Gröbner bases for decoding

Mario de Boer and Ruud Pellikaan *

Appeared in *Some tapas of computer algebra*
(A.M. Cohen, H. Cuypers and H. Sterk eds.),
Chap. 11, Gröbner bases for decoding, pp. 260-275,
Springer, Berlin 1999,
after the EIDMA/Galois minicourse on Computer Algebra,
September 27-30, 1995, Eindhoven.

*Both authors are from the Department of Mathematics and Computing Science, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

Contents

1 Introduction

From the previous chapter one might get the impression that the theory of error-correcting codes is equivalent to the theory of finite geometry or arrangements over finite fields. This is not true from a practical point of view. A code is useless without a decoding algorithm. For engineers the total performance of the encoding and decoding scheme is important.

An introduction to the decoding problem is given in Section 2. In Section 3 we first restrict ourselves to cyclic codes where the system of syndrome equations can be explicitely solved using Gröbner basis techniques and later, in Section 5, to arbitrary linear codes. Although this method decodes up to half the true minimum distance, the complexity is not polynomial, because there is no polynomial algorithm known to compute Gröbner bases. The algorithms of Euclid, Sugiyama and Berlekamp-Massey give an efficient way to decode cyclic codes is by solving the key equation.

All references and suggestions for further reading will again be given in the notes of Section 6.

2 Decoding

Let C be a linear code. Decoding is the inverse operation of encoding. A *decoder* is a map

$$\mathcal{D} : \mathbb{F}_q^n \longrightarrow C \cup \{?\},$$

such that $\mathcal{D}(\mathbf{c}) = \mathbf{c}$ for all $\mathbf{c} \in C$. Let $\mathbf{y}$ be a *received word*. Then $\mathcal{D}(\mathbf{y})$ is a codeword or equal to ?, in case of a *decoding failure*

Decoding by *error detection* does the following. Let H be a parity check matrix of C. The output of the decoder is $\mathbf{y}$ if $\mathbf{y}H^T = 0$, and ? otherwise.

If the received word $\mathbf{y}$ is again a codeword, but not equal to the one sent, then the decoder gives $\mathbf{y}$ as output and we have a *miscorrection* also called a *decoding error*.

Let $C \subseteq \mathbb{F}_q^n$ be the code with minimum distance d that is used to transmit information over a noisy channel. If the codeword $\mathbf{c}$ is transmitted at one side of the channel and $\mathbf{y}$ is received at the other end, then we say that the *error* $\mathbf{e} = \mathbf{y} - \mathbf{c}$ has occurred:

$$\mathbf{y} = \mathbf{c} + \mathbf{e}.$$

A decoder $\mathcal{D}$ is called a *minimum distance decoder* if $\mathcal{D}(\mathbf{y})$ is a codeword that is nearest to $\mathbf{y}$ with respect to the Hamming metric for all $\mathbf{y}$.

Minimum distance decoding is similar to finding a codeword of minimal weight. If $\mathbf{y}$ is a received word, then one has to find a word in the coset $\mathbf{y} + C$ of minimal weight. Such a word is called a *coset leader*. Having a list of all coset leaders requires a memory of q^{n-k} of such elements and is only efficient for codes of small redundancy.

If the Hamming weight of the error-vector is at most $\lfloor (d-1)/2 \rfloor$, then $\mathbf{c}$ is the unique codeword which has the smallest distance to $\mathbf{y}$, so the error can be corrected. The value $t = \lfloor (d-1)/2 \rfloor$ is called the *error-correcting capability* or *capacity* of the code.

Let H be a parity check matrix for C, so $\mathbf{c}H^T = 0$ for all $\mathbf{c} \in C$. After receiving $\mathbf{y}$ one computes the vector of *syndromes*

$$\mathbf{s} = \mathbf{y}H^T.$$

Since $\mathbf{y} = \mathbf{c} + \mathbf{e}$ we have that $\mathbf{s} = \mathbf{y}H^T = \mathbf{c}H^T + \mathbf{e}H^T = \mathbf{e}H^T$ and the problem becomes: given $\mathbf{s}$, find a vector $\mathbf{e}$ of lowest Hamming weight such that $\mathbf{e}H^T = \mathbf{s}$.

A decoder $\mathcal{D}$ is called a *bounded distance decoder* which *corrects t errors* if $\mathcal{D}(\mathbf{y})$ is a codeword that is nearest to $\mathbf{y}$ for all $\mathbf{y}$ such that $d(\mathbf{y}, C) \leq t$. We say that $\mathcal{D}$ *decodes up to half the minimum distance* if it corrects $\lfloor (d-1)/2 \rfloor$ errors.

Proposition 2.1 *Let C be a linear code in $\mathbb{F}_q^n$ with parity check matrix H. Suppose we have a received word $\mathbf{y}$ with error vector $\mathbf{e}$ and we know a set J with at most $d(C) - 1$ elements and that contains the set of error positions. Then the error-vector $\mathbf{e}$ is the unique solution for $\mathbf{x}$ of the following linear equations:*

$$\mathbf{x}H^T = \mathbf{y}H^T \quad \textit{and} \quad x_j = 0 \ \textit{ for } \ j \notin J.$$

Exercise 2.2 Prove Proposition 2.1 and deduce that the syndrome of a received word with at most $\lfloor (d-1)/2 \rfloor$ errors is unique.

Proposition 2.1 shows that error decoding can be reduced to the problem of finding the error positions. If we want to decode all received words with t errors, then there are $\binom{n}{t}$ possible t-sets of error positions one has to consider. This number grows exponentially with n if t/n tends to a non-zero real number. The decoding problem is hard. Only for special families of codes this problem has an efficient solution with practical applications. We will consider only bounded distance decoders.

Exercise 2.3 Assume that the channel is a *q-ary symmetric channel*. This means that the probability that the symbol $x \in \mathbb{F}_q$ is changed in the symbol $y \in \mathbb{F}_q$ is the same for all $x, y \in \mathbb{F}_q$ and $x \neq y$, and does not depend on the position. The probability that a fixed symbol is changed in another symbol, distinct from the original one, is called the *crossover* probability and is denoted by P. Prove that the probability that an error vector $\mathbf{e}$ is equal to the word $\mathbf{c}$ of weight t is given by

$$\text{Prob}\{\mathbf{e} = \mathbf{c}\} = \left(\frac{P}{q-1}\right)^t (1-P)^{n-t}.$$

Show that the *undetected error probability* is given by

$$W_C\left(1-P, \frac{P}{q-1}\right) - (1-P)^n,$$

where $W_C(X, Y)$ is the homogeneous weightenumerator of C.

