On the Threshold of Maximum-Distance Separable Codes

Bruno Kindarji*† Gérard Cohen† Hervé Chabanne*†

* Sagem Sécurité
Osny, France

†Institut Télécom
Télécom ParisTech
Paris, France

***Abstract*—Starting from a practical use of Reed-Solomon codes in a cryptographic scheme published in Indocrypt'09, this paper deals with the threshold of linear q-ary error-correcting codes. The security of this scheme is based on the intractability of polynomial reconstruction when there is too much noise in the vector. Our approach switches from this paradigm to an Information Theoretical point of view: is there a class of elements that are so far away from the code that the list size is always superpolynomial? Or, dually speaking, is Maximum-Likelihood decoding almost surely impossible?**

We relate this issue to the decoding threshold of a code, and show that when the minimal distance of the code is high enough, the threshold effect is very sharp. In a second part, we explicit lower-bounds on the threshold of Maximum-Distance Separable codes such as Reed-Solomon codes, and compute the threshold for the toy example that motivates this study.

I. Introduction

In [1], Bringer *et al.* proposed a low-cost mutual authentication protocol, that uses a Reed-Solomon code structure. This protocol is pretty simple: Bob owns two secret polynomials P_b, P'_b of degree less than k known only by Alice; to authenticate herself to Bob, Alice proves the knowledge of P_b by sending $\langle i, P_b(\alpha_i)\rangle$ where α_i is the i-th element of a $\mathbb{F}_q$. Bob proves his identity by replying with $\langle P'_b(\alpha_i)\rangle$. This protocol (illustrated in Figure 1) is made such that if Alice speaks to many people, it is hard to trace Bob out of all the conversations, and it is hard to impersonate Alice (or Bob).

Alice		Bob
Pick j	$\xrightarrow{\langle j, a=P_b(\alpha_j)\rangle}$	$a \overset{?}{==} P_b(\alpha_j)$
$P'_b(\alpha_j) \overset{?}{==} b$	$\xleftarrow{\langle b=P'_b(\alpha_j)\rangle}$	

Figure 1. Low-cost Mutual Authentication Protocol [1]

The security of the protocol (can an impostor interrogate Bob?) is based on an algorithmic assumption, saying that the polynomial reconstruction problem is hard for the vectors of $\mathbb{F}_q^n$ that are far enough from the code. So does the privacy (can an eavesdropper trace Bob's communications?). The best known algorithms that solve polynomial reconstruction are those of Guruswami-Sudan [2] and, on a related problem, Guruswami-Rudra [3], which can basically reconstruct a polynomial given $\sqrt{kn}$ correct values.

This shows how the Polynomial Reconstruction problem is of interest both in cryptography and in coding theory. [4] proposes more primitives whose security depends on the hardness of this problem. The same reference assumes that the problem is difficult as soon as there are less than $\sqrt{kn}$ correct values (hardness of the polynomial reconstruction problem assumption). The security result that can be deduced from this statement are based on an algorithmic hypothesis, which is somehow unsatisfying, for it should be possible to exhibit better decoding algorithms. We therefore take interest in the information-theoretic aspect of polynomial reconstruction.

The solution of the problem raised by [1] is to look at the output of a list-decoder centred around the received values, and to output the possible polynomials as candidate values for P_b or P'_b. Our approach consists in looking at a usually ignored side of list-decoding. For a certain class of words x that are far enough from the code, we look at the radii r such that list-decoding x with radius r provides a list that is always lower-bounded by a large enough number. This differs from the literature concerning list-decoding, which usually looks for radii for which the size is always upper-bounded by a maximum list size, or tries to exhibit a counter-example.

The "large enough" list size can be obtained easily by imposing that Maximum-Likelihood Decoding to be most improbable. For that, we focus on the all-or-nothing behaviour of the ML decoder. Inspired by percolation theory [5], and code-applied graph theory [6], we will show how it is possible to conservatively estimate, before, after, and around a threshold, the all-or-nothing probability of ML decoding.

Notations

For a n-dimensional space H, the Hamming distance d over H is the number of differing coordinates between two vectors $x, y \in H$, *i.e.* $d(x,y) = |i \in \{1,\ldots,n\} : x_i \neq y_i|$. The weight of $x \in H$ is the number of non-zero coordinates $w(x) = d(x,0)$, and its support is the set of all its non-zero coordinates: $supp(x) = \{i \in \{1,\ldots,n\} : x_i \neq 0\}$ (in other words, $w(x) = |supp(x)|$.

