TOKYO J. MATH.
VOL. 26, NO. 1, 2003

On Some Properties of the Hyper-Kloosterman Codes

Koji CHINEN

Osaka Institute of Technology

(Communicated by K. Nakamula)

Abstract. The hyper-Kloosterman code was first defined as a trace code by Chinen-Hiramatsu [1]. In this article, two basic parameters of it, the minimum distance and the dimension are estimated. Analysis of the dimension shows that it is one of few examples of trace codes, of which the dimensions do not reduce when taking the trace, and are determined explicitly. It is also shown that the hyper-Kloosterman code can be realized as a quasi-cyclic code. It implies a method of explicit construction of quasi-cyclic codes of a new type.

1. Introduction

Let p be a prime, $r \geq 2$, $m \geq 2$ and $q = p^r$. We denote the finite field with q elements by $\mathbf{F}_q$. In Chinen-Hiramatsu [1], a new linear code $C_m(q)$ is defined as a trace code, and is named "hyper-Kloosterman code":

DEFINITION 1.1. For any integer $m \geq 2$, the hyper-Kloosterman code $C_m(q)$ of degree $m-1$ is defined to be the image of the map

$$\varphi_m : \mathbf{F}_q^m \to \mathbf{F}_p^{(q-1)^{m-1}}$$

given by

$$\varphi_m(\boldsymbol{a}) = \left\{\mathrm{Tr}\,(\boldsymbol{a}, \boldsymbol{x})\right\}_{\boldsymbol{x} \in (\mathbf{F}_q^\times)^{m-1}},$$

where

$$\mathrm{Tr}\,(\boldsymbol{a}, \boldsymbol{x}) = \mathrm{tr}\,(a_1x_1 + a_2x_2 + \cdots a_{m-1}x_{m-1} + a_m(x_1x_2\cdots x_{m-1})^{-1})$$

for $\boldsymbol{a} = (a_1, a_2, \cdots, a_m) \in \mathbf{F}_q^m$ and $\boldsymbol{x} = (x_1, x_2, \cdots, x_{m-1})$, and tr $=\mathrm{trace}_{\mathbf{F}_q/\mathbf{F}_p}$.

The symbol $\{\quad\}_{\boldsymbol{x}\in(\mathbf{F}_q^\times)^{m-1}}$ represents a vector obtained by letting $\boldsymbol{x}$ run through the set $(\mathbf{F}_q^\times)^{m-1}$ (such a notation is often used in the literature on the trace codes). The code $C_m(q)$ is a generalization of the Kloosterman code, which is the dual of the Melas code. The Kloosterman code has been investigated by many authors. See for example, Hiramatsu [2], Lachaud [3] and Wolfmann [10]. The original paper by C. M. Melas is [6].

In our previous paper [1], two properties of $C_m(q)$ are deduced: one is a certain uniform distribution property of the Hamming weights, and the other is a divisibility property of them.

Received November 20, 2001; revised June 17, 2002
1991 Mathematics Subject Classification. 11T71, 11T23, 94B40.

In this article, we deduce some more properties of $C_m(q)$. In Section 2, we estimate the minimum distance $d(m, q)$ of $C_m(q)$. The Hamming weight of the codeword in $C_m(q)$ can be expressed by the hyper-Kloosterman sums

$$K_m(\boldsymbol{a}; q) = \sum_{\boldsymbol{x} \in (G^{\times})^{m-1}} e(\mathrm{Tr}\,(\boldsymbol{a}, \boldsymbol{x})) \quad (e(x) = e^{2\pi i x/p}) .$$

In some cases $K_m(\boldsymbol{a}; q)$ becomes trivial (see (2.1)), but in other cases (see (2.2)), we need the Deligne bound

$$|K_m(\boldsymbol{a}; q)| \leq m q^{\frac{m-1}{2}} \tag{1.1}$$

which is valid for those $K_m(\boldsymbol{a}; q)$'s with $\boldsymbol{a} \in (\mathbf{F}_q^{\times})^m$, to estimate the Hamming weight. The estimate (1.1) is obtained in [7, p. 219]. Indeed, putting $a = a_1 a_2 \cdots a_m \in \mathbf{F}_q^{\times}$ for $\boldsymbol{a} = (a_1, a_2, \cdots, a_m) \in (\mathbf{F}_q^{\times})^m$, we can see

$$K_m(\boldsymbol{a}; q) = K_m(a; q) := \sum_{\substack{x_i \in \mathbf{F}_q^{\times} \\ x_1 x_2 \cdots x_m = a}} e(\mathrm{tr}\,(x_1 + x_2 + \cdots + x_m)) . \tag{1.2}$$

