Powers of Codes and Applications to Cryptography

Ignacio Cascudo, Department of Computer Science, Aarhus University.

Presented at IEEE Information Theory Workshop, Jerusalem, April 2015.*

Abstract

Given a linear error correcting code C, its m-th power is defined as the linear span of the set of all coordinate-wise products of m (not necessarily distinct) codewords in C. The study of powers of codes (and especially squares) is relevant in a number of recent results in several areas of cryptography where we need to bound certain parameters (such as the dimension and the minimum distance) of both a linear code and some power of it simultaneously. These areas include most notably secret sharing and multiparty computation, but also two-party cryptography and public key cryptography. In this paper, some of these applications will be discussed together with several recent results and some open challenges.

1 Powers of codes

We start by recalling some definitions from coding theory and fixing some notations. For additional information on the topic the reader is referred to [27].

Let $\mathbb{K}$ be a finite field. A *linear code* C over $\mathbb{K}$ of length n is a $\mathbb{K}$-vector subspace $C \subseteq \mathbb{K}^n$. The *dimension* of C is its dimension as a $\mathbb{K}$-vector space. Given a vector $\mathbf{v} \in \mathbb{K}^n$, its i-th coordinate will be denoted as v_i, so $\mathbf{v} = (v_1, v_2, \ldots, v_n)$. For a subset $A = \{i_1, i_2, \ldots, i_r\}$ of $\{1, \ldots, n\}$, where $i_1 < i_2 < \cdots < i_r$, $\mathbf{v}_A$ denotes the vector $(v_{i_1}, v_{i_2}, \ldots, v_{i_r})$. The support of $\mathbf{v}$ is $\operatorname{supp} \mathbf{v} := \{i : v_i \neq 0\}$ and its Hamming weight is $w_H(\mathbf{v}) := \#\operatorname{supp}(\mathbf{v})$, i.e., the number of nonzero coordinates of $\mathbf{v}$. The Hamming distance between two vectors $\mathbf{v}, \mathbf{w} \in \mathbb{K}^n$ is $d_H(\mathbf{v}, \mathbf{w}) := w_H(\mathbf{v} - \mathbf{w})$.

The minimum distance of C, denoted as $d(C)$, is

$$d(C) := \min\{d_H(\mathbf{c}, \mathbf{c}') : \mathbf{c}, \mathbf{c}' \in C, \mathbf{c} \neq \mathbf{c}'\}.$$

*Proceedings version pbulished by IEEE with DOI: 10.1109/ITW.2015.7133155, can be found at `https://ieeexplore.ieee.org/abstract/document/7133155`.

Equivalently, $d(C) = \min\{w_H(\mathbf{c}) : \mathbf{c} \in C, \mathbf{c} \neq \mathbf{0}\}$.

The dual $C^\perp \subseteq \mathbb{K}^n$ of C is defined as $C^\perp := \{\mathbf{c}^* \in \mathbb{K}^n : \langle \mathbf{c}^*, \mathbf{c}\rangle = 0 \ \forall\, \mathbf{c} \in C\}$ (where $\langle \cdot, \cdot \rangle$ denotes inner product). This is also a linear code of length n. If the dimension of C is k then the dimension of $C^\perp$ is $n - k$.

Given $\mathbf{v}, \mathbf{w} \in \mathbb{K}^n$, their coordinate product is denoted by $\mathbf{v} * \mathbf{w}$, i.e., $\mathbf{v} * \mathbf{w} = (v_1 w_1, v_2 w_2, \ldots, v_n w_n)$.

DEFINITION 1.1. *Let $C \subseteq \mathbb{K}^n$ be a linear code and $m \geq 1$ be an integer. The m-th power of C is the $\mathbb{K}$-linear span of the set $\{\mathbf{c}^{(1)} * \mathbf{c}^{(2)} * \cdots * \mathbf{c}^{(m)} : \mathbf{c}^{(1)}, \mathbf{c}^{(2)}, \ldots, \mathbf{c}^{(m)} \in C\}$ and it is denoted C^{*m}.*

That is, C^{*m} is the smallest linear code containing all products of m *non-necessarily distinct* words in C. In the case $m = 2$, we talk about the square of C.

It is easy to see that $\dim C^{*m} \geq \dim C$. Moreover $\operatorname{supp}(\mathbf{v} * \mathbf{v}) = \operatorname{supp} \mathbf{v}$ holds for every $\mathbf{v} \in \mathbb{K}^n$ and consequently we have $d(C^{*m}) \leq d(C)$. For the special case of binary codes (linear codes over a field of two elements $\mathbb{K} = \mathbb{F}_2$), we have $C \subseteq C^{*m}$ for every m (because $\mathbf{v} * \mathbf{v} = \mathbf{v}$ for every $\mathbf{v} \in \mathbb{F}_2^n$), but this does not hold for arbitrary finite fields $\mathbb{K}$.

