On the decoding of algebraic-geometric codes

Tom Høholdt and Ruud Pellikaan *

Appeared in: IEEE Trans. Inform. Theory. IT-**41** (1995), 1589-1614.

Abstract This paper provides a survey of the existing literature on the decoding of algebraic-geometric codes. Definitions, theorems and cross references will be given. We show what has been done, discuss what still has to be done and pose some open problems. The following subjects are examined in a more or less historical order.

*Manuscript received January 27, 1995, revised July 31, 1995. The first author is from the Mathematical Institute, Technical Univerity of Denmark, Bldg 303, DK 2800, Lyngby, Denmark. The second author is from the Department of Mathematics and Computing Science, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands.

1 Introduction

The theory of algebraic-geometric codes is a fascinating topic where two extremes meet: the highly abstract and deep mathematics of modular curves and the very concrete problems in the engineering of information transmission.

Algebraic curves over finite fields were used by Goppa [33, 34, 35, 36, 37] to construct codes. Nowadays these codes are called *algebraic-geometric* or *geometric Goppa* codes. One type is obtained by evaluating (at points $P_1, \ldots, P_n$) certain rational functions having zeros and poles prescribed by a given divisor G; and the other type, dual to the first, is obtained by computing residues of differential forms. The formal sum $P_1 + \cdots + P_n$ is denoted by D. The former code is denoted by $C_L(D, G)$ and the latter by $C_\Omega(D, G)$. We denote the *Goppa designed minimum distance* of these codes by δ_Γ, which is equal to $n - \deg(G)$ for the functional codes and equal to $\deg(G) - 2\gamma + 2$ for the differential codes, where γ is the *genus* of the curve. If the genus is zero, the curve is just the projective line, the functional codes are extended Reed-Solomon codes, and the differential codes classical Goppa codes.

At the beginning of the eighties, Ihara [44], Tsfasman, Vlăduţ and Zink [107, 108] proved the existence of curves over finite fields with many rational points, giving asymptotically good codes.

A first attempt to decode algebraic-geometric codes was made by Driencourt [15] for codes on elliptic curves. This algorithm corrects $\lfloor (\delta_\Gamma - 1)/4 \rfloor$ errors. At the end of the eighties Justesen, Larsen, Elbrønd Jensen, Havemose and Høholdt [41, 48] found for algebraic-geometric codes on plane curves a generalization of the decoding algorithm of Arimoto [3] and Peterson [78] for RS codes. This algorithm finds an *error-locator polynomial* in two variables which has the error positions among its zeros. This was generalized to arbitrary curves by Skorobogatov and Vlăduţ [100]. Independently Krachkovskii proved the same result, but his work was only published as a conference paper in Russian [56]. In this way one gets the *basic* and *modified* decoding algorithm [100, 107]. The basic algorithm $\mathcal{A}(F)$ depends on a suitable divisor F and corrects $\lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor$ errors. The modified algorithm of Skorobogatov and Vlăduţ [100] depends on an increasing sequence of divisors $F_1 \leq \cdots \leq F_s$, was *extended* by Duursma [16, 18] and corrects $(\delta_\Gamma - 1)/2 - \sigma$ errors, where σ is the *Clifford defect*. Only in very special cases is σ equal to zero and for plane curves σ is roughly equal to $\gamma/4$. It was shown that the modified algorithm does not always correct up to half the minimum distance, but the number of error patterns where the algorithm does not decode the error is relatively small [16, 47, 74].

It was shown by Pellikaan [72] that a number of basic algorithms in parallel decodes $(\delta_\Gamma - 1)/2$ errors when a technical condition involving the *Jacobian* of the curve is fulfiled. The existence of a sequence of divisors $F_1, \ldots, F_s$ on a maximal curve is proved such that at least one of the basic algorithms $\mathcal{A}(F_i)$ corrects a received word. The argument was generalized by Vlăduţ [109] for almost all curves. The smallest value for s was studied by Carbonne, Henocq, Rotillon and Thiong-Ly [11, 42, 43]. This algorithm is not effective, that is to say the existence of these s divisors is guaranteed by a counting argument using the *zeta function* of the curve, but the actual construction of those was only achieved for *hyperelliptic* curves of genus at most 4 by Le Brigand [58, 59] and the *Klein curve* by Henocq and Rotillon [43, 81]. It looks like a difficult (possibly hopeless) problem in general, which is, moreover, obsolete from the decoding point of view, given the solutions of the decoding problem which we will discuss below.

In his thesis Porter [79] gave another decoding algorithm generalizing the solution of the *key equation* with *Euclid's algorithm* by Sugiyama, Kasahara, Hirasawa and Namekawa [104]. The correctness of the algorithm and the fact that it decodes $(\delta_\Gamma - 1)/2 - \sigma$ errors was shown in a paper of Porter, Shen and Pellikaan [79, 80] and by Ehrhard [20, 21]. The latter proved moreover that the modified algorithm and Porter's algorithm are equivalent. Euclid's algorithm is replaced by the *subresultant sequence* [79, 94].

An effective algorithm which corrects $(\delta_\Gamma - 1)/2$ errors was given by Ehrhard [22]. The problem of finding the divisors $F_1, \ldots, F_s$ in advance was circumvented by letting the algorithm find those divisors depending on the received word.

An elegant solution of the decoding problem for algebraic-geometric codes by a *majority vote* for *unknown syndromes* was proposed by Feng and Rao [23]. They showed in collaboration with Duursma [17, 18] that the algorithm corrects $(\delta_\Gamma - 1)/2$ errors. The origin of these methods stems from the decoding of cyclic codes beyond the BCH error-correcting capacity by Feng and Tzeng [28]. As a result of the majority voting it was noticed that sometimes one can even correct beyond half the Goppa designed minimum distance [24]. This was formalized by Kirfel and Pellikaan [53] who introduced the *Feng-Rao designed minimum distance* δ_{FR} for one point codes $C_\Omega(D, mP)$, which is determined by the *semigroup of non-gaps* of the point P. Shen and Tzeng [97] applied the majority voting scheme to Porter's algorithm. Another contribution by Feng, Wei, Rao and Tzeng [24, 25, 29], which resulted from decoding by majority voting, is that both the construction and the determination of the parameters of the codes can be done in an elementary way.

The Berlekamp-Massey algorithm [5, 67] on *linear recurring relations* in one vari-

able was generalized by Sakata [85, 86] to several variables, and will be called the *algorithm of BMS*. This algorithm was applied by Justesen, Larsen, Elbrønd Jensen and Høholdt [49] and Shen [95, 96] to get faster implementations of the modified algorithm, and later the majority voting was also incorporated by Sakata, Justesen, Madelung, Elbrønd Jensen and Høholdt [65, 89, 90, 91].
Using the block-Hankel structure of codes on plane curves, Feng, Wei, Rao and Tzeng [29] could lower the complexity of the majority voting scheme too.

The *error-locator ideal* was determined by Saints and Heegard [83, 84] and Leonard [60] with the algorithm of BMS. The former authors enlarged the notion of algebraic-geometric codes in such a way that it gives a unified treatment of (multi)cyclic, Hyperbolic Cascaded Reed Solomon, Reed-Muller and AG codes. The latter author [61] used a change of an admissible order on the monomials to get a generalization of Forney's algorithm for the error values.

The basic algorithm can be phrased in terms of arbitrary linear codes by the notion of an *error-correcting pair* of Pellikaan [71, 74, 76], later independently found by Kötter [54]. This has been applied to decode cyclic codes beyond their BCH error-correcting capacity by Duursma and Kötter [19]. Majority voting was generalized to cyclic and AG codes with the notion of *generalized Newton identities* by Feng and Tzeng [28] and Shen and Tzeng [98, 99], and to arbitrary linear codes with the notion of an *error-correcting array* and the *shift bound* by Pellikaan [75, 77].

With the basic algorithm Skorobogatov and Vlăduţ gave an algorithm to correct *errors-and-erasures* up to $\delta_\Gamma - \gamma$. Several authors [26, 55, 88, 99] have papers in preparation which deal with the extension of the various decoding algorithms to error-and-erasure correction. We will not survey these papers and only mention that as a result one now has *soft-decision* decoding algorithms for AG codes, by combining these extensions with the *generalized minimum distance* algorithm of Forney [31].

Some specific examples on the Klein curve are examined in detail. These examples are primarily chosen from a didactical point of view and not because the codes are good. We want to illustrate the concepts and properties, and compare the distinct algorithms in these examples.

As a reference on coding theory we use MacWilliams and Sloane [69]. There are some introductory works on the algebraic-geometric codes. See [37, 62, 63, 64]. For a self-contained treatment on algebraic curves over finite fields we recommend the books of Chevalley [12] and Stichtenoth [102] which treat curves from the algebraic point of view, that is to say by function fields of one variable. A more geometric point of view

is taken by Fulton [32]. Furthermore we mention Moreno [70]. These books are sufficient to understand the papers concerning the decoding of algebraic-geometric codes. A standard reference of algebraic geometry for mathematicians is Hartshorne [40]. Abhyankar [2] is written for engineers. For a deeper understanding of modular curves and their codes we mention the book of Tsfasman and Vlăduţ [107]. For computational algebra, that is to say for Gröbner bases and algorithms for computing them, we refer to Buchberger [10] and Cox, Little and O'Shea [14].

Notation $\mathbb{F}$ denotes a field, $\mathbb{F}_q$ denotes the finite field with q elements. $\mathbb{Z}$ is the set of integers, $\mathbb{N}$ the set of positive integers and $\mathbb{N}_0$ the set of non-negative integers. Vectors will usually be row vectors, H^T is the transpose of a matrix H. The standard inner product of two vectors $\mathbf{a}$ and $\mathbf{b}$ is equal to $\mathbf{a}\mathbf{b}^T = \sum a_i b_i$. The dual of a subset A of $\mathbb{F}_q^n$ is defined by $A^\perp = \{\mathbf{b} \in \mathbb{F}_q^n \mid \mathbf{b}\mathbf{a}^T = 0 \text{ for all } \mathbf{a} \in A\}$. The number of elements of a set A is denoted by $\#A$. If A is a subset of a vector space or semigroup, then $\langle A \rangle$ is the subspace or sub-semigroup generated by A. Define the star multiplication $\mathbf{a} * \mathbf{b}$ of two elements $\mathbf{a}$ and $\mathbf{b}$ of $\mathbb{F}_q^n$ by coordinatewise multiplication, that is $(\mathbf{a} * \mathbf{b})_i = a_i b_i$. For two subsets A and B of $\mathbb{F}_q^n$ we denote the set $\{\mathbf{a} * \mathbf{b} \mid \mathbf{a} \in A, \mathbf{b} \in B\}$ by $A * B$. Define $A^\perp = \{\mathbf{b} \in \mathbb{F}_q^n \mid \mathbf{b}\mathbf{a}^T = 0 \text{ for all } \mathbf{a} \in A\}$ for a subset A of $\mathbb{F}_q^n$.

2 The decoding problem

In this paper we consider error-correction using *codes*, these are sets of words of a fixed length n and with symbols from the finite field $\mathbb{F}_q$ with q elements. The *Hamming distance* $d(\mathbf{a}, \mathbf{b})$ between two words $\mathbf{a}$ and $\mathbf{b}$ of $\mathbb{F}_q^n$ is by definition the number of positions where they differ. The *weight* $wt(\mathbf{a})$ of a word $\mathbf{a}$ is the number of non-zero positions. Let C be a code in $\mathbb{F}_q^n$, the *minimum distance* of C is the minimum distance between two distinct words of C and will be denoted by $d(C)$ or d. If $\mathbf{c}$ is the transmitted word and $\mathbf{c} + \mathbf{e}$ is the received word, then we call $\mathbf{e}$ the *error vector* and $\{i | e_i \neq 0\}$ the set of *error positions*, the e_i's are called the *error values* and $wt(\mathbf{e})$ is the *number of errors* of the received word. If $\mathbf{y}$ is the received word and the distance of $\mathbf{y}$ to the code C is t', then there exists a codeword $\mathbf{c}'$ and an error vector $\mathbf{e}'$ such that $\mathbf{y} = \mathbf{c}' + \mathbf{e}'$ and $wt(\mathbf{e}') = t'$. If the number of errors is at most $(d-1)/2$, then we are sure that $\mathbf{c} = \mathbf{c}'$ and $\mathbf{e} = \mathbf{e}'$. In other words, the nearest codeword to $\mathbf{y}$ is unique when $\mathbf{y}$ has distance at most $(d-1)/2$ to C.

We are only interested in *linear codes*, which are subspaces of $\mathbb{F}_q^n$. The dimension of a code C is denoted by $k(C)$ or k. For linear codes the minimum distance is equal to the minimum weight of a non-zero word of C. The *information rate* k/n of the code is denoted by R. A *generator matrix* of an $[n, k]$ code C over $\mathbb{F}_q$ is a $k \times n$ matrix G

such that $C = \{\mathbf{x}G \mid \mathbf{x} \in \mathbb{F}_q^k\}$. Thus the map

$$\mathcal{E} : \mathbb{F}_q^k \longrightarrow \mathbb{F}_q^n,$$

defined by $\mathcal{E}(\mathbf{x}) = \mathbf{x}G$ is linear and encodes words of length k to codewords of C of length n. Define $C^* = C \cup \{?\}$. A map

$$\mathcal{D} : \mathbb{F}_q^n \longrightarrow C^*$$

is called a *decoder* for the code C if $\mathcal{D}(\mathbf{c}) = \mathbf{c}$, for all $\mathbf{c} \in C$. We allow the decoder to give as outcome "?", when it fails to find a codeword.

Definition 2.1 A *minimum distance decoder* for a code C is a decoder $\mathcal{D}$ such that $\mathcal{D}(\mathbf{y})$ is a closest codeword to $\mathbf{y}$ for all $\mathbf{y}$.

A *decoding error* of a decoder occurs when the decoded word is different from the transmitted word. A *maximum likelihood decoder* minimizes the probability of a decoding error. Minimum distance decoding is equivalent to maximum likelihood decoding for a *q-ary symmetric channel*. In a q-ary symmetric channel the probability that a symbol will be changed to another one is the same for all letters in the alphabet and does not depend on the position in the word.

If $C = \{\mathbf{c} \in \mathbb{F}_q^n \mid H\mathbf{c}^T = 0\}$ for an $(n-k) \times n$ matrix H, then H is called a *parity check matrix* for C. We have the *standard bilinear form* on $\mathbb{F}_q^n$ defined by $\mathbf{a}\mathbf{b}^T = \sum_i a_i b_i$. If A is a subset of $\mathbb{F}_q^n$, then we define the *dual* $A^\perp$ of A in $\mathbb{F}_q^n$ with respect to the standard bilinear form by $A^\perp = \{\mathbf{b} \mid \mathbf{a}\mathbf{b}^T = 0 \text{ for all } \mathbf{a} \in A\}$. So in this definition A is not necessarily linear but $A^\perp$ is always linear. Thus the rows $\mathbf{h}_i, i = 1, \ldots, n-k$ of a parity check matrix H of a linear code C form a basis for the dual of C. For a received word $\mathbf{y} \in \mathbb{F}_q^n$ and a parity check matrix H for a code, the *syndromes* of $\mathbf{y}$ are by definition $s_i(\mathbf{y}) = \mathbf{h}_i\mathbf{y}^T, i = 1, \ldots, n-k$. An important decoding method will be to consider an extension of the parity check matrix H to $\hat{H}$ of size $n \times n$ such that the n rows $\mathbf{h}_i, i = 1, \ldots, n$ are a basis for $\mathbb{F}_q^n$ and the first $n-k$ rows are from H. The n syndromes $s_i(\mathbf{e}) = \mathbf{e}\mathbf{h}_i^T$ for $i = 1, \ldots, n$ determine the error vector uniquely, but only the first $n-k$ syndromes are known, since $s_i(\mathbf{e}) = s_i(\mathbf{y})$, for $i = 1, \ldots, n-k$. The remaining syndromes are called *unknown syndromes*, and we will see how *majority voting* gives a procedure to compute the unknown syndromes for some classes of codes.

The set of all words with the same syndrome as $\mathbf{y}$ is the *coset* $\mathbf{y} + C$. An element of the coset $\mathbf{y} + C$ of minimal weight is called a *coset leader*. A simple minimum distance decoder consists of an exhaustive search for a coset leader. An alternative would be to make a list of all coset leaders. It is clear that both methods have exponential complexity as a function of the codelength, since either one has to search among q^{Rn}

elements of the coset of a received word to find one of minimal weight, or one has to store $q^{(1-R)n}$ coset leaders. There is a vast amount of literature on the various algorithms for minimum distance decoders but they will not be discussed here. All known minimum distance decoding algorithms have exponential complexity.

If $\mathcal{A}$ is an algorithm which has as input a pair $(C, \mathbf{y})$, where C is a linear code and $\mathbf{y}$ a word of the same length as C, then $\mathcal{A}_C$ is the *restriction* of the algorithm $\mathcal{A}$ to C, that is to say $\mathcal{A}_C$ has as input a word $\mathbf{y}$ of the same length as C and as output $\mathcal{A}(C, \mathbf{y})$ computed by $\mathcal{A}$. Consider the following problem: Find an algorithm $\mathcal{A}$ which has as input $(C, \mathbf{y})$, where C is a linear code given say by a parity check matrix and $\mathbf{y}$ a (received) word of the same length as C, and as output a word $\mathcal{A}(C, \mathbf{y})$ in C^* such that $\mathcal{A}_C$ is a minimum distance decoder for C. This problem is NP hard. See [6] and also [4]. McEliece cryptosystem [68] is based on this fact. The problem as posed above has two parts. Firstly the *preprocessing* part done at a laboratory and a factory where for an appropriate code C a decoder $\mathcal{A}_C$ is built and this is allowed to be time consuming. Seconly the actual operating of the many copies of the decoder for consumers which should work very fast. So one considers the problem of *minimum distance decoding with preprocessing*. This problem also turns out to be hard. See [9].

Thus the decoding algorithms which we will treat below have a twofold application: error-correction and cryptology. From the error-correction point of view it seems pointless to decode a bad code, but if one wants to break the McEliece system one must be able to decode efficiently all, or almost all, codes, including bad codes.

The known decoding algorithms which have polynomial complexity decode only up to a certain bound, for instance up to half the (designed) minimum distance.

Definition 2.2 A decoder $\mathcal{D}$ for a code C is called a *bounded distance* decoder which *corrects t errors* if $\mathcal{D}(\mathbf{y})$ is a nearest codeword for all $\mathbf{y} \in \mathbb{F}_q^n$ such that $d(\mathbf{y}, C) \leq t$. A decoder $\mathcal{D}$ for a code C of minimum distance d *decodes up to half the minimum distance* if $\mathcal{D}(\mathbf{y})$ is the nearest codeword for all $\mathbf{y} \in \mathbb{F}_q^n$ such that $d(\mathbf{y}, C) \leq (d-1)/2$.

Concerning statements about the complexity of an algorithm we use the "big O" notation. We say $f(n) = \mathcal{O}(g(n))$ for $n \to \infty$ if and only if there exists a constant C and an integer n_0 such that $f(n) \leq Cg(n)$ for all $n \geq n_0$. All decoding algorithms for algebraic-geometric codes, which we will treat below, decode up to half the designed minimum distance and have complexity $\mathcal{O}(n^3)$ or less for $n \to \infty$. Whether this is the case for all linear codes we pose as the following problem from [4] on decoding up to half the minimum distance (with preprocessing):

Problem 2.3 Is there an algorithm $\mathcal{A}$ which has as input $(C, \mathbf{y})$, where C is a linear code and $\mathbf{y}$ a word of the same length as C and as output a word $\mathcal{A}(C, \mathbf{y})$ in C^*

such that $\mathcal{A}_C$ is a decoder for C which decodes up to half the minimum distance and the complexity of the algorithm $\mathcal{A}_C$ is polynomial as function of the codelength and independent of C ?

We finish this section by showing a well-known fact that errors can be corrected if we have enough information about the error positions, in other words if we have erasures only.

Proposition 2.4 *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 which contains the set of error positions. Then the error-vector $\mathbf{e}$ is the unique solution of the following linear equations:*

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

It is clear that the error vector is a solution. If $\mathbf{x}$ is another solution, then $H(\mathbf{x}-\mathbf{e})^T = 0$. Therefore $\mathbf{x}-\mathbf{e}$ is an element of C, and moreover it is supported at J. So its weight is at most $d(C)-1$. Thus it must be zero; so $\mathbf{x} = \mathbf{e}$.

Thus we have shown that we can reduce error decoding 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 when t/n tends to a non-zero real number. From Proposition 2.4 above it is enough to find an $(n, d-1, t)$ *covering system.* That is to say a collection $\mathcal{J}$ of subsets J of $\{1, \ldots, n\}$, such that all $J \in \mathcal{J}$ have $d-1$ elements and every subset of $\{1, \ldots, n\}$ of size t is contained in at least one $J \in \mathcal{J}$. The size of such a covering set is considerably smaller than the number of all possible t sets, but is at least $\binom{n}{t}/\binom{d-1}{t}$. This number still grows exponentially with n.

Most of the decoding algorithms which we will survey give a polynomial, a vector, a function or an ideal of functions which locates the errors, that is to say the set of zeros of the function(s) will contain the error positions.

