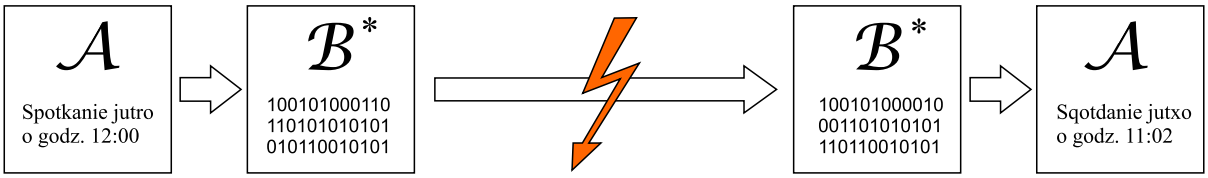
Kodowanie, Kompresja, Kryptografia

Konspekt IV, 3.IV.2023

Kodowanie w obecności szumu



Zakodowany sygnał $b_1b_2...b_n \in \mathcal{B}^*$ po przesłaniu odbieramy jako $b'_1b'_2...b'_n$. Niech B_k i B'_k oznaczają zmienne losowe o wartościach k-tej nadanej i otrzymanej litery b_k i b'_k.

Założenie:

$$P(\, b'_1b'_2...b'_n \text{ odebrane} \,|\, b_1b_2...b_n \text{ nadane}\,) \;=\; \prod_{k=1}^{n} P(B'_k = b'_k \mid B_k = b_k)\,,$$

Prawdopodobieństwo otrzymania b'_k zależy tylko od tego co nadano jako b_k, a nie od liter wcześniej nadanych — kanał *bez pamięci*. Zachowanie kanału wyczerpująco opisuje macierz przejścia

$$p_{ij} \;=\; P(B'_k = i \mid B_k = j), \qquad i, j \in \mathcal{B}$$

niezależna od k — kanał *stacjonarny*.

Przykład $\mathcal{B} = \{0, 1, ..., 9\}$ oraz

$$P(i \text{ otrzymane} \,|\, j \text{ nadane}) \;=\; \begin{cases} \frac{3}{4} & \text{gdy } i = j \\ \frac{1}{4} & \text{gdy } i = j + 1 (\mathrm{mod}\, 10) \\ 0 & \text{w przeciwnym razie.} \end{cases}$$

Prawdopodobieństwo bezbłędnego otrzymania $b_1b_2...b_n$ wynosi $(\frac{3}{4})^n$, np. dla $n = 4$ to zaledwie 0.32.

Majority vote: Jeśli prześlemy każdą cyfrę 3-krotnie, prawdopodobieństwo tego, że co najmniej 2 kopie zostaną odebrane bez błędu wynosi

$$(\tfrac{3}{4})^3 + 3(\tfrac{3}{4})^2(\tfrac{1}{4}) \;=\; \tfrac{27}{32} \;\approx\; 0.844 \;>\; 0.75$$

Wtedy $(\frac{27}{32})^4 \approx 0.51$ za cenę 3-krotnego wydłużenia komunikatu.

W tym wypadku można poprawić kodowanie przez użycie mniejszego alfabetu $\mathcal{C} \subset \mathcal{B}$, $\mathcal{C} = \{0, 2, 4, 6, 8\}$, kodując cyfry $\mathcal{B}$ za pomocą słów $\mathcal{C}^*$. Przekłamanie znaku $\mathcal{C}$ przez ten kanał jest teraz *nieistotne*: jeśli odbieramy 0 lub 1, z prawdopodobieństwem 1 oznacza to, że nadano 0, jeśli odbieramy 2 lub 3, na pewno nadano 2, itd.

Poprzednio twierdzenie Shannona-Fano określało otymalną średnią długość słowa kodowego $c : \mathcal{A} \to \mathcal{B}^*$ (koszt kodowania informacji) zależnie od entropii źródła $H(A)$. Teraz do niepewności $H(A)$ dochodzi niepewność co do poprawności przekazu $H(B'|B)$ albo $H(B|B')$ — te entropie nie muszą być takie same, zależy to od symetrii $P(B'|B)$. Entropia kanału także wpływa na średnią długość transmitowanego kodu gwarantującego odpowiedni poziom poprawności przekazu. Jest to przedmiotem twierdzenia Shannona o kodowaniu w obecności szumu.

Nasza dyskusja transmisji w obecności szumu dotyczyć będzie kolejno:

- podejścia kombinatorycznego
- opisu geometrycznego
- zastosowania metod algebraicznych
- aspektów teorio-informacyjnych

Rozkład Bernoulliego

Prawd. błędnego przekazu pojedynczego $b \in \mathcal{B}$ wynosi p, poprawnego $1 - p$.

Prawdopodobieństwo, że w serii n transmisji wystąpiło dokładnie k błędów

$$P(n, k) \;=\; \binom{n}{k} p^k (1-p)^{n-k}$$

... co najmniej k błędów

$$\sum_{m=k}^{n} \binom{n}{m} p^m (1-p)^{n-m}$$

Symetryczny kanał binarny

$$\mathcal{B} = \{0,1\}, \qquad P \;=\; \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}, \qquad 0 < p < 0.5.$$

Prawdopodobieństwo przekłamania przynajmniej jednego bitu dla różnych p i n

	$n=4$	$n=6$	$n=8$	$n=16$
0.01	0.039	0.059	0.077	0.149
0.05	0.185	0.265	0.337	0.560
0.10	0.344	0.469	0.570	0.815

To samo przy przesyłaniu 3 kopii każdego bitu (majority vote)

	$n=4$	$n=6$	$n=8$	$n=16$
0.0003	0.0012	0.0018	0.0024	0.0048
0.0073	0.0287	0.0427	0.0565	0.1099
0.0280	0.1074	0.1567	0.2032	0.3652