For some well known families of linear codes, their powers belong again to the same family. For example, let n, k be integers with $0 \leq k \leq n \leq \#\mathbb{K}$. For vectors $\mathbf{x}, \mathbf{y} \in \mathbb{K}^n$, where the coordinates of $\mathbf{x}$ are pairwise distinct and the coordinates of $\mathbf{y}$ are nonzero, and a polynomial $f \in \mathbb{K}[x]$, let $\mathbf{c}_f(\mathbf{x}, \mathbf{y}) := (y_1 f(x_1), y_2 f(x_2), \ldots, y_n f(x_n))$. We define the code

$$GRS_k(\mathbf{x}, \mathbf{y}) := \{\mathbf{c}_f(\mathbf{x}, \mathbf{y}) : f \in \mathbb{K}[x], \deg f < k\},$$

a Generalized Reed Solomon code over $\mathbb{K}$ of dimension k, length n and minimum distance $n - k + 1$. Let $C := GRS_k(\mathbf{x}, \mathbf{y})$. Then for every $m \geq 1$, $C^{*m} = GRS_{k'}(\mathbf{x}, \mathbf{y}^m)$ where $k' = \min\{(k-1)m + 1, n\}$ and $\mathbf{y}^m = (y_1^m, y_2^m, \ldots, y_n^m)$. We can therefore determine the parameters (dimension and minimum distance) of the powers of C.

Another (more general) example is algebraic geometric evaluation codes, briefly described below (for more information, refer to [33]). Let $F/\mathbb{K}(x)$ be a function field with field of constants $\mathbb{K}$. Let $P_1, P_2, \ldots, P_n$ be pairwise distinct places of degree 1. Denote $D = P_1 + P_2 + \cdots + P_n$. For a function $f \in F$ without poles in any P_i, let $\mathbf{c}_f(D) = (f(P_1), f(P_2), \ldots, f(P_n))$. Now, for a divisor G of F such that no P_i is in the support of G, we define the linear code

$$C_{\mathcal{L}}(D, G) = \{\mathbf{c}_f(D) : f \in \mathcal{L}(G)\}$$

where $\mathcal{L}(G)$ denotes the Riemann-Roch space of divisor G. Its dimension is $\dim \mathcal{L}(G) - \dim \mathcal{L}(G - D)$ and it has minimum distance at least $n - \deg G$. Let $C = C_{\mathcal{L}}(D, G)$. Then $C^{*m} \subseteq C_{\mathcal{L}}(D, mG)$. In this case equality does not hold in general. However, the statement above is enough to bound the minimum distance as $d(C^{*m}) \geq d(C_{\mathcal{L}}(D, mG)) \geq n - m \deg G$.

2 Powers of codes in cryptography

2.1 Multiplicative secret sharing

A secret sharing scheme is a method to distribute the knowledge of some secret information among a number n of pieces of data, called shares, in such a way that only certain subsets of these shares allow to reconstruct the secret. A formal definition of a secret sharing scheme is not included here, for reasons of space, but it can be found in many other works, for example [16]. We will index the shares by the numbers in $\{1, \ldots, n\}$. A set $A \subseteq \{1, \ldots, n\}$ is a *privacy set* of the secret sharing scheme if the knowledge of the shares in A gives no more information about the secret than what is known a priori. On the other hand, A is a *reconstructing set* if the secret is completely determined given the shares in A. A secret sharing scheme has *t-privacy* if any set of shares of cardinality t is a privacy set, and it has *r-reconstruction* if any set of shares of cardinality r is a reconstructing set. [1]

There is a well known construction of secret sharing schemes based on linear codes, first pointed out by Massey [28]. Let C be a linear code over $\mathbb{K}$ of length $n + \ell$, dimension at least ℓ, such that the set I of its first ℓ coordinates is an information set. Then one can define the following secret sharing scheme $\Sigma_\ell(C)$, where the space of possible secrets is $\mathbb{K}^\ell$ and where each share is in $\mathbb{K}$. In order to share the secret $\mathbf{s} = (s_1, \ldots, s_\ell) \in \mathbb{K}^\ell$, select uniformly at random a word $\mathbf{c} \in C$ such that $c_I = \mathbf{s}$ (which can always be done, by the assumptions on C) and define as j-th share the value $c_{\ell+j}$. Secret sharing schemes of this form belong to a class known as *linear secret sharing schemes (LSSS)*.[2] The privacy and reconstruction thresholds of $\Sigma_\ell(C)$ can be bounded using the minimum distance of C and its dual $C^\perp$. Indeed we have:

