Distance Enumerators for Number-Theoretic Codes

Takayuki Nozaki

Yamaguchi University
(This work supported by Inamori Research Grants)

ISIT2021

Abstract

Number-theoretic codes

- widely used in insertion/deletion correcting codes
- in general, non-linear codes
  $\Rightarrow$ not enough analyzed

Distance Enumerator (DE)

- used for analyzing maximum likelihood decoding performance
- For a non-linear code C, derivation of DE requires $\mathcal{O}(|C|^2)$
  $\Rightarrow$ complexity is high

Main Contribution

- We present an identity of DE for Number-theoretic codes
- We show that the identity efficiently derives DE for a special case

Distance Enumerator

Distance (in coding theory)

mapping from two sequences $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{A}^n$ to a non-negative integer:

$$\mathrm{d} : \mathcal{A}^n \times \mathcal{A}^n \to \mathbb{Z}_{\geq 0}$$

Example:

- Hamming distance
  $\mathrm{d_H}(\boldsymbol{x}, \boldsymbol{y}) = \min$ (number of substitution for $\boldsymbol{x} \to \boldsymbol{y}$)
- Insdel distance
  $\mathrm{d_{ID}}(\boldsymbol{x}, \boldsymbol{y}) = \min$ (number of insertion/deletion for $\boldsymbol{x} \to \boldsymbol{y}$)
- Levenshtein distance
  $\mathrm{d_L}(\boldsymbol{x}, \boldsymbol{y}) = \min$ (number of ins./del./substi. for $\boldsymbol{x} \to \boldsymbol{y}$)

Distance Enumerator

Distance Enumerator (DE)

Distance enumerator $\mathcal{D}(C; z)$ for a code C is

$$\mathcal{D}(C; z) := \sum_{i \geq 0} D_i z^i = \sum_{\boldsymbol{x} \in C} \sum_{\boldsymbol{y} \in C} z^{\mathrm{d}(\boldsymbol{x}, \boldsymbol{y})}$$

$$\begin{aligned} D_i &:= |\{(\boldsymbol{x}, \boldsymbol{y}) \in C^2 \mid \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) = i\}| \\ &= (\text{Number of pairs of codewords with distance } i) \end{aligned}$$

Example: Hamming distance enumerator

$$\mathcal{D}_{\mathrm{H}}(C; z) = \sum_{\boldsymbol{x} \in C} \sum_{\boldsymbol{y} \in C} z^{\mathrm{d}_{\mathrm{H}}(\boldsymbol{x}, \boldsymbol{y})}$$

Example: Hamming distance enumerator

Linear code case:

Hamming DE comes down to Hamming weight enumerator

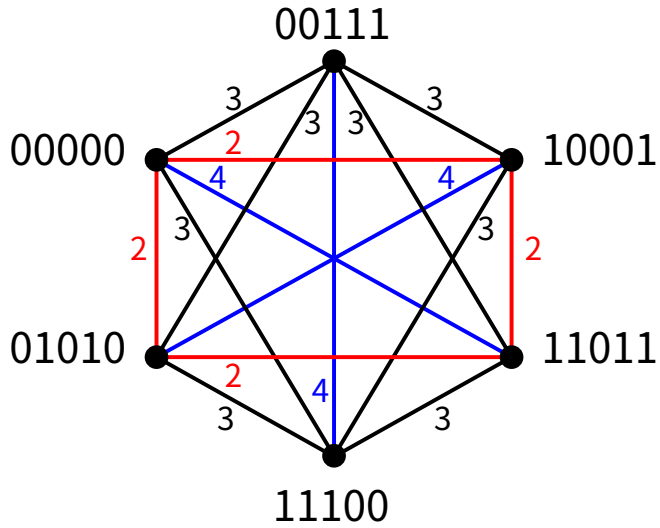
$$\mathcal{D}_{\mathrm{H}}(C;z)=\sum_{\boldsymbol{x}\in C}\sum_{\boldsymbol{y}\in C}z^{\mathrm{d}(\boldsymbol{x}-\boldsymbol{y},\boldsymbol{0})}=\sum_{\boldsymbol{x}'\in C}\sum_{\boldsymbol{y}\in C}z^{w(\boldsymbol{x}')}=|C|\mathcal{W}(C;z)$$

Non-linear code case:

Enumerate codewords and calculate distance... $\Rightarrow$ Complexity $\mathcal{O}(|C|^2)$

$$C=\{00000,10001,01010,\\ 00111,11100,11011\}$$

$$\mathcal{D}_{\mathrm{H}}(C;z)=6+2(4z^2+8z^3+3z^4)$$



Question

How to efficiently derive the distance enumerator for non-linear codes?