3 Algebraic-geometric codes

For this section we refer to the work of Goppa [33, 34, 35, 36, 37], the survey papers on AG codes in this special issue and many other references [2, 12, 32, 62, 63, 64, 70, 102, 107]. By a *curve* over a field $\mathbb{F}_q$ we mean an affine or projective variety of dimension one, which is absolutely irreducible and non-singular and whose defining equations are (homogeneous) polynomials with coefficients in $\mathbb{F}_q$. For an algebraic curve $\mathcal{X}$ over the

field $\mathbb{F}_q$ we denote its *function field* by $\mathbb{F}_q(\mathcal{X})$. A *rational point* of a variety over $\mathbb{F}_q$ is a point whose coordinates are in $\mathbb{F}_q$. The *Frobenius map* of a variety over $\mathbb{F}_q$ sends a point with coordinates $(a_1, \ldots, a_n)$ to $(a_1^q, \ldots, a_n^q)$ and similarly for homogeneous coordinates. A *place* of degree m of a variety $\mathcal{X}$ over $\mathbb{F}_q$ is a set of m points of the variety which are rational over $\mathbb{F}_{q^m}$ and which are cyclically permuted by the Frobenius map. With every place P of a curve $\mathcal{X}$ one can associate the *valuation* v_P on the field $\mathbb{F}_q(\mathcal{X})$ of rational functions on $\mathcal{X}$ which measures the order of zero or pole of a function at this place. A *divisor* is a formal sum $\sum m_P P$ of places P with integer coefficients m_P such that finitely many coefficients are non-zero. One can add divisors coefficientwise. One has a partial order on the divisors by comparing them coefficientwise. The *support* of a divisor G is the set of places, P, for which m_P is non-zero. A divisor is called *effective* if all the coefficients are not negative. The *degree* of a divisor G of the form $\sum m_P P$ is by definition

$$\deg(G) = \sum m_P \deg(P).$$

A rational function f has a divisor (f), given by

$$(f) = \sum v_P(f) P,$$

which we call the *principal divisor* of the function. The principal divisor of a rational function f is the difference of two effective divisors, $(f) = (f)_0 - (f)_\infty$, where $(f)_0$ is the *divisor of zeros* and $(f)_\infty$ is the *divisor of poles* of f. Thus

$$(f)_0 = \sum_{v_P(f)>0} v_P(f) P \quad \text{and} \quad (f)_\infty = \sum_{v_P(f)<0} -v_P(f) P.$$

The degree of a principal divisor is zero. Two divisors G and H are called *equivalent* if their difference is a principal divisor, this is denoted by $G \equiv H$.

For every divisor G one defines the finite dimensional vector space $L(G)$ over $\mathbb{F}_q$ of rational functions with zeros and poles prescribed by G as follows

$$L(G) = \{f \in \mathbb{F}_q(X) \mid f = 0 \text{ or } (f) \geq -G\}.$$

The dimension of $L(G)$ is denoted by $l(G)$. The *Theorem of Riemann* says that there exists a non-negative integer m such that for every divisor G

$$l(G) \geq \deg(G) + 1 - m,$$

and the smallest non-negative integer with this property is called the *genus*, and is denoted by $\gamma(\mathcal{X})$ or γ. The genus is zero if and only if the curve is the projective line. Curves of genus one are also called *elliptic*. The *gonality* of a curve is the smallest degree of a divisor G such that $l(G) > 1$, or equivalently, it is the smallest degree of a

non-constant morphism from the curve to the projective line, and is denoted by $\tau(\mathcal{X})$ or τ. The gonality is one if and only if the curve is the projective line. A curve which is not elliptic is called *hyperelliptic* if it has gonality two.

Let $\mathcal{X}$ be a curve over $\mathbb{F}_q$. Let $P_1, \ldots, P_n$ be n distinct rational points of $\mathcal{X}$. We denote the divisor $P_1 + \cdots + P_n$ by D. Let G be a divisor with support disjoint form the support of D. Then we can evaluate f at all $P_i's$ and consider the map

$$\mathrm{ev}_D : L(G) \longrightarrow \mathbb{F}_q^n,$$

defined by $\mathrm{ev}_D(f) = (f(P_1), \ldots, f(P_n))$. This map is $\mathbb{F}_q$-linear and injective when $\deg(G) < n$. We denote the image of this map by $C_L(D, G)$. This code is called an *algebraic-geometric* or a *generalized Reed-Solomon* code on $\mathcal{X}$. The following proposition is a direct consequence of the Theorem of Riemann-Roch.

Proposition 3.1 *The code $C_L(D, G)$ is $\mathbb{F}_q$-linear with parameters $[n, k, d]$ such that $d \geq n - deg(G)$ and $k \geq deg(G) + 1 - \gamma$ when $deg(G) < n$. Moreover if $deg(G) > 2\gamma - 2$, then $k = deg(G) + 1 - \gamma$.*

The *Goppa designed minimum distance* of $C_L(D, G)$ is by definition $n - \deg(G)$ and is denoted by δ_Γ. The minimum distance of the dual of a code can be estimated in terms of the minimal number of dependent columns of a generator matrix of the code. Thus we have the following property: $d(C^\perp) > t$ if and only if $\dim(C(I)) = k - \#I$ for all $I \subseteq \{1, \ldots, n\}$ and $\#I \leq t$, where $C(I)$ is the subcode of words in C which are zero at all $i \in I$. If $Q = \sum_{i \in I} P_i$, then $C_L(D, G)(I) = C_L(D, G - Q)$. The dimension of $C_L(D, G - Q)$ is $\deg(Q) = \#I$ smaller than the dimension of $C_L(D, G)$ if $\#I < \deg(G) - (2\gamma - 2)$ and $\deg(G) < n$. Thus the dual of $C_L(D, G)$ has at least minimum distance $\deg(G) - 2\gamma + 2$.

Another way to get an estimate for the minimum distance of the dual code of $C_L(D, G)$ is with the help of *differential forms*, also called *differentials*. Think of differentials as objects of the form fdg, where f and g are rational functions and dg is the differential of g, such that the map which sends g to dg is a *derivation*. A derivation is $\mathbb{F}_q$-linear and the *Leibniz rule* $d(fg) = fdg + gdf$ holds. We denote the set of differentials on $\mathcal{X}$ by $\Omega_\mathcal{X}$. One can talk about zeros and poles of differentials. At every place P there exists a *local parameter* that is a function u which has valuation 1 at P, and for every differential ω there exists a function f such that $\omega = fdu$. The valuation $v_P(\omega)$ of ω at P is now by definition $v_P(f)$, so we say ω has a zero of order ρ if $\rho = v_P(f) > 0$ and ω has a pole of order ρ if $\rho = -v_P(f) > 0$. The divisor (ω) of ω is by definition $(\omega) = \sum v_P(\omega)P$. The divisor of a differential is called *canonical* and always has degree $2\gamma - 2$. Any two canonical divisors are equivalent. In the same

way as we have defined $L(G)$ for functions we now define the vector space $\Omega(G)$ of differentials with zeros and poles prescribed by G as

$$\Omega(G) = \{\omega \in \Omega_{\mathcal{X}} \mid \omega = 0 \text{ or } (\omega) \geq G\}.$$

Now one could have defined the genus as the dimension of the vector space of differentials without poles, that is of $\Omega(O)$, where O is the divisor with coefficient 0 at every place. The dimension of $\Omega(G)$ is called the *index of speciality* of G and is denoted by $i(G)$. The *Theorem of Riemann-Roch* states that

$$l(G) = \deg(G) + 1 - \gamma + i(G).$$

Furthermore $i(G) = l(K - G)$ for all divisors G and canonical divisors K.

Let ω be a differential form. If P is a place of degree m and u is a local parameter at P, then there exists a rational function f such that $\omega = fdu$. This function f has a formal Laurent series $\sum_{i=\rho}^{\infty} a_i u^i$, where the coefficients a_i are in $\mathbb{F}_{q^m}$ and $\rho = v_P(\omega)$ and $a_\rho \neq 0$. The *residue* of ω at P is by definition $Tr(a_{-1})$ and denoted by $\mathrm{res}_P(\omega)$, where Tr is the trace map from $\mathbb{F}_{q^m}$ to $\mathbb{F}_q$. If $P_1, \ldots, P_n$ are n distinct rational points on the curve and G is a divisor with none of the P_i in its support, then consider the map

$$\mathrm{res}_D : \Omega(G - D) \longrightarrow \mathbb{F}_q^n,$$

defined by $\mathrm{res}_D(\omega) = (\mathrm{res}_{P_1}(\omega), \ldots, \mathrm{res}_{P_n}(\omega))$. Its image is denoted by $C_\Omega(D, G)$. Such codes are called *geometric Goppa* or *algebraic-geometric*. As a corollary of the Theorem of Riemann-Roch we get

Proposition 3.2 *The code $C_\Omega(D, G)$ is $\mathbb{F}_q$-linear with parameters $[n, k, d]$ such that $d \geq deg(G) - 2\gamma + 2$ and $k \geq n - deg(G) - 1 + \gamma$ when $deg(G) > 2\gamma - 2$. Moreover if $deg(G) < n$, then $k = n - deg(G) - 1 + \gamma$.*

We call $\deg(G) - 2\gamma + 2$ the *Goppa designed minimum distance* of the code $C_\Omega(D, G)$ and denote it by δ_Γ as well.

The fact that the *functional* code $C_L(D, G)$ and the *differential* code $C_\Omega(D, G)$ are dual codes is seen by the *Residue Theorem*, which states that the sum of all residues over all places of a fixed differential is zero. That both the functional and the differential codes are called algebraic-geometric codes is justified by the fact that for every curve and every choice of n rational points $P_1, \ldots, P_n$ there exists a canonical divisor K of a differential form ω, which has simple poles at P_i with $res_{P_i}(\omega) = 1$ for all i, such that $C_L(D, G) = C_\Omega(D, K + D - G)$ for all divisors G. This fact also shows that it is not a loss of the generality to consider only decoding algorithms for the differential

codes.

In many papers *one point codes* are considered, that is to say codes of the form $C_L(D, mP)$ or $C_\Omega(D, mP)$, where P is a rational point which is distinct from all $P_1, \ldots, P_n$ and m an integer. Let m be a non-negative integer. If $l(mP) = l((m-1)P)$, then m is called a *(Weierstrass) gap* of P. The number of gaps of P is equal to the genus γ of the curve, since $l(iP) = i + 1 - \gamma$ if $i > 2\gamma - 2$, and

$$1 = l(0) \leq l(P) \leq \cdots \leq l((2\gamma - 1)P) = \gamma.$$

If $m \in \mathbb{N}_0$, then m is a non-gap of P if and only if there exists a rational function which has a pole of order m at P and no other poles. If m_1 and m_2 are non-gaps of P, then $m_1 + m_2$ is also a non-gap of P, thus the non-gaps form a *semigroup* in $\mathbb{N}_0$. Let $(\rho_i | i \in \mathbb{N})$ be an enumeration of all the non-gaps of P in increasing order, so $\rho_1 = 0$. The semigroup of non-gaps is always generated by $\rho_2, \rho_3, \ldots, \rho_{\gamma+2}$. Let $g_i \in L(\rho_i P)$ be such that $v_P(g_i) = -\rho_i$ for $i \in \mathbb{N}$. The ring $K_\infty(P)$ of rational functions on the curve with only poles at P plays an important role. Now

$$K_\infty(P) = \mathbb{F}_q[g_2, g_3, \ldots, g_{\gamma+2}],$$

and

$$\mathbb{F}_q[Y_1, \ldots, Y_{\gamma+1}]/I \cong K_\infty(P),$$

where the isomorphism is given by sending Y_i to g_{i+1}, and I is some ideal in the ring $\mathbb{F}_q[Y_1, \ldots, Y_{\gamma+1}]$. See [80]. If the semigroup of non-gaps is generated by $\rho_{i_1}, \ldots \rho_{i_l}$, then

$$K_\infty(P) = \mathbb{F}_q[g_{i_1}, \ldots, g_{i_l}],$$

such that

$$\mathbb{F}_q[Z_1, \ldots, Z_l]/J \cong K_\infty(P),$$

where the isomorphism is given by sending Z_j to g_{i_j}, and J is some ideal in the ring $\mathbb{F}_q[Z_1, \ldots, Z_l]$. This has the following geometric meaning: if P is a point on the curve $\mathcal{X}$ then one can embed $\mathcal{X} \setminus \{P\}$ in affine space of dimension l with defining equations given by the elements of the ideal J. The corresponding projective curve has only P in the hyperplane at infinity.

Example 3.3 Reed-Solomon codes.
Let $\alpha = (\alpha_1, \ldots, \alpha_n)$ be an n-tuple of n distinct elements of $\mathbb{F}_q$. Define the k-dimensional *Reed-Solomon* code over $\mathbb{F}_q$ by

$$RS_k(\alpha, \mathbf{x}) = \{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_q[X], \deg(f) < k\}.$$

Let $\mathbf{x}$ be an n-tuple of non-zero elements of $\mathbb{F}_q$. The k-dimensional *Generalized Reed-Solomon* code over $\mathbb{F}_q$ is defined by

$$GRS_k(\alpha, \mathbf{x}) = \{(f(\alpha_1)x_1, \ldots, f(\alpha_n)x_n) \mid f \in \mathbb{F}_{q^m}[X], \deg(f) < k\}.$$

Let g be a polynomial in $\mathbb{F}_q[X]$ such that $g(\alpha_i) = x_i$ for all i. Let $P_i = (\alpha_i : 1)$ and $P = (1 : 0)$ be the point at infinity of the projective line. Let $D = P_1 + \cdots + P_n$. Let the divisor G on the projective line be defined by $G = (k-1)P_\infty - (g)$. Then $GRS_k(\alpha, \mathbf{x}) = C_L(D, G)$. The set of extended Generalized RS codes is exactly the set of codes on the projective line. That is why the functional codes $C_L(D, G)$ on curves are called Generalized RS codes too by some authors.

Example 3.4 Goppa codes.
Let $L = \{\alpha_1, \ldots, \alpha_n\}$ be a set of n distinct elements of $\mathbb{F}_{q^m}$. Let g a polynomial in $\mathbb{F}_{q^m}[X]$ which is not zero at α_i for all i. The *(classical) Goppa code* $\Gamma(L, g)$ is defined by

$$\Gamma(L, g) = \{\mathbf{c} \in \mathbb{F}_q^n \mid \sum \frac{c_i}{X - \alpha_i} \equiv 0 \ (mod\ g\ )\}.$$

Let the P_i, P and D be as in the previous example. If we take for E the divisor of zeros of g, then $\Gamma(L, g) = C_\Omega(D, E - P)$ and

$$\mathbf{c} \in \Gamma(L, g) \text{ if and only if } \sum \frac{c_i}{X - \alpha_i} dX \in \Omega(E - P - D).$$

This is the reason that some authors extend the definiton of Geometric Goppa codes to subfield subcodes of differential codes of the form $C_\Omega(D, G)$. The duality between functional and differential codes is for instance seen by the well-known fact [69] that the parity check matrix of the Goppa code $\Gamma(L, g)$ is equal to the following generator matrix of a Generalized RS code

$$\begin{pmatrix} g(\alpha_1)^{-1} & \ldots & g(\alpha_n)^{-1} \\ \alpha_1 g(\alpha_1)^{-1} & \ldots & \alpha_n g(\alpha_n)^{-1} \\ \vdots & \cdots & \vdots \\ \alpha_1^{r-1} g(\alpha_1)^{-1} & \ldots & \alpha_n^{r-1} g(\alpha_n)^{-1} \end{pmatrix},$$

where r is the degree of the Goppa polynomial g.

Example 3.5 Let $\mathcal{X}$ be a non-singular projective plane curve of degree m over $\mathbb{F}_q$. Then the curve is also absolutely irreducible. Let the affine part of $\mathcal{X}$ be given by the equation $F(X, Y) = 0$, where $F(X, Y)$ is a polynomial in the variables X and Y with coefficients in $\mathbb{F}_q$. The coordinate ring of this affine part is equal to $\mathbb{F}_q[X, Y]/(F)$. The cosets of X and Y modulo the ideal generated by F are denoted by x and y, respectively. Let H be the intersection divisor of the curve with the line at infinity.

Then jH has degree jm and $L(jH)$ is generated by all $x^a y^b$ such that $a+b \leq j$, $a, b \in \mathbb{N}_0$, and the relations come from multiples of F. Thus $l(jH) = \binom{j+2}{2}$ if $j < m$, and

$$l(jH) = \binom{j+2}{2} - \binom{j-m+2}{2} = jm + 1 - \frac{(m-1)(m-2)}{2}$$

if $j \geq m$. See [32, 48]. The genus of such a curve is $(m-1)(m-2)/2$.

Example 3.6 Many papers on the decoding of codes on curves use the *Hermitian curve* as an example which is given by the homogeneous equation

$$X^{r+1} + Y^{r+1} + Z^{r+1} = 0$$

over $\mathbb{F}_q$, where $q = r^2$. This curve has r^3+1 rational points over $\mathbb{F}_q$ and genus $r(r-1)/2$. This curve is isomorphic to the curve with affine equation

$$Y^r + Y = X^{r+1},$$

which has exactly one point P at infinity and $n = r^3$ rational points $P_1, \ldots, P_n$ in the affine plane. The semigroup of non-gaps at P is generated by r and $r+1$ and the corresponding functions are x and y, so

$$K_\infty(P) = \mathbb{F}_q[x, y] \cong \mathbb{F}_q[X, Y]/I,$$

where I is the ideal of $\mathbb{F}_q[X, Y]$ generated by $Y^r + Y - X^{r+1}$. Hermitian codes are codes of the form $C_L(P_1 + \cdots + P_n, mP)$. See [41, 48, 64, 65, 95, 96, 101, 106, 112].

Example 3.7 The *Klein quartic* over F_8 has the affine equation

$$X^3Y + Y^3 + X = 0.$$

It has genus $\gamma = 3$, and 3 points $R_1 = (1:0:0)$, $R_2 = (0:1:0)$ and $R_3 = (0:0:1)$ which are rational over $\mathbb{F}_2$, and 21 points $P_1, \ldots, P_{21}$ which are rational over $\mathbb{F}_8$, but not over $\mathbb{F}_2$. The codes $C(m) = C_\Omega(P_1 + \cdots + P_{23}, mP)$, where $4 < m < 23$, $P_{22} = R_3$, $P_{23} = R_1$ and $P = R_2$, have parameters $[23, 25-m, \geq m-4]$. This curve has been studied most often in papers on AG codes and their decoding [16, 18, 25, 38, 43, 63, 81, 98]. In these papers however one considers codes of the form $C_\Omega(D, G)$, where $D = P_1 + \cdots + P_{21}$ and $G = m(R_1 + R_2 + R_3)$. In [60, 61] and the following decoding algorithms the codes $C(m)$ are used as an example.

The homogeneous equation of the Klein quartic is

$$X^3Y + Y^3Z + Z^3X = 0$$

and from this we readily see that the intersection divisor of the curve with the line with equation $X = 0$ is equal to $3R_3 + R_2$, with the line $Y = 0$ is $3R_1 + R_3$, and with the line $Z = 0$ is $3R_2 + R_1$. The function x is the quotient of X and Z, so the principal divisors of x is

$$(x) = (3R_3 + R_2) - (3R_2 + R_1) = 3R_3 - R_1 - 2R_2.$$

Similarly, if $y = Y/Z$, then

$$\begin{array}{rcl} (y) & = & 2R_1 + R_3 - 3R_2 \\ (xy) & = & R_1 + 4R_3 - 5R_2 \\ (y^2) & = & 4R_1 + 2R_3 - 6R_2 \\ (x^2y) & = & 7R_3 - 7R_2 \end{array}$$

and in general

$$(x^i y^j) \quad = \quad (3i + j)R_3 + (2j - i)R_1 - (2i + 3j)R_2.$$

The non-gaps at $P = R_2$ less than or equal to $2\gamma + 1 = 7$ are $0, 3, 5, 6$ and 7, and the corresponding functions are $g_1 = 1$, $g_2 = y$, $g_3 = xy$, $g_4 = y^2$ and $g_5 = x^2y$. Furthermore $g_{3j-2} = y^j$, $g_{3j-1} = x^2y^{j-1}$ and $g_{3j} = xy^j$ for $j \geq 2$. There are some obvious equations between the g_i such as

$$g_2g_5 = g_3^2, \;\; g_4 = g_2^2, \;\; g_{3j-2} = g_2^j, \;\; g_{3j-1} = g_5g_2^{j-2} \quad \text{and} \quad g_{3j} = g_3g_2^{j-1}.$$

From the affine equation of the Klein quartic follows

$$g_2^4 + g_3 + g_3g_5 = 0,$$

and there is another equation which is more difficult to find

$$g_2^3g_3 + g_5 + g_5^2 = 0.$$

See [61]. One can show that all equations are consequences of these equations, thus

$$K_\infty(P) = \mathbb{F}_8[g_2, g_3, g_5] \cong \mathbb{F}_8[Z_1, Z_2, Z_3]/J,$$

where

$$J = (Z_1Z_3 + Z_2^2, Z_1^4 + Z_2 + Z_2Z_3, Z_1^3Z_2 + Z_3 + Z_3^2).$$

A point R on the affine plane model of the curve with coordinates (a, b) has coordinates (b, ab, a^2b) on the model in affine 3-space. R_1 and R_2 are the only points on the plane model at the line at infinity $Z = 0$. R_1 is a zero of g_2, g_3 and $g_5 + 1$, so the corresponding point in 3-space has coordinates $(0, 0, 1)$. The point $P = R_2$ is a pole of g_2, g_3 and g_5, and is therefore the unique point at infinity of the curve in 3-space. See

[84] for a description of $\mathbb{F}_8[g_2, g_3, g_5 + 1]$.

The effective canonical divisors are exactly the intersection divisors of the projective plane curve with a line. The intersection divisor of the curve and the line with equation $Z = 0$ is equal to $3R_2 + R_1$ and is therefore a canonical divisor. Since $(x) = 3R_3 - R_1 - 2R_2$ and $(y) = 2R_1 + R_3 - 3R_2$ we get

$$(\frac{y}{x}) = (y) - (x) = 3R_1 - R_2 - 2R_3.$$

We show below that $(dx) = 4R_2 + 2R_3 - 2R_1$ and therefore

$$(\frac{y}{x}dx) = R_1 + 3R_2.$$

At a point Q with local parameter t, we have that

$$x = \sum_{i \geq \rho} a_i t^i,$$

where $\rho = v_Q(x)$, so

$$dx = \sum_{i \geq \rho,\, i \text{ is odd}} i a_i t^{i-1} dt,$$

since the characteristic is even. Thus if ρ is odd, then $v_Q(dx) = \rho - 1$; if ρ is even, then $v_Q(dx) \geq \rho$; and if ρ is non-negative, then $v_Q(dx) \geq 0$. Thus

$$(dx) = 2R_3 - 2R_1 - 2R_2 + A,$$

where A is an effective divisor, since $(x) = 3R_3 - R_1 - 2R_2$. Let $t = x/y$. Then t is a local parameter at R_2, so

$$x = t^{-2} + a_{-1}t^{-1} + a_0 + \cdots \text{ and } y = t^{-3} + a_{-1}t^{-2} + a_0 t^{-1} + a_1 + \cdots .$$

The equation $x^3y + y^3 + x = 0$ implies $y^4t^3 + y^3 + yt = 0$, that is $t = y^2(1 + yt^3)$. Substituting the formal power series in t for y gives

$$t = (t^{-6} + a_{-1}^2 t^{-4} + a_0^2 t^{-2} + a_1^2 + a_2^2 t^2 + a_3 t^4 + a_4^2 t^6 + \cdots)(a_1 t + a_0 t^3 + a_1 t^3 + \cdots)$$

and therefore $a_i = 0$ for all $i < 5$ and $a_5 = 1$. Thus

$$x = t^{-2} + t^5 + \text{ higher order terms }, \quad \text{so}$$

$$dx = (t^4 + \text{ higher order terms })dt.$$

Therefore $(dx) = 2R_3 - 2R_1 + 4R_2 + B$, where B is effective, but $\deg(dx) = 2\gamma - 2 = 4$, so we must have that $B = 0$, which proves the desired result.