PROPOSITION 2.1. *Let $\Sigma_\ell(C)$ be a LSSS. Then*

- *If $d(C) \geq n - r + \ell + 1$, then $\Sigma_\ell(C)$ has r-reconstruction.*
- *If $d(C^\perp) \geq t + \ell + 1$, then $\Sigma_\ell(C)$ has t-privacy.*

A more refined result can be obtained by replacing the minimum distance by the following notion. Let $\overline{I}$ denote the set of the n last coordinates in C and define

$$w^\ell(C) = min\{w_H(\mathbf{c}_{\overline{I}}) : \mathbf{c} \in C, \mathbf{c}_I \neq 0\}.$$

PROPOSITION 2.2. *Let $\Sigma_\ell(C)$ be a LSSS. Then*

- *$\Sigma_\ell(C)$ has r-reconstruction if and only if $w^\ell(C) \geq n - r + 1$.*
- *$\Sigma_\ell(C)$ has t-privacy if and only if $w^\ell(C^\perp) \geq t + 1$.*

[1] It is usually required that a secret sharing scheme has n-reconstruction, but this condition is dropped here for simplicity.

[2] The actual definition of a LSSS is more general, as it allows for shares that consist of more than one element of the field.

Proposition 2.2 is an easy consequence of the results in [28]. It is easy to see that $d(C) \leq w^\ell(C) + \ell$ and hence Proposition 2.2 implies Proposition 2.1.

While secret sharing has practical uses as a stand-alone notion, e.g. as a secure data storage mechanism, it is perhaps more interesting because of its use as a building block in cryptographic protocols. One of the most relevant applications, and the one will be concerned with here, is in the area of *secure multiparty computation* and requires *linear* secret sharing schemes with additional algebraic (*multiplicative*) properties, which are defined as follows.

DEFINITION 2.3. *Let $\Sigma_\ell(C)$ be a linear secret sharing scheme. We say that a set A of shares is product-reconstructing if there exists a linear function $\rho_A : \mathbb{K}^{\#A} \to \mathbb{K}^\ell$ such that for any secrets $\mathbf{s}, \mathbf{s}' \in \mathbb{K}^\ell$ and any valid sharings $\mathbf{a}, \mathbf{a}' \in \mathbb{K}^n$ of $\mathbf{s}$ and $\mathbf{s}'$ respectively, we have $\mathbf{s} * \mathbf{s}' = \rho_A(\mathbf{a}_A * \mathbf{a}'_A)$.*

We say that $\Sigma_\ell(C)$ has r'-product reconstruction if A is product reconstructing for any set A of size r'.

The following definitions were introduced in [15].

DEFINITION 2.4. *We say that a linear secret sharing scheme is*

- Multiplicative *if it has n-product reconstruction.*
- t-strongly multiplicative *if it has t-privacy and $(n-t)$-product reconstruction.*

It is not difficult to see that given a LSSS $\Sigma_\ell(C)$, we can also define $\Sigma_\ell(C^{*2})$ (since linear independence of the first ℓ coordinates is preserved under squaring) and, furthermore, a set A is product reconstructing in $\Sigma_\ell(C)$ if and only if it is reconstructing in $\Sigma_\ell(C^{*2})$. Combining this observation with Propositions 2.1 and 2.2 we have

PROPOSITION 2.5. *Let $\Sigma_\ell(C)$ be a LSSS.*

- *$\Sigma_\ell(C)$ is multiplicative if and only if $w^\ell(C^{*2}) \geq 1$.*
- *$\Sigma_\ell(C)$ is t-strongly multiplicative if and only if $w^\ell(C^{*2}) \geq t+1$ and $w^\ell(C^\perp) \geq t+1$.*

In particular, if $d(C^{*2}) \geq \ell + 1$ then $\Sigma_\ell(C)$ is multiplicative, and if $d(C^{*2}) \geq t + \ell + 1$ and $d(C^\perp) \geq t + \ell + 1$, then $\Sigma_\ell(C)$ is t-strongly multiplicative.