To samo (prawdopodobieństwo niewykrycia błędu) z przesyłaniem dodatkowego 1 bitu kontrolnego

	$n=4$	$n=6$	$n=8$	$n=16$
0.01	0.0010	0.0020	0.0026	0.0034
0.05	0.0215	0.0408	0.0635	0.1653
0.10	0.0734	0.1266	0.1797	0.3445

Przesyłamy bity $b_1 b_2 ... b_n$ dodając 1 bit parzystości

$$b_{n+1} \;=\; b_1 + b_2 + \cdots + b_n \pmod 2$$

Kontrola po stronie odbiorcy

$$b'_1 + b'_2 + \cdots + b'_n + b'_{n+1} \;=\; 0 \pmod 2$$

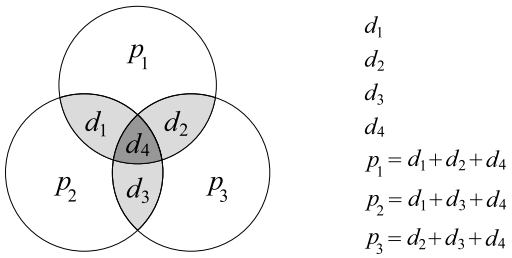
Niezgodność oznacza przekłamanie 1, 3, ... bitów — ponowna transmisja.

Kod ISBN 13 (International Standard Book Number): ISBN 978-83-271-6008-9

$$c_{13} \;=\; \left[10 \;-\; \Big(c_1 + \cdots + c_{11} + 3(c_2 + \cdots + c_{12})\Big) \pmod{10}\right] \pmod{10}$$

Kody Hamminga

Kod $H(7,4)$ złożony z 4 bitów użytkowych i 3 bitów kontrolnych



Kodowanie można wyrazić w formie macierzowej: $\boldsymbol{b} = \boldsymbol{C}\boldsymbol{d}$,

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \end{bmatrix}$$

Można sprytniej ułożyć bity w wektorze $\boldsymbol{b} = [p_1\ p_2\ d_1\ p_3\ d_2\ d_3\ d_4]^T$, tj. zmienić kolejność wierszy w macierzy kodującej $\boldsymbol{C}$. Wtedy

$$\boldsymbol{S}\boldsymbol{b} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \boldsymbol{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

bo $\boldsymbol{S}\boldsymbol{C} = \boldsymbol{0}$. Jeśli przekłamaniu ulegnie bit i-ty, $\boldsymbol{b}' = \boldsymbol{b} + \boldsymbol{e}_i$, otrzymamy

$$\boldsymbol{S}\boldsymbol{b}' = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix}$$

Ciąg $s_3 s_2 s_1$ jest wtedy binarnym numerem przekłamanego bitu.

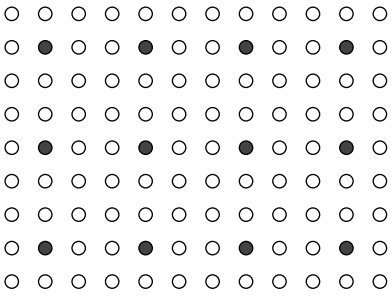
Hamming podał systematyczną metodę konstruowania takich kodów różnej długości $H(n,k)$ (p. np. Wikipedia).

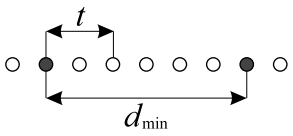
Odległość Hamminga

Odległość w przestrzeni ciągów binarnych $\mathcal{B}^n = \{0,1\}^n$

$$d(\boldsymbol{x},\boldsymbol{y}) \;=\; \sum_{i=1}^{n} x_i \oplus y_i$$

Wśród ciągów $\mathcal{B}^n$ wybieramy podzbiór słów kodowych $\mathcal{C} \subset \mathcal{B}^n$ — np. kody Hamminga, w których tylko 4 bity spośród 7 są niezależne, a pozostałe obliczane jako sumy kontrolne.





Jeśli minimalna odległość Hamminga między słowami kodowymi wynosi $d_{\min}$, wówczas kod jest w stanie wykryć przekłamanie nie więcej niż $d_{\min} - 1$ bitów. Przekłamanie nie więcej niż $t = \left\lfloor \dfrac{d_{\min} - 1}{2} \right\rfloor$ może być skorygowane.

Bit parzystości dodany do n-bitowych słów: $d_{\min} = 2,\ \ t = 0.$
Kod Hamminga $H(7,4)$: $d_{\min} = 3,\ \ t = 1.$
Kod Hamminga $H(7,4)$ zabezpieczony bitem parzystości: $d_{\min} = 4,\ \ t = 1.$

4 bity danych bez ochrony (b), w postaci kodu Hamminga (h), z bitem parzystości (hp)

	b+	b−	h+	h±	h−	hp−
0.01	0.961	0.039	0.998	0.99997	0.00003	0.0000007
0.05	0.815	0.185	0.956	0.996	0.004	0.0004
0.10	0.656	0.344	0.850	0.974	0.026	0.0050

b+, h+: prawdopodobieństwo odbioru poprawnego (skorygowanego) słowa
h±: jak wyżej lub prawdopodobieństwo wykrycia niekorygowalnego błędu
b−, h−, hp−: prawdopodobieństwo niewykrycia błędu

n — liczba bitów, k — liczba bitów użytkowych, t — liczba korygowanych błędów

W otoczeniu każdego słowa kodowego $\boldsymbol{x}$, w odległości $\leq t$ znajduje się

$$V_n(t) = \binom{n}{0} + \binom{n}{1} + \cdots + \binom{n}{t}$$

słów, które można zdekodować jako $\boldsymbol{x}$. Liczba wszystkich słów 2^n nie może być mniejsza niż liczba słów kodowych 2^k razy objętość kuli $V_n(t)$ wokół każdego z nich,

$$2^n \;\geq\; 2^k \sum_{j=0}^{t} \binom{n}{j} \qquad \Rightarrow \qquad n-k \;\geq\; \log_2 \sum_{j=0}^{t} \binom{n}{j} \qquad (*)$$