Consider a hypersurface in the affine space over the algebraic closure of $\mathbf{F}_q$, defined by

$$V_a = \{(x_1, x_2, \cdots, x_m) \in \mathbf{A}^m \ ; \ x_1 x_2 \cdots x_m = a\} .$$

Then there exist m complex numbers α_1, α_2, $\cdots$, α_m with $|\alpha_j| \leq q^{(m-1)/2}$ such that $K_m(a; q) = (-1)^{m-1}(\alpha_1 + \alpha_2 + \cdots + \alpha_m)$ (see [7, p. 221]). Thus we have (1.1).

We also need numerical calculation of $K_m(\boldsymbol{a}; q)$'s for small m's and q's.

In Section 3, we are interested in the code $\bar{C}_m(q)$ over $\mathbf{F}_q$, which satisfies $\mathrm{tr}\,\bar{C}_m(q) = C_m(q)$, where $\mathrm{tr}\,C = \{(\mathrm{tr}\,c_1, \mathrm{tr}\,c_2, \cdots, \mathrm{tr}\,c_n) | (c_1, \cdots, c_n) \in C\}$ for a code C over $\mathbf{F}_q$. Especially we construct explicitly a generator matrix $G_{m,q}$ of $\bar{C}_m(q)$, which allows us to know $\dim C_m(q)$. It should be noted that we can find the exact value $\dim C_m(q)$, in spite of the definition of $C_m(q)$ as a trace code: generally speaking, for a code C over $\mathbf{F}_q$, we can say no more than

$$\dim C \leq \dim(\mathrm{tr}\,C) \leq r \cdot \dim C$$

(see MacWilliams-Sloane [5, p. 208] or Stichtenoth [8, p. 222]). Theorem 3.1 shows that the space $\bar{C}_m(q)$ does not "collapse" when it is mapped by the trace function to $C_m(q)$, due to the special form of the matrix $G_{m,q}$.

From the observation of $G_{m,q}$, we also notice that $C_m(q)$ can be realized as a quasi-cyclic code. Moreover, this fact implies a new method of explicit construction of quasi-cyclic codes by using exponential sums of several variables. The quasi-cyclic property is discussed in Section 4.

In Appendix, we provide a useful, efficient algorithm of calculating exponential sums over finite fields. This algorithm is used in deducing the results of Section 2, and is applicable to various other problems of this kind.

ACKNOWLEDGEMENT. The author would like to express his sincere gratitude to Professor T. Hiramatsu for introducing him to this subject and encouraging him throughout.

2. Minimum distance of $C_m(q)$

We consider the minimum distance of $C_m(q)$. In this section we restrict ourselves to the case $(p-1)|m$. Under this condition, the Hamming weight of the codeword $\varphi_m(\boldsymbol{a}) \in C_m(q)$ $(\boldsymbol{a} = (a_1, a_2, \cdots, a_m) \in \mathbf{F}_q^m)$ takes one of the following values (see Chinen-Hiramatsu [1, Section 2]):

$$w_s = \frac{1}{2}\{(q-1)^{m-1} + (-1)^{m-s+1}(q-1)^{s-1}\}, \quad (1 \le s \le m) \tag{2.1}$$

$$W_a = \frac{1}{2}\{(q-1)^{m-1} - K_m(a;q)\}, \quad (a \in \mathbf{F}_q^\times), \tag{2.2}$$

where $s = \sharp\{i | 1 \le i \le m, a_i = 0\}$ and $K_m(a;q)$ is defined in (1.2). Clearly if $m \ge 3$, we have

$$\min_{\substack{1 \le s \le m \\ w_s \ne 0}} w_s = w_{m-2} = \frac{1}{2}\{(q-1)^{m-1} - (q-1)^{m-3}\}.$$

Therefore the biggest one of $(q-1)^{m-3}$ and $K_m(a;q)$ gives the minimum distance. We know the estimate of $K_m(a;q)$, the Deligne bound (1.1) . So basically, we compare two values $(q-1)^{m-3}$ and $mq^{(m-1)/2}$.