The best known example of LSSS is Shamir's scheme [32], which is $\Sigma_1(RS_{t+1})$ where RS_{t+1} is a Reed-Solomon code of dimension $t+1$. It has t-privacy and $t+1$-reconstruction and it is t-strong multiplicative as long as $3t+1 \leq n$. On the other hand, Shamir's scheme can only be defined for $n < \#\mathbb{K}$.

As an aside, it is worth mentioning that a more general notion, known as *arithmetic codex*, has been introduced in [5]. Multiplicative and strongly multiplicative LSSS are special cases of this notion, which also encompasses as particular cases the notion of bilinear multiplication algorithm from algebraic complexity. For reasons of space this notion is not discussed further here, see [5, 16] for more details.

In sections 2.2,2.3,2.4 some applications of multiplicative secret sharing schemes will be presented.

2.2 Robust secret sharing

The property of r-reconstruction of a secret sharing scheme ensures that the secret is determined by any r correct shares, but does not provide any guarantee if some of these shares are incorrect. A *robust* secret sharing scheme is one that allows to reconstruct the secret even if a few of the shares are incorrect (and we do not know which). The following result, which appeared in [18], shows that strongly multiplicative secret sharing schemes are robust.

THEOREM 2.6. *Let $\Sigma_\ell(C)$ be a t-strongly multiplicative LSSS over $\mathbb{K}$. For any $(\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n) \in \mathbb{K}^n$ there exists at most one vector $(s_1, s_2, \ldots, s_\ell) \in \mathbb{K}^\ell$ such that there is (at least) one word $(s_1, s_2, \ldots, s_\ell, a_1, a_2, \ldots, a_n) \in C$ with $d_H((a_1, a_2, \ldots, a_n), (\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n)) \leq t$. Furthermore, there exists a polynomial-time (in n) algorithm that, on input $(\tilde{a}_1, \tilde{a}_2, \ldots, \tilde{a}_n)$, either outputs $(s_1, s_2, \ldots, s_\ell)$ if this value exists, or $\perp$ otherwise.*

The theorem shows that if a LSSS is t-strongly multiplicative, the secret can be recovered efficiently from the set of all shares in the presence of at most t erroneous ones.

2.3 Secure multiparty computation

Secure computation is concerned with the following situation: n parties, each holding some private input x_i, want to correctly compute $f(x_1, x_2, \ldots, x_n)$ for some agreed upon function f in such a way that the intended output is the only new information released. This guarantee should be fulfilled even if a small number t of the players cheat. Collusions of malicious players are modelled as an external adversary who corrupts these players and collects all information seen by them. The adversary is passive if corrupted players follow the specified protocol and active if they can behave arbitrarily. For formal security definitions and more information about the topic, see [16].

For the case of a *computationally unbounded* adversary, Ben-Or et al. [2] and Chaum et al. [9] established seminal results, proving that secure computation of any function is possible as long as less than $n/2$ players are passively corrupted, or less than $n/3$ players are actively corrupted[3]. The results make implicit use of algebraic properties (linearity, multiplicativity) of the secret sharing scheme involved in their protocols, which is Shamir's scheme. The notions of multiplicativity were formally defined in [15], where it was shown that, in fact, secure multiparty computation protocol secure against a passive (resp. active) adversary corrupting t players can be constructed from any multiplicative secret sharing scheme with t-privacy (resp. t-strong multiplicative secret sharing scheme). The idea of the protocol is as follows. Suppose $\Sigma_1(C)$ is a LSSS over $\mathbb{K}$. Let $s, s' \in \mathbb{K}$ be two secrets shared with $\Sigma_1(C)$ and denote the sharings $[s] = (a_1, \ldots, a_n)$ and $[s'] = (a'_1, \ldots, a'_n)$. Now it is easy to see that $[s] + [s']$ is a sharing of $s + s'$. Indeed $(s + s', a_1 + a'_1, \ldots, a_n + a'_n)$ is in C,

[3]Here we suppose each pair of players is connected by a secure point-to-point channel, but we do not assume the existence of a broadcast channel.

and the distribution of this codeword is uniform conditioned to the secret being $s + s'$. Similarly $\lambda[s]$ is a sharing of λs for any $\lambda \in \mathbb{K}$. This implies that from sharings $[s^{(1)}], \ldots, [s^{(m)}]$, and for any *linear* function $f : \mathbb{K}^m \to \mathbb{K}$, a sharing $[f(s^{(1)}, \ldots, s^{(m)})]$ can be computed by applying f to the vector containing the i-th shares of $[s^{(1)}], \ldots, [s^{(m)}]$, for every $i = 1, \ldots, n$. These arguments yield a multiparty computation protocol to compute linear functions secure against a passive adversary: at the beginning of the protocol, each party shares his input, making sure the i-th player always receives the i-th share; then, each player simply computes f on his shares and broadcasts the result; the output can then be reconstructed from the broadcast values. If the adversary corrupts at most t parties and $\Sigma_1(C)$ has t-privacy, the protocol leaks no information other than the output of the computation.

