Heuristic Construction Of „Good“ Error-Correcting Linear Codes

Johannes Zwanzger

University of Bayreuth

Magdeburg
November 17th, 2007

Introduction I

- A linear code C over $\mathbb{F}_q$ of *blocklength* n and *dimension* k is a $k-$dimensional subspace of $\mathbb{F}_q^n$

Introduction I

- A linear code C over $\mathbb{F}_q$ of *blocklength* n and *dimension* k is a $k-$dimensional subspace of $\mathbb{F}_q^n$

- elements of C are called *codewords* and written as *row vectors*

Introduction I

- A linear code C over $\mathbb{F}_q$ of *blocklength* n and *dimension* k is a $k-$dimensional subspace of $\mathbb{F}_q^n$

- elements of C are called *codewords* and written as *row vectors*

- weight $wt(c)$ of $c \in C$: number of nonzero components in c

Introduction I

- A linear code C over $\mathbb{F}_q$ of *blocklength* n and *dimension* k is a $k-$dimensional subspace of $\mathbb{F}_q^n$

- elements of C are called *codewords* and written as *row vectors*

- weight $wt(c)$ of $c \in C$: number of nonzero components in c

- *Hamming distance* between c, $c' \in C$: $dist(c, c') := wt(c - c')$

Introduction II

- The *minimum distance* of C is the minimum Hamming distance between any two *different* codewords of C.

Introduction II

- The *minimum distance* of C is the minimum Hamming distance between any two *different* codewords of C.

- In the linear case the minimum distance equals the minimum weight over all nonzero codewords in C

Introduction II

- The *minimum distance* of C is the minimum Hamming distance between any two *different* codewords of C.

- In the linear case the minimum distance equals the minimum weight over all nonzero codewords in C

- C has minimum distance $d \Rightarrow$ up to $\lfloor \frac{d-1}{2} \rfloor$ errors can be corrected

Introduction II

- The *minimum distance* of C is the minimum Hamming distance between any two *different* codewords of C.

- In the linear case the minimum distance equals the minimum weight over all nonzero codewords in C

- C has minimum distance $d \Rightarrow$ up to $\lfloor \frac{d-1}{2} \rfloor$ errors can be corrected

- C can be described by a *generator matrix* $\Gamma \in \mathbb{F}_q^{k \times n}$, whose rows form a basis of C

Example

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

generates a code with parameters $n = 6,\ k = 3,\ d = 3$ over $\mathbb{F}_3$.

Observations:

Example

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

generates a code with parameters $n = 6,\ k = 3,\ d = 3$ over $\mathbb{F}_3$.

Observations:

- codewords arise via multiplication of $\mathbb{F}_q^k$ with Γ: $C = \{v\Gamma : v \in \mathbb{F}_q^k\}$

Example

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

generates a code with parameters $n = 6$, $k = 3$, $d = 3$ over $\mathbb{F}_3$.

Observations:

- codewords arise via multiplication of $\mathbb{F}_q^k$ with Γ: $C = \{v\Gamma : v \in \mathbb{F}_q^k\}$
- the $j-$th component of a codeword $c = (c_1, c_2, \ldots, c_n) = v\Gamma$ only depends from the $j-$th column of Γ: $c_j = v\Gamma_{*j}$

Example

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

generates a code with parameters $n = 6,\ k = 3,\ d = 3$ over $\mathbb{F}_3$.

Observations:

- codewords arise via multiplication of $\mathbb{F}_q^k$ with Γ: $C = \{v\Gamma : v \in \mathbb{F}_q^k\}$
- the $j-$th component of a codeword $c = (c_1, c_2, \ldots, c_n) = v\Gamma$ only depends from the $j-$th column of Γ: $c_j = v\Gamma_{*j}$
- multiplying a column of Γ with $\lambda \in \mathbb{F}_q^*$ has no influence on the weights

Example

$$\Gamma = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 & 2 \end{pmatrix}$$

generates a code with parameters $n = 6$, $k = 3$, $d = 3$ over $\mathbb{F}_3$.

