Identification codes in cryptographic protocols

Julien Bringer*
*Morpho
Osny, France

Hervé Chabanne*†

Gérard Cohen †

Bruno Kindarji *†
†Institut Télécom
Télécom ParisTech
Paris, France

***Abstract*—Identification codes were introduced by Ahlswede and Dueck more than twenty years ago. There is today a lot of studies to identify objects such as contactless devices (for instance RFID tags) but, surprisingly, no one has considered the use of this kind of codes in the literature for that purpose until the recent work of Bringer *et al.* at Indocrypt'09. We here show how the security of these new identification protocols is related to some well-known problems in coding theory. We also extend the original proposal to a new problem.**

I. Introduction

Identification codes have been introduced by Ahlswede and Dueck in [1] to answer to a different problem than transmission codes. Transmission codes can correct errors which happen during a noisy emission of a message whereas identification codes enable to test whether a particular message was sent. To quote [2], transmission codes answer to the question "What message has been sent?" and identification codes to "Has message m been sent?".

Informally, an identification code is given by a set of probabilistic coding functions, along with (deterministic) decoding sets. On one hand, this has a negative impact as we have to take into account the probabilities of false-negative and false-positive identifications. On the other hand, this gives us two interesting properties:

- The identification of one element among n others is possible by conveying only $\log\log n$ bits. This means that the string we have to transmit can be very short.
- The probabilistic coding scheme increases a lot the job of the eavesdropper who would like to track a particular identity as the same identifying bit string is not used twice except with a small probability.

This second observation is the basis of [3] where the authors prove the security of their identification BCCK protocol in Vaudenay's model of privacy [4].

There have been different proposals for constructing identification codes [5], [6], [7]. The one of Moulin and Koetter [7] has, besides its simplicity of description, a particular interest as it enabled [3] to rely for their proof of security on a classical cryptographic assumption known as the Polynomial Reconstruction (PR) problem [8], [9], [10], [11]. Later on, the context of this proof has been reconsidered by [12] in an information theoretical point of view.

We here recall the definition of idenfication codes in Sec. II. A focus is made on the construction of Moulin and Koetter in Sec. II-B.

We describe the BCCK idenfication protocol in Sec. III and explain its links with the PR problem in Sec. IV.

We conclude in Sec. V with a way to extend the work of [3] to more difficult problems.

II. Identification Codes

A. General Definition

Let $\mathcal{X}, \mathcal{Y}$ be two alphabets, η a message length, and W^η a channel from $\mathcal{X}^\eta$ to $\mathcal{Y}^\eta$, defined as a conditional probability law: $W^\eta(y^\eta|x^\eta)$ is the probability to receive a message $y^\eta \in \mathcal{Y}^\eta$ given a transmitted message $x^\eta \in \mathcal{X}^\eta$. By extension, for a given subset $E \subset \mathcal{Y}^\eta$, $W^\eta(E|x^\eta)$ is the probability to receive a message belonging to E when x^η has been transmitted.

Definition 1 (Identification Code, [1]): A $(\eta, N, \lambda_1, \lambda_2)$-identification code from $\mathcal{X}$ to $\mathcal{Y}$ is given by a family $\{(Q(\cdot|i), \mathcal{D}_i)\}_i$ with $i \in \{1, \ldots, N\}$ where:

- $Q(\cdot|i)$ is a probability distribution over $\mathcal{X}^\eta$, that encodes i,
- $D_i \subset \mathcal{Y}^\eta$ is the **decoding set**,
- λ_1 and λ_2 are the first-kind and second-kind error rates, with

$$\lambda_1 \geq \sum_{x^\eta \in \mathcal{X}^\eta} Q(x^\eta|i) W^\eta(\overline{D_i}|x^\eta)$$

and

$$\lambda_2 \geq \sum_{x^\eta \in \mathcal{X}^\eta} Q(x^\eta|j) W^\eta(D_i|x^\eta)$$

(where $W^\eta(D_i|x^\eta)$ is the probability to be in the decoding set D_i given a transmitted message x^η and $W^\eta(\overline{D_i}|x^\eta)$ the probability to be outside the decoding set)

for all $i, j \in \{1, \ldots, N\}$ such that $i \neq j$.

Given $Q(\cdot|i)$, the **encoding set** of i is defined as the set of messages x^η for which $Q(x^\eta|i) > 0$, in other words, the set of messages likely to encode i.

