This is to certify that the

dissertation entitled

THE TRACE OPERATOR AND GENERALIZED GOPPA CODES

presented by

Albert Manuel Roseiro

has been accepted towards fulfillment
of the requirements for

Ph. D. degree in Electrical Engineering

Major professor

Date July 6, 1989.

THE TRACE OPERATOR AND GENERALIZED GOPPA CODES

By

Albert Manuel Roseiro

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1989

ABSTRACT

THE TRACE OPERATOR AND GENERALIZED GOPPA CODES

By

Albert Manuel Roseiro

The computation of the dimension of an error correcting block code is essential to achieve an implementation of these codes in practice. Most of the known results on the dimension of block codes are usually derived by exhaustive search using a computer which limits the number of codes that can be studied.

A new analytical method has been developed for the study of the dimension of generalized Goppa codes using properties of the trace operator over finite fields. This method does not require the use of a computer and can be applied to the family of generalized Goppa codes.

New bounds have been obtained for a general class of Goppa codes analytically. Two specific set of Goppa codes defined by $G_1(X) = X^{2^r}+X$ and $G_2(X) = X^{2^r+1}+1$ over a $GF(2^{2s})$ locator field are studied in detail and tighter bounds than previously reported in the literature are derived for any $s>1$.

With tender love and affection for Evelyne

ACKNOWLEDGMENTS

The author is grateful to major Professor M. Siegel for his guidance throughout this research. He wishes to thank guidance committee members Pr. J. Hall, Pr. J. Adney, Pr. H. Rigas and Pr. R. Zapp for their encouragement and support. He is also thankfull to the Department of Electrical Engineering for making office and computers facilities available. Finally, he acknowledges all the other personal advices given by Dr. O. Krauss and Pr. E. Strangas.

TABLE OF CONTENTS

LIST OF TABLES

INTRODUCTION

During 1948 and 1949, Shannon [1,2] published very interesting results about the fundamentals of information theory. The notion of channel capacity showed that there is always a way of coding information to obtain a probability of error during a transmission as small as possible given that the rate of transmission is compatible with the properties of the channel introducing the errors. Unfortunately, Shannon's channel capacity theorem doesn't indicate which coding method should be used.

In 1950, Hamming [3] and Golay [4] were able to lay down the fundamentals of error detection and correction coding theory. The Hamming codes could correct one error but still were very far from the limits of information theory. Hocquenghem [5] in 1959, Bose and Chaudhuri [6] in 1960 introduced an important family of codes known as BCH that could correct more than one error and generalized the Hamming codes. In fact, BCH codes are a subset of the Reed-Solomon codes found by Reed and Solomon [7]. Still, BCH code cannot reach the theoretical limits announced by information theory.

Peterson [8,9] was the first to introduce in 1960 an algebraic decoding method for BCH codes and later, an even more efficient method was derived by Berlekamp [10].

Important parameters associated with linear error correcting block codes are the length n, the dimension k, and the distance d. These codes are usually referred to as (n,k,d)-codes. The problem from a theoretical standpoint is for a given distance and length to find a code

having the largest dimension possible. A family of codes is said to be good asymptotically if for a fixed $d/n > 0$, there exist codes in the family with $k/n > 0$ as $n \rightarrow \infty$. It has been shown that primitive BCH codes are not good asymptotically, Lin and Weldon [11], Berlekamp [12]. Nevertheless, the Varshamov-Gilbert bound indicates that linear block codes are good, Peterson [9, pp. 51-52]. Interestingly, two sub-families of the linear block codes, namely the Alternant family (Helgert [13,14,15], Mac Williams and Sloane [16, pp. 332-350]) and the Goppa family (Goppa [17,18]) still reach the Varshamov-Gilbert bound.

The standard decoding methods are Euclid's algorithm, Mac Williams and Sloane [16, pp. 365-368], Berlekamp's algorithm, Berlekamp [10] and MPR (minimal partial realization) Conan [19]. These algorithms indicate typically that for a binary Goppa code of length n and constructive distance $d_c = 2t+1$, the redundancy $n-k$ is at most mt where a Galois field $GF(2^m)$ is used as a locator field. The Varshamov-Gilbert bound shows that for some Goppa codes, the actual true minimum distance can be greater than the constructive distance even if the redundancy remains mt. Since it is only possible to decode up to the constructive distance with the actually known decoding algorithms (the minimum distance decoding method, Lin and Costello [20] is not considered in the discussion), the actual problem is to find for a given constructive distance and length, a code with the largest dimension k or equivalently the smallest redundancy $n-k$.

The standard analytical approaches to finding good Goppa codes are partitioning and algebraic transformations, Moreno [21], Chen [22], Berman *et al* [23]. Thus far, the study of the rank of the parity check matrix has been done by computer search or by the use of minimal polynomials in the case of BCH codes.

By using the trace operator over $GF(p^m)$ (Berlekamp *et al* [24], Mac williams and Sloane [16]) a new equation referred to, as the redundancy equation of a generalized Goppa

code can be obtained. The solutions of that equation form a vector space over $GF(p)$. The dimension of that vector space is related to the true redundancy, namely, the number of independent rows over $GF(p)$ of the parity check matrix of a generalized Goppa code. A computer, thus is not needed to find the dimension of such codes given that the redundancy equation can be solved analytically.

Applying the derived equations to specific codes has provided original bounds (not previously reported) on the dimension of a general class of binary Goppa codes. In 1984, Loeloeian and Conan [25,26] introduced a set of Goppa codes defined by $G_1(X) = X^{2^r}+X$ and locator field $GF(2^{2s})$. In 1987, Bezzateev and Shekhunova [27] found a (55,16,19) Goppa code defined by $G(X) = X^9+1$ with locator field $GF(2^6)$. This latter code is generalized here by introducing the polynomial $G_2(X) = X^{2^r+1}+1$ for any $s > 1$ over a locator field $GF(2^{2s})$. Tighter bounds on the dimension of the two previous sets of codes will be obtained by partially solving the redundancy equation (a simulation has shown that these bounds are actually met for $s = 2,3,4,5$).

Since the (55,16,19)-code (case $s=3$) is for the moment the best binary linear block code known for $n=55$ and $d_c=19$ (Verhoeff [28]), codes defined by $G_2(X)$ are interesting especially for values of $s>3$ and may have practical applications.

Chapter 1 contains a general survey of linear error correcting block codes. The generalized Goppa family (Loeloeian and Conan [29]) has been chosen for the sake of clarity (this family is strictly equivalent to the Alternant family).

Chapter 2 is devoted to important properties of the trace operator and its extension to rational polynomial ring modulo $X^{p^m}-X$.

Chapter 3 derives some new relationships between the dimension of generalized Goppa codes and the trace operator; the redundancy equation is then defined. A particular case of binary Goppa codes where $G^{2^s}(X) \equiv G(X) \ mod \ (X^{2^{2s}} + X)$ allows the redundancy equation to be solved partially, and original bounds on the dimension of these codes to be obtained.

Finally, Chapter 4 provides a further study of the dimension of the binary Goppa codes defined by $G_1(X)$ and $G_2(X)$ for $s>1$.

Due to the extreme importance of finite field algebra, Appendix A provides a review of the basic properties of such fields.

CHAPTER 1

REVIEW OF THE LINEAR ERROR CORRECTING BLOCK CODES

1.1. Introduction:

The role of linear error correcting block codes is well established and the applications of such codes is constantly growing. It appears reasonable to study one of the largest linear family known, the generalized Goppa codes introduced by Loeloeian and Conan [29]. This family contains in particular all the Hamming codes, BCH codes, Goppa codes and is equivalent to the Alternant family introduced earlier by Helgert [13,14,15].

1.2. What is an error correcting block code?

Important parameters associated with an error correcting block code are the length n, the dimension k and the distance d. These codes are usually refered to as (n , k , d)-code. There are k symbols of information available to the user. Through an isomorphic mapping, the k symbols of information are uniquely mapped to n symbols ($n > k$). This operation is equivalent to add $n-k$ symbols of redundancy to the k symbols of information thus providing an error correcting capacity. The mapping (or encoding) is closely related to the structure of the code used.

The distance d of the code defines exactly the maximum number of independent errors the code can correct. The decoding consists of actually correcting the errors that have occurred during a transmission, whenever feasible.

1.3. Hamming distance and maximum likelihood decoding method:

A block code of length n can be seen as a set of n-tuples with coefficients belonging to some set.

Definition 1.1. The Hamming distance (or Hamming weight) between two n-tuples C_1 and C_2 is:

$$d(C_1 , C_2) = \sum_{i=1}^{n} \begin{Bmatrix} 0 \text{ if } c_{1i} = c_{2i} \\ 1 \text{ if } c_{1i} \neq c_{2i} \end{Bmatrix}$$

It can be verified from Def. A.12. that the above definition is really a distance.

Definition 1.2. The distance of a block code C is defined by:

$$d_C = min(d(C_i , C_j) \mid C_i , C_j \in C , i \neq j)$$

Proposition 1.1. If a code C has distance $d = 2t+1$, then it is possible when using the minimum distance decoding scheme to correct up to t errors in any codeword. This is called a decoding situation as opposed to a non decoding situation when more than t errors have occurred.

Proof. Since the distance is $2t+1$, it is possible by using the induced geometry of the Hamming distance to put spheres of radius t around each codeword, each sphere having a codeword at the center and no other codeword being contained inside each sphere. The decoding consists first of computing the distance of the received codeword to all the possible codewords. Then, if the minimum of all the previous distances is less than or equal to t, the

corresponding codeword is the corrected codeword.

1.4. Channel capacity:

Shannon [1,2] has shown that when a channel introduces an error bit probability p uniformly distributed, there always exists a binary block code (the coefficients of the codewords being only 0 or 1) that can be transmitted with a probability of error as small as possible given that:

$$k/n \leq 1-H_2(p)$$

where $H_2(x) = -x\log_2 x - (1-x)\log_2(1-x)$ is the entropy function. In fact, Shannon proved that if one tries to transmit information at a rate higher than that predicted by the channel capacity, it is then not possible to transmit without errors.

1.5. Complexity considerations:

The first problem that arises with Shannon's channel capacity is that the proof is only an existence one and actually doesn't tell how to choose good block codes. Furthermore, an exhaustive computer search is impractical.

Additionally, the minimum distance decoding method can be very difficult and costly to implement in practice when n becomes large since it would be necessary to store in memory 2^k n-tuples plus comparing them each time to the received word in order to compute the Hamming distance.

For all these reasons, considerable research has been done to put some specific algebraic structure on codes which would not require as much memory to store all the codewords, and especially that could correct the errors without having to compute the Hamming

distance but rather decode by algebraic methods closely related to the structure of the code.

Since the code lengths of interest are finite, it appears normal to use finite field algebra to induce algebraic properties on these codes.

1.6. Linear block codes:

Given a code C of length n, each codeword being a n-tuples with coefficients belonging to some finite field $GF(q)$. This code might be then viewed as a $GF(q)^n$ vector space over $GF(q)$ when using Def. A.10.

Definition 1.3. A block code C is said to be linear if and only if it forms a vector space over $GF(q)$ with the two binary composition corresponding to Def. A.11.:

(i) $C_1 , C_2 \varepsilon C$ then $C_1+C_2 \varepsilon C$

(ii) $C_1 \varepsilon C$ and $\lambda \varepsilon GF(q)$ then $\lambda C_1 \varepsilon C$.

Since a linear code is a vector space and there are by construction finitely many codewords, C can be generated by a finite basis, in other words, C has a finite dimension over $GF(q)$. The dimension of the code is usually denoted k. Let's call $\{G_1 , G_2 , \ldots , G_k\}$ the basis of C, then every codeword can be represented as a linear combination over $GF(q)$ of the G_i's. Representing each vector of the basis with the n-tuples notation, the linear combination can be rewritten with a matrix, namely:

$$\begin{bmatrix} C_1 \\ C_2 \\ . \\ . \\ C_n \end{bmatrix} = \begin{bmatrix} G_{11} & G_{21} & . & . & G_{k1} \\ G_{12} & G_{22} & . & . & G_{k2} \\ . & . & . & . & . \\ . & . & . & . & . \\ G_{1n} & G_{2n} & . & . & G_{kn} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ . \\ I_k \end{bmatrix}$$

The vector $(I_1, I_2, \ldots, I_k)$ represents the k symbols of information that have to be encoded by the code C defined by the above matrix. In fact, there is a one to one correspondence between all the linear codes (n, k, d) and the set of all the matrices $(n \times k)$ with coefficients over $GF(q)$. The matrix G is called the generator matrix of the code. Since the dual of a vector space of the dimension k is also a vector space of dimension $n-k$, there is a matrix H of size $(n-k \times n)$ known as the parity check matrix such that:

$$HG = 0$$

It will be seen later on how to get the distance d from the matrix H.

1.7. Separable codes:

Definition 1.4. A block code is said to be separable if and only if it is possible to separate after encoding the k bits of information from the $n-k$ bits of redundancy.

Proposition 1.2. For any separable block code, $d \leq n-k+1$ (otherwise known as the Singleton bound).

Proof. The proof is provided only for linear codes. Since the code is separable, the smallest Hamming weight of k-tuple information is 1 (otherwise the encoding would give the null codeword in the linear case). Then, the worst case is after encoding to have all the redundancy bits not equal to 0 indicating a distance of at most $n-k+1$.

For the non linear case, see Delsarte [30] *Q.E.D.*

Proposition 1.3. All the linear block codes are separable.

Proof. It is a well known fact that by linear combination of rows and eventual column permutations, it is possible to transform the parity check matrix to obtain a separated form of the parity check matrix:

$$H_s = (I_{n-k}\ E)$$

where I_{n-k} is the identity matrix of rank $n-k$. Whenever a column permutation was required to derive H_s , the parity check matrices H and H_s still define the same code (they are said to be equivalent) as long as the corresponding symbol coordinate is permutted.

One possible separated form for separated generator matrix can be:

$$G_s = \begin{bmatrix} -E \\ I_k \end{bmatrix}$$

Multiplying the matrix G_s by an information vector of dimension k gives a vector of length n, the last k coefficients of the corresponding codeword are the k informations symbols. *Q.E.D.*

The control matrix can be defined as a parity check matrix having some linear dependent rows added to it. Later on, it will be common to regroup the rows by pack and represent them with elements of some $GF(q^m)$, $m \geq 1$.

It is important to remember that not all separable codes are linear !

1.8. The distance of a linear code:

Proposition 1.4. If a linear block code has a distance d, then it is impossible to find a non null codeword of weight less than d belonging to the kernel of the parity check matrix (or of the control matrix).

Proof. From Def. 1.2., the distance d of a block code C is the minimum Hamming distance between every possible pair of codewords. Since for a linear code, the sum or difference of two codewords is a codeword of the same code, this implies that the null codeword always belongs to any linear code.

Then, from the linearity argument, it is sufficient to find the non-zero codewords with the smallest weight that belongs to the kernel of H. If the distance of a code is d, then there is no way that a non-zero codeword C_1 of weight less than d could verify $HC_1 = 0$ *Q.E.D.*

When performing the decoding of a linear code, a case of false decoding might appear since there could be a number of d errors which would send a given codeword to another codeword at a distance d. The result belonging of course to the kernel of H, the user would then think that no errors have occurred!

1.9. Other bounds:

Proposition 1.5. The family of linear codes reach the Varshamov-Gilbert bound, namely there always exist a $(n\ ,\ k\ ,\ d)$-code such that:

$$\frac{n-k}{n} \leq H_2(\frac{d-1}{n})$$

Proof. See Peterson [9, pp. 51-52].

Unfortunately, the proof of Prop. 1.5 is an existence proof and doesn't indicate how to choose good codes inside the linear family of block codes. Information theory says that for $n \rightarrow \infty$, the code has to correct an average of $t = np$ errors (if p is the average error bit probability) in order to ensure that the probability of losing a block of information is as small as possible. From Prop. 1.5., this implies that $\frac{d-1}{n} = \frac{2t}{n} \rightarrow 2p$ so:

$$\frac{k}{n} \geq 1 - H_2(2p)$$

For small values of p, the asymptotic behaviour of the Varshamov-Gilbert bound is very close to the channel capacity, thus the linear family of block code is an interesting family to study.

1.10. The generalized Goppa family:

This family was introduced by Loeloeian and Conan [29] and it will be shown later that the Alternant family is equivalent to the generalized Goppa family. The approach used by Loeloeian is very practical because of its simplicity.

