathscr{A}'$ wins the dGA-IP game with non-negligible advantage, also $\mathscr{A}$ wins the Hiding(Π_{Com}) game with non-negligible advantage.

□

We summarize the reductions between the hiding game of the commitment scheme and the two assumptions in Figure 5.5.

5.4 Linkable Commitments

The proposed bit commitment has the following additional feature. Given two commitments $\mathtt{com}_0$ and $\mathtt{com}_1$, if we suppose that the sender is honest, there is a way to prove that their committed value is the same. Based on this notion, we define the concept of *linkable commitment*. We require that the sender is honest to be assured that the commitments lie either in the orbit of $\langle t_0 \rangle$ or $\langle t_1 \rangle$. To the best of our knowledge, this property has not been formally defined before. However, it is well known that, for example, Pedersen commitments enjoy this property which is used, among other things, in the Monero's RingCT protocol [72].

Definition 5.4.1. *Let* $\Pi_{\mathsf{Com}} = (\mathsf{PGen}, \mathsf{Commit}, \mathsf{Open})$ *be a commitment scheme. Let* m_0 *and* m_1 *be two messages and let* $(\mathtt{com}_0, r_0) = \mathsf{Commit}(m_0)$ *and* $(\mathtt{com}_1, r_1) = \mathsf{Commit}(m_1)$. *We say that* Π_{Com} *is* linkable *if there exist the two following* PPT *algorithms:*

1. $\mathsf{LinkMaterial}(r_0, r_1)$, *whose output is a value* r_L*;*
2. $\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L)$*, that returns 1 if* $m_0 = m_1$ *and 0 otherwise.*

In order to be secure, a linkable bit commitment must satisfy some security properties for these two additional algorithms Link and LinkMaterial as well. First, we want that the linking material r_L does not reveal any information about the committed value. This means that an adversary that has access to two commitments of m and the linking material r_L does not learn anything about m. We call this property *linkable-hiding*. Then, it must not be possible to link two commitments that are obtained starting from two distinct values. A linkable commitment with this property is said *linkable-binding*. Finally, we focus on how the value r_L can be generated. We want that, if a user (somehow) knows that two commitments are linked without knowing their opening material, he can not generate a proof of that (via the linking material). In other words, being m a message, and being $(\mathtt{com}_0, r_0) = \mathsf{Commit}(m)$ and $(\mathtt{com}_1, r_1) = \mathsf{Commit}(m)$, no one can generate a value r_L such that $\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1$ without knowledge of any information regarding the opening materials r_0 and r_1. This additional property is called *link secrecy*.
We formalize these new properties in the following definition.

Definition 5.4.2. *Let* $\mathrm{HidingLink}(\Pi_{\mathsf{Com}})$ *be the game described in Figure 5.6. We define the advantage of an adversary* $\mathscr{A}$ *of the game* $\mathrm{HidingLink}(\Pi_{\mathsf{Com}})$ *as*

$$\mathsf{Adv}(\mathscr{A}, \mathrm{HidingLink}(\Pi_{\mathsf{Com}})) = \left| \Pr\left[\mathscr{A} \textit{ wins } \mathrm{HidingLink}(\Pi_{\mathsf{Com}})\right] - \frac{1}{2} \right|.$$

Let λ *be the security parameter. A linkable bit commitment*
$\Pi_{\mathsf{Com}} = (\mathsf{PGen}, \mathsf{Commit}, \mathsf{Open}, \mathsf{LinkMaterial}, \mathsf{Link})$ *is said*

- *computationally* linkable-hiding *if for all* PPT *adversaries* $\mathscr{A}$ *there is a negligible function* $\mu(\lambda)$ *such that*

$$\mathsf{Adv}(\mathscr{A}, \mathrm{HidingLink}(\Pi_{\mathsf{Com}})) \leq \mu(\lambda);$$

- *computationally* linkable-binding *if for all* PPT *adversaries* $\mathscr{A}$ *there is a negligible function* $\mu(\lambda)$ *such that*

$$\Pr\left[\begin{array}{c|c} \mathsf{pp} \leftarrow \mathsf{PGen}(1^\lambda), & m_0 \neq m_1, \\ (m_0, \mathtt{com}_0, m_1, \mathtt{com}_1, r_L) \leftarrow \mathscr{A}(\mathsf{pp}) & \mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1 \end{array}\right] \leq \mu(\lambda);$$

- *computationally* link secret *if for all* PPT *adversaries* $\mathscr{A}$ *there is a negligible function* $\mu(\lambda)$ *such that*

$$\Pr\left[\mathscr{A} \textit{ wins } \text{LinkSecrecy}(\Pi_{\mathsf{Com}})\right] \leq \mu(\lambda),$$

where $\text{LinkSecrecy}(\Pi_{\mathsf{Com}})$ *is the linking secrecy game in Figure 5.7.*

In the above definitions, whenever $\mu(\lambda) = 0$*, we say that the property is* perfect.