3 Decoding cyclic codes with Gröbner bases

Let C be an $[n,k,d]$ cyclic code with generator polynomial $g(X)$ and defining set $J = \{j_1, \ldots, j_r\}$. Let $\mathbb{F}_{q^e}$ an extension of $\mathbb{F}_q$ that contains all the zeros of $g(X)$. Let $\alpha \in \mathbb{F}_{q^e}$ be a primitive n-th root of unity. Then a parity check matrix of C is

$$H = \begin{pmatrix} 1 & \alpha^{j_1} & \alpha^{2j_1} & \cdots & \alpha^{(n-1)j_1} \\ 1 & \alpha^{j_2} & \alpha^{2j_2} & \cdots & \alpha^{(n-1)j_2} \\ 1 & \alpha^{j_3} & \alpha^{2j_3} & \cdots & \alpha^{(n-1)j_3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{j_r} & \alpha^{2j_r} & \cdots & \alpha^{(n-1)j_r} \end{pmatrix}.$$

Now let $\mathbf{e} = e(X)$ be an error vector of a received word $\mathbf{y} = y(X)$. Then $\mathbf{s} = \mathbf{y}H^T = \mathbf{e}H^T$ and

$$s_i = y(\alpha^{j_i}) = e(\alpha^{j_i})$$

is the ith component of $\mathbf{s}$ for $i = 1, \ldots, r$. It is more convenient to consider the extension $\hat{H}$ of the matrix H, where $\hat{H}$ is the $n \times n$ matrix with ith row

$$(1 \ \alpha^i \ \alpha^{2i} \ \cdots \ \alpha^{(n-1)i})$$

for $i = 1, \ldots, n$. Define $\hat{\mathbf{s}} = \mathbf{e}\hat{H}^T$. The jth component of $\hat{\mathbf{s}}$ is

$$\hat{s}_j = e(\alpha^j) = \sum_{i=0}^{n-1} e_i \alpha^{ij}$$

for $j = 1, \ldots, n$. If $j \in J(C)$, then $\hat{s}_j = e(\alpha^j) = y(\alpha^j)$, so these syndromes are *known*.

From now on $\hat{s}_j$ will be denoted by s_j. Notice that the old s_i is now denoted by s_{j_i}.

Let $\mathbf{e} = e(X)$ be an error vector with error positions $i_1, i_2, \ldots, i_t$ and error values $e_{i_1}, e_{i_2}, \ldots, e_{i_t}$. Then the known syndromes will be

$$s_j = \sum_{m=1}^{t} e_{i_m} (\alpha^{i_m})^j, \qquad j \in J(C).$$

Consider the following system of equations over $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$:

$$\mathcal{S}(\mathbf{s}, v) = \begin{cases} \sum_{m=1}^{v} Y_m X_m^j = s_j & \text{for } j \in J \\ Y_m^q = Y_m & \text{for } m = 1, \ldots, v \\ X_m^n = 1 & \text{for } m = 1, \ldots, v. \end{cases}$$

Conclude that $X_m = \alpha^{i_m}$ and $Y_m = e_{i_m}$ for $m = 1, \ldots, t$ is a solution of $\mathcal{S}(\mathbf{s}, t)$.

Exercise 3.1 Show that the equation $\sum_{m=1}^{v} Y_m X_m^{jq} = s_{jq}$ is a consequence of $\mathcal{S}(\mathbf{s}, v)$ for all $j \in J$.

Example 3.2 Let $J = \{1, 2\}$. If C is a cyclic code with defining set J, then its minimum distance is at least 3 by the BCH bound. So one can correct at least 1 error. The equations

$$\begin{cases} Y_1 X_1 &= s_1 \\ Y_1 X_1^2 &= s_2 \end{cases}$$

imply that the error position is $x_1 = s_2/s_1$ if there is exactly one error. If moreover $q = 2$, then $s_2 = s_1^2$, so $x_1 = s_1$.

We have the following.

Proposition 3.3 *Suppose that t errors occurred and $t \leq (d-1)/2$. Then the system $\mathcal{S}(\mathbf{s}, v)$ over $\mathbb{F}_{q^e}$ has no solution when $v < t$, and a unique solution up to permutations, corresponding to the error vector of lowest weight that satisfies the syndrome equations when $v = t$. The X_i of the solution are the error-locators and the Y_i the corresponding error values. If $v > t$, then for every j the system has a solution with $X_1 = \alpha^j$.*

Exercise 3.4 Prove Proposition 3.3 using Proposition 2.1.

The system $\mathcal{S}(\mathbf{s}, v)$ defines an ideal in the ring $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$. By abuse of notation we denote this ideal also by $\mathcal{S}(\mathbf{s}, v)$. The zero set of this ideal gives the error vector that occurred during the transmission. Gröbner basis techniques can be used to find the solutions of the equations.

Let $\prec_L$ be the lexicographic order with $Z_1 \prec_L Z_2 \prec_L \cdots \prec_L Z_w$. Then $\prec_L$ is an *elimination order*, that is to say it satisfies the following property.

Proposition 3.5 *Let I be an ideal in $\mathbb{F}[Z_1, Z_2, \ldots, Z_w]$. Let $\mathcal{G}$ be a Gröbner basis of I with respect $\prec_L$. Then $\mathcal{G} \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$ is a Gröbner basis of $I \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$.*

Let I be an ideal in $\mathbb{F}[Z_1, Z_2, \ldots, Z_w]$ with finitely many zeros over $\bar{\mathbb{F}}$ which are all defined over $\mathbb{F}$. Let V be the zero set in $\mathbb{F}^w$ of the ideal I. Then the zero set of $I \cap \mathbb{F}[Z_1, Z_2, \ldots, Z_i]$ is equal to the projection of V on the first i coordinates. This fact and Proposition 3.5 have a direct application to our problem of finding the solutions to system $\mathcal{S}(\mathbf{s}, v)$. Indeed, if $(x_1, \ldots, x_v)$ is the X-part of a solution to $\mathcal{S}(\mathbf{s}, v)$, then also any permutation of the x_i will be a solution (apply the same permutation to the Y-part of the solution). Hence every error-locator will appear as the first coordinate of a solution to $\mathcal{S}(\mathbf{s}, v)$. Thus we have sketched the proof of the following.