The Hamming ball of radius r centred around $x \in H$ is the set of all vectors at a distance to x less than r, and is noted $B(x,r)$. The volume of such a ball is independent of x, and is noted $V(r) = \sum_{t=0}^{r} \binom{n}{t}(q-1)^t$.

For a subset $U \subset H$, $\overline{U}$ is its complementary $\overline{U} = \{x \in H : x \notin U\}$.

II. The Threshold of a Code

The existence of a threshold is motivated by the classical question of percolation : given a graph, with a source, and a sink, and given the probability p for a “wet” node of the graph to “wet” an adjacent node, *what is the probability for the source to wet the sink*? It appears that this probability has a threshold effect; in other words, there exists a limit probability p_c such that, if $p > p_c$, then the sink is almost surely wet, and if $p < p_c$, then the sink is almost never wet. The threshold effect is illustrated in Fig. 2.

This question can be transposed into the probability of error-correcting a code. Given a proportion of errors p, with a decoding algorithm, what is the probability of correctly recovering the sent codeword? It was shown in [7] that for every binary code, and every decoding algorithm, this probability also follows a threshold.

In this paper, we show that this property also applies to q-ary codes. In the following part, we show that the threshold behaviour that was seen on binary codes can be obtained again.

A. The Margulis-Russo Identity

The technique used to derive threshold effects in discrete spaces is to integrate an isoperimetric inequality; for that, the Margulis-Russo identity is required.

Let $H = \{0,1\}^n$ be the Hamming space; the Hamming distance $d(x, y)$ provides the number of different coordinates between vectors x and y. Consider the measure $\mu_p : H \to [0, 1]$ defined by $\mu_p(x) = p^{w(x)}(1 - p)^{n-w(x)}$.

The number of limit-vectors of a subset $U \subset H$ is a function defined as

$$h_U(x) = |B(x, 1) \cap \overline{U}| \text{ for } x \in U. \qquad (1)$$

For $U \subset H$ such that U is increasing (*i.e.* if $x \in U$, and $y \geq x$, then $y \in U$ with $\geq$ defined component-wise), Margulis and Russo showed :

$$\frac{d\mu_p(U)}{dp} = \frac{1}{p}\int_U h_U(x)d\mu_p(x)$$

Let $q \in \mathbb{N}, q > 2$. This section shows that this equality is also true in $H_q = \{0, ...q - 1\}^n$.

We redefine the measure function $\mu_p(x)$ over H_q by $\mu_p(x) = \left(\frac{p}{q-1}\right)^{w(x)}(1 - p)^{n-w(x)}$. This definition is consistent with a measure, as $\mu_p(H_n) = \sum_{x\in H_q}\mu_p(x) = 1$.

Note the inclusion $\subset$ to be the relation between a set and a (general) subset (*i.e.* for all X, $X \subset X$). The support inclusion generalises the component-wise $\leq$ that was used in the binary case.

Lemma 1 (Margulis-Russo Identity over q-ary alphabets): Let U be an increasing subset of H_q, *i.e.* such that if $y \in U$, for all $x \in H_q$ such that $supp(y) \subset supp(x)$, then $x \in U$. Then

$$\frac{d\mu_p(U)}{dp} = \frac{1}{p}\int_U h_U(x)d\mu_p(x)$$

where h_U is defined by (1).

Proof: The proof of this lemma is an adaptation of Margulis’ proof in [8]. For this, we use the notation:

- $[U, V] = |\{x, y\} \in U \times V : d(x, y) = 1|$ where $U, V \subset H_q$, is the number of links from U to V
- for $k \in \{0, \dots, n\}$, $Z_k = \{x \in H_q : w(x) = k\}$,
- for $U \subset H_q$, $U_k = U \cap Z_k$ (U is the union of the U_k);
- $D_k = \sum_{x\in U_k} h_U(x)$ is the number of limit-vectors next to elements of weight k.