The goal of this section is the following:

THEOREM 2.1. *Suppose $(p-1)|m$ and let $d(m,q)$ be the minimum distance of $C_m(q)$. Then we have the following*:

(i)

$$d(m,q) = \frac{p-1}{p}\{(q-1)^{m-1} - (q-1)^{m-3}\} \quad \textit{if} \begin{cases} m \ge 6, & q = 8, q \ge 16, \\ m \ge 8, & q = 4, 9. \end{cases}$$

(ii)

$$d(m,q) \ge A_{m,q} := \frac{p-1}{p}\{(q-1)^{m-1} - m \cdot q^{\frac{(m-1)}{2}}\}$$

$$\textit{if} \begin{cases} 3 \le m \le 7, & q = 4, \\ 2 \le m \le 6, & q = 9, \\ 2 \le m \le 5, & q = 8, q \ge 16, \end{cases}$$

except for $(m,q) = (2,4)$.

REMARK. We give the explicit value for the missing case $(m,q) = (2,4)$: $d(2,4) = 2$.

First we prove an easy lemma:

LEMMA 2.2. *Suppose $(q-1)^{m_0-3} \ge m_0 q^{(m_0-1)/2}$ for some m_0 and q $(m_0, q \in \mathbf{Z}$, $m_0 \ge 2$, $q \ge 4)$. Then for all $m \ge m_0$ we have $(q-1)^{m-3} \ge mq^{(m-1)/2}$.*

PROOF. It is easy to see

$$\max_{\substack{m \ge 2 \\ q \ge 4}} \frac{m+1}{m} \cdot \frac{\sqrt{q}}{q-1} \le 1.$$

So we have $(m_0+1)\sqrt{q}/m_0(q-1) \leq 1$ for m_0 and q in the assumption. Multiplying this and $m_0 q^{(m_0-1)/2} \leq (q-1)^{m_0-3}$, we get

$$(m_0+1)q^{\frac{m_0}{2}} \leq (q-1)^{m_0-2}\,.$$

This shows that the desired formula holds for $m = m_0+1$. Thus, by induction on m, we obtain the conclusion. □

PROOF OF THEOREM 2.1. (i) First we consider the function

$$f(x) = (x-1)^3 - 6x^{\frac{5}{2}}\,.$$

We can verify $f(x) \geq 0$ if $x \geq 49$. This together with Lemma 2.2 brings

$$(q-1)^{m-3} \geq mq^{\frac{m-1}{2}} \tag{2.3}$$

for all prime powers $q \geq 49$ and all $m \geq 6$. There are 7 prime powers (not primes themselves) less than 49 ($q = 4, 8, 9, 16, 25, 27, 32$). For these, we can see, by Lemma 2.2 again, that (2.3) holds when

$$\begin{aligned} m &\geq 8, \quad q = 8, 9, 16, 25, 27, 32, \\ m &\geq 13, \quad q = 4. \end{aligned}$$

Moreover, computer calculation of $K_m(a;q)$ leads to the conclusion (see the tables at the end of this section).

(ii) When $m = 2$, we have $w_1 = \{(q-1)+1\}/2$ and $w_2 = 0$. Clearly $w_1 \geq \{(q-1)-2\sqrt{q}\}/2$, and all the nonzero Hamming weights are greater than $A_{2,q}$ for $q \neq 2$ (when $q = 4$, $A_{2,q} < 0$ and estimation becomes trivial). When $3 \leq m \leq 5$, the smallest w_s but 0 is w_{m-2}, and we can easily verify $(q-1)^{m-3} \leq mq^{(m-1)/2}$ for $q \geq 4$. Thus we have proved (ii) for $2 \leq m \leq 5$, $q \geq 4$ ($(m,q) \neq (2,4)$). The remaining cases are due to computer calculation of $K_m(a;q)$. □

As this theorem shows, for large values of m's, some of the vectors $\boldsymbol{a} \in (\mathbf{F}_q)^m$ with zero entries give the codewords of the minimum distance, but for small m's, they are given by $\boldsymbol{a} \in (\mathbf{F}_q^\times)^m$. In these cases the minimum distance is expressed by the sums $K_m(a;q)$. The exact value of $K_m(a;q)$ is hard to determine, so there is no other way than to evaluate $d(m,q)$ by the value $A_{m,q}$.

Here are the results of numerical calculation. In the tables below, $\max K_m(a;q)$ means $\max_{a\in\mathbf{F}_q^\times} K_m(a;q)$. Finding the explicit values of $K_m(a;q)$'s requires some non-trivial algorithm, which is explained in Appendix.

