Efficient decoding of (binary) cyclic codes above the correction capacity of the code using Gröbner bases

Daniel AUGOT, Magali BARDET , Jean-Charles FAUGÈRE

November 1, 2002

1 Introduction

This paper revisits the topic of decoding cyclic codes with Gröbner bases. It has been shown that the problem of decoding cyclic codes up to their true minimum distance can be solved by the use of Gröbner bases [CRHT94c, CRHT94b, LVY97]. The principle is to rewrite the decoding problem into an algebraic system of equations, which must have the following properties:

1. *Decoding property*: its solutions are closely related to the error e,

2. *Computational property*: the computation of its solutions can be done in reasonable time.

But as long as a Gröbner basis computation is used to find the solutions, the computational times vary a lot. For details on ideals, Gröbner basis and polynomial system solving, the reader should refer to [CLO97].

Motivated by the problem of decoding quadratic residue (QR) codes, for which no general decoding algorithm is known, we improve on several points. First we introduce modified systems, without high degree equations, which still have the decoding property but for which the Gröbner basis computation is much easier. The Gröbner basis computation can be done either as a preprocessing, with the parameters taken as variables, or for each word to be decoded, with the parameters computed from the word. In the first case (*formal decoding*), we get formulas and we just have to compute the parameters and to evaluate the formulas to decode a word. In the second case (*online decoding*), we compute for each word the parameters and a Gröbner basis of the specialized system, but each system has less variables and the Gröbner basis is much easier to compute than in the formal case.

We show on the example of the [41,21,9] QR code that the size of formulas obtained in the formal Gröbner basis is too large to be useful, since the remainder evaluation for each word takes too much time. Hence, even if the formal computation can be achieved, the online decoding approach seems to be faster. We get efficient and automatic decoding algorithms which work for *any* cyclic code and enable to decode *above* the true minimum distance. We give many examples of decoding (for BCH codes of length 75, 511, for QR codes of length 73, 89, 113, 151 and for a code of length 75 which does not belong to a known class of codes). Moreover, using a general compilation method useful for systems with parameters, we improve the efficiency of our algorithms: for each cyclic code, we automatically generate a C program which, executed on any word, gives the corresponding solution without computing directly a Gröbner basis. We know exactly the complexity of the decoding algorithm (i.e. the number of arithmetic operations) for any of these programs. For any weight v, we get all the codewords at distance less than or equal to v from the error, hence we are able to decode above the correction capacity of the code.

The paper is organized as follows: we recall in Section 2 basic facts about cyclic codes, and previous work. We introduce in Section 3 our new systems and show that they satisfy the decoding property. We study the size of the formulas for the [41,21,9] QR code. Section 4 presents various examples of decoding algorithms by online Gröbner basis computation, in particular the compilation method used for more efficiency. We explain how it decodes above the correction capacity of the code. We give finally in Section 5 many examples of decoding algorithms, with computational times and complexity.

2 Notations and previous work

Let $\mathcal{C}$ be a cyclic code of length n over $\mathbb{F}_2$, with defining set $\mathcal{Q} \subset \{1, \dots, n\}$ and correction capacity t, and let $\alpha \in \mathbb{F}_{2^m}$ be a primitive n-th root of unity.

If $\tilde{c} = c + e$ is the received word, where $e(x) = \sum_{r=0}^{n-1} e_r x^r$ is the error of weight v to be decoded, if $r_1, \dots, r_v$ locate the positions of the non-zero e_r's, we define the following quantities: for $1 \leq j \leq v$ let $Z_j = \alpha^{r_j}$ denote the locators of the error, $L(Z) = \prod_{j=1}^{v}(Z - Z_j) = Z^v + \sigma_1 Z^{v-1} + \cdots + \sigma_{v-1}Z + \sigma_v$ denote the locator polynomial of e and $S_i = e(\alpha^i) = \sum_{j=1}^{v} Z_j^i$ denote the elementary power functions of the Z_j's for $1 \leq i \leq n$. From the received word $\tilde{c}$ we are able to compute the *syndroms* of e, $S_i^* = e(\alpha^i) = \tilde{c}(\alpha^i) \quad \forall i \in \mathcal{Q}$.