Observations:

- codewords arise via multiplication of $\mathbb{F}_q^k$ with Γ: $C = \{v\Gamma : v \in \mathbb{F}_q^k\}$
- the $j-$th component of a codeword $c = (c_1, c_2, \ldots, c_n) = v\Gamma$ only depends from the $j-$th column of Γ: $c_j = v\Gamma_{*j}$
- multiplying a column of Γ with $\lambda \in \mathbb{F}_q^*$ has no influence on the weights
- $\lambda \in \mathbb{F}_q^*,\ v \in \mathbb{F}_q^k \Rightarrow wt(v\Gamma) = wt(\lambda v\Gamma)$

Theorem

Let $t := \frac{q^k-1}{q-1}$ *and* $\Omega_{k,q} = \left(\omega_{\langle v\rangle,\langle u\rangle}\right) \in \mathbb{N}^{t\times t}$ *be the matrix (well-)defined by*

$$\omega_{\langle v\rangle,\langle u\rangle} := \begin{cases} 0 & \text{if } \langle v,u\rangle_{\mathbb{F}_q} = 0 \\ 1 & \text{else} \end{cases}$$

for $\langle v\rangle$, $\langle u\rangle \in PPG(k-1,q)$ *with* v, $u \in \mathbb{F}_q^{k*}$. *Then:*

Existence of a nonredundant linear $(n,k,d,q)-$*code*

$$\Updownarrow$$

Existence of a multiset $\{\langle u_1\rangle, \langle u_2\rangle, \ldots, \langle u_n\rangle\} \subset PPG(k-1,q)$ *so that*

$$\sum_{i=1}^{n} \omega_{\langle v\rangle,\langle u_i\rangle} \geq d$$

is true for each $\langle v\rangle \in PPG(k-1,q)$.

Example

	0 **0** **1**	**0** **1** **0**	0 1 1	**0** **1** **2**	**1** **0** **0**	1 0 1	1 0 2	1 1 0	**1** **1** **1**	1 1 2	1 2 0	1 2 1	**1** **2** **2**	$\sum$
0 0 1	**1**	**0**	1	**1**	**0**	1	1	0	**1**	1	0	1	**1**	**4**
0 1 0	**0**	**1**	1	**1**	**0**	0	0	1	**1**	1	1	1	**1**	**4**
0 1 1	**1**	**1**	1	**0**	**0**	1	1	1	**1**	0	1	0	**1**	**4**
0 1 2	**1**	**1**	0	**1**	**0**	1	1	1	**0**	1	1	1	**0**	**3**
1 0 0	**0**	**0**	0	**0**	**1**	1	1	1	**1**	1	1	1	**1**	**3**
1 0 1	**1**	**0**	1	**1**	**1**	1	0	1	**1**	0	1	1	**0**	**4**
1 0 2	**1**	**0**	1	**1**	**1**	0	1	1	**0**	1	1	0	**1**	**4**
1 1 0	**0**	**1**	1	**1**	**1**	1	1	1	**1**	1	0	0	**0**	**4**
1 1 1	**1**	**1**	1	**0**	**1**	1	0	1	**0**	1	0	1	**1**	**4**
1 1 2	**1**	**1**	0	**1**	**1**	0	1	1	**1**	0	0	1	**1**	**6**
1 2 0	**0**	**1**	1	**1**	**1**	1	1	0	**0**	0	1	1	**1**	**4**
1 2 1	**1**	**1**	0	**1**	**1**	1	0	0	**1**	1	1	0	**1**	**6**
1 2 2	**1**	**1**	1	**0**	**1**	0	1	0	**1**	1	1	1	**0**	**4**

Heuristic Algorithm

Input:

Heuristic Algorithm

Input:

- code parameters n, k, d, q

Heuristic Algorithm

Input:

- code parameters n, k, d, q
- initial column multiset X_0

Heuristic Algorithm

Input:

- code parameters n, k, d, q
- initial column multiset X_0
- evaluation function $eval : \mathcal{P}(PPG(k-1, q)) \to \mathbb{R}$

Heuristic Algorithm

Input:

- code parameters n, k, d, q
- initial column multiset X_0
- evaluation function $eval : \mathcal{P}(PPG(k-1,q)) \to \mathbb{R}$

Output: column multiset X for a $(n,k,d,q)-$code or FAILED

Heuristic Algorithm

Input:

- code parameters n, k, d, q
- initial column multiset X_0
- evaluation function $eval : \mathcal{P}(PPG(k-1, q)) \to \mathbb{R}$

Output: column multiset X for a $(n, k, d, q)-$code or FAILED

(1) Set $X \leftarrow X_0$.
(2) For each $x \in PPG(k-1, q)$, compute $eval(X \cup \{x\})$.
(3) Choose a point x^* maximizing the value in (2). Set $X \leftarrow X \cup \{x^*\}$.
(4) If $|X| < n$, go to (2).
(5) If $d_X \geq d$, return X; otherwise, return FAILED.