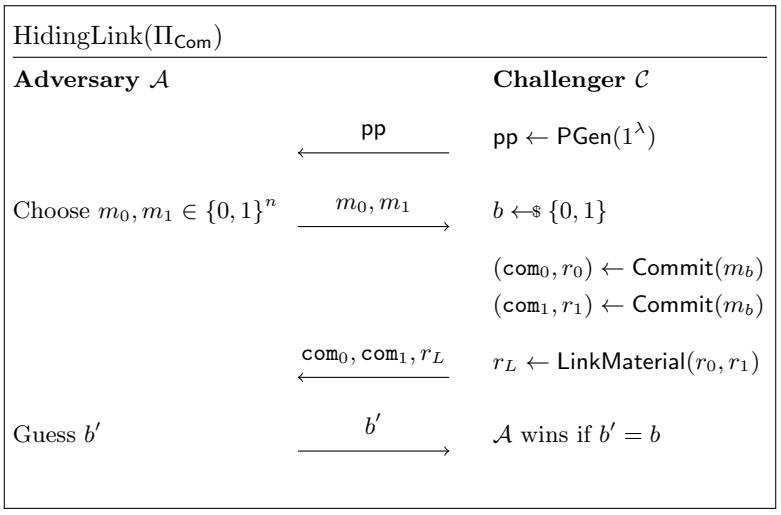

Fig. 5.6 Linkable-hiding game.

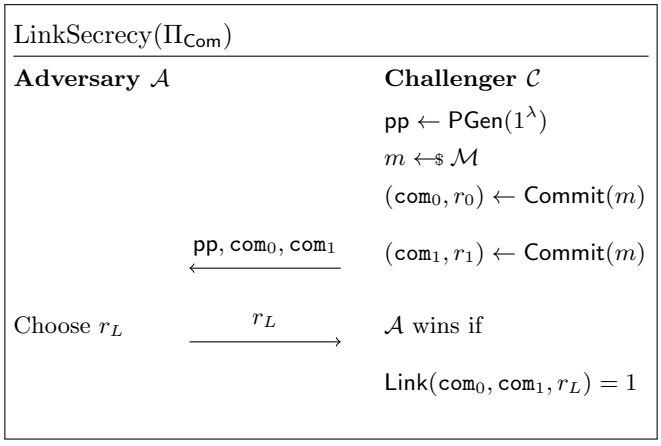

Fig. 5.7 Link secrecy game.

$\mathsf{LinkMaterial}(m, r_0, r_1)$

1 : **return** $r_0 r_1^{-1}$

$\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L)$

1 : **if** $r_L \star \mathtt{com}_1 = \mathtt{com}_0$
2 : **return** 1
3 : **else return** 0

Fig. 5.8 Algorithm for linking commitment from a GACE.

5.4.1 Linkable bit commitment from GACE

Using the bit commitment shown in Subsection 5.3.1, we can endow the scheme to obtain a linkable bit commitment. This extension is natural since the commitments of a chosen message are in the orbit of that message, and showing that they are linked reduces to exhibit a group element which sends one into the other.

Theorem 5.4.3. *The bit commitment scheme in Figure 5.1 endowed with the algorithms in Figure 5.8 is a secure linkable bit commitment scheme under the One-Way Group Action and dGA-IP assumptions.*

Proof. We have already proven in Theorem 5.3.3 that the bit commitment in Figure 5.1 is secure under the dGA-IP assumption. Now, we prove that the linkable commitment scheme is secure, namely it is computationally linkable-hiding, perfectly linkable-binding and computationally link secret.

- **Linkable-hiding**. We show that the Hiding game reduces to the HidingLink game. The idea is to let the adversary of the $\mathrm{Hiding}(\Pi_{\mathsf{Com}})$ game to simulate the HidingLink game challenger by creating a new random commitment (and the linking material) to the same message of the commitment it has received from its challenger. Now we explain it in greater detail.