From now on, we shall distinguish an actual value from an indeterminate (variable) by appending a $*$ to it. For instance S_1^* is a value given to the indeterminate S_1. We also use the following notations: $\underline{Z}_v = (Z_j)_{j \in [1,v]}$, $\underline{\sigma}_v = (\sigma_j)_{j \in [1,v]}$, $\underline{S} = (S_i)_{i \in Q}$, $\overline{S} = (S_i)_{i \notin Q}$. As long as $v \leq t$, the system

$$\text{POWER FUNCTIONS} \begin{cases} S_i - \sum_{j=1}^{v} Z_j^i, & i \in Q \end{cases} \tag{1}$$

specialized for $S_i = S_i^*$ has a unique solution (cf. [RTCY92] pp. 981) and the error can be corrected. In addition to the system (1), the syndroms S_i^* and the elementary symmetric functions σ_j^* of then Z_j's are solutions of the following systems:

$$\text{SYMMETRIC FUNCTIONS} \begin{cases} \sigma_j - \sum_{l_1 < \cdots < l_j} Z_{l_1} \cdots Z_{l_j} & j \in [1, v] \end{cases} \tag{2}$$

$$\text{NEWTON'S IDENTITIES} \begin{cases} S_i + \sum_{j=1}^{i-1} \sigma_j S_{i-j} + i\sigma_i & i \in [1, v] \\ S_i + \sum_{j=1}^{v} \sigma_j S_{i-j} & i \in [v, v+n-1] \end{cases} \tag{3}$$

and the Z_j's are n-th roots of unity, hence $Z_j^{n+1} - Z_j = 0, \ 1 \leq j \leq v$.

These systems have already been studied [CRHT94c, CRHT94b, LVY97, CRHT94a]. P. Loustaunau and E. Von York prove in particular in [LVY97] that a Gröbner basis of the system

$$\begin{cases} S_i - \sum_{j=1}^{v} Z_j^i & i \in Q \\ Z_j^{n+1} - Z_j & j \in [1, v] \end{cases} \tag{4}$$

for a Lexicographical order such that $\underline{Z}_v > \underline{S}$ (where $\underline{Z}_v$ are *variables* and $\underline{S}$ are *parameters*) contains polynomials that become the locator polynomial of an error when specialized on the syndroms S_i^* of the error. Hence, the precomputation of a Gröbner basis for this system in $\mathbb{F}_2[\underline{Z}_v, \underline{S}]$ gives formulas, and the decoding algorithm is: for any error e, evaluate the formulas on the syndroms of e. It works also as an *online* decoding: for any error e, specialize the system (4) on the syndroms of e, and compute the Gröbner basis of this system in $\mathbb{F}_{2^m}[\underline{Z}_v]$ for a lex order, then it consists of only one polynomial, the locator polynomial of e.

For the *formal* decoding, the burden of Gröbner basis computation is supported only during preprocessing, but in almost all cases the computation is infeasible. This comes from the high degree polynomials always contained in the Gröbner basis (e.g. $S_i^{2^m} - S_i$), which follows from the *field equations* $Z_j^{n+1} - Z_j$. It makes the computation of the formal Gröbner basis intractable for codes of length greater than 31 for QR codes. Even when the precomputation can be achieved, we will see in the next section that the size of the formulas seems to be too large to be used for a fast decoding, since its evaluation on syndroms would have a cost corresponding to it size.

3 New Systems of positive dimension

We consider the new systems obtained from the preceding ones by removing the fields equations. We prove in proposition 2 that they have good properties for the decoding problem. We show in section 4 that these systems are also very efficient from the computational point of view.

Using the Newton's identities (3), we eliminate the unknowns syndroms S_i, $i \notin \mathcal{Q}$ by successive substitutions, and we obtain the system for the weight v

$$(\mathrm{BIN}_v)\,\{S_i - f_i(\sigma_1,\ldots,\sigma_v) = 0 \quad i \in Q$$

where the f_i's are the Waring functions (see [LN97] pp. 30):

$$f_i(\sigma_1,\ldots,\sigma_v) = \sum_{i_1+2i_2+\cdots+vi_v=i} \frac{(i_1+i_2+\cdots+i_v-1)!}{i_1!\cdots i_v!} \cdot i \cdot \sigma_1^{i_1}\ldots\sigma_v^{i_v}$$