The fundamental noisy-channel coding theorem [1, Theorem 1] states that given a channel W^η of (Shannon) capacity κ, it is possible to define identification codes of identification capacity $R_{id} = \frac{\eta}{\log\log N}$ asymptotically close to κ.

The proof of this result is based on the following generic construction: $A_1, \ldots, A_N \subset X^\eta$ are N subsets such that each A_i has the same cardinal n and each intersection $A_i \cap A_j$ for $i \neq j$ contains at most λn elements. Define the encoding distribution $Q(\cdot|i)$ as the uniform distribution over A_i; in the noiseless case (the channel W^η is the identity function) the

decoding sets are also the A_i's. Note that in that case the false-negative rate λ_1 is equal to 0 and the false-positive rate λ_2 is λ. [1] actually proves that such subsets do exist.

B. Moulin and Koetter Identification Codes Family

Though the fundamental theorem states that there exist families of identification codes of capacity-approaching rate, practical construction were still to be found. Among others, Moulin and Koetter proposed the following code, based on an Error-Correcting Code C of length n, size $N = |C|$ and minimum distance d over some alphabet.

For a word $c_i = (c_i^{(1)}, \dots c_i^{(n)}) \in C$, the corresponding set A_i is the collection of all $(u, c_i^{(u)})$, for $u \in \{1, \dots, n\}$. Note that we indeed have sets A_i of constant size n; moreover, the intersection of two different sets $A_i \cap A_j$ contains at most $n-d$ elements, which induces $\lambda_2 = \frac{n-d}{n} = 1 - \frac{d}{n}$.

The instantiation of this construction explicitly described in [7] uses Reed-Solomon codes. A Reed-Solomon code over a finite field $A = \mathbb{F}_q$, of length $n < q-1$, and dimension k, is the set of the evaluations of all polynomials $P \in \mathbb{F}_q[X]$ of degree less than $k-1$, over a subset $F \subset \mathbb{F}_q$ of size n ($F = \{\alpha_1, \dots, \alpha_n\}$). In other words, for each k-tuple $(x_0, \dots, x_{k-1}) \in \mathbb{F}_q^k$, the corresponding Reed-Solomon word is the n-tuple $(y_1, \dots, y_n)$ where $y_i = \sum_{j=0}^{k-1} x_j \alpha_i^j$. In the sequel, we identify a source word $(x_0, \dots, x_{k-1}) \in \mathbb{F}_q^k$ with the corresponding polynomial $P = \sum_{j=0}^{k-1} x_j X^j \in \mathbb{F}_q[X]$.

Definition 2 (Moulin-Koetter RS-Identification Codes): Let $\mathbb{F}_q$ be a finite field of size q, $k \leq n \leq q-1$ and an evaluation domain $F = \{\alpha_1, \dots, \alpha_n\} \in \mathbb{F}_q$. Set $A_P = \{(j, P(\alpha_j)) | \, j \in \{1, \dots, n\}\}$ for P any polynomial on $\mathbb{F}_q$ of degree at most $k-1$.

The Moulin-Koetter RS-Identification Codes are defined by the family of encoding and decoding sets $\{(A_P, A_P)\}_{P \in \mathbb{F}_q[X], \deg P < k}$.

This leads to a $(\log_2 n + \log_2 q, q^k, 0, \frac{k-1}{n})$-identification code from $\{0,1\}$ to $\{0,1\}$.

Using a Reed-Solomon code of dimension k, this gives $\lambda_2 = \frac{k-1}{n}$ since $d = n - k + 1$.

III. Description of the BCCK Identification Protocol

This protocol takes place between a verifier and multiple contactless devices (CLD). The goal of the identifier is to identify the CLDs that are present in communication range by sending them a message.

Each CLD stores two different random polynomials of degree at most $k-1$. For instance, let $p_l, p'_l \in \mathbb{F}_q[X]$ be associated with CLD_l. The verifier's database contains these 2 polynomials for all CLD.

More precisely, suppose that a set of $M < q^k$ devices is initialized. The memory of these devices is then filled with a set of $p_l(\alpha_j)$, for $\alpha_j \in F$, with F a public subset of $\mathbb{F}_q$, *i.e.* the devices contain the evaluation of p_l over a subset of $\mathbb{F}_q$. The verifier is then given the polynomial p_l.

