Weight Enumerators for Number-Theoretic Codes and Cardinalities of Tenengolts' Non-binary Codes

Takayuki Nozaki

Yamaguchi University
(This work supported by Inamori Research Grants)

ISIT2020

Background

Error correcting codes are evaluated by

- Error correcting capability
  (e.g., Decoding error rate, the number of correctable errors)
- Code rate

Once error correcting capability is fixed, we should design a code with large rate.

[Definition] Rate of r-ary code

- $[\![r]\!] := \{0, 1, 2, \ldots, r-1\}$
- $C \subset [\![r]\!]^n$: r-ary code of length n
- $|C|$: cardinality of the code (or number of codewords)

$$\text{(Rate)} := \frac{\log_r |C|}{n}$$

The rate is derived from the cardinality of the code.

Derivation of Cardinality

Linear Codes

- Defined by the parity check matrix $\mathbf{H}$

$$C := \{\boldsymbol{x} \in \mathbb{F}_r^n \mid \mathbf{H}\boldsymbol{x}^T = \mathbf{0}^T\}$$

- Most of error correcting codes are linear codes
- Cardinality is easily derived from code length n and rank of $\mathbf{H}$

$$|C| = r^{n-\mathrm{rank}(\mathbf{H})}$$

Number Theoretic Codes

- Defined by one or more congruences $\rho_i(\boldsymbol{x}) \equiv a_i \pmod{m_i}$
  $\rho_i : [\![r]\!]^n \to \mathbb{Z}$

  $$C := \{\boldsymbol{x} \in [\![r]\!]^n \mid \rho_1(\boldsymbol{x}) \equiv a_1 \pmod{m_1}, \rho_2(\boldsymbol{x}) \equiv a_2 \pmod{m_2}, \dots\}$$

- Most of deletion correcting codes are number theoretic codes
- Derivation of cardinality is not easy problem

$$|C| = ?$$

How to derive the cardinalities of number theoretic codes?

Example: Cardinality of Binary VT Code

Binary VT Code [Varshamov-Teneholtz1965]

$$\mathrm{BVT}_a(n) := \{\boldsymbol{x} = x_1 x_2 \cdots x_n \in \{0,1\}^n \mid \sum_{i=1}^{n} i x_i \equiv a \pmod{n+1}\}$$

$a \in \{0, 1, \ldots, n\}$

- Single insertion/deletion correcting code [Levenshtein1966]

$n = 3$

$$\mathrm{BVT}_0(3) = \{000, 101\},$$
$$\mathrm{BVT}_1(3) = \{100, 011\},$$
$$\mathrm{BVT}_2(3) = \{010, 111\},$$
$$\mathrm{BVT}_3(3) = \{110, 001\}$$

$n = 4$

$$\mathrm{BVT}_0(4) = \{0000, 1001, 0110, 1111\},$$
$$\mathrm{BVT}_1(4) = \{1000, 0101, 1110\},$$
$$\mathrm{BVT}_2(4) = \{0100, 1101, 0011\},$$
$$\mathrm{BVT}_3(4) = \{1100, 0010, 1011\},$$
$$\mathrm{BVT}_4(4) = \{1010, 0001, 0111\},$$

- Cardinality depends on not only code length n but also parameter a.
- Which parameter a achieves the maximum cardinality (i.e., rate)?

Cardinality of Binary VT Code

[Ginzburg1967], [Stanley-Yoder1972]

$$|\mathrm{BVT}_a(n)| = \frac{1}{2(n+1)} \sum_{d|(n+1), d:\text{odd}} c_d(a) 2^{(n+1)/d}$$

- $\sum_{d|(n+1), d:\text{odd}}$ expresses summation over odd d that divides $(n+1)$
- $c_d(a)$: Ramanujan's sum

$$c_d(a) = \phi(d) \frac{\mu(\frac{d}{\gcd(a,d)})}{\phi(\frac{d}{\gcd(a,d)})}$$

- $\gcd(a,b)$: Greatest common divisor of a and b
- $\mu(n)$: Möbius function
- $\phi(n)$: Euler's totient function