Proposition 1. *We have the following relations between those systems:*

$$\begin{aligned}\langle \mathrm{BIN}_v\rangle &= \langle S_i - \sum_{j=1}^{v} Z_j^i \quad i\in Q, \sigma_j - \sum_{l_1<\cdots<l_j} Z_{l_1}\cdots Z_{l_j}\ j\in[1,v]\rangle \bigcap \mathbb{F}_2[\underline{\sigma}_v,\underline{S}] \\ &= \langle S_i + \sum_{j=1}^{i-1}\sigma_j S_{i-j} + i\sigma_i, \quad i\in[1,v], S_{v+i} + \sum_{j=1}^{v}\sigma_j S_{v+i-j}, \quad i\in[1,n]\rangle \bigcap \mathbb{F}_2[\underline{\sigma}_v,\underline{S}]\end{aligned}$$

and this ideal is a prime one (see [CLO97] chapter 7 §4).

The main difference with the preceding systems is that, because we removed the field's equations, the set of solutions associated with the ideal contains now solutions in the algebraic closure of $\mathbb{F}_{2^m}$. The following proposition show that these parasitic solutions (solutions in the algebraic closure not corresponding to a codeword) are well known, and that the computations of the real solutions is still easy.

Proposition 2 (Unicity). *We denote by $V(I)$ the variety associated with an ideal I, and by Π_l the projection eliminating the l first coordinates, i.e. $\Pi_l(x_1,\ldots,x_k) = (x_{l+1},\ldots,x_k)$. Let $\underline{S}^* \subset \mathbb{F}_{2^m}$ be the syndrom of an error $e \in \mathbb{F}_2[x]/(x^n-1)$ of Hamming's weight $v \le t$, where t is the correction capacity of the code.*

- *the specialized system $(\mathrm{BIN}_v(\underline{S}^*))$ has a unique solution $(\sigma_1^*,\ldots,\sigma_v^*)$ and $L_e(Z) = \sum_{j=0}^{v}\sigma_j^* Z^{v-j}$ is the locator polynomial of e.*
- *$\exists! w$, $\exists!(\sigma_1^*,\ldots,\sigma_w^*)$ such that $(\sigma_1^*,\ldots,\sigma_w^*,0_{t-w},\underline{S}^*) \in V(\mathrm{BIN}_t)$ and $(0_{t-w+1},\underline{S}^*) \notin \Pi_{t-w+1}(V(\mathrm{BIN}_t))$. The weight of e is therefore exactly $v = w$ and $\sum_{j=0}^{v}\sigma_j^* Z^{v-j}$ is the locator polynomial $L_e(Z)$ of e.*
- *if $\tilde{\sigma}^* = (\tilde{\sigma}_1^*,\ldots,\tilde{\sigma}_t^*)$ is such that $(\tilde{\sigma}^*,\underline{S}^*) \in V(\mathrm{BIN}_t)$, then $L_e(Z)$ divides $L(Z) = \sum_{j=0}^{t}\tilde{\sigma}_j^* Z^{t-j}$ and $L_e(Z)$ can be obtained considering the factors of $L(Z)$ with odd multiplicity and distinct from Z. More precisely, $L(Z) = p_1^{e_1}\cdots p_k^{e_k}Z^a \Rightarrow L_e(Z) = p_1^{(e_1 \bmod 2)}\cdots p_k^{(e_k \bmod 2)}$.*

Proof. Omitted because of space requirements. ☐

This means that if we compute a Gröbner basis of (BIN) for a degree ordering with $\underline{\sigma} > \underline{S}$, we will get formulas for the σ_j's in terms of the S_i's of *minimal degree*. We expect these equations to be of degree one, but even if they are of high degree, they have a unique root when specialized on a syndrom.

Any syndrom S_i gives an equation of degree i. If we take all these equations, we get equations of high degree, and the Gröbner basis is hard to compute. A first remark is that if $i \in Q$ and $2i \in Q$ we have the relation $S_{2i} = S_i^2$, so we only need to consider the odd syndroms. Moreover, if we select a subset $E \subset \mathcal{Q}$ and obtain linear equations for σ_j in the Gröbner basis of $\{S_i - f_i(\underline{\sigma}) : i \in E\}$ then we have decoding formulas.

Example. We have reproduced for the QR code [31,16,7] the formulas found by hand in [RYT90]. The full Gröbner basis contains 32 polynomials, but we can select one linear polynomial for each σ_i, which gives valid formulas provided that the coefficient in σ_j does not vanish when specialized on a syndrom.