$\underline{q = 4}$

m	6	7	8	9	10	11	12
$(q-1)^{m-3}$	27	81	243	729	2187	6561	19683
$mq^{(m-1)/2}$	192	448	1024	2304	5120	11264	24576
$\max K_m(a;q)$	43	85	171	341	683	1365	2731

$\underline{q = 8}$

m	6	7
$(q-1)^{m-3}$	343	2401
$mq^{(m-1)/2}$	1086.1...	3584
$\max K_m(a;q)$	247	713

$\underline{q = 9}$

m	6
$(q-1)^{m-3}$	512
$mq^{(m-1)/2}$	1458
$\max K_m(a;q)$	584

$\underline{q = 16}$

m	6	7
$(q-1)^{m-3}$	3375	50625
$mq^{(m-1)/2}$	6144	28672
$\max K_m(a;q)$	1519	9745

$\underline{q = 27}$

m	6
$(q-1)^{m-3}$	17576
$mq^{(m-1)/2}$	22727.9...
$\max K_m(a;q)$	6398

$\underline{q = 32}$

m	6	7
$(q-1)^{m-3}$	29791	923521
$mq^{(m-1)/2}$	34755.7...	229376
$\max K_m(a;q)$	11359	60577

REMARK. Are the hyper-Kloosterman codes good? To see the efficiency of $C_m(2^r)$, let us use the Gilbert-Varshamov curve as a criterion:

$$y = 1 - H_2(x)\,,$$

where $H_2(x) = -x\log_2 x - (1-x)\log_2(1-x)$. Suppose $C_m(2^r)$ is a $[n,k,d]$-code. Then we can prove

$$\frac{k}{n} \geq 1 - H_2\left(\frac{d}{n}\right)$$

if $2 \leq m \leq 5$ and r is sufficiently large. This shows that good binary hyper-Kloosterman codes would exist in the range $2 \leq m \leq 5$.

3. Determination of $\dim C_m(q)$

For a code C over $\mathbf{F}_q$, we denote the trace code of C (over $\mathbf{F}_p$) by $\operatorname{tr} C$:

$$\operatorname{tr} C := \{(\operatorname{tr} c_1, \cdots, \operatorname{tr} c_n) \mid (c_1, \cdots, c_n) \in C\}\,.$$

In this section we show one way of describing the generator matrix of $\bar{C}_m(q)$, where $\operatorname{tr}\bar{C}_m(q) = C_m(q)$. This allows us to know the dimension of $C_m(q)$.

Let γ be a primitive element of $\mathbf{F}_q$, i.e. γ is a generator of the multiplicative group $\mathbf{F}_q^\times$. First we number all the vectors in $(\mathbf{F}_q^\times)^{m-1}$ as follows:

$$\begin{array}{ll}
\boldsymbol{f}_0 = (1, 1, \cdots, 1) & \cdots\cdots \\
\boldsymbol{f}_1 = (\gamma, 1, \cdots, 1) & \boldsymbol{f}_{(q-1)^2} = (1, \gamma^2, 1, \cdots, 1) \\
\boldsymbol{f}_2 = (\gamma^2, 1, \cdots, 1) & \cdots\cdots \\
\quad\cdots\cdots & \boldsymbol{f}_{(q-1)^{m-2}} = (1, \cdots, 1, \gamma) \\
\boldsymbol{f}_{q-2} = (\gamma^{q-2}, 1, \cdots, 1) & \cdots\cdots \\
\boldsymbol{f}_{q-1} = (1, \gamma, 1, \cdots, 1) & \boldsymbol{f}_{(q-1)^{m-1}-1} = (\gamma^{q-2}, \gamma^{q-2}, \cdots, \gamma^{q-2})\,. \\
\boldsymbol{f}_q = (\gamma, \gamma, 1, \cdots, 1) &
\end{array}$$