To identify CLD_l , the verifier broadcasts $(ACK, j, p_l(\alpha_j))$ over the wireless channel, where ACK is a session number issued to help the verifier to sort out the messages when several such transmissions are emitted. All the present CLD's have to check whether there is a match with the stored polynomial. For instance, CLD with polynomials p, p' in its memory, evaluates whether $p_l(\alpha_j) = p(\alpha_j)$. In this case, it responds with $(ACK, p'(\alpha_j))$. Otherwise, it remains silent.

CLD identifiers p, p'	parameters $\mathbb{F}_q, (\alpha_1, \dots, \alpha_n)$	Verifier (l, p_l, p'_l)
	$\xleftarrow{(ACK, j, a = p_l(\alpha_j))}$	Pick j
If $p(\alpha_j) = a$	$\xrightarrow{(ACK, b = p'(\alpha_j))}$	$p'_l(\alpha_j) \stackrel{?}{=} b$

Figure 1. CLD identification via Moulin-Koetter identification codes

At the end, CLD_l is identified by the verifier if its answer corresponds to $p'_l(\alpha_j)$.

To thwart replay attacks , a flag bit is added in each CLD to tell whether α_j was already used or not. Of course, this flag bit has to be switched when $p_l(\alpha_j)$ is received.

Practical Parameters

An advantage of the BCCK protocol is that – even for a noisy channel – the CLD's will only have to compute the distance between the received message and the stored one to check for a given equality $p_l(\alpha_j) = p(\alpha_j)$. If the channel is supposed error-less – in other words, if a transmission code is applied – then the equality should be true after decoding; if not, adding redundancy to the messages enables to reduce the equality check to a distance bounding test.

Consider a memory size of $2^{18} = 256$k bits, with $q = 2^{64}$, CLD's are then able to store $2^{18-6} = 2^{12} = 4096$ 64-bit fields elements. As each interrogation consumes 2 memory elements, this enables 2048 interrogations of a CLD by the verifier, and implies that the length of the corresponding code is $n = 2^{11}$, which is consistent with $n \leq q-1$.

IV. Security Results

In this section we sum up the security properties of the protocol which are proved in [3]. The protocol's aim is identification of objects via their interrogation followed by an authentication step. The security is analysed in the model [4] which includes security of the authentication phase and privacy of the objects.

A. Requirements

Basically the communications, operations and more generally the actions that an adversary can take, request or interfere with are formalized by oracles. For instance, an adversary can send a message to a device to receive the corresponding answer (if any). He can also ask the verifier to launch a new protocol instance. The adversary can even handle several protocol instances in parallel to try to learn information, for instance by mixing some messages. Vaudenay's model also

introduces the corruption of a device to learn its state and the internal secret. Here, secrets are not correlated between devices so this operation is not really a risk against security of the protocol (see the list of requirements below).

The security or privacy properties related to this model and the protocol are the following:

- **Correctness**: The identification of a legitimate device should fail only with a negligible probability.
- **Resistance against impersonation of devices**: Only legitimate devices should be able to be authenticated. The adversary is active, *i.e.* he may take the place of verifiers and devices during the communications, he can eavesdrop and modify the messages. At the end an adversary should not be identified by the verifier as a legitimate device, except with a negligible probability. One specific constraint is that the adversary cannot replay a past protocol instance between a legitimate device and the verifier.
- **Replay attacks**: A specific impersonation risk not handled by the previous requirement is when a legitimate verifier broadcasts twice the same message during two different protocol instances. Then after eavesdropping of the first instance, impersonation is easy. This should not happen, except with a negligible probability.
- **Resistance against impersonation of the verifier**: as the protocol is based on the idea of interrogation of devices, we are concerned also with the situation where an adversary tries to be recognized as a legitimate verifier. If this would be possible, then tracking of a device would become feasible. This is related to the notion of mutual authentication.
- **Privacy**: In [4], privacy is defined as the non-ability for any adversary to distinguish a simulated system from the actual one. The simulated system runs thanks to simulated oracles which are aware neither of the secret parameters nor of the device's secret. This definition of privacy is more general than anonymity and untraceability. In the sequel, the adversary against privacy is not allowed to corrupt the devices (this corresponds to a weak adversary in [4]).