When trying the same computation for the QR code 41, which odd syndroms are $\{1, 5, 9, 21, 23, 25, 31, 33, 37, 39\}$, we were not able to get linear formulas: either the Gröbner basis could not be computed, either σ_4 appeared as a free variable (when too much equations were removed). We use as in [RTCY92] the fact that -1 is a syndrom, hence the relation $\sigma_3 = S_{40}\sigma_4$ is satisfied. More generally, eliminating the unknowns syndroms starting from the last, we can derive as for the system (BIN) equations $\sigma_v^{n-i}S_i = f_{n-i}(\frac{\sigma_{v-1}}{\sigma_v},\ldots,\frac{1}{\sigma_v})\sigma_v^{n-i}$

which are polynomials of degree $n-i+1$. Here we get another equation $S_{36}\sigma_4^5+\sigma_3^5+\sigma_4^2\sigma_2^2\sigma_3+\sigma_4^3\sigma_2\sigma_1+\sigma_4^3\sigma_3+\sigma_4\sigma_3^3\sigma_2+\sigma_4^2\sigma_3^2\sigma_1$. We use 5 equations given by the syndroms $\{1,5,9,36,40\}$. The first and the last one are used to eliminate the variables σ_1 and σ_3. The Gröbner basis for the syndroms $\{5,9\}$ gives a polynomial P_1 of degree 5 in σ_4, and the Gröbner basis for the syndroms $\{9,36\}$ gives a polynomial P_2 of degree 4 in σ_4. The formal gcd of these two polynomials in σ_4 could be computed using Magma, and we obtained a linear polynomial in σ_4, which coefficients are polynomials in $S_1,S_5,S_9,S_{36},S_{40}$ of total degree 170, and with 29828 terms. This indicates that, in general, the size of the linear formula with formal parameters is very large, and even if it can be obtained, it is useless for decoding. Thus the idea of doing precomputation of formal Gröbner basis is not relevant for effective decoding of cyclic codes.

4 Practical decoding

We turn back to the original approach of [CRHT94c, CRHT94b], and consider in this Section online decoding. For each word e, we construct the system $\textsc{Bin}(\underline{S}^*)$, and compute its Gröbner basis over $\mathbb{F}_{2^m}$. The system has a unique solution (cf. Proposition 2), and it turns out that formulas of degree one are obtained (but we did not proved that there is no multiplicity). This means that the Gröbner basis has in practice the shape

$$\textsc{Bin}(\underline{S}^*)\begin{cases}\sigma_1+\sigma_1^*\\ \sigma_2+\sigma_2^*\\ \quad\vdots\\ \sigma_v+\sigma_v^*\end{cases} \tag{5}$$

where the σ_i^*'s are the actual coefficients of the locator polynomial.

We use a general method for solving systems with parameters. A specialization property (see [FGT01]) tells us that the result of an online Gröbner basis computation over $\mathbb{F}_{2^m}$ is the specialization of the formal Gröbner basis over $\mathbb{F}_2$. This property seems to extend to the fact that all steps of the computation of the specialized basis are the specialization of all steps of the computation of the formal basis. In other words, the behavior of the Gröbner basis computation is the same for all the possible values of the syndroms, provided that it corresponds to an error of a given weight. We use this remark to drastically reduce the complexity of the online computation (we gain a factor 1000).

We describe the method in the particular case of the F4 algorithm ([Fau99]), because this is the one we use in practice, but it also applies to other algorithms. The F4 algorithm uses the correspondence between polynomial algebra and linear algebra. It constructs several matrices from polynomials, and uses linear algebra to compute the Row Echelon form for each of these matrices (see [Fau99] for more details).

Considering the computation of a Gröbner basis of $\textsc{Bin}(S_{e_0}^*)$ for a given error e_0 of weight v_0, we can record the trace of the computation (in our case the program which compute the Gröbner basis generate another C program). Now let e be another error of weight v_0, we can run the C program on the syndroms of e. It successively constructs matrices, in the same way as for e_0, and perform linear algebra on it, as for e_0. The fact that experimentally we always obtain exactly the Gröbner basis of $\textsc{Bin}(S_e^*)$ comfort the idea that the specialization property extends to all steps of the computation. These considerations justify the following algorithm:

- Preprocessing: compute a Gröbner basis for $\textsc{Bin}_v(S_{e_0}^*)$ for a randomly chosen error e_0 of weight v, and record the trace of all linear algebra computations performed (for instance as a C program).
- Decoding: for an error e, execute the C program on $\textsc{Bin}_v(S_e^*)$ and get the values of the σ_j's.