Moreover we define $I(\boldsymbol{x})$ for $\boldsymbol{x} = (x_1, x_2, \cdots, x_{m-1}) \in (\mathbf{F}_q^\times)^{m-1}$ by

$$I(\boldsymbol{x}) = (x_1 x_2 \cdots x_{m-1})^{-1}\,.$$

Then we form the matrix $G_{m,q}$ as follows:

$$G_{m,q} = \begin{pmatrix} \boldsymbol{f}_0^{\mathrm{T}} & \boldsymbol{f}_1^{\mathrm{T}} & \cdots & \boldsymbol{f}_{(q-1)^{m-1}-1}^{\mathrm{T}} \\ I(\boldsymbol{f}_0) & I(\boldsymbol{f}_1) & \cdots & I(\boldsymbol{f}_{(q-1)^{m-1}-1}) \end{pmatrix}, \tag{3.1}$$

where $\boldsymbol{x}^{\mathrm{T}}$ is the transposed vector of $\boldsymbol{x}$. If we consider a code $\bar{C}_m(q)$ over $\mathbf{F}_q$ with a generator matrix $G_{m,q}$, we can see $C_m(q) = \mathrm{tr}\,\bar{C}_m(q)$. Using $G_{m,q}$, we get the following:

THEOREM 3.1. *Let $m \geq 2$ and $r \geq 2$. Then we have*

$$\dim C_m(p^r) = \begin{cases} 2\,, & \textit{if } m = p = r = 2\,, \\ mr\,, & \textit{otherwise}\,. \end{cases}$$

PROOF. When $m = p = r = 2$, direct calculation shows $\dim C_2(4) = 2$. Otherwise, we can see $\operatorname{rank} G_{m,q} = m$, and can verify by the definition of $G_{m,q}$, that the linear mapping $\mathrm{tr} : \bar{C}_m(q) \to \mathrm{tr}\,\bar{C}_m(q) = C_m(q)$ becomes injective (otherwise tr would be a constant mapping). The code $\bar{C}_m(q)$ has $q^m = p^{mr}$ vectors, and so does $C_m(q)$. □

EXAMPLE 3.2. The generator matrix of $\bar{C}_3(4)$.

Let γ be a primitive element of $\mathbf{F}_4$. Then $\gamma^3 = 1$ and $\gamma^2 + \gamma + 1 = 0$. Then we have

$$\begin{aligned} G_{3,4} &= \begin{pmatrix} 1 & \gamma & \gamma^2 & 1 & \gamma & \gamma^2 & 1 & \gamma & \gamma^2 \\ 1 & 1 & 1 & \gamma & \gamma & \gamma & \gamma^2 & \gamma^2 & \gamma^2 \\ 1 & \gamma^{-1} & \gamma^{-2} & \gamma^{-1} & \gamma^{-2} & 1 & \gamma^{-2} & 1 & \gamma^{-1} \end{pmatrix} \\ &= \begin{pmatrix} 1 & \gamma & \gamma^2 & 1 & \gamma & \gamma^2 & 1 & \gamma & \gamma^2 \\ 1 & 1 & 1 & \gamma & \gamma & \gamma & \gamma^2 & \gamma^2 & \gamma^2 \\ 1 & \gamma^2 & \gamma & \gamma^2 & \gamma & 1 & \gamma & 1 & \gamma^2 \end{pmatrix}. \end{aligned}$$

4. Quasi-cyclic Property

As we mentioned in Introduction, the hyper-Kloosterman code $C_m(q)$ is a generalization of the Kloosterman code, which is a cyclic code. The Hamming weights of the codewords of $C_m(q)$ can be expressed by the hyper-Kloosterman sums $K_m(\boldsymbol{a}; q)$, which are obtained by increasing the number of variables of the Kloosterman sums:

$$K(\alpha, \beta; q) = \sum_{x \in \mathbf{F}_q^\times} e(\mathrm{tr}\,(\alpha x + \beta x^{-1})). \quad (\alpha, \beta \in \mathbf{F}_q) \tag{4.1}$$

What will happen to the code $C_m(q)$ by this generalization? The answer is the quasi-cyclic property. So we begin this section by introducing the notion of the quasi-cyclic code:

DEFINITION 4.1. A code C is called s-quasi-cyclic if

$$c_{n-s+1} \cdots c_n c_1 c_2 \cdots c_{n-s} \in C$$

holds for every codeword $c_1 c_2 \cdots c_n \in C$.