Definition 1.5. Let's choose three polynomials $G(X)$, $P(X)$ and $\pi(X)$ with coefficients over $GF(q^m)$ and respective degrees r, s and n. It is necessary that $\pi(X)$ splits entirely in $GF(q^m)$. Let's $\alpha_1, \alpha_2, \ldots, \alpha_n$ be the n roots of $\pi(X)$ with the restriction that the α_i's are not roots of $G(X)$ and $P(X)$. Then the generalized Goppa code $\Gamma(\pi(X), P(X), G(X))$ of length n consists of all the codewords $(a_1, a_2, \ldots, a_n)$ belonging to $GF(q)^n$ such that:

$$\sum_{i=1}^{n} \frac{a_i P(\alpha_i)}{X-\alpha_i} \equiv 0 \quad mod\ G(X) \tag{1.1}$$

Lemma 1.1. If $gcd(G(X), X-\alpha) = 1$ then:

$$\frac{1}{X-\alpha} \equiv \frac{-(G(X)-G(\alpha))G^{-1}(\alpha)}{X-\alpha} \quad mod\ G(X) \tag{1.2}$$

Proof. Since $gcd(G(X), X-\alpha) = 1$, it is equivalent to say that $G(\alpha) \neq 0$ or, $G(X)$ and $X-\alpha$ are relatively prime. Then from Theorem A.1., there exist two polynomials $U(X)$ and $V(X)$ with *deg* $U(X)$ and *deg* $V(X)$ both less than *deg* $G(X)$ such that:

$$U(X)(X-\alpha)+V(X)G(X) = 1 \tag{1.3}$$

Furthermore, doing an Euclidian division of $G(X)$ by $X-\alpha$ yields:

$$G(X) = \lambda(X)(X-\alpha)+G(\alpha) \tag{1.4}$$

Eq. (1.3) shows that $\frac{1}{X-\alpha} \equiv U(X)$ *mod* $G(X)$. Then from Eq. (1.3) and (1.4):

$$U(X)(X-\alpha)+V(X)G(X) = G^{-1}(\alpha)(G(X)-\lambda(X)(X-\alpha)) \tag{1.5}$$

By identifying the factors of $G(X)$ and $(X-\alpha)$ in Eq. (1.5), it is found:

$$\begin{cases} V(X) = G^{-1}(\alpha) \\ U(X) = -\lambda(X)G^{-1}(\alpha) \end{cases} \tag{1.6}$$

Finally, using Eq. (1.6) and (1.4) yields:

$$U(X) = \frac{-G^{-1}(\alpha)(G(X)-G(\alpha))}{X-\alpha} \tag{1.7}$$

In other words, the inverse of $(X-\alpha)$ *mod* $G(X)$ is:

$$U(X) = -G^{-1}(\alpha)\sum_{j=1}^{r} g_j \sum_{l=0}^{j-1} \alpha^{j-l-1}X^l \tag{1.8}$$

Proposition 1.6. The control matrix of a generalized Goppa code is:

$$H = \begin{bmatrix} 1 & 1 & . & . & 1 \\ \alpha_1 & \alpha_2 & . & . & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & . & . & \alpha_n^2 \\ . & . & . & . & . \\ . & . & . & . & . \\ \alpha_1^{r-1} & \alpha_2^{r-1} & . & . & \alpha_n^{r-1} \end{bmatrix} \begin{bmatrix} \frac{P(\alpha_1)}{G(\alpha_1)} & 0 & . & . & 0 \\ 0 & \frac{P(\alpha_2)}{G(\alpha_2)} & . & . & . \\ . & 0 & . & . & . \\ . & . & . & . & 0 \\ 0 & 0 & . & . & \frac{P(\alpha_n)}{G(\alpha_n)} \end{bmatrix} \tag{1.9}$$

Proof. Replacing Eq. (1.8) into Eq. (1.1) yields:

$$\sum_{i=1}^{n} \frac{a_i P(\alpha_i)}{X-\alpha_i} \equiv -\sum_{i=1}^{n} a_i P(\alpha_i) G^{-1}(\alpha_i) \sum_{j=1}^{r} g_j \sum_{l=0}^{j-1} \alpha_i^{j-l-1} X^l \quad mod\ G(X) \tag{1.10}$$

For the sake of clarity, the following matrices are defined:

$$\Delta = \begin{bmatrix} g_r & 0 & . & . & 0 \\ g_{r-1} & g_r & . & . & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ g_1 & g_2 & . & . & g_r \end{bmatrix} \qquad V = \begin{bmatrix} 1 & 1 & . & . & 1 \\ \alpha_1 & \alpha_2 & . & . & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & . & . & \alpha_n^2 \\ . & . & . & . & . \\ . & . & . & . & . \\ \alpha_1^{r-1} & \alpha_2^{r-1} & . & . & \alpha_n^{r-1} \end{bmatrix}$$

$$D = \begin{bmatrix} \dfrac{P(\alpha_1)}{G(\alpha_1)} & 0 & . & . & 0 \\ 0 & \dfrac{P(\alpha_2)}{G(\alpha_2)} & . & . & . \\ . & 0 & . & . & . \\ . & . & . & . & 0 \\ 0 & 0 & . & . & \dfrac{P(\alpha_n)}{G(\alpha_n)} \end{bmatrix}$$

Since the degree of Eq. (1.10) is less that r and the computations are done in a residue class ring modulo a polynomial of degree r, it is equivalent to cancelling each power of X^i for $i = 0, 1, \ldots, r-1$. Using the previous matrices, it is derived:

$$\Delta V D \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ a_n \end{bmatrix} = 0$$

Δ being diagonal and $g_r \neq 0$(because the degree of $G(X)$ is equal to r), it is then invertible. Multiplying both sides by Δ^{-1} yields the desired form for H. *Q.E.D.*

Proposition 1.7. The family of Alternant code is strictly equivalent to the family of the generalized Goppa code for a fixed $\pi(X)$, n and r.

Proof. Defining $y_i = P(\alpha_i)G^{-1}(\alpha_i)$ shows that generalized goppa code is also an Alternant code (for the definition of Alternant code, see Helgert [14,15] or Mac Williams and

Sloane [16]).

Now for a given y_i's and n, pick any $G(X)$ of degree r such that $G(\alpha_i) \neq 0$. Then, using the Lagrange interpolation formula, it is possible to derive $P(X)$ by:

$$P(X) = \sum_{i=1}^{n} y_i G(\alpha_i) \prod_{\substack{j=1 \\ j \neq i}}^{n} \frac{X-\alpha_j}{\alpha_i - \alpha_j} \quad Q.E.D.$$

Proposition. 1.8. The generalized Goppa code $\Gamma(\pi(X), P(X), G(X))$ defined over $GF(q^m)$ and with coefficients over $GF(q)$ has the following parameters:

$$\begin{cases} n = deg\ \pi(X) \\ r = deg\ G(X) \\ n-k \leq mr \\ r+1 \leq d \leq mr+1 \end{cases} \tag{1.11}$$

Proof. $n = deg\ \pi(X)$ and $r = deg\ G(X)$ follows imediately from Def. 1.15 and Eq. (1.9). $n-k$ is the rank of the control matrix defined by Eq. (1.9) when projected over $GF(q)$ using a basis of m elements to represent $GF(q^m)$; there are r rows in $GF(q^m)$ so using any basis with m vectors, mr rows are derived over $GF(q)$. In the worst case, there no dependent rows over $GF(q)$ so $n-k \leq mr$.

Supposing there is a non-zero codeword with Hamming weight $d \leq r$ having coefficients over $GF(q)$, then this codeword by definition would belong to the kernel of H (Eq. (1.9)). The product $a_i G^{-1}(\alpha_i)$ has also the same weight as a_i since $G(\alpha_i) \neq 0$. Thus if $d \leq r$ is possible, that would mean that the Vandermonde of order r has a non zero solution, which is impossible.

It is then clear that $d \geq r+1$. The fact that $d \leq mr+1$ is due from Prop. 1.2. *Q.E.D.*

Proposition 1.9. It is a well known fact that the Alternant family reach the Varshamov Gilbert bound (so do the generalized Goppa family).

Proof. See Mac Williams and Sloane [16].

Proposition 1.10. If $G(X)$ is separable in some splitting field, let's denote $x_1, x_2, \ldots, x_r$ its r distinct roots. Then, another form for the control matrix of a generalized Goppa code over the splitting field of $G(X)$ is:

$$\begin{bmatrix} \frac{P(\alpha_1)}{x_1-\alpha_1} & \frac{P(\alpha_2)}{x_1-\alpha_2} & \cdots & \frac{P(\alpha_n)}{x_1-\alpha_n} \\ \frac{P(\alpha_1)}{x_2-\alpha_1} & \frac{P(\alpha_2)}{x_2-\alpha_2} & \cdots & \frac{P(\alpha_n)}{x_2-\alpha_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{P(\alpha_1)}{x_r-\alpha_1} & \frac{P(\alpha_2)}{x_r-\alpha_2} & \cdots & \frac{P(\alpha_n)}{x_r-\alpha_n} \end{bmatrix} \tag{1.12}$$

Proof. Obvious from Eq. (1.1).

1.11. Practical decoding of a generalized Goppa code:

1.11.1. The key equation of a generalized Goppa code:

A generalized Goppa code of length n with $deg\ G(X) = r$ is used. Let's assume that a codeword $C = (c_1, c_2, \ldots, c_n)$ was sent through a channel. The received word will be called $R = (r_1, r_2, \ldots, r_n)$. R is different from C when errors have occurred during the transmission. In order to simplify the notation and assuming that e errors occurred, the position of these errors are called:

$$X_1 = \alpha_{l_1}, X_2 = \alpha_{l_2}, \ldots, X_e = \alpha_{l_e} \tag{1.13}$$

and the error values denoted:

$$E_1, E_2, \ldots, E_e \tag{1.14}$$

If the channel is additive, it is obtained:

$$\begin{cases} R_i = C_i & \text{for } i \neq l_1, l_2, \ldots, l_e \\ R_i = C_i + E_i & \text{for } i = l_1, l_2, \ldots, l_e \end{cases} \tag{1.15}$$

To simplify the proof, the Alternant notation will be used, namely, $y_i = P(\alpha_i)G^{-1}(\alpha_i)$. Let's define the syndrome polynomial $S(X)$, the locator polynomial $\sigma(X)$ and the evaluator polynomial $\omega(X)$ by:

$$\begin{cases} S(X) = \sum_{i=0}^{r-1} (\sum_{j=1}^{n} y_j R_j \alpha_j^i) X^i \\ \sigma(X) = \prod_{i=1}^{e} (1 - X_i X) \\ \omega(X) = \sum_{i=1}^{e} y_{l_i} E_i \prod_{\substack{j=1 \\ j \neq i}}^{e} (1 - X_j X) \end{cases} \tag{1.16}$$

From Eq. (1.13), (1.14), (1.15), (1.16) and the fact the syndrome polynomial of any codeword is null (Eq. (1.9)), it is then deduced:

$$S(X) = \sum_{i=0}^{r-1} (\sum_{j=1}^{e} y_{l_j} E_j X_j^i) X^i$$

$$= \sum_{j=1}^{e} y_{l_j} E_j \sum_{i=0}^{r-1} (X_j X)^i$$

$$= \sum_{j=1}^{e} y_{l_j} E_j \frac{1 - (X_j X)^r}{1 - X_j X}$$

hence the following equation otherwise known as the key equation of an Alternant code (or generalized Goppa code) is derived:

$$\begin{cases} S(X)\sigma(X) \equiv \omega(X) \quad mod \quad X^r \\ deg\ \omega(X) < deg\ \sigma(X) = e \end{cases} \tag{1.17}$$

Berlekamp [10] has shown that Eq. (1.17) has a unique solution $(S(X)\ ,\ \omega(X))$ for a fixed r and $S(X)$, given that $e \leq r/2$.

Efficient methods for solving Eq. (1.17) are available; the Berlekamp algorithm, Berlekamp [10] and the MPR (minimal partial realization) Conan [19]. In the general case, MPR is the most easy to implement. In the particular case where $S(X)$ has the special property $S_{2i-1} = S_i^2$ for $i = 1\ ,\ 2\ ,\ \ldots\ ,\ r$ and $q = 2^m$ for some arbitrary positive integer m, the simplified version of Berlekamp's algorithm [10] requires about one half the computations than MPR and is thus recommended.

When the key equation is solved, it is necessary to find the roots of $\sigma(X)$ which correspond to the location of the errors. Then, using $\omega(X)$ and the roots of $\sigma(X)$, the error values can be obtained. It is important to keep in mind that $\sigma(X)$ has to divide $\pi(X)$ otherwise it is a non decoding situation.

In the binary case ($q = 2$), it is not necessary to find $\omega(X)$ since the location of an error is sufficient to correct the error (just add 1 to the corresponding received symbol).

1.11.2. MPR algorithm:

Define the following syndromes:

$$V_j = \sum_{i=1}^{n} R_i y_i \alpha_i^j$$

Initialization:

$$\sigma(X) = 1 \ , \ \omega(X) = 0 \ , \ b(X) = 0 \ , \ c(X) = -1 \ , \ d_p = 1$$

Iterative procedure: Do for $m = 0$ to $r-1$

$$d = \sum_{j=0}^{deg\ \sigma(X)} V_{m-j}\sigma_{w-j}$$

If $d \neq 0$ then $u = d_p - deg(\sigma(X))$

$$\begin{aligned}
&If\ u \leq 0 \ , \ \sigma(X) = \sigma(X) - d.X^{-u} b(X) \\
&\qquad \omega(X) = \omega(X) - d.X^{-u} c(X) \\
&\qquad d_p = d_p + 1 \ , \quad continue \\
&else \quad d_p = deg(\sigma(X)) \ , \ t_1(X) = \sigma(X) \ , \ t_2(X) = \omega(X) \\
&\sigma(X) = X^u \sigma(X) - d.b(X) \ , \ \omega(X) = X^u \omega(X) - d.c(X) \\
&b(X) = d^{-1}.t_1(X) \ , \ c(X) = d^{-1} t_2(X) \\
&d_p = d_p + 1 \ , \ continue
\end{aligned}$$

$else \ , \ d_p = d_p + 1 \ , \ continue$

1.11.3. Simplified Berlekamp algorithm:

Let's define $S_i = V_{i-1}$ for $i = 1 \ , \ 2 \ ,...,r$ and $S_0 = 1$. Assume that $S_{2i} = S_i^2$ and $q = 2^m$.

Initialization:

$$\sigma_0(X) = 1 \ , \ b(X) = 0$$

while $k < r/2$, do:

$$\begin{cases}
\sigma_{k+1}(X) = \sigma(X) + \Delta_k X b(X) \\
b(X) = X^2 b(X) \text{ if } \Delta_k = 0 \ Or \ deg\ \sigma_k(X) > k \\
b(X) = \Delta^{-1} X \sigma(X) \text{ if } \Delta_k \neq 0 \ Or \ deg\ \sigma_k(X) \leq k
\end{cases}$$

where Δ_k is defined by:

$$\Delta_k = \sum_{i=0}^{deg(\sigma(X))} \sigma_i \, S_{2k+1-i}$$

1.12. BCH codes:

Definition 1.6. For any given $G(X)$ of fixed degree r, a BCH code is defined by $P(X) = X^b G(X)$ and $\alpha_j = \alpha^j$ (α being an element of $GF(p^m)$ of order n where p is a prime number and the code having a length n).

If $n = p^m - 1$, the BCH code is said to be primitive otherwise, it is said non-primitive. If $b=1$, the corresponding BCH codes are called narrow sense, otherwise for $b > 1$, wide sense.

Proposition 1.11. A BCH code of length n has the following control matrix:

$$\begin{bmatrix} 1 & \alpha^b & \cdots & (\alpha^b)^{n-1} \\ 1 & \alpha^{b+1} & \cdots & (\alpha^{b+1})^{n-1} \\ \vdots & \vdots & \cdots & \vdots \\ 1 & \alpha^{b+r-1} & \cdots & (\alpha^{b+r-1})^{n-1} \end{bmatrix} \tag{1.18}$$

Proof. Obvious from Prop. (1.6) and Def. (1.6).

Proposition 1.12. For a BCH code of length n with coefficients over $GF(p)$ (p a prime number), the redundancy is:

$$n-k = deg \; lcm(M_{\alpha^b}(X), M_{\alpha^{b+1}}(X), \ldots, M_{\alpha^{b+r-1}}(X)) \tag{1.19}$$

Proof. Let's represent a BCH codeword with coefficients c_i by:

$$c(X) = \sum_{i=0}^{n-1} c_i X^i$$

then if C belongs to the kernel of Eq. (1.18):

$$C(\alpha^b) = C(\alpha^{b+1}) = \ldots = C(\alpha^{b+r-1}) = 0 \tag{1.20}$$

Since $C(X)$ ε $GF(p)[X]$, then from Prop. A.20. it is also equivalent to having the minimal polynomials of α^b , α^{b+1} , . . . , α^{b+r-1} divide $C(X)$. Defining:

$$E(X) = lcm(M_{\alpha^b}(X) , M_{\alpha^{b+1}}(X) , \ldots , M_{\alpha^{b+r-1}}(X))$$

and $n-k = deg\ E(X)$, then the encoding of k bits of information represented by $K(X) = \sum_{i=n-k}^{n-1} I_i X^i$ is done by the following Euclidian division:

$$K(X) = \delta(X)E(X)+R(X) \quad with\ deg\ R(X) < deg\ E(X) = n-k$$

the separable encoded codeword being $C(X) = K(X)-R(X)$, which satisfies by construction Eq. (1.20). *Q.E.D.*

Proposition 1.13. A narrow sense binary BCH code ($q = 2$) with $r = 2t$ has the following control matrix over $GF(2^m)$ with $d \geq 2t+1$ and $n-k \leq mt$ (Hamming codes correspond to $t = 1$):

$$\begin{bmatrix} 1 & \alpha & . & . & \alpha^{n-1} \\ 1 & \alpha^3 & . & . & (\alpha^3)^{n-1} \\ . & . & . & . & . \\ . & . & . & . & . \\ 1 & \alpha^{2t-1} & . & . & (\alpha^{2t-1})^{n-1} \end{bmatrix}$$

Proof. If b=1, then Eq. (1.19) is simplified, because from Prop. A.15, $M_\alpha(X) = M_{\alpha^2}(X)$, $M_{\alpha^3}(X) = M_{\alpha^6}(X)$ and so on. In other words:

$$n-k = deg\ lcm(M_\alpha(X) , M_{\alpha^3}(X) , \ldots , M_{\alpha^{2t-1}}(X)) \tag{1.20}$$

Clearly from section 1.11, this BCH can correct up to t errors when solving the key equation

since $r = 2t$ and $n-k \leq mt$ *Q.E.D.*

1.13. Goppa codes:

Definition 1.7. Let $P(X) = 1$ and $G(X)$ be a polynomial of degree r. The corresponding code is called a Goppa code(introduced by Goppa [17,18]) and satisfies from Eq. (1.1):

$$\sum_{i=1}^{n} \frac{a_i}{X-\alpha_i} \equiv 0 \ mod \ G(X) \tag{1.21}$$

Prop. 1.8 has already shown that the distance of a Goppa code ensures $d \geq r+1$.

Proposition 1.14. For a binary separable Goppa code ($G(X)$ is square free in some splitting field), $d \geq 2r+1$.