Trivially, $D_k = [U_k, Z_{k+1} - U_{k+1}] + [U_k, Z_{k-1} - U_{k-1}] + [U_k, Z_k - U_k]$. We now note that :

- $[U_k, Z_{k-1}] = |U_k| \cdot k$, as to go from U_k to Z_{k-1}, the only way (in one move) is to put one coordinate to 0;
- $[U_k, Z_{k+1}] = |U_k| \cdot (n-k)(q-1)$ with the same reasoning;
- $[U_k, Z_k - U_k] = [U_k, Z_{k+1} - U_{k+1}] = 0$ as U is increasing.
- Combining these equalities, we get $[U_k, U_{k+1}] = |U_k|(n - k)(q - 1)$;
- $[U_k, Z_k] = 0$ as it is necessary to switch a non-zero coordinate to 0 and a zero to $\{1, ...q - 1\}$.

Finally $D_k = [U_k, Z_{k-1}] - [U_k, U_{k-1}] = k|U_k| - (n - k + 1)(q - 1)|U_{k-1}|$ for $k > 0$ and $D_0 = 0$ (or $U = H_q$).

Back to the identity desired, we observe that

$$\begin{aligned}\int_U h_U(x)d\mu_p(x) &= \sum_{k=0}^{n}\sum_{x\in U_k} h_U(x)(\frac{p}{q-1})^k(1-p)^{n-k} \\ &= \sum_{k=0}^{n} D_k\left(\frac{p}{q-1}\right)^k(1-p)^{n-k}\end{aligned}$$

$$= \sum_{k=1}^{n}(k|U_k| - (n - k + 1)(q - 1)|U_{k-1}|) \cdot \left(\frac{p}{q-1}\right)^k(1-p)^{n-k}$$

$$= \textstyle\sum_{k=0}^{n}|U_k|(k - p\frac{n-k}{1-p})\left(\frac{p}{q-1}\right)^k(1-p)^{n-k}$$

on the other hand,

$$\frac{d\mu_p(U)}{dp} = \textstyle\sum_{k=0}^{n}|U_k|\frac{d}{dp}\left(\left(\frac{p}{q-1}\right)^k(1-p)^{n-k}\right)$$

$$= \textstyle\sum_{k=0}^{n}|U_k|\left(\frac{p}{q-1}\right)^k(1-p)^{n-k}\left(\frac{k}{p} + \frac{-(n-k)}{1-p}\right)$$

Hence the identity. ■

This lemma shows that the Margulis-Russo identity is also true on $\{0...(q - 1)\}^n$; it was the keystone of the reasoning done in [6] to show an explicit form of the threshold behaviour of Maximum-Likelihood Error Correction.

B. A Threshold for Error-Decoding q-ary codes

In the following, we use $\varphi(t) = \frac{1}{\sqrt{2\pi}}e^{-\frac{t^2}{2}}$ the normal distribution, $\Phi(x) = \int_{-\infty}^{x}\varphi(t)dt$ the accumulate normal function, and $\Psi(x) = \varphi(\Phi^{-1}(x))$ (so that $\forall x, \Psi(x) \cdot \Phi'^{-1}(x) = 1$).

A monotone property is a set $U \subset H_q$ such that U is increasing, or $\overline{U}$ is increasing.

Theorem 1: Let U be a monotone property of H_q. Suppose that $\exists\Delta \in \mathbb{N}^\star : \forall x \in U, h_U(x) = 0$ or $h_U(x) \geq \Delta$.

Let $\theta \in [0,1]$ be (the unique real) such that $\mu_\theta(U) = \frac{1}{2}$. Let $g_\theta(p) = \Phi\left(\sqrt{2\Delta}(\sqrt{-\ln\theta} - \sqrt{-\ln p})\right)$.

Then the measure of U, $\mu_p(U)$ is bounded by :

$$\begin{array}{lcll} \mu_p(U) & \leq & g_\theta(p) & \text{for } p \in (0;\theta] \\ \mu_p(U) & \geq & g_\theta(p) & \text{for } p \in [\theta;1) \end{array}$$

Sketch of Proof

The proof is exactly the same as the one from [6]. The whole idea is to derive the upper-range:

$$\int_U \sqrt{h_U} d\mu_p \geq \sqrt{2\ln\frac{1}{p}}\Psi(\mu_p(U))$$

The integration of this equation, together with the Margulis-Russo lemma, gives the result. ■

To conclude this part, we remark that the non-decoding region of a given point, for a q-ary code, is an increasing region of $\mathbb{F}_q^n$. For linear codes, this non-decoding region can always be translated to that of 0 without loss of generality; let $U_0 = \{x \in \mathbb{F}_q^n \text{ s.t. } \exists c \in C, c \neq 0 : d(x,c) \leq d(x,0)\}$. The probability of error decoding of C is then $\mu_p(U_0)$.