This property depends on the permutation of the coordinates, but if we take $G_{m,q}$ as the generator matrix of $\bar{C}_m(q)$, we can realize $C_m(q)$ as a quasi-cyclic code:

THEOREM 4.2. *The code $C_m(q)$ is $(q-1)^{m-2}$-quasi-cyclic.*

PROOF. Take a codeword $(a_1, a_2, \cdots, a_m)G_{m,q}$ of $\bar{C}_m(q)$ and apply a cyclic shift of $(q-1)^{m-2}$ digits to it. Then we can verify that the resulting vector is

$$(a_1, a_2, \cdots, a_{m-2}, a_{m-1}\gamma^{q-2}, a_m\gamma^{-(q-2)})G_{m,q}$$

and it is an element of $\bar{C}_m(q)$. Thus $\bar{C}_m(q)$ is $(q-1)^{m-2}$-quasi-cyclic, and so is $C_m(q)$. □

EXAMPLE 4.3. $C_3(4)$.

We can obtain the following table of the codewords of $C_3(4)$ (this is one of the applications of the algorithm explained in Appendix):

Weight 0:
000000000

Weight 2:
000010001
000100010
000001100
010001000
100010000
001100000
001000010
010000100
100000001

Weight 4:
011100100
110001001
101010010
011001010
101100001
110010100
100100011
001001110
010010101
100001101
010100110
001010011
100011100
001110001
010101010
100110010
010011001
001101100
000101110
110000101
101110000
101000011
011101000
000011101
110011000
000110011
011000110

Weight 6:
110111010
101001111
101111100
011010111
110100111
011111001
000111111
111010110
001111101
010111011
111100101
111001011
100111110
111111000
111101001
010110111
100101111
111011010
001011111
111110100
111000111
011011011
110110110
101101101
011110101
101011110
110101011

Since this is the case $q = 4$ and $m = 3$, it is 3-quasi-cyclic. Take for example, the codeword 000010001 (the first one of weight 2). Move the last 3 digits of it to the beginning and shift the remainder to follow them. Then we get another codeword 001000010 (the seventh one of weight 2). The same procedure to 001000010 will produce 010001000, which is the fourth one. This also holds for all other codewords.

Quasi-cyclic codes have been investigated by lots of authors since Townsend-Weldon [9], but it seems that no one has ever considered the codes of our type, quasi-cyclic trace codes. Thus our argument suggests a new method of constructing quasi-cyclic codes: in principle, a trace code described by roots of polynomials have a quasi-cyclic generalization in a similar way. Here we give another example:

EXAMPLE 4.4. Quasi-cyclic generalization of the simplex code.

Let p be a prime, $q = p^r$ $(r \geq 2)$ and $\mathbf{F}_q^\times = \langle \gamma \rangle$. The simplex code (the dual of the Hamming code when $p = 2$) is the trace code of the code over $\mathbf{F}_q$ with a generator matrix

$$G = (1, \gamma, \gamma^2, \cdots, \gamma^{q-2}) .$$

For $m \geq 1$, we define a m by $(q-1)^m$ matrix

$$G_m = \left(\begin{array}{ccccc|ccccc|c} 1 & \gamma & \gamma^2 & \cdots & \gamma^{q-2} & 1 & \gamma & \gamma^2 & \cdots & \gamma^{q-2} & \\ 1 & 1 & \cdots & & 1 & \gamma & \gamma & \cdots & & \gamma & \\ 1 & 1 & \cdots & & 1 & 1 & 1 & \cdots & & 1 & \cdots \\ \vdots & \cdots & & & & \vdots & \cdots & & & & \\ 1 & 1 & \cdots & & 1 & 1 & 1 & \cdots & & 1 & \end{array} \right.$$

$$\left. \begin{array}{c|ccccc} & 1 & \gamma & \gamma^2 & \cdots & \gamma^{q-2} \\ & \gamma^{q-2} & \gamma^{q-2} & \cdots & & \gamma^{q-2} \\ \cdots & \gamma^{q-2} & \gamma^{q-2} & \cdots & & \gamma^{q-2} \\ & \vdots & \cdots & & & \\ & \gamma^{q-2} & \gamma^{q-2} & \cdots & & \gamma^{q-2} \end{array} \right)$$

and consider the trace code of the code with a generator matrix G_m. It has a realization

$$S_m(q) = \{\psi_m(\boldsymbol{a}) := \{\mathrm{tr}\,(a_1x_1 + \cdots + a_mx_m)\}_{(x_1,\cdots,x_m)\in(\mathbf{F}_q^\times)^m} \mid \boldsymbol{a} = (a_1, \cdots, a_m) \in (\mathbf{F}_q^\times)^m\} .$$

It is a $[(q-1)^m, mr]$-code (by the same argument as in Theorem 3.1), and is a $(q-1)^{m-1}$-quasi-cyclic generalization of the simplex code.