Dla $t=1$: $n-k \;\geq\; \log(1+n)$, a więc jeśli $n = 2^q - 1$

$$2^q - 1 - k \;\geq\; \log 2^q \;=\; q \qquad \Rightarrow \qquad k \;\leq\; 2^q - q - 1\,.$$

q	$n = 2^q - 1$	$k \leq n - q$
3	7	4
4	15	11
5	31	26
6	63	57

Dla korekty 2 bitów: $2^{n-k} \geq 1 + n + \frac{n(n-1)}{2}$ czyli

$$n + 1 - \log_2(n^2 + n + 2) \;\geq\; k\,,$$

n	16	24	32	64
k	8	15	22	52

Gwarancja skuteczności przy poziomie szumu p:

liczba bitów korygowanych $\;\approx\;$ $p \cdot$ liczba bitów przesyłanych

czyli $t \approx np$. Stąd oraz z $(*)$

$$k \;\lessapprox\; n(1 - p + p\log p)$$

Liniowe kody cykliczne

Przestrzeń $\mathcal{B}^n$ traktujemy jako przestrzeń liniową $\mathbb{F}_2^n$: wetkory o składowych $\{0,1\}$ z działaniami arytmetycznymi mod 2. W szczególności

$$\boldsymbol{x} \circ \boldsymbol{y} \;=\; \sum_{i-1}^{n} x_i y_i \;\; (\mathrm{mod}\, 2)\,.$$

Kodowanie z sumami kontrolnymi opisujemy za pomocą operacji macierzowych

$$\boldsymbol{x} \;\to\; \boldsymbol{C}\boldsymbol{x} \;=\; \left[\begin{array}{c} 1 \\ \ddots \\ 1 \\ \hline \\ \boldsymbol{A} \\ \\ \end{array}\right] \left[\begin{array}{c} x_1 \\ \vdots \\ x_k \end{array}\right] \;=\; \left[\begin{array}{c} \boldsymbol{x} \\ \hline \\ \boldsymbol{A}\boldsymbol{x} \\ \\ \end{array}\right] \;=\; \boldsymbol{y}$$

$\boldsymbol{C} : \mathbb{F}_2^k \to \mathbb{F}_2^n$ jest macierzą $n \times k$, a $\boldsymbol{A}$ ma rozmiar $(n-k) \times k$.

Macierz syndromu $\boldsymbol{S}$ kontroluje poprawność transmisji:

$$\boldsymbol{S} : \mathbb{F}_2^n \to \mathbb{F}_2^{n-k}, \qquad \boldsymbol{S}\boldsymbol{y} \;=\; \boldsymbol{0} \quad \text{gdy } \boldsymbol{y} \text{ bezbłędne}$$

czyli $\mathrm{Ker}\,\boldsymbol{S} \;=\; \mathrm{Im}\,\boldsymbol{C} \;=\; \mathcal{C}$. Jeśli $\boldsymbol{C} = \left[\begin{array}{c} \boldsymbol{I}_k \\ \hline \boldsymbol{A} \end{array}\right]$, to $\boldsymbol{S} = \left[\; -\boldsymbol{A} \;\mid\; \boldsymbol{I}_{n-k} \;\right]$, bo

$$\mathrm{Ker}\,\boldsymbol{S} \;=\; \left\{ \left[\begin{array}{c} \boldsymbol{u} \\ \boldsymbol{v} \end{array}\right] : \;\; \boldsymbol{S} \left[\begin{array}{c} \boldsymbol{u} \\ \boldsymbol{v} \end{array}\right] = -\boldsymbol{A}\boldsymbol{u} + \boldsymbol{v} = \boldsymbol{0} \right\} \;=\; \left\{ \left[\begin{array}{c} \boldsymbol{u} \\ \boldsymbol{A}\boldsymbol{u} \end{array}\right] : \;\; \boldsymbol{u} \in \mathbb{F}_2^k \right\} \;=\; \mathrm{Im}\,\boldsymbol{C}\,.$$

Metryka Hamminga jest translacyjnie niezmiennicza: $d(\boldsymbol{y}, \boldsymbol{y}') = d(\boldsymbol{0}, \boldsymbol{y} - \boldsymbol{y}')$. Stąd

$$d_{\min} \;=\; \min_{\boldsymbol{0} \neq \boldsymbol{y} \in \mathrm{Im}\boldsymbol{C}} d(\boldsymbol{0}, \boldsymbol{y})\,.$$

Dla dowolnego $\boldsymbol{y} \in \mathbb{F}_2^n$ wielkość $d(\boldsymbol{0}, \boldsymbol{y})$ nazywamy *wagą* wektora $\boldsymbol{y}$.

Dekodowanie z korektą błędów

Niech $\boldsymbol{x} \to \boldsymbol{C}\boldsymbol{x} = \boldsymbol{y} \longrightarrow \boldsymbol{y}' \neq \boldsymbol{C}\boldsymbol{x}$. A więc $\boldsymbol{S}\boldsymbol{y}' = \boldsymbol{u} \neq \boldsymbol{0}$.

Szukamy takiego $\boldsymbol{z} \in \boldsymbol{S}^{-1}(\boldsymbol{u})$, które ma najmniejszą wagę. Można to zrobić na początku, przed transmisją, katalogując takie $\boldsymbol{z}$ dla wszytskich możliwych wartości $\boldsymbol{u} \in \mathbb{F}_2^{n-k}$. Oznaczmy takie optymalne $\boldsymbol{z}$ przez $\boldsymbol{z}_0(\boldsymbol{u})$. Mamy oczywiście $\boldsymbol{S}(\boldsymbol{z}_0(\boldsymbol{u})) = \boldsymbol{u}$.