Proof. For each codeword of weight w, define the weight polynomial by:

$$\sigma(X) = \prod_{j=1}^{w}(X-\alpha_{(j)})$$

where (j) denotes the indice of the j^{th} non zero component of the codeword. Using the formal derivative on finite field (section A.8), it is clear since $a_i \in GF(2)$ that:

$$\sigma'(X) = \sigma(X)\sum_{j=1}^{w}\frac{1}{X-\alpha_{(j)}}$$

The codeword with components a_i belongs to the Goppa code defined by $G(X)$ if and only if:

$$\sum_{i=1}^{n} \frac{a_i}{X-\alpha_i} \equiv \sum_{j=1}^{w}\frac{1}{X-\alpha_{(j)}} \equiv \frac{\sigma'(X)}{\sigma(X)} \equiv 0 \ mod \ G(X) \tag{1.22}$$

Since $G(X)$ doesn't have any common roots with $\pi(X)$, $G(X)$ and $\sigma(X)$ are relatively prime and invoking Theorem A.1., there must exist two polynomials $A(X)$ and $B(X)$ of degree less

than $deg\ G(X)$ such that:

$$A(X)\sigma(X)+B(X)G(X) = 1 \tag{1.23}$$

In other words, Eq. (1.23) is equivalent to :

$$\sigma(X)A(X) \equiv 1\ mod\ G(X) \tag{1.24}$$

so $A(X)$ is the inverse of $\sigma(X)$ modulo $G(X)$. Rewriting Eq. (1.24) yields:

$$\sigma(X)\sigma'(X)A(X) \equiv \sigma'(X)\ mod\ G(X)$$

but from Eq. (1.22):

$$\sigma'(X)A(X) \equiv 0\ mod\ G(X)$$

It is then clear that:

$$\sigma'(X) \equiv 0\ mod\ G(X) \tag{1.25}$$

Practically, Eq. (1.25) implies that $G(X)$ divides $\sigma'(X)$. Since the derivative of any polynomial in a field of characteristic two is always a perfect square polynomial, the multiplicity order of the roots of $\sigma'(X)$ is even. Every root of $G(X)$ is a root of $\sigma'(X)$ but every root of $\sigma'(X)$ has an even order so the following polynomial also divides $\sigma'(X)$:

$$G^*(X) = G(X) \prod_{\substack{G(\gamma)=0 \\ \gamma\ odd\ order}} (X-\gamma)$$

Using degree considerations, it is clear that $w-1 \geq deg\ G^*(X)$, so:

$$d \geq deg\ G^*(X)+1 = r+1$$

If $G(X)$ is square free (or separable) in some splitting field (for example an irreducible polynomial) then $G^*(X) = G^2(X)$ which proves:

$$d \geq 2.deg\ G(X)+1 = 2r+1 \quad Q.E.D.$$

Of course, the binary separable Goppa codes are the most interesting since they guarantee a good distance. The most studied are the irreducible ones, and the Srivastava codes (Helgert [31], $G(X)$ is separable and split entirely in $GF(2^m)$, the locator field).

Proposition 1.15. Irreducible Goppa codes reach asymptotically the Varshamov-Gilbert bound.

Proof. See Goppa [17,18]

Proposition 1.16. The narrow sense BCH codes are Goppa codes and can be represented by $G(X) = X^r$.

Proof. Obvious from Eq. (1.9) and (1.18).

Proposition 1.17. If $G(X)$ is separable in some splitting field (denote $x_1, x_2, \ldots, x_r$ its r roots), then another form for the control matrix of the corresponding Goppa code is a Cauchy matrix, namely:

$$\begin{bmatrix} \frac{1}{x_1-\alpha_1} & \frac{1}{x_1-\alpha_2} & \cdots & \frac{1}{x_1-\alpha_n} \\ \frac{1}{x_2-\alpha_1} & \frac{1}{x_2-\alpha_2} & \cdots & \frac{1}{x_2-\alpha_n} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{1}{x_r-\alpha_1} & \frac{1}{x_r-\alpha_2} & \cdots & \frac{1}{x_r-\alpha_n} \end{bmatrix}$$

Proof. Obvious from Prop. 1.10.

Practically, in the binary case ($q = 2$), a separable polynomial $G(X)$ of degree t is chosen and the MPR algorithm is used with $G^*(X)$. This gives the same control matrix as the one defined by $G(X)$, but allows the correction of up to t errors algebraically with at most mt bits of redundancy whenever a $GF(2^m)$ locator field is used. For binary narrow

sense BCH codes, Prop. 1.12. indicates that the encoding can be done using an Euclidian division rather than a control matrix and can correct up to t errors with at most mt bits of redundancy whenever a $GF(2^m)$ locator field is used (Berlekamp algorithm is recommended for complexity reasons).

1.14. Examples of encoding:

1.14.1. BCH (15 , 5 , 7):

From Prop. 1.13., choosing $t = 3$, $m = 4$ and α a primitive element in $GF(16)$ of order 15 implies that the encoding polynomial is:

$$E(X) = lcm(M_{\alpha}(X) , M_{\alpha^3}(X) , M_{\alpha^5}(X))$$

This code can correct up to 3 errors by construction so $d \geq 7$ and has length $n = 15$. It can be verified from section A.11. that in $GF(16)$:

$$\begin{cases} M_{\alpha}(X) = X^4+X+1 \\ M_{\alpha^3}(X) = X^4+X^3+X^2+X+1 \\ M_{\alpha^5}(X) = X^2+X+1 \end{cases}$$

in other words:

$$E(X) = X^{10}+X^8+X^5+X^4+X^2+X+1$$

so $n-k = 5+5+2 = 10$, or equivalently $k = 5$.

See the proof of Prop. 1.12. for the encoding procedure from $E(X)$.

1.14.2. The (11 , 1 , 11) Goppa code:

Choosing $G(X) = X^5+X^4+X$ to define a primitive Goppa code defines the roots of $\pi(X)$ in a $GF(16)$ (see section A.11) to be $\{1,2,3,4,5,6,7,8,10,12,15\}$ or in exponential notation $\{1,\alpha,\alpha^4,\alpha^2,\alpha^8,\alpha^5,\alpha^{10},\alpha^3,\alpha^9,\alpha^6,\alpha^{12}\}$.

The code has length $n = 16 - deg\ G(X) = 11$ and from Prop. 1.14, $d_c = 2.deg\ G(X)+1 = 11$. Computing the control matrix from Eq. (1.9) and projecting over $GF(2)$ yields:

$$H = \left[\begin{array}{l} 10011011001 \\ 01111111111 \\ 01111111111 \\ 00000000000 \\ \hline 10010101101 \\ 00110110011 \\ 01010111010 \\ 01111001111 \\ \hline 11110110101 \\ 01001001001 \\ 00101000110 \\ 01111000000 \\ \hline 11001011010 \\ 00011111001 \\ 01100110110 \\ 00000000000 \\ \hline 10001101011 \\ 01001111100 \\ 00101110101 \\ 01111001111 \end{array}\right]$$

After performing an elimination of the linearly dependent rows of H over $GF(2)$ (some permutations of columns might be necessary in some cases when no pivot is found in the desired column), the separable form of H is obtained, namely:

$$H_s = \begin{bmatrix} 10000000001 \\ 01000000001 \\ 00100000001 \\ 00010000001 \\ 00001000001 \\ 00000100001 \\ 00000010001 \\ 00000001001 \\ 00000000101 \\ 00000000011 \end{bmatrix}$$

It is clear from section 1.7. that $k = 1$ (a repetition code) since 10 linear dependent rows were found.

CHAPTER 2

THE TRACE OPERATOR

2.1 Introduction:

Due to the importance of the trace operator in the next chapters, a review of the useful properties of such operator over a $GF(p^m)$ is presented in detail. An extension of these properties is proposed for the ring of residue classes over $GF(p^m)$ modulo $(X^{p^m}-X)$. It is assumed that the reader has knowledge of algebraic computation over $GF(p^m)$. Important properties about basis of $GF(p^m)$ over $GF(p)$ will be derived.

2.2 General properties:

Definition 2.1. The trace of an element $x \in GF(p^m)$ is defined by:

$$T_m(x) := \sum_{i=0}^{m-1} x^{p^i}$$

Definition 2.2. The restricted trace of order r $(r \leq m)$ of an element $x \in GF(p^m)$ is defined by:

$$T_r(x) := \sum_{i=0}^{r-1} x^{p^i}$$

where all the computations are done in $GF(p^m)$.

Proposition 2.1. $T_m(x^p) = T_m(x) = T_m^p(x)$

Proof. Applying Def. 2.1 and noting that $x^{p^m} = x$ (since $x \in GF(p^m)$), it follows:

$$T_m(x^p) = \sum_{i=0}^{m-1} (x^p)^{p^i} = \sum_{i=0}^{m-1} x^{p^{i+1}} = x^{p^m} + \sum_{i=1}^{m-1} x^{p^i} = T_m(x)$$

$$T_m^p(x) = (\sum_{i=0}^{m-1} x^{p^i})^p = \sum_{i=0}^{m-1} (x^{p^i})^p = \sum_{i=0}^{m-1} x^{p^{i+1}} = T_m(x) \quad Q.E.D.$$

Proposition 2.2. $T_m(x) \in GF(p)$

Proof. Since $T_m(x) \in GF(p^m)$ and $GF(p)$ is imbedded from Prop. A.21. in $GF(p^m)$, Prop. 2.1. indicates that $T_m^p(x) - T_m(x) = 0$ or in other words, $T_m(x) \in GF(p)$. *Q.E.D.*

Proposition 2.3. The trace operator is linear over $GF(p)$, namely, for any $x, y \in GF(p^m)$ and $\lambda \in GF(p)$:

$$\begin{cases} T_m(x+y) = T_m(x) + T_m(y) \\ T_m(\lambda x) = \lambda T_m(x) \end{cases}$$

Proof. Using Prop. A.12., it follows that

$$T_m(x+y) = \sum_{i=0}^{m-1} (x+y)^{p^i} = \sum_{i=0}^{m-1} x^{p^i} + y^{p^i} = T_m(x) + T_m(y)$$

Since $\lambda \in GF(p)$, then $\lambda^p = \lambda$ and:

$$T_m(\lambda x) = \sum_{i=0}^{m-1} (\lambda x)^{p^i} = \sum_{i=0}^{m-1} \lambda^{p^i} x^{p^i} = \sum_{i=0}^{m-1} \lambda x^{p^i} = \lambda T_m(x) \quad Q.E.D.$$

Proposition 2.4. $T_m(x)$ is not identically equal to zero, namely there exists at least one element $x \in GF(p^m)$ such that $T_m(x) \neq 0$

Proof. Since $T_m(X)$ is a polynomial of degree p^{m-1}, it has at most p^{m-1} roots in $GF(p^m)$ so it remains at least one element of $GF(p^m)$ with a non null trace. *Q.E.D.*

Proposition 2.5. The trace operator is uniformly distributed, namely, $|\{x \mid T_m(x) = i\}| = p^{m-1}$ for any $i \in GF(p)$.

Proof. Let's define $A_i = \{ x \mid T_m(x) = i \ , x \in GF(p^m) \}$. It is clear from Prop. 2.3. that A_0 is a vector space over $GF(p)$ so $|A_0| = p^j$ for some positive integer j. Also from Prop. 2.4., there is an element $\alpha \in GF(p^m)$ such that $T_m(\alpha) = k$ with $k \in GF(p)$-{0}. For any $x \in A_0$, $T_m(\alpha+x) = k$. Supposing there exists $y \in GF(p^m)$ such that $T_m(y) = k$ and $y \neq \alpha+x$ for any $x \in A_0$, then $y = \alpha+x+\gamma$. In other words $T_m(\gamma) = 0$, which is a contradiction. So it may be concluded that $|A_k| = p^j$.

The following sequence $y_i = \sum_{l=1}^{i} y = y \sum_{l=1}^{i} 1$ for $i = 1 , \cdots , p$ generates respectively one element of each A_i's. Using the same argument as before, it can be derived that $|A_i| = p^j$ for $i = 0 , \ldots , p-1$. Since the A_i's are disjoint, then:

$$p^m = \sum_{i=0}^{m-1} |A_i| = pp^j = p^{j+1}$$

hence $j = m-1$. *Q.E.D.*

Proposition 2.6. If X denotes a polynomial indeterminate variable, then:

$$T_m(X)-s = \prod_{T_m(\beta)=s} (X-\beta)$$

Proof. Since the polynomial $T_m(X) - s$ has degree p^{m-1}, this polynomial must have p^{m-1} roots in some splitting field. It is clear that if $T_m(\beta) = s$, β is a root and from Prop. 2.5, there are exactly p^{m-1} distincts β's so the degrees match. *Q.E.D.*

2.3 Polynomial extension

It is possible to keep the previous properties of the trace operator when extended to rational polynomials with coefficients over $GF(p^m)$ given that the trace operator is defined in residue classes *mod* $(X^{p^m}-X)$. It can be noted that the ring of these residue classes is not an integral domain since $X^{p^m}-X$ is not irreducible over $GF(p^m)[\,X\,]$. When not specified otherwise, the symbol $\equiv$ indicates that the computations are done *mod* $(X^{p^m}-X)$.

Definition 2.3. The ring of fractional polynomials $Q(p^m)[X]$ is defined by:

$$Q(p^m)[X] = \left\{ \frac{f(X)}{h(X)} \mid f(X)\,,\, h(X) \in GF(p^m)[\,X\,]\,,\, h(X) \neq 0 \right\}$$

Definition 2.4. For any $g(X) \in Q(p^m)[X]$, the trace of $g(X)$ in the residue class *mod* $(X^{p^m}-X)$ is then defined by:

$$T_m(g(X)) :\equiv \sum_{i=0}^{m-1} g^{p^i}(X)$$

Definition 2.5. For any $g(X) \in Q(p^m)[X]$, the restricted trace of order r $(r \leq m)$ of $g(X)$ in the residue class *mod* $(X^{p^m}-X)$ is then defined by:

$$T_r(g(X)) :\equiv \sum_{i=0}^{r-1} g^{p^i}(X)$$

Proposition 2.7. For any $g(X) \in Q(p^m)[X]$, $g^{p^m}(X) \equiv g(X)$

Proof. It is enough to show this result for $f(X)$ or either $h(X) \in GF(p^m)[\,X\,]$ (since $g(X) = \frac{f(X)}{h(X)}$). Using Theorem A.2. on the coefficients of $f(X)$, it is shown:

$$f^{\,p^m}(X) \equiv \Big(\sum_{i=0}^{deg\, f(X)} f_i X^i\Big)^{p^m} \equiv \sum_{i=0}^{deg\, f(X)} f_i^{p^m} (X^{p^m})^i \equiv \sum_{i=0}^{deg\, f(X)} f_i X^i \equiv f(X) \quad Q.E.D.$$

It is worthwhile noting that $T_m^p(g(x)) = T_m(g^p(X))$.

Proposition 2.8. $T_m(g^p(X)) \equiv T_m(g(X)) \equiv T_m^p(g(X))$

Proof. Applying the definition and Prop. 2.7,

$$T_m(g^p(X)) \equiv \sum_{i=0}^{m-1}(g^p(X))^{p^i} \equiv \sum_{i=0}^{m-1} g^{p^{i+1}}(X) \equiv g^{p^m}(X) + \sum_{i=1}^{m-1} g^{p^i}(X) \equiv T_m(g(X))$$

$$T_m^p(g(X)) \equiv (\sum_{i=0}^{m-1} g^{p^i}(X))^p \equiv \sum_{i=0}^{m-1}(g^{p^i}(X))^p \equiv \sum_{i=0}^{m-1} g^{p^{i+1}}(X) \equiv T_m(g(x)) \quad Q.E.D.$$

Proposition 2.9. if $g^{p^r}(X) \equiv g(X)$ with $r \leq m$ then $T_r^p(g(X)) \equiv T_r(g(X))$

Proof. Using Def. 2.5 and the hypothesis:

$$T_r^p(g(X)) \equiv \sum_{i=0}^{r-1} g^{p^{i+1}}(X) \equiv g^{p^r}(X) + \sum_{i=1}^{r-1} g^{p^i}(X)$$

$$\equiv g(X) + \sum_{i=1}^{r-1} g^{p^i}(X) \equiv T_r(g(X)) \quad Q.E.D.$$

2.4 Special case $m = 2s$:

Proposition 2.10. If $T_s(x) = 0$ for $x \in GF(p^{2s})$ then $x \in GF(p^s)$

Proof.

$$T_s^p(x) - T_s(x) = (\sum_{i=0}^{s-1} x^{p^i})^p - \sum_{i=0}^{s-1} x^{p^i}$$

$$= \sum_{i=1}^{s-1} x^{p^i} + x^{p^s} - x - \sum_{i=1}^{s-1} x^{p^i}$$

$$= x^{p^s} - x$$

Since $T_s(x) = 0$, this implies $x^{p^s} - x = 0$, so in other words, $x \in GF(p^s)$ (it is necessary to

recall that $GF(p^s)$ is embedded in $GF(p^{2s})$ from Prop. A.21. and $T_s(x) \in GF(p^{2s})$) *Q.E.D.*

Proposition 2.11. $T_{2s}(g(X)) = T_s(g(X)) + T_s^{p^s}(g(X)) = T_s(g(X) + g^{p^s}(X))$.

Proof.

$$T_{2s}(g(X)) = \sum_{i=0}^{2s-1} g^{p^i}(X)$$

$$= \sum_{i=0}^{s-1} g^{p^i}(X) + (\sum_{i=0}^{s-1} g^{p^i}(X))^{p^s}$$

$$= T_s(g(X)) + T_s^{p^s}(g(X))$$

$$= T_s(g(X) + g^{p^s}(X)) \quad Q.E.D.$$

2.5 Basis of $GF(p^m)$ and trace operator:

From Prop. A.11., $GF(p^m)$ is a vector space over $GF(p)$ of dimension m. Let $\beta_1, \beta_2, \ldots, \beta_m$ be one possible basis.