Ideally, we would want to extend this idea to any function f. Since every function can be computed as an arithmetic circuit over a finite field (for example a Boolean circuit), it would be enough if a sharing $[ss']$ can be created from $[s]$ and $[s']$. However, if s and s' are secret shared as above, the vector $(ss', a_1a'_1, \ldots, a_na'_n)$ belongs to C^{*2}, but not necessarily to C. Further multiplications would yield vectors in higher powers of C. Hence, if d is the multiplicative depth of f as an arithmetic circuit, it would be needed that $\Sigma_1(C^{*d})$ has n-reconstruction.

A better alternative is to allow more interaction among the players. This only requires that $\Sigma_1(C)$ is multiplicative (i.e., that $\Sigma_1(C^{*2})$ has n-reconstruction). If this is the case, by definition there is a linear function ρ such that $ss' = \rho(a_1a'_1, \ldots, a_na'_n)$. The following protocol allows to create a sharing $[ss']$ from $[s]$ and $[s']$: Each player i shares the product $a_ia'_i$ of his shares. Now, players have shares in $[a_1a'_1], [a_2a'_2] \ldots, [a_na'_n]$ and they can locally compute $[ss'] = [\rho(a_1a'_1, \ldots, a_na'_n)]$ since ρ is linear. This can be shown to be secure against a passive adversary corrupting at most t parties if the scheme has t-privacy and it is enough to argue that any function can be securely computed by n players in this situation.

The case of an active adversary is considerably more difficult, because corrupt players may decide to deal inconsistent shares to the honest players or to share wrong values. The solution in [15] involves using *verifiable secret sharing*, in which the dealer sends, on top of the shares, additional information to each player for the purpose of verification. They show that, if the underlying LSSS is t-strongly multiplicative, this leads to a secure protocol to compute any function in the presence of an adversary corrupting at most t players.

Therefore, in order to tolerate an adversary who corrupts many parties, it is sufficient to use t-strong multiplicative secret sharing schemes for large t. In turn, by Proposition 2.5 these can be constructed from codes C such that both $C^\perp$ and C^{*2} have large minimum distance.

2.4 Two-party cryptography

In recent years, secure multi-party computation protocols have found unexpected applications in the area of two-party computation. This is in great part

due to the MPC-in-the-head technique, which was introduced in [24] for the purpose of zero knowledge proofs with high communication efficiency. In short, the idea is as follows. Alice simulates the execution of a multiparty computation protocol that will output 1 if and only if the statement she wants to prove is true. She keeps this execution in her head and only reveals the views of some subset, chosen by Bob, of the "players" of this virtual protocol. Bob then verifies the consistency of the views and that the output is indeed 1. The views revealed still give no information about the input of the protocol to Bob, but will make sure that with high probability Alice will be caught if she tries to cheat. The work of [24] has inspired subsequent work using similar ideas in a host of applications in areas such as multiparty computation with dishonest majority (including two party computation) [26], [20], OT combiners [22], OT from noisy channels [23], correlation extraction [25], zero knowledge proofs of algebraic relations [17] and UC homomorphic commitment schemes [19]. In these applications, the best results in terms of efficiency are attained by setting up multiparty computation protocols with a very large number of players, while the size of the field should be small. This has provided an additional motivation to the study of *asymptotics* (large number of players, large t, constant size field) of t-strongly multiplicative secret sharing schemes. Some results in this direction will be mentioned later (see Theorem 3.1).

2.5 Attacks to the McEliece cryptosystem

The McEliece cryptosystem is a public key cryptosystem whose security is based on the hardness of decoding a general linear code. Roughly speaking, the McEliece cryptosystem works along the following lines. The public key is a random generator matrix of a linear code chosen from certain family of linear codes where t errors can be decoded efficiently. The private key is an efficient decoding algorithm of this code. In order to encrypt a message, this is first encoded with the code and then a random error of weight t is added. The security relies on the fact that the public generator matrix should appear random and not reveal any algebraic structure of the code that may lead to an efficient decoding algorithm. Several families of codes have been proposed for its use in McEliece cryptosystem. To this day some of them are still considered secure. However, in other cases there have been attacks that allow to recover the secret key. Some of these attacks are based on the idea that one can distinguish generator matrices of codes that have certain algebraic structure from those of random codes, by observing how the dimension of the *square* of the code grows. Indeed, consider for example the case of Reed-Solomon codes where squaring basically doubles the dimension; on the other hand, for a random linear code C of dimension k, the dimension of C^{*2} is $\min\{n, \binom{k}{2}\}$ with high probability; see Section 3. Based on these ideas, key-recovery attacks for McEliece cryptosystems have been found for some variants based on subcodes of generalized Reed Solomon codes [34, 12], certain families of Goppa codes [14] and certain subcodes of algebraic geometric codes [13].