Evaluation Function I

Observation:

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'
- $\leadsto p := \frac{q^{k-1}}{t} = \frac{(q-1)(q^{k-1})}{q^k-1}$

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'
- $\rightsquigarrow p := \frac{q^{k-1}}{t} = \frac{(q-1)(q^{k-1})}{q^k-1}$

Consequences:

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'
- $\rightsquigarrow p := \frac{q^{k-1}}{t} = \frac{(q-1)(q^{k-1})}{q^k-1}$

Consequences:

$\langle v \rangle :=$ arbitrary row index of $\Omega_{k,q}$
$X' :=$ random multiset of m column indizes $:= \{\langle u_1 \rangle, \langle u_2 \rangle, \ldots, \langle u_m \rangle\}$
$j \leq m \in \mathbb{N}$

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'
- $\rightsquigarrow p := \frac{q^{k-1}}{t} = \frac{(q-1)(q^{k-1})}{q^k-1}$

Consequences:

$\langle v \rangle :=$ arbitrary row index of $\Omega_{k,q}$
$X' :=$ random multiset of m column indizes $:= \{\langle u_1 \rangle, \langle u_2 \rangle, \ldots, \langle u_m \rangle\}$
$j \leq m \in \mathbb{N}$

- $\mathrm{Prob}\left(\sum_{l=1}^{m} \omega_{\langle v \rangle, \langle u_l \rangle} = j\right) = p^j(1-p)^{m-j}\binom{m}{j} =: r_{m,j}$

Evaluation Function I

Observation:

- For each row of $\Omega_{k,q}$ there are exactly q^{k-1} columns with a '1'
- $\rightsquigarrow p := \frac{q^{k-1}}{t} = \frac{(q-1)(q^{k-1})}{q^k-1}$

Consequences:

$\langle v \rangle :=$ arbitrary row index of $\Omega_{k,q}$
$X' :=$ random multiset of m column indizes $:= \{\langle u_1 \rangle, \langle u_2 \rangle, \ldots, \langle u_m \rangle\}$
$j \leq m \in \mathbb{N}$

- $\mathrm{Prob}\left(\sum_{l=1}^{m} \omega_{\langle v \rangle, \langle u_l \rangle} = j\right) = p^j (1-p)^{m-j} \binom{m}{j} =: r_{m,j}$
- $\mathrm{Prob}\left(\sum_{l=1}^{m} \omega_{\langle v \rangle, \langle u_l \rangle} \geq j\right) = \sum_{l=j}^{m} r_{m,l} =: s_{m,j}$

Evaluation Function II

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1, q)$ with $|X| := n' < n$.

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1, q)$ with $|X| := n' < n$.

Goal: Find a 'sensible' evaluation for X

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1, q)$ with $|X| := n' < n$.

Goal: Find a 'sensible' evaluation for X

Approach:

$$\begin{array}{rcl} R_i & := & \text{set of rows of } \Omega_{k,q} \text{ where sum 'over X' equals } i \\ a_i & := & |R_i| \\ Y & := & \text{random multisubset of } PPG(k-1,q) \text{ with } |Y| = n - n' \end{array}$$

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1,q)$ with $|X| := n' < n$.

Goal: Find a 'sensible' evaluation for X

Approach:

$$
\begin{array}{lll}
R_i & := & \text{set of rows of } \Omega_{k,q} \text{ where sum 'over X' equals } i \\
a_i & := & |R_i| \\
Y & := & \text{random multisubset of } PPG(k-1,q) \text{ with } |Y| = n-n'
\end{array}
$$

- $X \cup Y$ multiset for $(n,k,d,q)-$code $\Leftrightarrow$
  $\forall i \leq d-1$: for each row in R_i the row sum 'over Y' is $\geq d-i$

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1,q)$ with $|X| := n' < n$.