Proposition 2.12. There always exits a complementary basis $\lambda_1, \ldots, \lambda_m$ of the basis $\beta_1, \ldots, \beta_m$ such that:

$$T_m(\lambda_i \beta_j) = \begin{cases} 0 \text{ if } i \neq j \\ 1 \text{ if } i = j \end{cases} \quad \text{for} \quad 1 \leq i, j \leq m$$

Proof. In order to simplify the proof, the tensor notation will be used. Define the matrix A by:

$$A = (T_m(\beta_i \beta_j))_{ij} \quad 1 \leq i, j \leq m$$

Clearly, from Prop. 2.2, A has all its coefficients on $GF(p)$. Suppose that A is not invertible

over the set of matrices with coefficients belonging to $GF(p)$, then there must exits a vector b with coefficients b_i's over $GF(p)$ such that $Ab = 0$. By isomorphism, b can be seen as an element of $GF(p^m)$, namely:

$$\lambda = \sum_{i=1}^{m} b_i \beta_i$$

and $Ab = 0$ becoming equivalent to $T_m(\beta_i \lambda) = 0$ for $i = 1, 2, \ldots, m$. Due to Prop. 2.3, it is then derived that $T_m(x\lambda) = 0$ for any $x \in GF(p^m)$. Prop. 2.4 shows that the trace operator is not the null operator, it is then concluded that $b = 0$ so A has an inverse.

Take the matrix $B = (b_{jk})_{jk}$ for $1 \leq j, k \leq m$, then it is clear that:

$$AB = \left(T_m(\beta_i (\sum_{j=1}^{m} b_{jk} \beta_j)) \right)_{ik}$$

Defining the following set of elements of $GF(p^m)$ by:

$$\lambda_k = \sum_{j=1}^{m} b_{jk} \beta j$$

induces that:

$$A.B = (T_m(\beta_i \lambda_k))_{ik}$$

Since A has an inverse and taking $B = A^{-1}$ proves that $\lambda_1, \lambda_2, \ldots, \lambda_m$ is a complementary basis of $\beta_1, \beta_2, \ldots, \beta_m$ since $A.B = (\delta_{ik})_{ik}$ (δ being the Kronecker function). *Q.E.D.*

Proposition 2.13. Let $a, b \in GF(p^m)$ and $a = \sum_{i=1}^{m} a_i \beta_i$ and $b = \sum_{i=1}^{m} b_i \lambda_i$ with $a_i, b_i \in GF(p)$, then:

$$T_m(ab) = \sum_{i=1}^{m} a_i b_i$$

Proof. Using the distributive law on the product ab, the fact that β_i's and λ_i's form two complementary basis completes the proof. *Q.E.D.*

Proposition 2.14. For any $a \in GF(p^m)$ then:

$$a = \sum_{i=1}^{m} T_m(a\lambda_i)\beta_i$$

Proof. Follows immediately from Prop. 2.13. *Q.E.D.*

Proposition 2.15. It is possible to make any linear combination of the $a_i's$ over $GF(p)$ by selecting the appropriate b and taking the trace of ab.

Proof. From Prop. 2.13., it is enough to select the b_i's in $GF(p)$ to get the desired linear combination of the a_i's. Using then Prop. 2.14., b is uniquely constructed. *Q.E.D.*

CHAPTER 3

THE TRACE OPERATOR AND GENERALIZED GOPPA CODES

3.1 Introduction:

It will be shown that Prop. 2.15. can be used to derive a new analytical approach to the determination of the dimension of the generalized Goppa codes introduced by Loeloeian and Conan [29]. This family contains in particular all the Alternant codes, Goppa codes, Srivastava codes, BCH codes and Hamming codes.

Some original bounds for specific Goppa codes will be derived with this analytical approach without the need of a computer. It can be noted that it is not surprising to have the trace operator related to the dimension of a linear block, in particular Delsarte [32] proved a general result involving the dimension of a subfield code, its orthogonal code and the trace operator.

3.2 The redundancy equation of a generalized Goppa code:

Let $G(X)$ be a fixed polynomial with coefficients over $GF(p^m)$ with degree t, $P(X)$ another polynomial of degree s, and $\pi(X)$ a separable polynomial of degree n that splits entirely in $GF(p^m)$ such that $gcd(\,G(X)\,,\,\pi(X))=1$ and $gcd(\,P(X)\,,\,\pi(X))=1$.

If the roots of $\pi(X)$ are $\alpha_1, \ldots, \alpha_n$, then from Chapter 1, the generalized Goppa code $\Gamma(\pi(X), G(X), P(X))$ of length n and constructive distance (or designed distance) $d_c = t+1$ has the following control matrix H:

$$H_{ij} = \alpha_j^i P(\alpha_j) G^{-1}(\alpha_j) \quad , 1 \le j \le n, \quad 0 \le i < t \tag{3.1}$$

When dealing with the corresponding generalized Goppa code, only the n-tuples with coefficient over $GF(p)$ belonging to the kernel of H are kept. It has been shown that it is equivalent to project H over $GF(p)$ as a vector space which yields a matrix having mt rows with coefficients over $GF(p)$ instead of m rows over $GF(p^m)$.

In general, if it is possible to find one linear dependent row in the projected matrix H over $GF(p)$, Prop. 2.15 shows the existence of a polynomial $A(X)$ such that:

$$\begin{cases} A(X) = \sum_{i=0}^{e} A_i X^i \ , \ e = deg\ A(X) < deg\ G(X) = t \ , \ A_i \in GF(p^m) \\ \sum_{k=0}^{e} A_k \alpha_j^k P(\alpha_j) G^{-1}(\alpha_j) = \gamma_j \ \ , T_m(\gamma_j) = 0 \ \ , \ \ 1 \le j \le n \end{cases} \tag{3.2}$$

The coefficients A_i correspond to the linear combination of the m rows defined by H_{ij} for a fixed j. Since it might be necessary to use all the terms for $0 \le i < t$ to really eliminate one possible dependent row over $GF(p)$, this explain why $deg\ A(X) < t$. Of course, the same linear combination has to succeed on all the columns of H. This combination succeeds if and only if the trace of the corresponding γ_j is equal to zero, but it is not necessary to have all the γ_j's equal to a same element.

From Prop. 2.5., there are exactly p^{m-1} elements γ of $GF(p^m)$ having $T_m(\gamma) = 0$. One possible way to have a polynomial interpretation of Eq. 3.2 is for any fixed γ having a null trace, find all the α_j satisfying:

$$\sum_{k=0}^{e} A_k \alpha_j^k P(\alpha_j) G^{-1}(\alpha_j) = \gamma$$

This is equivalent to finding the solutions of $A(X)P(X)-\gamma G(X) = 0$ in $GF(p^m)$. It is then hoped that solving the p^{m-1} corresponding equations will yield all the roots of $\pi(X)$, implying that the linear combination uniquely defined by the choice of the coefficients of $A(X)$ succeeded on all the columns of H. A necessary and sufficient condition for eliminating one of the dependent rows is then:

$$\prod_{T_m(\gamma)=0} (A(X)P(X)-\gamma G(X)) = \lambda(X)\pi(X) \tag{3.3}$$

which implies that the linear combination conditioned by $A(X)$ succeeded on the n columns of H because of the divisibility by $\pi(X)$ in Eq. (3.3).

Prop. 2.6 (for $s = 0$) and Eq. (3.3) yield:

$$\begin{cases} G^{p^{m-1}}(X) T_m \left[A(X)P(X)G^{-1}(X) \right] \equiv 0 \ \ mod \ (\pi (X)) \\ A(X) = \sum_{i=0}^{e} A_i X^i \ , \quad e = deg \ A(X) < deg \ G(X) = t \ , \ A_i \ \varepsilon \ GF(p^m) \end{cases} \tag{3.4}$$

It is worthwhile noting at this point that $G^{p^{m-1}}(X) T_m \left[A(X)P(X)G^{-1}(X) \right]$ is a polynomial over $GF(p^m)[X]$ (despite its fractional appearance).

Definition 3.1. Eq. (3.4) will be refered as the Redundancy Equation of a generalized Goppa Code and, $S(G(X),P(X),\pi(X))$ the set of all the solutions $A(X)$ satisfying Eq. (3.4).

3.3 Interpretation:

Proposition 3.1. $S(G(X), P(X), \pi(X))$ forms a vector space over $GF(p)$ given the following rules:

If $A_1(X)$, $A_2(X)$ ε $S(G(X)$, $P(X)$, $\pi(X))$ and λ ε $GF(p)$ then:

$$\begin{cases} A_1(X)+A_2(X) \ \varepsilon \ S(G(X) \ , \ P(X) \ , \ \pi(X)) \\ \lambda A_1(X) \ \varepsilon \ S(G(X) \ , \ P(X) \ , \ \pi(X)) \end{cases} \tag{3.5}$$

Proof. Clearly, proving Eq. (3.5) shows that $S(G(X)$, $P(X)$, $\pi(X))$ is a vector space over $GF(p)$.

From Def. 3.1., $A_1(X)$ and $A_2(X)$ ε $S(G(X)$, $P(X)$, $\pi(X))$ is equivalent to saying that *deg* $A_1(X)$ and *deg* $A_2(X)$ is less that *deg* $G(X)$ and:

$$\begin{cases} G^{p^{m-1}}(X)T_m \ [\ \dfrac{A_1(X)P(X)}{G(X)} \ ] \ \equiv 0 \ \ mod \ (\ \pi \ (X) \ ) \\ G^{p^{m-1}}(X)T_m \ [\dfrac{A_2(X)P(X)}{G(X)} \ ] \ \equiv 0 \ \ mod \ (\ \pi \ (X) \ ) \end{cases} \tag{3.6}$$

Combining the two equations of Eq. (3.6) with the residue classes properties and linear properties of the trace operator, yields:

$$G^{p^{m-1}}(X)T_m \ [\ \frac{(A_1(X)+A_2(X))P(X)}{G(X)}] \equiv G^{p^{m-1}}(X)T_m \ [\ \frac{A_1(X)P(X)}{G(X)} \ ]$$

$$+G^{p^{m-1}}(X)T_m \ [\ \frac{A_2(X)P(X)}{G(X)} \ ]$$

$$\equiv 0 \ \ mod \ (\ \pi \ (X) \ )$$

so $A_1(X)+A_2(X)$ ε $S(G(X)$, $P(X)$, $\pi(X))$.

Using the same ideas for λ ε $GF(p)$:

$$G^{p^{m-1}}(X)T_m \ [\ \frac{(\lambda A_1(X))P(X)}{G(X)} \ ] \equiv G^{p^{m-1}}(X)\lambda T_m \ [\ \frac{A_1(X)P(X)}{G(X)} \ ]$$

$$\equiv 0 \ \ mod \ (\ \pi \ (X) \ )$$

so $\lambda A_1(X) \in S(G(X), P(X), \pi(X))$.

Finally, it is clear that:

$$\begin{cases} deg\ A_1(X)+A_2(X) < deg\ G(X) \\ deg\ \lambda A_1(X) < deg\ G(X) \end{cases} \qquad Q.E.D.$$

Proposition 3.2. The redundancy $n-k$ of the generalized Goppa code satisfies:

$$n-k = m.deg\ G(X) - dim\ S(G(X),P(X),\pi(X)) \tag{3.7}$$

Proof. From Prop. 3.1., the dimension of $S(G(X),P(X),\pi(X))$ over $GF(p)$ is well defined. The control matrix of the $\Gamma(\pi(X), G(X), P(X))$ when projected over $GF(p)$ has exactly $m.deg\ G(X)$ rows. The dimension of $S(G(X),P(X),\pi(X))$ is the number of dependent non-null rows of the control matrix out of the mt initial rows and the number of remaining independent rows is $n-k$ from Prop. 1.3. *Q.E.D.*

3.4 Linear mapping:

Proposition 3.3. $dim\ S(G(X),P(X),\pi(X)) = dim\ S(G(aX+b),P(aX+b),\pi(aX+b))$ for any $a \in GF(p^m)$-{0} and $b \in GF(p^m)$.

Proof. Since the transformation $X \rightarrow aX+b$ $(a \neq 0)$ doesn't change the degrees of $A(X)$, $G(X)$ and $\pi(X)$, the solutions of $S(G(X),P(X),\pi(X))$ are mapped isormophically into $S(G(aX+b),P(aX+b),\pi(aX+b))$. *Q.E.D.*

Proposition 3.4. It is enough to study monic generalized Goppa codes.

Proof. It is clear that $S(G(X),P(X),\pi(X))$ is isomorphic to $S(a^{-1}G(X),b^{-1}P(X),\pi(X))$ for any $a, b \in GF(p^m)$-{0}. Taking a equal to the highest order coefficient of $G(X)$ and b equal to the highest order coefficient of $P(X)$ completes the proof. *Q.E.D.*

3.5 Sub classes of the generalized Goppa family:

It has been shown in Chapter 1 how the family of Alternant codes is equivalent to the family of generalized Goppa codes. For the moment, the only well known Alternant codes are the Goppa codes and the BCH codes so, additional results using the trace operator are derived for these particular codes.

3.5.1. Wide sense BCH codes:

A simpler form of the redundancy equation can be obtained for the narrow sense BCH codes. From Prop. 1.11., another form for the control matrix of BCH codes is:

$$H_{ij} = \alpha^{(b+i)j} \quad \text{for } 1 \leq j \leq n \quad 0 \leq i \leq r-1 \tag{3.8}$$

so using the general approach developed in section 3.2 leads to the definition of the following polynomial for the redundancy equation:

$$\begin{cases} B(X) = X^b \sum_{i=0}^{e} B_i X^i \quad , B_i \ \varepsilon \ GF(p^m) \quad , \ e < r \\ T_m(B(X)) \equiv 0 \ \ mod \ (X^{n-1}-1) \end{cases}$$

n being the length of the code or in other words, the order of α.

3.5.2. Binary narrow sense BCH codes:

In the particular case of binary narrow sense BCH codes (b = 1), Prop. 1.13. indicates an even simpler form for the redundancy equation leading to:

$$\begin{cases} C(X) = \sum_{i=1}^{e} C_i X^{2i-1} \quad , \ C_i \ \varepsilon \ GF(2^m) \quad , \ e \leq t \\ T_m(C(X)) \equiv 0 \ \ mod \ (X^{n-1}-1) \end{cases} \tag{3.9}$$

n being the length of the code or in other words, the order of α.

3.5.3. Goppa codes:

Def. 1.7. says that Goppa codes correspond to all the cases where $P(X) = 1$, Eq. (3.4) becomes then:

$$\begin{cases} G^{p^{m-1}}(X)T_m\ [\ A(X)G^{-1}(X)\ ] \equiv 0 \ \ mod\ (\ \pi\ (X)\ ) \\ A(X) = \sum_{i=0}^{e} A_i X^i\ ,\quad e = deg\ A(X) < deg\ G(X) = t\ ,\ A_i\ \varepsilon\ GF(p^m) \end{cases} \tag{3.10}$$

3.6. Primitive Goppa codes:

The binary Goppa codes that have been most studied previously are the primitive ones. This require that $(X^{p^m}-X)$ divides $G(X)\pi(X)$. Since $G(X)$ might not split entirely in $GF(p^m)$, multiplying both sides of Eq. (3.10) yields:

$$\begin{cases} G(X)G^{p^{m-1}}(X)T_m\ [\ A(X)G^{-1}(X)\ ] \equiv 0 \ \ mod\ (\ X^{p^m}-X) \\ A(X) = \sum_{i=0}^{e} A_i X^i\ ,\quad e = deg\ A(X) < deg\ G(X) = t\ ,\ A_i\ \varepsilon\ GF(p^m) \end{cases} \tag{3.11}$$

It is worthwile noting that this multiplication does not increase the number of solutions as $(\pi(X)\ ,\ G(X)) = 1$.

Definition 3.2. $R_m(G(X)) = S(G(X)\ ,\ P(X)\ ,\ \pi(X))$ where $P(X) = 1$ and $(X^{p^m}-X)$ divides $G(X)\pi(X)$.

3.7 Unification and case $p = 2$:

Using Prop. 2.11. when $m = 2s$ and $p = 2$, Eq. (3.11) can be simplified and becomes:

$$G(X)G^{2^{2s-1}}(X)T_s \left[\frac{A(X)}{G(X)} + \frac{A^{2^s}(X)}{G^{2^s}(X)} \right] \equiv 0 \ \ mod \ (X^{2^{2s}} + X) \tag{3.12}$$

Assuming that:

$$G^{2^s}(X) \equiv G(X) \ mod \ (X^{2^{2s}} + X) \tag{3.13}$$

Eq. (3.12) becomes:

$$G(X)G^{2^{s-1}}(X)T_s \left[\frac{A(X)+A^{2^s}(X)}{G(X)} \right] \equiv 0 \ \ mod \ (X^{2^{2s}} + X) \tag{3.14}$$

Proposition 3.5. If $G(X)$ satisfies Eq. (3.13), then:

$$n-k \leq 2s.deg \ G(X) - s$$

Proof. Take $A(X) = A_0$ such that $A_0^{2^s} + A_0 = 0$, then $A(X)$ is always a solution of Eq. (3.14). There are exactly 2^s distinct solutions A_0 in $GF(2^{2s})$ from Prop. A.23. It is also clear that these particular solutions form a vector space over $GF(2)$ of dimension s which is contained in $R_{2s}(G(X))$. Invoking Prop. 3.2. completes the proof. *Q.E.D.*

One might ask if there is any solutions $A(X)$ with $deg \ A(X) > 0$. The following proposition gives some more insight in the matter.