Number-Theoretic Code

- Code defined by single/multiple congruence(s)
- Used for insertion/deletion correction
- Non-linear codes (in general)

Example: Varshamov-Tenengolts (VT) code [Varshamov-Tenengolts-1965]

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

$$\begin{aligned} \mathrm{VT}_0(5) &= \left\{ \boldsymbol{x} \in \{0,1\}^5 \mid x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 \equiv 0 \pmod{6} \right\} \\ &= \{00000, 10001, 01010, 11100, 11011, 00111\} \end{aligned}$$

Number-Theoretic Code

- Code defined by single/multiple congruence(s)
- Used for insertion/deletion correction
- Non-linear codes (in general)

Example: Shifted VT code [Schoeny et al. 2017]

$$\mathrm{SVT}_{a,b}(n,s) = \left\{ \boldsymbol{x} \in \{0,1\}^n \mid \sum_{i=1}^{n} i x_i \equiv a \pmod{s}, \sum_{i=1}^{n} x_i \equiv b \pmod{2} \right\}$$

Example: Non-binary Tenengolts code [Tenengolts-1984]

$$\mathrm{T}_{a,b}(n,q) = \left\{ \boldsymbol{x} \in [\![q]\!]^n \mid \gamma(\boldsymbol{x}) \equiv a \pmod{n}, \sum_{i=1}^{n} x_i \equiv b \pmod{q} \right\}$$

$$[\![q]\!] := \{0, 1, \ldots, q-1\}$$

Goal of This Research

Goal

To present an efficient algorithm/formula to derive DE for number-theoretic codes

(Number-theoretic codes) = (SC code)

Simultaneous congruences (SC) code [N2020]

Define $\rho_i : [\![q]\!]^n \to \mathbb{Z}$ for $1 \le i \le s$.
$\boldsymbol{\rho} = (\rho_1, \rho_2, \ldots, \rho_s)$
$\boldsymbol{m} = (m_1, m_2, \ldots, m_s) \in \mathbb{Z}^s$
$\boldsymbol{a} = (a_1, a_2, \ldots, a_s)$ $(a_i \in \{0, 1, \ldots, m_s - 1\})$

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

Extended Distance Enumerator

We consider the extended DE to describe the main theorem easily.

Define $\rho_i : [\![q]\!]^n \to \mathbb{Z}$ for $1 \le i \le s$.
$\boldsymbol{\rho} = (\rho_1, \rho_2, \ldots, \rho_s)$, $\boldsymbol{u} = (u_1, u_2, \ldots, u_s)$, $\boldsymbol{v} = (v_1, v_2, \ldots, v_s)$
The extended DE $\mathcal{D}_{\boldsymbol{\rho}}(C; z, \boldsymbol{u}, \boldsymbol{v})$ parameterized by $\boldsymbol{\rho}$ for $C \subseteq [\![q]\!]^n$ is

$$\mathcal{D}_{\boldsymbol{\rho}}(C; z, \boldsymbol{u}, \boldsymbol{v}) = \sum_{\boldsymbol{x} \in C} \sum_{\boldsymbol{y} \in C} z^{\mathrm{d}(\boldsymbol{x}, \boldsymbol{y})} \prod_{i=1}^{s} u_i^{\rho_i(\boldsymbol{x})} v_i^{\rho_i(\boldsymbol{y})}.$$

(c.f.) Distance enumerator

$$\mathcal{D}_{\mathrm{H}}(C; z) = \sum_{\boldsymbol{x} \in C} \sum_{\boldsymbol{y} \in C} z^{\mathrm{d}(\boldsymbol{x}, \boldsymbol{y})}$$

$$\mathcal{D}_{\boldsymbol{\rho}}(T; z, \mathbf{1}, \mathbf{1}) = \mathcal{D}(T; z).$$

$\mathbf{1} := (1, 1, \ldots, 1)$

Extended DE for SC Codes

[Theorem 1] Extended DE for SC Codes

- $e(a) := \exp[2\pi \mathtt{i} a] = \cos(2\pi a) + \mathtt{i}\sin(2\pi a)$
- $\boldsymbol{u}e\left(\frac{\boldsymbol{k}}{\boldsymbol{m}}\right) := (u_1 e(\frac{k_1}{m_1}), u_2 e(\frac{k_2}{m_2}), \ldots, u_s e(\frac{k_s}{m_s}))$