Goal: Find a 'sensible' evaluation for X

Approach:

$$\begin{array}{rcl} R_i & := & \text{set of rows of } \Omega_{k,q} \text{ where sum 'over X' equals } i \\ a_i & := & |R_i| \\ Y & := & \text{random multisubset of } PPG(k-1,q) \text{ with } |Y| = n - n' \end{array}$$

- $X \cup Y$ multiset for $(n,k,d,q)-$code $\Leftrightarrow$
  $\forall i \leq d-1$: for each row in R_i the row sum 'over Y' is $\geq d-i$
- Probability for a single row is $s_{n-n',d-i}$

Evaluation Function II

Let n, k, d, q be fixed.
Consider $X \subset PPG(k-1,q)$ with $|X| := n' < n$.

Goal: Find a 'sensible' evaluation for X

Approach:

$$
\begin{array}{rcl}
R_i & := & \text{set of rows of } \Omega_{k,q} \text{ where sum 'over X' equals } i \\
a_i & := & |R_i| \\
Y & := & \text{random multisubset of } PPG(k-1,q) \text{ with } |Y| = n-n'
\end{array}
$$

- $X \cup Y$ multiset for $(n,k,d,q)-$code $\Leftrightarrow$
  $\forall i \leq d-1$: for each row in R_i the row sum 'over Y' is $\geq d-i$
- Probability for a single row is $s_{n-n',d-i}$
- Assumption of stochastic indipendence $\rightsquigarrow$...

Definition

$$eval(X) := \prod_{i=1}^{d-1} s_{n-n',d-i}^{a_i}$$

Definition

$$eval(X) := \prod_{i=1}^{d-1} s_{n-n',d-i}^{a_i}$$

Example

$$\Gamma_1 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Assume we want to construct a linear $(10, 5, 4, 2)$-code. What is the evaluation of Γ_1?

Definition

$$eval(X) := \prod_{i=1}^{d-1} s_{n-n',d-i}^{a_i}$$

Example

$$\Gamma_1 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Assume we want to construct a linear $(10, 5, 4, 2)$-code. What is the evaluation of Γ_1?

- weight-polynomial is $W_{C_1}(x) = x^0 + 15x^2 + 15x^4 + x^6$

Definition

$$eval(X) := \prod_{i=1}^{d-1} s_{n-n',d-i}^{a_i}$$

Example

$$\Gamma_1 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

Assume we want to construct a linear $(10, 5, 4, 2)$-code. What is the evaluation of Γ_1?

- weight-polynomial is $W_{C_1}(x) = x^0 + 15x^2 + 15x^4 + x^6$
- here: $p = \frac{16}{31} \Rightarrow s_{4,2} = \frac{656896}{923521} \Rightarrow eval(\Gamma_1) = \left(\frac{656896}{923521}\right)^{15} \approx 6.04 \cdot 10^{-3}$

Example

$$\Gamma_2 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

And what is $eval(\Gamma_2)$?

Example

$$\Gamma_2 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

And what is $eval(\Gamma_2)$?

- weight-polynomial is $W_{C_2}(x) = x^0 + 1x^1 + 10x^2 + 10x^3 + 5x^4 + 5x^5$

Example

$$\Gamma_2 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

And what is $eval(\Gamma_2)$?

- weight-polynomial is $W_{C_2}(x) = x^0 + 1x^1 + 10x^2 + 10x^3 + 5x^4 + 5x^5$
- $eval(\Gamma_2) = s^1_{4,3} \cdot s^{10}_{4,2} \cdot s^{10}_{4,1} \approx 6.36 \cdot 10^{-3}$

Example

$$\Gamma_2 := \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

And what is $eval(\Gamma_2)$?

- weight-polynomial is $W_{C_2}(x) = x^0 + 1x^1 + 10x^2 + 10x^3 + 5x^4 + 5x^5$
- $eval(\Gamma_2) = s_{4,3}^{1} \cdot s_{4,2}^{10} \cdot s_{4,1}^{10} \approx 6.36 \cdot 10^{-3}$
- $\Rightarrow$ although $mindist(C_1) > mindist(C_2)$, Γ_2 is preferred over Γ_1

Results

$q = 2,\ k = 10$:

n	181	186
d	86	**88**

$q = 5,\ k = 6$:

n	47
d	32

$q = 5,\ k = 7$:

n	19	33	37	44	52
d	**10**	20	23	28	34

$q = 7,\ k = 4$:

n	77
d	63

$q = 7,\ k = 5$:

n	56	62	68
d	43	48	53

$q = 7,\ k = 6$:

n	62	67	73	77
d	46	50	55	58

$q = 9,\ k = 5$

n	33
d	25