Proposition 3.6. If $G(X)$ satisfies Eq. (3.13) and $deg \ G(X) = 2^s + 1$, then:

$$n-k \leq s\,2^{s+1} - s$$

Proof. Since $deg \ G(X) = 2^s + 1$, all the possible solutions $A(X)$ require by definition that $deg \ A(X) \leq 2^s$. Let's compute the following equation:

$$A(X)+A^{2^s}(X) \equiv A_0+A_1X+A_2X^2+\ldots+A_{2^s}X^{2^s}$$

$$+A_0^{2^s}+A_1^{2^s}X^{2^s}+A_2^{2^s}X^{2.2^s}+\ldots+A_{2^s}^{2^s}X^{2^s 2^s}$$

$$\equiv (A_0+A_0^{2^s})+(A_1+A_{2^s}^{2^s})X+A_2X^2+\ldots+A_{2^s-1}X^{2^s-1}$$

$$+(A_{2^s}+A_1^{2^s})X^{2^s}+A_2^{2^s}X^{2.2^s}+\ldots+A_{2^s-1}^{2^s}X^{2^s(2^s-1)} \quad mod\ (X^{2^{2s}}+X)$$

Clearly, the only way $A(X)+A^{2^s}(X) \equiv 0$ is by having:

$$\begin{cases} A_0+A_0^{2^s} = 0 \\ A_1+A_{2^s}^{2^s} = 0 \\ A_1^{2^s}+A_{2^s} = 0 \\ A_2 = A_2 = \cdots = A_{2^s-1} = 0 \end{cases} \tag{3.15}$$

Eq. (3.15) has from Prop. A.23. 2^s distinct solutions A_0. It is also possible to choose independently of A_0 any $A_1 \in GF(2^{2s})$, which then uniquely determines A_{2^s}. It can be noted from Prop. A.22. that $A_1+A_{2^s}^{2^s} = 0$ is equivalent to $A_1^{2^s}+A_{2^s} = 0$ in $GF(2^{2s})$.

Overall, there are $2^s 2^{2s} = 2^{3s}$ distinct solutions. These solutions also form a sub-vector space over $GF(2)$ of $R_{2s}(G(X))$ of dimension $3s$ so by Prop 3.2:

$$n-k \leq 2s(2^s+1)-3s = s2^{s+1}-s \qquad Q.E.D.$$

It is worthwhile noting that Prop. 3.6 provides a tighter bound than Prop. 3.5.

CHAPTER 4

THE TRACE OPERATOR AND LOELOEIAN CODES

4.1. Introduction:

Loeloeian and Conan [26] introduced a family of Goppa codes defined by $G_1(X) = X^{2^s}+X$. One possible generalization of the Goppa code found by Bezzateev and Shekhunova [27] could be the family of Goppa codes defined by $G_2(X) = X^{2^{s+1}}+1$ (the case $s=3$ was only considered by these authors and corresponds to a (55,16,19) code). Both of these families require a $GF(2^{2s})$ as a locator field and also satisfy Eq. (3.13) providing a nice unification. Furthermore, the codes derived are primitive with in particular; $G_1(X)\pi_1(X) = X^{2^{2s}}+X$ and $G_2(X)\pi_2(X) = X^{2^{2s}}+X$.

Since the (55,16,19)-code is for the moment the best binary linear block code known for $n=55$ and $d_c=19$ (Veorheff [28]), codes defined by $G_2(X)$ might be very interesting to study especially for $s>3$. Loeloeian and Conan [25,26] have also shown that for spectral considerations, codes defined by $G_1(X)$ are closely related to the ones defined by $G_2(X)$.

Tighter bounds than the one given by Prop. 3.5 and Prop. 3.6. for these specific codes in the general case $s>1$ will be obtained by partially solving Eq. (3.14).

4.2. Simulation:

Using the results of computer simulation, the cases s=2,3,4,5 can be summarized in the following tables where n is the length of the code, k its dimension, d_G the Goppa distance bound derived from Prop. 1.14., d_L the Loeloeian distance bound (Loeloeian and Conan [26]). An upper bound d_S on the distance is derived from the actual coefficients of the corresponding parity check matrix; namely by forcing respectively every information bit to zero except for one and computing the weight of the encoded codeword (this is done k times and d_S corresponds to the smallest weight obtained).

Table 1

$G_1(X) = X^{2^s}+X$						
s	n	d_G	d_L	d_S	k	$dim\ R_{2s}(G_1(X))$
2	12	9	12	12	1	5
3	56	17	20	20	16	8
4	240	33	36	42	123	11
5	992	65	68	118	686	14

Table 2

$G_2(X) = X^{2^s+1}+1$						
s	n	d_G	d_L	d_S	k	$dim\ R_{2s}(G_2(X))$
2	11	11	11	11	1	10
3	55	19	19	19	16	15
4	239	35	35	40	123	20
5	991	67	67	118	686	25

On a first approach, it looks like that $dim\ R_{2s}(G_1(X)) = 3s-1$ and $dim\ R_{2s}(G_2(X)) = 5s$ for s=2,3,4,5. For computational reasons, such results cannot be proved using a computer for large values of s. It is hoped that the redundancy equation will provide bounds on the redundancy of these two families of Goppa codes.

It has been shown previously that the true minimum distance d of a Goppa code verifies $d_G \leq d_L \leq d \leq d_S$. Table 1 indicates that $d_L = d_G+3$ for $G_1(X)$.

4.3. Important remarks:

Remark 4.1. When not specified otherwise, the symbol ≡ indicates from now on that the computations are done "*mod* $(X^{2^{2s}}+X)$".

Remark 4.2: $G_1^{2^s}(X) \equiv G_1(X)$ and $G_2^{2^s}(X) \equiv G_2(X)$

Proof. Applying the definition of $G_1(X)$ and $G_2(X)$ yields:

$$G_1^{2^s}(X) \equiv (X^{2^s}+X)^{2^s} \equiv X^{2^{2s}}+X^{2^s} \equiv X+X^{2^s} \equiv G_1(X)$$

$$G_2^{2^s}(X) \equiv (X^{2^s+1}+1)^{2^s} \equiv X^{2^{2s}+2^s}+1 \equiv X^{1+2^s}+1 \equiv G_2(X) \quad Q.E.D.$$

Remark 4.3. $G_1(X)\pi_1(X) = X^{2^{2s}}+X$ and $G_2(X)\pi_2(X) = X^{2^{2s}}+X$. In other words, these codes are primitive.

Proof. The existence of $\pi_1(X)$ is clear using Prop. A.23. For $G_2(X)$, pick a primitive element $\alpha \in GF(2^{2s})$, then all the roots of $G_2(X)$ in $GF(2^{2s})$ are $\alpha^{i(2^s-1)}$ for $i = 0, 1, \ldots, 2^s$. Since $G_2(X)$ splits entirely in $GF(2^{2s})$, $\pi_2(X)$ exists. *Q.E.D.*

Remark 4.4. *dim* $R_{2s}(G_1(X)) \geq s$ and *dim* $R_{2s}(G_2(X)) \geq 3s$

Proof. From Prop. 3.5., 3.6. remark 4.2 and Remark 4.3. *Q.E.D.*

4.4. Study of the case s=2:

Remark (4.4) yields bounds on the dimension of the redundancy vector space which are still too far from the one expected in Table 1 and Table 2. Before attempting an extensive analytical approach, it is interesting to derive the set of equations that have to be simultaneously solved when dealing with Eq. (3.14) where s=2 and for respectively, $G_1(X)$ and $G_2(X)$. This will help to find a heuristic solution for better bounds. In this particular case, all the residue computations are done *mod* $(X^{16}+X)$. Attempting here to solve Eq. (3.14) for $s = 2$ yields:

$$G(X)G^2(X)T_2[\frac{A(X)+A^4(X)}{G(X)}] \equiv G^2(X)[A(X)+A^4(X)]+G(X)[A(X)+A^4(X)]^2 \quad (4.1)$$

4.4.1. Loeloeian codes:

Since $G(X) = G_1(X) = X^4+X$ and $deg\ A(X) < deg\ G(X) = 4$ then:

$$A(X)+A^4(X) \equiv A_0+A_1X+A_2X^2+A_3X^3+A_0^4+A_1^4X^4+A_2^4X^8+A_3^4X^{12}$$

Replacing the latter equation into Eq. (4.1) gives:

$$\begin{aligned}&(X^8+X^2)(A_0+A_1X+A_2X^2+A_3X^3+A_0^4+A_1^4X^4+A_2^4X^8+A_3^4X^{12})\\&+(X^4+X)(A_0^2+A_1^2X^2+A_2^2X^4+A_3^2X^6+A_0^8+A_1^8X^8+A_2^8X^{16}+A_3^8X^{24}) \equiv 0\end{aligned} \tag{4.2}$$

Using the fact that $X^{16} \equiv X$ and developing Eq. (4.2) leads to:

$$\begin{aligned}&(A_0+A_0^4)X^2+A_1X^3+A_2X^4+A_3X^5+A_1^4X^6+A_2^4X^{10}+A_3^4X^{14}\\&+(A_0+A_0^4)X^8+A_1X^9+A_2X^{10}+A_3X^{11}+A_1^4X^{12}+A_2^4X+A_3^4X^5\\&+(A_0^2+A_0^8)X+A_1^2X^3+A_2^2X^5+A_3^2X^7+A_1^8X^9+A_2^8X^2+A_3^8X^{10}\\&+(A_0^2+A_0^8)X^4+A_1^2X^6+A_2^2X^8+A_3^2X^{10}+A_1^8X^{12}+A_2^8X^5+A_3^8X^{13} \equiv 0\end{aligned} \tag{4.3}$$

After further complete simplification of Eq. (4.3):

$$\begin{aligned}&(A_0^8+A_0^2+A_2^4)X+(A_0^4+A_0+A_2^8)X^2+(A_1^2+A_1)X^3\\&+(A_0^8+A_0^2+A_2)X^4+(A_2^2+A_3+A_3^4+A_2^8)X^5+(A_1^4+A_1^2)X^6\\&+A_3^2X^7+(A_0^4+A_2^2+A_0)X^8+(A_1^8+A_1)X^9\\&+(A_2^4+A_3^2+A_2+A_3^8)X^{10}+A_3X^{11}+(A_1^8+A_1^4)X^{12}\\&+A_3^8X^{13}+A_3^4X^{14} \equiv 0\end{aligned} \tag{4.4}$$

Since the degree of Eq. (4.4) is less than 16 no further residue simplification can be done, Eq. (4.4) is equivalent to the following system of equations:

$$\begin{cases} A_0^8+A_0^2+A_2^4 = 0 \\ A_0^4+A_0+A_2^8 = 0 \\ A_1^2+A_1 = 0 \\ A_0^8+A_0^2+A_2 = 0 \\ A_2^2+A_3+A_3^4+A_2^8 = 0 \\ A_1^4+A_1^2 = 0 \\ A_3^2 = 0 \\ A_0^4+A_2^2+A_0 = 0 \\ A_1^8+A_1 = 0 \\ A_2^4+A_3^2+A_2+A_3^8 = 0 \\ A_3 = 0 \\ A_1^8+A_1^4 = 0 \\ A_3^8 = 0 \\ A_3^4 = 0 \end{cases} \tag{4.5}$$

Using the field equation, namely, $\alpha^{16} = \alpha$ for $\alpha = A_0 , A_1 , A_2 , A_3$(since it is desired to find the unknowns in $GF(2^4)$) allows for the simplification of system (4.5) as:

$$\begin{cases} A_0^4+A_0+A_2^2 = 0 \\ T_2(A_1) = 0 \\ A_2 \varepsilon GF(2^2) \\ A_3 = 0 \end{cases} \tag{4.6}$$

4.4.2. Heuristic for general Loeloeian codes:

System (4.6) could be generalized by:

$$\begin{cases} A_0^{2^s}+A_0+A_{2^{s-1}}^2 = 0 \\ T_s(A_1) = 0 \\ A_{2^{s-1}} \varepsilon GF(2^s) \\ A_2 = A_3 = \cdots = A_{2^{s-1}-1} = A_{2^{s-1}+1} = A_{2^{s-1}+2} = \ldots = A_{2^s-1} = 0 \end{cases} \tag{4.7}$$

4.4.3. Bezzateev codes:

Since $G(X) = G_2(X) = X^5+1$ and $deg\ A(X) < deg\ G(X) = 5$ then:

$$A(X)+A^4(X) \equiv A_0+A_1X+A_2X^2+A_3X^3+A_4X^4+A_0^4+A_1^4X^4+A_2^4X^8+A_3^4X^{12}+A_4^4X^{16}$$

Replacing the latter equation in Eq. (4.1) gives:

$$\begin{aligned}
&(X^{10}+1)(A_0+A_1X+A_2X^2+A_3X^3+A_4X^4+A_0^4+A_1^4X^4 \\
&+A_2^4X^8+A_3^4X^{12}+A_4^4X^{16}) \\
&+(X^5+1)(A_0^2+A_1^2X^2+A_2^2X^4+A_3^2X^6+A_4^2X^8+A_0^8+A_1^8X^8 \\
&+A_2^8X^{16}+A_3^8X^{24}+A_4^8X^{32}) \equiv 0
\end{aligned} \tag{4.8}$$

Using the fact that $X^{16} \equiv X$ and developing Eq. (4.8) leads to:

$$\begin{aligned}
&(A_0+A_0^4)+A_1X+A_2X^2+A_3X^3+A_4X^4+A_1^4X^4+A_2^4X^8+A_3^4X^{12}+A_4^4X \\
&(A_0+A_0^4)X^{10}+A_1X^{11}+A_2X^{12}+A_3X^{13}+A_4X^{14}+A_1^4X^{14}+A_2^4X^3+A_3^4X^7+A_4^4X^{11} \\
&(A_0^2+A_0^8)+A_1^2X^2+A_2^2X^4+A_3^2X^6+A_4^2X^8+A_1^8X^8+A_2^8X+A_3^8X^9+A_4^8X^2 \\
&(A_0^2+A_0^8)X^5+A_1^2X^7+A_2^2X^9+A_3^2X^{11}+A_4^2X^{13}+A_1^8X^{13}+A_2^8X^6+A_3^8X^{14}+A_4^8X^7 \equiv 0
\end{aligned} \tag{4.9}$$

After further complete simplification of Eq. (4.9):

$$\begin{aligned}
&+(A_0^8+A_0^4+A_0^2+A_0)+(A_1+A_4^4+A_2^8)X+(A_2+A_1^2+A_4^8)X^2+(A_3+A_2^4)X^3 \\
&+(A_4+A_2^2+A_1^4)X^4+(A_0^8+A_0^2)X^5+(A_3^2+A_2^8)X^6 \\
&+(A_1^2+A_3^4+A_4^8)X^7+(A_1^8+A_4^2+A_2^4)X^8+(A_2^2+A_3^8)X^9 \\
&+(A_0^4+A_0)X^{10}+(A_3^2+A_1+A_4^4)X^{11}+(A_2+A_3^4)X^{12} \\
&+(A_1^8+A_4^2+A_3)X^{13}+(A_1^4+A_4+A_3^8)X^{14} \equiv 0
\end{aligned} \tag{4.10}$$

Since the degree of Eq. (4.10) is less than 16 no further residue simplification can be done, Eq. (4.10) is equivalent to the following system of equations:

$$\begin{cases} A_0^8+A_0^4+A_0^2+A_0 = 0 \\ A_1+A_4^4+A_2^8 = 0 \\ A_2+A_1^2+A_4^8 = 0 \\ A_3+A_2^4 = 0 \\ A_4+A_2^2+A_1^4 = 0 \\ A_0^8+A_0^2 = 0 \\ A_3^2+A_2^8 = 0 \\ A_1^2+A_3^4+A_4^8 = 0 \\ A_1^8+A_4^2+A_2^4 = 0 \\ A_2^2+A_3^8 = 0 \\ A_0^4+A_0 = 0 \\ A_3^2+A_1+A_4^4 = 0 \\ A_2+A_3^4 = 0 \\ A_1^8+A_4^2+A_3 = 0 \\ A_1^4+A_4+A_3^8 = 0 \end{cases} \tag{4.11}$$

Using the field equation, namely, $\alpha^{16} = \alpha$ for $\alpha = A_0 , A_1 , A_2 , A_3 , A_4$ (since it is desired to find the unknowns in $GF(2^4)$) allows for the simplification of system (4.11) as:

$$\begin{cases} A_0 \varepsilon GF(2^2) \\ A_1+A_3^2+A_4^4 = 0 \\ A_2^4+A_3 = 0 \end{cases} \tag{4.12}$$

4.4.4. Heuristic for general Bezzateev codes:

System (4.12) could be generalized by:

$$\begin{cases} A_0 \varepsilon GF(2^s) \\ A_1+A_{2^{s-1}+1}^2+A_{2^s}^{2^s} = 0 \\ A_{2^{s-1}}^{2^s}+A_{2^{s-1}+1} = 0 \\ A_2 = A_3 =\ldots = A_{2^{s-1}-1} = A_{2^{s-1}+2} = A_{2^{s-1}+3} =\ldots = A_{2^s-1} = 0 \end{cases} \tag{4.13}$$

4.5. Improved bound for $G_1(X)$:

Proposition 4.1. Eq. (4.7) has exactly 2^{3s-1} distinct solutions $A(X)$.

Proof. One solution being determined by one possible $A(X)$, one way of completely solving Eq. (4.7) can be done by first choosing independently $A_{2^{s-1}}$ (there are 2^s possible values since $A_{2^{s-1}} \varepsilon GF(2^s)$), then for each given $A_{2^{s-1}}$ derive the 2^s values A_0 using Prop. A.23 and finally picking any A_1 such that $T_s(A_1) = 0$ (from Prop. 2.5. and Prop. 2.10, there are exactly 2^{s-1} possible values). This means that there exactly $2^s 2^s 2^{s-1} = 2^{3s-1}$ distinct solutions $A(X)$.

Proposition 4.2. *dim* $R_{2s}(G_1(X)) \geq 3s-1$.