The protocol relies on the transmission of evaluations of polynomials. The security and privacy are then depending on the ability of an adversary to recover a polynomial or to detect links between values. If the verifier would communicate with only one device this would be straightforward. Nevertheless, the protocol is designed to achieve good properties when a verifier interrogates many different devices among a cloud of devices: the messages are actually mixed between values of uncorrelated polynomials. The security and privacy are related to the problem of polynomial reconstruction in presence of noise.

B. Assumptions

1) Hardness of the Polynomial Reconstruction Problem: The definition of the basic Polynomial Reconstruction (PR) problem follows. The problem is now well known and has been suggested for various cryptographic schemes [8], [9], [10], [11].

Definition 3: Given n, k, t such that $1 \leq t \leq n$, $k \leq n$ and given $z, y \in \mathbb{F}_q^n$, with $z_i \neq z_j$ for $i \neq j$,

- output all (p, I) where $p \in \mathbb{F}_q[X]$, $\deg(P) < k$, $|I| \geq t$, and for all $i \in I$, $p(z_i) = y_i$.

We denote $PR^z_{n,k,t}$ such an instance of the PR problem.

The PR problem is unconditionally secure when $t < k$, and is easy to solve when $t \geq \sqrt{kn}$, as list decoding of Reed-Solomon codes via the Guruswami-Sudan algorithm [13] is possible.

If the number t of noiseless components is in the range $\{k, \ldots, \lfloor\sqrt{kn}\rfloor\}$, then it is more difficult to determine whether the PR problem is hard or not. [12] takes interest in this issue, and considers the information-theoretic side of this problem: given vectors n, k, t, z, v as in Definition 3, *how many* polynomials $p \in \mathbb{F}_q[X]$ are there of degree less than k that interpolate y on z ? In other words, what is the size of the list after the list-decoding of y, allowing at most $n - t$ differences between elements of the list and y? If the list is of exponential size, then the PR instance is *necessary* difficult. [12] shows that the size of the list is linked to the Maximum-Likelihood (ML) threshold of the Reed-Solomon code, which can be interpreted as the number of coordinates required to be exact in y in order for the ML decoder to output the original codeword with a large probability.

[12] also derives an explicit formula to approximate the threshold, and it appears that for a Maximum-Distance Separable code, it is very close to the lower-bound k. That means that there might exist polynomial-time algorithms that can beat the threshold of the Guruswami-Sudan list-decoding algorithm. However, finding such an algorithm is reputed to be hard, and as of today, it is safe to assume that when $t < \sqrt{kn}$, solving the PR problem is computationally hard.

2) The Decisional Reconstruction Problem: The Decisional PR problem (DPR) [8] consists in deciding if a given $i \in \{1, \ldots, n\}$ is in I for an instance y of $PR^z_{n,k,t}$ which admits at least one solution (p, I). Assuming that DPR is hard, [8] shows that PR instances do not leak any partial information on the polynomials.

In the security analysis, the PR and DPR problems are assumed to be hard even in the case where the noise is generated by the other transmissions. In this case, the noise is not random as the received elements are evaluations of a few different polynomials. [14] explained the link between polynomial reconstruction in the mixture model [15] and Reed-Solomon list decoding, and it is not easier to reconstruct a polynomial polluted with a structured noise as input than to reconstruct a polynomial with random noise.

Finally the last problem related to the protocol is the list recovery problem [16]. In fact, when a verifier interrogates many devices, this leads to several mixed PR instances with many transmitted values for the same index i. The list recovery problem is to retrieve the solutions to all underlying PR instances. This is equivalent to solving one longer PR instance and thus leads to a similar bound. Let l be the maximum

number of collisions per position i, the list recovery problem of Reed-Solomon codes is assumed to be hard when $t < \sqrt{nkl}$.

C. Security against Eavesdropping

Against an eavesdropper, who is by definition a passive adversary, the security relies on the impossibility to reconstruct a polynomial associated to a device and the privacy relies on the impossibility to distinguish transmitted values. In both cases, the only information available to an adversary are the transmitted messages.