Example 3.8 The curve with equation

$$Y^8 + Y = X^2(X^8 + X)$$

over $\mathbb{F}_8$ has 64 rational points $P_1, \ldots, P_{64}$ in the affine plane and one rational point P at infinity, and genus $\gamma = 14$. This is the first in the series of curves connected with the *Suzuki group* considered in [39]. At P the semigroup of non-gaps is generated by $8, 10, 12, 13$ and the corresponding functions are $x, y, x^5 + y^4$ and $xy^4 + x^{20} + y^{16}$. The corresponding codes $C_\Omega(D, mP)$, $D = P_1 + \cdots + P_{64}$ have parameters $[64, \geq 77 - m, \geq m - 26]$ for $26 < m < 64$.

Example 3.9 Let r be a prime power such that $r \equiv 1(\bmod 3)$. The curve in affine 3-space over $\mathbb{F}_q$, $q = r^2$, defined by the *Kummer extensions*:

$$Y^{r+1} = X^r + X,$$

$$Z^{r+1} = -XY^r - YX^r - 1$$

has $(q-1)^2$ rational points and genus $r^3 + r^2 - 3$. See [110]. The functions x, y and z have a common pole at P and their pole orders at P are $(r+1)^2$, $r(r+1)$ and $r(r+2)$, respectively, and these 3 numbers are the generators of the semigroup of non-gaps at P.

4 The basic and modified algorithm

Recall that the Goppa designed minimum distance δ_Γ of $C_\Omega(D, G)$ is equal to $\deg(G) - 2\gamma + 2$.

Let $\mathbf{y}$ be a received word with error-vector $\mathbf{e}$ of t errors, where $t \leq (\delta_\Gamma - 1)/2$. So $\mathbf{y} = \mathbf{c} + \mathbf{e}$ for a codeword $\mathbf{c}$ in $C_\Omega(D, G)$ and $wt(\mathbf{e}) = t$. Let $I = \{i \mid e_i \neq 0\}$ be the set of error positions. We have explained in Section 2 how we can get the error values if we have found a set J of at most $d - 1$ elements which contains the error positions. We will discuss in the following the *basic* and *modified algorithm* for the code $C_\Omega(D, G)$ as presented in [48, 56, 100] and which are generalizations of the decoding algorithm of Arimoto-Peterson [3, 78] for RS codes. Under certain assumptions both algorithms find, for a received word $\mathbf{y}$, a rational function f which is zero at all error positions. Such functions are called *error-locator* functions. We do not assume that the set of zeros of an error-locator function is equal to the set of error positions, as is the case for error-locator polynomials for RS and BCH codes. On a curve of genus $\gamma > 0$, if one has found a function f which is zero at t prescribed points, then in most cases such a function has γ zeros more. This is the reason that the basic and modified algorithm

does not decode up to half the Goppa designed minimum distance.

Let F be any divisor with support which is disjoint from the support of D. Let Q be the divisor of error positions defined by $Q = \sum_{i \in I} P_i$. The set of error-locator functions in $L(F)$ is a subspace of $L(F)$ defined by imposing t linear conditions and is equal to $L(F - Q)$. We want to find a non-zero error-locator function in $L(F)$, so $L(F - Q)$ should be non-zero for all choices of Q. This is the case if the dimension of $L(F)$ is at least $t + 1$, which is satisfied if the degree of F is at least $t + \gamma$, by the Theorem of Riemann.

If $f \in L(F)$ and $g \in L(G-F)$, then $fg \in L(G)$, so the corresponding word $ev_D(fg)$ is an element of $C_L(D, G)$, which is the dual of $C_\Omega(D, G)$, thus $ev_D(fg)\mathbf{c}^T = 0$. If moreover f is zero at all error positions, then

$$\sum y_i f(P_i)g(P_i) = \sum e_i f(P_i)g(P_i) = \sum_{i \in I} e_i f(P_i)g(P_i) = 0.$$

Define for a received word $\mathbf{y}$ and a divisor F the kernel

$$K(\mathbf{y}, F) = \{f \in L(F) \mid \sum y_i f(P_i)g(P_i) = 0 \text{ for all } g \in L(G - F)\}.$$

Denote $K(\mathbf{y}, F)$ by $K(\mathbf{y})$ or $K(F)$ for short in situations where F or $\mathbf{y}$, respectively, is fixed. Thus $K(\mathbf{y}, F)$ is an object we can compute as soon as we receive the word $\mathbf{y}$. It contains all error-locator functions of $L(F)$ hence

$$L(F - Q) \subseteq K(\mathbf{y}, F).$$

The righthand side is an object we know, and the lefthand side is the part we want to know. We assumed that $\deg(F) \geq t+\gamma$. Suppose moreover that $\deg(G-F) > t+2\gamma-2$, then $C_\Omega(Q, G - F) = 0$. If $f \in K(\mathbf{y}, F)$, then

$$0 = \sum y_i f(P_i)g(P_i) = \sum_{i \in I} e_i f(P_i)g(P_i),$$

for all $g \in L(G - F)$. The word $\mathbf{w}$ with entries $w_i = e_i f(P_i)$ is an element of the dual of $C_L(Q, G - F)$, and we saw that this dual is zero. Thus $e_i f(P_i) = 0$ for all $i \in I$, so f is zero at all error positions. Therefore $L(F - Q) = K(\mathbf{y}, F)$.

Two assumptions were made on the degree of F. Firstly $\deg(F) \geq \gamma + t$ and secondly $\deg(G - F) > 2\gamma - 2 + t$, which conflict each other when t is too large. The largest possible value for t which meets both assumptions is $\lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor$. Let H be a parity check matrix for $C_\Omega(D, G)$. Let $t = \lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor$. Take a divisor F of degree $t + \gamma$ with support disjoint from D. One can show that such divisors always exist. We

consider the construction of such a divisor F and finding bases for the vector spaces $L(F)$ and $L(G-F)$ as part of the preprocessing of the following decoding algorithm.

Algorithm 4.1 Basic Algorithm $\mathcal{A}(F)$

1. Input: $\mathbf{y}$ (a received word)
2. Compute the Kernel $K(\mathbf{y}, F)$. If $K(\mathbf{y}, F) = 0$, then goto 6, else
3. Take a non-zero element f of $K(\mathbf{y}, F)$ and let $J = \{j \mid f(P_j) = 0\}$
4. Compute the solution space $L(\mathbf{y})$ of all $\mathbf{x}$ such that:
   $H\mathbf{x}^T = H\mathbf{y}^T$ and $x_j = 0$ for all $j \notin J$.
5. If $L(\mathbf{y})$ has the unique solution $\mathbf{x}_0$, then goto 7, else
6. Output: ?
7. Output: $\mathbf{y} - \mathbf{x}_0$

The assumptions on the degree of F also imply that a non-zero element of $K(\mathbf{y})$ has at most $(\delta_\Gamma - 1)$ zeros, so we can use Proposition 2.4 to get the error values. The computation of the kernel $K(\mathbf{y})$ requires solving a set of $l(F)$ linear homogeneous equations in $l(G-F)$ unknowns, since we have computed a basis for $L(F)$ and one for $L(G-F)$ in the preprocessing. The error values can be computed by solving a linear system of $n-k$ equations in at most $\delta_\Gamma - 1$ unknowns. Thus we have sketched the proof of the following proposition. See [48, 56, 100].

Theorem 4.2 *The basic algorithm corrects* $\lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor$ *errors. The complexity of the algorithm is at most* $\mathcal{O}(n^3)$.

Remark 4.3 The Goppa designed minimum distance is negative for functional codes $C_L(D, G)$ with $\deg(G) > \deg(D) = n$. The minimum distance of codes $C_L(D, G)$ such that G is *abundant*, that is to say equivalent to a divisor of the form $D + A$, where A is an effective divisor of degree a, is at least $\tau - a$, where τ is the gonality of the curve. If the curve has at least $n+2$ points, then abundant divisors could be used to show that there exists a divisor F such that the basic algorithm $\mathcal{A}(F)$ corrects $\lfloor (\delta_\Gamma - 1 - \gamma + \tau)/2 \rfloor$ errors. See [73].

Remark 4.4 If F is an arbitrary divisor, then there exists a divisor F' which is equivalent with F and has support disjoint from the support of D. But one can define $K(\mathbf{y}, F)$ even when the support of F is not disjoint from the support of D. If

$$K(\mathbf{y}, F) = \{f \in L(F) \mid \sum y_i(fg)(P_i) = 0 \text{ for all } g \in L(G-F)\},$$

then fg is well-defined at all points P_i, for all $f \in L(F)$ and $g \in L(G-F)$. This follows since $fg \in L(G)$ and G is assumed to have disjoint support from the support of D. See [18]. If $f \in L(F-Q)$ and $g \in L(G-F)$, then $fg \in L(G-Q)$. Now fg is zero at all error positions because G has disjoint support from the support of D and so

$$\sum y_i((fg)(P_i)) = \sum e_i((fg)(P_i)) = \sum_{i \in I} e_i((fg)(P_i)) = 0.$$

Therefore it still follows that $L(F-Q) \subseteq K(\mathbf{y}, F)$. The elements of $L(F-Q)$ are, by definition, error-locator functions. Notice that the assumption $\deg(G-F) > 2\gamma - 2 + t$ implies that $\Omega(G-F-Q) = 0$ for every divisor Q of t error positions. The following lemma gives a weakening of the assumptions for the basic algorithm. See [16, 18, 21].

Lemma 4.5 *Let H be a parity check matrix of the code $C_\Omega(D, G)$. Let $\mathbf{y}$ be a received word. Let $\mathbf{e}$ be the error vector and Q the divisor of error positions and let F be an arbitrary divisor.*
1) *(Existence) If $\Omega(G-F-Q) = 0$, then $K(\mathbf{y}, F) = L(F-Q)$, so all elements of the kernel $K(\mathbf{y}, F)$ are error locator functions. Moreover if $L(F-Q) \neq 0$, then there exists a non-zero element of $K(\mathbf{y}, F)$.*
2) *(Uniqueness) If $\Omega(G-F-Q) = 0$, $L(F-Q) \neq 0$ and f is a non-zero element of $K(\mathbf{y}, F)$ with set of zero positions $J = \{j \mid f(P_j) = 0\}$, then the set of equations $H\mathbf{x}^T = H\mathbf{y}^T$ and $x_j = 0$ for all $j \notin J$ has the unique solution $\mathbf{x} = \mathbf{e}$.*

Remark 4.6 The following heuristic argument shows that the basic algorithm corrects $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors most of the time, but it has not been made precise yet. If $t = \lfloor (\delta_\Gamma - 1)/2 \rfloor$ and F is a divisor of degree $t + \gamma$, then $L(F-Q) \neq 0$ for all divisors Q of t error positions. The set of divisors Q of degree t such that $\Omega(G-F-Q) \neq 0$ defines a hypersurface in the variety of all effective divisors of degree t. If this hypersurface is irreducble, then the percentage of error patterns of weight t where the basic algorithm fails for this reason is roughly $1/q$.

It may be the case that $\Omega(G-F-Q) \neq 0$ and we still have $K(\mathbf{y}, F) = L(F-Q)$. The following is more precise and comes from [18].

Proposition 4.7 *Let F be a divisor with support which is disjoint from the support of D. Then*

$$K(\mathbf{y}, F) = L(F-Q) \quad \textit{if and only if} \quad \mathbf{e} * C_L(Q, F) \cap C_\Omega(Q, G-F) = 0.$$

The basic algorithm depends on the choice of the divisor F. So one may try to find a divisor which has a larger dimension than is expected from its degree. We took the lower bound $\deg(F) + 1 - \gamma$ for the dimension of F, but $l(F) = \deg(F) + 1 - \gamma + i(F)$. A divisor F is called *special* if both $l(F)$ and $i(F)$ are not zero. The degree of a special

divisor is between 0 and $2\gamma - 2$. *Clifford's Theorem* gives an upper bound for the dimension of a divisor:

$$\text{if } \ 0 \leq \deg(F) \leq 2\gamma - 2, \ \text{ then } \ l(F) \leq \frac{\deg(F)}{2} + 1,$$

and equality holds only for certain divisors on hyperelliptic curves.

Remark 4.8 If we take for the basic algorithm a special divisor F and assume $l(F) > t$, instead of $\deg(F) \geq t + \gamma$, and moreover $\deg(G - F) > 2\gamma - 2 + t$, then $t \leq (\delta_\Gamma - 1)/3$. See [72].

Another way to improve the basic algorithm is to apply it with a sequence of divisors $F_1, \ldots, F_s$. In the *modified algorithm* [100] consider a sequence of divisors $F_1 \leq \cdots \leq F_s \leq G$ and take a non-zero function of $K(\mathbf{y}, F_i)$ for the smallest value of i such that the corresponding kernel is not zero, and continue as in the basic algorithm for F_i. If $G = jH$, then the sequence of divisors $F_i = iH$ for $i = 0, 1, \ldots, j$ is used.

Definition 4.9 Let m be the degree of H. Define

$$s(H) = \max\{\lfloor \frac{im + m + 1}{2} \rfloor - l(iH) \mid i \in \mathbb{Z}\}.$$

From [100] we quote:

Theorem 4.10 *The modified algorithm corrects* $\lfloor (\delta_\Gamma - 1)/2 - s(H) \rfloor$ *errors. The complexity of the algorithm is at most* $\mathcal{O}(n^3)$.

Remark 4.11 The number $s(H)$ is about $\gamma/4$ for plane curves. In the original decoding algorithm [48] the codes were defined on affine plane curves and were of the form $C_\Omega(D, jH)$, where H is the intersection divisor of the curve with the line at infinity. This can be done similarly for one point codes $C_\Omega(D, jP)$, where $H = P$ is a rational point and not equal to one of the P_i in D, and the sequence of divisors is now given by $F_i = iP$ for $i = 0, 1, \ldots, j$.

The modified algorithm was *extended* in [16, 18] by means of the following definition.

Definition 4.12 The *Clifford defect* $\sigma(E)$ of a divisor E such that $0 \leq \deg(E) \leq 2\gamma - 2$ is defined by

$$\sigma(E) = \frac{\deg(E)}{2} + 1 - l(E).$$

Remark 4.13 It follows from Clifford's Theorem that $\sigma(E)$ is not negative. Suppose that the designed minimum distance of $C_\Omega(D, G)$ is odd. If $\mathcal{E} = \{E_0, E_1, \ldots, E_{\gamma-1}\}$ is

a set of divisors such that $\deg(E_i) = 2\gamma - 2 - 2i$, then define $\sigma_0(\mathcal{E})$ to be the maximum over all $\sigma(E_i)$ for $i = 0, 1, \ldots, \gamma - 1$. Let $F_0, F_1, \ldots, F_\gamma$ be a sequence of divisors, with supports disjoint from the support of D, defined recursively by letting F_0 be a divisor of degree $(\delta_\Gamma - 1)/2$, and F_i be a divisor which is equivalent to $G - F_{i-1} - E_{i-1}$. Let $t = (\delta_\Gamma - 1)/2 - \sigma_0(\mathcal{E})$. Then $\Omega(G - F_0 - Q) = 0$, $L(F_\gamma - Q) \neq 0$, and for any divisor Q of t error positions,

$$\text{if} \quad L(F_i - Q) = 0, \quad \text{then} \quad \Omega(G - F_{i+1} - Q) = 0.$$

Thus we can take a non-zero function in $K(\mathbf{y}, F_i)$ for the smallest i such that the kernel is not zero and proceed with the basic algorithm for this F_i. If the designed minimum distance is even, define similiarly, a sequence of divisors $F_1, \ldots, F_\gamma$ for a given set $\mathcal{E} = \{E_1, \ldots, E_{\gamma-1}\}$ of divisors such that $\deg(E_i) = 2\gamma - 1 - 2i$, and $\sigma_1(\mathcal{E})$ is the maximum over all $\sigma(E_i)$ for $i = 1, \ldots, \gamma - 1$.

In this way the following theorem is obtained. See [16, 18].

Theorem 4.14 *The extended modified algorithm corrects $(\delta_\Gamma - 1)/2 - \sigma_i(\mathcal{E})$ errors, where $i = 0$ when the Goppa designed minimum distance is odd and $i = 1$ otherwise. The complexity of the algorithm is at most $\mathcal{O}(n^3)$.*

Remark 4.15 Only curves of genus 0 and (hyper)elliptic curves have the property that there exists a divisor H such that $\sigma(H) = 0$ or $\sigma(E) = 0$. The Clifford defect is computed for several curves [16, 52] and is about $\gamma/4$ for plane curves.

Example 4.16 The code $C(m)$ from the Klein quartic, see Example 3.7, has designed distance $\delta_\Gamma = m - 4$, and is therefore $t = \lfloor (m-5)/2 \rfloor$-error-correcting, but since $(\delta_\Gamma - 1 - \gamma)/2 = (m-8)/2$ the basic algorithm corrects $t-2$ errors when m is odd and $t-1$ errors when t is even. The modified algorithm corrects $t-1$ errors by Theorem 4.10, since $s(P) = 1$. Let $E_0 = 4P$, $E_1 = 2P$ and $E_2 = 0$, and let $\mathcal{E} = \{E_0, E_1, E_2\}$. Then $\sigma_0(\mathcal{E}) = 1$, the corresponding divisors F_i are $F_0 = tP$, $F_1 = (t+1)P$, $F_2 = (t+2)P$ and $F_3 = (t+3)P$, and the extended modified algorithm corrects $t-1$ errors by Theorem 4.14, when m is odd. If m is even one can take $E_1 = P$, $E_2 = 3P$ and $\mathcal{E} = \{E_1, E_2\}$. Then $\sigma_1(\mathcal{E}) = 1/2$, the corresponding divisors F_i are $F_1 = (t+1)P$, $F_2 = (t+2)P$ and $F_3 = (t+3)P$, and the extended modified algorithm corrects t errors.

Example 4.17 The code $C(11)$ from the Klein quartic, see Example 3.7, has designed distance $\delta_\Gamma = 7$, and is therefore 3-error-correcting, but since $\lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor = 1$ the basic algorithm only corrects one error. Let $F = 4P$. Then $G - F = 7P$. The functions $1, y$ form a basis for $L(4P)$, which is equal to $L(3P)$, and the functions $1, y, xy, y^2$ and x^2y give a basis for $L(7P)$. Let $\mathbb{F}_8 = \mathbb{F}_2[T]/(T^3 + T + 1)$ and let ξ be the class of T in

$\mathbb{F}_8$. Then ξ is a primitive element of $\mathbb{F}_8$.

The basic algorithm $\mathcal{A}(3P)$ corrects all received words with one error, (as well as those with at most three errors when the error positions lie on a line with equation $Y + cZ = 0$).

Now we consider the basic algorithm $\mathcal{A}(5P)$. Let (Q_1, Q_2) be a couple of two distinct points of the points $P_1, \ldots, P_{23}$ and let $Q = Q_1 + Q_2$. Then $L(5P - Q) \neq 0$. If $\Omega(6P - Q) \neq 0$, then $6P - Q$ is a canonical divisor and equivalent to $3P + R_1$. So $Q + R_1$ is equivalent to $3P$. Thus there exists a non-zero function in $L(3P)$ which is zero at R_1. So this function is equal to a non-zero scalar multiple of y and furthermore $Q_1 = R_3$, $Q_2 = R_1$. Now $C_L(Q, 5P) = C_L(Q, 6P) = C_\Omega(Q, 6P)$ is generated by $(1, 1)$. Therefore $K(\mathbf{y}, 5P) = L(5P - Q)$ for a received word $\mathbf{y}$ with Q as error positions and error vector $\mathbf{e}$ if and only if $\mathbf{e} * C_L(Q, 5P) \cap C_\Omega(Q, 6P) = 0$. But, by Proposition 4.7, this is equivalent to $e_1 \neq e_2$. Thus the number of decoding failures of the basic algorithm $\mathcal{A}(5P)$ to decode 2 errors is equal to 7 out of $\binom{23}{2} * 7^2$, the number of all possible error vectors with 2 errors, which is approximately 0.06 percent.

We will consider the following 2-error-pattern where $Q_1 = (1, \xi)$ and $Q_2 = (\xi^2, 1)$ and the corresponding error values are $e_1 = \xi$ and $e_2 = \xi^4$ with respect to $\mathcal{A}(5P)$. In order to compute $K(\mathbf{y}, 5P)$ we take $1, y, xy$ as basis for $L(5P)$ and $1, y, xy, y^2$ as basis for $L(6P)$. Now $K(\mathbf{y}, 5P)$ is by definition equal to the set of all $a_0 + a_1 y + a_2 xy$ such that:

$$\begin{pmatrix} \xi^2 & \xi & 1 \\ \xi & \xi^6 & \xi^4 \\ 1 & \xi^4 & 1 \\ \xi^6 & 0 & \xi^3 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = 0$$

and this has as basis $f = \xi^4 + \xi y + xy$. The zeros of f are $(1, \xi)$, $(\xi^2, 1)$, (ξ^5, ξ^5) and (ξ^6, ξ^6). The function f has pole-order 5 at P, so one expects a fifth zero, and indeed one has to count the zero $(\xi^2, 1)$ with multiplicity two, so

$$(f) = (1, \xi) + 2(\xi^2, 1) + (\xi^5, \xi^5) + (\xi^6, \xi^6) - 5P.$$

In order to find the error values we apply Proposition 2.4 where the parity check matrix H of $C(11)$ is obtained by evaluating the basis 1, y, xy, y^2, x^2y, xy^2, y^3, x^2y^2, xy^3 of

$L(11P)$. One gets the following system of linear equations:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ \xi & 1 & \xi^5 & \xi^6 \\ \xi & \xi^2 & \xi^3 & \xi^5 \\ \xi^2 & 1 & \xi^3 & \xi^5 \\ \xi & \xi^4 & \xi & \xi^4 \\ \xi^2 & \xi^2 & \xi & \xi^4 \\ \xi^3 & 1 & \xi & \xi^4 \\ \xi^2 & \xi^4 & \xi^6 & \xi^3 \\ \xi^3 & \xi^2 & \xi^6 & \xi^3 \end{pmatrix} \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{pmatrix} = \begin{pmatrix} \xi^2 \\ \xi \\ 1 \\ \xi^6 \\ \xi^4 \\ \xi^4 \\ 0 \\ 1 \\ \xi^3 \end{pmatrix}$$

which indeed has $(\xi, \xi^4, 0, 0)^T$ as the unique solution.