Proposition 3.6 *Suppose that t errors occurred and $t \leq (d-1)/2$. Let $g(X_1)$ be the monic generator of the ideal $\mathcal{S}(\mathbf{s}, t) \cap \mathbb{F}_{q^e}[X_1]$. Then the zeros of g are the error-locators.*

Before giving the final algorithm for the decoding, we must worry about one more thing: we assumed we knew how many errors occurred (the v occurring in

system $\mathcal{S}(\mathbf{s}, v)$). Now note that the work required to solve the system $\mathcal{S}(\mathbf{s}, v)$ for large v is much more than for small v, and remark that in general words with many errors occur less often than words with few or no errors. The following theorem leads the way to an algorithm that implements this idea.

Theorem 3.7 *Suppose t errors occurred and $t \leq (d-1)/2$. Denote the monic error-locator polynomial by $l(X_1)$, that is to say $l(x) = 0$ if and only if x is an error-locator. Let $g(X_1)$ be the monic generator of the ideal $\mathcal{S}(\mathbf{s}, v) \cap \mathbb{F}_{q^e}[X_1]$, with $\mathcal{S}(\mathbf{s}, v)$ the ideal in $\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v]$. Then*

$$g(X_1) = \begin{cases} 1 & \text{if } v < t \\ l(X_1) & \text{if } v = t \\ X_1^n - 1 & \text{if } v > t \end{cases}$$

Exercise 3.8 Show that in Proposition 3.3 and Theorem 3.7 it is allowed to replace the assumption "$t \leq (d-1)/2$" by the weaker statement "the received word has a unique closest codeword".

Exercise 3.9 Let $\mathcal{S}'(\mathbf{s}, v)$ be the system of equations which is obtained by replacing the equation $Y_m^q = Y_m$ in $\mathcal{S}(\mathbf{s}, v)$ by $Y_m^{q-1} = 1$ for all $m = 1, \ldots, v$. So the variables Y_m disappear if $q = 2$. How should Proposition 3.3 and Theorem 3.7 be restated for $\mathcal{S}'(\mathbf{s}, v)$?

We are now ready to state the algorithm to decode cyclic codes.

Algorithm 3.10
input($\mathbf{y}$);
$\mathbf{s} := \mathbf{y}H^T$;
if $s_j = 0$ for all $j \in J$
then output($\mathbf{y}$); stop; {no errors occurred}
else $v := 1$;
  $\mathcal{G} := \{1\}$;
  while $1 \in \mathcal{G}$ do
    $\mathcal{S} := \{\sum_{m=1}^{v} Y_m X_m^j - s_j, j \in J\} \cup \{Y_m^q - Y_m, X_m^n - 1,\ m = 1, \ldots, v\}$;
    $\mathcal{G} :=$ Gröbner($\mathcal{S}$);
    $v := v + 1$;
  od;
  {$1 \notin \mathcal{G}$ so there are solutions}
  $g(X_1) :=$ the unique element of $\mathcal{G} \cap \mathbb{F}_{q^e}[X_1]$};
  if $\deg(g(X_1)) > v$
  then output(?); stop { too many errors }
  else error-locators := {zeros of $g(Z_1)$}
    find error vector $\mathbf{e}$ by solving the linear equations
    as in Proposition 2.1
    output($\mathbf{y} - \mathbf{e}$)

We will treat an example in the project on the Golay codes.

3.1 One-step decoding of cyclic codes

In the system of equations $\mathcal{S}(\mathbf{s}, v)$ the syndromes s_j are considered to be known constants. In this section we treat the syndromes as variables and consider the corresponding system of equations

$$\mathcal{S}(v) = \begin{cases} \sum_{m=1}^{v} Y_m X_m^j &= S_j & \text{for } j \in J \\ Y_m^q &= Y_m & \text{for } m = 1, \ldots, v \\ X_m^n &= 1 & \text{for } m = 1, \ldots, v. \end{cases}$$

to define an ideal in the ring

$$\mathbb{F}_{q^e}[X_1, \ldots, X_v, Y_1, \ldots, Y_v, S_j, j \in J].$$

This of course has the advantage that we have to solve these equations only once, and that this can be done before we start to use the code. This is called the *preprocessing* of the decoding algorithm. In the actual running of the algorithm the values of the syndromes s_j of a received word are substituted in the variables S_j for $j \in J$.

Exercise 3.11 Let $\prec$ be a reduction order on the monomials $X_1, \ldots, X_v, Y_1, \ldots, Y_v$ and $S_j, j \in J$ such that the variables $S_j, j \in J$ are larger than $X_1, \ldots, X_v$ and $Y_1, \ldots, Y_v$. Show that $\mathcal{S}(v)$ is a Gröbner basis with respect to $\prec$.

The exercise gives the impression that we are done. But we have to elimination the variables $X_2, \ldots, X_v$ and $Y_1, \ldots, Y_v$. Therefore the variables X_1, $S_j, j \in J$ need to be smaller than $X_2, \ldots, X_v$, $Y_1, \ldots, Y_v$.

As an example, we have applied one-step decoding to binary cyclic codes with defining sets $\{1, 3\}$, $\{1, 3, 5\}$ and $\{1, 3, 5, 7\}$, respectively. Remark that the complete defining sets contain $\{1, 2, 3, 4\}$, $\{1, 2, 3, 4, 5, 6\}$ and $\{1, \ldots, 8\}$, respectively. From the BCH-bound we know that these codes can correct $2, 3$ and 4 errors, respectively. The Gröbner basis is computed with a lexicographic order in a way such that the basis contains a polynomial in X_1 and the syndrome-variables S_j. We consider binary codes. Thus the error values are always 1. Therefore we delete the variables Y_i in the equations. The equations of the form $X_m^n = 1$ are also left out. So the number of solutions is not finite anymore. The results are as follows.

