ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE

ALGO LAB

Good Ensembles of Goppa Codes

Semester Project
Winter 2006-2007

Ghid Maatouk
Professor: Amin Shokrollahi
Supervisor: Mahdi Cheraghchi

Abstract

It is well-known that random error-correcting codes achieve the Gilbert-Varshamov bound with high probability. In [2], the authors describe a construction which can be used to yield a polynomially large family of codes of which a large fraction achieve the Gilbert Varshamov bound. In this project, we investigate ways to obtain codes known to achieve this bound, given such a family of codes. Since computing the minimum distance of a code is NP-hard, we work with a class of Goppa codes described in [1] whose minimum distance is known. We know that there exist Goppa codes which achieve the Gilbert-Varshamov bound, but we do not know if there are codes in this class which achieve it. We investigate various approaches to determining the rate of a code and try to apply them to this class of codes in order to determine if they achieve the Gilbert-Varshamov bound. These approaches include investigating upper bounds on the covering radius of a code and refining an existing lower bound on the code dimension. We also implemented the described class of Goppa codes using the PARI/GP computer algebra system [5], in order to obtain numerical values which would allow us to detect patterns and formulate conjectures regarding those codes.

Contents

1 Motivation

We expose the results of the construction in [2] and explain why they make it meaningful to investigate the dimension of codes whose minimum distance is known, so as to determine if they achieve the Gilbert-Varshamov Bound. We start by stating this bound.

1.1 The Gilbert-Varshamov Bound

Constructions of error-correcting codes try to achieve a good tradeoff between a high rate and a large minimum distance. The Gilbert-Varshamov (GV) bound defines a tradeoff between rate and minimum distance.

Theorem 1 *[4] Let $0 \leq \delta < 0.5$. Then there exists an infinite sequence of $[n, k, d]$ binary linear codes such that $\frac{d}{n} \geq \delta$ and with rate $r = \frac{k}{n}$ satisfying*

$$r \geq 1 - H(\frac{d}{n}) \; \forall n.$$

Moreover, we know that a random code achieves the GV bound with high probability.

1.2 Using Pseudorandomness

In [2], the authors used pseudorandomness under some hardness assumptions to obtain a construction which results in the following: given a family of efficiently samplable objects of which an ϵ-fraction satisfies a property P verifiable in polynomial space, and such that the size of the family is exponential in the representation size n of the objects, the construction outputs a family of size polynomial in n, of which at least an $\epsilon - \frac{1}{n^k}$ fraction still satisfies P.
We can apply this construction to error-correcting codes as follows: given a code of block length n, its representation size is polynomial in n and the number of such codes is exponential in n. We know that a randomly chosen code achieves the GV bound with high probability, hence a large fraction of the codes of block length n achieves the GV bound. By applying the construction in [2], we obtain a polynomially large family of codes of which still a large fraction achieves the GV bound. It hence becomes feasible to

enumerate these codes and check various properties of theirs.
Given such a polynomially large family of codes, we are guaranteed to find a code (actually, many codes) among them which achieves the GV bound, but we still don't know which. In order to check if a code achieves the GV bound, we need to know its minimum distance and its dimension. We can compute the rate of a code of block length n in time polynomial in n. However, computing the minimum distance of a code is NP-hard. Hence it becomes interesting to focus on codes whose minimum distance is already known. An example of such codes will be investigated in section 3.
In this project, we focused on Goppa codes (section 2), and specifically on the class of Goppa codes described in [1]. We implemented the construction of this class of codes to obtain numerical values of the codes dimensions.
Other possible approaches to the problem of finding a single good code involve investigating properties of the covering radius of codes, and improvements of a known lower bound on the code dimension (section 6).

2 Goppa codes

Definition 1 *Given a finite field $\mathbb{F}_{q^m}$ and a polynomial $G(x)$ of degree t with coefficients from $\mathbb{F}_{q^m}$, we define a subset $L = \{\gamma_i\}_{i=1}^n$ of $\mathbb{F}_{q^m}$ such that no γ_i is a root of $G(x)$. Then the Goppa code $\Gamma(L, G)$ consists of all vectors $c \in \mathbb{F}_q^n$ such that*

$$\sum_{i=1}^{n} \frac{c_i}{x - \gamma_i} \equiv 0 \pmod{G(x)}$$

Goppa codes have the following properties [4]:

- A well-known lower bound on the code dimension k is given by

$$k \geq n - mt.$$

- The minimum distance d of the code is lower bounded by $d \geq t + 1$. If the Goppa code is binary and such that $G(x)$ has no multiple zeros, we get the tighter lower bound $d \geq 2t + 1$.