Cardinality of Binary VT Code

[Ginzburg1967], [Stanley-Yoder1972]

$$|\mathrm{BVT}_a(n)| = \frac{1}{2(n+1)} \sum_{d|(n+1), d:\text{odd}} c_d(a) 2^{(n+1)/d}$$

Outline of Derivation:

1. Calculating Hamming weight enumerator

$$\mathcal{H}(\mathrm{BVT}_a(n); z) := \sum_{\boldsymbol{x} \in \mathrm{BVT}_a(n)} z^{\mathrm{wt_H}(\boldsymbol{x})} = \sum_{i=0}^{n} A_i z^i$$

$\mathrm{wt_H}(\boldsymbol{x})$: Hamming weight (the number of non-zero entries) of $\boldsymbol{x}$
A_i : the number of codewords of weight i in $\mathrm{BVT}_a(n)$

2. Deriving cardinality by the following identity

$$\mathcal{H}(\mathrm{BVT}_a(n); 1) = \sum_{\boldsymbol{x} \in \mathrm{BVT}_a(n)} 1 = |\mathrm{BVT}_a(n)|$$

Binary Linear Congruence Code

Binary Linear Congruence (BLC) Code [Bibak-Milenkovic2018]

- A general class of number-theoretic code
- Binary code defined by a linear congruence

$$\mathrm{BLC}_a(n, m, \boldsymbol{h}) := \{\boldsymbol{x} \in \underbrace{\{0,1\}^n}_{\text{binary}} \mid \underbrace{\textstyle\sum_{i=1}^n h_i x_i \equiv a \pmod{m}}_{\text{linear congruence}}\}$$

$\boldsymbol{h} = (h_1, h_2, \ldots, h_n) \in \mathbb{N}^n$

[Bibak-Milenkovic2018] derived weight enumerate of BLC Code

Codes included in BLC code

- VT code $\{\boldsymbol{x} \in \{0,1\}^n \mid \sum_{i=1}^n i x_i \equiv a \pmod{n+1}\}$
- Levenshtein code $\{\boldsymbol{x} \in \{0,1\}^n \mid \sum_{i=1}^n i x_i \equiv a \pmod{m}\}$
- Helberg code
- Odd coefficient code $\{\boldsymbol{x} \in \{0,1\}^n \mid \sum_{i=1}^n (2i-1) x_i \equiv b \pmod{R}\}$

Binary Linear Congruence Code

Binary Linear Congruence (BLC) Code [Bibak-Milenkovic2018]

- A general class of number-theoretic code
- Binary code defined by a linear congruence

$$\mathrm{BLC}_a(n, m, \boldsymbol{h}) := \{\boldsymbol{x} \in \underbrace{\{0,1\}^n}_{\text{binary}} \mid \underbrace{\textstyle\sum_{i=1}^{n} h_i x_i \equiv a \pmod{m}}_{\text{linear congruence}}\}$$

$\boldsymbol{h} = (h_1, h_2, \ldots, h_n) \in \mathbb{N}^n$

However, BLC code does not contain several number-theoretic codes.

(Example) Tenengolts' Non-binary code [Tenengolts1984]

Non-binary code defined by two non-linear/linear congruences

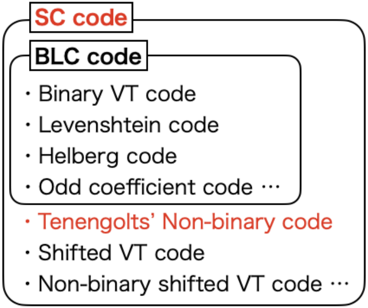
$$\mathrm{T}_{a,b}(n, r) = \{\boldsymbol{x} \in \underbrace{[\![r]\!]^n}_{r\text{-ary}} \mid \underbrace{\gamma(\boldsymbol{x}) \equiv a \pmod{n}}_{\text{non-linear cong.}}, \underbrace{\sigma(\boldsymbol{x}) \equiv b \pmod{r}}_{\text{linear cong.}}\}$$

- $\gamma(\boldsymbol{x}) := \sum_{i=1}^{n-1} i\mathbb{I}\{x_i > x_{i+1}\}$ (non-linear mapping)
- $\sigma(\boldsymbol{x}) := \sum_{i=1}^{n} x_i$
- $\mathbb{I}\{P\} = \begin{cases} 1 & P \text{ is true} \\ 0 & P \text{ is false} \end{cases}$