3 Results

In recent years powers of codes have been the subject of study of several papers. A nice exposition on the topic of powers (and more generally products) of codes, including many basic results can be found in the paper by H.Randriambololona [31]. For basic results on multiplicative secret sharing see [5] and [16]. Here the focus will be on a few results on asymptotics that are particularly interesting for the applications mentioned in this paper. First, we consider asymptotics of t-strongly multiplicative secret sharing schemes, where we fix a finite field $\mathbb{K}$ and consider families of secret sharing schemes over $\mathbb{K}$ with number of shares growing to infinity. The following result has been established in a series of papers [10, 3, 4].

THEOREM 3.1. *For every finite field $\mathbb{K}$, there is a family $\{\Sigma_{\ell_n}(C_n)\}_{n\in N}$ of t_n-strongly multiplicative secret sharing schemes with n shares over $\mathbb{K}$ where $N \subseteq \mathbb{N}$ is an infinite set, $t_n = \Omega(n)$ and $\ell_n = \Omega(n)$.*

These results are attained using algebraic geometric codes defined on towers of function fields with many rational places. The result was first established in [10] for large enough finite fields, and later the paper [3] extended it to all finite fields, by using concatenation of the codes in [10] over extension fields with a dedicated field descent map. Finally, [4] (see also [7]) applied considerably more involved algebraic geometric arguments to show that the construction [10] can be used directly over almost all fields. For more information about these results see the references mentioned above, as well as [5] and the forthcoming book [16]. On the other hand, limitations have also been shown [6]: the limit $\frac{t_n}{n-1}$ cannot approach asymptotically $\frac{1}{3}$ and hence we have to pay a price for asymptotics (when $n < \#\mathbb{K}$ this ratio is attained by Shamir's scheme and it is in fact optimal [15]). The results in [10] and [4] also show the existence of asymptotically good families of linear codes such that both their duals and squares are also asymptotically good[4] over all finite fields $\mathbb{K}$ with $\#\mathbb{K} = 8, 9$ or $\#\mathbb{K} \geq 16$. However, it is not known if one can extend this result to the remaining finite fields, as the concatenation in [3] does not preserve this property. Nevertheless, if we do not worry about the duals but only about the squares, then there is the following result [30].

THEOREM 3.2. *For every finite field $\mathbb{K}$, there is an asymptotically good family $\{(C_n)\}_{n\in N}$ of linear codes over $\mathbb{K}$ such that the family $\{(C_n^{*2})\}_{n\in N}$ is also asymptotically good.*

This construction also consists in a concatenation of a family of algebraic geometric codes, over a large enough finite field, with an appropriate field descent map. However, more elaborated arguments than in [3] are needed: the proofs require bounding the distance not only of the squares but also of higher powers of the initial algebraic geometric codes, and very recent results on asymptotically

[4] A family of codes such that their length grows to infinity is asymptotically good if both the dimension and minimum distance grow linearly with the length.

good towers of function fields [1] are essential for the argument to work. Upper bounds on the minimum distance of C^{*2} relative to the length and dimension of C have also been derived. See [29] and [31]. Finally, for every $m > 2$, it is not known if for all fields $\mathbb{K}$ (and in particular for the binary field) there exist good linear codes C over $\mathbb{K}$ with good powers C^{*m}.

All asymptotical constructions mentioned above rely on algebraic geometry codes over asymptotically good towers of function fields. Computing the generator matrices of these codes has a high complexity. It is then natural to wonder if there are more "elementary" constructions attaining the results in Theorems 3.1 and 3.2. Random linear codes may appear as a natural candidate, as it is well known that with high probability they are asymptotically good, in fact attaining the Gilbert-Varshamov bound [27]. However, this is not the case for the *squares* of random codes. The following result was shown in [8].