Proof. It is clear that the solutions of Eq. (4.7) form a vector space over $GF(2)$. Using Eq. (4.7) and Prop. 4.1. indicates that *dim* $R_{2s}(G_1(X)) \geq 3s-1$ if and only if all the solutions derived from Eq. (4.7) really satisfy Eq. (3.14), the redundancy equation of $G_1(X)$. First compute the quantity $A^{2^s}(X)+A(X)$ using Eq. (4.7) (it has been shown in Prop. 2.10. that $T_s(A_1) = 0$ with $A_1 \varepsilon GF(2^{2s})$ implies $A_1 \varepsilon GF(2^s)$, this result will be used later in this proof):

$$A(X)+A^{2^s}(X) \equiv A_0+A_1X+A_{2^{s-1}}X^{2^{s-1}}+A_0^{2^s}+A_1^{2^s}X^{2^s}+A_{2^{s-1}}^{2^s}X^{2^s 2^{s-1}}$$

$$\equiv A_{2^{s-1}}^2+A_1G_1(X)+A_{2^{s-1}}G_1^{2^{s-1}}(X) \tag{4.14}$$

Introducing Eq. (4.14) into Eq. (3.14):

$$T_s\left[\frac{A(X)+A^{2^s}(X)}{G_1(X)}\right] \equiv T_s\left[\frac{A_{2^{s-1}}^2}{G_1(X)}\right]+T_s[A_1]+T_s\left[\frac{A_{2^{s-1}}G_1^{2^{s-1}}(X)}{G_1(X)}\right]$$

$$\equiv T_s\left[\frac{A_{2^{s-1}}^2}{G_1(X)}\right]+T_s\left[\frac{A_{2^{s-1}}G_1^{2^{s-1}}(X)}{G_1(X)}\right] \tag{4.15}$$

Using again Prop. 2.9, Remark 4.2 and the fact that $A_{2^{s-1}} \varepsilon GF(2^s)$ (or equivalently $A_{2^{s-1}}^{2^s} + A_{2^{s-1}} = 0$), Eq. (4.15) becomes:

$$T_s\left[\frac{A(X)+A^{2^s}(X)}{G_1(X)}\right] \equiv T_s^{2^{s-1}}\left[\frac{A_{2^{s-1}}^2}{G_1(X)}\right] + T_s\left[\frac{A_{2^{s-1}} G_1^{2^{s-1}}(X)}{G_1^{2^s}(X)}\right]$$

$$\equiv T_s\left[\frac{A_{2^{s-1}}}{G_1^{2^{s-1}}(X)}\right] + T_s\left[\frac{A_{2^{s-1}}}{G_1^{2^{s-1}}(X)}\right]$$

$$\equiv 0$$

Finally, from Prop. 2.11.:

$$G_1(X) G_1^{2^{2s-1}}(X) T_{2s}\left[\frac{A(X)}{G_1(X)}\right] \equiv 0$$

which proves that the 2^{3s-1} solutions of Eq. (4.7) are indeed in the redundancy vector space of the Goppa code defined by $G_1(X)$, in other words $dim\ R_{2s}(G_1(X)) \geq 3s-1$. An inequality is needed since it is not clear that Eq. (4.7) provides the unique solutions of Eq. (3.14) when $G(X) = G_1(X)$.

4.6. Improved bound for $G_2(X)$:

Proposition 4.3: Eq. (4.13) has exactly 2^{5s} distinct solutions $A(X)$.

Proof. One solution being determined by one possible $A(X)$, a way of completely solving Eq. (4.13) can be found by first choosing independently A_0 (there are 2^s possible values since $A_0 \varepsilon GF(2^s)$), then picking any $A_{2^{s-1}+1}$ in $GF(2^{2s})$ (2^{2s} possible values) which determine automatically $A_{2^{s-1}}$, and finally taking $A_{2^s} \varepsilon GF(2^{2s})$ induces uniquely A_1. This means

that there are exactly $2^s 2^{2s} 2^{2s} = 2^{5s}$ distincts solutions $A(X)$.

Proposition 4.4: *dim* $R_{2s}(G_2(X)) \geq 5s$.

Proof. It is clear that the solutions of Eq. (4.13) form a vector space over $GF(2)$. Using Eq. (4.13) and Prop. 4.3. indicates that *dim* $R_{2s}(G_2(X)) \geq 5s$ if and only if all the solutions derived from Eq. (4.13) really satisfy Eq. (3.14), the redundancy equation of $G_2(X)$. First compute the quantity $A^{2^s}(X)+A(X)$ using Eq. (4.13):

$$A(X)+A(X)^{2^s} \equiv A_0+A_1X+A_{2^{s-1}}X^{2^{s-1}}+A_{2^{s-1}+1}X^{2^{s-1}+1}+A_{2^s}X^{2^s}$$

$$+A_0^{2^s}+A_1^{2^s}X^{2^s}+A_{2^{s-1}}^{2^s}X^{2^{s-1}2^s}+A_{2^{s-1}+1}^{2^s}X^{(2^{s-1}+1)2^s}+A_{2^s}^{2^s}X^{2^{2s}}$$

$$A(X)+A(X)^{2^s} \equiv A_{2^{s-1}+1}^{2}X+A_{2^{s-1}+1}^{2^s}X^{2^{s-1}}+A_{2^{s-1}+1}X^{2^{s-1}+1}$$

$$+(A_{2^{s-1}+1}^{2})^{2^s}X^{2^s}+A_{2^{s-1}+1}X^{2^s2^{s-1}}+A_{2^{s-1}+1}^{2^s}X^{2^s(2^{s-1}+1)} \qquad (4.16)$$

Using Prop. 2.11, remark 4.2 and substituting Eq. (4.16) implies:

$$T_s[\frac{A(X)+A^{2^s}(X)}{G_2(X)}] \equiv T_{2s}[\frac{A_{2^{s-1}+1}^{2}X+A_{2^{s-1}+1}^{2^s}X^{2^{s-1}}+A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

From Prop. 2.8., it is then obtained:

$$T_s[\frac{A(X)+A^{2^s}(X)}{G_2(X)}] \equiv T_{2s}^{2^{s-1}}[\frac{A_{2^{s-1}+1}^{2}X}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}^{2^s}X^{2^{s-1}}}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

$$\equiv T_{2s}[A_{2^{s-1}+1}^{2^s}X^{2^{s-1}}(\frac{1}{G_2^{2^{s-1}}(X)}+\frac{1}{G_2(X)})]$$

$$+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}] \qquad (4.17)$$

It is clear from Remark (4.2) that:

$$\frac{1}{G_2^{2^{s-1}}(X)}+\frac{1}{G_2(X)} \equiv \frac{1}{G_2^{2^{s-1}}(X)}+\frac{1}{G_2^{2^s}(X)}$$

$$\equiv \frac{G_2^{2^{s-1}}(X)+1}{G_2(X)}$$

$$\equiv \frac{X^{(2^s+1)2^{s-1}}+1+1}{G_2(X)}$$

$$\equiv \frac{X^{(2^s+1)2^{s-1}}}{G_2(X)} \qquad (4.18)$$

Replacing Eq. (4.18) into eq. (4.17) combined with Prop. 2.8. and Remark 4.2 yields:

$$T_s[\frac{A(X)+A^{2^s}(X)}{G_2(X)}] \equiv T_{2s}[\frac{A_{2^{s-1}+1}^{2^s}X^{2^{2s-1}+2^s}}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

$$\equiv T_{2s}^{2^s}[\frac{A_{2^{s-1}+1}^{2^s}X^{2^{2s-1}+2^s}}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

$$\equiv T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{3s-1}+2^{2s}}}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

$$\equiv T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]+T_{2s}[\frac{A_{2^{s-1}+1}X^{2^{s-1}+1}}{G_2(X)}]$$

$$\equiv 0$$

Finally, it is clear that:

$$G_2(X)G_2^{2^{2s-1}}T_{2s}[\frac{A(X)}{G_2(X)}] \equiv 0$$

which proves that the 2^{5s} solutions of Eq. (4.13) are indeed in the redundancy vector space of the Goppa code defined by $G_2(X)$; in other words $dim\ R_{2s}(G_2(X)) \geq 5s$. An inequality is needed since it is not clear that Eq. (4.13) provides the unique solutions of Eq. (3.14) when $G(X) = G_2(X)$.

4.7. Maximality of the solutions:

So far, when comparing real values from Table 1 and Table 2 to $dim\ R_{2s}(G_1(X))$ and $dim\ R_{2s}(G_2(X))$, the bounds provided by Prop. 4.2. and Prop. 4.4. are reached (or maximal) for s=2,3,4,5.

In fact, when s=2, it was shown that Eq. (4.7) and (4.13) are equivalent to Eq. (3.14). It is, nevertheless, an open problem to verify this statement for any $s > 2$; such a study is beyond the scope of this dissertation.

Interestingly, these bounds do not depend on the choice of the basis of $GF(2^{2s})$.

4.8. Practical interpretation:

Theorem 4.1. For $G_1(X)$, $n-k \leq s2^{s+1}-3s+1$.

Proof. Using Prop. 3.2, Def. 3.2 and Prop. 4.2. *Q.E.D.*

Theorem 4.2. For $G_2(X)$, $n-k \leq s2^{s+1}-3s$.

Proof. Using Prop. 3.2, Def. 3.2 and Prop. 4.4. *Q.E.D.*

Proposition 4.5. the Goppa codes defined by $G_2(X) = X^{2^s+1}+1$ and $G_3(X) = X^{2^s+1}+X^{2^s}+X$ are equivalent, in particular their corresponding parity check matrix have the same rank.

Proof. Use Prop. 3.3 and the mapping $X \rightarrow X+1$ *Q.E.D.*

It is always better when possible to have zero as a root for $G(X)$ for computational reasons (it is not necessary to program $0^0 = 1$ which saves one test), $G_3(X)$ is then preferred to $G_2(X)$.

Finally, puncturing (Mac williams and Sloane [16]) one redundancy bit of the code defined by $G_1(X)$ yields the same redundancy bound as the one provided by Theorem 4.2. without changing its constructive distance. If the results of Loeloeian and Conan [26] concerning the spectral properties of the Goppa codes defined by $G_1(X)$ are true for any s, namely $d_L = d_G+3$, then $G_1(X)$ and $G_2(X)$ are similar in decoding performance when using a MPR algorithm decoding scheme; algebraic decoding up to 2^s+1 errors with $n-k \leq s\,2^{s+1}-3s$ and $n \leq 2^{2s}-2^s-1$.

CONCLUSIONS

A new polynomial theory for the dimension of generalized Goppa codes is possible when using the trace operator. This approach to the determining of the dimension of generalized Goppa codes does not require a computer search, given that the redundancy equation can be solved analytically.

Applying the derived equations to two specific codes has provided original bounds (Theorem 4.1 and Theorem 4.2) on the dimension of a general class of binary Goppa codes; the results match the computer simulation for $s = 2,3,4,5$. These bounds do not depend on the basis of the finite field used so the results are general.

The condition given by Eq. (3.13) unifies the particular codes introduced by Loeloeian and Conan [25,26] and by Bezzateev and Shekhunova [27]. Additionally, general bounds for Goppa codes verifying Eq. (3.13) have been derived (Prop. 3.5 and Prop. 3.6).

RECOMMENDATIONS

The redundancy equation should be applied to other codes different from $G_1(X)$ and $G_2(X)$, in particular the ones unified by Eq. (3.13), namely:

$$G^{2^s}(X) \equiv G(X) \ \ mod \ \ (X^{2^{2s}}+X)$$

Some possible future research topics are:

- studying how large is the real distance of these Goppa codes compared to their constructive distance.
- proving the maximality of the bounds (Th. 4.1. and Th. 4.2).
- studying the redundancy equation when $p > 2$ for Goppa codes defined by $G_4(X) = X^{p^s}-X$ and $G_5(X) = X^{p^s+1}-1$ with locator field $GF(p^{2s})$.

APPENDIX

APPENDIX A

REVIEW OF THE FINITE FIELD ALGEBRA

A.1. Introduction:

Due to the fundamental role of modern algebra in error correcting code theory, it seems appropriate to include a general survey of the most important properties of finite fields needed when using linear error correcting block codes. It will be shown in particular how polynomial rings with residue classes are related to the pratical construction of the Galois fields.

Not all the proofs will be presented, the primary goal here is to gain understanding of the relations between algebra and error correcting theory. If additional information is required, Albert [33] or Jacobson [34,35] are good references.

A.2. Monoids:

Let S be a set of elements. A binary composition * on S is a rule that assigns to each pair of elements a and b of S a third unique element $c = a*b$. If for any $a, b, c \in S$, $a*(b*c) = (a*b)*c$, then * is said to be associative. If for any $a, b \in S$, $a*b = b*a$, then * is said to be commutative.

Definition A.1: A set M with the binary operation * is a monoid if the following conditions are satisfied:

(i) M is non-empty

(ii) * is well defined on M, namely for any x and $y \in M$, $x*y \in M$.

(iii) there is one element $1 \in M$ such that for any $a \in M$, $a*1 = 1*a = a$.

(iv) * is associative

A set satisfying the above conditions is usually noted $(M, *, 1)$. For example, the set of the counting numbers N with the standard addition is a monoid.

Proposition A.1. The unit element of a monoid is uniquely determined.

Proof. Suppose there are two units 1 and $1'$ in M then from Def. A.1.(iii):

$$\begin{cases} 1'*1 = 1*1' = 1 \\ 1'*1 = 1*1' = 1' \end{cases} \rightarrow 1' = 1 \qquad Q.E.D.$$

A.3. Groups:

Definition A.2. A set G with the binary composition * is a group if and only if:

(i) $(G, *, 1)$ is a monoid

(ii) every element x of G has an inverse in G, namely there exists an element y such that $x*y = y*x = 1$.

A set satisfying the above conditions is usually noted $(G, *, 1)$. For example, the set of integers Z with the standard addition is a group.

Proposition A.2. The inverse of any element x of a group is uniquely determined (it is usually denoted x^{-1}).

Proof. Let y and y' be two inverses of x, then from Def. A.2.(ii), $x*y' = y'*x = 1$. Multiplying both sides by y and using the associativity yields $(y*x)*y' = y*y'*x = y$, in other words $y = y'$ *Q.E.D.*

It is a common rule to have for $n \in Z$:

$$a^n = a*a*...*a \quad (n \text{ times})$$

which leads to the following useful properties for any $x, y \in G$ and $n, m \in Z$:

$$\begin{cases} x^{m+n} = x^n * x^m \\ x^{-n} = (x^n)^{-1} \\ x^0 = 1 \end{cases}$$

A group $(G, *, 1)$ is said to be abelian if $*$ is commutative. If G is abelian, then another useful property is derived for any $x, y \in G$ and $n \in Z$, namely:

$$(x*y)^n = x^n * y^n$$

An abelian group G is generated by finitely many elements if there exist some positive integer n and $a_1, a_2, \ldots, a_n \in G$ such that any element a of G can be represented by:

$$a = a_1^{i_1} * a_2^{i_2} * ... * a_n^{i_n} \quad \text{for } \textit{some} \quad i_1, i_2, \ldots, i_n \in Z$$

It is common use to write $G = \langle a_1, a_2, \ldots, a_n \rangle$.

Definition A.3. A group $(G, *, 1)$ is cyclic if it is generated by only one element, namely: $G = \langle a \rangle = \{ a^n \mid n \in Z \}$. Since the consecutive powers of a generates entirely G, a is called a primitive element of G.

A.4. Finite groups:

A group is said to be finite if it has finitely many elements. The cardinality of a group is usually written as $|G|$, so for a finite group, $|G| < \infty$.

It is interesting to study for a given $\alpha \in G$ the following sequence; α , α^2 , α^3 ,... The group G being finite, there must exist two positive integers k and l $(k > l)$ such that $\alpha^k = \alpha^l$, in other words $\alpha^{\mu} = 1$ for $\mu = k-l$. Since μ is finite, it is possible to have the following definition.

Definition A.4. Let G be a finite group, the order of an element $a \in G$ is the smallest positive integer e such that $a^e = 1$ and $a^i \neq 1$ for $0 < i < e$ (the order of a is denoted $o(a)$). The exponent of a group is the smallest strictly positive integer m such that for all $a \in G$, $a^m = 1$ (it is usually denoted $exp(G)$).

Proposition A.3. Let G be a finite abelian group. If for some positive integer n and element $a \in G$, $a^n = 1$ then $o(a)$ divides n.

Proof. Let's call $m = o(a)$. Using the Euclidian division on n and m yields $n = \lambda m + r$ with $0 \leq r < m$. This implies that:

$$a^n = (a^m)^{\lambda} * a^r$$
$$1 = 1 * a^r$$
$$a^r = 1$$

Suppose that $r \neq 0$, then the above equation shows a contradiction since the order of a would be $r < m$ and m was the smallest positive integer satisfying $a^m = 1$ so $r = 0$ and m must divide n. *Q.E.D.*

Proposition A.4. Let G be an finite abelian group and $a , b \in G$. If $gcd(o(a) , o(b)) = 1$ then $o(a*b) = o(a)o(b)$.

Proof. Let's denote $m = o(a)$, $n = o(b)$ and $k = o(a*b)$.

$$(a*b)^{mn} = (a^m)^n * (b^n)^m = 1 * 1 = 1$$

so from Prop. A.3, k divides mn. On the other hand:

$$(a*b)^k = 1 \rightarrow a^k = b^{-k} \rightarrow a^{kn} = (b^n)^{-k} \rightarrow a^{kn} = 1$$

so m divides kn but $gcd(m, n) = 1$ which shows that m divides k.

$$(a*b)^k = 1 \rightarrow b^k = a^{-k} \rightarrow b^{km} = (a^m)^{-k} \rightarrow b^{km} = 1$$

so n divides km but $gcd(m, n) = 1$ which shows that n divides k. Finally, n and m both dividing k and $gcd(m, n) = 1$ implies that mn divides k but it was shown before that k divides mn so $mn = k$. *Q.E.D.*

Proposition A.5. Let G be a finite abelian group of exponent $exp(G)$, then there is at least one element of order $exp(G)$.