- There exist sequences of Goppa code which asymptotically meet the Gilbert Varshamov bound (see 4.1).

3 The construction in [1]

In [1], the authors construct a subclass of Goppa codes over $\mathbb{F}_2$ such that the Goppa polynomial is a separable polynomial of degree t given by

$$g(x) = x^t + A,$$

where $t \mid (2^m - 1)$, A is a t-th power in $\mathbb{F}_{2^m} \setminus \{0\}$ and

$$L = \{\gamma \in \mathbb{F}_{2^m} : G(\gamma) \neq 0\}.$$

They prove that the minimum distance of such codes is equal to the design distance $d = 2t + 1$.
The interest of this construction from the perspective of [2] is that we obtain a family of codes for which the minimum distance is known. Hence it becomes possible to check in polynomial time which codes of the family output by the construction of [2] achieve the GV bound.

3.1 An interesting property of the family of Goppa codes in [1]

Definition 2 *Two linear codes $\mathcal{C}$ and $\mathcal{C}'$ are said to be equivalent if and only if there exists a permutation π of of the codeword components such that*

$$\pi : \mathcal{C} \to \mathcal{C}'$$

is an isomorphism.

Claim
The class of Goppa codes defined in [1] is such that given m and t, different choices of A define equivalent codes.

Proof
Let $\mathcal{C}$ and $\mathcal{C}_\alpha$ be the Goppa codes defined over $\mathbb{F}_{2^m}$ by the Goppa polynomials $G(x) = x^t + 1^t = x^t + 1$ and $G(x) = x^t + \alpha^t$, respectively. We will prove the claim by showing that there exists a permutation π of the codeword components such that

$$\pi : \mathcal{C} \to \mathcal{C}_\alpha$$

is an isomorphism, so that $\mathcal{C}$ and $\mathcal{C}_\alpha$ are equivalent codes.
Let $c = (c_1, \ldots, c_n)$ be a codeword of $\mathcal{C}$. It satisfies the condition

$$\sum_{i=1}^{n} \frac{c_i}{x - \gamma_i} \equiv 0 \pmod{x^t + 1}.$$

We can index each coordinate c_i of the codeword by the corresponding element of $\mathbb{F}_{2^m}$, γ_i, where each γ_i is such that it is not a root of the Goppa polynomial $x^t + 1$, i.e. it is not a tth root of unity.
Define the permutation π_α over $\mathbb{F}_{2^m}$ as $\pi_\alpha(a) = \alpha a$. π_α is indeed a permutation since $\pi_\alpha(a) = \pi_\alpha(b)$ means $\alpha a = \alpha b$, hence $a = b$. Moreover, π_α maps the tth roots of unity $\{1, \xi, \ldots, \xi^{t-1}\}$ to the tth roots of α $\{\alpha, \alpha\xi, \ldots, \alpha\xi^{t-1}\}$; and it maps each element γ_i which is not a tth root of unity to the element $\alpha\gamma_i$, which is not a tth root of α. We can therefore view π_α as simply a permutation of the components of the codeword c, such that each c_i is now indexed by $\alpha\gamma_i$. It is enough then to show that the obtained vector is a codeword in the Goppa code $\mathcal{C}_\alpha$, i.e. that it satisfies

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha\gamma_i} \equiv 0 \pmod{x^t + \alpha^t}.$$