The benefits of using such a C program instead of using a generic algorithm for computing Gröbner basis is the gain in efficiency. Indeed, the C program only performs linear algebra operations, in a prescribed manner. Using an analogy, it is the same as performing a Gaussian elimination with all the pivoting elements and the row operations known in advance.

Let us note that the execution of the C program succeeds only if the error e has the same weight v as e_0. For a given code $\mathcal{C}$ correcting t errors, a decoding algorithm consists in t programs $P_1,\ldots,P_t$, one for each possible weight. To decode, execute the programs in sequence, starting from P_1 to P_t, until the resulting

	% of errors having n_3 solutions of weight 3 and n_4 of weight 4)								
(n_3, n_4)	(0,1)	(0,2)	(1,1)	(0,3)	(1,2)	(0,4)	(1,3)	(0,5)	(1,4)
	31%	29,6%	4,9%	14,8%	5,9%	5,9%	4,4%	1,5%	2%

Figure 1: Decoding the errors of weight 4 for the QR code of length 31.

δ	k	d	t	number of errors	number of solutions	number of $*$	number of random tests giving more than 1 solution
93	175	95	47	48 to 49	1	$2^{15.4}$	0/100000
				50	1	$2^{16.9}$	0/100000
				51	1	$2^{22.7}$	0/1000
91	184	91	45	46 to 47	1	$2^{15.2}$	0/100
				48	1	$2^{15.7}$	0/100
				49	1	$2^{19.8}$	0/100
				50	1	$2^{25.5}$	0/100

Figure 2: Decoding BCH Codes $[511, k, \delta]$ beyond t over $\mathbb{F}_{512}$

system does not contain 1. Note that now, contrarily to the computation of a Gröbner basis using a general algorithm, we are able to predict the time needed for the decoding (see Section 5). As we only perform linear algebra, we can give explicitly the number of arithmetic operations in the field $\mathbb{F}_{2^m}$ that are needed to decode a word.

Indeed, the set of solutions of the system $\mathrm{B{\scriptsize IN}}_v(\underline{S}^*)$ is the set of all the errors of weight less than or equal to v which have $\underline{S}^*$ as syndroms. As long as there exists only one error of weight less than v with syndroms $\underline{S}^*$, this error can be decoded, even if v is greater than t. In the other cases, this enables to do list decoding up to the weight v. The size of the list is not known, and may be large. Note also that the complexity of the Gröbner basis computation increases with the size of the list. We illustrate this result with the [31,16,7] QR code: we made an exhaustive search for all errors of weight 4 (there are 31465 of them). As we can see in Figure 1, 31% of these errors can be decoded (i.e. there is a single codeword at minimal distance less than or equal to 4), and for 4.9% of them the set of solutions of $\mathrm{B{\scriptsize IN}}_4(\underline{S}^*)$ contains one solution of weight 4 and one solution of weight 3. This set of solutions is always of size at most 5.

5 Results

We present here results for some selected codes. For each code, we give the number of multiplication (*) in the field $\mathbb{F}_{2^m}$ which occurred during the execution of the C programs.

Figure 2 presents the decoding of two BCH codes of length 511, with designed distance 93 and 91, above their correction capacity. We are able to decode far beyond the correction capacity of the code. For instance, for the BCH code [511,175,93] the true minimum distance is 95, hence it corrects 47 errors, but we are able to correct up to 51 errors. Figure 3 shows decoding algorithms for some QR codes (for which no decoding algorithm was known before).

We compare finally in Figure 4 two codes of length 75 : the BCH code [75,31,7] and a code of type [75,33,7] and defining set $\{1, 3, 25\}$ which does not belong to a known class of codes. Our method is independent of taking the code in a specific class, any cyclic code can be decoded in the same way. We chose a code which is better than the corresponding BCH code (it has the same length, the same minimum distance, but its dimension is smaller, and it behaves a little better above 4 errors).