The modified algorithm corrects 2 errors for the code $C(11)$ by Theorem 4.10 with the divisors $3P$, $5P$ and $6P$, since $s(P) = 1$. Let $E_0 = 4P$, $E_1 = 2P$, and $E_2 = 0$; and let $\mathcal{E} = \{E_0, E_1, E_2\}$. Then $\sigma_0(\mathcal{E}) = 1$, and the corresponding divisors F_i are $F_0 = 3P$, $F_1 = 4P$, $F_2 = 5P$ and $F_3 = 6P$. The extended modified algorithm corrects 2 errors by Theorem 4.14 and is in fact the same as the modified algorithm, since $L(4P) = L(3P)$. In the following we will completely describe the instances where the modified algorithm fails to correct 3 errors with respect to the code $C(11)$.

Let (Q_1, Q_2, Q_3) be a 3-tuple of distinct points of the points $P_1, \ldots, P_{23}$ and let $Q = Q_1 + Q_2 + Q_3$. If the three points Q_1, Q_2, and Q_3 are collinear with R_1, then we have seen that $\mathcal{A}(3P)$ corrects the errors. If $L(5P - Q) = 0$, then $\Omega(5P - Q) = 0$, and $L(6P - Q)$ is always non-zero. Thus in this case $\mathcal{A}(6P)$ corrects these 3 errors. This leaves the case $L(3P - Q) = 0$ and $L(5P - Q) \neq 0$ and $\Omega(6P - Q) \neq 0$. Note that for every effective divisor R of degree 3 on a non-singular plane curve of degree 4, we have that $\Omega(R) \neq 0$ if and only if there exists a rational point E such that $E + R$ is canonical; that is to say, it is the intersection divisor of the curve with a line. Thus there exists an effective divisor R of degree 3 and a rational point E such that $E + R$ is collinear and $6P$ is equivalent with $Q + R$. So there exists a non-zero rational function g such that $(g) = Q + R - 6P$. If P is in the support of R, then $R = R' + P$ where R' is an effective divisor of degree 2, hence $(g) = Q + R' - 5P$, so g is of the form $a_0 + a_1 y + xy$. The function g is zero at R' which is collinear with P. So $a_0 = 0$, $g = (a_1 + x)y$ and $Q_1 = R_3 = (0 : 0 : 1)$, $Q_2 = R_1 = (1 : 0 : 0)$, and Q_3 and R' are on the line with equation $a_1 + X = 0$. If P is not in the support of R, then g is of the form $a_0 + a_1 y + a_2 xy + y^2$. So the quadratic form $a_0 + a_1 Y + a_2 XY + Y^2$ is zero at R which is a divisor of degree 3. That implies that the form is reducible and is of the form $(b + Y)(c + Y)$ with b and c elements of $\mathbb{F}_8$, or $a_0 = 0$ and $a_2 \neq 0$. If $g = (b + y)(c + y)$, then Q is on the line with equation $b + Y = 0$ and R is on

the line $c + Y = 0$. So $b + y$ is a non-zero element of $L(3P - Q)$, which contradicts the assumptions. If $g = (a_1 + a_2x + y)y$ and $a_2 \neq 0$, then $Q_1 = R_3$, $Q_2 = R_1$, and $Q_3 + R$ is the intersection divisor of the line $a_1 + a_2X + Y = 0$ with the quartic. Thus there are exactly 21 triples Q out of $\binom{21}{3}$, the number of 3 error positions such that $L(3P - Q) = 0$, $L(5P - Q) \neq 0$, and $\Omega(6P - Q) \neq 0$.

Now let $Q_1 = R_1$, $Q_2 = R_3$ and $Q_3 = (a, b)$, with $a^3b + b^3 + a = 0$, $ab \neq 0$ and $a, b \in \mathbb{F}_8$. The code $C_L(Q, 5P)$ is equal to $C_L(Q, 6P)$ and is generated by $(1, 1, 0)$ and $(0, 0, 1)$. The code $C_\Omega(Q, 6P)$ is generated by $(1, 1, 0)$. Therefore $K(\mathbf{y}, 5P) = L(5P-Q)$ for a received word $\mathbf{y}$ with Q as error positions and error vector $\mathbf{e}$ if and only if $\mathbf{e} * C_L(Q, 5P) \cap C_\Omega(Q, 6P) = 0$ if and only if $e_1 \neq e_2$. Thus the number of decoding failures is equal to $21 * 7^2$ out of $\binom{23}{3} * 7^3$, the number of all possible error vectors with 3 errors (which is less than 0.2 percent). In Section 8 we consider the 3-error pattern $Q_1 = R_1$, $Q_2 = R_3$ and $Q_3 = (1, \xi)$ and $e_1 = e_2 = e_3 = 1$ as an example where other algorithms succeed and the modified algorithm fails.

If we consider the 3-error pattern from [61] where $Q_1 = (1, \xi)$, $Q_2 = (\xi^2, 1)$ and $Q_3 = (\xi, 1)$, and the corresponding error values are 1, ξ^3 and ξ, then it turns out that $K(\mathbf{y}, 3P) = K(\mathbf{y}, 5P) = 0$ but $K(\mathbf{y}, 6P) = \langle \xi + \xi^3 y + y^2 \rangle$. So taking $f = \xi + \xi^3 y + y^2$, which has as the zeros $(\xi, 1)$, $(\xi^2, 1)$, $(\xi^4, 1)$ and $(1, \xi)$, (ξ^4, ξ), (ξ^5, ξ). The error values are obtained by Proposition 2.4. This example will be considered in Sections 8 and 10.

Example 4.18 Consider the code $C(23)$ from the Klein quartic. It has dimension 2, and designed minimum distance 19, and is therefore 9-error-correcting. However it is possible to choose an error pattern of weight 9 where the modified algorithm fails. Let Q_1, Q_2 and Q_3 be the affine points on the quartic and the line with equation $X + 1 = 0$, that is $Q_1 = (1, \xi)$, $Q_2 = (1, \xi^2)$ and $Q_3 = (1, \xi^4)$. Let $Q_4 = (\xi^6, \xi^3)$, $Q_5 = (\xi^6, \xi^4)$, $Q_6 = (\xi^4, \xi^3)$, $Q_7 = (\xi, \xi^6)$, $Q_8 = (\xi^3, \xi^3)$ and $Q_9 = (\xi^2, 1)$. These 6 points together with the points R_1 and R_3 lie on the intersection of the quartic and the quadric with equation $Y^2 + \xi^5 XY + \xi^3 YZ + \xi XZ = 0$. See [42, 43]. Put $Q = \sum_{i=1}^{9} Q_i$. The function $f = y(x + 1)(y^2 + \xi^5 xy + \xi^3 y + \xi x)$ has as divisor $Q + 2R_3 - 11P$, so f is a non-zero element of $L(11P - Q)$. Now $L(10P - Q) = 0$, since otherwise there exists a rational point E such that $10P \equiv Q + E$. Therefore $11P \equiv P + Q + E$, which gives $Q + 2R_3 \equiv P + Q + E$. So $2R_3 \equiv Q + E$, in contradiction to the fact that 2 is a gap at R_3 (or that the curve is not hyperelliptic). The vector space $\Omega(12P - Q)$ is not zero, since $12P - Q \equiv P + 2R_3 \leq P + 3R_3$ is a canonical divisor. Finally it can be seen that $C_\Omega(Q, 12P)$ is generated by $\mathbf{e} = (\xi^3, \xi, \xi^3, \xi, \xi, \xi^2, \xi^3, \xi^2, 1)$. Thus since it is obvious that the all ones vector is an element of $C_L(Q, 11P)$ we have that $\mathbf{e} * C_L(Q, 11P) \cap C_\Omega(12P, Q) \neq 0$, which means that the kernel $K(\mathbf{e}, 11P)$ contains functions which are not error locators by Proposition 4.7.

Example 4.19 We refer to [16, 18] for a worked out example of a Hermitian code where the number of decoding failures of the modified algorititthm is computed.

Problem 4.20 What is the number of correctable error patterns when the basic and modified algorithm is used ? A start has been made to investigate this question [16, 18, 47, 74], but a thorough investigation is still lacking.

5 Decoding and the Jacobian of a curve

We give an upper bound for the number s of divisors $F_1, \ldots, F_s$ which are needed in order that for each received word $\mathbf{y}$, with at most $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors, at least one of the basic algorithms $\mathcal{A}(F_i)$ will correct $\mathbf{y}$. See [72, 109].

We consider the decoding of codes of the form $C_\Omega(D, G)$. Assume $\deg(G) > 2\gamma - 2$. Let $t = \lfloor (\delta_\Gamma - 1)/2 \rfloor$. Suppose that δ_Γ is odd for simplicity. Let $F_1, \ldots, F_s$ be a collection of effective divisors of degree $t + \gamma$, then $l(F_i) > t$, so $L(F_i - Q)$ is not zero for all i and all divisors Q of t error positions. Let $\mathbf{y}$ be a received word with divisor of error positions Q. If there exsits an index i such that $\Omega(G - F_i - Q) = 0$, then $\mathcal{A}(F_i)$ will decode $\mathbf{y}$ by Lemma 4.5. So suppose on the contrary that $\Omega(G - F_i - Q)$ is not zero for all i. Then there exists a differential form ω_i such that $(\omega_i) \geq G - F_i - Q$, that is to say there exists an effective divisor E_i such that

$$(\omega_i) = E_i + G - F_i - Q.$$

Comparing the degrees of the divisors in this equality gives that $\deg(E_i) = \gamma - 1$ for all i. Two divisors are *equivalent* if their difference is a principal divisor, this indeed defines an equivalence relation on the set of divisors and we denote the class of a divisor D by $[D]$. All canonical divisors, that is divisors of non-zero differential forms, are equivalent and form the canonical class $\mathcal{K}$. Thus $[E_i - F_i] = \mathcal{K} - [G - Q]$ does not depend on i, so $[E_i - E_j] = [F_i - F_j]$ for all i and j. The set of all divisors on a curve is a free abelian group with the set of places of the curve as its generator. The principal divisors form a subgroup so we can form the quotient of divisors modulo the principal divisors and we get in this way the *Picard* group, its elements being divisor classes. The divisors of degree zero on the curve $\mathcal{X}$ modulo the principal divisors form a subgroup of the Picard group which is called the *class group* or the *Jacobian* of the curve and we will denote it by $J(\mathcal{X})$. We denote the set of all effective divisors of degree k by $\mathbb{D}_k$. So E_i is an element of $\mathbb{D}_{\gamma-1}$ and F_i is an element of $\mathbb{D}_{t+\gamma}$, and furthermore $E_i - E_j$ and $F_i - F_j$ represent the same class in $J(\mathcal{X})$ for all i, j. Now consider the map

$$\psi_k^s : \mathbb{D}_k^s \longrightarrow J(\mathcal{X})^{s-1},$$

defined by $\psi_k^s(D_1, \ldots, D_s) = ([D_1 - D_2], \ldots, [D_{s-1} - D_s])$. It follows that

$$\psi_{\gamma-1}^s(E_1, \ldots, E_s) = \psi_{t+\gamma}^s(F_1, \ldots, F_s).$$

It is an easy consequence of the Theorem of Riemann-Roch that the map ψ_k^s is surjective for all $k \geq \gamma$ and s. Now suppose that $\psi_{\gamma-1}^s$ is not surjective, then we could have started with an s-tuple $(F_1, \ldots, F_s)$ such that $\psi_{t+\gamma}^s(F_1, \ldots, F_s)$ is not in the image of $\psi_{\gamma-1}^s$. In this way we would have obtained a contradiction with the assumption that $\Omega(G - F_i - Q)$ is not zero for all i. Thus there exists an i such that $\Omega(G - F_i - Q) = 0$, and therefore $\mathcal{A}(F_i)$ corrects the errors of $\mathbf{y}$. We have sketched the proof from [72] of the following proposition.

Proposition 5.1 *Let $(F_1, \ldots, F_s)$ be an s-tuple of effective divisors of degree $t+\gamma$ such that $\psi_{t+\gamma}^s(F_1, \ldots, F_s)$ is not in the image of $\psi_{\gamma-1}^s$. Then for every received word $\mathbf{y}$ with at most $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors there is at least one i such that the basic algorithm $\mathcal{A}(F_i)$ corrects $\mathbf{y}$.*

The objects $\mathbb{D}_k$ and $J(\mathcal{X})$ are finite for curves over finite fields and can be computed by means of the *zeta function* of the curve $\mathcal{X}$ which is defined by

$$Z(\mathcal{X}, T) = \sum_{k=0}^{\infty} a_k T^k,$$

where a_k is the number of elements of $\mathbb{D}_k$. The zeta function is a rational function

$$Z(\mathcal{X}, T) = \frac{P(T)}{(1-T)(1-qT)}.$$

The numerator $P(T)$ is a polynomial in T with integer coefficients of degree 2γ and

$$P(T) = \prod_{i=1}^{\gamma} (1 - \alpha_i T)(1 - \bar{\alpha}_i T),$$

where the modulus of the α_i is equal to $\sqrt{q}$. The number of elements of $J(\mathcal{X})$ is equal to $P(1)$. With the help of these properties of the zeta function it was shown that $\psi_{\gamma-1}^{2\gamma}$ is not surjective for all maximal curves [72], for all curves when $q \geq 37$, and for all curves of genus $\gamma \geq \gamma_0(q)$ when $q \geq 16$ [109]. As a result of the above we have:

Theorem 5.2 *There exist s divisors $F_1, \ldots, F_s$ such that for every received word with at most $\lfloor (\delta_\Gamma - 1)/2 \rfloor$ errors, at least one of the basic algorithms $\mathcal{A}(F_1), \ldots, \mathcal{A}(F_s)$ corrects the error with respect to the code $C_\Omega(D, G)$ over $\mathbb{F}_q$, for all $q \geq 37$ and divisors G such that deg$(G) > 2\gamma - 2$. Furthermore $s = \mathcal{O}(n)$ and the complexity of the algorithm is $\mathcal{O}(n^4)$ for $n \to \infty$.*

Remark 5.3 In [16, 21] it was shown that the condition $\deg(G) \geq 4\gamma - 2$ in [72] could be replaced by $\deg(G) > 2\gamma - 2$.

Remark 5.4 If the curve has gonality τ and has at least two rational points, then $\psi^{\tau}_{\tau-2}$ is not surjective and there exist 2τ divisors $F_1, \ldots, F_{2\tau}$, which are known explicitly, such that the corresponding basic algorithms, run in parallel, correct $\lfloor (\delta_\Gamma - 1 - \gamma + \tau)/2 \rfloor$ errors. See [21].

Remark 5.5 Several authors have investigated the smallest value of s such that $\psi^{s}_{\gamma-1}$ is not surjective [11, 42, 43, 81].

Remark 5.6 In the basic algorithm it is assumed that the divisor F has support disjoint from the support of D. But if $(F_1, \ldots, F_s)$ is an s-tuple of divisors which satifies the conditions of Proposition 5.1, then there exists an s-tuple $(F'_1, \ldots, F'_s)$ such that F'_i is equivalent with F_i and has support disjoint from the support of D, for all i. This new s-tuple has all required properties. Alternatively we allow the F_i to have a non-empty intersection with D and apply Remark 4.4. One may consider the actual construction of such an s-tuple as part of the preprocessing of the algorithm, but it is done effectively only for hyperelliptic curves of genus at most 4 [58, 59] and the Klein quartic over $\mathbb{F}_8$ [43, 81]. It looks like a difficult (possibly hopeless) problem in general, which is, moreover, obsolete from the decoding point of view, given the solutions of the decoding problem [17, 22, 23] which we will discuss in Sections 7 and 8.

Example 5.7 The zeta function of the Klein quartic over $\mathbb{F}_8$ is equal to

$$\frac{(1 + 5T + 8T^2)^3}{(1 - T)(1 - 8T)}$$

so its Jacobian has 14^3 elements. One can take $s = 3$ and if the Goppa designed minimum distance is equal to $2t + 1$, then the following 3 divisors

$$\begin{gathered} F_1 = K + R_3 - (1, \xi) + (t - 1)P \\ F_2 = (1, \xi^4) + (\xi^3, \xi^2) + (\xi^4, \xi^3) + (\xi^5, \xi^1) + (t - 1)P \\ F_3 = K + R_1 - R_3 + (t - 1)P, \end{gathered}$$

where K is a canonical divisor, have the property that for every received word with at most t errors at least one of the algorihtms $\mathcal{A}(F_1)$, $\mathcal{A}(F_2)$ or $\mathcal{A}(F_3)$ will correct the errors. See [42, 43].

6 The key equation

Around the same time that the paper [48] on the decoding algorithm for plane curves was published, Porter [79] found another decoding algorithm which was a generalization of solving the key equation of classical Goppa codes by Euclid's algorithm in the

ring of polynomials in one variable [104]. The correctness of the proposed algorithm was proved in [20, 21, 80, 94, 95, 96].

One can view the ring of polynomials in one variable as the ring of rational functions on the projective line with only poles at the point at infinity. The ring of polynomials in one variable is replaced by the ring $K_\infty(P)$ of rational functions on the curve with only poles at a fixed rational point P, where P is not equal to one of the points P_i used to construct the geometric Goppa code. The *degree* or *weight* of a function $f \in K_\infty(P)$ is by definition equal to $-v_P(f)$, the pole order of f at P, and is denoted by $\deg(f)$ or $w(f)$. We will use the notation $w(f)$ in the rest of this paper. The ring $K_\infty(P)$ with this weight function is not an *Euclidean domain* unless the genus of the curve is zero, but it still has very similar properties. For all $f, g \in K_\infty(P)$ we have that:

$$w(fg) = w(f) + w(g),$$

$$w(f+g) \leq \max\{w(f), w(g)\},$$

$$\text{if } w(f) = w(g), \text{ then there exists a } \lambda \in \mathbb{F}_q^* \text{ such that } w(f - \lambda g) < w(f).$$

Firstly one can show that for arbitrary geometric Goppa codes one may assume, when the number of rational points on the curve is greater than n, that the divisor G in the definition of the code $C_\Omega(D, G)$ is of the form $E - \mu P$, where E is an effective divisor and μ a positive integer. Secondly one can show that there always exist n independent differentials $\varepsilon_1, \ldots, \varepsilon_n \in \Omega(-D - \mu P)$ such that $\mathrm{res}_{P_i}(\varepsilon_j)$ is 1 if $i = j$ and 0 otherwise. For every differential $\omega \in \Omega(E - \mu P - D)$ we have

$$\omega = \sum \mathrm{res}_{P_i}(\omega)\varepsilon_i.$$

If we let $\varepsilon(\mathbf{y}) = \sum y_i \varepsilon_i$, then $\mathrm{res}_D(\varepsilon(\mathbf{y})) = \mathbf{y}$. Therefore the map ε is the right inverse of res_D and

$$\varepsilon(\mathbf{y}) \in \Omega(E - \mu P - D) \text{ if and only if } \mathbf{y} \in C_\Omega(D, E - \mu P).$$

Suppose that E is the divisor of zeros of a function $g \in K_\infty(P)$ which has not a zero at P_i for all i. We want to define the *syndrome* of a received word. In order to represent the syndrome as a rational function, one proves the existence of a particular differential η first. The syndrome $S(\mathbf{y})$ of a received word $\mathbf{y}$ is now defined as follows.

$$S(\mathbf{y})\eta = \sum y_i \frac{g(P_i) - g}{g(P_i)} \varepsilon_i.$$

The syndrome is an element of the ring $K_\infty(P)$. If E is the divisor of zeros of $g \in K_\infty(P)$, then

$$\mathbf{y} \in C_\Omega(D, E - \mu P) \text{ if and only if } S(\mathbf{y}) \equiv 0 \pmod{g}.$$

For simplicity assume that η is a differential such that $(\eta) = (2\gamma - 2)P$. Notice that this assumption is satisfied for the Hermitian curves but not for the Klein quartic.

Now one searches for *solutions* of the *key equation*, that is for pairs (f, r) with $f, r \in K_\infty(P)$ such that there exists an $a \in K_\infty(P)$ with the property

$$fS(\mathbf{y}) = r + ag.$$

A solution is called *valid* if moreover $w(r) - w(f) \leq 2\gamma - 2 + \mu$. A valid solution (f, r) is called *minimal* if $w(f)$ is minimal among all the weights of f' such that (f', r') is a valid solution. In this way we get [20, 21, 80]:

Theorem 6.1 *Let $\mathbf{y} \in \mathbb{F}^n$ with $\mathbf{y} = \mathbf{c} + \mathbf{e}$, where $\mathbf{c}$ is a code word of $C_\Omega(D, E - \mu P)$ and $\mathbf{e}$ is an error vector.*
1) *(Existence) There exists a valid solution (f, r) of the key equation of $\mathbf{y}$ such that*

$$\frac{r}{f}\eta \in \Omega(-D - \mu P) \quad \text{and} \quad res_D(\frac{r}{f}\eta) = \mathbf{e}.$$

2) *(Uniqueness) Let $t = (\delta_\Gamma - 1)/2 - \sigma$, where δ_Γ is the Goppa designed minimun distance of the code and σ is the Clifford defect of P. Suppose $wt(\mathbf{e}) \leq t$. If (f, r) is a minimal valid solution of the key equation of $\mathbf{y}$, then*

$$\frac{r}{f}\eta \in \Omega(-D - \mu P) \quad \text{and} \quad res_D(\frac{r}{f}\eta) = \mathbf{e}.$$

Example 6.2 The above is a direct generalization of classical Goppa codes, see Example 3.4. Consider the projective line. Let $\alpha_1, \ldots, \alpha_n$ be n distinct elements of $\mathbb{F}_q$. Let $P_i = (\alpha_i : 1)$, $P = (1 : 0)$ the point at infinity and $D = P_1 + \cdots + P_n$. Then we can take $\eta = dX$ and $\varepsilon_i = dX/(X - \alpha_i)$ to get the desired properties.