We have indeed

$$\begin{aligned} \sum_{i=1}^{n} \frac{c_i}{x - \alpha\gamma_i} &= \sum_{i=1}^{n} \frac{c_i/\alpha}{x/\alpha - \gamma_i} \\ &= \frac{1}{\alpha} \sum_{i=1}^{n} \frac{c_i}{y - \gamma_i} \\ &\equiv 0 \pmod{y^t + 1} \end{aligned}$$

where we have performed a change in the formal variable x by replacing it with $y = x/\alpha$. But

$$y^t + 1 = (\frac{x}{\alpha})^t + 1 = \frac{1}{\alpha^t}(x^t + \alpha^t).$$

So we get that

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha\gamma_i} \equiv 0 \pmod{x^t + \alpha^t},$$

hence we obtained a codeword in the Goppa code $\mathcal{C}_\alpha$. Therefore $\mathcal{C}$ and $\mathcal{C}_\alpha$ are equivalent Goppa codes. □

Clearly, two equivalent codes have the same minimum distance, since their lowest-weight codewords are permutations of each other and hence have the same weight. Also, two equivalent linear codes have the same dimension since they have the same number of codewords.

4 Asymptotic Behaviour of Goppa Codes in General and the Class of [1] in Particular

In this section, we reproduce the proof that there exist Goppa codes which achieve the GV bound. We investigate the applicability of this proof to the specific case of the class of codes described in [1]. We show that the proof cannot be adapted to this particular case. Hence we cannot say anything yet about the asymptotic goodness of this class of Goppa codes.

4.1 A Counting-Based Argument to Prove that Goppa Codes are Asymptotically Good

Theorem 2 *[7] There exists a sequence of Goppa codes over $\mathbb{F}_q$ which meets the GV bound.*

Proof
Given parameters $n = q^m$, t, d, we want to find a Goppa code $\Gamma(L, G)$ with minimum distance d, with $L = \mathbb{F}_{q^m} = \{\alpha_0, \ldots, \alpha_{n-1}\}$, and with $G(x)$ being a degree-t polynomial irreducible over $\mathbb{F}_{q^m}$.
Consider any word $\mathbf{c}$= $(c_0, \ldots, c_{n-1})$ of weight j. We can write

$$\sum_{i=0}^{n-1} \frac{c_i}{x - \alpha_i} = \frac{f_c(x)}{\prod_{i=0}^{n-1}(x - \alpha_i)}$$

where the degree of $f_c(x)$ is less than j. Then for $\mathbf{c}$ to be a codeword, it must satisfy

$$\frac{f_c(x)}{\prod_{i=0}^{n-1}(x - \alpha_i)} = 0 \pmod{G(x)}$$

Since $G(x)$ is irreducible over $\mathbb{F}_{q^m}$, it has no common factors with the denominator of this fraction and must divide $f_c(x)$ for $\mathbf{c}$ to be a codeword. But $f_c(x)$ has degree at most $j-1$, so there are at most $\lfloor \frac{j-1}{t} \rfloor$ irreducible polynomials of degree t that can divide $f_c(x)$.
Counting over the codewords that we do not want to include in our code, i.e. all codewords of weight $j < d$, the number of irreducible polynomials that include such codewords in the corresponding Goppa code is at most

$$\sum_{j=1}^{d-1} \lfloor \frac{j-1}{t} \rfloor (q-1)^j \binom{n}{j} \quad \leq \quad \sum_{j=1}^{d-1} \frac{d}{t} (q-1)^j \binom{n}{j}$$

where, for a given j, the $\binom{n}{j}$ factor comes from the choice of the nonzero coordinates of $\mathbf{c}$, and the $(q-1)^j$ factor from the choice of the values of those coordinates.
Using the fact that the volume $V_q(n, d-1)$ of a ball of radius $d-1$ is given by $V_q(n, d-1) = \sum_{j=0}^{d-1} (q-1)^j \binom{n}{j}$, we get that the number of polynomials to exclude is less than

$$\frac{d}{t} V_q(n, d-1)$$

On the other hand, we know from [7] that the number of irreducible polynomials of degree t over $\mathbb{F}_{q^m}$ exceeds

$$\frac{1}{t} q^{mt} (1 - q^{-(1/2)mt+m})$$

Hence a sufficient condition for the existence of Goppa codes with minimum distance at least d is

$$\frac{d}{t} V_q(n, d-1) < \frac{1}{t} q^{mt} (1 - q^{-(1/2)mt+m}) \tag{1}$$

Taking base-q logarithms, dividing by n and taking the limit as $n \to \infty$, we can write this condition as