THEOREM 3.3. *Let $k \leq n \leq \frac{k(k+1)}{2}$. Write $t = \frac{k(k+1)}{2} - n$. Let C be a code chosen uniformly at random among all codes of length n and dimension k. Then*

$$\Pr\left[C^{*2} = \mathbb{K}^n\right] = 1 - 2^{-\Theta(k)} - 2^{-\Theta(t)}.$$

This means that if, in particular $k = \Theta(n)$, then C^{*2} will be the full space $\mathbb{K}^n$ with overwhelming probability and hence C and C^{*2} will not be good simultaneously. As an aside, the following complementary result was also shown in [8].

THEOREM 3.4. *Let $n \geq \frac{k(k+1)}{2}$. Write $s = n - \frac{k(k+1)}{2}$. Let C be a code chosen uniformly at random among all codes of length n and dimension k. Then*

$$\Pr\left[\dim C^{*2} = \frac{k(k+1)}{2}\right] = 1 - 2^{-\Theta(s)}.$$

It is therefore an important open question to determine whether elementary, perhaps probabilistic, constructions of codes can be found that attain the results in Theorems 3.1 and 3.2. This would have important consequences in the computational complexity of several of the applications presented here.

4 Acknowledgements

I acknowledge support from the Danish National Research Foundation and The National Science Foundation of China (under the grant 61361136003) for the Sino-Danish Center for the Theory of Interactive Computation and from the Center for Research in Foundations of Electronic Markets (CFEM), supported by the Danish Strategic Research Council. I would like to thank Alain Couvreur, Ronald Cramer, Ivan Damgård, Nico Döttling and Irene Giacomelli for comments and discussions regarding this paper.

References

[1] A. Bassa, P. Beelen, A. Garcia, H. Stichtenoth. Towers of function fields over non-prime finite fields. Preprint, 2012.

[2] M. Ben-Or, S. Goldwasser, A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. *Proc. of STOC 1988*, pp. 1–10. ACM Press, 1988.

[3] I. Cascudo, H. Chen, R. Cramer, C. Xing. Asymptotically Good Ideal Linear Secret Sharing with Strong Multiplication over *Any* Finite Field. *Proc. of 29th Annual IACR CRYPTO*, Springer Verlag LNCS, vol. 5677, pp. 466-486, 2009.

[4] I. Cascudo, R. Cramer, C. Xing. The Torsion-Limit for Algebraic Function Fields and Its Application to Arithmetic Secret Sharing. *Proc. of 31st Annual IACR CRYPTO*, Springer Verlag LNCS, vol. 6842, pp. 685-705, 2011.

[5] I. Cascudo, R. Cramer, C. Xing. The Arithmetic Codex. *Proceedings of IEEE Information Theory Workshop (ITW 2012)*, pp. 75 - 79, 2012.

[6] I. Cascudo, R. Cramer, C. Xing. Bounds on the Threshold Gap in Secret Sharing and its Applications. *IEEE Transactions on Information Theory*, 59(9):5600-5612, 2013.

[7] I. Cascudo, R. Cramer, C. Xing. Torsion Limits and Riemann-Roch Systems for Function Fields and Applications. *IEEE Transactions on Information Theory*, 60(7):3871-3888, 2014.

[8] I. Cascudo, R. Cramer, D. Mirandola and G. Zemor. Squares of Random Linear Codes. *IEEE Transactions on Information Theory*, 61(3):1159-1173, 2015.

[9] D. Chaum, C. Crépeau, I. Damgaard. Multi-party unconditionally secure protocols. *Proc. of STOC 1988*, pp. 11–19. ACM Press, 1988.

[10] H. Chen, R. Cramer. Algebraic Geometric Secret Sharing Schemes and Secure Multi-Party Computation over Small Fields. *Proc. of 26th Annual IACR CRYPTO*, Springer Verlag LNCS, vol. 4117, pp. 516-531, 2006.

[11] H. Chen, R. Cramer, S. Goldwasser, R. de Haan, V. Vaikuntanathan. Secure Computation from Random Error Correcting Codes. *Proc. of 27th Annual IACR EUROCRYPT*, Springer Verlag LNCS, vol. 4515, pp. 291-310, 2007.

[12] A. Couvreur, P. Gaborit, V. Gauthier, A. Otmani, J.-P. Tillich. Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes. *Designs Codes and Cryptography*, vol. 73, pp. 641-666, November 2014.

[13] A. Couvreur, I. Márquez–Corbella, R. Pellikaan, "Cryptanalysis of Public-Key Cryptosystems that use Subcodes of Algebraic Geometry Codes". Presented at the 4th International Castle Meeting on Coding Theory and Applications. http://arxiv.org/abs/1409.8220

[14] A. Couvreur, A. Otmani, J.-P. Tillich. Polynomial Time Attack on Wild McEliece Over Quadratic Extensions. *Proc. of 33rd Annual IACR EUROCRYPT*, Springer Verlag LNCS, vol. 8441, pp. 17-39, 2014.