Remark 6.3 The explicit computation of the differentials $\varepsilon_1, \ldots, \varepsilon_n$ and finding formulas for the syndromes is in general quite elaborate, but we consider this as part of the preprocessing of the algorithm. For the Hermitian codes this has been done in [95, 96].

Remark 6.4 Euclid's algorithm gives in the case of classical Goppa codes a sequence of solutions of the key equation, and the first valid solution in this sequence is also a minimal valid solution. The ring $K_\infty(P)$ is not a Euclidean ring whenever the genus is not zero. The sequence of solutions in the Euclidean algorithm is replaced by an algorithm giving the so called *subresultant sequence*. See [30, 79, 94, 96].

Example 6.5 The above algorithm is worked out in detail for Hermitian curves and their codes in [80, 94, 95, 96]

The equivalence of the modified algorithm and the decoding by solving the key equation for one point codes was shown in [20, 21] and will be discussed in the next section.

7 Improvement of the modified algorithm

In this section we compare Porter's and the modified algorithm [20, 22] and give Ehrhard's algorithm [22] which gives an effective solution of the decoding of AG codes.

The value $f(P)$ of a rational function f at a point P is only defined in case that f does not have a pole at P. We have seen in Remark 4.4 how we can redefine the kernel $K(F)$ in such a way that the divisor F is allowed to have a non-empty intersection with the support of D. We follow the alternative of [22] where the fact is used that the residue map is always defined for every differential form and any point.

The code $C_\Omega(D, G)$ is the image under the residue map

$$\mathrm{res}_D : \Omega(G - D) \to \mathbb{F}_q^n$$

and this map is injective if we assume $\deg(G) > 2\gamma - 2$. There exists a divisor $G' \leq G$ such that $\mathrm{res}_D : \Omega(G' - D) \to \mathbb{F}_q^n$ is surjective. Moreover there exists a linear map

$$\mathbb{F}_q^n \longrightarrow \Omega(G' - D), \;\; \mathbf{y} \mapsto \eta_{\mathbf{y}}$$

such that $\mathrm{res}_D(\eta_{\mathbf{y}}) = \mathbf{y}$ for all $\mathbf{y}$. The map ε of the last section is a more explicit description of such a map. Let F be a divisor such that $\deg(F) < \delta_\Gamma$ or equivalently $\deg(G - F) > 2\gamma - 2$. Then

$$\Omega(G - D - F) \cap \Omega(G' - F) = \Omega(G - F) = 0.$$

Thus $\Omega(G - D - F) + \Omega(G' - F)$ is a direct sum, that is to say for every element ω in this sum there exist unique $\alpha \in \Omega(G - D - F)$ and $\beta \in \Omega(G' - F)$ such that $\omega = \alpha + \beta$. Let

$$\pi : \Omega(G - D - F) + \Omega(G' - F) \longrightarrow \Omega(G' - F)$$

be the projection along $\Omega(G - D - F)$. Define

$$K'(\mathbf{y}, F) = \{f \in L(F) \mid f\eta_{\mathbf{y}} \in \Omega(G - D - F) + \Omega(G' - F)\},$$

which we denote by $K'(F)$ for short when the received word $\mathbf{y}$ is fixed.

Remark 7.1 It is shown in [20, 21] that $K(F) = K'(F)$ when F has support disjoint from the support of D and $\deg(F) > \max\{\deg(G'), \deg(G - D)\}$. It was noted in [18] that one can always work with $K(F)$ instead of $K'(F)$, see also Remark 4.4. We denote the dimension of $K(F)$ by $k(F)$.

The following result, which is similar to Theorem 6.1 (2), is from [20, 21].

Proposition 7.2 *Suppose* $L(F-Q) \neq 0$ *and* $\Omega(G-F-Q)=0$. *If* f *is a non-zero element of* $K(F)$, *then*

$$res_D\left(\frac{\pi(f\eta_{\mathbf{y}})}{f}\right) = \mathbf{e}$$

is the error vector of $\mathbf{y}$.

Remark 7.3 We compare the modified algorithm and Porter's algorithm in the special case that $G = mP$ and there exists a differential η with divisor $(2\gamma-2)P$. See [20, 21]. If f is a non-zero element of $K(jP)$ for the smallest j such that $K(jP)$ is not zero, then there exists an $r \in K_\infty(P)$ such that (f,r) is a valid solution of the key equation. Conversely, if (f,r) is a valid solution of the key equation and $j = \deg(f)$, then f is a non-zero element of $K(jP)$, and j is the smallest integer such that $K(jP)$ is not zero.

Remark 7.4 Let $t = \lfloor(\delta_\Gamma - 1)/2\rfloor$. In the approach of Section 5 the sequence of divisors $(F_1,\ldots,F_s)$ is fixed and should be found before the algorithm is executed. It works for all received words with at most t errors. In the following we explain *Ehrhard's algorithm* [22] which produces a sequence of divisors $(F_1,\ldots,F_s)$ which depends on the received word $\mathbf{y}$ and has the property that the basic algorithm $\mathcal{A}(F_s)$ decodes $\mathbf{y}$ when there are at most t errors. In this way the elaborate problem of constructing the sequence of divisors is circumvented, although this algorithm still has the complexity of solving a system of linear equations. Consider the following algorithm $\mathcal{B}(F)$ which depends on the choice of a divisor F.

Algorithm 7.5 $\mathcal{B}(F)$

1. Input: $\mathbf{y}$
2. Set $j := 1$ and $F_1 := F$
3. Look for an index i such that $k(F_j - P_i) \leq k(F_j) - 2$. If there is such an i, then: set $F_{j+1} = F_j - P_i$, $j := j+1$ and goto 3, else
4. If $k(F_j) = 0$, then goto 5 , else goto 6
5. Output: ?
6. $\mathbf{e} := \mathrm{res}_D(\pi(f\eta_{\mathbf{y}})/f)$ for some non-zero $f \in K(F_j)$
7. Output: $\mathbf{y} - \mathbf{e}$

An alternative of the above algorithm is to apply the basic algorithm $\mathcal{A}(F_j)$ at line 6. See [18]. We quote the following proposition from [22].

Theorem 7.6 *Let $\mathcal{X}$ be a curve of genus γ, and $C = C_\Omega(D, G)$ an algebraic-geometric code of Goppa designed minimum distance $\delta_\Gamma \geq 6\gamma$. Let F be any divisor of degree $2\gamma + t$, where $t = \lfloor (\delta_\Gamma - 1)/2 \rfloor$. Then $\mathcal{B}(F)$ is a decoder for C which corrects t errors with complexity $\mathcal{O}(n^3)$.*

Remark 7.7 Using Clifford's Theorem one can show that the assumption "$\delta_\Gamma \geq 6\gamma$" can be replaced by the weaker condition "$\delta_\Gamma \geq 4\gamma$". See [18].

Remark 7.8 If one applies both the algorithms $\mathcal{B}(F)$ and $\mathcal{B}(G - F)$ for a divisor F such that $\deg(F) = \gamma + t$, then it is enough to assume that $\delta_\Gamma \geq 4\gamma - 2\tau$, where τ is the gonality of the curve. See [18]. Moreover it is shown in an example that this cannot be improved.

In the next section another solution of the decoding problem of AG codes will be considered.

8 Majority voting for unknown syndromes

In this section it is shown how for one point codes one can extend the parity check matrix H with rows $\mathbf{h}_i, i = 1, \ldots, n-k$ to an $n \times n$ matrix $\hat{H}$ with rows $\mathbf{h}_i, i = 1, \ldots, n$. This is done in such a way that the *unknown syndromes* $s_i(\mathbf{e}) = \mathbf{h}_i \mathbf{e}^T$ $i > n - k$ can be obtained recursively from known syndromes $s_i(\mathbf{e}) = s_i(\mathbf{y}), 1 \leq i \leq n - k$ by a *majority vote*. See [23, 17, 18] and also [53, 75]. This procedure has its predecessor in what is called *threshold, majority logic* [66] or *sequential code reduction* decoding [82] and the decoding of cyclic codes beyond the BCH error-correcting capacity [28].

Let $(\rho_i | i \in \mathbb{N})$ be the non-gap sequence of P, that is to say $\rho_i < \rho_{i+1}$ and $\{\rho_i \mid i \in \mathbb{N}\}$ is the semigroup of non-gaps at P. The code $C_\Omega(D, mP)$ is denoted by $C(m)$. Abbreviate $C(\rho_r)$ by C_r from now on. Let g_i be a rational function which has a pole of order ρ_i at P and no other poles. Then $g_1, \ldots, g_r$ is a basis for $L(\rho_r P)$. Let $\mathbf{h}_i = ev_D(g_i)$. Let H_r be the $r \times n$ matrix with $\mathbf{h}_i, 1 \leq i \leq r$ as rows. Then H_r is a parity check matrix of the code C_r. Note that the rows of H_r need not to be independent. Define a matrix of syndromes $(s_{ij}(\mathbf{e}) \mid i, j \in \mathbb{N})$ with respect to an error vector $\mathbf{e}$ by:

$$s_{ij}(\mathbf{e}) = \sum_{l=1}^{n} e_l g_i(P_l) g_j(P_l).$$

If $\mathbf{y}$ is a received word with error vector $\mathbf{e}$ with respect to the code C_r and $\rho_i + \rho_j \leq \rho_r$, then $g_i g_j \in L(\rho_r P)$, so $s_{ij}(\mathbf{e}) = s_{ij}(\mathbf{y})$. Thus $s_{ij}(\mathbf{e})$ is a known entry of the matrix of syndromes for all i, j such that $\rho_i + \rho_j \leq \rho_r$. Abbreviate $s_{ij}(\mathbf{e})$ and $s_r(\mathbf{e})$ by s_{ij} and s_r, respectively.

Definition 8.1 Define the set of pairs N_r by

$$N_r = \{\ (i,j) \in \mathbb{N}^2 \mid \rho_i + \rho_j = \rho_{r+1}\}.$$

Let n_r be the number of elements of N_r and define the *Feng-Rao designed minimum distance* $\delta_{FR}(r)$ of C_r by:

$$\delta_{FR}(r) = \min\ \{\ n_s \mid s \geq r\ \}.$$

Remark 8.2 From the fact that decoding by majority voting for unknown syndromes corrects $\lfloor (\delta_{FR}(r)-1)/2 \rfloor$ errors, which is discussed later in this section, it follows that the minimum distance of C_r is at least $\delta_{FR}(r)$, but it is also possible to give a direct proof first. See [53, 75].

Remark 8.3 One has that $\delta_{FR} \geq \delta_\Gamma$ and equality holds if $r > 3\gamma - 2$. Note that the definition of δ_{FR} depends only on the semigroup of non-gaps of P. It has been computed for semigroups with two generators and more generally for *telescopic semigroups* [53]. In many examples δ_{FR} is strictly greater than δ_Γ for small r.

Example 8.4 The non-gap sequence at the point P of the Suzuki curve of Example 3.8 is generated by $8, 10, 12$ and 13. The following table compares the Goppa and the Feng-Rao designed minimum distance of C_r in the range $4 \leq r \leq 21$. If $r \geq 41$, then $\delta_{FR} = \delta_\Gamma$.

r	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
n_r	2	3	4	5	4	4	4	5	4	7	8	6	8	8	9	10	12	8
δ_{FR}	2	3	4	4	4	4	4	4	4	6	6	6	8	8	8	8	8	8
δ_Γ	-	-	-	-	-	-	-	-	-	0	1	2	3	4	5	6	7	8

Remark 8.5 The entries of the matrix of syndromes with index $(i,j) \in N_r$ are the first unknown syndromes we encounter with respect to the code C_r. As soon as we know one s_{ij} with $(i,j) \in N_r$, we know all the others $s_{i'j'}$ with $(i',j') \in N_r$, since each one of the functions $g_i g_j$, $g_{i'} g_{j'}$ or g_{r+1} is a generator of the one dimensional vector space $L(\rho_{r+1}P)$ modulo $L(\rho_r P)$. In other words, there exist $\mu_{ij}, \mu_{ijl} \in \mathbb{F}_q$ such that $\mu_{i,j}$ is not zero and

$$g_i g_j = \mu_{ij} g_{r+1} + \sum_{l \leq r} \mu_{ijl} g_l$$

for all i,j with $\rho_i + \rho_j = \rho_{r+1}$. Therefore

$$s_{ij} = \mu_{ij} s_{r+1} + \sum_{l \leq r} \mu_{ijl} s_l$$

and this relation is the same for all error vectors. Consider the matrix

$$\mathcal{S}(i,j) = (s_{i',j'} \mid 1 \leq i' \leq i, 1 \leq j' \leq j).$$

If $\rho_i + \rho_j = \rho_{r+1}$, then all entries of this matrix, except s_{ij}, are known.

$$\begin{pmatrix} s_{1,1} & \dots & s_{1,j-1} & s_{1,j} \\ \vdots & & \vdots & \vdots \\ s_{i-1,1} & \dots & s_{i-1,j-1} & s_{i-1,j} \\ s_{i,1} & \dots & s_{i,j-1} & ? \end{pmatrix}.$$

Remark 8.6 If $\rho_i + \rho_j = \rho_r$, then $\mathcal{S}(i,j)$ is a matrix of the linear map from $L(\rho_j P)$ to $L(\rho_i P)$ which is used to compute the kernel $K(\mathbf{y}, \rho_j P)$ in the basic algorithm $\mathcal{A}(\rho_j P)$ for the code $C_r = C_\Omega(D, \rho_r P)$. The rectangular submatrices $\mathcal{S}(i,j)$ with $\rho_i + \rho_j = \rho_r$, is the collection of matrices which one encounters in the modified algorithm for C_r. If g is a non-zero error locator function in $L(\rho_j P)$ and $g = \sum_{l=1}^{j} \lambda_l g_l$, then the columns of the matrix $\mathcal{S}(i,j)$ are dependent:

$$\sum_{l=1}^{j} \lambda_l s_{k,l} = 0 \quad \text{for all} \quad 1 \leq k \leq i.$$

Definition 8.7 If $(i,j) \in N_r$, that is to say $\rho_i + \rho_j = \rho_{r+1}$, and the three matrices $\mathcal{S}(i-1,j-1)$, $\mathcal{S}(i-1,j)$ and $\mathcal{S}(i,j-1)$ have equal rank, then (i,j) is called a *candidate* with respect to C_r. If (i,j) is a candidate, then there is a unique value s'_{ij} to assign to the unknown entry s_{ij} such that the matrices $\mathcal{S}(i,j)$ and $\mathcal{S}(i-1,j-1)$ have equal rank. The element s'_{ij} is called the *predicted* or *candidate value* of the unknown syndrome s_{ij}. A candidate is called *correct* or *true* when $s'_{ij} = s_{ij}$ and *incorrect* or *false* otherwise. Denote the number of true candidates by T and the number of false candidates by F. An entry (i,j) is called a *discrepancy* if the three matrices $\mathcal{S}(i-1,j-1)$, $\mathcal{S}(i-1,j)$ and $\mathcal{S}(i,j-1)$ have equal rank and the matrices $\mathcal{S}(i,j)$ and $\mathcal{S}(i-1,j-1)$ do not have equal rank.

Remark 8.8 The discrepancies are the *pivots* if one applies *Gaussian elimination* (without interchanging rows or columns) to the matrix of syndromes. The total number of discrepancies is equal to the rank of the matrix of syndromes. Furthermore this matrix can be written as a triple product of matrices with the diagonal matrix in the middle and the entries of the error-vector $\mathbf{e}$ on the diagonal. See [3, 78]. Therefore the total number of discrepancies is at most equal to $wt(\mathbf{e})$, the number of errors.

Let $\mathbf{y}$ be a received word with error vector $\mathbf{e}$ which has at most $(n_r - 1)/2$ errors with respect to the code C_r. Then all syndromes s_{ij} such that $\rho_i + \rho_j \leq \rho_r$ are known,

and the remaining syndromes are unknown. Denote the number of known discrepancies by K. A candidate is incorrect if and only if it is a discrepancy, so

$$K + F \leq \text{ total number of discrepancies } \leq wt(\mathbf{e}).$$

If entry (i, j) is a known discrepancy, then all entries (i, j') in the ith row with $j' > j$, and all entries (i', j) in the jth column with $i' > i$ are non-candidates. If $(i, j) \in N_r$ is not a candidate, then there is at least one known discrepancy in the same row or column. Thus the number of pairs $(i, j) \in N_r$ which are non-candidates is at most $2K$. The number of pairs $(i, j) \in N_r$ which are candidates is equal to $T + F$. Therefore

$$n_r = \text{ \# candidates } + \text{ \# non-candidates } \leq (T + F) + 2K.$$

Furthermore we assumed that

$$\text{\# errors } = wt(\mathbf{e}) \leq (n_r - 1)/2.$$

Combining the above four inequalities gives

$$F < T.$$

There is no direct way to see whether a candidate is true or false. But we assigned a predicted value s'_{ij} of the syndrome $s_{i,j}$ to every candidate, and this gives a predicted value or vote $s_{r+1}(i, j)$ for s_{r+1} by Remark 8.5. All T true candidates yielded the same, correct, value for s_{r+1}. Thus a proof of the following has been given.

Proposition 8.9 *If the number of errors of a received word with respect to the code C_r is at most $(n_r - 1)/2$, then the majority of the candidates vote for the correct value of s_{r+1}.*

In this way all unknown syndromes can be found by induction and from this the error vector is obtained, as was remarked in Section 2. Thus the proof of the following theorem from [17, 18, 23] has been sketched. See also [53, 75].

Theorem 8.10 *Majority voting for unknown syndromes corrects $\lfloor (\delta_{FR} - 1)/2 \rfloor$ errors with complexity $\mathcal{O}(n^3)$.*

Example 8.11 Consider the following error pattern from [61] for the code $C(11)$ on the Klein quartic with error positions $Q_1 = (1, \xi)$, $Q_2 = (\xi^2, 1)$ and $Q_3 = (\xi, 1)$ and error values $e_1 = 1$, $e_2 = \xi^3$ and $e_3 = \xi$. It was shown in Example 4.17 that the modified algorithm corrects this pattern. In the following matrix we denote ξ^i by i,

and 0 by $*$. The matrix of known syndromes with respect to the functions 1, y, xy, y^2, x^2y, xy^2, y^3, x^2y^2, xy^3 and y^4 is:

$$\begin{pmatrix} * & 3 & 0 & 6 & * & 5 & 1 & 4 & * & - \\ 3 & 6 & 5 & 1 & 4 & * & - & & & \\ 0 & 5 & 4 & * & - & & & & & \\ 6 & 1 & * & + & & & & & & \\ * & 4 & - & & & & & & & \\ 5 & * & & & & & & & & \\ 1 & - & & & & & & & & \\ 4 & & & & & & & & & \\ * & & & & & & & & & \\ - & & & & & & & & & \end{pmatrix}.$$

The known discrepancies are located at the entries $(1,2)$, $(2,1)$ and $(3,3)$. Thus there are no candidates in the first, second and third row, and there are no candidates in the first, second and third column. Notice that $\rho_9 = 11$, so $C(11) = C_9$ and the set N_9 consists of the pairs $(1,10)$, $(2,7)$, $(3,5)$, $(4,4)$, $(5,3)$, $(7,2)$ and $(10,1)$. There is exactly one candidate at the entry with index $(4,4)$ indicated by a plus sign $+$, the non-candidates are indicated by a minus sign $-$. For the first three rows we have that:

$$\xi s_{i1} + \xi^3 s_{i2} + s_{i4} = 0 \text{ for all } 1 \leq i \leq 3,$$

This corresponds to the error-locator function $\xi + \xi^3 y + y^2$. This gives the correct value ξ^5 for $s_{4,4}$, so $s_{1,10}$, $s_{2,7}$, $s_{4,4}$, $s_{7,2}$ and $s_{10,1}$ all have the value ξ^5, and $s_{3,5} = s_{5,3} = \xi^4$. The number of known discrepancies is 3, this is equal to the number of errors, so there are no other discrepancies. Thus in the next steps of the procedure all entries, except in the first, second and third row and column are candidates, and the voting is unanimous. In this way we get all unknown syndromes.