  Let $\mathscr{A}'$ be an adversary that wins the HidingLink game with non-negligible advantage $\varepsilon(\lambda)$. We can define an adversary $\mathscr{A}$ for the Hiding game that wins with a non-negligible advantage. Since we are in the binary case, the challenger $\mathscr{C}$ picks a message b and sends to $\mathscr{A}$ the commitment $\mathtt{com}$ of b. Now $\mathscr{A}$ picks a random element g in G and computes $\mathtt{com}' = g \star \mathtt{com}$, which is a valid and randomly generated commitment to b. $\mathscr{A}$ queries to $\mathscr{A}'$, the adversary of the HidingLink game, the commitments $\mathtt{com}$, $\mathtt{com}'$ and the linking material g. Note that $\mathscr{A}$ correctly simulates the challenger of the HidingLink

game since the bit b and $\mathtt{com}$ are chosen at random from $\mathscr{C}$, $\mathtt{com}'$ is chosen at random from $\mathscr{A}$ and the linking material is valid.

$\mathscr{A}'$ returns a bit b' which $\mathscr{A}$ sends to $\mathscr{C}$ as its guess. If $\mathscr{A}'$ correctly guesses the bit committed to in $\mathtt{com}$ and $\mathtt{com}'$ then clearly also $\mathscr{A}$ wins its game. Therefore the advantage of $\mathscr{A}$ is the same as the one of $\mathscr{A}'$ and is non-negligible.

We can conclude that, since the commitment Π_{Com} is computationally hiding under the dGA-IP assumption, it is also computationally linkable-hiding.

- **Perfectly linkable-binding**. Suppose that an adversary returns with positive probability a tuple $(m_0, m_1, \mathtt{com}_0, \mathtt{com}_1, r_L)$ such that $m_0 \neq m_1$ and

$$\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1.$$

By construction, there exist two elements g_0 and g_1 in G such that

$$\mathtt{com}_0 = g_0 \star \langle m_0 \rangle \text{ and } \mathtt{com}_1 = g_1 \star \langle m_1 \rangle$$

From $\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1$ we have that $r_L \star \mathtt{com}_1 = \mathtt{com}_0$, and hence $\mathtt{com}_0$ and $\mathtt{com}_1$ are in the same orbit. Since $m_0 = f(\mathtt{com}_0) = f(\mathtt{com}_1) = m_1$, where f is the invariant function in the GACE, we have a contradiction. Hence, there are no adversaries that can output such a tuple with positive probability.

- **Computationally link secret**. We show that, if a PPT adversary $\mathscr{A}$, on input $\mathtt{com}_0$ and $\mathtt{com}_1$, can find r_L such that $\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1$, then it contradicts the One-way group action assumption. Essentially, if $\mathtt{com}_0$ and $\mathtt{com}_1$ are commitments to m_0, then they are in the same orbit of $\langle m_0 \rangle$. Finding an r_L in G such that $\mathsf{Link}(\mathtt{com}_0, \mathtt{com}_1, r_L) = 1$ means finding an element of G sending $\mathtt{com}_1$ to $\mathtt{com}_0$, and this is intractable by hypothesis.

□

Remark 5.4.4. *Observe that, if an inadmissible value is committed, for instance, an element x that is not in the orbit of $\langle t_0 \rangle$ nor $\langle t_1 \rangle$, then the linkability continues to work. In fact, two commitments of the above x can be linked. Therefore we refer to the above scheme as a* honest sender *linkable commitment. To cover even the case where the sender may commit to an inadmissible value, some techniques from ring signature schemes can be used. Using the framework of Beullens, Katsumata*

and Pintore [11], a proof of the legitimacy of the commitment can be generated in the random oracle model. In the commit phase, the sender generates $(\mathtt{com}, r)$ *from* $\mathsf{Commit}(b)$*, then attaches to* $\mathtt{com}$ *a non-interactive proof of the OR-relation*

$$\{(\mathtt{com}, g) \mid \mathtt{com} = g \star \langle t_0 \rangle \text{ or } \mathtt{com} = g \star \langle t_1 \rangle\}.$$

We refer to [11] for the details. However, this proof needs many repetitions to achieve a reasonable security level, leading to a huge cost in communication.