Purpose and Outline of This Talk

Purpose of this research

Deriving the cardinality of the Tenengolts' non-binary code



1. Define SC code
   - More general class of number theoretic code
   - Include Tenengolts' non-binary code
2. Weight enumerator for SC code
3. Weight enumerator for Tenengolts' non-binary code
4. Cardinality for Tenengolts' non-binary code

Non-binary Simultaneous Congruences Code

Non-binary simultaneous congruences (SC) code

For all $1 \le i \le s$, $\rho_i : [\![r]\!]^n \to \mathbb{Z}$

r-ary $\quad$ s non-linear congruences

$$C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r) = \{\boldsymbol{x} \in [\![r]\!]^n \mid \begin{array}{l} \rho_1(\boldsymbol{x}) \equiv a_1 \pmod{m_1}, \\ \rho_2(\boldsymbol{x}) \equiv a_2 \pmod{m_2}, \\ \vdots \\ \rho_s(\boldsymbol{x}) \equiv a_s \pmod{m_s}\} \end{array}$$

Comparison with BLC code

	defined by			
BLC Code	binary	single	linear	congruence
SC Code	r-ary	multiple	non-linear	congruences

Note: SC code is a generalization of BLC code

Codes included in SC code

$$C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r) = \{\boldsymbol{x} \in [\![r]\!]^n \mid \begin{aligned} \rho_1(\boldsymbol{x}) &\equiv a_1 \pmod{m_1}, \\ \rho_2(\boldsymbol{x}) &\equiv a_2 \pmod{m_2}, \\ &\vdots \\ \rho_s(\boldsymbol{x}) &\equiv a_s \pmod{m_s}\} \end{aligned}$$

Codes included in SC code

$s = 2$

- Shifted VT code
  $\sum_{i=1}^{n} i x_i \equiv a_1 \pmod{R}$, $\sum_{i=1}^{n} x_i \equiv a_2 \pmod{2}$
- Tenengolts' non-binary code
  $\sum_{i=1}^{n-1} i\mathbb{I}\{x_i > x_{i+1}\} \equiv a_1 \pmod{n}$, $\sum_{i=1}^{n} x_i \equiv a_2 \pmod{r}$

$s = 3$

- Non-binary Shifted VT code
  $\sum_{i=1}^{n-1} i\mathbb{I}\{x_i > x_{i+1}\} \equiv a_1 \pmod{R}$,
  $\sum_{i=1}^{n-1} \mathbb{I}\{x_i > x_{i+1}\} \equiv a_2 \pmod{2}$, $\sum_{i=1}^{n} x_i \equiv a_3 \pmod{r}$

Extended Weight Enumerator

[Definition] Extended weight enumerator

- $\boldsymbol{z} = (z_1, z_2, \dots, z_s)$
- $\boldsymbol{w} = (w_0, w_1, \dots, w_{r-1})$
- $\tau_j(\boldsymbol{x}) := |\{i \mid x_i = j\}|$: number of entries which equals to j in $\boldsymbol{x}$

$$\mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r); \boldsymbol{z}, \boldsymbol{w}) = \sum_{\boldsymbol{x} \in C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r)} \prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})} \prod_{j=0}^{r-1} w_j^{\tau_j(\boldsymbol{x})}$$

[Example] Extended weight enumerator for Tenengolts' NB code

$$\mathrm{T}_{a_1,a_2}(n,r) = \{\boldsymbol{x} \in [\![r]\!]^n \mid \gamma(\boldsymbol{x}) \equiv a_1 \pmod{n}, \sigma(\boldsymbol{x}) \equiv a_2 \pmod{r}\}$$

$$\mathcal{W}_{\gamma,\sigma}(\mathrm{T}_{a_1,a_2}(n,r); (z_1, z_2), \boldsymbol{w}) = \sum_{\boldsymbol{x} \in \mathrm{T}_{a_1,a_2}(n,r)} z_1^{\gamma(\boldsymbol{x})} z_2^{\sigma(\boldsymbol{x})} \prod_{j=0}^{r-1} w_j^{\tau_j(\boldsymbol{x})}$$