Example 8.12 Consider the following error pattern for the code $C(11)$ on the Klein quartic which error positions $Q_1 = R_1$, $Q_2 = R_3$ and $Q_3 = (1, \xi)$ and error values $e_1 = \xi^3$, $e_2 = \xi$ and $e_3 = 1$. Example 4.17 showed that the modified algorithm corrects this pattern since $e_1 \neq e_2$. The matrix of known syndromes with respect to

the functions 1, y, xy, y^2, x^2y, xy^2, y^3, x^2y^2, xy^3 and y^4 is:

$$\begin{pmatrix} * & 1 & 1 & 2 & 0 & 2 & 3 & 2 & 3 & - \\ 1 & 2 & 2 & 3 & 2 & 3 & - & & & \\ 1 & 2 & 2 & 3 & + & & & & & \\ 2 & 3 & 3 & + & & & & & & \\ 0 & 2 & + & & & & & & & \\ 2 & 3 & & & & & & & & \\ 3 & - & & & & & & & & \\ 2 & & & & & & & & & \\ 3 & & & & & & & & & \\ - & & & & & & & & & \end{pmatrix}.$$

Notice that the syndrome which corresponds to the function x^iy^j is equal to ξ^j for all $j > 0$ and $2i \neq j$, since those functions are zero at Q_1 and Q_2. The known discrepancies are located at the entries $(2, 1)$ and $(1, 2)$. Thus there are no candidates in the first and second row, and there are no candidates in the first and second column. There are exactly three candidates at the entries $(3, 5)$, $(4, 4)$ and $(5, 3)$, giving the predictions ξ^2, ξ^4 and ξ^2, respectively, for $s'_{i,j}$, and three times the same vote $s_{12}(i, j) = \xi^4$ for the syndrome s_{12} corresponding to y^4. The next step gives again unanimous voting which results in the following matrix:

$$\begin{pmatrix} * & 1 & 1 & 2 & 0 & 2 & 3 & 2 & 3 & 4 & 3 & - \\ 1 & 2 & 2 & 3 & 2 & 3 & 4 & 3 & - & & & \\ 1 & 2 & 2 & 3 & 2 & 3 & + & & & & & \\ 2 & 3 & 3 & 4 & 3 & + & & & & & & \\ 0 & 2 & 2 & 3 & + & & & & & & & \\ 2 & 3 & 3 & + & & & & & & & & \\ 3 & 4 & + & & & & & & & & & \\ 2 & 3 & & & & & & & & & & \\ 3 & - & & & & & & & & & & \\ 4 & & & & & & & & & & & \\ 3 & & & & & & & & & & & \\ - & & & & & & & & & & & \end{pmatrix}.$$

As before the known discrepancies are located at the entries $(2, 1)$ and $(1, 2)$. Thus there are exactly five candidate at the entries $(3, 7)$, $(4, 6)$, $(5, 5)$, $(6, 4)$ and $(7, 3)$, giving the predictions ξ^4, ξ^4, 1, ξ^4 and ξ^4, respectively, for $s'_{i,j}$. Which, in turn, give $s_{14}(i, j)$, the respective votes ξ^4, ξ^4, 0, ξ^4 and ξ^4 for the syndrome s_{14} corresponding to xy^4. The majority voting gives $s_{14} = \xi^4$.

Example 8.13 Consider the same code $C(11)$ and the same error positions but with error values $e_1 = e_2 = e_3 = 1$. Example 4.17 showed that the modified algorithm fails

for this pattern. The matrix of known syndromes with respect to the functions 1, y, xy, y^2, x^2y, xy^2, y^3, x^2y^2, xy^3 and y^4 is:

$$\begin{pmatrix} 0 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 3 & - \\ 1 & 2 & 2 & 3 & 2 & 3 & - & & & \\ 1 & 2 & 2 & 3 & - & & & & & \\ 2 & 3 & 3 & + & & & & & & \\ 3 & 2 & - & & & & & & & \\ 2 & 3 & & & & & & & & \\ 3 & - & & & & & & & & \\ 2 & & & & & & & & & \\ 3 & & & & & & & & & \\ - & & & & & & & & & \end{pmatrix}.$$

The known discrepancies are located at the entries $(1,1)$, $(5,2)$ and $(2,5)$. Thus there are no candidates in the first, second and fifth row, and there are no candidates in the first, second and fifth column. There is exactly one candidate at the entry with index $(4,4)$. The number of known discrepancies is 3. This is equal to the number of errors, so there are no other discrepancies. Thus in the next steps of the procedure all entries, except in the first, second and fifth row and column are candidates, and the voting is unanimous.

Example 8.14 Consider the error pattern for the code $C(23)$ on the Klein quartic which was discussed in Example 4.18 and Example 7.9, where it was shown that the modified algorithm fails and Ehrhard's algorithms manages to correct this error. The matrix of known syndromes with respect to the functions 1, y, xy, y^2, x^2y, xy^2, y^3,

x^2y^2, xy^3, y^4, x^2y^3, xy^4, y^5, x^2y^4, xy^5, y^6, x^2y^5, xy^6, y^7, x^2y^6, xy^7 and y^8 is:

$$\begin{pmatrix}
* & * & * & * & * & * & * & * & * & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 6 & * & 4 & 1 & - \\
* & * & * & * & * & * & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 6 & * & 4 & 1 & - & & & \\
* & * & * & * & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 6 & 0 & 4 & 1 & - & & & & & \\
* & * & * & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 6 & * & 4 & 1 & - & & & & & \\
* & * & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 0 & 0 & 4 & 5 & - & & & & & & \\
* & * & 2 & 0 & 3 & 6 & 1 & 1 & 3 & 6 & 0 & 4 & 1 & - & & & & & & & \\
* & * & 0 & 3 & 6 & 1 & 1 & 3 & 6 & * & 4 & 1 & - & & & & & & & & \\
* & 2 & 3 & 6 & 1 & 1 & 3 & 0 & 0 & 4 & 5 & - & & & & & & & & & \\
* & 0 & 6 & 1 & 1 & 3 & 6 & 0 & 4 & 1 & - & & & & & & & & & & \\
* & 3 & 1 & 1 & 3 & 6 & * & 4 & 1 & + & & & & & & & & & & & \\
2 & 6 & 1 & 3 & 0 & 0 & 4 & 5 & - & & & & & & & & & & & & \\
0 & 1 & 3 & 6 & 0 & 4 & 1 & - & & & & & & & & & & & & & \\
3 & 1 & 6 & * & 4 & 1 & - & & & & & & & & & & & & & & \\
6 & 3 & 0 & 4 & 5 & - & & & & & & & & & & & & & & & \\
1 & 6 & 4 & 1 & - & & & & & & & & & & & & & & & & \\
1 & * & 1 & - & & & & & & & & & & & & & & & & & \\
6 & 1 & - & & & & & & & & & & & & & & & & & & \\
* & - & & & & & & & & & & & & & & & & & & & \\
4 & & & & & & & & & & & & & & & & & & & & \\
1 & & & & & & & & & & & & & & & & & & & & \\
- & & & & & & & & & & & & & & & & & & & &
\end{pmatrix}.$$

All entries in the upper left corner of the matrix are zero, denoted by $*$. This is in agreement with the fact that the error vector $\mathbf{e}$ is an element of $C_\Omega(Q, 12P)$, so $s_{i,j} = 0$ for all i, j such that $\rho_i + \rho_j \leq 12$. The 9th column depends on the previous ones, since

$$f = \xi xy + \xi^3 y^2 + \xi x^2 y + \xi^2 xy^2 + y^3 + \xi^5 x^2 y^2 + xy^3$$

is an error locator function, see Example 4.18. So $K(\mathbf{y}, 11P) = \langle 1, f \rangle$. The 10th column depends on the previous columns and one relation corresponds with the function

$$g = y + \xi^6 xy + \xi^4 x^2 y + \xi^3 xy^2 + \xi^3 y^3 + \xi x^2 y^2 + y^4,$$

which is not an error-locator function. Now it follows that

$$L(12P - Q) = \langle f \rangle \quad \text{and} \quad K(\mathbf{y}, 12P) = \langle 1, f, g \rangle.$$

Beware that at this point of the algorithm it cannot be decided whether the functions f and g are error-locators or not, unless we know already the error positions.

There are 9 known discrepancies and they are located at the entries $(1, 11)$, $(2, 8)$, $(3, 6)$, $(4, 5)$, $(5, 4)$, $(6, 3)$, $(7, 7)$, $(8, 2)$, and $(11, 1)$. Thus there is exactly one candidate at the entry $(10, 10)$, which gives as outcome that $s_{10} = 0$. The number of known discrepancies is 9 and equal to the number of errors, so there are no other discrepancies. Thus in the next steps the voting is unanimous.

Remark 8.15 In order to give an example where the modified algorithms fails and the majority voting succeeds and is not unanimous we need to look at codes on a curve of higher genus such as the Hermitian curve [23, 18, 57].

Remark 8.16 The Fundamental Iterative Algorithm [27] and the Modified Fundamental Iterative Algorithm [23] are generalizations of Gaussian elimination for a partial matrix to get the unknown syndromes.

Remark 8.17 It is not neccessary to compute all unknown syndromes, one could stop as soon as one has the unknown syndromes $s_{r+1}, \ldots, s_{r+\gamma}$ and apply the basic algorithm to the code $C_{r+\gamma}$. A more efficient stop criterion is treated in [105].

Remark 8.18 We have seen that the decoding by majority voting gave a new bound for geometric Goppa codes and this is the basis of an elementary treatment of these codes in [24, 25, 29].

Remark 8.19 The majority voting scheme is also incorporated in Porter's algorithm [97]. See also the treatment of an example with an Hermitian code.

Remark 8.20 Suppose that C' is a subcode of C and that we have a decoding algorithm for C'. A *coset decoding* algorithm has as input a received word $\mathbf{y}$ with error vector $\mathbf{e}$ with respect to C and gives as output a word $\mathbf{y}'$ which has the same coset as $\mathbf{e}$ with respect to C'. Coset decoding is applied to Hermitian codes in [112]. A different but equivalent point of view with respect to majority voting for syndromes is taken in [17, 18], where the computation of $\mathbf{y}'$ is done by a majority vote. This is called *majority coset decoding*.

Problem 8.21
What is the relation between Ehrhard's decoding algorithm and majority voting ?

Problem 8.22
Does majority voting correct more than $\lfloor (\delta_{FR} - 1)/2 \rfloor$ errors ?

9 Linear recurring relations in several variables

The Berlekamp-Massey algorithm [5, 67] on linear recurring relations in one variable was generalized to two and several variables, respectively, by Sakata [85, 86], and will be called the *algorithm of BMS*. This algorithm was subsequently used to get fast implementations of the modified algorithm in [49] and [45, 46], of Porter's algorithm in [96] and the majority voting scheme in [49, 65, 89, 90, 91]. In this section an outline of the algorithm is given.

The algorithm takes as input an N-dimensional array of elements of a field $\mathbb{F}$ and produces as output a so-called minimal set of polynomials corresponding to linear recurring relations satisfied by the array. In order to describe the algorithm some notation from [86] is required.

Let Σ_0 be defined as the set of all N-tuples of non-negative integers, that is $\Sigma_0 = \mathbb{N}_0^N$. For any subset $\Gamma \subseteq \Sigma_0$, an array over the field $\mathbb{F}$ is a mapping $u : \Gamma \to \mathbb{F}$, which is written $u = (u_\alpha)$ where $u_\alpha = u(\alpha), \alpha \in \Gamma$, is the *value* of the array of the point α, and Γ is called the *domain* of u.

A well-ordering of the elements in Σ_0 is required.

Definition 9.1 A well-ordering $<_T$ of the elements in Σ_0 is called *admissible* if the following holds:
1) For any $\alpha \in \Sigma_0$: $\quad (0, \ldots, 0) \leq_T \alpha$.
2) For any $\alpha, \beta, \gamma \in \Sigma_0$: if $\alpha <_T \beta$, then $\alpha + \gamma <_T \beta + \gamma$.

Remark 9.2 A total order satisfying (1) and (2) is sometimes called a *monomial* or *reduction* order in the Gröbner-basis literature [10, 14].

Example 9.3 The *lexicographic order* $<_L$ is defined by $(i_1, \ldots, i_N) <_L (j_1, \ldots, j_N)$ if and only if $i_1 = j_1, \ldots i_{l-1} = j_{l-1}$ and $i_l < j_l$ for some $1 \leq l \leq N$. The *total degree lexicographic order* $<_D$ is defined by $(i_1, \ldots, i_N) <_D (j_1, \ldots, j_N)$ if and only if $\sum i_l < \sum j_l$ or ($\sum i_l = \sum j_l$ and $(i_1, \ldots, i_N) <_L (j_1, \ldots, j_N)$). The lexicographic order and the total degree lexicographic order are the most common admissible orders.

It is convenient to represent *linear recurring relations* by means of N-variate polynomials $f \in \mathbb{F}[Z] = \mathbb{F}[Z_1, \ldots, Z_N]$. Any such polynomial can be written as

$$f = \sum f_\alpha Z^\alpha,$$

where $Z^\alpha = Z_1^{\alpha_1} \cdots Z_N^{\alpha_N}$, the sum is taken over all α in Σ_0, and $f_\alpha = 0$ for all but finitely many α. The finite subset Γ_f of Σ_0 of all α such that $f_\alpha \neq 0$ is called the

support of f. The maximum element in Γ_f with respect to the total order $<_T$ is called the $<_T$-*degree* of f and is written as $\mathrm{Deg}(f)$, not to be confused with $\deg(f)$ which denotes the ordinary degree of the polynomial f. A polynomial f is said to be *valid at a point* β for an array u, if $\beta \geq \mathrm{Deg}(f)$ and

$$f[u]_\beta := \sum f_\alpha u_{\alpha+\beta-\sigma} = 0,$$

where $\sigma = \mathrm{Deg}(f)$ and the sum is taken over all α such that $\alpha + \beta - \sigma \in \Gamma$. Here $\geq$ is the natural partial order on Σ_0 defined by $\alpha \geq \beta$ if and only if $\alpha_i \geq \beta_i$ for all $i = 1, \ldots, N$.

A polynomial f is said to be *valid* for an array u if f is valid at point β for u for all $\beta \geq \mathrm{Deg}(f)$, and the set of valid polynomials for an array u is denoted $VALPOL(u)$.

Definition 9.4 For an array u over $\mathbb{F}$, a *minimal polynomial set* is a finite subset $\mathcal{F}$ of $\mathbb{F}[Z]$ such that:
1) $\mathcal{F} \subseteq VALPOL(u)$.
2) Let $\mathcal{S} = \{\mathrm{Deg}(f) \mid f \in \mathcal{F}\}$, then for any σ and τ, if $\sigma \in \mathcal{S}$ and $\tau > \sigma$ then $\tau \notin \mathcal{S}$.
3) If $g \in VALPOL(u)$, then there exists a $\sigma \in \mathcal{S}$ such that $\sigma \leq \mathrm{Deg}(g)$.

Remark 9.5 Let $\Delta = \Delta(\mathcal{F})$ be the complement of $\{\tau \in \Sigma_0 \mid \sigma \leq \tau \text{ for some } \sigma \in \mathcal{S}\}$ in Σ_0. The third condition can now be rephrased by saying that there exists no polynomial $g \in VALPOL(u)$ such that $\mathrm{Deg}(g) \in \Delta$. This set Δ is called the *delta set* or the *footprint* [7] of $\mathcal{F}$. It is seen from this that the word "minimal" in the term "minimal polynomial set" refers to the Degree's of the polynomials in the set $\mathcal{F}$.

The *algorithm of BMS (Berlekamp-Massey-Sakata)* takes as input the elements of an array u and produces as output a minimal polynomial set for the array. The algorithm considers the elements of the array step by step. At each step one has a minimal polynomial set $\mathcal{F}$ for the part of the array seen so far. When the next element of the array is taken into consideration, the algorithm starts to check if the polynomials $f \in \mathcal{F}$ are still valid for the new array. If this is not the case, they are updated and a new polynomial set and a new Δ-set is produced.

Any element $\alpha \in \Sigma_0$ has an immediate *successor* with respect to the well order $<_T$, which we denote by $\alpha + 1$, and when it has an immediate *predecessor* we denote it by $\alpha - 1$. If u is an array with domain Γ, then u^β is the *restriction* of u to the domain $\{\alpha \in \Gamma \mid \alpha <_T \beta\}$, that is to say $u^\beta(\alpha) = u(\alpha)$ for all $\alpha \in \Gamma$ such that $\alpha <_T \beta$. The updating is based on Lemma 6 from [86].

Lemma 9.6 *Let $f(Z), g(Z) \in \mathbb{F}[Z]$, $Deg(f) = \sigma$ and $Deg(g) = \tau$. Suppose that f is valid for the array $u^{\beta-1}$ but*

$$\sum f_\alpha u_{\alpha+\beta-\sigma} = d_f \neq 0,$$

and that g is valid for $u^{\gamma-1}$ but

$$\sum g_\alpha u_{\alpha+\gamma-\tau} = d_g \neq 0.$$

If $\beta >_T \gamma$, then

$$h(f,g) = Z^{\rho-\sigma} f - \frac{d_f}{d_g} Z^{\rho-\beta+\gamma-\tau} g$$

is valid at β and $Deg(h(f,g)) = \rho$, where $\rho = max\{\sigma, \beta - \gamma + \tau\}$.

This construction of h is called the *Berlekamp Procedure*. The fundamental lemma that corresponds to *Massey's Theorem* in the one-dimensional case is Lemma 2 from [86].

Lemma 9.7 *If $f \in VALPOL(u^\beta)$ and $f \notin VALPOL(u^{\beta+1})$, then there exists no polynomial $g \in VALPOL(u^{\beta+1})$ such that $Deg(g) \leq \beta - Deg(f)$.*

The algorithm works with two sets $\mathcal{F}$ and $\mathcal{G}$, where $\mathcal{F}$ is a minimal polynomial set for the part of the array seen so far and the set $\mathcal{G}$ contains polynomials which are to be used in the Berlekamp procedure, that is polynomials which failed to be valid, together with the point where this happened.

One should note that if $VALPOL(u)$ is an ideal, which is the case relevant for decoding, the output of the algorithm of BMS is a Gröbner basis for that ideal, see Section 11.

We will not present the algorithm in detail and refer the interested reader to [86]. See also [84]. However we do give an example.

Example 9.8 Define a *weighted degree* or *order* w on $\mathbb{N}_0^3$ by $w(i_1, i_2, i_3) = 3i_1 + 5i_2 + 7i_3$. The total order $<_T$ used in the following example is a *weighted lexicografic order* defined by $(i_1, i_2, i_3) <_T (j_1, j_2, j_3)$ if and only if $w(i_1, i_2, i_3) < w(j_1, j_2, j_3)$ or
($w(i_1, i_2, i_3) = w(j_1, j_2, j_3)$ and $(i_1, i_2, i_3) <_L (j_1, j_2, j_3)$). The reason for this ordering will be clear from the next section.

As input to the algorithm of BMS take the 3 dimensional array over $\mathbb{F}_8$ given by the following table.

u_{000}	u_{100}	u_{010}	u_{200}	u_{001}	u_{110}	u_{300}	u_{020}	u_{101}	u_{210}
0	ξ^3	1	ξ^6	0	ξ^5	ξ	ξ^4	ξ^4	0

In this table and the continuation of this example below the abbreviation ijk is used for the index (i, j, k). The algorithm gives:

Point	$\mathcal{F}$	$\mathcal{G}$
000	1	$\emptyset$
100	Z_1^2	1
010	$Z_1^2,\ \xi^4 Z_1 + Z_2$	1
200	$\xi^3 Z_1 + Z_1^2,\ \xi^4 Z_1 + Z_2$	$1,\ Z_1^2$
001	$\xi^3 Z_1 + Z_1^2,\ \xi^4 Z_1 + Z_2,\ Z_3$	$1,\ Z_1^2$
110	$\xi^3 Z_1 + Z_1^2,\ \xi^6 + \xi^4 Z_1 + Z_2,\ Z_3$	$1,\ Z_1^2,\ \xi^4 Z_1 + Z_2$
300	$\xi + \xi^3 Z_1 + Z_1^2,\ \xi^6 + \xi^4 Z_1 + Z_2,\ Z_3$	$1,\ \xi^3 Z_1 + Z_1^2,\ \xi^4 Z_1 + Z_2$
020	$\xi + \xi^3 Z_1 + Z_1^2,\ \xi^6 + \xi^4 Z_1 + Z_2,\ Z_3$	$1,\ \xi^3 Z_1 + Z_1^2,\ \xi^4 Z_1 + Z_2$
101	$\xi + \xi^3 Z_1 + Z_1^2,\ \xi^6 + \xi^4 Z_1 + Z_2,\ \xi + Z_3$	$1,\ \xi^3 Z_1 + Z_1^2,\ \xi^4 Z_1 + Z_2$
210	$\xi + \xi^3 Z_1 + Z_1^2,\ Z_1(\xi^6 + \xi^4 Z_1 + Z_2),\ \xi + Z_3$	$Z_1^2,\ \xi^4 Z_1 + Z_2,\ \xi^6 + \xi^4 Z_1 + Z_2$

Thus $\#\Delta(\mathcal{F}) = 3$.

Remark 9.9 A general implementation of the algorithm of BMS is given in [1].

Remark 9.10 The problem solved by the algorithm of BMS could of course be treated by solving a system of linear equations. The point here is that this algorithm, in many cases, has lower complexity.

10 Faster decoding

The basic and the modified algorithm as well as the majority scheme have the complexity of solving systems of linear equations, both for finding the error locations and the error values. If one uses the special structure of the syndrome matrix, the complexity of the decoding can be improved. This is done in [29], using the block-Hankel structure. The algorithm of BMS is used to get fast implementations of, the modified algorithm in [49] and [45, 46], Porter's algorithm in [96], and the majority voting scheme in [49, 65, 89, 90, 91]. In this section it is shown how the latter is used for decoding one-point codes up to half of the Feng-Rao distance. The section is based on [90].

Consider codes of the form $C = C_L^{\perp}(D, mP) = C_\Omega(D, mP)$ where P is not in the support of $D = P_1 + \cdots + P_n$. Let $a_1, a_2, \ldots, a_N$ be a minimal set of generators for the semigroup of non-gaps at P in increasing order, and let g_i be a function with pole order a_i at P and with no other poles. Then to any vector $\alpha = (\alpha_1, \ldots, \alpha_N)$ of integers there corresponds the function

$$g^\alpha := \prod_{j=1}^{N} g_j^{\alpha_j}$$

having only poles at P of order

$$w(g^\alpha) = w(\alpha) := \sum_{j=1}^{N} a_j \alpha_j.$$

The functions g^α, with $w(\alpha) \leq m$, span the space $L(mP)$, but are not necessarily independent. In Section 6 the following property was mentioned.

Lemma 10.1 *If $w(\alpha) = w(\alpha')$, then there exists $c \in \mathbb{F}_q^*$ and a rational function g such that*

$$g^\alpha = c g^{\alpha'} + g, \quad \textit{and} \quad w(g) < w(\alpha).$$

For fixed function g^α we associate with each word $\mathbf{y} \in \mathbb{F}_q^n$ a syndrome $s_\alpha(\mathbf{y})$ by

$$s_\alpha(\mathbf{y}) = \sum_{j=1}^{N} g^\alpha(P_j) y_j.$$

Then it follows that

$$\mathbf{c} \in C \text{ if and only if } s_\alpha(\mathbf{c}) = 0 \text{ for all } \alpha \text{ with } w(\alpha) \leq m.$$

In the decoding situation $\mathbf{y} = \mathbf{c} + \mathbf{e}$ is received and here $s_\alpha(\mathbf{c}+\mathbf{e}) = s_\alpha(\mathbf{e})$ if $w(\alpha) \leq m$. These can be easily calculated. It was mentioned in Section 2 that the error vector is determined when all syndromes are known. The following variation of the *Discrete Fourier Transform* method [7] gives an explicit formula.