5.5 An Instantiation with Tensors

5.5.1 GACE and bit commitment from tensors

Recall that, given a tensor T in $\mathbb{F}_q^n \otimes \mathbb{F}_q^n \otimes \mathbb{F}_q^n$, its rank is the smallest r such that T can be written as sum of rank-one tensors (see Subsection 2.5.1). Moreover, the action of $\mathrm{GL}(n) \times \mathrm{GL}(n) \times \mathrm{GL}(n)$ does not change the rank of the tensors. Starting from this group action , we want to build a Group Action with Canonical Element. Since the computation of the rank is supposed to be hard, we set $T = \mathbb{N}$ and

$$f : \mathbf{V} \to \mathbb{N},\ M \mapsto \mathrm{rk}(M).$$

In order to define the function $\langle \cdot \rangle$, we need to do some observations. From Eq. (**??**), we see that the rank of a tensor is at most n^3 and with a simple trick it can be shown that it is at most n^2. Actually, the maximal rank is strictly less than this value. As shown in [49], the maximal rank attainable by a tensor in $\mathbf{V}$ is between $\frac{1}{3}n^2$ and $\frac{3}{4}n^2$. Moreover, an open problem in this field is to exhibit the explicit construction of a high-rank tensor. Even if there are some results [14, 84, 2], we are not able to construct a tensor of any given rank. Luckily, there is a set of integers for which we can easily exhibit tensors of a given rank. Let $T' = \{1, \ldots, n\}$ and we can define the function

$$\begin{aligned} \langle \cdot \rangle :\ & T' \to \mathbf{V}, \\ & r \mapsto \sum_{i=1}^{r} e_i \otimes e_i \otimes e_i. \end{aligned}$$

We can see that $f(\langle r \rangle) = r$ for any r in $T' = \{1, \ldots, n\}$, hence the tuple $(G, \mathbf{V}, \star, f, \langle \cdot \rangle)$ is a GACE. In fact, computing the rank of a random tensor of promised rank between 1 and n is hard, while recognizing the rank of $\langle r \rangle$ is easy.

The non-interactive bit commitment scheme we present is based on the general one in Figure 5.1. During the parameter generation phase, we choose $n-1$ and n as elements of T' encoding the bits 0 and 1, respectively.

Concretely, given a security parameter λ, a prime power q, an integer n and the tensor space $\mathbf{V} = \mathbb{F}_q^n \otimes \mathbb{F}_q^n \otimes \mathbb{F}_q^n$, the public parameters are

$$(G, \mathbf{V}, \star, f, \langle \cdot \rangle, n-1, n).$$

Let us analyze the assumptions on this particular group action. The dGA-IP assumption for tensors is related to the Tensor Isomorphism problem [44, 45], which is complete for a large class of problems and it is conjectured hard even for a quantum computer. The One-Way assumption on tensors is linked to the computational version of the dGA-IP problem: given two tensors in the same orbit, find the group element that links them. This problem is believed to be hard and it is directly used in various cryptosystems [25, 50], while other constructions use polynomially equivalent problems [83]. When we consider just the orbits of rank n and $n-1$, these assumptions seem to remain intractable.

Summarizing, to commit to a bit b, the sender picks a random g in G and obtains the commitment `com` equal to $g \star \langle n-1 \rangle$ if $b = 0$ or $g \star \langle n \rangle$ if $b = 1$. The opening material is given by g. To open the commitment `com`, the sender communicates to the receiver b and g and the latter checks that $g^{-1} \star \texttt{com}$ is equal to $\langle n-1 \rangle$ or $\langle n \rangle$. There is one additional check to take care of during the opening phase: the receiver must verify the membership of g to G. In fact, if $g = (A, B, C)$ and A, B or C are non-invertible, then g can send a tensor of rank n to a tensor of rank $n-1$, breaking the binding property.
Analogously, a linkable bit commitment can be designed on tensors with the constructions given in Subsection 5.4.1.

5.5.2 An attack

At the conference CRYPTO 2024, Gilchrist, Marco, Petit and Tang presented an attack to the above instantiation with 3-tensors [69]. The proposed attack exploits the use of low-rank tensors and the fact that they admit low-rank points. Moreover, the article proposes an algorithm for the decisional and computational Tensor Isomorphism problem on low-rank elements, on which the commitment scheme bases its security, in particular, the hiding property does not hold.
On the theoretical side, they present an efficient way to compute the rank of a low-rank tensor, investigating both the Tensor Rank problem and the Tensor Isomorphism problem in some special cases. On the practical side, they show that the presented example of GACE is not secure and capable of cryptographic constructions. However, the authors propose a fix to the commitment scheme: the GACE framework is not needed anymore, but the binding property became statistical instead of the previous perfect. This downgrade is ineffective for a lot of cryptographic applications that use bit commitments.