Let M be the number of devices which are queried by one verifier during a given period and let T be the number of eavesdropped interrogations. Assume that the devices are interrogated almost uniformly by the verifier (remember also that there are false-positives with an identification code, so the verifier may address several devices simultaneously and will in that case receive as many replies). If T/M is small compared to $\sqrt{kn}$ then we know that there is almost surely no solvable PR instance. Otherwise, the adversary can try to find one solvable PR instance among all the possible ones (*i.e.* he has to deal with collisions by choosing for each index i one message among the different messages eavesdropped). In [3], it is shown that the probability to find a good instance becomes negligible for $e\sqrt{\frac{n}{k}} \leq M \leq \sqrt{q}$, where e is the Euler's number (exponential of 1). The other strategy for the adversary is to wait for a sufficient number of interrogations so that the list recovery problem becomes tractable (*i.e.* if T/M is greater then $\sqrt{nkl}$ with $l = T/n$ the approximate number of collisions per index). This gives the following result (the indistinguishability property is deduced thanks to the decisional version of PR).

Proposition 1: Let $e\sqrt{\frac{n}{k}} \leq M \leq \sqrt{q}$ and $T < M^2k$. A passive adversary that eavesdrops at most T requests

- cannot reconstruct one polynomial associated to a device, except with a negligible probability;
- cannot determine whether two non-identical interrogation requests correspond to the same device except with a negligible probability.

Note that identical interrogation requests happen with probability $1/\sqrt{n}$. In the sequel we assume once and for all that $e\sqrt{\frac{n}{k}} \leq M \leq \sqrt{q}$ and $T < M^2k$.

D. Security against Impersonation

In the protocol, a device answers to a broadcast message only if the message is related to its internal polynomial, *i.e.* if the verifier is believed to be genuine.

As eavesdropping does not give any advantage to the adversary, he may try to emit random values. With respect to a cloud of M devices, the probability when emitting a random value to interrogate correctly one of them is $\frac{M}{q}$. However the security against impersonation of the verifier is in fact important only with respect to a pre-fixed device:

Proposition 2: An adversary cannot impersonate a verifier to interrogate one given device, without replaying an eavesdropped transcript, except with probability $\frac{1}{q}$.

Concerning impersonation of a device, we know that eavesdropping would not give any useful information. As targeting a specific device via interrogations is not possible except with probability $\frac{1}{q}$ (thanks to the previous result), it is not possible to isolate the answers of one device. Thus, guessing the device answer is not possible except with probability $\frac{1}{q}$.

Proposition 3: The protocol is resilient to device impersonation attacks.

E. Privacy

Remember that privacy is ensured when simulations are indistinguishable from the real system. The simulation is in fact easy by sending random values for the verifier and by answering randomly to one request over M for a device. This is not distinguishable due to [8] (PR instances are not distinguishable from random sequences) and the previous results (eavesdropping does not give any information and impersonation of a verifier is not feasible so it is not possible to determine whether a device reacts when it should do so).

Proposition 4: The protocol ensures the privacy of the devices when the number M of queried devices during the same period satisfies $e\sqrt{\frac{n}{k}} \leq M \leq \sqrt{q}$ and when the number of eavesdropped interrogation requests is at most $T < M^2k$.

The proofs of Propositions 1, 2, 3 and 4 are given in [3].

Note that replay attacks are prevented in the protocol only for privacy concern: When a device is isolated, the fact that it does not accept to reply to a replayed message would lead to the possibility for tracking. However, in the context of interrogation of devices among a cloud, this is not an issue any more. Moreover if a verifier's message could be replayed then tracking a device would be easy as the device would answer with the same message. For security, this is not a risk as this does not give additional information.

With the chosen parameters, we suggest the use of polynomials of degree $k - 1 = 255$. This permits to define $q^k = 2^{64\times256}$ possible polynomials; the number M of devices needed in the cloud around a verifier has then to be greater than $e \times \sqrt{\frac{n}{k}}$, *i.e.* at least 8. With $M = 256$, this leads to the restriction $T < 2^{24}$, which is automatically satisfied here as $T \leq Mn = 2^{19}$.

V. Hidden Identification Codes

We now want to introduce a second line of defence to the BCCK identification protocol. We here consider the problem known as the Code Reverse Engineering (CRE) problem [17], [18], [19], [20]. This problem can be stated as follows: the adversary acting as an eavesdropper does not know anything on the characteristics of the identification code which is used during the BCCK protocol; he thus may want to recover the code. To the best of our knowledge, this CRE problem has never been solved for identification codes. At first thought, this context does not seem favourable to the adversary. We thus suggest to strengthen the previous work by restricting the information available on the identification codes.