Proposition 10.2 *Assume that that all coordinates of the points P_l are non-zero. If all syndromes $s_\alpha(\mathbf{e}), 0 \leq \alpha_i \leq q-2$, $i = 1, \ldots, N$ are known, then*

$$e_l = (-1)^l \sum_{\alpha} s_\alpha(\mathbf{e}) g^{-\alpha}(P_l),$$

where the summation extends over all vectors α with $0 \leq \alpha_j \leq q-2$.

Remark 10.3 In light of this it suffices to know all syndromes in order to calculate the error-magnitudes (and hence positions as well). But the algorithm of BMS provides efficient means for finding them all. Alternatively, it is possible to use the algorithm of BMS (with majority voting) to get a Gröbner basis for the ideal of error-locators. From this it is then possible to find the error positions and then the error-magnitudes. This approach was used in [60, 83, 84], see Section 11.

Choose a well-ordering on Σ_0 corresponding to the code by defining $\alpha <_T \beta$ if and only if $w(\alpha) < w(\beta)$ or $w(\alpha) = w(\beta)$ and there exists an l such that $\alpha_l < \beta_l$ and $\alpha_i = \beta_i$ for all $i < l$. Moreover choose a subset Σ' of Σ_0, such that Σ' contains exactly one element corresponding to each pole order.

In the algorithm of BMS only consider polynomials for which the $<_T$-degree belongs to Σ', which is possible by 10.1. As a consequence use Σ' instead of Σ_0 in the definition of $\Delta = \Delta(\mathcal{F})$, which means that mutually distinct points in Δ correspond to functions with mutually distinct pole orders. With this modification, and the array of known syndromes as input to the algorithm one can prove the following lemma from [90].

Lemma 10.4 *Suppose that the number of errors that occurred is equal to t and at most $\lfloor(\delta_{FR} - 1)/2\rfloor$. Then, at each step in the algorithm the number of points in the Δ-set is at most t.*

Let $s_\alpha = s_\alpha(\mathbf{e})$. Suppose now all syndromes s_α are known, where $w(\alpha) < m'$, and all s_α, where $\alpha \in \Sigma$ and $w(\alpha) = m'$, are to be determined.

Let $\gamma \in \Sigma'$ satisfy $w(\gamma) = m'$. Put $\gamma_0 = \gamma$ and let $\gamma_1, \gamma_2, \ldots$ be all other elements of Σ_0 with order m'.

Let $\mathcal{F} = \{f_1, f_2, \ldots, f_k\}$ denote a minimal polynomial set for the array s with values s_α in the domain $\Gamma = \{\alpha \mid w(\alpha) < m'\}$, where $\mathrm{Deg}(f_i) \in \Sigma'$. Suppose without loss of generality that all f_i have leading coefficient 1.

Let $\mathrm{Deg}(f_i) = \sigma_i$ and suppose $\sigma_i \leq \gamma_j$, then it is possible to calculate

$$-\sum_{\alpha \in \Gamma_i} f_{i,\alpha} s_{\alpha+\gamma_j-\sigma_i},$$

where Γ_i is the support of f_i with σ_i deleted, and denote it by s'_{γ_j}. If f_i is valid at γ_j, then

$$s_{\gamma_j} - s'_{\gamma_j} = f[s]_{\gamma_j} = 0,$$

so the value of the sum s'_{γ_j} is equal to s_{γ_j}. Thus it is possible to use s'_{γ_j} combined with the syndrome equation corresponding to Lemma 10.1 to give predictions of s_γ.

To state the results precisely put

$$K(\gamma) = \{\alpha \in \Sigma' \mid \alpha \leq \gamma_j \text{ and } \gamma_j - \alpha \in \Sigma' \text{ for some } j\}.$$

For each σ_i with $\mathrm{Deg}(f_i) = \sigma_i$ check if there is a γ_j with $\gamma_j \geq \sigma_i$ and $\gamma_j - \sigma_i \in \Sigma'$. If this is the case use s'_{γ_j} to get a prediction of s_γ, if not the f_i is not used. Put

$$K_i = \{\alpha \in K(\gamma) \mid \alpha \leq \gamma_j - \sigma_i\}$$

and let $w_1, \ldots, w_l$ be the different predictions v_i for s_γ obtained by the above method.

For each $p = 1, \ldots, l$, let

$$L_p = \bigcup_{v_i = w_p} K_i \setminus \Delta.$$

Then the following theorem holds. See [90].

Theorem 10.5 *Suppose that the number t of errors satifies $t \leq (\delta_{FR} - 1)/2$. Let $p \in \{1, \ldots, l\}$ be the index for which the number of elements of L_p is maximal. Then $s_\gamma = w_p$.*

Here

$$\delta_{FR} = \min\{ \#K(\rho) \mid \rho \in \Sigma' \text{ and } w(\rho) > m \}$$

which is equivalent to the definition in Section 8. See [90].

The algorithm works for the following reason. If one chooses a wrong value v_i of the unknown syndrome, then by Lemma 9.7 the Δ-set is increased by $K_i \setminus \Delta$. So it can be seen that the next Δ-set has size greater than t, violating Lemma 10.4.

Example 10.6 This is a continuation of the example from the Klein quartic as in Examples 3.7, 4.17 and 8.11. The semigroup of non-gaps at P is generated by the numbers 3, 5 and 7. The corresponding functions are y, xy and x^2y; and these will be denoted by the variables Z_1, Z_2 and Z_3, respectively. For the code $C(11)$ we consider the 3-error pattern from [61] where $Q_1 = (1, \xi)$, $Q_2 = (\xi^2, 1)$ and $Q_3 = (\xi, 1)$ and the corresponding error values are 1, ξ^3 and ξ, respectively. From this the following syndromes are obtained:

Poleorder	0	3	5	6	7	8	9	10	11
Function	1	y	xy	y^2	x^2y	xy^2	y^3	x^2y^2	xy^3
Monomial	1	Z_1	Z_2	Z_1^2	Z_3	Z_1Z_2	Z_1^3	Z_2^2 , Z_1Z_3	$Z_1^2Z_2$
Syndrome	s_{000}	s_{100}	s_{010}	s_{200}	s_{001}	s_{110}	s_{300}	s_{020}, s_{101}	s_{210}
Syndromevalue	0	ξ^3	1	ξ^6	0	ξ^5	ξ	ξ^4	0

Note that the syndromes s_{020} and s_{101} are equal, since both Z_1Z_3 and Z_2^2 represent x^2y^2. It was mentioned in Example 4.17 that the modified algorithm already corrects this error pattern, but it is illustrative to see how the implementation of the algorithm of BMS in the majority voting scheme gives all the unknown syndromes and the error locator ideal. If we want to decode this using the method described above we first use the algorithm of BMS on the known syndromes as we did in Example 9.8. This points out the reason for the choice of the total order. The output of the algorithm of BMS on the known syndromes was:

$$\xi + \xi^3 Z_1 + Z_1^2, \;\; Z_1(\xi^6 + \xi^4 Z_1 + Z_2), \;\; \xi + Z_3$$

At the point 400 corresponding to pole-order 12, the polynomial $\xi + \xi^3 Z_1 + Z_1^2$ predicts the value ξ^5 for s_{400}. At the point 011, also corresponding to pole-order 12, the polynomial $\xi + Z_3$ predicts the value ξ for s_{011}. From the equation of the curve it follows that $Z_2Z_3 = Z_2 + Z_1^4$ and so $s_{011} = s_{010} + s_{400}$. Hence the prediction at s_{400} also gives $s_{011} = 1 + \xi^5 = \xi^4$. By analyzing K_i it can be seen that the correct value for s_{011} is ξ^4, and hence $s_{400} = \xi^5$. The following update of the algorithm of BMS gives the set $\mathcal{F}$ with the elements:

$$\begin{gathered} \xi + \xi^3 Z_1 + Z_1^2 \\ \xi^6 Z_1 + \xi^4 Z_1^2 + Z_1 Z_2 \\ (\xi + Z_3) + \xi^4(\xi^6 + \xi^4 Z_1 + Z_2) = 1 + \xi Z_1 + \xi^4 Z_2 + Z_3 \end{gathered}$$

Corresponding to pole-order 13 are the points 201 and 120 and from the curve it follows that $Z_1Z_2^2 = Z_1^2Z_3$, so $s_{120} = s_{201}$. At 201 the polynomial $\xi + \xi^3 Z_1 + Z_1^2$ predicts the value 1 for s_{201}. At 120 the polynomial $\xi^6 Z_1 + \xi^4 Z_1^2 + Z_1 Z_2$ predicts the value ξ^4 for $s_{201} = s_{120}$. Analyzing the the two sets K_i shows that $s_{201} = s_{120} = 1$, and the following update of the algorithm of BMS gives the set $\mathcal{F}$ with the elements:

$$\begin{gathered} \xi + \xi^3 Z_1 + Z_1^2 \\ (\xi^6 Z_1 + \xi^4 Z_1^2 + Z_1 Z_2) + (\xi^6 + \xi^4 Z_1 + Z_2) \\ 1 + \xi Z_1 + \xi^4 Z_2 + Z_3 \end{gathered}$$

Thus

$$\mathcal{F} = \left\{ \xi + \xi^3 Z_1 + Z_1^2, \;\; \xi^6 + \xi^3 Z_1 + Z_2 + \xi^4 Z_1^2 + Z_1 Z_2, \;\; 1 + \xi Z_1 + \xi^4 Z_2 + Z_3 \right\}.$$

From here on it turns out that the possible predictions all give the same values. Indeed this $\mathcal{F}$-set is now a basis for the ideal of error locators and the common zeros are the error positions. It is possible to continue and get all the syndromes and then use Proposition 10.2 to get the correct error values.

Example 10.7 The codes from Example 3.8 can be decoded using the 4-dimensional version of the algorithm of BMS, and the decoding is then done up to half the Feng-Rao bound which is sometimes greater than the Goppa bound in the range $r \leq 40$, see Example 8.4.

Remark 10.8 One should be a little careful when talking about complexity, and in particular the use of the "big $\mathcal{O}$" notation. Regular affine plane curves have at most q^2 rational points over $\mathbb{F}_q$, so the length of the code is bounded above by this. Thus one could say that the algorithm BMS has complexity $\mathcal{O}(n^{7/3})$ for plane curves, but then q is not fixed and one is comparing codes over distinct alphabets. In this way one is counting a multiplication in for instance $\mathbb{F}_2$ and in $\mathbb{F}_{256}$ both as one unit. If one fixes q, then in an asymptotic sense the algorithm of BMS is not faster than solving systems of linear equations. What makes sense here is to compare different decoding algorithms for the same class of codes and counting the actual number of additions and multiplications.

The complexity, in terms of the number of additions and multiplications in $\mathbb{F}_q$, for the whole decoding procedure described above has the following upper bound

$$Aa_1n^2 + Bq^{N+1}(a_1 + \cdots + a_N) + CnNq^N,$$

where A, B and C are (small) constants. How good or bad this is of course depends on the actual situation. For the Hermitian curve we have $n = r^3$ and $q = r^2$, and $N = 2$, $a_1 = r$ and $a_2 = r + 1$ if P is the point at infinity. If we express the above upper bound for the complexity in terms of n we get $A_2n^{7/3}$. If we express the complexity in n of codes on curves in affine 3-space we get $A_3n^{5/2}$. If one continues in this way one gets $A_rn^{3-\frac{2}{r+1}}$ as the total number of additions and multiplications for the algorithm for curves in affine r-space. See [45, 46].

The algorithm of BMS has also been used in [96] to obtain the same decoding complexity for codes on Hermitian curves using Porter's algorithm.

Remark 10.9 The procedure described above was implemented in [65] for codes from Hermitian curves over $\mathbb{F}_{256}$, and the general algorithm of BMS was recently implemented in [1]. The decoding algorithm for $N > 2$ has yet to be implemented.

Remark 10.10 The general problem of solving linear equations can be done faster than Gaussian elimination. Its complexity can be reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^{2.38})$, where n is the number of variables. See [92, 103] for survey papers on this topic.

Problem 10.11 Is there a decoding algorithm which decodes all algebraic-geometric codes up to half the designed minimum distance with complexity $\mathcal{O}(n^2)$ for $n \to \infty$?

Problem 10.12 Apart from the complexity one should also investigate the actual performance of a decoding algorithm, that is to say the error probability of decoding. See [51]. As a start one needs to know the weight enumerator of the code. See [50, 107]. For both problems not much is known.

11 Error-locator ideals and Gröbner bases

The solution of the key equation for cyclic and RS codes can be done by Euclid's algorithm [104] or by the Berlekamp-Massey algorithm [5, 67] for polynomials in one variable. The first algorithm has as output the greatest common divisor of two polynomials f and g, that is to say, a generator of the ideal generated by f and g. The polynomials generated at intermediate steps are element of this ideal, and their degrees are decreasing. In the second, the intermediate polynomials generally have increasing degrees and are seldom elements of this ideal.

Definition 11.1 Let $<_T$ be an admissible order on $\mathbb{N}_0^N$, see Section 9. The set of monomials in the variables $Z_1, \ldots, Z_N$ is denoted by $\mathcal{M}(N)$. We say $Z^\alpha <_T Z^\beta$ if and only if $\alpha <_T \beta$. If $f = \sum_{\alpha \leq \beta} \lambda_\alpha Z^\alpha$ and $\lambda_\beta \neq 0$, then $\{Z^\alpha \mid \lambda_\alpha \neq 0\}$ is called the *support* of f and is denoted by $supp(f)$, the *leading monomial* of f is Z^β and is denoted by $lm(f)$ and $\lambda_\beta Z^\beta$ is called the *leading term* of f and is denoted by $lt(f)$. Let $\mathcal{B}$ a finite set of polynomials. We say that f *reduces* to g with respect to $\mathcal{B}$ if there exists a monomial Z^α in the support of f and an element $b \in \mathcal{B}$ such that $lm(b) \leq Z^\alpha$ and

$$g = f - \lambda_\alpha \frac{Z^\alpha}{lt(b)} b.$$

This is denoted by $f \rightarrow_{\mathcal{B}} g$, or $f \rightarrow g$ for short. If $f = g$ or there is a sequence $f_1, \ldots, f_k$ such that $f = f_1$, $g = f_k$ and $f_i \rightarrow_{\mathcal{B}} f_{i+1}$ for all $1 \leq i < k$, then we denote this by $f \rightarrow_{\mathcal{B}}^\bullet g$. The ideal generated by $\mathcal{B}$ is denoted by $(\mathcal{B})$. If $f \rightarrow_{\mathcal{B}}^\bullet 0$, then $f \in (\mathcal{B})$. A finite set $\mathcal{B}$ in $\mathbb{F}[Z_1, \ldots, Z_m]$ is called a *Gröbner basis* if the converse holds as well; that is to say, if $f \rightarrow_{\mathcal{B}}^\bullet 0$ for all $f \in (\mathcal{B})$.

The following theorem characterizes Gröbner bases. See [10, 14].

Theorem 11.2 *Let $\mathcal{B}$ be a finite set in $\mathbb{F}[Z_1, \ldots, Z_N]$. Then $\mathcal{B}$ is a Gröbner basis if and only if*

$$\{lm(f) \mid f \in (\mathcal{B}), f \neq 0\} = \{lm(b)m \mid b \in \mathcal{B}, b \neq 0,\ m \in \mathcal{M}(N)\}.$$

Remark 11.3 *Buchberger's algorithm* [10, 14] is a common generalization of Euclid's algorithm for polynomials in one variable and Gaussian elimination for polynomials in

several variables of degree one, to polynomials in several variables and arbitrary degree. The output of this algorithm is a Gröbner basis of the ideal generated by a given set of polynomials as input. From start to finish the set of polynomials generates the same ideal. The complexity of Buchberger's algorithm is not polynomial. The algorithm of BMS gives as output a minimal polynomial set, and in the intermediate steps the sets are Gröbner bases of the ideals they generate, but these ideals vary. The algorithm of BMS has polynomial complexity.

Remark 11.4 In [30] another method for solving the same problem as the algorithm of BMS is developed. It is a variation of the Gröbner basis algorithm and has therefore in general higher complexity.

Remark 11.5 In [84] the authors also use the algorithm of BMS to decode algebraic-geometric codes $C_\Omega(D, G)$, up to half the Feng-Rao bound. In their method the voting procedure is replaced by a threshold rule. The updating in the algorithm of BMS is done for each of the predictions of the unknown syndrome and the correct value is then the only one where the new Δ-set has size less than or equal to $t = \lfloor (\delta_{FR} - 1)/2 \rfloor$. The paper also relates the calculation of the ideal of error locators to general Gröbner basis techniques, since in the decoding situation the algorithm of BMS indeed gives a Gröbner basis.

In [60, 61] Gröbner basis theory is also applied to give a basis for the ideal of error locators with respect to a (weighted) total degree lexicographic order. Moreover [61] produces a second Gröbner basis with respect to a lexicographic order and this is subsequently used to give a generalized Forney formula for the error values. An analysis of the complexity of this formula and a comparison with the Discrete Fourier Transform method remains to be done.

Remark 11.6 The algorithm of BMS was already used to decode multi-cyclic codes. See [87] and also [83]. Multi-cyclic, Hyperbolic Cascaded RS, Reed-Muller and AG codes were treated in a unified way in [84] and also [25, 77].

12 Decoding linear codes up to half the minimum distance

In this section the literature on decoding up to half the minimum distance by error location and majority voting for unknown syndromes for arbitrary linear codes is surveyed.

The following definition generalizes the ingredients necessary for the basic algorithm [71, 74, 76]. The same kind of reasoning was published in [54].

Definition 12.1 Let C be an $\mathbb{F}_q$-linear code of length n. Let A and B be $\mathbb{F}_{q^m}$-linear codes of length n, then (A, B) is called a *t-error-correcting pair* for C over $\mathbb{F}_{q^m}$, if the following conditions hold:
1) $C \subseteq (A * B)^{\perp}$
2) $k(A) > t$
3) $d(B^{\perp}) > t$
4) $d(A) + d(C) > n$.

Remark 12.2 Suppose (A, B) is a t-error correcting pair for C. Let $\mathbf{y}$ be a received word with at most t errors. Define the kernel $K(\mathbf{y})$ as in the case for the basic algorithm by

$$K(\mathbf{y}) = \{\mathbf{a} \in A \mid \sum y_i a_i b_i = 0 \quad \text{for all} \quad \mathbf{b} \in B\}.$$

If $A * B \subseteq C^{\perp}$ and $\mathbf{y}$ is a word with error $\mathbf{e}$, then $K(\mathbf{y}) = K(\mathbf{e})$. Suppose J is a subset of $\{1, \ldots, n\}$. Define

$$A(J) = \{\mathbf{a} \in A \mid a_j = 0 \text{ for all } j \in J\}.$$

If I is the set of error positions, then

$$A(I) \subseteq K(\mathbf{y}).$$

Condition (3) guarantees that $K(\mathbf{y}) = A(I)$. A non-zero element $\mathbf{a}$ of $K(\mathbf{y})$ is ensured by condition (2). The set of zeros of $\mathbf{a}$ contains I and condition (4) implies that $\mathbf{a}$ has at most $d(C) - 1$ zeros. The error values can be found by Proposition 2.4. The Basic Algorithm is defined now and one can show the following theorem along the same lines. See [74].

Proposition 12.3 *If C is an $\mathbb{F}_q$-linear code of length n and (A, B) is a t-error correcting pair for C over $\mathbb{F}_{q^m}$, then the basic algorithm corrects all words with at most t errors with complexity $\mathcal{O}((mn)^3)$.*

Example 12.4 Classical Goppa codes, see Example 3.4. Let $\alpha = (\alpha_1, \ldots, \alpha_n)$ be an n-tuple of n distinct elements of $\mathbb{F}_q$. Let $L = \{\alpha_1, \ldots, \alpha_n, \}$ and g a polynomial of degree r which has no zeros in L. Let $A = RS_{t+1}(\alpha)$ and $B = GRS_t(\alpha, \mathbf{x})$, where $t = \lfloor r/2 \rfloor$ and $x_i = g(\alpha_i)^{-1}$. Then (A, B) is a t-error correcting pair for the classical Goppa code $\Gamma(L, g)$.

Remark 12.5 The condition $A * B \subseteq C^{\perp}$ is easy to fulfill for algebraic-geometric codes. Suppose we want to decode the code $C = C_{\Omega}(D, G)$. It was remarked already

that the dual of this code is $C_L(D,G)$. Let F be any divisor with support disjoint from the support of D, then $G-F$ has also support disjoint from the support of D. Moreover if $f \in L(F)$ and $g \in L(G-F)$, then $fg \in L(G)$. If $\mathbf{a} = \mathrm{ev}_D(f)$ and $\mathbf{b} = \mathrm{ev}_D(g)$, then $\mathbf{a} * \mathbf{b} = \mathrm{ev}_D(fg)$. Thus

$$C_L(D,F) * C_L(D,G-F) \subseteq C_L(D,G) = C_\Omega(D,G)^\perp.$$

With the above estimates for the dimension and minimum distance of AG codes we have the following proposition from [71, 74].

Proposition 12.6 *Let F and G be divisors with support disjoint from D. Let $A = C_L(D,F)$, $B = C_L(D,G-F)$ and $C = C_\Omega(D,G)$. Then*
1) $A * B \subseteq C^\perp$.
2) *If $t + \gamma \leq deg(F) < n$, then $k(A) > t$.*
3) *If $deg(G-F) > t + 2\gamma - 2$, then $d(B^\perp) > t$.*
4) *If $deg(G-F) > 2\gamma - 2$, then $d(A) + d(C) > n$.*

Remark 12.7 Every algebraic-geometric code over $\mathbb{F}_q$ has a $\lfloor (\delta_\Gamma - 1 - \gamma)/2 \rfloor$-error-correcting pair over $\mathbb{F}_q$ if $\deg(G) > 2\gamma - 2$. See [74].