Example of Extended Weight Enumerator

[Example] Tenengolts' non-binary code ($n = 3, r = 3$)

$$\mathrm{T}_{0,0}(3,3) = \{000, 012, 111, 210, 222\}$$

$\boldsymbol{x}$	$\gamma(\boldsymbol{x})$	$\sigma(\boldsymbol{x})$	$\tau_0(\boldsymbol{x})$	$\tau_1(\boldsymbol{x})$	$\tau_2(\boldsymbol{x})$	$z_1^{\gamma(\boldsymbol{x})} z_2^{\sigma(\boldsymbol{x})} \prod_{j=0}^{2} w_j^{\tau_j(\boldsymbol{x})}$
000	0	0	3	0	0	w_0^3
012	0	3	1	1	1	$z_2^3 w_0 w_1 w_2$
111	0	3	0	3	0	$z_2^3 w_1^3$
210	3	3	1	1	1	$z_1^3 z_2^3 w_0 w_1 w_2$
222	0	6	0	0	3	$z_2^6 w_2^3$

$\sum$

$$\mathcal{W}_{\gamma,\sigma}(\mathrm{T}_{0,0}(3,3); (z_1, z_2), \boldsymbol{w}) = w_0^3 + z_2^3 w_0 w_1 w_2 + z_2^3 w_1^3 + z_1^3 z_2^3 w_0 w_1 w_2 + z_2^6 w_2^3$$

$$\gamma(\boldsymbol{x}) := \sum_{i=1}^{n-1} i \mathbb{I}\{x_i > x_{i+1}\}, \qquad \sigma(\boldsymbol{x}) := \sum_{i=1}^{n} x_i$$

Property of Extended Weight Enumerator

Extended weight enumerator is a generalization of Hamming weight enumerator

$$\mathcal{H}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);w) = \mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);\mathbf{1},\boldsymbol{w}^*)$$

$\mathbf{1} := (1,1,\ldots,1)$, $\boldsymbol{w}^* := (1,w,w,\ldots,w)$

$$\mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);\boldsymbol{z},\boldsymbol{w}) = \sum_{\boldsymbol{x}\in C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r)} \left(\prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})}\right)\left(w_0^{\tau_0(\boldsymbol{x})}\prod_{j=1}^{r-1} w_j^{\tau_j(\boldsymbol{x})}\right)$$

$$\mathcal{H}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);w) = \sum_{\boldsymbol{x}\in C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r)} w^{\mathrm{wt_H}(\boldsymbol{x})}$$

Note: $\mathrm{wt_H}(\boldsymbol{x}) = \sum_{j=1}^{r-1}\tau_j(\boldsymbol{x})$.

Extended weight enumerator for SC code

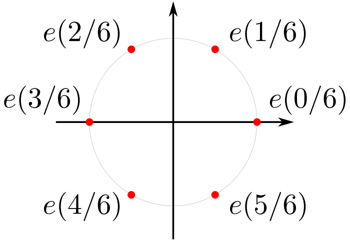
Theorem 1

$$\mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);\boldsymbol{z},\boldsymbol{w}) = \sum_{k_1=1}^{m_1}\sum_{k_2=1}^{m_2}\cdots\sum_{k_s=1}^{m_s}\mathcal{W}_{\boldsymbol{\rho}}([\![r]\!]^n;\boldsymbol{z}e(\boldsymbol{k}/\boldsymbol{m}),\boldsymbol{w})\prod_{i=1}^{s}\frac{1}{m_i}e\left(-\frac{a_ik_i}{m_i}\right),$$

where $\boldsymbol{z}e(\boldsymbol{k}/\boldsymbol{m}) := (z_1e(k_1/m_1), z_2e(k_2/m_2), \ldots, z_te(k_s/m_s))$.