Example 3.12 $q = 2$, $\{1, 3\} \subseteq J(C)$.

$$\mathcal{S} = \begin{cases} X_1 + X_2 - S_1 = 0 \\ X_1^3 + X_2^3 - S_3 = 0 \end{cases}$$

Order: $X_2 > X_1 > S_3 > S_1$
Error-locator polynomial with $X = X_1$:

$$S_1 X^2 + S_1^2 X + (S_1^3 + S_3).$$

Example 3.13 $q = 2$, $\{1, 3, 5\} \subseteq J(C)$.

$$\mathcal{S} = \left\{ \begin{array}{ccccccccc} X_1 & + & X_2 & + & X_3 & - & S_1 & = & 0 \\ X_1^3 & + & X_2^3 & + & X_3^3 & - & S_3 & = & 0 \\ X_1^5 & + & X_2^5 & + & X_3^5 & - & S_5 & = & 0 \end{array} \right.$$

Order: $X_3 > X_2 > X_1 > S_5 > S_3 > S_1$
Error-locator polynomial:

$$(S_3 + S_1^3)X^3 + (S_3S_1 + S_1^4)X^2 + (S_5 + S_3S_1^2)X + (S_5S_1 + S_3^2 + S_3S_1^3 + S_1^6).$$

Example 3.14 $q = 2$, $\{1, 3, 5, 7\} \subseteq J(C)$.

$$\mathcal{S} = \left\{ \begin{array}{ccccccccccc} X_1 & + & X_2 & + & X_3 & + & X_4 & - & S_1 & = & 0 \\ X_1^3 & + & X_2^3 & + & X_3^3 & + & X_4^3 & - & S_3 & = & 0 \\ X_1^5 & + & X_2^5 & + & X_3^5 & + & X_4^5 & - & S_5 & = & 0 \\ X_1^7 & + & X_2^7 & + & X_3^7 & + & X_4^7 & - & S_7 & = & 0 \end{array} \right.$$

Order: $X_4 > X_3 > X_2 > X_1 > S_7 > S_5 > S_3 > S_1$
Error-locator polynomial:

$$\begin{array}{c} (S_1^6 + S_3^2 + S_5S_1 + S_3S_1^3)X^4 + (S_5S_1^2 + S_3^2S_1 + S_3S_1^4 + S_1^7)X^3 + \\ (S_7S_1 + S_5S_3 + S_3S_1^5 + S_1^8)X^2 + (S_7S_1^2 + S_5S_1^4 + S_3^3 + S_3S_1^6)X + \\ (S_7S_3 + S_7S_1^3 + S_5^2 + S_5S_3S_1^2 + S_5S_1^5 + S_3^3S_1 + S_3S_1^7 + S_1^{10}). \end{array}$$

Example 3.15 The error-locator polynomial for the 6-error correcting binary BCH code took four hours using Axiom. The coefficient of X^i has 20, 20, 22, 22, 20, 24 and 46 terms for $i = 6, 5, \ldots, 1$ and 0, respectively.

Exercise 3.16 Give S_i weighted degree i and let $\mathrm{wdeg}(X) = 1$. Notice that in the above examples the error-locator polynomial is homogeneous of total weighted degree $\binom{t+1}{2}$ if the BCH bound is $2t + 1$. Show that this is always the case.

Looking at the formulas for the $2, 3$ and 4 error-correcting BCH codes one gets the impression that the number of terms grows exponentially (we do not know whether this is a fact). Thus specializing the values for the syndromes still would not give a decoding algorithm of polynomial complexity.

It is a priori not clear that substituting values for the syndromes in the variables after elimination gives the same answer as the original method with the syndromes as constants.

To make this point clear we introduce some notation. Let $\mathcal{G}$ be a subset of the polynomial ring in the variables $S_j, j \in J$, $X_1, \ldots, X_v$ and more. Then $\mathcal{G}_1$ is the subset of $\mathcal{G}$ of polynomials in the variables $S_j, j \in J$ and X_1 only. Let $\mathbf{s} = (s_j, j \in J)$ be a vector with coordinates in $\mathbb{F}_q$. Then $\mathcal{G}_1(\mathbf{s})$ is the set obtained from $\mathcal{G}_1$ by substituting the value s_j in S_j for all elements of $\mathcal{G}_1$ and $j \in J$.

Let $\prec_E$ be an elimination order on the monomials $X_1, \ldots, X_v, Y_1, \ldots, Y_v$ and $S_j, j \in J$ with the variables $X_1, \ldots, X_v$ and $Y_1, \ldots, Y_v$ larger than $S_j, j \in J$. That the one-step method works is stated as a fact in the following

Theorem 3.17 *Let $\mathcal{G}$ be a Gröbner basis of $\mathcal{S}(t)$ with respect to $\prec_E$. Let $\mathbf{y}$ be a received word such that t errors occurred. Let $\mathbf{s}$ be its syndrome. Assume that the closest codeword to $\mathbf{y}$ is unique. Then $\mathcal{G}_1$ is the Gröbner basis of $\mathcal{S}(t) \cap \mathbb{F}_q[X_1, S_j, j \in J]$ and $\mathcal{G}_1(\mathbf{s})$ generates the ideal of the error-locator polynomial.*

The proof relies on the fact that $\mathcal{S}(t)$ has a finite number of solutions.

The set $\mathcal{G}_1(\mathbf{s})$ consist of polynomials in the variable X_1. In examples the greatets common divisor of $\mathcal{G}_1(\mathbf{s})$ is among its elements, that is to say that $\mathcal{G}_1(\mathbf{s})$ is a (nonreduced) Gröbner bases. It is conjectured that this is always the case.

4 The key equation

Let C be a cyclic code of length n such that $\{1, 2, \ldots, \delta - 1\} \subset J(C)$. From the BCH bound we see that the minimum distance of C is at least δ. In this section we will give a decoding algorithm for such a code which has an efficient implementation and is used in practice. A drawback of the algorithm is that it only corrects errors of weight at most $(\delta - 1)/2$, whereas the true minimum distance can be larger than δ. An example of this phenomenon will be treated in the project on the Golay codes.

The algorithms in this section work for cyclic codes that have any $\delta - 1$ consecutive elements in their complete defining set. We leave it to the reader to make the necessary adjustments in the case where these elements are not $\{1, 2, \ldots, \delta - 1\}$.