Dekodowanie błędnego $\boldsymbol{y}'$:

$$\boldsymbol{y}_0 \;=\; \boldsymbol{y}' - \boldsymbol{z}_0(\boldsymbol{S}\boldsymbol{y}')\,.$$

Mamy

$$\boldsymbol{S}\boldsymbol{y}_0 \;=\; \boldsymbol{S}(\boldsymbol{y}' - \boldsymbol{z}_0(\boldsymbol{u})) \;=\; \boldsymbol{S}\boldsymbol{y}' - \boldsymbol{S}(\boldsymbol{z}_0(\boldsymbol{u})) \;=\; \boldsymbol{u} - \boldsymbol{u} = \boldsymbol{0}\,,$$

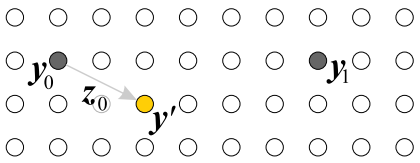
a zatem $\boldsymbol{y}_0 \in \mathcal{C}$.

Przekonajmy się, że $\boldsymbol{y}_0$ jest słowem kodowym w $\mathcal{C}$ o najmniejszej odległości od $\boldsymbol{y}'$. Niech $\boldsymbol{y}_1$ będzie innym słowem kodowym w $\mathcal{C}$, $\boldsymbol{y}_1 = \boldsymbol{C}\boldsymbol{x}_1$. Wtedy

$$\boldsymbol{S}(\boldsymbol{y}' - \boldsymbol{y}_1) \;=\; \boldsymbol{S}\boldsymbol{y}' - \boldsymbol{S}\boldsymbol{y}_1 \;=\; \boldsymbol{u} - \boldsymbol{0} \;=\; \boldsymbol{u}\,,$$

a więc $\boldsymbol{y}' - \boldsymbol{y}_1 \in \boldsymbol{S}^{-1}(\boldsymbol{u})$. Zatem

$$d(\boldsymbol{0},\, \boldsymbol{z}_0(\boldsymbol{u})) \;\leq\; d(\boldsymbol{0},\, \boldsymbol{y}' - \boldsymbol{y}_1) \qquad \Rightarrow \qquad d(\boldsymbol{y}',\, \boldsymbol{y}_0) \;\leq\; d(\boldsymbol{y}',\, \boldsymbol{y}_1)\,.$$



Jeśli $d_{\min}$ jest małe, to możliwych $\boldsymbol{z}_0$ w $\boldsymbol{S}^{-1}(\boldsymbol{u})$ do sprawdzenia jest niewiele, ale nie można też wtedy korygować wielu błędów. Gdy odległości słów w $\mathcal{C}$ rosną, pojemność $\boldsymbol{S}^{-1}(\boldsymbol{u})$ rośnie podobnie jak objętości kul $V(n,d) \simeq \binom{n}{d}$, a więc wykładniczo względem d. Specjalna struktura przestrzeni słów kodowych $\mathcal{C}$ pozwala uprościć znacznie złożoność obliczania $\boldsymbol{z}_0(\boldsymbol{u})$. Taką własnośc mają kody BCH.

Cykliczne kody liniowe nad $\mathbb{Z}_2$

Kody cykliczne: $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C} \subseteq \mathbb{F}^n$

i) $\boldsymbol{x} + \boldsymbol{y} \in \mathcal{C}$

ii) cykliczne przesunięcie bitów w $\boldsymbol{x}$, $C\boldsymbol{x} \in \mathcal{C}$

Przykład

$$\begin{array}{cccccccccccccc}
1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\
0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\
0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1
\end{array}$$

do tego wektor samych zer (jako wynik $\boldsymbol{x} + \boldsymbol{x}$), razem 16 z 32 elementów $\mathbb{F}_2^5$.

To samo przy pomocy wielomianów w $\mathbb{F}[x]$ (dokładniej z $\mathbb{F}[x]/(x^n - 1)$):

$$g(x) = x + 1, \quad xg(x) = x^2 + x, \quad x^2g(x) = x^3 + x^2, \quad x^3g(x) = x^4 + x^3$$

Przeniesienie cykliczne (w $\mathbb{F}_2$ $-1 = +1$):

$$x^4g(x) \;=\; x^5 + x^4 \,(\mathrm{mod}\, x^5 - 1) \;=\; x^5 + x^4 + 1 - 1 \,(\mathrm{mod}\, x^5 - 1) \;=\; x^4 + 1$$

Ogólnie $g(x) = g_k x^k + g_{k-1}x^{k-1} + \cdots + g_1 x + g_0, \quad k = n - m$

$$xg(x) = g_k x^{k+1} + g_{k-1}x^k + \cdots + g_1 x^2 + g_0 x \,(\mathrm{mod}\, x^n - 1)$$

$g(x) \to [g_0\ g_1\ \ldots\ g_k\ 0\ \ldots 0], \qquad xg(x) \to [0\ g_0\ g_1\ \ldots\ g_k\ 0\ \ldots\ 0]$

Przesunięcie o m bitów: $\quad x^m g(x) \,(\mathrm{mod}\, x^n - 1) \;\to\; [g_k\ 0\ \ldots\ 0\ g_0\ \ldots\ g_{k-1}]$

$$\begin{aligned}
x^m g(x) &= g_k x^{m+k} + g_{k-1}x^{m+k-1} + \cdots + g_1 x^{m+1} + g_0 x^m \,(\mathrm{mod}\, x^n - 1) \\
&= g_k(x^n - 1) + g_{k-1}x^{n-1} + \cdots + g_1 x^{m+1} + g_0 x^m + g_k \,(\mathrm{mod}\, x^n + 1) \\
&= g_{k-1}x^{n-1} + g_{k-2}x^{n-2} + \cdots + g_1 x^{m+1} + g_0 x^m + 0x^{m-1} + \cdots 0x + g_k
\end{aligned}$$

Zgodnie z (i) i (ii) jeśli $g(x) \in \mathcal{C}$ to także $x^i g(x) + x^j g(x) \,(\mathrm{mod}\, x^n - 1) \in \mathcal{C}$,

czyli ogólnie $u(x) \cdot g(x) \;=\; (u_p x^p + u_{p-1}x^{p-1} + \cdots + u_0) \cdot g(x) \,(\mathrm{mod}\, x^n - 1) \in \mathcal{C}$.