As a practical example of such a construction, we use the Moulin-Koetter construction of Sec. II-B, with an error-correcting code C defined by its generating matrix $G \in$

$\mathbb{F}_q^{n\times k} = (G_1, \ldots, G_n)$. The encoding (and decoding) sets of the induced identification code are the $E_i = \{(j; G_j{}^t x_i), 1 \leq j \leq n\}$ where x_i is the ith vector of $\mathbb{F}_q^k$. If the generating matrix is kept secret, and known only by the verifier, then in order to breach privacy or security, an attacker should be able to reverse-engineer parts of the matrix G.

REFERENCES

[1] R. Ahlswede and G. Dueck, "Identification via channels," *IEEE Transactions on Information Theory*, vol. 35, no. 1, pp. 15–29, Jan 1989.

[2] K. Eswaran, "Identification via channels and constant-weight codes," http://www.eecs.berkeley.edu/~ananth/229BSpr05/Reports/KrishEswaran.pdf.

[3] J. Bringer, H. Chabanne, G. D. Cohen, and B. Kindarji, "Private interrogation of devices via identification codes," in *INDOCRYPT*, ser. Lecture Notes in Computer Science, B. K. Roy and N. Sendrier, Eds., vol. 5922. Springer, 2009, pp. 272–289.

[4] S. Vaudenay, "On privacy models for RFID," in *ASIACRYPT*, ser. Lecture Notes in Computer Science, K. Kurosawa, Ed., vol. 4833. Springer, 2007, pp. 68–87.

[5] K. Kurosawa and T. Yoshida, "Strongly universal hashing and identification codes via channels," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 2091–2095, Sep 1999.

[6] S. Verdu and V. Wei, "Explicit construction of optimal constant-weight codes for identification via channels," *IEEE Transactions on Information Theory*, vol. 39, no. 1, pp. 30–36, Jan 1993.

[7] P. Moulin and R. Koetter, "A framework for the design of good watermark identification codes," in *SPIE*, E. J. D. III and P. W. Wong, Eds., vol. 6072, no. 1. SPIE, 2006, p. 60721H. [Online]. Available: http://link.aip.org/link/?PSI/6072/60721H/1

[8] A. Kiayias and M. Yung, "Cryptographic hardness based on the decoding of Reed-Solomon codes," *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2752–2769, June 2008.

[9] ——, "Cryptographic hardness based on the decoding of Reed-Solomon codes with applications," in *Electronic Colloquium on Computational Complexity (ECCC)*, no. 017, 2002.

[10] ——, "Cryptographic hardness based on the decoding of Reed-Solomon codes," in *ICALP*, ser. Lecture Notes in Computer Science, P. Widmayer, F. T. Ruiz, R. M. Bueno, M. Hennessy, S. Eidenbenz, and R. Conejo, Eds., vol. 2380. Springer, 2002, pp. 232–243.

[11] ——, "Polynomial reconstruction based cryptography," in *Selected Areas in Cryptography*, ser. Lecture Notes in Computer Science, S. Vaudenay and A. M. Youssef, Eds., vol. 2259. Springer, 2001, pp. 129–133.

[12] B. Kindarji, G. Cohen, and H. Chabanne, "On the Threshold of Maximum-Distance Separable Codes," in *IEEE International Symposium on Information Theory, ISIT*, 2010.

[13] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometry codes," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 1757–1767, 1999.

[14] ——, "Reflections on "improved decoding of Reed-Solomon and algebraic-geometric codes"," 2002.

[15] S. Ar, R. J. Lipton, R. Rubinfeld, and M. Sudan, "Reconstructing algebraic functions from mixed data," *SIAM J. Comput.*, vol. 28, no. 2, pp. 487–510, 1998.

[16] A. Rudra, "List decoding and property testing of error correcting codes," Ph.D. dissertation, University of Washington, 2007.

[17] A. Valembois, "Detection and recognition of a binary linear code," *Discrete Applied Mathematics*, vol. 111, no. 1-2, pp. 199–218, 2001.

[18] M. Cluzeau, "Block code reconstruction using iterative decoding techniques," in *IEEE International Symposium on Information Theory, ISIT*, 9-14 2006, pp. 2269 – 2273.

[19] C. Chabot, "Recognition of a code in a noisy environment," in *IEEE International Symposium on Information Theory, ISIT*, 24-29 2007, pp. 2211 – 2215.

[20] M. Cluzeau and J. Tillich, "On the code reverse engineering problem," in *IEEE International Symposium on Information Theory, ISIT*, 6-11 2008, pp. 634 – 638.