Let α be a primitive n-th root of unity. Let $\mathbf{c} = c(X) \in C$ be the transmitted codeword that is received as $\mathbf{y} = y(X) = c(X) + e(X)$, with $w = wt(\mathbf{e}) \leq (\delta - 1)/2$. The support of $\mathbf{e}$ will be denoted by I. We then can compute the syndromes

$$s_i = A_i = e(\alpha^i) = y(\alpha^i) \qquad \text{for } i \in J(C),$$

where the A_i are the coefficients of the MS polynomial of $e(X)$, see Section **??**. Since $\{1, 2, \ldots, \delta - 1\} \subseteq J(C)$ and $2w \leq \delta - 1$ we know all $A_1, A_2, \ldots, A_{2w}$. Write σ_i for the i-th symmetric function of the error positions and form the following set of generalized Newton identities, see Proposition **??**

$$\left\{ \begin{array}{ccccccccc} A_{v+1} & + & \sigma_1 A_v & + & \cdots & + & \sigma_v A_1 & = & 0 \\ A_{v+2} & + & \sigma_1 A_{v+1} & + & \cdots & + & \sigma_v A_2 & = & 0 \\ & & & & \vdots & & & \vdots & \vdots \\ A_{2v} & + & \sigma_1 A_{2v-1} & + & \cdots & + & \sigma_v A_v & = & 0. \end{array} \right. \tag{1}$$

From the system with $v = w$ we have to find the σ_i. After we have done this, we can find the polynomial

$$\sigma(Z) = 1 + \sigma_1 Z + \sigma_2 Z^2 + \cdots + \sigma_w Z^w$$

which has as its zeros the reciprocals of the error locations. Finding the zeros of this polynomial is an easy task. We return to the problem of finding the coefficients σ_i.

Exercise 4.1 Consider the system of equations (1) as linear in the unknown $\sigma_1, \ldots, \sigma_w$ with coefficients in $\mathbb{F}_q(A_1, \ldots, A_w)$ the field of rational functions in $A_1, \ldots, A_w$, which are treated now as variables. Then

$$\sigma_i = \frac{\Delta_i}{\Delta_0}$$

where Δ_i is the determinant of a certain $w \times w$ matrix according to Cramers rule. Then the Δ_i are polynomials in the A_i. Conclude that

$$\Delta_0 X^w + \Delta_1 X^{w-1} + \cdots + \Delta_w$$

is a closed form of the *generic* error-locator polynomial.

Substitute $A_{2i+1} = S_{2i+1}$ and $A_{2i} = S_i^2$ and compare the result with Examples 3.12, 3.13 and 3.14.

Exercise 4.2 Show that the matrix $(A_{i+j-1} | 1 \leq i, j \leq v)$ is nonsingular if and only if $v = w$, the number of errors. Hint: try to write the matrix as a triple product of matrices of known rank as done in Exercise **??**.

The algorithm of *Arimoto-Peterson-Gorenstein-Zierler* (APGZ) solves the systems of linear equations (1) for $v = 1, \ldots, w$ by Gaussian elimination.

Exercise 4.3 What is the complexity of the algorithm of APGZ ?

Write

$$S(Z) = \sum_{i=1}^{\delta-1} A_i Z^{i-1},$$

then an alternative way of formulating (1) is that there exist polynomials $q(Z)$ and $r(Z)$ such that

$$\sigma(Z)S(Z) = q(Z)Z^{\delta-1} + r(Z), \quad \deg(r(Z)) \leq w - 1,$$

or that there exists a polynomial $\omega(Z)$ of degree at most $w - 1$ such that

$$\omega(Z) \equiv \sigma(Z)S(Z) \bmod Z^{\delta-1}. \tag{2}$$

This is called the *key equation*.

Exercise 4.4 Check that

$$\omega(Z) = \sum_{i \in I} e_i \alpha^i \prod_{j \in I \setminus \{i\}} (1 - \alpha^j Z),$$

by rewriting $\omega(Z)/\sigma(Z) \bmod Z^{\delta-1}$.

Exercise 4.5 Let $\sigma'(Z)$ be the formal derivative of $\sigma(Z)$. Show *Forney's formula* for the error values:

$$e_i = -\frac{\omega(\alpha^{-i})}{\sigma'(\alpha^{-i})}$$

for all error positions i. The polynomial $\omega(Z)$ is called the *error evaluator polynomial*.

We will discuss two algorithms that are faster than the one proposed in Exercise 4.3.

4.1 The algorithms of Euclid and Sugiyama

The *Euclidean algorithm* is a well known algorithm that can be used to compute the *greatest common divisor* of two univariate polynomials. We assume that the reader is familiar with this algorithm. In order to fix a notation, suppose we want to compute $\gcd(r_{-1}(Z), r_0(Z))$. Then the Euclidean algorithm proceeds as follows.

$$\begin{array}{rcllcll}
r_{-1}(Z) & = & q_1(Z)r_0(Z) & + & r_1(Z), & \deg(r_1) & < & \deg(r_0) \\
r_0(Z) & = & q_2(Z)r_1(Z) & + & r_2(Z), & \deg(r_2) & < & \deg(r_1) \\
 & \vdots & & \vdots & & & \vdots & \\
r_{j-2}(Z) & = & q_j(Z)r_{j-1}(Z) & + & r_j(Z), & \deg(r_j) & < & \deg(r_{j-1}) \\
r_{j-1}(Z) & = & q_{j+1}(Z)r_j(Z). & & & & &
\end{array}$$

From this we can conclude that $\gcd(r_{-1}(Z), r_0(Z)) = r_j(Z)$. The key equation can be solved with the algorithm of *Sugiyama* in the following way.

Algorithm 4.6 Set

$$r_{-1}(Z) = Z^{\delta-1}, \quad r_0(Z) = S(Z), \quad U_{-1}(Z) = 0, \quad U_0(Z) = 1,$$

and proceed with the algorithm of Sugiyama until an $r_k(Z)$ is reached such that

$$\deg(r_{k-1}(Z)) \geq \frac{1}{2}(\delta - 1) \qquad \text{and} \qquad \deg(r_k(Z)) \leq \frac{1}{2}(\delta - 3),$$

also updating

$$U_i(Z) = q_i(Z)U_{i-1}(Z) + U_{i-2}(Z).$$

Then the error-locator and evaluator polynomial are

$$\begin{array}{rcl}
\sigma(Z) & = & \epsilon U_k(Z) \\
\omega(Z) & = & (-1)^k \epsilon r_k(Z)
\end{array}$$

where ϵ is chosen such that $\sigma_0 = \sigma(0) = 1$.