Hence, after this attack, the problem of finding suitable group actions with canonical elements remains open.

References

[1] Alamati, N., De Feo, L., Montgomery, H., and Patranabis, S. (2020). Cryptographic group actions and applications. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 411–439. Springer.

[2] Alexeev, B., Forbes, M. A., and Tsimerman, J. (2011). Tensor rank: Some lower and upper bounds. In *2011 IEEE 26th Annual Conference on Computational Complexity*, pages 283–291. IEEE.

[3] Babai, L. (2016). Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697.

[4] Barenghi, A., Biasse, J.-F., Persichetti, E., and Santini, P. (2021). LESS-FM: fine-tuning signatures from the code equivalence problem. In *International Conference on Post-Quantum Cryptography*, pages 23–43. Springer.

[5] Barenghi, A., Biasse, J.-F., Persichetti, E., and Santini, P. (2023). On the computational hardness of the code equivalence problem in cryptography. *Advances in Mathematics of Communications*, 17(1):23–55.

[6] Battagliola, M., Borin, G., Meneghetti, A., and Persichetti, E. (2024). Cutting the grass: threshold group action signature schemes. In *Cryptographers' Track at the RSA Conference*, pages 460–489. Springer.

[7] Benčina, B., Budroni, A., Chi-Domínguez, J.-J., and Kulkarni, M. (2023). Properties of Lattice Isomorphism as a Cryptographic Group Action. Post-Quantum Cryptography: 15th International Workshop, PQCrypto 2024. https://eprint.iacr.org/2023/1093.

[8] Bernstein, D. J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., Papachristodoulou, L., Schneider, M., Schwabe, P., and Wilcox-O'Hearn, Z. (2015). Sphincs: practical stateless hash-based signatures. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 368–397. Springer.

[9] Bernstein, Daniel J (2023). Has anyone really cracked anything recently? https://archive.ph/BHGOM#selection-31853.41-31853.66. Accessed: 2024-07-18.

[10] Beullens, W., Dobson, S., Katsumata, S., Lai, Y.-F., and Pintore, F. (2023). Group signatures and more from isogenies and lattices: generic, simple, and efficient. *Designs, Codes and Cryptography*, pages 1–60.

[11] Beullens, W., Katsumata, S., and Pintore, F. (2020). Calamari and Falafl: logarithmic (linkable) ring signatures from isogenies and lattices. In *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security, Daejeon, South Korea, December 7–11, 2020, Proceedings, Part II*, pages 464–492. Springer.

[12] Beullens, W., Kleinjung, T., and Vercauteren, F. (2019). CSI-FiSh: efficient isogeny based signatures through class group computations. In *Advances in Cryptology–ASIACRYPT 2019: 25th International Conference on the Theory and Application of Cryptology and Information Security, Kobe, Japan, December 8–12, 2019, Proceedings, Part I*, pages 227–247. Springer.

[13] Biasse, J.-F., Micheli, G., Persichetti, E., and Santini, P. (2020). Less is more: code-based signatures without syndromes. In *Progress in Cryptology-AFRICACRYPT 2020: 12th International Conference on Cryptology in Africa, Cairo, Egypt, July 20–22, 2020, Proceedings 12*, pages 45–65. Springer.

[14] Bläser, M. (2014). Explicit tensors. *Perspectives in Computational Complexity: The Somenath Biswas Anniversary Volume*, pages 117–130.

[15] Blum, M. and Micali, S. (2019). How to generate cryptographically strong sequences of pseudo random bits. In *Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali*, pages 227–240.

[16] Boppana, R. B., Hastad, J., and Zachos, S. (1987). Does co-NP have short interactive proofs? *Information Processing Letters*, 25(2):127–132.

[17] Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J. M., Schwabe, P., Seiler, G., and Stehlé, D. (2018). CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 353–367. IEEE.

[18] Brassard, G. and Yung, M. (1991). One-way group actions. In *Advances in Cryptology-CRYPTO'90: Proceedings 10*, pages 94–107. Springer.

[19] Bruck, J. and Naor, M. (1990). The hardness of decoding linear codes with preprocessing. *IEEE Transactions on Information Theory*, 36(2):381–385.