For $x \in U_0$, we show that either $h_{U_0}(x) = 0$, or $h_{U_0}(x) \geq \frac{d}{2}$, where d is the minimal distance of C.

Indeed, if $h_{U_0}(x) > 0$, then there exists $c \in C, c \neq 0$ such that $d(x,c) \leq d(x,0)$, and $x_1 \in \overline{U_0}$ at Hamming distance 1 from x. The monotonic property of A_0 provides $|w(x_1) - w(x)| = 1$, and as x is further from 0 than x_1, $w(x_1) = w(x) - 1$. Then all the vectors obtained by replacing one of the coordinates of x by 0 are out of U_0; in particular, $h_{U_0}(x) \geq w(x)$. Let $d_c = w(c) \geq d$ be the weight of c; as x is nearer to c than to 0, $w(x) \geq \frac{d_c}{2}$. Thus the previous assertion.

Combining the previous results, we just showed that for any q-ary code, the probability of error is, as for binary codes, bounded by a threshold function. This can be expressed by the following theorem, which has the same form as the one showed in [6]:

Theorem 2: Let C be a code of any length, and of minimal distance d. Over the q-ary symmetric channel, with transition probability p, the probability of decoding error $P_e(p)$ associated with C is such that there exists a unique $p_c \in (0;1)$ such that $P_e(p_c) = \frac{1}{2}$, and P_e is bounded by:

$$P_e(p) \lesseqqgtr 1 - \Phi(\sqrt{d}(\sqrt{-\ln(1-p_c)} - \sqrt{-\ln(1-p)}))$$

The upper-bound ($\leq$) is true when $p \in (0;p_c]$; the lower-bound ($\geq$) is true when $p \in [p_c;1($.

Even though linearity was asked so that all decoding regions are isometric, it is not a requirement for this theorem. Indeed, the bounding equations are true for every codeword c by replacing d by $\min_{c' \in C, c' \neq c} d(c,c')$. Assuming that the codewords sent are distributed in a uniform way over C, we thus obtain this result.

The behaviour of this function is illustrated in Fig 2. Around $p \approx 0$ (actually, for all $p < p_c - \epsilon$ for a reasonable ϵ), P_e is extremely flat above its limit 0; around $p \approx 1$ (and, symmetrically, for all $p > p_c + \epsilon$, P_e is extremely flat below its limit 1. Finally, around the threshold p_c, the slope is $\frac{\sqrt{d}}{\sqrt{2\pi}(1-p_c)}$, which is almost vertical when the minimal distance d is large.

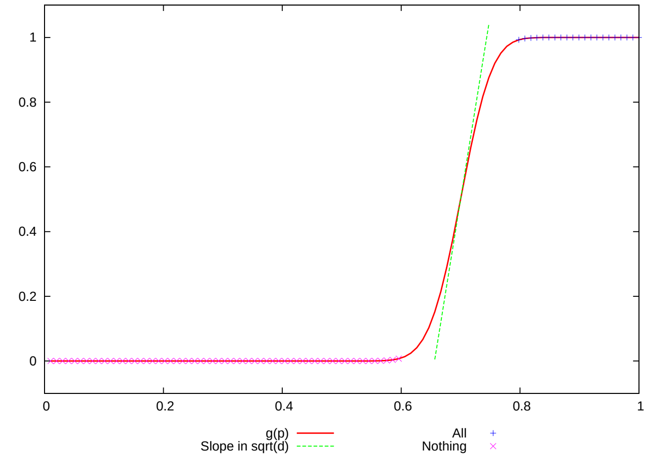

Figure 2. Illustration of the threshold effect, $d = 400$, $p_c = 0.7$

III. Explicit Computation of the Threshold for Maximum-Distance Separable Codes

In this section, we only take interest in linear codes over $\mathbb{F}_q^n$.