Exercise 4.7 Show that the $\sigma(Z)$ and $\omega(Z)$ resulting from the algorithm satisfy
1. $\omega(Z) = \sigma(Z)S(Z) \bmod Z^{\delta-1}$
2. $\deg(\sigma(Z)) \leq \frac{1}{2}(\delta - 1)$
3. $\deg(\omega(Z)) \leq \frac{1}{2}(\delta - 3)$.

We will not prove the correctness of the algorithm. The algorithm of Sugiyama is used to decode in the project on Golay codes.

4.2 The algorithm of Berlekamp-Massey

The algorithm of *Berlekamp-Massey* is an example of *dynamic programming* The algorithm is iterative, and in the j-th iteration the following problem is solved: find the pair $(\sigma_j(Z), \omega_j(Z))$ such that

1. $\sigma_j(0) = 1$
2. $\sigma_j(Z)S(Z) = \omega(Z) \bmod Z^j$
3. $d_j = \max\{\deg(\sigma_j), \deg(\omega_j) + 1\}$ is minimal.

It is rather technical to work out what has to be updated when proceeding to the next iteration. After the algorithm we will give a few remarks on the variables that are used.

Algorithm 4.8

1. $j = 0; \quad \sigma_0 = -\omega'_0 = 1; \quad \sigma'_0 = \omega_0 = 0; \quad d_0 = 0; \quad \Delta = 1.$
2. $\Delta_j =$ coefficient of Z^j in $\sigma_j(Z)S(Z) - \omega_j(Z)$.
3. If $\Delta_j = 0$ then
   $d_{j+1} := d_j; \quad \sigma_{j+1} := \sigma_j; \quad \omega_{j+1} := \omega_j;$
   $\sigma'_{j+1} := Z\sigma'_j; \quad \omega'_{j+1} := Z\omega'_j$
4. If $\Delta_j \neq 0$ and $2d_j > j$ then
   $d_{j+1} := d_j; \quad \sigma_{j+1} := \sigma_j - \Delta_j\Delta^{-1}\sigma'_j; \quad \omega_{j+1} := \omega_j - \Delta_j\Delta^{-1}\omega'_j;$
   $\sigma'_{j+1} := Z\sigma'_j; \quad \omega'_{j+1} := Z\omega'_j$
5. If $\Delta_j \neq 0$ and $2d_j \leq j$ then
   $d_{j+1} := j + 1 - d_j; \quad \sigma_{j+1} := \sigma_j - \Delta_j\Delta^{-1}\sigma'_j; \quad \omega_{j+1} := \omega_j - \Delta_j\Delta^{-1}\omega'_j;$
   $\Delta := \Delta_j; \quad \sigma'_{j+1} := Z\sigma_j; \quad \omega'_{j+1} := Z\omega_j$
6. If S_{j+1} is known then $j := j + 1$ and go to step 2; otherwise stop.

In the algorithm, the variables σ'_j and ω'_j are auxiliary. The Δ_j measures how far a solution to the j-th iteration is from being a solution to the $(j + 1)$-th iteration. If $\Delta_j = 0$, the solution passes to the next iteration. If $\Delta_j \neq 0$, then the solution must be adjusted in such a way that the resulting $d_{j+1} = \max\{\deg(\sigma_{j+1}), \deg(\omega_{j+1}) + 1\}$ is minimal. In order to minimize this degree, the two cases 4 and 5 have to be distinguished.

Notice that in the algorithm of Sugiyama the degree of the polynomial decreases during the algorithm, whereas in the Berlekamp-Massey algorithm the degree of the polynomial increases. This is an advantage, since error vectors of small weight are more likely to occur than those of high weight.

5 Gröbner bases and arbitrary linear codes

We will start by a general construction of a code, and later show that in fact this gives all linear codes.

Let $\mathcal{P} = \{P_1, P_2, \ldots, P_n\} \subseteq \mathbb{F}_q^m$ be the set of zeros of a set of polynomials $\mathcal{G} = \{G_1, \ldots, G_u\}$ in $\mathbb{F}_q[X_1, X_2, \ldots, X_m]$. Let I be the ideal generated by $\mathcal{G}$. Define the ring R as

$$R = \mathbb{F}_q[X_1, \ldots, X_m]/I,$$

Let $F_1, F_2, \ldots, F_r$ be a basis of the $\mathbb{F}_q$-vector subspace L of R. Consider the evaluation map

$$ev_{\mathcal{P}} : L \longrightarrow \mathbb{F}_q^n.$$

The codes we consider here are

$$C = Im(ev_{\mathcal{P}})^{\perp}.$$

Thus $H = (F_i(P_j))$ is a parity check matrix of C. After introducing this algebraic setting, it is clear how Gröbner bases can be used for the decoding problem. Let d be the minimum distance of C. Suppose we receive a vector $\mathbf{y}$ and we want to decode t errors, with $t \leq \lfloor (d-1)/2 \rfloor$. Then, after computing the syndromes

$$s_i = \sum_{j=1}^{n} y_j F_i(P_j)$$

we can form the following system of equations $\mathcal{S}(\mathbf{s}, v)$

$$\begin{cases} \sum_{j=1}^{v} Y_j F_i(X_{1j}, \ldots, X_{mj}) = s_i & \text{for } i = 1, \ldots, r \\ G_i(X_{1j}, \ldots, X_{mj}) = 0 & \text{for } j = 1, \ldots, v \text{ and } i = 1, \ldots, u \\ Y_j^q = Y_j & \text{for } j = 1, \ldots, t, \end{cases}$$

with variables $X_{1j}, \ldots, X_{mj}$ for the coordinates of a copy of $\mathbb{F}_q^m$ for all $j = 1, \ldots, v$, and the variables $Y_1, \ldots, Y_v$ for the error values in $\mathbb{F}_q$. As in the case with cyclic codes, we see that if $(\mathbf{x}_1, \ldots, \mathbf{x}_v, \mathbf{y}_1, \ldots, \mathbf{y}_v)$, with $\mathbf{x}_j = (x_{1j}, \ldots, x_{mj})$, is a solution to $\mathcal{S}(\mathbf{s}, v)$, then so is

$$(\mathbf{x}_{\pi(1)}, \ldots, \mathbf{x}_{\pi(v)}, \mathbf{y}_{\pi(1)}, \ldots, \mathbf{y}_{\pi(v)}),$$

for any permutation π of $\{1, \ldots, v\}$. Hence a Gröbner basis $\mathcal{G}$ for the ideal $\mathcal{S}(\mathbf{s}, t)$ with respect to the lexicographic order with

$$Y_m > \cdots > Y_1 > X_{mv} > \cdots > X_{1v} > \cdots > X_{m1} > \cdots > X_{11}$$

will have elements that are polynomials in $X_{m1}, \ldots, X_{11}$ only. These elements generate the ideal $\mathcal{S}(\mathbf{s}, v) \cap \mathbb{F}_q[X_{11}, \ldots, X_{m1}]$. This intersection has no solution when $v < t$. If $v = t$, then the intersection is the *error-locator ideal*, that means that it has the set of error postions as zero set in $\mathbb{F}_q^m$. The error values can be found as before for cyclic codes with Proposition 2.1.