Remark: Weight enumerator for SC code $\mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);\boldsymbol{z},\boldsymbol{w})$ is derived from Weight enumerator for whole space $\mathcal{W}_{\boldsymbol{\rho}}([\![r]\!]^n;\boldsymbol{z},\boldsymbol{w})$

- $\mathtt{i} = \sqrt{-1}$
- $e(\frac{j}{m}) = \exp[2\pi\mathtt{i}\frac{j}{m}]$ (points of unit circle on complex plain)



Proof of Theorem 1

Lemmas to derive Theorem 1

1. Condition of summation is written by an indicator function:

$$\sum_{\boldsymbol{x} \in C \subseteq [\![r]\!]^n} f(\boldsymbol{x}) = \sum_{i \in [\![r]\!]^n} f(\boldsymbol{x}) \mathbb{I}[\boldsymbol{x} \in C]$$

2. If $C := \{\boldsymbol{x} \mid \forall i \in \{1, 2, \ldots, s\} \;\; \rho_i(\boldsymbol{x}) \equiv a_i \pmod{m_i}\}$, then

$$\mathbb{I}[\boldsymbol{x} \in C] = \prod_{i=1}^{s} \mathbb{I}[\rho_i(\boldsymbol{x}) \equiv a_i \pmod{m_i}]$$

3.

$$\begin{aligned}\mathbb{I}[A \equiv B \pmod{m}] &= \mathbb{I}[(A - B) \equiv 0 \pmod{m}] \\ &= \frac{1}{m} \sum_{j=1}^{m} e\left(\frac{(A-B)j}{m}\right)\end{aligned}$$

$$
\begin{aligned}
&\mathcal{W}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r);\boldsymbol{z},w)\\
&= \sum_{\boldsymbol{x}\in C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r)} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})} \prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})}\\
&\overset{(1)}{=} \sum_{\boldsymbol{x}\in[\![r]\!]^n} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})}\mathbb{I}[\boldsymbol{x}\in C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r)] \prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})}\\
&\overset{(2)}{=} \sum_{\boldsymbol{x}\in[\![r]\!]^n} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})} \left(\prod_{i=1}^{s}\mathbb{I}\{\rho_i(\boldsymbol{x})\equiv a_i \pmod{m_i}\}\right)\left(\prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})}\right)\\
&\overset{(3)}{=} \sum_{\boldsymbol{x}\in[\![r]\!]^n} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})} \left(\prod_{i=1}^{s}\frac{1}{m_i}\sum_{k_i=1}^{m_i} e\left(\frac{(\rho_i(\boldsymbol{x})-a_i)k_i}{m_i}\right)\right)\left(\prod_{i=1}^{s} z_i^{\rho_i(\boldsymbol{x})}\right)\\
&= \sum_{\boldsymbol{x}\in[\![r]\!]^n} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})} \left(\prod_{i=1}^{s}\sum_{k_i=1}^{m_i}\frac{1}{m_i}e\left(\frac{-a_ik_i}{m_i}\right)e\left(\frac{\rho_i(\boldsymbol{x})k_i}{m_i}\right)z_i^{\rho_i(\boldsymbol{x})}\right)\\
&= \sum_{k_1=1}^{m_1}\cdots\sum_{k_s=1}^{m_s}\left(\prod_{i=1}^{s}\frac{1}{m_i}e\left(\frac{-a_ik_i}{m_i}\right)\right)\sum_{\boldsymbol{x}\in[\![r]\!]^n} \boldsymbol{w}^{\boldsymbol{\tau}(\boldsymbol{x})}\left(\prod_{i=1}^{s}\left(z_ie\left(\frac{k_i}{m_i}\right)\right)^{\rho_i(\boldsymbol{x})}\right)\\
&= \sum_{k_1=1}^{m_1}\cdots\sum_{k_s=1}^{m_s}\left(\prod_{i=1}^{s}\frac{1}{m_i}e\left(\frac{-a_ik_i}{m_i}\right)\right)\mathcal{W}_{\boldsymbol{\rho}}([\![r]\!]^n;\boldsymbol{z}e(\boldsymbol{k}/\boldsymbol{m}),\boldsymbol{w})
\end{aligned}
$$

Cardinality of Tenengolts' non-binary code

Theorem 3

For any $a_1 \in [\![n]\!], a_2 \in [\![r]\!]$, the cardinality of Tenengolts' non-binary code is

$$|\mathrm{T}_{a_1,a_2}(n,r)| = \frac{1}{nr} \sum_{d \in \mathbb{Z}^+} c_d(a_1) r^{\frac{n}{d}} \gcd(r,d) \mathbb{I}\{d \mid n\} \mathbb{I}\{\gcd(r,d) \mid a_2\}.$$