A. Another Estimation of the Decoding Threshold

By linearity, we can again without loss of generality assume that the sent codeword was the all-zero vector 0. A rough estimation of the probability of wrongly decoding (for a crossover probability p) can be estimated by the proportion $g(p)$ as follows:

$$g(p) = \frac{|\{x : \text{ s.t. } \exists c \in C, c \neq 0 : d(x,c) < w(x) \leq np\}|}{|\{x : w(x) \leq np\}|}.$$

$g(p)$ is in fact the proportion of vectors $x \in \mathbb{F}_q^n, w(x) \neq np$, that are closer to a non-zero codeword than to 0.

Let $vol(q,n,t) = \frac{1}{n}\log_q(V(t))$. It is well known that when $t \leq n(1-\frac{1}{q})$, $vol(q,n,t) = H_q(\frac{t}{n}) + o_n(1)$, where $H_q(x) = -x\log_q x - (1-x)\log_q(1-x) + x\log_q(q-1)$ is the q-ary entropy of $x \in [0,1]$.

To compute the numerator, we suggest, for each codeword $c \in C$ to compute the number of vectors x that are nearer to c than to 0. This number actually only depends on the weight of c, and will be noted $\nu_{pn}(w(c))$. It actually suffices to consider codewords whose weight is between d and $2pn$.

As there are $A_{w(c)}$ codewords of weight $w(c)$ in the code (with the standard notation), the function $g(p)$ can be approximated by:

$$g(p) \leq \frac{\sum_{l=d}^{2pn} A_l \nu_{pn}(l)}{q^{nvol(q,n,pn)}} \tag{2}$$

The different quantities used in this equation are illustrated in Fig 3.

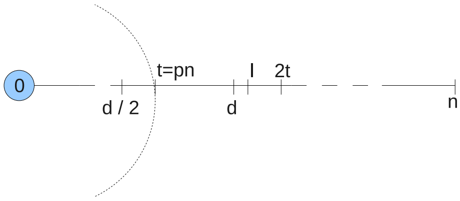

Figure 3. Different quantities used in Eq 2

The number $\nu_t(w)$ is obtained in the following combinatorial way. Let c be a codeword of weight w. Let $x \in \mathbb{F}_q^n$ be a vector with the following constraints:

- $d(x,0) \leq t$, *i.e.* x is the result of the transmission of 0 with at most t errors.
- $d(x,0) \geq d(x,c)$, *i.e.* x is wrongly decoded.

We note α the number of coordinates i in x such that $x_i \neq c_i$ and $x_i = 0$; β is the number of coordinates i such that $x_i \neq c_i$ and $x_i \neq 0$; γ is the number of coordinates i such that $x_i \neq c_i$ and $c_i = 0$.

The previous constraints on x can be rewritten into the system (S):

$$(S) : \begin{cases} 1) & 0 \leq \alpha, \beta \leq w \\ 2) & 0 \leq \gamma \leq n - w \\ 3) & \gamma \leq t + \alpha - w \\ 4) & \beta + \gamma \leq t \\ 5) & 2\alpha + \beta \leq w \end{cases}$$

We then obtain

$$\nu_t(w) = \sum_{\alpha,\beta,\gamma} \binom{w}{\alpha+\beta}\binom{\alpha+\beta}{\beta}(q-2)^\beta \binom{n-w}{\gamma}(q-1)^\gamma.$$

Remark 1: It is easy to see that $\nu_t(w)$ is at most the volume of a ball of radius $w - \frac{d}{2}$; this estimation will be used in the next part.

B. Application to MDS codes

Maximum-Distance Separable (**MDS**) Codes are codes such that their dimension k and minimal distance d fulfil the Singleton bound, so that:

$$k + d = n - 1.$$

A well known family of MDS codes are the Reed-Solomon codes, for which a codeword is made of the evaluation of a degree $k-1$ polynomial over n field elements $\alpha_1, \ldots, \alpha_n$. Reed-Solomon codes over $\mathbb{F}_q$ can have a length up to $q-1$, but shorter such codes are also MDS.