5. Appendix — Calculation of the Kloosterman sums

The purpose of this Appendix is to propose an efficient algorithm for calculating the Kloosterman sums over finite fields. But the essential idea is how to calculate fast the trace of elements in the finite field $\mathbf{F}_q$ to the prime field $\mathbf{F}_p$. Thus it can be applied to other similar cases where values of the trace function are needed (see Example 4.3).

5.1. The main idea. Let p be a prime number and $q = p^r$ for some integer $r \geq 2$. We denote by $\mathbf{F}_q$ the finite field with q elements. We would like to calculate $\mathrm{tr}\,(\alpha)$ for $\alpha \in \mathbf{F}_q{}^\times$ which is given by

$$\mathrm{tr}\,(\alpha) = \alpha + \alpha^p + \cdots + \alpha^{p^{r-1}} . \tag{5.1}$$

Fix a primitive element $\gamma \in \mathbf{F}_q{}^\times$ and let $f(X) \in \mathbf{F}_p[X]$ be the minimal polynomial of γ. We assume that we know the explicit form of $f(X)$ (i.e. we know all the coefficients of $f(X)$ explicitly). In this situation, we can calculate tr (α)'s fast and easily.

To begin with, note that we can express $\alpha \in \mathbf{F}_q{}^\times$ in 2 ways: one is the expression in a power of γ, and the other is, so to speak, the "polynomial expression", i.e. the expression in polynomials of γ over $\mathbf{F}_p$, of which the degrees are less than r. Let $\alpha = \gamma^j$ $(0 \le j \le q-2)$. Then the polynomial expression of α is obtained by reduction of γ^j mod $f(\gamma)$. Suppose for any k $(0 \le k \le q-2)$, the polynomial expression of γ^k is given by $h_k(\gamma)$:

$$\gamma^k = h_k(\gamma) \quad (h_k(X) \in \mathbf{F}_p[X])\,. \tag{5.2}$$

The following theorem is the key for fast calculation:

THEOREM 5.1. *For any* j $(0 \le j \le q-2)$, *we have*

$$\mathrm{tr}\,(\gamma^j) = h_j(0) + h_{jp \pmod{q-1}}(0) + \cdots + h_{jp^{r-1} \pmod{q-1}}(0)\,.$$

PROOF. The definition of the trace function implies

$$\mathrm{tr}\,(\gamma^j) = h_j(\gamma) + h_{jp \pmod{q-1}}(\gamma) + \cdots + h_{jp^{r-1} \pmod{q-1}}(\gamma)\,.$$

But the terms except for the constant term with respect to γ in the above formula must vanish because $\{1, \gamma, \gamma^2, \cdots, \gamma^{r-1}\}$ is a basis of $\mathbf{F}_q$ as a vector space over $\mathbf{F}_p$, and tr $(\gamma^j) \in \mathbf{F}_p$. Hence we get the theorem. □

This theorem shows that all the information we need to calculate tr (γ^j)'s is the constant terms of the polynomial expressions $h_j(\gamma)$'s.

5.2. Kloosterman sums. In this section we will see how Theorem 5.1 can be used in numerical calculation by looking at an example of the Kloosterman sums (4.1).

If $a := \alpha\beta \in \mathbf{F}_q{}^\times$, then we have

$$K(\alpha, \beta; q) = K(a; q) := \sum_{x \in \mathbf{F}_q^\times} e(\mathrm{tr}\,(ax + x^{-1}))\,. \tag{5.3}$$

If we fix a field $\mathbf{F}_q$, its primitive element γ, and know the minimal polynomial $f(X)$ of γ, then we can write a simple and fast program to calculate all the $K(a; q)$'s. Here is an example program in the C language:

EXAMPLE 5.2. Kloosterman sums over $\mathbf{F}_{81}$.