Remark 12.8 Every algebraic-geometric code over $\mathbb{F}_q$ has a $\lfloor (\delta_\Gamma - 1)/2 \rfloor$-error-correcting pair over $\mathbb{F}_{q^m}$ if $m > log_q(2\binom{n}{t} + \binom{n}{t+1} + 1)$, by the same counting argument with the zeta function and the Jacobian as was explained in Section 5. See [76]. The existence of the pair (A,B) is not effectively given.

Remark 12.9 The number of decoding failures of received words with t errors is investigated in [74] under the assumption that there exists a pair (A,B) which satisfies conditions $(1), (2)$ and (4) for a linear code C of minimum distance $2t+1$.

Remark 12.10 It is not true that every linear code of minimum distance d has a $\lfloor (d-1)/2 \rfloor$-error-correcting pair. An $[n, n-4, 5]$ code has a 2-error-correcting pair if and only if it is an Extended Generalized RS code or, what is the same, an algebraic-geometric code on a curve of genus zero. See [76].

Remark 12.11 Error-correcting pairs have been found for many cyclic codes to decode beyond the BCH error-correcting capacity [19]. Majority voting for unkown syndromes has been generalized for arbitrary linear codes as well, with the notion of an *error-correcting array*. See [53, 75]. The most general scheme for getting bounds on the minimum distance and decoding up to half this bound is found in the notion of the *shift bound*. See [77, 98, 99]. This is implcit in [28].

13 Conclusion

The development of an algorithm has three steps:

$$E^{E^E}$$

First one proves its *existence*, but it is intractable, thereafter one tries to find an *effective* algorithm, but it is not good enough for practical implementation or not fast enough to compete with other algorithms, and at last one finds an algorithm which is *efficient*. See [13]. The decoding of RS codes is in the third step. The decoding of algebraic-geometric codes has just finished the second stage and is at the start of the third step. The basic and modified algorithm of Section 4 and the decoding by solving the key equation of Section 6 are effective, but they do not decode up to half the minimum distance. The algorithm which uses many divisors in parallel of Section 5 corrects errors up to half the designed minimum distance, but it is an existence proof and the algorithm is not effective. Both the improvement of the modified algorithm of Section 7 and the decoding by majority voting of Section 8 are effective and decode up to half the designed minimum distance. The effective algorithms are made efficient by the use of linear recurring sequences in several variables of Sections 9 and 10.

Who was first ? Amazingly often the same idea or solution of a problem is found independently by several persons, roughly at the same time but at distinct places. A classical example is the discovery of non-Euclidean geometry. In the history of the decoding of linear codes one sees this phenomenon already in the decoding of RS codes by Arimoto [3] and Peterson [78], and later by the decoding of algebraic-geometric codes by Justesen et al. [48], Krachkovskii [56] and Porter [79], and finally the algorithms of Ehrhard [22] and Feng-Rao [23]. Although we do not advocate the view of Sheldrake [93] on morphic resonance which says that: "If someone has discovered something for the first time, the same solution will be found more easily after that by everybody else", it remains a somewhat mysterious phenomenon, of which you may say: "It is in the air", or which has a very down-to-earth explanation: "If everyone in the field is looking at a solution of a certain problem, then it is not a surprise that at a certain moment several persons find similar solutions".

At the beginning of the invention of RS codes the engineering community had the impression that these codes would never be used in practice, since it used advanced mathematics such as Galois fields. It was considered quite an esoteric subject, nice for research but of no practical importance. This has dramatically changed after the development of chip technology and fast decoding algorithms, the application to the compact disc player and the error-correction in deep space information transmission. See [111]. One might hope that the attitude towards algebraic-geometric codes, which

is considered to be a difficult and remote topic nowadays, will change too for the better in the future.

At the moment there are two conflicting predictions concerning the practical use of algebraic-geometric codes. One forecast is saying that they will not be used since RS codes are good enough for the coming 50 years. Although the parameters of AG codes, for instance the Hermitian codes, are superior to those of RS codes, the decoding algorithms for the latter are much faster than the present algorithms for the former codes. We think that it is safe to wager with R. Blahut [8] on the second prediction which says that algebraic-geometric codes will be used in practice within 20 years.

Acknowledgement We hope that all those who have contributed to the decoding of algebraic-geometric codes recognize their fair share in this survey.

References

[1] J. Aaberg: "An implementation of Sakata's algorithm," Master's thesis, Techn. Univ. Lund. Sweden, 1994.

[2] S.S. Abhyankar, *Algebraic geometry for scientists and engineers*, Math. Surveys and Monographs vol. 35, Amer. Math. Soc., Providence, Rhode Island, 1990.

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

[4] A. Barg, "Some new NP-complete coding problems," *Probl. Peredachi Inform.* vol. 30, No. 3, pp. 23-28, July-Sept. 1994. Translation: *Probl. Inform. Transmission.*, vol. 30, No. 2, 1994.

[5] E.R. Berlekamp, *Algebraic coding theory*, McGraw-Hill, New York, 1968.

[6] 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, May 1978.

[7] R.E. Blahut, "Algebraic coding theory in one and two dimensions," Lectures at the Eindhoven Univ. Techn., June 1994.

[8] R. Blahut, I. Csiszár, D. Forney, P. Narayan, M. Pinsker and S. Verdu, "Shannon theory: present and future," *IEEE Inform. Theory Soc. Newsletter*, vol. 44, pp. 1-10, Dec. 1994.

[9] J. Bruck and M. Naor, "The hardness of decoding linear codes with preprocessing," *IEEE Trans. Inform. Theory* vol. IT-36, pp. 381-385, March 1990.

[10] B. Buchberger, "Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal," Ph.D. thesis, Univ. of Innsbruck, Austria, 1965.

[11] Ph. Carbonne and A. Thiong Ly, "Minimal exponent for Pellikaan's decoding algorithm," *Proceedings Eurocode 92, CISM Courses and Lectures*, vol. 339, pp. 31-42, Springer-Verlag, Wien New York, 1993.

[12] C. Chevalley, *Introduction to the theory of algebraic functions in one variable*, Math. Surveys VI, Providence, AMS 1951.

[13] A.M. Cohen, "Wat heb je nu aan algebra," Intreerede, Eindhoven Univ. Techn., Nov. 1993.

[14] D. Cox, J. Little and D. O'Shea, *Ideals, varieties, and algorithms: an introduction to computational algebraic geometry*, Springer-Verlag, Berlin etc., 1992.

[15] Y. Driencourt, "Some properties of elliptic codes over a field of cahracteristic 2," *Proceedings AAECC-3, Grenoble 1985, Lect. Notes Comp. Sc.* vol. 229, pp. 185-193, 1986.

[16] I.M. Duursma, "Algebraic decoding using special divisors," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 694-698, March 1993.

[17] I.M. Duursma, "Majority coset decoding," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 1067-1071, May 1993.

[18] I.M. Duursma, "Decoding codes from curves and cyclic codes," Ph.D. thesis, Eindhoven Univ. Techn., Sept. 1993.

[19] I.M. Duursma and R. Kötter, "On error locating pairs for cyclic codes," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 1108-1121, July 1994.

[20] D. Ehrhard, "Decoding algebraic-geometric codes by solving a key equation," Proceedings AGCT-3, Luminy 1991, Lect. Notes Math. vol. 1518, pp. 18-25, 1992.

[21] D. Ehrhard, "Über das Dekodieren Algebraisch-Geometrischer Codes," Ph.D. thesis, Universität Düsseldorf, July 1991.

[22] D. Ehrhard, "Achieving the designed error capacity in decoding algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-39, pp.743-751, May 1993.

[23] 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.

[24] G.-L. Feng and T.R.N. Rao, "A simple approach for construction of algebraic-geometric codes from affine plane curves," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 1003-1012, July 1994.

[25] G.-L. Feng and T.R.N. Rao, "Improved geometric Goppa codes Part I, Basic Theory," *IEEE Trans. Inform. Theory* special issue on AG codes.

[26] G.-L. Feng and T.R.N. Rao, "Erasures-and-errors decoding of algebraic geometric codes," *Proceedings 1993 IEEE ITW*, Susono-shi, Shizuoka, Japan, pp. 79-80, June 1993.

[27] G.-L. Feng and K.K. Tzeng, "A generalization of the Berlekamp-Massey algorithm for multisequence shift-register synthesis with applications to decoding cyclic codes," *IEEE Trans. Inform. Theory*, vol. IT-37, pp. 1274-1287, Sept. 1991.

[28] G.-L. Feng and K.K. Tzeng, "A new procedure for decoding cyclic and BCH codes up to actual minimum distance," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 1364-1374, Sept. 1994.

[29] G.-L. Feng, V. Wei, T.R.N. Rao and K.K. Tzeng, "Simplified understanding and efficient decoding of a class of algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 981-1002, July 1994.

[30] P. Fitzpatrick and G.H. Norton, "Finding a basis for the characteristic ideal of an n-dimensional linear recurring sequence," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 1480-1487, Nov. 1990.

[31] G.D. Forney, "Generalized minimum distance decoding," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 125-131, April 1966.

[32] W. Fulton, *Algebraic curves. An introduction to algebraic geometry*, W.A. Benjamin Inc., New York Amsterdam, 1969.

[33] V.D. Goppa, "Codes associated with divisors," *Probl. Peredachi Inform.* vol. 13 (1), pp. 33-39, 1977. Translation: *Probl. Inform. Transmission*, vol. 13, pp. 22-26, 1977.

[34] V.D. Goppa, "Codes on algebraic curves," *Dokl. Akad. Nauk SSSR* vol. 259, pp. 1289-1290, 1981. Translation: *Soviet Math. Dokl.*, vol. 24, pp. 170-172, 1981.

[35] V.D. Goppa, "Algebraico-geometric codes," *Izv. Akad. Nauk SSSR*, vol. 46, 1982. Translation: *Math. USSR Izvestija*, vol. 21, pp. 75-91, 1983.

[36] V.D. Goppa, "Codes and information," *Russian Math. Surveys*, vol. 39, pp. 87-141, 1984.

[37] V.D. Goppa, *Geometry and codes*, Mathematics and its Applications, vol. 24, Kluwer Acad. Publ., Dordrecht, 1991.

[38] J.P. Hansen, "Codes from the Klein quartic, ideals and decoding," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 923-925, Nov. 1987.

[39] J.P. Hansen and H. Stichtenoth, "Group codes on certain algebraic curves with many rational points," *AAECC*, vol. 1, pp. 67-77, 1990.

[40] R. Hartshorne, *Algebraic geometry*, Graduate Texts in Math. vol. 52, Springer-Verlag, Berlin Heidelberg New York, 1972.

[41] A. Havemose, "Decoding algebraic geometric codes," Ph.D. thesis, Danmarks Tekniske Højskole, Aug. 1989.

[42] T. Henocq, "Jacobienne et fonction zeta des courbes algébriques. Décodage des codes géométriques," Ph.D. thesis, Univ. of Paul Sabatier, Toulouse, Dec. 1994.

[43] T. Henocq and D. Rotillion, "The Theta divisor of a Jacobian variety and the decoding of geometric Goppa codes," to appear in *Journ. Pure and Applied Algebra*.

[44] Y. Ihara, "Some remarks on the number of rational points of algebraic curves of finite fields," *Journ. Fac. Sc. Univ. Tokyo* IA, vol. 28, pp. 721-724, 1981.

[45] C.Dahl Jensen, "Codes and geometry," Ph.D. thesis, Denmarks Tekniske Højskole, May 1991.

[46] C. Dahl Jensen, "Fast decoding of codes from algebraic geometry," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 223-230, Jan. 1994.

[47] H. Elbrønd Jensen, T. Høholdt and J. Justesen, "On the number of correctable errors for some AG-codes," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 681-684, March 1993.

[48] 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.

[49] J. Justesen, K.J. Larsen, H. Elbrønd Jensen and T. Høholdt, "Fast decoding of codes from algebraic plane curves," *IEEE Trans. Inform. Theory*, vol. IT-38, pp. 111-119, Jan. 1992.

[50] G.L. Katsman and M.A. Tsfasman, "Spectra of algebraic-geometric codes," *Probl. Peredachi Inform.* **23** (4) (1987), 19-34. Translation: *Probl. Inform. Transmission* **23** (1987), 262-275.

[51] G.L. Katsman, M.A. Tsfasman and S.G. Vlăduţ, "Spectra of linear codes and error probability of decoding," *Proceedings AGCT-3, Luminy 1991, Lect. Notes Math.*, vol. 1518, pp. 82-99, 1992.

[52] C. Kirfel, "On the Clifford defect for special curves," to appear in the *Proceedings of AGCT-4, Luminy 1993*, Walter de Gruyter.

[53] C. Kirfel and R. Pellikaan, "The minimum distance of codes in an array coming from telescopic semigroups," to appear in the special issue on AG codes of *IEEE Trans. Inform. Theory.*

[54] R. Kötter, "A unified description of an error locating procedure for linear codes," *Proceedings ACCT-3, Voneshta Voda*, June 1992.

[55] R. Kötter, "Fast generalized minimum distance distance decoding of algebraic geometric and Reed Solomon codes," preprint Linköping, Sweden, 1993.

[56] V. Yu. Krachkovskii, "Decoding of codes on algebraic curves," (in Russian), Conference Odessa, 1988.

[57] N. Lausten, "Design og afkodning af algebraiske geometrikoder," Masters's Thesis, Danmarks Tekniske Højskole, March 1993.

[58] D. Le Brigand, "Decoding codes on hyperelliptic curves," *Lect. Notes Comp. Sc.* vol. 514, pp. 126-134, 1991.

[59] D. Le Brigand, "Sur les codes géométriques: codage et décodage. Problème du nombre de classes dans les corps de fonctions," Habilitation, Univ. Pierre et Marie Curie, Paris 6, Jan. 1995.

[60] D. Leonard, "Error-locator ideals for algebraic-geometric codes," *IEEE Trans. Inform. Theory* vol. IT-41, pp. 819-824, May 1995.

[61] D. Leonard, "A generalized Forney formula for AG codes," submitted to *IEEE Trans. Inform. Theory.*

[62] J.H. van Lint, "Algebraic geometric codes," in *Coding Theory and Design Theory*, part I, IMA Volumes Math. Appl. vol. 21, pp 137-162, Springer-Verlag, Berlin, 1990.

[63] J.H. van Lint and G. van der Geer, *Introduction to coding theory and algebraic geometry*, DMV Seminar vol. 12, Birkhäuser Verlag, Basel Boston Berlin, 1988.

[64] J.H. van Lint and T.A. Springer, "Generalized Reed-Solomon codes from algebraic geometry," *IEEE Trans. Inform. Theory* vol. IT-33, pp. 305-309, May 1987.

[65] Y. Madelung, "Implementation of a decoding algorithm for AG-codes from the Hermitian curve," report IT-93-137. The Technical Univ. of Denmark, 1993.

[66] J.L. Massey, *Threshold decoding*, M.I.T Press, Cambridge, Massachusetts, 1963.

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

[68] R.J. McEliece, "A public-key crypto system based on algebraic coding theory," DSN Progress Report 42-44, JPL, Pasadena, Jan. and Feb. 1978.

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

[70] C. Moreno, *Algebraic curves over finite fields*, Cambridge Tracts in Math. vol. 97, Cambridge Univ. Press, Cambridge, 1991.

[71] R. Pellikaan, "On decoding linear codes by error correcting pairs," preprint Eindhoven Univ. of Techn., 1988.

[72] R. Pellikaan, "On a decoding algorithm for codes on maximal curves," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 1228-1232, Nov. 1989.

[73] R. Pellikaan, "On the gonality of curves, abundant codes and decoding," *Proceedings AGCT-3, Luminy 1991, Lect. Notes Math.*, vol. 1518, pp. 132-144, 1992.

[74] R. Pellikaan, "On the decoding by error location and the number of dependent error positions," *Discrete Math.*, vol. 106/107, pp. 369-381, 1992.

[75] 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.

[76] R. Pellikaan, "On the existence of error-correcting pairs," to appear in *Proceedings of the Shanghai Conference on Designs, Codes and Finite Geometries, Journal of Statistical Planning and Inference*, 1992.

[77] R. Pellikaan, "The shift bound for cyclic, Reed-Muller and geometric Goppa codes," to appear in *Proceedings AGCT-4, Luminy 1993*, Walter de Gruyter.

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

[79] S.C. Porter, "Decoding codes arising from Goppa's construction on algebraic curves," Ph. D. thesis, Yale Univ., Dec. 1988.

[80] S.C. Porter, B.-Z. Shen and R. Pellikaan, "On decoding geometric Goppa codes using an extra place," *IEEE Trans. Inform. Theory*, vol. IT-38, pp. 1663-1676, Nov. 1992.

[81] D. Rotillon and J.A. Thiong Ly, "Decoding codes on the Klein quartic," *Proceedings Eurocode 90, Lect. Notes in Comp. Sc.*, vol. 514, pp. 135-150, 1991.

[82] L.D. Rudolph and C.R.P. Hartmann, "Decoding by sequential code reduction," *IEEE Trans. Inform. Theory*, vol. IT-19, pp.549-555, July 1973.

[83] K. Saints and C. Heegard, "On hyperbolic cascaded Reed-Solomon codes," *Proceedings AAECCC-10, San Juan de Puerto Rico, May 1993, Lect. Notes Comp. Sc.*, vol. 673, pp. 291-303, Springer-Verlag, Berlin etc., 1993.

[84] K. Saints and C. Heegard, "Algebraic-geometric codes and multidimensional cyclic codes: A unified theory and algorithms for decoding using Gröbner bases," to appear in *IEEE Trans. Inform. Theory.*

[85] 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.

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

[87] S. Sakata, "Decoding binary 2-D cyclic codes by the 2-D Berlekamp-Massey algorithm," *IEEE Trans. Inform. Theory*, vol. 37, pp. 1200-1203, July 1991.

[88] S. Sakata, "Fast erasure-and-error decoding of any one-point AG code up to the Feng-Rao bound," preprint March 1995.

[89] S. Sakata, J. Justesen, Y. Madelung, H. Elbrønd Jensen and T. Høholdt, "Fast decoding of algebraic geometric codes up to the designed minimum distance," *IEEE Trans. Inform. Theory* special issue on AG codes.

[90] S. Sakata, H. Elbrønd Jensen and T. Høholdt, "Generalized Berlekamp-Massey decoding of algebraic geometric codes up to half the Feng-Rao bound, to appear in *IEEE Trans. Inform. Theory.*

[91] S. Sakata, J. Justesen, Y. Madelung, H. Elbrønd Jensen and T. Høholdt, "A fast decoding method of AG codes from Miura-Kamiya curves C_{ab} up to Half the Feng-Rao bound," *Finite Fields and their Applications* vol. 11, pp. 83-101, 1995.

[92] A. Schönhage, "Equation solving in terms of complexity," *Proceedings ICM 1986*, Berkeley, vol. 1, Amer. Math. Soc., pp. 131-153, 1986.

[93] R. Sheldrake, *The presence of the past; morphic resonance and the habits of nature*, Times Books, New York, 1988.

[94] B.-Z. Shen, "Solving a congruence on a graded algebra by a subresultant sequence and its application," *Journ. of Symbolic Computation*, vol. 14, pp. 505-522, 1992.

[95] B.-Z. Shen, "On encoding and decoding of the codes from Hermitian curves," *Cryptography and Coding III, the IMA Conference Proceedings Series*, Oxford Univ. Press, New Series Number 45, pp. 337-356, Clarendon Press, Oxford, 1993.

[96] B.-Z. Shen, "Algebraic-geometric codes and their decoding algorithm," Ph.D. thesis, Eindhoven Univ. of Techn., Sept. 1992.

[97] B.-Z. Shen and K.K. Tzeng, "Decoding geometric Goppa codes up to designed minimum distance by solving a key equation in a ring," to appear in *IEEE Trans. Inform. Theory.*

[98] B.-Z. Shen and K.K. Tzeng, "Generation of matrices for determining minimum distance and decoding of algebraic-geometric codes" to appear in *IEEE Trans. Inform. Theory.*

[99] B.-Z. Shen and K. K. Tzeng, "A code decomposition approach for decoding cyclic and algebraic-geometric codes," to appear in *IEEE Trans. Inform. Theory.*

[100] A.N. Skorobogatov and S.G. Vlăduţ, "On the decoding of algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 1051-1060, Nov. 1990.

[101] H. Stichtenoth, "A note on Hermitian codes over GF(q^2)," *IEEE Trans. Inform. Theory*, vol. IT-34, pp. 1345-1348, Nov. 1988.

[102] H. Stichtenoth, *Algebraic function fields and codes*, Universitext, Springer-Verlag, Berlin 1993.

[103] V. Strassen, "Algebra and complexity," *First European Congress of Math.*, vol II (part 2), Progress in Math. 120, pp. 429-446, Birkhäuser Verlag, Basel, 1994.

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

[105] M. O'Sullivan, "Decoding of Codes Defined by a Single Point on a Curve," *IEEE Trans. Inform. Theory* special issue on AG codes.

[106] H.J. Tiersma, "Codes coming from Hermitian curves," *IEEE Trans. Inform. Theory*, vol. IT-33, pp. 605-609, July 1987.

[107] M.A. Tsfasman and S.G. Vlăduţ, *Algebraic-geometric codes*, Mathematics and its Applications vol. 58, Kluwer Acad. Publ., Dordrecht, 1991.

[108] M.A. Tsfasman, S.G. Vlăduţ and T. Zink, "Modular curves, Shimura curves and Goppa codes, better than Varshamov-Gilbert bound," *Math. Nachrichten*, vol. 109, pp. 21-28, 1982.

[109] S.G. Vlăduţ, "On the decoding of algebraic-geometric codes over GF(q) for q$\geq$ 16," *IEEE Trans. Inform. Theory*, vol IT-36, pp. 1461-1463, Nov. 1990.

[110] C. Voss and T. Høholdt, "A family of Kummer extensions of the Hermitian function field," to appear in *Communications in Algebra*.

[111] S.B. Wicker and V.K. Bhargava (eds.), *Reed-Solomon codes and their applications*, IEEE Press, New York, 1994.

[112] T. Yaghoobian and I.F. Blake, "Hermitian codes as generalized RS codes," *Designs, Codes and Cryptogrphy*, vol. 2, pp. 15-18, 1992.