$$H_q(\delta) + o(1) < \frac{mt}{n} + o(1) \tag{2}$$

where δ is the required minimum distance, i.e. $\delta = \lim_{n \to \infty} d/n$.
We can see that by our choice of n (hence m) and t, we can make $\frac{mt}{n}$ very

close to $H_q(\delta)$, say $\frac{mt}{n} = H_q(\delta) + \epsilon$. For these values of the parameters, we know that there exists a Goppa code $\Gamma(L, G)$ since condition 2 is satisfied. Also, by a property of Goppa codes, the rate r of this code will satisfy

$$\begin{aligned} r &\geq 1 - \frac{mt}{n} \\ &> 1 - H_q(\delta) - \epsilon \end{aligned}$$

Hence the resulting Goppa code achieves the GV bound. □

4.2 Applicability of the Argument to the Construction in [1]

If we try to apply the proof in [7] to the specific subclass of Goppa codes described in [1], setting $q = 2$, the difference lies in the total number of irreducible polynomials of degree t. Since now the Goppa polynomial has to be of the form

$$G(x) = x^t + A$$

with A being a t-th power in $\mathbb{F}_{2^m}$, the number of such polynomials is limited by the number of possible choices of A.
Let α be a generator of $\mathbb{F}_{2^m}$. Since we must have $A = \gamma^t$ for some element γ of $\mathbb{F}_{2^m}$, and since any element of $\mathbb{F}_{2^m}$ can be written as a power of α, we see that A has to be one of

$$(\alpha^1)^t, (\alpha^2)^t, \ldots, (\alpha^{\frac{2^m-1}{t}})^t.$$

So there are only $\frac{2^m-1}{t}$ possible choices of A for a given t, and therefore only that many irreducible polynomials of degree t to choose from.
Hence 1 becomes

$$\frac{d}{t} V_2(n, d-1) < \frac{2^m - 1}{t} \tag{3}$$

Taking base-2 logarithms, dividing both sides by n and letting $n \to \infty$, the condition becomes

$$H(\delta) + o(1) < \frac{m}{n} + o(1) \tag{4}$$

But we know that $n = |L| = 2^m$, hence 4 is equivalent to

$$H(\delta) + o(1) < \frac{\log n}{n} + o(1) \tag{5}$$

which cannot be achieved for constant δ as the right-hand side goes to 0 for large n. □

5 Implementation of the family of Goppa codes described in [1]

5.1 Implementation details

We used the PARI/GP computer algebra system to implement the class of Goppa codes described in [1] for various choices of the parameters m and t. The idea was to calculate the resulting code dimension in each case. This would allow us to see if there exist some codes within this family which achieve the GV bound, if the $n - mt$ lower bound is tight enough for various parameter settings, and if there are certain patterns in the observed results that would allow us to formulate plausible conjectures about this family of codes.

We start the implementation by an appropriate choice of m, such that $2^m - 1$ is not prime (so that there are meaningful choices for the value of t as a divisor of $2^m - 1$). For this choice of m, we generate an irreducible polynomial f of degree m, and generate the elements of the field $\mathbb{F}_{2^m}$, represented as polynomials modulo f, with coefficients in $\mathbb{F}_2$.

Having set up the field, we can now vary t over all meaningful values among the divisors of $2^m - 1$. A meaningful value is one that does not yield a relative minimum distance δ which is too small (less than around 0.01) or too large (more than around 0.5), where the relative minimum distance is given by $\delta = \frac{2t+1}{n}$.

Given the choice of t, we set the Goppa polynomial to be

$$G(x) = x^t + 1^t = x^t + 1.$$

By the property of this family of Goppa codes explained in section 3.1, different choices of α in the expression $G(x) = x^t + \alpha^t$ will lead to equivalent codes, hence codes with the same dimension. We can then simply fix $\alpha = 1$ and calculate the corresponding dimension.