Let γ be a root of $X^4 + X + 2$, which is a primitive polynomial over $\mathbf{F}_3$. The following program finds all the $K(\gamma^k; 81)$'s for $0 \le j \le q-2$. The former half of the program (lines 7–19) calculates the constant term of the polynomial expression $h_j(\gamma)$, which is stored in the array `x[j]`. The latter half is the main part, the calculation of $K(\gamma^k; 81)$'s (lines 21–39). According to Theorem 5.1, tr (γ^j) is given by

```
(x[j]+x[3*j%80]+x[9*j%80]+x[27*j%80])%3
```

in the notation of the C language. Let

$$t(k) = \sharp\{x \in \mathbf{F}_q^\times \mid \mathrm{tr}\,(x + ax^{-1}) = k\} \quad (0 \le k \le p-1)\,.$$

Then we can easily see that

$$K(\gamma^k; 81) = \sum_{k=0}^{2} t(k)\cos(2\pi k/3) = t(0) - (t(1)+t(2))/2\,.$$

In the program below, the element γ is denoted by a.

```
#include <stdio.h>
main()
{
  int b, j, k, t[3];
  char coef[5], x[80];
 /*  ----- Calculation of the polynomial expressions -----  */
  coef[0]=1;
  for( k=1; k<80; ++k )
  {
    for( j=4; j>0; --j )coef[j]=coef[j-1];
    coef[0]=0;
    if( coef[4]>0 )
    {
      coef[1]=(coef[1]+coef[4]*2)%3; coef[0]=(coef[0]+coef[4])%3;
/*    coef[4]=0;         This line can be omitted.     */
    }
    x[k]=coef[0];
  }
  x[0]=1;
/*  ----- Calculation of the Kloosterman sums -----  */
  for( k=0; k<80; ++k )
  {
    t[0]=0; t[1]=0; t[2]=0;
    for( j=0; j<80; ++j )
    {
      b=( x[j]+x[(80+k-j)%80]
        +x[3*j%80]+x[3*(80+k-j)%80]
        +x[9*j%80]+x[9*(80+k-j)%80]
        +x[27*j%80]+x[27*(80+k-j)%80])%3;
      switch( b )
      {
        case 0: ++t[0]; break;
        case 1: ++t[1]; break;
        default: ++t[2]; break;
      }
    }
    printf( "K(a^%d, 81)=%d\n", k, t[0]-(t[1]+t[2])/2 );
  }
}
```

REMARK. It is quite easy to write a program for the hyper-Kloosterman sums (1.2) with $m \geq 3$, using the above algorithm. But in practice, it is hard to calculate them because the complexity increases rapidly as m becomes large. It is faster to use the linear recurrence relation

$$K_m(\gamma^k; q) = \sum_{j=0}^{q-2} e(\mathrm{tr}\,(\gamma^{k-j}))K_{m-1}(\gamma^j; q)$$

(which is a variant of the formula (2.1) of [4]) with the initial values $K_2(\gamma^k; q) = K(\gamma^k; q)$. The tables of Section 2 are obtained by the algorithm of this section, together with this recurrence.

References

[1] K. CHINEN and T. HIRAMATSU, Hyper-Kloosterman sums and their applications to the coding theory, Appl. Algebra Engrg. Comm. Comput. **12** (2001), 381–390.

[2] T. HIRAMATSU, Uniform distribution of the weights of the Kloosterman codes, SUT J. Math. **31** (1995), 29–32.

[3] G. LACHAUD, Distribution of the weights of the dual of the Melas code, Discrete Math. **79** (1989/90), 103–106.

[4] D. H. and E. LEHMER, The cyclotomy of hyper-Kloosterman sums, Acta Arith. **14** (1968), 89–111.

[5] F. J. MACWILLIAMS and N. J. A. SLOANE, *The Theory of Error Correcting Codes*, North Holland (1977).

[6] C. M. MELAS, A cyclic code for double error correction, IBM J. Res. Devel. **4** (1960), 364–366.

[7] *Séminair de géométrie algébrique du Bois-marie SGA*$4\frac{1}{2}$, Lecture Notes in Math. **569**, Springer (1977).

[8] H. STICHTENOTH, *Algebraic Function Fields and Codes*, Springer (1993).

[9] R. L. TOWNSEND and E. L. WELDON, JR., Self-orthogonal quasi-cyclic codes, *IEEE Trans. Inform. Theory*, IT-13 No. 2 (1967), 183–195.

[10] J. WOLFMANN, The weights of the dual code of the Melas code over GF(3), Discrete Math. **74** (1989), 327–329.

Present Address:
DEPARTMENT OF MATHEMATICS, FACULTY OF ENGINEERING,
OSAKA INSTITUTE OF TECHNOLOGY,
OMIYA, ASAHI-KU, OSAKA, 535–8585 JAPAN.
e-mail: YHK03302@nifty.ne.jp