For MDS codes, the number A_l of codewords of given weight is known. This number is:

$$A_{n-i} = \sum_{j=1}^{n-1} (-1)^{j-i} \binom{n}{j}\binom{j}{i}(q^{k-j} - 1)$$

From this identity, it is easy to derive the more usable formula:

$$A_l = \binom{n}{l} \sum_{j=0}^{l-d} (-1)^j \binom{l}{j} (q^{1+l-d-j} - 1) \qquad (3)$$

It is now possible to approximate quite nicely the error probability while under the threshold - indeed, the numerator and denominator are correct as long as a vector x is not close to 2 different codewords with a weight in the range $[d; pn]$, *i.e.* as long as the list of codewords at a distance less than pn from x is reduced to a single element.

C. Short MDS Codes over Large Fields

We now focus on the specific problem presented in the Introduction, and motivated by the beckoning and authentication protocol from [1]. This setting is characterized by the following:

- The underlying code is a Reed-Solomon over a field $\mathbb{F}_q$;
- The field size q is very large for cryptographic reasons;
- The code length n is very short (with respect to q) as nq is the size of embedded low-cost devices' memory.

This application fits into the framework depicted in the previous sections. Moreover, the information "n much smaller than q" ($n = o(q)$) enables to compute an asymptotic first order estimation of the threshold in such codes.

Indeed, if $g(p) \leq f(p)$, then $g^{-1}(\frac{1}{2}) \geq f^{-1}(\frac{1}{2})$. We now compute an upper bound on $g(p)$, to derive an estimation on the threshold θ. More precisely, we aim at computing $\iota(p)$ the first-order value of $\log_q(g(p))$; then, $\iota^{-1}(0)$ is a lower-approximation of the threshold.

To estimate the weight enumerator A_l, we use formula (3) to derive

$$A_l \leq \binom{n}{l} 2^l q^{1+l-d} \leq 2^{n+l} q^{1+l-d}.$$

The number of targeted vectors for each codeword $\nu_t(l)$ is not easy to evaluate; we note its first order development $\log_q \nu_t(l) := n\mu(l,t) + o_q(1)$, so that $\nu_t(l) \leq q^{n\mu(l,t)} \cdot o_q(q)$. (Here, the term $o(q)$ is a bounded by a polynomial in n.) We know that

$$0 \leq n\mu(l,t) \leq l - \frac{d}{2} \qquad (4)$$

Combining these elements with equation (2), we obtain

$$g(p) \leq \sum_{l=d}^{2pn} q^{(n+l)\log_q(2)+1+l-d+n\mu(l,pn)-nvol(q,n,pn)}.$$

As $vol(q,n,t) = H_q(\frac{t}{n}) + o_n(1) = \frac{t}{n} + o_q(1)$, the first order of $g(p)$ is bounded by: $\log_q g(p) \leq \max_{l\in[d,pn]}(1 + l - d - pn + n\mu(l,pn)) + o_q(1)$.

The bounding (4) of μ shows that the right-hand side of this inequality is between $1 + pn - d$ and $1 + 3pn - \frac{3d}{2}$, which shows that the threshold $g^{-1}(\frac{1}{2})$ is asymptotically between $\frac{\delta}{2}$ and δ.

Unfortunately, a more precise evaluation of μ strongly depends on the context. Indeed, according to Section III-A,

$$\nu(l,t) = o_q(q) \cdot \max_{\alpha,\beta,\gamma:(S)} q^{\beta+\gamma} \binom{n-l}{\gamma} \binom{l}{\alpha+\beta} \binom{\alpha+\beta}{\beta}.$$

This maximum can be obtained by evaluating the term to be maximized on all vertices of the polytope defined by the system (S). (S) is made of 9 inequalities of 3 unknown, the vertices are obtained by selecting 3 of these equations, thus at most $\binom{9}{3} = 84$ vertices. However, it is not possible to exhibit here a general answer as the solution depends on the minimal distance of the code, *i.e.* on the rate of the Reed-Solomon code.

D. Numerical Application to a $(2048, 256, 1793)_{2^{64}}$ *MDS Code*

In the case of a code over a finite field of reasonable dimension, it is possible to exactly compute the ratio that approximates the Maximum Likelihood threshold. However, the exact threshold cannot be easily computed yet; it is still an open problem related to the list-decoding capacity of Reed-Solomon codes.