We then find the elements of L by testing each element of the field for being a root of $G(x)$. Note that since $G(x)$ is separable, it has t roots in $\mathbb{F}_{2^m}$ (namely, for our choice of $G(x)$, these are the tth roots of unity in $\mathbb{F}_{2^m}$). Hence the code block length is given by $n = 2^m - t$. We can now compute a parity check matrix $\mathbb{F}_{2^m}$ for the code. The formal derivation of this matrix is explained in section 5.2. We then transform this matrix into a matrix over $\mathbb{F}_2$ by simply replacing each entry of the matrix, which is an element of $\mathbb{F}_{2^m}$,

by the corresponding m-component column vector with components from $\mathbb{F}_2$ (this can be done since $\mathbb{F}_{2^m}$ is isomorphic to $\mathbb{F}_2^m$). Then computing the rank of this matrix will give us the code dimension and hence the rate.
Section 5.3 shows the obtained code dimension for various choice of the parameters, along with the GV bound calculated for these parameters, given by $1 - H(\delta)$, where H denotes the binary entropy function.

5.2 The Parity Check Matrix

A parity check matrix for a Goppa code in the family of [1], with Goppa polynomial

$$G(x) = x^t + \alpha^t$$

can be derived as follows.
The constraints on the codewords are given by the equation

$$\sum_{i=1}^{n} \frac{c_i}{x - \gamma_i} \equiv 0 \pmod{G(x)}.$$

We can derive a parity check matrix for the code from these conditions. Since no γ_i is a root of $G(x)$, $x - \gamma_i$ is invertible $\pmod{G(x)}$ and, from [4], its inverse is

$$-\frac{G(x) - G(\gamma_i)}{x - \gamma_i} G(\gamma_i)^{-1}.$$

In our case, this is

$$-\frac{x^t - \gamma_i^t}{x - \gamma_i} \frac{1}{\gamma_i^t + \alpha^t}.$$

Hence the conditions become

$$\sum_{i=1}^{n} c_i \frac{x^t - \gamma_i^t}{x - \gamma_i} \frac{1}{\gamma_i^t + \alpha^t} \equiv 0 \pmod{G(x)}$$

i.e.

$$\sum_{i=1}^{n} c_i (x^{t-1} + \gamma_i x^{t-2} + \gamma_i^2 x^{t-3} + \ldots + \gamma_i t - 1) \frac{1}{\gamma_i^t + \alpha^t} = 0$$

This is an identity, not a congruence modulo $G(x)$, since the polynomial on the left-hand-side is of degree less than t.

Hence we can derive a parity check matrix for the code by equating the coefficients of the polynomial

$$(x^{t-1} + \gamma_i x^{t-2} + \gamma_i^2 x^{t-3} + \ldots + \gamma_i^{t-1})$$

to zero for each i. We obtain a parity check matrix

$$H = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \gamma_1 & \gamma_2 & \cdots & \gamma_n \\ \gamma_1^2 & \gamma_2^2 & \cdots & \gamma_n^2 \\ & & \vdots & \\ \gamma_1^{t-1} & \gamma_2^{t-1} & \cdots & \gamma_n^{t-1} \end{bmatrix} \begin{bmatrix} \frac{1}{\gamma_1^t + \alpha^t} & & & \\ & \frac{1}{\gamma_2^t + \alpha^t} & & \\ & & \vdots & \\ & & & \frac{1}{\gamma_n^t + \alpha^t} \end{bmatrix}$$

Note that more generally, a parity check matrix for any Goppa code can be derived similarly.

5.3 Results

The following table displays the output of the program for various settings of the input parameters.
For each setting of m and t, the corresponding n is calculated using $n = 2^m - t$. The dimension column corresponds to the main output of the program, and can be compared to the lower bound given by $n - mt$. The rate is given by $\frac{k}{n}$ where k denotes the code dimension. It can be compared to the GV bound given by $1 - H(\delta)$.

m	**t**	**n**	**dimension**	**lower bound**	**rate**	δ	$1 - H(\delta)$
4	3	13	2	1	0.154	0.538	0.00427
6	3	61	43	43	0.705	0.115	0.486
6	7	57	17	15	0.298	0.263	0.169
6	9	55	16	1	0.291	0.345	0.0701
8	3	253	229	229	0.905	0.0277	0.817
8	5	251	211	211	0.841	0.0438	0.740
8	15	241	124	121	0.515	0.129	0.446
8	17	239	123	103	0.515	0.146	0.399
8	51	205	2	-203	0.00976	0.502	0.0000172
9	7	505	442	442	0.875	0.0297	0.807
9	73	439	58	-218	0.132	0.335	0.0802
10	3	1021	991	991	0.971	0.00686	0.941
10	11	1013	903	903	0.891	0.0227	0.844
10	31	993	687	683	0.692	0.0634	0.659
10	33	991	686	661	0.692	0.0676	0.643
10	93	931	105	1	0.113	0.201	0.276
11	23	2025	1772	1772	0.875	0.0232	0.841