Example 5.1 Let C be an $[n, k, d]$ linear code with $r \times n$ parity check matrix H, where $r = n - k$. Consider the n columns of H as points $P_1, \ldots, P_n \in \mathbb{F}_q^r$ and set $\mathcal{P} = \{P_1, \ldots, P_n\}$. Then $\mathcal{P}$ is finite, so it is an algebraic set:

$$\mathcal{P} = V(I, \mathbb{F}_q), \quad I = \{G \in \mathbb{F}_q[X_1, \ldots, X_r] \mid G(P_1) = \cdots = G(P_n) = 0\}.$$

If we take as an r-dimensional vector space L the coordinate functions

$$L = \langle X_1, \ldots, X_r \rangle,$$

then it is clear that $C = Im(ev_{\mathcal{P}})^{\perp}$.

Exercise 5.2 Describe the Hamming code by the above method. What is the vanishing ideal in $\mathbb{F}_2[X_1, X_2, X_3]$ if one applies the above procedure to the Hamming code ?

Although in principle every linear code could be described and decoded in this way, the large number of variables will make it very impractical. The following exercise relaxes the number of variables a bit.

Exercise 5.3 Let C be an q-ary $[n, k, d]$ code. Let $r = n - k$. Let $H = (h_{ij})$ be a parity check matrix of C. Let m be a positive integer such that $q^m \geq n$. Show that there exist n distinct points $P_1, \ldots, P_n$ in $\mathbb{F}_q^m$ and polynomials $F_1, \ldots, F_r$ in $\mathbb{F}_q[X_1, \ldots, X_m]$ such that $F_i(P_j) = h_{ij}$.

Example 5.4 Let C be a cyclic code with defining set J. Instead of treating this as an arbitrary linear code as in the previous example, it is better to use the structure of the parity check matrix, as follows. Take $\mathcal{P} = \{1, \alpha, \ldots, \alpha^{n-1}\} \subseteq \mathbb{F}_{q^e}$, the set of n-th roots of unity. Hence

$$I = (X^n - 1)\mathbb{F}_{q^e}[X].$$

If we take for L the vector space

$$L = \langle X^j \mid j \in J \rangle$$

over $\mathbb{F}_{q^e}$, it is clear that C is a code as described above, and that the system $\mathcal{S}(\mathbf{s}, t)$ we have to solve, equals the one we already met in Section 3.

One-step decoding is done in the same way as for cyclic codes by treating the s_j as variables and the corresponding Theorem 3.17 holds.

The same methods applies for getting the minimal weight codewords of a linear code.

6 Notes

That the general decoding problem is hard can be made precise in terms of complexity theory. See [3, 5].

Formulas for the probability of a decoding error or failure for several decoders and the relation with the weightenumerator is given in [6, 24]. Some history of the origins of decoding algorithms can be found in [2].

The original idea of one-step decoding is from [9, 10] and [30]. See also [38].

The method to decode cyclic codes up to half the actual minimum distance using Gröbner basis is from [11, 12, 13]. The extension to arbitrary linear codes is from [17, 18]. Theorem 3.17 is from [17, 18, 25]. The conjecture concerning $\mathcal{G}_1(\mathbf{s})$ is from [25]. The remark in Exercise 3.11 is from [25]. In this paper the work of [15] is used to transform a Gröbner basis of a zero dimensional ideal with respect to one reduction order into a Gröbner basis with respect to another one. The decoding is considerably faster by this method as is seen in the Project on the Golay code. Decoding constacyclic codes in Lee metric by the use of Gröbner bases is explained in [28]

A more efficient way to decode cyclic codes is by solving the key equation [1, 4, 20, 27, 31, 37]. The formula for the error values is from [19].

The material of Section 4 is from [6, 7, 26, 32]. This formulation of the Berlekamp-Massey algorithm is from [14].

For Reed-Solomon codes a hybrid of the algorithm of Berlekamp-Massey and Gröbner bases techniques is given in [39, 40, 41] to get all closest codewords of a received word.

Decoding arbitrary linear codes with Gröbner bases is from [17, 18]. This method can also be applied to get all minimal weight codewords as explained for cyclic codes in the previous chapter.

There are many papers on decoding algebraic geometry codes and we refer to the literature [8, 16, 21, 22, 23, 29].

The Berlekamp-Massey algorithm is generalized to polynomials in several variables by [34, 35, 36]. This theory has very much to do with the theory of Gröbner bases, but it solves another problem than Buchbergers algorithm. The algorithm is implemented in the decoding of algebraic geometry codes. See the literature cited above and [33]. The name *footprint* for the Δ-set is from [8].

References

[1] S. Arimoto, "Encoding and decoding of p-ary group codes and the correction system," (in Japanese) *Inform. Processing in Japan,* vol. 2, pp. 320-325, Nov. 1961.

[2] A. Barg, "At the dawn of the theory of codes," *Math. Intelligencer,* vol. 15, No. 1, pp. 20-27, 1993.

[3] A. Barg, "Complexity issues in coding theory," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[4] E.R. Berlekamp, *Algebraic coding theory,* Aegon Park Press, Laguna Hills CA, 1984.

[5] E.R. Berlekamp, R.J. McEliece and H.C.A. van Tilborg, "On the inherent intractibility of certain coding problems," *IEEE Trans. Inform. Theory,* vol. IT-24, pp. 384-386, 1978.

[6] R.E. Blahut, *Theory and practice of error control codes*, Addison-Wesley, Reading 1983.

[7] R.E. Blahut, *Fast algorithms for digital signal processing*, Addison-Wesley, Reading 1985.