We therefore used the NTL open-source library [9] to compute the values A_l, $\nu_t(l)$ and $|B(t)|$ in order to have an accurate enough approximation of the the function $g(p)$ described earlier. The parameters are those that were proposed in [1]. The results show that the decoding threshold of such a code is between 0.8 and 0.875, in other words, just below the conservative upper-bound $1 - \frac{k}{n}$

The slope around the threshold is around 115, so for p “small” (in fact, a bit smaller than p_c) $g(p)$ is very near to 0, while as p goes to 1, $g(p)$ is much greater than the maximum probability of 1. This was predicted earlier, and expresses the fact that the list-size of radius pn is always greater than 1. The threshold value $g^{-1}(\frac{1}{2}) \approx \iota^{-1}(0)$ is a lower-bound for the threshold of the code, though the intuition says that this lower-bound is pretty near to the real threshold.

This result is coherent with previous results on the hardness of decoding Reed-Solomon codes. [10] study the hardness of the List-Decoding problem, and shows that for the furthest vectors (vectors that accomplish the covering radius of the code, a.k.a. “Deep Holes”) maximum-likelihood decoding is an NP-hard problem. The covering radius is, in this case, $\rho = n - k = d - 1$. Moreover, the average list size is the missing factor in inequality (2), if it is small before the numerator of (2) then the approximation is accurate. [11] shows that after list-decoding, the average list size is small up to the distance $\rho - 1$. All these elements indicate that the estimation (2) is tight.

IV. Conclusion

As a conclusion, let us look back to the starting point of our reasoning. The initial goal was to revise the conditions of security of the construction depicted in [1]: from a received vector x of $\mathbb{F}_q^n$, for what parameters is the size of the list of radius pn exponentially large? This problem can be reduced to that of the threshold probability of a linear error-correcting code. Indeed, below the threshold of the code, when the minimal distance of the code is large enough, the error decoding probability of the code is exponentially small, and it is exponentially close to 1 above the threshold. For our class of parameters, ensuring that the error rate is above the threshold is enough to show the security of the scheme.

We show that the threshold behaviour can be demonstrated for q-ary codes as well as for binary codes; we then compute a lower-bound on the threshold of MDS codes.

Applying these results to the initial problem, we show that the threshold for a (highly) truncated Reed-Solomon code over a finite field $\mathbb{F}_{2^{64}}$ is very near to normalized the minimal distance $d = n - k + 1$ of this code. As a conclusion, to switch from an algorithmic assumption (the hardness of the Polynomial Reconstruction Problem [4]) to Information-Theoretical security, we recommend to raise the dimension k of the underlying code. This lowers the decoding threshold of the code; the downside is that storage of a codeword is more costly.

V. Acknowledgements

We thank Gilles Zémor for the useful comments and fruitful discussions, and the anonymous referees for pointing some missing references and for the helpful remarks.

References

[1] 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.

[2] V. Guruswami and M. Sudan, “Reflections on ”improved decoding of reed-solomon andalgebraic-geometric codes”,” 2002.

[3] V. Guruswami and A. Rudra, “Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy,” *Information Theory, IEEE Transactions on*, vol. 54, no. 1, pp. 135 –150, jan. 2008.

[4] A. Kiayias and M. Yung, “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.

[5] G. R. Grimmett, “Percolation,” 1997.

[6] J.-P. Tillich and G. Zémor, “Discrete isoperimetric inequalities and the probability of a decoding error,” *Comb. Probab. Comput.*, vol. 9, no. 5, pp. 465–479, 2000.

[7] G. Zémor, “Threshold effects in codes,” in *Algebraic Coding*, ser. Lecture Notes in Computer Science, G. D. Cohen, S. Litsyn, A. Lobstein, and G. Zémor, Eds., vol. 781. Springer, 1993, pp. 278–286.

[8] G. A. Margulis, “Probabilistic characteristics of graphs with large connectivity,” *Problemy Peredači Informacii*, vol. 10, no. 2, pp. 101–108, 1974.

[9] V. Shoup, “Ntl: A library for doing number theory.” [Online]. Available: http://www.shoup.net/ntl

[10] V. Guruswami and A. Vardy, “Maximum-likelihood decoding of reed-solomon codes is np-hard,” in *SODA ’05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2005, pp. 470–478.

[11] R. J. Mceliece, “On the average list size for the guruswami-sudan decoder,” in *ISCTA03*, 2003.