Inną przestrzeń cykliczną $\mathcal{C}$ otrzymamy jeśli wyjdziemy od innego *wektora generującego*, np. $\boldsymbol{x} = [11100]$ (nie należy do poprzedniej przestrzeni $\mathcal{C}$). To samo otrzymamy równoważnie biorąc $g(x) = x^2 + x + 1$ jako generator przestrzeni cyklicznej.

Twierdzenie: *Niech $\mathcal{C} \neq \{\mathbf{0}\}$ będzie cyklicznym kodem liniowym długości n. Wówczas istnieje dokładnie jeden wielomian $g(x) \in \mathcal{C}$ minimalnego stopnia k, taki że*

i) $g(x),\ xg(x),\ \ldots,\ x^{n-k-1}g(x)$ *tworzą bazę przestrzeni słów kodowych $\mathcal{C}$ (wymiaru $n-k$)*

ii) $\mathcal{C} = \{p(x) \cdot g(x) \pmod{(x^n - 1)} : \quad p(x) \in \mathbb{F}[x]\}$

iii) $g(x)$ *jest podzielnikiem* $x^n - 1$.

Kody cykliczne odpowiadają ideałom w pierścieniu wielomianów $\mathcal{I} \subseteq \mathbb{F}[x]$, tj. takim zbiorom, dla których jeśli $p(x) \in \mathcal{I}$, wówczas także $p(x)q(x) \in \mathcal{I}$ dla dowolnego $q(x)$.

DOWÓD: Jako $g(x)$ wybierzmy wielomian odpowiadający dowolnemu niezerowemu wektorowi $\boldsymbol{y}$ z $\mathcal{C}$, dla którego $y_k = 1$ i $y_{k+1} = y_{k+2} = \ldots = y_{n-1} = 0$, przy czym k jest minimalne. A więc g ma stopień k. Łatwo zauważyć, że wektory $g(x),\ xg(x),\ \ldots,\ x^{n-k}g(x)$ należą do $\mathcal{C}$ (cykliczność) i są liniowo niezależne. Kombinacje liniowe tych wektorów można zapisać w postaci

$$\mathcal{C}_0 \ = \ \{p(x) \cdot g(x) : \quad p(x) \text{ jest stopnia } \leq n-k\}\,.$$

Oczywiście $\mathcal{C}_0 \subseteq \mathcal{C}$. Weźmy dowolne $c(x) \in \mathcal{C}$ stopnia $\geq k$. Wtedy

$$c(x) \ = \ p(x) \cdot g(x) + r(x), \qquad \operatorname{st} r(x) \ < \ k\,, \quad \operatorname{st} p(x) \ \leq \ n-k\,.$$

Ponieważ $r(x) = c(x) - p(x)g(x)$, zatem $r(x) \in \mathcal{C}$ i ma stopień mniejszy niż $g(x)$. Jedyna możliwość to $r(x) \equiv 0$, a więc $\mathcal{C}_0 = \mathcal{C}$.

Niech teraz $x^n - 1 \ = \ q(x) \cdot g(x) + s(x)$, i znów $\operatorname{st} s(x) < k$. Wtedy

$$s(x) \ = \ -q(x) \cdot p(x) \pmod{(x^n - 1)}$$

czyli $s(x) \in \mathcal{C}$ i ma stopień niższy niż k?! Zatem $s(x) \equiv 0$, a więc $g(x)$ jest podzielnikiem $x^n - 1$.

Różne ideały w $\mathbb{F}[x]/(x^n-1)$ odpowiadają różnym podzielnikom $g(x)$ wielomianu $x^n - 1$.

Niech $x^n - 1 = g(x) \cdot h(x)$, przy czym $\operatorname{st} g(x) = k$, $\operatorname{st} h(x) = m = n - k$. Wtedy

$$\mathcal{C} \;=\; \{u(x) \cdot g(x) \pmod{x^n - 1}, \quad u(x) \in \mathbb{F}[x]\}$$

$u(x)$ – wiadomość do przesłania
$w(x) = u(x) \cdot g(x) \pmod{x^n + 1}$ – komunikat zakodowany w $\mathcal{C}$ (m bitów użytkowych, k kontrolnych)
$\tilde{w}(x)$ – komunikat odebrany
$s(x) = \tilde{w}(x) \cdot h(x) \pmod{x^n - 1}$ – kontrola parzystości (syndrom)

Jeśli $s(x) = 0$, przyjmujemy $\tilde{w}(x)$ jako bezbłędny, jeśli nie, $s(x)$ zawiera informację o położeniu błędnych bitów.

Jeśli $\tilde{w}(x) = w(x)$

$$s(x) = w(x) \cdot h(x) = u(x) \cdot g(x) \cdot h(x) = u(x) \cdot (x^n - 1) = 0 \pmod{x^n - 1}$$

Jeśli $\tilde{w}(x) = w(x) + e(x)$ (e – wektor błędu np. [0010010])

$$\begin{aligned} s(x) \;&=\; \tilde{w}(x) \cdot h(x) \;=\; (w(x) + e(x)) \cdot h(x) \\ &=\; u(x) \cdot g(x) \cdot h(x) + e(x) \cdot h(x) \;=\; e(x) \cdot h(x) \pmod{x^n - 1} \end{aligned}$$

Stąd $e(x) = s(x) \cdot \big(h(x)\big)^{-1} \pmod{x^n - 1}$. Jednak $\mathbb{F}[x]/(x^n - 1)$ nie jest ciałem, bo $x^n - 1$ jest wielomianem rozkładalnym, a na dodatek $h(x)$ jest jego podzielnikiem.