Corollary 1

Let $n, r \in \mathbb{Z}^+$, $a_1 \in [\![n]\!]$ and $a_2 \in [\![r]\!]$. For any n, r, a_1, a_2, the following holds

$$|\mathrm{T}_{0,0}(n,r)| \geq |\mathrm{T}_{a_1,a_2}(n,r)|$$

Note: $c_d(0) \geq c_d(a)$ for all $a \in \mathbb{Z}$.

Example:

$$\mathrm{T}_{a_1,a_2}(3,3) = \{\boldsymbol{x} \in [\![3]\!]^3 \mid \gamma(\boldsymbol{x}) \equiv a_1 \pmod 3, \sigma(\boldsymbol{x}) \equiv a_2 \pmod 3\}$$

	$a_2 = 0$	$a_2 = 1$	$a_2 = 2$
$a_1 = 0$	$\{000, 012, 111, 210, 222\}$	$\{001, 022, 112\}$	$\{002, 011, 122\}$
$a_1 = 1$	$\{102, 201\}$	$\{100, 202, 211\}$	$\{101, 200, 212\}$
$a_1 = 2$	$\{021, 120\}$	$\{010, 121, 220\}$	$\{020, 110, 221\}$

From Theorem 3,

$$\begin{aligned} |\mathrm{T}_{a_1,a_2}(3,3)| &= \tfrac{1}{9} \textstyle\sum_{d|3} 3^{3/d} c_d(a_1) \gcd(3,d) \mathbb{I}\{\gcd(3,d) \mid a_2\} \\ &= 3c_1(a_1) + \mathbb{I}\{3 \mid a_2\} c_3(a_1) \end{aligned}$$

Note that $c_1(0) = 1, c_3(0) = 2, c_3(1) = c_3(2) = -1$. We get

$$|\mathrm{T}_{a_1,a_2}(3,3)| = \begin{cases} 5, & \text{if } a_1 = 0 \text{ and } a_2 = 0, \\ 2, & \text{if } a_1 = 1,2 \text{ and } a_2 = 0, \\ 3, & \text{if } a_2 = 1,2. \end{cases}$$

Derivation of the cardinality of Tenengolts' non-binary code

$$\mathrm{T}_{a_1,a_2}(n,r) = \{\boldsymbol{x} \in [\![r]\!]^n \mid \gamma(\boldsymbol{x}) \equiv a_1 \pmod n, \sigma(\boldsymbol{x}) \equiv a_2 \pmod r\}$$

1. Deriving the extended weight enumerator for $[\![r]\!]^n$

$$\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (q,z), \boldsymbol{w})$$

2. Calculating the Hamming weight enumerator for $\mathrm{T}_{a_1,a_2}(n,r)$

$$\begin{aligned}
&\mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r); w) \\
&= \mathcal{W}_{\gamma,\sigma}(\mathrm{T}_{a_1,a_2}(n,r); (1,1), \boldsymbol{w}^*) \\
&= \sum_{k_1=1}^{n} \sum_{k_2=1}^{r} \mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (e(\tfrac{k_1}{n}), e(\tfrac{k_2}{r})), \boldsymbol{w}^*) \frac{1}{nr} e\left(-\tfrac{a_1 k_1}{n}\right) e\left(-\tfrac{a_2 k_2}{r}\right)
\end{aligned}$$

3. Deriving the cardinality

$$|\mathrm{T}_{a_1,a_2}(n,r)| = \mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r); 1)$$

Extended Weight Enumerator of Whole Space (Notation)

- Let $\tau_j(\boldsymbol{x})$ be number of symbol j in sequence $\boldsymbol{x} \in [\![r]\!]^n$, i.e.,

$$\tau_j(\boldsymbol{x}) := |\{i \in \{1, 2, \ldots, n\} \mid x_i = j\}|$$

(Example) $\tau_0(1020) = 2, \tau_1(1020) = 1, \tau_2(1020) = 1$

- The type of sequence $\boldsymbol{x}$ is

$$(\tau_0(\boldsymbol{x}), \tau_1(\boldsymbol{x}), \ldots, \tau_{r-1}(\boldsymbol{x}))$$

(Example) Type of sequence 1020 is $(2, 1, 1)$.