Proof. Let's define $o(a) = \max\{o(b) \mid b \in G\}$ and suppose that there is some $b \in G$ such that $b^{o(a)} \neq 1$. It is then always possible to find a set of distinct prime elements $p_1, p_2, \ldots, p_s$ and positive integers $e_1, e_2, \ldots, e_s, f_1, f_2, \ldots, f_s$ such that:

$$o(a) = p_1^{e_1} p_2^{e_2} \ldots p_s^{e_s}$$
$$o(b) = p_1^{f_1} p_2^{f_2} \ldots p_s^{f_s}$$

Supposing $b^{o(a)} \neq 1$ shows that $o(b)$ doesn't divide $o(a)$, in other words there exist some i such that $f_i > e_i$. After a renumbering, it can be determined that $f_1 > e_1$. Defining, then, $a' = a^{p_1^{e_1}}$ and $b' = b^{p_2^{f_2} p_3^{f_3} \cdots p_s^{f_s}}$ implies:

$$o(a') = p_2^{e_2} p_3^{e_3} \ldots p_s^{e_s}$$
$$o(b') = p_1^{f_1}$$

Clearly, $gcd(o(a'), o(b')) = 1$ then from Prop. A.4, $o(a'*b') = p_1^{f_1} p_2^{e_2} \ldots p_s^{e_s}$ which constitutes a contradiction because the order of $a'*b'$ would be greater than $o(a)$, the maximal order in G. Then, there always exists a maximal element a such that $exp(G) = o(a)$. *Q.E.D.*

Proposition A.6. If G is a finite abelian group, then G is cyclic if and only if $exp(G) = |G|$. In other words, there always exists at least one primitive element in G.

Proof. If G is cyclic, it is obvious that $exp(G) = |G|$.

If $exp(G) = |G|$, then from Prop. A.5, there is an element $a \in G$ such that $o(a) = exp(G) = |G|$. In other words, $|G| = |< a >|$ which proves that $G = < a >$ *Q.E.D.*

A.5. Rings:

Definition A.5. A set R with two binary composition + and * (0 being the identity with respect to + and 1 the identity with respect to *, $0 \neq 1$) is said to be a ring if and only if:

(i) $(R, +, 0)$ is an abelian group

(ii) $(R, *, 1)$ is a monoid

(iii) for any $x, y, z \in R$, $(x+y)*z = x*z+y*z$ and $z*(x+y) = z*x+z*y$ (distributivity property)

A set satisfying the above conditions is usually noted $(R, +, *, 1, 0)$. A ring R is said to be commutative if * is commutative. Usually, * is omitted for simplifying purposes when there is no ambiguity ($x*y = xy$). For example, Z with the standard addition and multiplication is a commutative ring.

Proposition A.7. For any element a belonging to a ring R, $a0 = 0a = 0$ (this property shows that 0 is an absorbant element of R).

Proof. Using the distributivity and the fact that every element has an additive inverse:

$$(b+0)a = ba \rightarrow ba+0a = ba \rightarrow 0a = 0$$
$$a(b+0) = ab \rightarrow ab+a0 = ab \rightarrow a0 = 0 \quad Q.E.D.$$

Definition A.6. A subset I of the ring R is said to be an ideal if:

(i) $(I, +, 0)$ is a abelian group

(ii) For any $a \in I$ and any $b \in R$, then ab and $ba \in I$.

For example, the set of multiples of $k \in Z$ is an ideal usually denoted $kZ = \{ kn \mid \text{for } n \in Z \}$. It can be shown that the quotient of R over an ideal forms a ring called quotient ring $R_{/I}$.

A.6. Fields:

Definition A.7. A ring F having two binary composition + and * is said to be a field if and only if:

(i) $(F, +, 0)$ is an abelian group

(ii) $(F-\{0\}, *, 1)$ is a group

A field is said to be commutative if * is commutative. For example, Q or R or C with the standard addition and multiplication are commutative fields. It is common practice to note the additive inverse of an element a by $-a$ and the multiplicative inverse of a non zero element a by a^{-1}.

Proposition A.8. If a, b belong to a field F, then:

$ab = 0 \rightarrow a = 0$ or $b = 0$

Proof. Supposing that $ab = 0$ with $a \neq 0$ and $b \neq 0$, then using Prop. A.7. and the fact $F-\{0\}$ is a multiplicative group:

$$ab = 0 \rightarrow a^{-1}ab = b = a^{-1}0 = 0 \rightarrow b = 0$$
$$ab = 0 \rightarrow abb^{-1} = a = 0b^{-1} = 0 \rightarrow a = 0 \quad Q.E.D.$$

Definition A.8. The characteristic of a field is the smallest positive integer c such that for any $a \in F$, $\sum_{i=1}^{c} a = 0$.

The fields Q , R , C have a characteristic 0. It will be shown later on that there are some fields with a characteristic different from 0.

Definition A.9. Let F_1 and F_2 be two fields. A mapping ϕ from F_1 to F_2 is called an isomorphism if for any elements x , $y \in F_1$:

(i) $\phi(x +_{F_1} y) = \phi(x) +_{F_2} \phi(y)$

(ii) $\phi(x *_{F_1} y) = \phi(x) *_{F_2} \phi(y)$

It can be easily verified that $\phi(1_{F_1}) = 1_{F_2}$ and $\phi(0_{F_1}) = 0_{F_2}$ which is equivalent to say that F_1 and F_2 behaves the same way, in other words, they are isomorphically identical. If $F_1 = F_2$, then an isomorphism defined on F_1 is also called an automorphism.

A.7. Vector spaces:

Definition A.10. Let $(F , +_F , 0_F , *_F , 1_F)$ be a commutative field, the abelian group $(V , +_V , 0_V)$ forms a vector space over F if for any a , $b \in F$ and any X , $Y \in V$ there is an external binary composition . such that:

(i) $a.X \in V$

(ii) $a.(X +_V Y) = a.X +_V a.Y$

(iii) $(a +_F b).X = a.X +_V b.X$

(iv) $(a *_F b).X = a.(b.X)$

(v) $1_F.X = X$

Definition A.11. A set $V_1, V_2, \ldots, V_n$ of the vector space V over the field F are linearly independent if and only if:

$$\sum_{i=1}^{n} a_i V_i = 0_V \text{ for } \textit{some } a_i \in F \rightarrow a_i = 0_F$$

A set $V_1, V_2, \ldots, V_n$ of the vector space V over the field F are linearly dependent if and only if there exist $a_i \in F$ not all equal to zero such that:

$$\sum_{i=1}^{n} a_i V_i = 0$$

The vector space V is generated over F by the set $V_1, V_2, \ldots, V_n$ if any $X \in V$ can be represented with some coefficients $a_i \in F$ such that:

$$X = \sum_{i=1}^{n} a_i V_i$$

Finally, the dimension of a vector space over F (denoted $dim_F V$) corresponds to the number of elements of the smallest set representing V over F. Such a minimal set is called a basis of V over F.

In general, for a given field F, the vector space F^n is represented by n-tuples $(a_1, a_2, \ldots, a_n)$ with the following composition rules:

$$\begin{cases}(a_1 , a_2 , \ldots , a_n)+(b_1 , b_2 , \ldots , b_n) = (a_1+b_1 , a_2+b_2 , \ldots , a_n+b_n) \\ \lambda.(a_1 , a_2 , \ldots , a_n) = (\lambda.a_1 , \lambda.a_2 , \ldots , \lambda.a_n)\end{cases}$$

Definition A.12. A function d from a vector space V to R^+ is called a distance if it verifies the following properties for any vectors $X , Y , Z \in V$:

$$\begin{cases}d(X , Y) \leq d(X , Z)+d(Z , Y) \\ d(X , Y) = d(Y , X) \\ d(X , Y) \geq 0 \\ d(X , X) = 0\end{cases}$$

A.8. Polynomial rings:

One of the most interesting rings are the rings of a polynomial with coefficients over a certain field F. All the usual definitions and properties of the polynomial ring over the field of reals R are in fact true for any commutative field F. Such properties will be used in this section without proof since they are equivalent to those previously developed with $F = R$.

Let F be a field, then $F[X]$ consists of all the possible polynomials with indeterminate X and coefficients over F, namely:

$$F[X] = \left\{ \sum_{i=0}^{n} a_i X^i \mid a_i \in F , n \in N \right\}$$

It can be easily verified that $F[X]$ with the standard rules of addition and multiplication of polynomials forms a ring, these rules being:

$$\begin{cases}\sum_{i=0}^{n} a_i X^i + \sum_{j=0}^{m} b_j X^j = \sum_{k=0}^{max(n,m)} (a_k+b_k)X^k \\ (\sum_{i=0}^{n} a_i X^i)(\sum_{j=0}^{m} b_j X^j) = \sum_{k=0}^{n+m} (\sum_{l=0}^{k} a_l b_{k-l})X^k\end{cases}$$

The Euclidian division of the polynomial $A(X)$ by the polynomial $I(X)$ consists of finding $\lambda(X)$ and $R(X)$ such that:

$$A(X) = \lambda(X)I(X)+R(X) \qquad with\ deg\ R(X) < deg\ I(X)$$

It can be shown that the couple $\lambda(X)$ and $R(X)$ are uniquely defined.

From the ring $F[X]$, it is always possible to derive the residue classes over an ideal consisting of the set of multiples of a given polynomial $I(X)$ by adjoining each polynomial $A(X)$ its remainder $F(X)$ when using the Euclidian division $A(X)$ by $I(X)$. Such ring is called $F[X]_{//I[X]}$ and contains all the polynomials of degree less than $deg\ I(X)$.

If $A(X) = B(X)C(X)$ for some polynomial $B(X)$ and $C(X)$, then $B(X)$ or $C(X)$ are called divisors of $A(X)$. Any polynomial in $F[X]$ can be uniquely factorized, namely be the unique product of monic irreducible polynomials and a constant.

A polynomial has at most a number of roots equal to its degree (in some splitting field containing its coefficients).

A polynomial $A(X)$ is irreducible if and only if it has $A(X)$ or any element of F-{0} as unique divisor.

The greatest common divisor is unique and is noted $gcd(A(X),B(X))$. The least common multiple is also unique and is noted $lcm(A(X),B(X))$. Two polynomials $A(X)$ and $B(X)$ are relatively prime if they don't have any common divisor other than a constant; meaning that $gcd(A(X),B(X)) = 1$ and $lcm(A(X),B(X)) = A(X)B(X)$.

Theorem A.1. This theorem is also known as Bezout's theorem. Let $C(X) = gcd(A(X),B(X))$ then there exist two polynomials $U(X)$ and $V(X)$ such that:

$$A(X)U(X)+B(X)V(X) = C(X) \ \ with\ deg\ U(X),V(X) < max(deg\ A(X),deg\ B(X))$$

Proposition A.9. $F[X]_{/I[X]}$ is a field if and only if $I[X]$ is irreducible over $F[X]$.

The formal derivative of a polynomial $A(X) = \sum_{i=1}^{n} a_i X^i$ is $A'(X) = \sum_{i=1}^{n} i a_i X^{i-1}$. The derivative of any constant polynomial is equal to zero and the formal derivative is a linear operator, namely:

$$\begin{cases} (A(X)+B(X))' = A'(X)+B'(X) \\ (\lambda A(X))' = \lambda A'(X) \quad \text{for } any \ \lambda \in F \\ (A(X)B(X))' = A'(X)B(X)+A(X)B'(X) \end{cases}$$

If $g'(\alpha) = 0$ and $g(\alpha) = 0$ for some $\alpha \in F$, then α is at least a double root of $g(X)$.

A.9. Linear algebra:

As with polynomial rings over field F, all the common properties of linear algebra involving matrix theory and determinants are still valid when the coefficients of the matrices belong to any commutative field F. It will be important to remember the following properties.

The kernel of a matrix is always a vector space over F.

The determinant of a matrix A, denoted $det(A)$ is equal to the determinant of the transpose of A (the transpose of A is noted A^T).

The determinant of a Vandermonde is never equal to zero which means that the following matrix has a non zero determinant for any n as long as the α_i's belonging to some commutative field F are distinct:

$$\begin{bmatrix} 1 & 1 & . & . & 1 \\ \alpha_1 & \alpha_2 & . & . & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & . & . & \alpha_n^2 \\ . & . & . & . & . \\ . & . & . & . & . \\ \alpha_1^n & \alpha_2^n & . & . & \alpha_n^n \end{bmatrix}$$

A.10. Commutative finite fields:

A finite field F is by definition a field with a finite number of elements. Let $q = |F|$ be the number of distinct elements of F. Since finite fields play such an important role in error correcting code theory, additional time will be spent to prove some important properties of these fields.

It is assumed for clarity that finite fields are commutative. In fact, such hypothesis is redundant since Wedderburn's theorem proves that all the finite fields are always commutative, Jacobson [35]. This theorem is difficult to prove and requires considerable knowledge of commutative algebra. Since a good comprehension of error correcting theory can be obtained without it, it is not included here.

Proposition A.10. The characteristic of a finite field is a prime number.

Proof. Examine the sequence 1, 1+1, 1+1+...+1 and so on. Since the unity $1 \in F$, the elements of the previous sequence also belong to F. But F is finite so there exist two positive integers k and l $(k > l)$ such that $\sum_{i=1}^{k} 1 = \sum_{j=1}^{l} 1$. In other words, there exists a positive integer p $(p = k-l)$ such that $\sum_{i=1}^{p} 1 = 0$. It is clear by the construction that $p \geq 2$.

Suppose now that p is not a prime number, namely $p = \lambda\delta$ where λ and δ are two positive integers greater than 1. This would imply in particular by using the distributivity law

on F that:

$$(\sum_{i=1}^{\lambda} 1)(\sum_{j=1}^{\delta} 1) = \sum_{i=1}^{p} 1 = 0$$

From Prop. A.8, it follows that either $\sum_{i=1}^{\lambda} 1 = 0$ or $\sum_{i=1}^{\delta} 1 = 0$ contradicting the fact that p was the smallest positive integer satisfying $\sum_{i=1}^{p} 1 = 0$, hence p is a prime number. Since for any $a \in F$, $a = a^* 1$, it is clear that every element of F has characteristic p. Q.E.D.

The previous construction indicates that the set of $\{0 , 1 , 1+1 , \ldots , \sum_{i=1}^{p-1} 1 \}$ is isomorphic to $Z_{/pZ}$.

In the particular case $p = 2$, any $\alpha \in F$ verifies $\alpha = -\alpha$.

Proposition A.11. F is a vector space over $Z_{/pZ}$ and $q = p^m$ for some prime number p and strictly positive integer m.

Proof. It is clear from the previous construction that $q \geq p$, p being the characteristic of F. A constructive method to determine a vector space basis over $Z_{/pZ}$ is to start with $\beta_1 = 1$ and generate all the possible linear combination using $Z_{/pZ}$. If there are still elements of F not generated by $< \beta_1 >$ then pick one of them for β_2 and generate $<\beta_1 , \beta_2>$. Continue this process until all the elements are generated by some minimal basis $<\beta_1 , \beta_2 , \ldots , \beta_m>$ for some finite m (this process must stop since F is finite). In fact, using Def. A.10, it is also clear that $<\beta_1 , \beta_2 , \ldots , \beta_m>$ is included in F so it is equal to F. The cardinality must then coincide so $q = p^m$ for some finite m. *Q.E.D.*

Proposition A.12. : For any $x , y \in F$, $(x+y)^p = x^p + y^p$

Proof. For every commutative field , the binomial formula is always true, so:

$$(x+y)^p=\sum_{i=0}^{p}\begin{bmatrix}p\\i\end{bmatrix}x^i y^{n-i}$$

Since p is prime, it is clear that for some integer λ:

$$\begin{bmatrix}p\\i\end{bmatrix}=\frac{p(p-1)...(p-i+1)}{i(i-1)...1}=\lambda p \qquad \text{for } 0<i<p$$

but the sum was over $Z_{/pZ}$, so $\lambda p \equiv 0 \ mod\ p$ *Q.E.D.*

Prop. A.12. can also be used in a polynomial ring over F and for example:

$$(X^3+1)^p = X^{3p}+1$$

Theorem A.2. Any $\alpha \in F$ satisfies the field equation, namely $\alpha^{p^m} = \alpha$. In fact, the polynomial $X^{p^m}-X$ splits entirely in F. This result is also known as Fermat's theorem.

Proof. The case $\alpha = 0$ is obvious. Let's define for $b \in F$-{0} the following map on F $\phi(x) = bx$. Clearly, ϕ is an automorphism of $F-\{0\}$. In particular, the set of all the non zero elements of F, let's call it $\{ a_1 , a_2 , \ldots , a_{p^m-1} \}$ is mapped into itself, so:

$$a_1 a_2 \cdots a_{p^m-1} = ba_1 ba_2 \cdots ba_{p^m-1}$$

$$= b^{p^m-1} a_1 a_2 ... a_{p^m-1}$$

which implies that $b^{p^m-1} = 1$. Multiplying both sides of the previous equation by b completes the proof. *Q.E.D.*

The notion of order is defined since (F-{0},*,1) is an abelian group.

Proposition A.13. Every finite field F has a primitive element generating all the non zero elements.

Proof. Since F is commutative, it is enough from Prop. A.6. to show that $exp(F-\{0\}) = q-1$. Call $e = exp(F-\{0\})$, then Theorem A.2. indicates that $e \leq q-1$. Assuming that the $e < q-1$ would require that the equation $X^e-1 = 0$ has $q-1$ roots in F which is a contradiction because a polynomial of degree e has at most e distinct roots in F so $e = q-1$. *Q.E.D.*

Proposition A.14. All the finite fields of same order are isomorphic. It is then sufficient to denote them with a unique terminology as $GF(q)$.

Proof. See Jacobson [35].