We notice that the tested codes achieve better than the GV bound in most cases. This could be due to the fact that our chosen values for the field size are still too small to exhibit asymptotic behaviour. However, the limitations of GP/PARI did not allow us to consider larger values of m.
Also note that the lower bound $n - mt$ is tight for many cases. However, for the "typical" cases (which correspond to "realistic" values of r) such as $n = 439$ and $r = 0.132$, or $n = 931$ and $r = 0.113$, the lower bound is very loose, as expected.

6 Other Approaches

Other possible approaches to finding a single good code, given a polynomial family of codes of which a large fraction achieves the GV bound, involve investigating properties of the covering radius of codes, and possible improvements to the $n - mt$ lower bound on the code dimension. In this section, we concisely describe these approaches.

6.1 Upper Bounding the Covering Radius

Definition 3 *Given a code C over $\mathbb{F}_2^n$, the covering radius ν of C is defined as the minimum radius such that the set of balls of radius ν centered around the codewords covers the whole $\mathbb{F}_2^n$ space. Formally,*

$$\nu = \min_l (\bigcup_{c \in C} B(c, l) = \mathbb{F}_2^n).$$

Intuitively, the smaller the covering radius, the better rate we have. More formally,

Fact

A code C with minimum distance δ achieves the GV bound if the covering radius is such that $\nu \leq \delta$.

Proof

Denote the rate of the code by r. By definition of the covering radius ν, we have

$$2^{nr} Vol(B_\nu) \geq 2^n.$$

But

$$\begin{aligned} Vol(B_\nu) &= \sum_{i=0}^{\nu} \binom{n}{i} \\ &\simeq 2^{nH(\nu/n)} \end{aligned}$$

Hence we get

$$nr \geq n(1 - H(\nu/n)),$$

therefore $r \geq 1 - H(\delta)$ if $\nu \leq \delta n$. □

In [3], the authors refine the existing upper bound for the covering radius of Goppa codes over $\mathbb{F}_{2^m}$, and prove that under a condition involving t, m and the number N of zeroes of $G(x)$ in $\mathbb{F}_{2^m}$, ν is upper bounded by $2t + 1$. Specifically,

Theorem 3 *[3] The covering radius of $\Gamma(L, G)$ is less than or equal to $2+1$ if*

$$2^m \geq 4(\mu + t - 1)^{4t+2} \left(\frac{2}{1 + \sqrt{1 - \frac{1}{(\mu+t-1)^{4t}}}} \right)^{4t+2},$$

where $\mu = \frac{N}{2^{m/2+1}}$.

It would thus be interesting to investigate properties of the rate of such Goppa codes given the upper bound on their covering radius. In this project we did not investigate further along this line.