References

[CLO97] David Cox, John Little, and Donal O'Shea. *Ideals, varieties, and algorithms.* Springer-Verlag, New York, second edition, 1997. An introduction to computational algebraic geometry and commutative algebra.

n	d	Field	number of errors	number of solutions	number of $*$	% of tests giving i solutions $i=1$	$i=2$	$i=3$	∞
73	13	2^9	1-6	1	$2^{5.6}$ to 2^{15}	100%			
			7	1-2	$2^{19.4}$	98.9%	1.1%		
			8	1-2	$2^{22.3}$	80%	17%	2%	1%
89	17	2^{11}	1-5	1	$2^{6.1}$ to $2^{12.8}$	100%			
			6	1	$2^{15.7}$	100%			
			7	1	$2^{19.2}$	100%			
			8	1	$2^{22.4}$	100%			
113	15	2^{28}	1-5	1	$2^{6.5}$ to $2^{11.7}$	100%			
			6	1	$2^{14.3}$	100%			
			7	1	$2^{18.1}$	100%			
			8	1	$2^{22.5}$				
151	19	2^{15}	1-5	1	$2^{7.7}$ to $2^{13.5}$	100%			
			6-7	1	$2^{16.7}$ to 2^{20}	100%			
			8	1	$2^{25.6}$	100%			

Figure 3: Decoding QR Codes $[n, \frac{n+1}{2}, d]$.

Code	number of errors	number of solutions	time (seconds)	number of random tests	number of tests giving i solutions $i=1$	2	3	4	5	6	8	12	∞
BCH	1-3	1	2^6 to 2^8	10000									
$[75, 31, 7]$	4	1 to 4	$2^{10.8}$	10000	9940	53		7					
	5	1 to 6	$2^{11.3}$	10000	9375	533		45		23			23
Random code	1-3	1	2^6 to 2^8	10000									
$\mathcal{Q} = \{1, 3, 25\}$	4	1 to 4	2^{10}	10000	9940	53		7					
$[75, 33, 7]$	5	1 to 6	$2^{10.7}$	10000	9618	344		35		2			
	6	1 to 12	$2^{15.9}$	10000	8823	882	70	91	30	54	16	3	31

Figure 4: Decoding the BCH Code $[75, 31, 7]$ and a code with $\mathcal{Q} = \{1, 3, 25\}$ and type $[75, 33, 7]$.

[CRHT94a] Xuemin Chen, I.S. Reed, T. Helleseth, and T.K. Truong. Algebraic decoding of cyclic codes: A polynomial ideal point of view. In Gary L. et al. Mullen, editor, *Finite fields: theory, applications and algorithms*, volume 168 of *Contemp. Math.*, pages 15–22. RI: American Mathematical Society., 1994.

[CRHT94b] Xuemin Chen, I.S. Reed, T. Helleseth, and T.K. Truong. General principles for the algebraic decoding of cyclic codes. *IEEE Transactions on Information Theory*, 40(5):1661–1663, 1994.

[CRHT94c] Xuemin Chen, I.S. Reed, T. Helleseth, and T.K. Truong. Use of Groebner bases to decode binary cyclic codes up to the true minimum distance. *IEEE Transactions on Information Theory*, 40(5):1654–1661, 1994.

[Fau99] Jean-Charles Faugère. A new efficient algorithm for computing Groebner bases (F_4). *J. Pure Appl. Algebra*, 139(1-3):61–88, 1999.

[FGT01] Elisabetta Fortuna, Patrizia Gianni, and Barry Trager. Degree reduction under specialization. *J. Pure Appl. Algebra*, 164(1-2):153–163, 2001. Effective methods in algebraic geometry (Bath, 2000).

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite fields*. Cambridge University Press, Cambridge, second edition, 1997. With a foreword by P. M. Cohn.

[LVY97] Philippe Loustaunau and Eric Von York. On the decoding of cyclic codes using Groebner bases. *Appl. Algebra Eng. Commun. Comput.*, 8(6):469–483, 1997.

[RTCY92] I.S. Reed, T.K. Truong, Xuemin Chen, and X. Yin. The algebraic decoding of the (41, 21, 9) quadratic residue code. *IEEE Transactions on Information Theory*, 38(3):974–986, 1992.

[RYT90] I.S. Reed, X. Yin, and T.K. Truong. Algebraic decoding of the (32,16,8) quadratic residue code. *IEEE Transactions on Information Theory*, 36(4):876–880, 1990.