Proposition A.15. Let $A(X)$ be a polynomial with coefficients in $GF(p)$, then $A^p(X) = A(X^p)$. In particular, if α is a root of $A(X)$ then α^p is also a root of $A(X)$.

Using Prop. A.12. on $A(X)$ yields:

$$A^p(X) = \sum_{i=0}^{n} a_i^p X^{ip}$$

Since $a_i \in G(p)$, $a_i^p = a_i$ from Theorem A.2., it then is obtained $A^p(X) = A(X^p)$

Let α be a root of $A(X)$ which implies $A(\alpha) = 0$, then:

$$A^p(\alpha) = A(\alpha^p) = 0 \quad Q.E.D.$$

Definition A.13. For any element α of F, the corresponding minimal polynomial $M_\alpha(X)$ is a monic polynomial with coefficients over $GF(p)$ and smallest degree such that $M_\alpha(\alpha) = 0$.

Proposition A.16. The minimal polynomial of any element in F is unique and irreducible over $GF(p)[X]$.

Proof. Supposing that α has two minimal polynomial $M_1(X)$ and $M_2(X)$ of same degree, then doing an Euclidian division of $M_1(X)$ by $M_2(X)$ gives for some $\lambda(X)$ and $R(X)$:

$$M_1(X) = M_2(X)\lambda(X)+R(X) \quad with \ deg \ R(X) < deg \ M_2(X)$$

Since $M_1(\alpha) = M_2(\alpha) = 0$, this implies that $R(\alpha) = 0$ which is a contradiction because $M_2(X)$ was the polynomial with the smallest degree having α as a root, it is then required that $R(X) = 0$. $M_1(X)$ and $M_2(X)$ have same degree and are both monic thus $\lambda(X) = 1$ which shows the unicity.

Clearly with the same argumentation as above, a minimal polynomial has to be irreducible. *Q.E.D.*

The unicity of the minimal polynomial of an element α leads to the definition of the degree of such element.

Definition A.14. Let $M_\alpha(X)$ be the minimal polynomial of $\alpha \in F$, then the degree of $M_\alpha(X)$ is called the degree of α.

Proposition A.17 The degree of an element of $GF(p^m)$ is less or equal to m

Proof. Since $GF(p^m)$ is a vector space over $GF(p)$ of dimension m, a set of $m+1$ vectors are linearly dependent. Taking the set $\{1, \alpha, \alpha^2, \ldots, \alpha^m\}$ shows that there must be a polynomial of degree less than $m+1$ with coefficients over $GF(p)$ having α as a root. If the corresponding polynomial is not monic, dividing it by its higher coefficient gives the required polynomial. *Q.E.D.*

Definition A.15. The minimal polynomial of a primitive element is called a primitive polynomial.

Proposition A.18. The degree of a primitive element on $GF(p^m)$ is always m.

Proof. From Prop. A.15., $M_\alpha(X)$ has at least α, α^p, α^{p^2}, . . . , $\alpha^{p^{m-1}}$ as roots. They are all distinct since α has order p^m-1. Also from Prop. A.17., $M_\alpha(X)$ has at most m roots in F and m distinct roots have been found. *Q.E.D.*

Proposition A.19. $GF(q^m)$ is isomorphic to $GF(q)_{/I(X)}$ where $I(X) \varepsilon GF(q)[X]$ is an irreducible polynomial of degree m (q being a prime power).

Proof. See application of Prop. A.9., the residue ring consists of all the polynomial with coefficients over $GF(q)$ of degree less than m, there are a total of q^m of them. *Q.E.D.*

Proposition A.20. Let $f(X)$ be a polynomial with coefficients over $GF(p)$ having a root $\alpha \varepsilon GF(p^m)$, then $M_\alpha(X)$ divides $f(X)$ over $GF(p)[X]$. In particular, the minimal polynomial of any $\alpha \varepsilon GF(p^m)$ divides $X^{p^m}-X$.

Proof. Since $f(\alpha) = 0$ and $M_\alpha(\alpha) = 0$, both polynomials have a common root. Performing an Euclidian division of $f(X)$ by $M_\alpha(X)$ implies that:

$$f(X) = \gamma(X)M_\alpha(X)+R(X) \quad with \ deg \ R(X) < deg \ M_\alpha(X)$$

so $R(\alpha) = 0$. If $R(X)$ is not equal to zero, that would contradict the minimality of $M_\alpha(X)$. *Q.E.D.*

Proposition A.21. $GF(p^m)$ can be embedded in $GF(p^{ms})$ for any positive integer $s > 1$.

Proof. From Prop A.19., pick an irreducible polynomial of degree s with coefficients over $GF(p^m)$. The existence of such polynomial is not shown here, see Jacobson [35].

Q.E.D.

Proposition A.22. The map from $GF(p^m)$ to itself defined by $\phi(X) = X^p$ is an automorphism (also known as the Frobenius automorphism). In particular, if an expression $A = 0$, it is equivalent to say that $A^p = 0$.

Proof. This map is clearly injective. It is also surjective since there is always a p^{th} root in $GF(p^m)$, namely:

$$x^{1/p} = x^{p^{m-1}} \qquad Q.E.D.$$

Proposition A.23. The equation $X^{2^s}+X+\alpha = 0$ has exactly 2^s distinct solutions in $GF(2^{2s})$ when $\alpha \in GF(2^s)$.

Proof. Define $H(X) = X^{2^s}+X+\alpha$, then the formal derivative of $H(X)$ is equal to 1 so it cannot vanish implying that $H(X)$ has distinct roots in some splitting field.

Let x_1 and x_2 be two distinct roots of $H(X)$, then:

$$\begin{cases} x_1^{2^s} + x_1 + \alpha = 0 \\ x_2^{2^s} + x_2 + \alpha = 0 \end{cases} \rightarrow (x_1+x_2)^{2^s} + (x_2+x_1) = 0$$

From Prop. A.21., $GF(2^{2s})$ contains $GF(2^s)$ so the previous equation shows that $(x_1+x_2) \in GF(2^s)$. If there is at least one root of $H(X)$ in $GF(2^{2s})$, then automatically there are 2^s distinct roots in that same field (constructing all the roots from the first one $\gamma \in GF(2^{2s})$ by $\beta+\gamma$ for $\beta \in GF(2^s)$).

Prop. A.22. also shows that for a root $\gamma \in GF(2^{2s})$ of $H(X)$:

$$\gamma^{2^s}+\gamma+\alpha = 0 \rightarrow \gamma+\gamma^{2^s}+\alpha^{2^s} = 0 \rightarrow \alpha^{2^s} = \alpha$$

in other words, $\alpha \in GF(2^s)$ is a required condition for $H(X)$ to have a root in $GF(2^{2s})$.

Define the following sets for $\alpha \in GF(2^s)$:

$$H_\alpha = \left\{ \beta \mid \beta^{2^s}+\beta = \alpha \, , \, \beta \in GF(2^{2s}) \right\}$$

It is clear that the H_α's are disjoint because assuming there is $\beta \in GF(2^{2s})$ and $\alpha_1 \, , \, \alpha_2 \in GF(2^s)$ such that $\beta^{2^s}+\beta+\alpha_1 = 0$ and $\beta^{2^s}+\beta+\alpha_2 = 0$ implies $\alpha_1 = \alpha_2$.

Finally, it is clear that scanning $\beta \in GF(2^{2s})$ creates non-empty sets H_α , each no empty-set containing exactly 2^s distinct elements. *Q.E.D.*

A.11. Example of the representation of the $GF(2^4) = GF(16)$:

The method described here can be applied for any p and m to generate $GF(p^m)$.

From Prop. A.19., it is sufficient enough to find one irreducible polynomial of degree 4 over $GF(2)$ to construct $GF(16)$. The irreducible polynomials of degree 1 are X and $X+1$.

For the degree 2, only X^2+X+1 is irreducible since X^2 and X^2+X can be divided by X and from Prop. A.12. $X^2+1 = (X+1)^2$ so X^2+1 is divided by $X+1$.

It is not necessary to find any irreducible polynomial of degree 3 if it can be verified that the only polynomial of degree 4 tested has a constant coefficient equals to 1 (not divided by X) and an odd number of non zero coefficients (not divided by $X+1$). If an irreducible polynomial of degree 4 could be divided by an irreducible polynomial of degree 3, this would imply that it would have also to be divided by X or $X+1$.

It is clear that X^4+X+1 cannot be divided by X or $X+1$ since 0 or 1 are not roots ($0+0+1 \neq 0$ and $1+1+1 = 1 \neq 0$). The only test that has to be done to verify the irreduciblity of X^4+X+1 is trying to divide it by X^2+X+1. Performing an Euclidian division yields:

$$X^4+X+1 = (X^2+X)(X^2+X+1)+1$$

Since the remainder is not equal to 0, this completes the test.

Whenever programming a finite field, it is always interesting to choose an irreducible polynomial which is also primitive. If the polynomial is irreducible and primitive, it has a primitive root α which generates all the non zero elements of the finite field.

In the particular case of $GF(16)$, X^4+X+1 is primitive because when using the dependency relation $\alpha^4 = \alpha+1$, all the consecutive powers of α are generated with a sequence of order 15, namely:

$\alpha^0 = 1$
$\alpha^1 = \alpha$
$\alpha^2 = \alpha^2$
$\alpha^3 = \alpha^3$
$\alpha^4 = \alpha+1$
$\alpha^5 = \alpha^2+\alpha$
$\alpha^6 = \alpha^3+\alpha^2$
$\alpha^7 = \alpha^4+\alpha^3 = \alpha^3+\alpha+1$
$\alpha^8 = \alpha^4+\alpha^2+\alpha = \alpha^2+1$
$\alpha^9 = \alpha^3+\alpha$
$\alpha^{10} = \alpha^4+\alpha^2 = \alpha^2+\alpha+1$
$\alpha^{11} = \alpha^3+\alpha^2+\alpha$
$\alpha^{12} = \alpha^4+\alpha^3+\alpha^2 = \alpha^3+\alpha^2+\alpha+1$
$\alpha^{13} = \alpha^4+\alpha^3+\alpha^2+\alpha = \alpha^3+\alpha^2+1$
$\alpha^{14} = \alpha^4+\alpha^3+\alpha = \alpha^3+1$
$\alpha^{15} = \alpha^4+\alpha = 1$ *cyclic structure*

One useful way to look at $GF(16)$ is to use 1 , α , α^2 , α^3 as a primitive basis. When simulating the $GF(16)$ with hardware or software, the coefficients of the basis representing any element of $GF(16)$ are the bits stored in memory.

Clearly, an addition is done by a *XOR* operation modulo 2. The multiplication is done using the exponential and logarithm in base α and remembering that $\alpha^{15} = 1$.

For example, 4+5 can be mapped isomorphically in base 2 by respectively α^2 and α^2+1. So 4+5 is $\alpha^2+\alpha^2+1 = 1$, then 4+5 = 1.

For the multiplication, 4 times 5 (noted 4.5) is:

$$\alpha^2(\alpha^2+1) = \alpha^4+\alpha^2 = \alpha^2+\alpha+1$$

so 4.5 is 7. The same result would have been obtained using the logarithm form, namely 4 is α^2 and 5 is α^8 , so 4.5 is $\alpha^{2+8} = \alpha^{10} = \alpha^2+\alpha+1$.

Finally, 4/5 is:

$$\alpha^2/\alpha^8 = \alpha^{-6} = \alpha^{15}\alpha^{-6} = \alpha^9 = \alpha^3+\alpha$$

which is 10.

It can be verified that the only irreducible binary polynomials of degree 4 are:

$$\begin{cases} X^4+X+1 \\ X^4+X^3+1 \\ X^4+X^3+X^2+X+1 \end{cases}$$

the first two polynomials being primitive.

It is important to remember that the finding of irreducible or primitive polynomials is something very difficult when the corresponding degree becomes large. For $p = 2$, referring to the tables provided by Peterson [9] is the quickest way to locate them. More information about computing in finite fields can be found in Berlekamp [10].

LIST OF REFERENCES

LIST OF REFERENCES

[1] C. Shannon, "A Mathematical Theory of Communication," *Bell Sytem Tech. Journal*, Vol. 27, Part I, Jul. 1948, pp. 379-423

[2] C. Shannon, "A Mathematical Theory of Communication," *Bell Sytem Tech. Journal*, Vol. 27, part II, Oct. 1948, pp. 623-656

[3] R. W. Hamming, "Error Detecting and Error Correcting Codes," *Bell System Tech. Journal*, Vol. 28, pp. 147-160, Apr. 1950

[4] M. J. E. Golay, "Notes on digital Coding," *Proc. IRE*, Vol. 37, pp. 657, June 1949

[5] A. Hocquenghem, "Codes Correcteurs d'erreurs," *Chiffres*, Vol. 2, pp. 147-156, 1959

[6] R. C. Bose and D. K. Ray-Chaudhuri, "On A Class Of Error Correcting binary Group Codes," *Info. And Control*, Vol. IT-3 , pp. 68-79, Mar. 1960

[7] I. S. Reed and G. Solomon, "Polynomial Codes over certain Finite Fields," *Journal Soc. Indust. Appl. Math.*, Vol. 8, pp. 300-304, 1960

[8] W. W. Peterson, "Encoding and error correction Procedures for the Bose Chaudhuri Codes," *IRE Trans. Inf. Theory*, Vol. IT-6, pp. 459-470, 1960

[9] W. W. Peterson, *Error Correcting Codes*, MIT, Cambridge, Mass and John Wiley and sons, New York, 1961

[10] E. R. Berlekamp *Algebraic Coding Theory*, New York, Mac Graw Hill, 1968

[11] S. Lin and E. J. Weldon, Jr., "Long BCH codes are bad," *Info. And Control*, Vol. IT-10 , pp. 445-451, Sept. 1967

[12] E. R. Berlekamp, "Long Primitive binary BCH Codes Have distance $d^{-} = 2n \ln R^{-1} / \log n$," *IEEE Trans. Info. Theory*, Vol. IT-18, pp. 415-426, May 1972

[13] H. J. Helgert, "Decoding of Alternant codes," *IEEE Trans. Info. Theory*, Vol. IT-23, pp. 513-514, July 1977

[14] H. J. Helgert, "Non Cyclic Generalisation of BCH and Srivastava Codes," *Info. And Control*, Vol.21 , pp. 280-290, Oct. 1972

[15] H. J. Helgert, "Alternant code," *Info. And Control*, Vol.26 , pp. 369-380, Dec. 1974

[16] F. J. Mac Williams and N. J. A. Sloane *The Theory of Error Correcting Codes*, Amsterdam-New York-Oxford: North Holland, 1977

[17] V. D. Goppa, "A New Class of Linear Error Correcting Codes," *Problems Info. Transmission*, Vol. 6, pp. 205-225, Sept. 1970

[18] V. D. Goppa, "Rational Representation of codes and (L,g) codes," *Problems Info. Transmission*, Vol. 7, pp. 115-225, Sept. 1971

[19] J. Conan, "A recursive procedure for the solution of the minimal partial realization problem for scalar rational sequences," *Revue Roumaine de Mathematiques Appliquees*, Vol. XXX, no. 8, pp. 625-645, Aug. 1985

[20] S. Lin and D. Costello, Jr. *Error Control Coding*,, Prentice-Hall, Inc., Englewoods Cliffs, N.J. 07632

[21] O. Moreno, "Symetries of binary Goppa codes," *IEEE Trans. Info. Theory*, Vol. IT-25, pp. 609-612, Sept. 1979

[22] C. L. Chen, "Equivalent Irreducible Goppa Codes," *IEEE Trans. Info. Theory*, Vol. IT-24, pp. 766-770, Nov. 1978

[23] A. L. Berman, H. J. Helgert and N. G. Berman, "Equivalent Classes Of Alternant Codes," *Comsat Technical Review*, Vol. 14, pp. 313-338, Fall 1985

[24] E. R. Berlekamp, H. Rumsey and G. Solomon, "On the solutions of algebraic equation over finite fields," *Info. And Control*, Vol.10 , pp. 553-564, May 1967

[25] M. Loeloeian and J. Conan, "A [55,16,19] Binary Goppa Code," *IEEE Trans. Info. Theory*, Vol. IT-30, p. 773, Sept. 1984

[26] M. Loeloeian and J. Conan, "A transform Approach to Goppa codes," *IEEE Trans. Info. Theory*, Vol. IT-33, pp. 105-115, Jan. 1987

[27] S. V. Bezzataev and N. A. Shekhunova, "Constructive Distance of the Best Among Known (55,16,19) Goppa Codes," *Problems Info. Transmission*, Vol. 23, No. 4, p. 352, Nov. 1987

[28] T. Verhoeff, "An Updated Table of Minimum-Distance Bounds for Binary Linear Codes," *IEEE Trans. Info. Theory*, Vol. IT-33, pp. 665-680, Sept. 1987

[29] M. Loeloeian and J. Conan, "On a class of generalized Goppa codes," presented at the IEEE Int. Symp. Information Theory, Ann Arbor, MI, Oct. 6-9, 1986

[30] P. Delsarte, "Bounds for unrestricted codes by linear programming," *Philips Res. Reports*, Vol. 27, pp. 272-289, 1972

[31] H. Helgert, "Srivastava codes," *IEEE Trans. Info. Theory*, Vol. IT-18, pp. 292-297, Mar. 1972

[32] P. Delsarte, "On Subfield subcodes of Reed Solomon codes," *IEEE Trans. Info. Theory*, Vol. IT-21, pp. 575-576, May 1975

[33] A. A. Albert, *Fundamental Concepts of Higher Algebra*, University of Chicago Press, Chicago, Ill. (1956)

[34] N. Jacobson, *Lectures in Abstract Algebra*, Van Nostrand, Princeton, New Jersey, Vol. 1, 1951, Vol. 2, 1953, Vol. 3, 1964

[35] N. Jacobson, *Basic Algebra 1*, 2^{nd} Edition, W.H. Freeman and Compagny, New York, 1985