Kodowanie informacji w notacji macierzowej

$$[u_0\ u_1\ \ldots\ u_m]\cdot\begin{bmatrix} g_0 & g_1 & \ldots & g_k & 0 & \ldots & 0 \\ 0 & g_0 & g_1 & \ldots & g_k & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & \ldots & 0 & g_0 & g_1 & \ldots & g_k \end{bmatrix} = [w_0\ w_1 \quad \ldots \quad w_n]$$

Kontrola parzystości

$$\begin{bmatrix} 0 & \ldots & 0 & h_m & \ldots & h_1 & h_0 \\ 0 & \ldots & h_m & \ldots & h_1 & h_0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ h_m & \ldots & h_1 & h_0 & 0 & \ldots & 0 \end{bmatrix}\cdot\begin{bmatrix} \tilde{w}_0 \\ \tilde{w}_1 \\ \\ \vdots \\ \\ \tilde{w}_n \end{bmatrix} = \begin{bmatrix} s_0 \\ s_1 \\ \vdots \\ s_k \end{bmatrix}$$

$$\begin{bmatrix} 0 & \ldots & 0 & h_m & \ldots & h_1 & h_0 \\ 0 & \ldots & h_m & \ldots & h_1 & h_0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ h_m & \ldots & h_1 & h_0 & 0 & \ldots & 0 \end{bmatrix}\cdot\left(\begin{bmatrix} w_0 \\ w_1 \\ \\ \vdots \\ \\ w_n \end{bmatrix} + \begin{bmatrix} e_0 \\ e_1 \\ \\ \vdots \\ \\ e_n \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + \begin{bmatrix} s_0 \\ s_1 \\ \vdots \\ s_k \end{bmatrix}$$

czyli $\boldsymbol{H}\cdot\boldsymbol{e} = \boldsymbol{s}$, a wyznaczenie $\boldsymbol{e}$ to rozwiązanie takiego wieloznacznego układu równań w $\mathbb{F}_2$.

Jeśli znajdziemy wszystkie $\{\boldsymbol{e}^{(1)}, \ldots, \boldsymbol{e}^{(r)}\}$ spełniające ten układ, wybieramy "najbardziej prawdopodobny", czyli ten o najmniejszej liczbie jedynek (minimalnej wadze Hamminga).

Problem: znaczna liczebność tego zbioru, $r \simeq V(n,d) \simeq \binom{n}{d}$.

Cykliczne kody 7-bitowe

$$x^7 + 1 \;=\; (x+1)(x^3+x+1)(x^3+x^2+1)$$

$g(x)$	$h(x)$	NAZWA
$x+1$	$x^6+x^5+x^4+x^3+x^2+x+1$	kod z bitem parzystości
x^3+x+1	x^4+x^2+x+1	kod Hamminga
x^3+x^2+1	$x^4+x^3+x^2+1$	kod Hamminga
$x^4+x^3+x^2+1$	x^3+x^2+1	dualny kod Hamminga
x^4+x^2+x+1	x^3+x+1	dualny kod Hamminga
$x^6+x^5+x^4+x^3+x^2+x+1$	$x+1$	kod powtórzeniowy

Technika CRC (cyclic redundancy check)

Zastosowania: zapis na płytach CD, transmisja USB, Bluetooth, sieci komórkowe itp.

Wybieramy wielomian generujący $g(x)$ stopnia k – jest to liczba bitów kontrolnych. Zakładamy, że wysyłać będziemy bloki n bitowe, co daje $m = n - k$ bitów użytkowej informacji na blok. Można wybrać n dowolnie duże, ale oczywiście skuteczność ochrony dłuższych bloków przez k bitów kontrolnych będzie się zmniejszać wraz ze wzrostem n. Jeśli CRC ma umożliwiać nie tylko detekcję, ale także automatyczną korektę pewnej liczby bitów, trzeba respektować nierówność

$$2^n \;\geq\; 2^{n-k} \sum_{p=0}^{t} \binom{n}{p}.$$

Np. jeden z kodów CRC używany w transmisji USB przesyła 16 bitowe pakiety z 5 bitami kontrolnymi, umożliwia to detekcję max 4 przekłamanych bitów, ale korektę tylko jednego przekłamania. W najprostszej postaci obliczenia wykonuje się tak

$w(x)$ – dane do przesłania
$g(x)$ – wielomian generujący CRC stopnia k
$r(x) = x^k \cdot w(x) \pmod{g(x)}$ – wartość sumy kontrolnej
$W(x) = x^k \cdot w(x) + r(x)$ – przesyłany pakiet
$s(x) = \tilde{W}(x) \pmod{g(x)}$ – kontrola parzystości: jeśli $s(x) = 0$, OK!

Przykład: Zakodować ciąg bitów 111001111 z wykorzystaniem CRC o generatorze 1011.

$$111001111\,000 \pmod{1011} = 110$$

Wysyłamy komunikat $111001111\,110$. Po odbiorze wykonujemy dzielenie

$$111001111\,110 \pmod{1011} = 0$$

Jeśli wynik różny od 0, ponowna transmisja pakietu.

Standardy CRC: kod CRC jest zapisywany jako `CRC-k-XXX/nnnn`, gdzie k jest stopniem wielomianu kontrolnego, `XXX` jest nazwą standardu, a `nnnn` jest numerycznym kodem (w postaci binarnej, ósemkowej lub szesnastkowej) określającym współczynniki wielomianu. Nie podaję się bitu przy k-tej potędze x bo on ma domyslnie wartość 1. Np.