[15] R. Cramer, I. Damgård, U. Maurer. General secure multi-party computation from any linear secret sharing scheme. *Proc. of 19th Annual IACR EUROCRYPT*, Springer Verlag LNCS, vol. 1807, pp. 316-334, 2000.

[16] R. Cramer, I. Damgård, J.B. Nielsen. *Secure Multiparty Computation and Secret Sharing - An Information Theoretic Approach*. Book, in preparation. Draft available at http://www.cs.au.dk/ jbn/mpc-book.pdf

[17] R. Cramer, I. Damgaard, V. Pastro. On the Amortized Complexity of Zero Knowledge Protocols for Multiplicative Relations. *Proc. 6th ICITS*, 2012.

[18] R. Cramer, V. Daza, I. Gracia, J. Jiménez Urroz, G. Leander, J. Martí-Farré, C. Padró. On codes, matroids and secure multi-party computation from linear secret sharing schemes. IEEE Transactions on Information Theory, 54(6):2644-2657, 2008. Earlier version: CRYPTO'05.

[19] I. Damgård, B. David, I. Giacomelli, J. B. Nielsen. Compact VSS and efficient homomorphic UC commitments. *Advances in Cryptology ASIACRYPT 2014*, volume 8874 of *Lecture Notes in Computer Science*, pages 213–232. Springer, 2014.

[20] I. Damgård, S. Zakarias. Multiparty Computation for Boolean Circuits with Constant Overhead in the Preprocessing Model. *Proc. of TCC 2013*: Vol. 7785 of Lecture Notes in Computer Science. Springer, pp. 621-641, 2013.

[21] A. Garcia and H. Stichtenoth. A tower of Artin-Schreier extensions of function fields attaining the Drinfeld-Vlăduţ bound. Invent. Math. 121, pp. 211-222, 1995.

[22] D. Harnik, Y. Ishai, E. Kushilevitz, J. Nielsen. OT-Combiners via Secure Computation. *Proc. of TCC 2008*, Lecture Notes in Computer Science, Volume 4948, pp. 393-411, 2008.

[23] Y. Ishai, E. Kushilevitz,R. Ostrovsky, M. Prabhakaran, A. Sahai, J. Wullschleger. Constant-Rate Oblivious Transfer from Noisy Channels. *Proc. of 31st Annual IACR CRYPTO*, Springer Verlag LNCS, vol. 6842, pp. 667-684, 2011.

[24] Y. Ishai, E. Kushilevitz, R. Ostrovsky, A. Sahai. Zero-knowledge from secure multiparty computation. *Proc. of 39th STOC*, San Diego, Ca., USA, pp. 21-30, 2007.

[25] Y. Ishai, E. Kushilevitz, R. Ostrovsky, A. Sahai. Extracting Correlations. *Proc. 50th IEEE FOCS*, pp. 261-270, 2009.

[26] Y. Ishai, M. Prabhakaran, A. Sahai. Founding Cryptography on Oblivious Transfer-Efficiently. *Proc. of 28th Annual IACR CRYPTO*, Springer Verlag LNCS, vol. 5157, pp. 572-591, 2008.

[27] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland Publishing Company, 2nd edition, 1978.

[28] J. Massey. Minimal Codewords and Secret Sharing. *Proc. of the 6th Joint Swedish-Russian International Workshop on Information Theory*, 1993.

[29] D. Mirandola. Schur products of linear codes: a study of parameters Master Thesis. Univ. Bordeaux 1 and Stellenbosch Univ., 2012.

[30] H. Randriambololona. Asymptotically good binary linear codes with asymptotically good self-intersection spans. *IEEE Transactions on Information Theory* 59(5): 3038-3045 (2013).

[31] H. Randriambololona. On products and powers of linear codes under componentwise multiplication To appear in *Contemporary Math.*, AMS, 2015.

[32] A. Shamir. How to share a secret. *Comm. of the ACM*, 22(11): 612-613, 1979.

[33] H. Stichtenoth. Algebraic function fields and codes. Springer Verlag, 1993. (New edition: 2009).

[34] C. Wieschebrink. Cryptanalysis of the Niederreiter Public Key Scheme Based on GRS Subcodes. *Proc. of PQCrypto*, Springer Verlag LNCS, vol. 6061, pp. 61-72, 2010.