[20] Camps-Moreno, E., Gorla, E., Landolina, C., García, E. L., Martínez-Peñas, U., and Salizzoni, F. (2022). Optimal anticodes, MSRD codes, and generalized weights in the sum-rank metric. *IEEE Transactions on Information Theory*, 68(6):3806–3822.

[21] Castryck, W. (2024). An Attack Became a Tool: Isogeny-based Cryptography 2.0. *Invited talk at Eurocrypt 2024*.

[22] Castryck, W. and Decru, T. (2023). An efficient key recovery attack on sidh. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 423–447. Springer.

[23] Castryck, W., Lange, T., Martindale, C., Panny, L., and Renes, J. (2018). CSIDH: an efficient post-quantum commutative group action. In *Advances in Cryptology–ASIACRYPT 2018: 24th International Conference on the Theory and Application of Cryptology and Information Security, Brisbane, QLD, Australia, December 2–6, 2018, Proceedings, Part III 24*, pages 395–427. Springer.

[24] Chou, T., Niederhagen, R., Persichetti, E., Randrianarisoa, T. H., Reijnders, K., Samardjiska, S., and Trimoska, M. (2023a). Take your meds: Digital signatures from matrix code equivalence. In *International Conference on Cryptology in Africa*, pages 28–52. Springer.

[25] Chou, T., Niederhagen, R., Persichetti, E., Randrianarisoa, T. H., Reijnders, K., Samardjiska, S., and Trimoska, M. (2023b). Take your meds: Digital signatures from matrix code equivalence. In *International Conference on Cryptology in Africa*, pages 28–52. Springer.

[26] Couveignes, J.-M. (2006). Hard homogeneous spaces. *Cryptology ePrint Archive*.

[27] Couvreur, A., Debris-Alazard, T., and Gaborit, P. (2020). On the hardness of code equivalence problems in rank metric. *arXiv preprint arXiv:2011.04611*.

[28] D'Alconzo, G. (2024). Monomial isomorphism for tensors and applications to code equivalence problems. *Designs, Codes and Cryptography*, pages 1–22.

[29] D'Alconzo, G. and Di Scala, A. J. (2023). Representations of Group Actions and their Applications in Cryptography. *Cryptology ePrint Archive*.

[30] D'Alconzo, G., Flamini, A., and Gangemi, A. (2023). Non-interactive commitment from non-transitive group actions. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 222–252. Springer.

[31] Darwish, A. and El-Gendy, M. M. (2017). A new cryptographic voting verifiable scheme for e-voting system based on bit commitment and blind signature. *Int J Swarm Intel Evol Comput*, 6(158):2.

[32] De Feo, L., Fouotsa, T. B., Kutas, P., Leroux, A., Merz, S.-P., Panny, L., and Wesolowski, B. (2023). SCALLOP: scaling the CSI-FiSh. In *IACR international conference on public-key cryptography*, pages 345–375. Springer.

[33] De Feo, L. and Galbraith, S. D. (2019). SeaSign: compact isogeny signatures from class group actions. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 759–789. Springer.

[34] Dickson, L. E. (1908). Representations of the general symmetric group as linear groups in finite and infinite fields. *Transactions of the American Mathematical Society*, 9(2):121–148.

[35] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., and Stehlé, D. (2018). CRYSTALS-Dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 238–268.

[36] Ducas, L., Postlethwaite, E. W., Pulles, L. N., and Woerden, W. v. (2023). Hawk: Module LIP makes lattice signatures fast, compact and simple. In *Advances in Cryptology–ASIACRYPT 2022: 28th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, December 5–9, 2022, Proceedings, Part IV*, pages 65–94. Springer.

[37] Ducas, L. and van Woerden, W. (2022). On the lattice isomorphism problem, quadratic forms, remarkable lattices, and cryptography. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 643–673. Springer.

[38] Fiat, A. and Shamir, A. (1986). How to prove yourself: Practical solutions to identification and signature problems. In *Conference on the theory and application of cryptographic techniques*, pages 186–194. Springer.

[39] Fouque, P.-A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z., et al. (2018). Falcon: Fast-Fourier lattice-based compact signatures over NTRU. *Submission to the NIST's post-quantum cryptography standardization process*, 36(5):1–75.