$$\mathcal{D}_{\boldsymbol{\rho}}(C_{\boldsymbol{\rho},\boldsymbol{a},\boldsymbol{m}}(n,r,s); z, \boldsymbol{u}, \boldsymbol{v})$$
$$= \sum_{j_1=0}^{m_1-1} \sum_{k_1=0}^{m_1-1} \sum_{j_2=0}^{m_2-1} \sum_{k_2=0}^{m_2-1} \cdots \sum_{j_s=0}^{m_s-1} \sum_{k_s=0}^{m_s-1}$$
$$\mathcal{D}_{\boldsymbol{\rho}}\left([\![q]\!]^n; z, \boldsymbol{u}e\left(\frac{\boldsymbol{j}}{\boldsymbol{m}}\right), \boldsymbol{v}e\left(\frac{\boldsymbol{k}}{\boldsymbol{m}}\right)\right) \prod_{i=1}^{s} \frac{1}{m_i^2} e\left(-\frac{a_i(j_i+k_i)}{m_i}\right).$$

Remark:

- If we have an explicit form of $\mathcal{D}_{\boldsymbol{\rho}}\left([\![q]\!]^n; z, \boldsymbol{u}, \boldsymbol{v}\right)$, we can get DE for a SC code.
- In some special case, we can obtain the explicit form of $\mathcal{D}_{\boldsymbol{\rho}}\left([\![q]\!]^n; z, \boldsymbol{u}, \boldsymbol{v}\right)$

Example: Hamming Distance Enumerator for BLC Codes

[Definition] Binary linear congruence (BLC) code [Bibak-Milenkovic-2018]

Binary code defined by a single linear congruence

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

[Corollary 1] Hamming DE for BLC code

$$\begin{aligned}
&\mathcal{D}_{\mathrm{H}}(\mathrm{BLC}_a(n, m, \boldsymbol{h}); z) \\
&= \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} \frac{1}{m^2} e\left(-\frac{a(j+k)}{m}\right) \\
&\quad \times \prod_{i=1}^{n} \left\{1 + e\left(\frac{h_i j}{m}\right) z + e\left(\frac{h_i k}{m}\right) z + e\left(\frac{h_i (j+k)}{m}\right)\right\}.
\end{aligned}$$

Complexity of derivation is $\mathcal{O}(nm^2)$

Example: Hamming DE for VT Codes

Let us derive the Hamming DE for VT code ($m := n + 1$).

$$\mathcal{D}_{\mathrm{H}}(\mathrm{VT}_a(n); z) = \sum_{j=0}^{m-1} \sum_{k=0}^{m-1} \frac{1}{m^2} e\left(-\frac{a(j+k)}{m}\right) \frac{B_{m,j,k}(z)}{2+2z},$$

$$B_{m,j,k}(z) := \prod_{i=1}^{m} \left\{1 + e\left(\tfrac{i(j+k)}{m}\right) + e\left(\tfrac{ij}{m}\right) z + e\left(\tfrac{ik}{m}\right) z\right\}$$

We need to derive $B_{m,j,k}$ for $j, k = 0, 1, \ldots, m-1$.

When $m = 9$, we need to derive $B_{9,j,k}$ $(j, k = 0, 1, \ldots, 8)$.

	0	1	2	3	4	5	6	7	8
0									
1									
2									
3									
4									
5									
6									
7									
8									

Example: Hamming DE for VT Codes

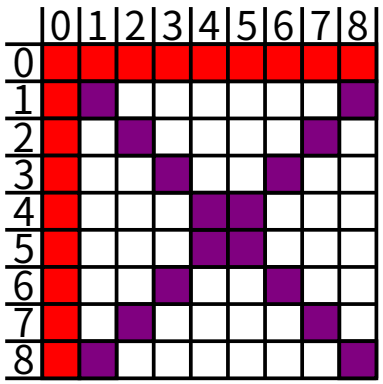
Explicit formula for some special j, k

Define $d := \gcd(m, j)$ and $m' := m/d$. Denote $\bar{m} := \lfloor \frac{m'-1}{2} \rfloor$.
Let $U_n(z), V_n(z)$ be the Chebyshev polynomials of 2nd and 3rd kind.
For any $m \in \mathbb{Z}$, $j \in [\![m]\!]$,

$$B_{m,j,0}(z) = B_{m,0,j}(z) = 2^d (1+z)^m \mathbb{I}[m' : \text{odd}] \tag{1}$$

$$B_{m,j,j}(z) = \begin{cases} (-1)^j 2^{2d} (z^2-1)^d \left[U_{\bar{m}}(z)\right]^{2d}, & (m' \text{:even}), \\ 2^d (z+1)^d \left[V_{\bar{m}}(z)\right]^{2d}, & (m' \text{:odd}). \end{cases} \tag{2}$$

$$B_{m,j,m-j}(z) = \begin{cases} 2^{2d} (z^2-1)^d [z^{\bar{m}} U_{\bar{m}}(z^{-1})]^{2d}, & (m' : \text{even}), \\ 2^d (z+1)^d [z^{\bar{m}} V_{\bar{m}}(z^{-1})]^{2d}, & (m' : \text{odd}). \end{cases} \tag{3}$$