6.2 Improving the Dimension Lower Bound

In [6], the author improved the known lower bound for the dimension of Goppa codes $k \geq n - mt$.
In a handwaving manner, the argument can be understood as follows. Let $\Gamma(L, G)$ be a Goppa code defined over $\mathbb{F}_{q^m}$. The parity check matrix over $\mathbb{F}_{2^m}$ as derived in section 5.2 has t rows. The corresponding matrix over $\mathbb{F}_q$ has mt rows, hence it has rank at most mt. Simply using the formula $k = n - \text{rank(PCM)}$ yields $k \geq n - mt$.
In our implementation, we transformed the parity check matrix over $\mathbb{F}_q$ by replacing each entry of the matrix by the corresponding m-component column vector with components from $\mathbb{F}_q$, which is the most practical way to get the matrix over $\mathbb{F}_q$. But we can compute the rank of this matrix in another way. If we denote the rows of the original matrix over $\mathbb{F}_{q^m}$ by $\{r_1, \ldots, r_t\}$, we can create a new matrix over $\mathbb{F}_{q^m}$ by appending to the original matrix t rows $\sigma(r_1), \ldots, \sigma(r_t)$, then t rows $\sigma^2(r_1), \ldots, \sigma^2(r_t)$, ..., then t rows $\sigma^{m-1}(r_1), \ldots, \sigma^{m-1}(r_t)$. Here σ denotes the Frobenius map $\sigma : \alpha \to \alpha^q$, and for a row vector $r_i = (r_{i1}, \ldots, r_{in})$, $\sigma(r_i) = (\sigma(r_{i1}), \ldots, \sigma(r_{in}))$. The obtained matrix can be shown to be the parity check matrix of a code over the extension field $\mathbb{F}_{q^m}$ which has another parity check matrix with elements from $\mathbb{F}_q$ that is also a parity check matrix of the subfield subcode over $\mathbb{F}_q$. Hence it has the same rank as the parity check matrix of the subfield subcode over $\mathbb{F}_q$.
The matrix obtained in this way will have dependencies if some of its rows were already row vectors over the base field $\mathbb{F}_q$. Hence if we denote the number of these rows by l, the dimension lower bound can be refined to become

$$k \geq n - mt + (m-1)l,$$

since the $k \geq n - mt$ bound counts these rows $m - 1$ times more than necessary. This is intuitively the end result of [6], but derived in a formal and precise manner.
The lower bound can be further refined using the same idea: there might be

rows of the original matrix which are row vectors over the field $\mathbb{F}_{q^{m'}}$, where $m' \mid m$. These rows are be counted $\frac{m}{m'} - 1$ times more than necessary in the original lower bound.
An important question is whether it is worth improving the lower bound further than what was done in [6]. The author states that his improved lower bound "coincides in many cases with the true dimension of the code". One aim of our implementation was to verify the tightness of the $n - mt$ lower bound and we were surprised to observe that it is actually tight in many cases. Again, this is possibly due to inaccurate results since our numbers do not reflect the asymptotic behaviour of the codes.

7 Conclusion and Future Work

We have tried to investigate various approaches to determining the rate of Goppa codes in general, and that of codes of the class described in [1] in particular. The relevance of this work is that if we can determine some properties of the rate of the codes in [1], we would be able to determine which (if any) of these codes achieve the GV bound. Such a result, combined with the construction of [2] which yields a polynomial family of codes of which a large fraction achieve the GV bound, could make it possible to find a single code known to be asymptotically good. We did not obtain results about the asymptotic behaviour of the codes in [1] within the scope of this project, but possible continuations of our work include deeper investigation of the approaches outlined in section 6, and the implementation of the codes in [1] over larger fields in order to be able to detect patterns of the code dimensions as the block length grows large, which would allow us to formulate conjectures regarding the asymptotic behaviour of these codes.

Acknowledgements

I would like to thank Prof. Shokrollahi for introducing me to this topic and for his continuous guidance and advice throughout the project, and Mahdi Cheraghchi for his constant availability, technical advice and patience. In general, I would like to thank the members of the ALGO lab for their friendliness and enjoyable atmosphere.

References

[1] S. Bezzateev and N. Shekhunova. *Subclass of Binary Goppa Codes with Minimal Distance Equal to the Design Distance*. IEEE Transactions on Information Theory, vol.41 no.2, March 1995.

[2] M. Cheraghchi, A. Shokrollahi and A. Wigderson. *Computational Hardness and Explicit Constructions of Error Correcting Codes*. Fourty-Fourth Annual Allerton Conference on Communication, Control and Computing, 2006.

[3] F. Levy-dit-Vehel and S. Litsyn. *Parameters of Goppa codes revisited*. IEEE Transactions on Information Theory, vol.43 no.6, November 1997.

[4] F.J. MacWilliams and N.J.A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland Mathematical Library 1977.

[5] The PARI Team. Laboratoire A2X, Université Bordeaux I (France). http://pari.math.u-bordeaux.fr/

[6] H. Stichtenoth. *On the Dimension of Subfield Subcodes*. IEEE Transactions on Information Theory, vol.36 no.1,January 1990.

[7] J.H. van Lint. *Introduction to Coding Theory, Third Edition*. Springer 1999.