[40] Frederiksen, T. K., Pinkas, B., and Yanai, A. (2018). Committed MPC: Maliciously Secure Multiparty Computation from Homomorphic Commitments. In *Public-Key Cryptography–PKC 2018: 21st IACR International Conference on Practice and Theory of Public-Key Cryptography, Rio de Janeiro, Brazil, March 25-29, 2018, Proceedings, Part I 21*, pages 587–619. Springer.

[41] Futorny, V., Grochow, J. A., and Sergeichuk, V. V. (2019). Wildness for tensors. *Linear Algebra and its Applications*, 566:212–244.

[42] Goldreich, O., Micali, S., and Wigderson, A. (1991). Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM (JACM)*, 38(3):690–728.

[43] Grochow, J. and Qiao, Y. (2023). On the Complexity of Isomorphism Problems for Tensors, Groups, and Polynomials I: Tensor Isomorphism-Completeness. *SIAM Journal on Computing*, 52(2):568–617.

[44] Grochow, J. A. and Qiao, Y. (2019). Isomorphism problems for tensors, groups, and cubic forms: completeness and reductions. *arXiv preprint arXiv:1907.00309*.

[45] Grochow, J. A. and Qiao, Y. (2021). On the complexity of isomorphism problems for tensors, groups, and polynomials I: Tensor Isomorphism-completeness. In *12th Innovations in Theoretical Computer Science Conference (ITCS 2021)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.

[46] Grochow, J. A., Qiao, Y., and Tang, G. (2021). Average-Case Algorithms for Testing Isomorphism of Polynomials, Algebras, and Multilinear Forms. In *38th International Symposium on Theoretical Aspects of Computer Science*.

[47] Håstad, J. (1989). Tensor rank is NP-complete. In *International Colloquium on Automata, Languages, and Programming*, pages 451–460. Springer.

[48] Heimberger, L., Hennerbichler, T., Meisingseth, F., Ramacher, S., and Rechberger, C. (2024). Oprfs from isogenies: Designs and analysis. In *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ASIA CCS '24, page 575–588, New York, NY, USA. Association for Computing Machinery.

[49] Howell, T. D. (1978). Global properties of tensor rank. *Linear Algebra and its Applications*, 22:9–23.

[50] Ji, Z., Qiao, Y., Song, F., and Yun, A. (2019). General linear group action on tensors: a candidate for post-quantum cryptography. In *Theory of Cryptography Conference*, pages 251–281. Springer.

[51] Juels, A., Luby, M., and Ostrovsky, R. (2006). Security of blind digital signatures. In *Advances in Cryptology—CRYPTO'97: 17th Annual International Cryptology Conference Santa Barbara, California, USA August 17–21, 1997 Proceedings*, pages 150–164. Springer.

[52] Kobler, J., Schöning, U., and Torán, J. (2012). *The graph isomorphism problem: its structural complexity*. Springer Science & Business Media.

[53] Lai, Y.-F. (2023). CAPYBARA and TSUBAKI: Verifiable Random Functions from Group Actions and Isogenies. *Cryptology ePrint Archive*.

[54] Lamport, L. (1979). Constructing digital signatures from a one way function.

[55] Leroux, A. and Roméas, M. (2024). Updatable encryption from group actions. In *International Conference on Post-Quantum Cryptography*, pages 20–53. Springer.

[56] Lidl, R. and Niederreiter, H. (1997). *Finite fields*, volume 20 of *Encyclopedia of Mathematics and its Applications*. Cambridge University Press, Cambridge, second edition. With a foreword by P. M. Cohn.

[57] Lyubashevsky, V., Nguyen, N. K., and Seiler, G. (2021). Shorter lattice-based zero-knowledge proofs via one-time commitments. In *Public-Key Cryptography–PKC 2021: 24th IACR International Conference on Practice and Theory of Public Key Cryptography, Virtual Event, May 10–13, 2021, Proceedings, Part I*, pages 215–241. Springer.

[58] MacWilliams, F. J. (1962). *Combinatorial problems of elementary abelian groups*. PhD thesis.

[59] Maino, L., Martindale, C., Panny, L., Pope, G., and Wesolowski, B. (2023). A direct key recovery attack on sidh. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 448–471. Springer.

[60] Martínez-Peñas, U. (2020). Hamming and simplex codes for the sum-rank metric. *Designs, Codes and Cryptography*, 88(8):1521–1539.

[61] Merkle, R. and Hellman, M. (1978). Hiding information and signatures in trapdoor knapsacks. *IEEE transactions on Information Theory*, 24(5):525–530.