Example: Hamming DE for VT Codes

Symmetries

For any $m \in \mathbb{Z}^+$, $j, k \in [\![m]\!]$,

$$B_{m,j,k}(z) = B_{m,k,j}(z), \tag{4}$$

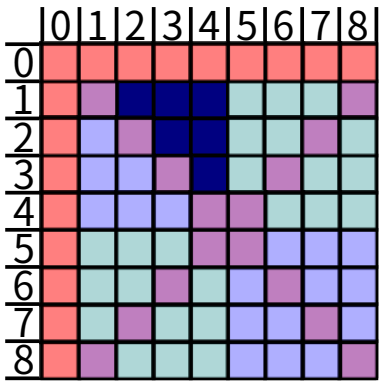
$$B_{m,j,k}(z) = (-1)^{j(m+1)} z^m B_{m,m-j,k}(z^{-1}), \tag{5}$$

$$B_{m,j,k}(z) = (-1)^{k(m+1)} z^m B_{m,j,m-k}(z^{-1}). \tag{6}$$

Reduction

For integer t coprime to m,

$$B_{m,jt,kt}(z) = B_{m,j,k}(z). \tag{7}$$



Example: Hamming DE for VT Codes

Define $d := \gcd(m, j, k)$. For any $m \in \mathbb{Z}^+$, $j, k \in [\![m]\!]$

$$B_{m,j,k}(z) = \left[B_{\frac{m}{d},\frac{j}{d},\frac{k}{d}}(z)\right]^d.$$

Suppose odd m. For any $j, k \in [\![m]\!]$ such that $\gcd(m, j, k) = 1$,

$$B_{m,j,k}(z) = 2^m(1+z)\prod_{i=1}^{\frac{m-1}{2}}\left\{\cos\left(\pi\frac{j+k}{m}i\right) + z\cos\left(\pi\frac{j-k}{m}i\right)\right\}^2 \quad (8)$$

Suppose even m. For any $j, k \in [\![m]\!]$ such that $\gcd(m, j, k) = 1$,

$$B_{m,j,k}(z) = 2^m(1-z^2)\mathbb{I}[j : \text{odd}]\mathbb{I}[k : \text{odd}] \times \prod_{i=1}^{\frac{m}{2}-1}\left\{\cos\left(\pi\frac{j+k}{m}i\right) + z\cos\left(\pi\frac{j-k}{m}i\right)\right\}^2 \quad (9)$$

Numerical Example:

Evaluation time[1]

Code length	Brute force (sec)	Proposed (sec)
20	102.2	0.02425
24	21793	0.03054
32	-	0.12909
255	-	11.5027

Theorem 1 allows us to efficiently derive the Hamming DE for VT code

[1]CPU: Intel Core i5-8400 (2.80GHz)
Memory: 32GB
Language: C++

Numerical Example: $\mathcal{D}_{\mathrm{H}}(\mathrm{VT}_a(15); z)$

i	$a=0$	$a=1,3,5,7,9,$ $11,13,15$	$a=2,6,10$ 14	$a=4,12$	$a=8$
0	2048	2048	2048	2048	2048
1	0	0	0	0	0
2	7184	7168	7168	7152	7184
3	64496	64512	64512	64528	64496
4	183488	183552	183456	183808	183488
5	375616	375552	375648	375296	375616
6	633152	632832	633280	631616	633152
7	831168	831488	831040	832704	831168
8	828352	828736	828160	832704	828352
9	635968	635584	636160	631616	635968
10	382528	382400	382624	375296	382528
11	176576	176704	176480	183808	176576
12	58384	58368	58368	64528	58384
13	13296	13312	13312	7152	13296
14	2048	2048	2048	0	2048
15	0	0	0	2048	0

$\mathrm{VT}_0(15)$ is not optimal in terms of DE.

Conclusion and Future Works

Conclusion

- We present an identity for extended DE for SC codes
- We give an algorithm to calculate Hamming DE for VT codes

Future works

- Derive insdel/Levenshtein DE for some SC codes
- Derive Hamming DE for other SC codes
- Give an efficient algorithm to derive DE for some SC codes (e.g., Hamming DE for BLC codes)