- Denote the set of sequences with type $\boldsymbol{t}$, by $S(\boldsymbol{t})$ i.e.,

$$S(\boldsymbol{t}) = \{\boldsymbol{x} : (\tau_0(\boldsymbol{x}), \tau_1(\boldsymbol{x}), \ldots, \tau_{r-1}(\boldsymbol{x})) = \boldsymbol{t}\}$$

(Example) $S(2, 1, 1) =$
$\{0012, 0021, 0102, 0120, 0201, 0210, 1002, 1020, 1200, 2001, 2010, 2100\}$

Extended Weight Enumerator of Whole Space

Lemma 2 [Ch.VI, MacMahon1915]

$$\sum_{\boldsymbol{x} \in S(\boldsymbol{t})} q^{\gamma(\boldsymbol{x})} = \begin{bmatrix} n \\ \boldsymbol{t} \end{bmatrix}_q := \frac{\prod_{j=1}^{n}(q^j - 1)}{\prod_{i=0}^{r-1} \prod_{k=1}^{t_i}(q^k - 1)}$$

- $T_n := \{(t_0, t_1, \dots, t_{r-1}) \in \mathbb{Z}_{\geq 0} : \sum_{i=0}^{r-1} t_i = n\}$,

Lemma 5

$$\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (q, z), \boldsymbol{w}) = \sum_{\boldsymbol{t} \in T_n} \begin{bmatrix} n \\ \boldsymbol{t} \end{bmatrix}_q \prod_{j=0}^{r-1} w_j^{t_j} z^{j t_j}$$

Proof: From Lemma 2

$$\mathcal{W}_{\gamma,\sigma}(S(\boldsymbol{t});\langle q,z\rangle,\boldsymbol{w}) = \sum_{\boldsymbol{x}\in S(\boldsymbol{t})} q^{\gamma(\boldsymbol{x})} \prod_{j\in[\![r]\!]} w_j^{t_j} z^{jt_j} = \begin{bmatrix} n \\ \boldsymbol{t} \end{bmatrix}_q \prod_{j\in[\![r]\!]} w_j^{t_j} z^{jt_j}.$$

Since $[\![r]\!]^n = \bigcup_{\boldsymbol{t}\in T_n} S(\boldsymbol{t})$ and $S(\boldsymbol{t}) \cap S(\boldsymbol{t}') = \emptyset$ for $\boldsymbol{t} \neq \boldsymbol{t}'$,

$$\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n;(q,z),\boldsymbol{w}) = \sum_{\boldsymbol{t}\in T_n} \mathcal{W}_{\gamma,\sigma}(S(\boldsymbol{t});(q,z),\boldsymbol{w})$$

Combining these

$$\begin{aligned}\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n;(q,z),\boldsymbol{w}) &= \sum_{\boldsymbol{t}\in T_n} \mathcal{W}_{\gamma,\sigma}(S(\boldsymbol{t});(q,z),\boldsymbol{w}) \\ &= \sum_{\boldsymbol{t}\in T_n} \begin{bmatrix} n \\ \boldsymbol{t} \end{bmatrix}_q \prod_{j\in[\![r]\!]} w_j^{t_j} z^{jt_j}\end{aligned}$$

Derivation of the cardinality of Tenengolts' non-binary code

1. Deriving the extended weight enumerator for $[\![r]\!]^n$

$$\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (q,z), \boldsymbol{w}) = \sum_{\boldsymbol{t} \in T_{n,r}} \begin{bmatrix} n \\ \boldsymbol{t} \end{bmatrix}_q \prod_{j=0}^{r-1} w_j^{t_j} z^{j t_j}$$