[8] R.E. Blahut, *Introduction to algebraic coding,* book in prepartation.

[9] A. Brinton Cooper III, "Direct solution of BCH decoding equations," *Communication, Control and Signal Processing,* Elsevier Sc. Publ., pp. 281-286, 1990.

[10] A. Brinton Cooper III, "Finding BCH error locator polynomials in one step," *Electronic Letters*, vol. 27 ,pp. 2090-2091, 1991.

[11] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "Algebraic decoding of cyclic codes: a polynomial point of view," *Contemporary Math.* vol. 168, pp. 15-22, 1994.

[12] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "Use of Gröbner bases to decode binary cyclic codes up to the true minimum distance," *IEEE Trans. Inform. Theory,* vol. IT-40, pp. 1654-1661, Sept. 1994.

[13] X. Chen, I.S. Reed, T. Helleseth and T.K. Truong, "General principles for the algebraic decoding of cyclic codes," *IEEE Trans. Inform. Theory,* vol. IT-40, pp. 1661-1663, Sept. 1994.

[14] J.L. Dornstetter, "On the equivalence of Berlekamp's and Euclid's algorithm," *IEEE Trans. Inform. Theory,* vol. IT-33, pp. 428-431, May 1987.

[15] J.C. Faugère, P. Gianni, D. Lazard and T. Mora, "Efficient computation of zero-dimensional Gröbner bases by a change of ordering," *Journ. Symb. Comp.,* vol. 16, pp. 329-344, 1993.

[16] G.-L. Feng and T.R.N. Rao, "Decoding of algebraic geometric codes up to the designed minimum distance," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 37-45, Jan. 1993.

[17] J. Fitzgerald, *Applications of Gröbner bases to linear codes,* Ph.D. Thesis, Louisiana State Un., Aug. 1996.

[18] J. Fitzgerald and R.F. Lax, "Decoding affine variety codes using Gröbner bases," to appear in *Designs, Codes and Cryptography.*

[19] G.D. Forney Jr., "On decoding BCH codes," *IEEE Trans. Inform. Theory,* vol. 11, pp. 549-557, 1965.

[20] D.C. Gorenstein and N. Zierler, "A class of error-correcting codes in p^m symbols," *Journ. SIAM,* vol. 9, pp. 207-214, 1961.

[21] T. Høholdt, J.H. van Lint and R. Pellikaan, "Algebraic geometry codes," to appear in *Handbook of Coding Theory,* (V.S. Pless, W.C. Huffman and R.A. Brualdi eds.), Elsevier.

[22] T. Høholdt and R. Pellikaan, "On decoding algebraic-geometric codes," *IEEE Trans. Inform. Theory,* vol. 41, pp. 1589-1614, Nov. 1995.

[23] J. Justesen, K.J. Larsen, H. Elbrønd Jensen, A. Havemose and T. Høholdt, "Construction and decoding of a class of algebraic geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 811-821, July 1989.

[24] T. Kløve and V.I. Korzhik, *Error detecting codes,* Kluwer Acad. Publ. , Dordrecht 1995.

[25] P. Loustaunau and E.V. York, "On the decoding of cyclic codes using Gröbner bases," preprint # 249, Dept. Math., Univ. of Notre Dame, Sept. 1996.

[26] F.J. MacWilliams and N.J.A. Sloane, *The theory of error-correcting codes,* North-Holland Math. Library vol. 16, North-Holland, Amsterdam 1977.

[27] J.L. Massey, "Shift-register synthesis and BCH decoding," *IEEE Trans. Inform. Theory* vol. IT-15, pp.122-127, January 1969.

[28] J. Maucher and R. Kötter, "Decoding constacyclic codes in Lee- and Mannheim metric by the use of Gröbner bases," preprint, August 1996.

[29] R. Pellikaan, "On the efficient decoding of algebraic-geometric codes," *Proceedings of Eurocode 92, CISM Courses and Lectures*, vol. 339, pp. 231-253, Springer-Verlag, Wien-New York, 1993.

[30] W.T. Penzhorn, "On the fast decoding of binary BCH codes," *Proc. IEEE Int. Symp. Inform. Theory,* San Antonio, pp. 103, Jan. 1993.

[31] W.W. Peterson, "Encoding and error-correction procedures for the Bose-Chauduri codes," *IRE Trans. Inform. Theory,* vol. IT-6, pp.459-470, 1960.

[32] W.W. Peterson and E.J. Weldon, *Error-correcting codes,* MIT Pres, Cambridge 1977.

[33] K. Saints and C. Heegard, "Algebraic-geometric codes and multidimensional cyclic codes: A unified theory and algorithms for decoding using Gröbner bases," *IEEE Trans. Inform. Theory,* vol. 41, pp. 1733-1751, Nov. 1995.

[34] S. Sakata, "On determining the independent point set for doubly periodic arrays and encoding two-dimensional cyclic codes and their duals," IEEE Trans. Inform. Theory, vol. IT-27, pp. 556-565, 1981.

[35] S. Sakata, "Finding a minimal set of linear recurring relations capable of generating a given finite two-dimensional array," *Journal of Symbolic Computation,* vol. 5, pp. 321-337, 1988.

[36] S. Sakata, "Extension of the Berlekamp-Massey algorithm to N dimensions," *Information and Computation,* vol. 84, pp. 207-239, 1990.

[37] Y. Sugiyama, M. Kasahara, S. Hirasawa and T. Namekawa, "A method for solving the key equation for decoding Goppa codes," *Information and Control,* vol. 27, pp. 87-99, 1975.

[38] H.-J. Weber, *Algebraische Algorithmen zur Decodierung zyklischer Codes,* Master's Thesis, Univ. Dortmund, March 1994.

[39] D-J. Xin, "New approach to decoding Reed-Solomon codes based on generalized rational interpolation," in *Proc. Sixth Swedish-Russian International Workshop Inform. Trans.* August 1993, pp. 219-223.

[40] D-J. Xin, "Homogeneous interpolation problem and key equation for decoding Reed-Solomon codes," *Science in China* (Series A), vol. 37 No. 11, Nov. 1994.

[41] D-J. Xin, " Extension of the Welch-Berlekamp theorem and universal stragegy of decoding algorithm beyond BCH bound," *Science in China* (Series A), vol. 38 No. 11, Nov. 1995.