[62] Morrison, K. (2014). Equivalence for rank-metric and matrix codes and automorphism groups of Gabidulin codes. *IEEE Transactions on Information Theory*, 60(11):7035–7046.

[63] Naor, M. (1991). Bit commitment using pseudorandomness. *Journal of cryptology*, 4:151–158.

[64] Neri, A. (2022). Twisted linearized Reed-Solomon codes: A skew polynomial framework. *Journal of Algebra*, 609:792–839.

[65] NIST (2017). Post-Quantum Cryptography Standardization. https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization. Accessed: 2024-07-18.

[66] NIST (2023). Post-quantum cryptography: Digital signature schemes. https://csrc.nist.gov/Projects/pqc-dig-sig/standardization. Accessed: 2024-04-18.

[67] Ostrovsky, R., Persiano, G., and Visconti, I. (2009). Simulation-based concurrent non-malleable commitments and decommitments. In *Theory of Cryptography: 6th Theory of Cryptography Conference, TCC 2009, San Francisco, CA, USA, March 15-17, 2009. Proceedings 6*, pages 91–108. Springer.

[68] Patarin, J. (1996). Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 33–48. Springer.

[69] Petit, C., Tang, G., Gilchrist, V., and Marco, L. (2024). Solving the Tensor Isomorphism Problem for special orbits with low rank points: Cryptanalysis and repair of an Asiacrypt 2023 commitment scheme. In *Advances in Cryptology – CRYPTO 2024*, Lecture Notes in Computer Science. Springer. Not yet published as of 17/07/2024. Conference to take place August 2024.; CRYPTO 2024 ; Conference date: 18-08-2024 Through 22-08-2024.

[70] Petrank, E. and Roth, R. M. (1997). Is code equivalence easy to decide? *IEEE Transactions on Information Theory*, 43(5):1602–1604.

[71] Pham, M. T. T., Duong, D. H., Li, Y., and Susilo, W. (2023). Threshold Ring Signature Scheme from Cryptographic Group Action. In *International Conference on Provable Security*, pages 207–227. Springer.

[72] Poelstra, A., Back, A., Friedenbach, M., Maxwell, G., and Wuille, P. (2019). Confidential assets. In *Financial Cryptography and Data Security: FC 2018 International Workshops, BITCOIN, VOTING, and WTSC, Nieuwpoort, Curaçao, March 2, 2018, Revised Selected Papers 22*, pages 43–63. Springer.

[73] Reijnders, K. (2023). Transparent Security for Cryptographic Group Actions. *Talk at CBCrypto 2023 International Workshop on Code-Based Cryptography*.

[74] Reijnders, K., Samardjiska, S., and Trimoska, M. (2024). Hardness estimates of the code equivalence problem in the rank metric. *Designs, Codes and Cryptography*, pages 1–30.

[75] Robert, D. (2023). Breaking sidh in polynomial time. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 472–503. Springer.

[76] Rostovtsev, A. and Stolbunov, A. (2006). Public-key cryptosystem based on isogenies. *Cryptology ePrint Archive*.

[77] Schaefer, M. and Štefankovič, D. (2018). The complexity of tensor rank. *Theory of Computing Systems*, 62:1161–1174.

[78] Sendrier, N. (2000). Finding the permutation between equivalent linear codes: The support splitting algorithm. *IEEE Transactions on Information Theory*, 46(4):1193–1203.

[79] Sendrier, N. and Simos, D. E. (2013). The Hardness of Code Equivalence over F_q and Its Application to Code-Based Cryptography. In *Post-Quantum Cryptography: 5th International Workshop, PQCrypto 2013, Limoges, France, June 4-7, 2013. Proceedings 5*, pages 203–216. Springer.

[80] Shitov, Y. (2016). How hard is the tensor rank? *arXiv preprint arXiv:1611.01559*.

[81] Shor, P. W. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. IEEE.

[82] Stolbunov, A. (2012). Cryptographic schemes based on isogenies.

[83] Tang, G., Duong, D. H., Joux, A., Plantard, T., Qiao, Y., and Susilo, W. (2022). Practical Post-Quantum Signature Schemes from Isomorphism Problems of Trilinear Forms. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 582–612. Springer.

[84] Weitz, B. (2011). An improvement on ranks of explicit tensors. *arXiv preprint arXiv:1102.0580*.