2. Calculating the Hamming weight enumerator for $\mathrm{T}_{a_1,a_2}(n,r)$

$$\mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r); w) = \sum_{k_1=1}^{n} \sum_{k_2=1}^{r} \mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (e(\tfrac{k_1}{n}), e(\tfrac{k_2}{r})), \boldsymbol{w}^*) \frac{1}{nr} e\left(-\frac{a_1 k_1}{n}\right) e\left(-\frac{a_2 k_2}{r}\right)$$

3. Deriving the cardinality

$$|\mathrm{T}_{a_1,a_2}(n,r)| = \mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r); 1)$$

Hamming weight enumerator for Tenengolts' NB-code

Lemma 6

For any $k_1 \in [\![n]\!], k_2 \in [\![r]\!]$, we get

$$\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (e(\tfrac{k_1}{n}), z), \boldsymbol{w}) = \left(\sum_{i \in [\![r]\!]} (w_i z^i)^{\frac{n}{\gcd(n,k_1)}}\right)^{\gcd(n,k_1)}$$

Proof: The following holds (generalization of [Hagiwara-Kong2019]):

$$\lim_{q \to e(1/d)} \begin{bmatrix} \sum_{j \in [\![r]\!]} t_i \\ \boldsymbol{t} \end{bmatrix}_q = \begin{pmatrix} \sum_{j \in [\![r]\!]} t_i/d \\ \boldsymbol{t}/d \end{pmatrix} \prod_{i \in [\![r]\!]} \mathbb{I}\{d \mid t_i\},$$

For a fixed k_1, define $d := \frac{n}{\gcd(n,k_1)}$. Then, $e(k_1/n)$ is a primitive d-th root of unity. Lemma 5 gives

$$\begin{aligned}\mathcal{W}_{\gamma,\sigma}([\![r]\!]^n; (e(\tfrac{k_1}{n}), z), \boldsymbol{w}) &= \sum_{\boldsymbol{t} \in T_n} \binom{n/d}{\boldsymbol{t}/d} \prod_{j \in [\![r]\!]} w_j^{t_j} z^{j t_j} \mathbb{I}\{d \mid t_j\} \\ &= \sum_{\boldsymbol{t}' \in T_{n/d}} \binom{n/d}{\boldsymbol{t}'} \prod_{j \in [\![r]\!]} (w_j z^j)^{d t'_j} = \left(\sum_{i \in [\![r]\!]} w_i^d z^{id}\right)^{n/d},\end{aligned}$$

Hamming weight enumerator for Tenengolts' NB-code

Theorem 2: Hamming weight enumerator

$$\mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r);w) = \frac{1}{nr} \sum_{d\in\mathbb{Z}^+, d|n} \sum_{e\in\mathbb{Z}^+, e|r} c_d(a_1) c_e(a_2) \times \left\{1 - w^d + r w^d \mathbb{I}\{e \mid d\}\right\}^{\frac{n}{d}},$$

Outline of Proof: Using the following properties

$$\mathcal{H}(\mathrm{T}_{a_1,a_2}(n,r);w) = \sum_{k_1=1}^{n} \sum_{k_2=1}^{r} \mathcal{W}_{\gamma,\sigma}(\llbracket r\rrbracket^n; (e(\tfrac{k_1}{n}), e(\tfrac{k_2}{r})), \boldsymbol{w}^*) \times \frac{1}{nr} e\left(-\tfrac{a_1 k_1}{n}\right) e\left(-\tfrac{a_2 k_2}{r}\right)$$

$$\mathcal{W}_{\gamma,\sigma}\left(\llbracket r\rrbracket^n; \left\langle e\left(\tfrac{k_1}{n}\right), e\left(\tfrac{k_2}{r}\right)\right\rangle, \boldsymbol{w}^*\right) = \left(1 - w^{\frac{n}{(n,k_1)}} + r w^{\frac{n}{(n,k_1)}} \mathbb{I}\{r \mid \tfrac{n k_2}{(n,k_1)}\}\right)^{(n,k_1)}.$$

$$c_d(a) = \sum_{j\in\llbracket 1,d\rrbracket, (j,d)=1} e\left(\frac{aj}{d}\right)$$

Conclusion

- We derived a formula of extended weight enumerator of the SC code
- We provided the Hamming weight enumerators of the Tenengolts' non-binary codes
- We gave the cardinalities of the Tenengolts' non-binary codes