`CRC-6-GSM/0x2F`, `0x2F=0b101111` $g(x) = x^6 + \underline{x^5 + x^3 + x^2 + x + 1}$

`CRC-8-Bluetooth/0xA7`
`0xA7=0b10100111` $g(x) = x^8 + \underline{x^7 + x^5 + x^2 + x + 1}$

`CRC-16-IBM/0x8005`
`0x8005=0b1000000000000101` $g(x) = x^{16} + \underline{x^{15} + x^2 + 1}$

Kody BCH

Zastosowanie metod algebry abstrakcyjnej do optymalnego wyznaczania korekty błędu na podstawie syndromu $\boldsymbol{Sy'} = \boldsymbol{u} \neq \mathbf{0}$.

Algebraiczne rozszerzenia ciała liczbowego

1. Pierścień wielomianow $\mathbb{F}[x]$ nad ciałem $\mathbb{F}$
2. Wielomian nierozkładalny w $\mathbb{F}[x]$

$$w(x) \neq q(x) \cdot p(x), \qquad \text{gdzie } \ 1 < \operatorname{st} p, \operatorname{st} q < \deg w$$

3. Pierścień ilorazowy $\mathbb{K} = \mathbb{F}[x]/w(x)$ jest ciałem:

$$\forall\, u(x) \neq \mathbf{0} \ \ \exists!\, v(x) \quad u(v) \cdot v(x) \equiv \mathbf{1} \pmod{w(x)}$$

   czyli $v(x) = u(x)^{-1}$ w $\mathbb{K}$.

4. $\mathbb{K}$ jest ciałem Galois $G\mathbb{F}(p^k)$, gdzie p jest charakterystyką ciała $\mathbb{F}$ ($\mathbb{F} = \mathbb{Z}_p$), a k stopniem $w(x)$.
5. $G\mathbb{F}(p^k)$ jest ciałem cyklicznym, tj. istnieje $\alpha \in G\mathbb{F}$ takie, że

$$1 = \alpha^0, \quad \alpha, \quad \alpha^2, \quad \ldots, \quad \alpha^k = 1$$

   obejmuje wszystkie elementy $\mathbb{F}$.

6. Generator α jest pierwiastkiem wielomianu $w(x)$.

PRZYKŁAD: $\mathbb{F} = \mathbb{R}$, $w(x) = x^2 + 1$, $\mathbb{K} = \mathbb{R}[x]/(x^2+1)$.

$\mathbb{K} = \{u(x) :\ u(x)$ jest możliwą postacią reszty modulo $(x^2+1)\} = \{ax+b :\ a, b \in \mathbb{R}\}$

Zapis uproszczony: $ax + b \in \mathbb{K} \quad \Leftrightarrow \quad [a, b]$.

Dodawanie w $\mathbb{K}$:

$$(ax + b) \ + \ (cx + d) \ (\mathrm{mod}\,(x^2 + 1)) \ = \ (a + c)x \ + \ (b + d)$$

czyli $[a, b] + [c, d] = [a + c,\ b + d]$. Stąd mamy też $-[a, b] = [-a, -b]$.

Monożenie:

$$\begin{aligned}(ax + b) \cdot (cx + d) \ &= \ (acx^2 + (ad + bc)x + bd) \ (\mathrm{mod}\,(x^2 + 1)) \\ &= \ (acx^2 + (ad + bc)x + bd) \ - \ ac(x^2 + 1) \\ &= \ (ad + bc)x \ + \ (bd - ac)\,,\end{aligned}$$

czyli $[a, b] \cdot [c, d] = [ad + bc,\ bd - ac]$. Można łatwo sprawdzić, że

$$[a, b]^{-1} \ = \ \left[\frac{a}{\sqrt{a^2 + b^2}},\ \frac{-b}{\sqrt{a^2 + b^2}}\right].$$

Zauważmy, że działania na elementach postaci $[0, b]$ odpowiadają zwykłym działaniom na liczbach $\mathbb{R}$:

$$[0, b] + [0, d] = [0, b + d], \qquad [0, b] \cdot [0, d] = [0, bd]\,,$$

a zatem $[0, b] \to b$. Zgodnie z regułami działania w $\mathbb{K}$

$$[1, 0] \cdot [1, 0] \ = \ [1 \cdot 0 + 1 \cdot 0,\ 0 \cdot 0 - 1 \cdot 1] \ = \ [0, -1]\,,$$

a więc $[1, 0]^2 = [0, -1] \to -1$ czyli $w([1, 0]) = [1, 0]^2 + [0, 1] = [0, 0]$

... Wniosek: $\mathbb{K} = \mathbb{C}, \quad [a, b] \to b + ai$.

Wielomiany nierozkładalne nad $\mathbb{Z}_2$

st. 1: $X, \quad X+1$

st. 2: X^2+X+1

st. 3: $X^3+X+1, \quad X^3+X^2+1$

st. 4: $X^4+X+1, \quad X^4+X^3+1, \quad X^4+X^3+X^2+X+1$

Wielomiany nierozkładalne nad $\mathbb{Z}_3$

st. 1: $X, \quad X+1, \quad X+2$

st. 2: $X^2+1, \quad X^2+X+2, \quad X^2+2X+2$

st. 3: $X^3+2X+1, \quad X^3+2X+2, \quad X^3+X^2+2,$
$X^3+X^2+X+2, \quad X^3+X^2+2X+1, \quad X^3+2X^2+1,$
$X^3+2X^2+X+1, \quad X^3+2X^2+2X+2$

Przykład: Tabelki działań na wielomianach w $\mathbb{Z}_3[X]/(X^2+1)$. Zawiera on wielomiany

- stopnia 0: 0, 1 i 2
- stopnia 1: X, X+1, X+2, 2X, 2X+1, 2X+2

Tabela dodawania dla $\mathbb{Z}_3[X]/(X^2+1)$ — wszystkie wielomiany stopnia mniejszego niż 2 naz $\mathbb{Z}_3$:

+	01	02	10	11	12	20	21	22
01	02	00	11	12	10	21	22	20
02	00	01	12	10	11	22	20	22
10	11	12	20	21	22	00	01	02
11	12	10	21	22	20	01	02	00
12	10	11	22	20	21	02	00	01
20	21	22	00	01	02	10	11	12
21	22	20	01	02	00	11	12	10
22	20	21	02	00	01	12	10	11

Dla mnożenia jest więcej pracy, np. dla $12 \cdot 21 = 20$ ponieważ

$(X+2)(2X+1) = 2X^2+2X+2 \quad \leftrightarrow \quad 2X^2+2X+2 \pmod{X^2+1} = 2X\,.$

$(aX+b)(cX+d) = acX^2+(ad+bc)X+bd \qquad \leftrightarrow \qquad [a\ b][c\ d] = [ac\ ad+bc\ bd] \pmod{p}$

Dzielenie modulo $X^2+1 \leftrightarrow [1\ 0\ 1]$

$$\begin{array}{ccccc} ac & ad+bc & bd & \\ 1 & 0 & 1 & \times ac \\ ac & 0 & ac & \\ \hline 0 & ad+bc & bd-ac & \end{array}$$

$$[a\ \ b][c\ \ d] \quad = \quad [ad+bc\ \ bd-ac]$$

$\times$	01	02	10	11	12	20	21	22
01	01	02	10	11	12	20	21	22
02	02	01	20	22	21	10	12	11
10	10	20	02	12	22	01	11	21
11	11	22	12	20	01	21	02	10
12	12	21	22	01	10	11	20	02
20	20	10	01	21	11	02	22	12
21	21	12	11	02	20	22	10	01
22	22	11	21	10	02	12	01	20

PRZYKŁAD Budowa kodu BCH w pierścieniu $\mathbb{Z}_2[X]/(x^n-1)$.

Wybieramy nieparzysty rozmiar kodu $n = m + k$ zgodnie z nierównościami Hamminga i projektowaną minimalną odlagłością słów kodowych d (zdolność detekcyjna $= d-1$, zdolność korekcyjna $= \lfloor \frac{d-1}{2} \rfloor$). W szczególności dla $n = 7$, $m = 3$, $k = 4$, $d = 2$, $t = 1$, z czynnikami nierozkładalnymi w $\mathbb{Z}_2[X]$:

$$X^7 + 1 \;=\; (X+1)(X^3+X+1)(X^3+X^2+1)$$

Tworzymy rozszerzenie Galois ciała $\mathbb{K} = \mathbb{Z}_2$ na bazie $w(X) = X^3 + X + 1$.

Możliwe postaci reszty mod $X^3 + X + 1$:

$$aX^2 + bX + c \quad \leftrightarrow \quad [a, b, c] \in \mathbb{Z}_2^3$$

Dodawanie w $\mathbb{K}[X]/(X^3 + X + 1)$:

$$[a, b, c] + [d, e, f] = [a + d, b + e, c + f] \,(\mathrm{mod}\, 2)\,.$$

Mnożenie: $(aX^2 + bX + c)(dX^2 + eX + f) \;=$
$$adX^4 + (ae+bd)X^3 + (af+be+cd)X^2 + (bf+ce)X + cf$$

$$[a\ b\ c] \cdot [d\ e\ f] \;=\; [ad \quad ae+bd \quad af+be+cd \quad bf+ce \quad cf]$$

Redukckja mod $X^3 + X + 1$:
$(aX^2 + bX + c)(dX^2 + eX + f) \;=\; (adX \;+\; ae{+}bd) \times (X^3 + X + 1) \;+\; r(X)$

$$X^3 + X + 1 \;\leftrightarrow\; [0\ 1\ 0\ 1\ 1], \qquad (X^3 + X + 1)X \;\leftrightarrow\; [1\ 0\ 1\ 1\ 0]$$

X^4	X^3	X^2	X	1	
ad	$ae+bd$	$af+be+cd$	$bf+ce$	cf	
1	0	1	1	0	$[1\ 0\ 1\ 1\ 0] \times ad$
0	1	0	1	1	$[0\ 1\ 0\ 1\ 1] \times (ae+bd)$
ad	0	ad	ad	0	
0	$ae+bd$	0	$ae+bd$	$ae+bd$	
0	0	$af{+}be{+}cd{-}ad$	$bf{+}ce{-}ad{-}ae{-}bd$	$cf{-}ae{-}bd$	

$$[a\ b\ c][d\ e\ f] \quad = \quad [af{+}be{+}cd{-}ad \quad bf{+}cd{-}ad{-}ae{-}bd \quad cf{-}ae{-}bd]$$

Np.

$$[1\ 0\ 1][1\ 1\ 1] = [1\ 1\ 0], \qquad [0\ 1\ 1][1\ 1\ 0] = [0\ 0\ 1], \qquad [1\ 1\ 1][1\ 1\ 1] = [0\ 1\ 1]$$

Mnożenie w $\mathbb{Z}_2[X]/(X^3+X+1)$

	001	010	011	100	101	110	111
001	001	010	011	100	101	110	111
010	010	100	110	011	001	111	101
011	011	110	101	111	100	001	010
100	100	011	111	110	010	101	001
101	101	001	100	010	111	011	110
110	110	111	001	101	011	010	100
111	111	101	010	001	110	100	011

Kwadraty i trzecie potęgi elementów

X	001	010	011	100	101	110	111
X^2	001	100	101	110	111	010	011
X^3	001	011	100	101	110	111	010

Policzymy wartości wielomianu generującego dla $\alpha \in \{000, 001, ..., 111\}$

$$F(\alpha) \;=\; \alpha^3 + \alpha + [001]$$

α	000	001	010	011	100	101	110	111
$F(\alpha)$	001	001	000	110	000	010	000	100

Policzymy potęgi pierwiastków $\alpha = 010,\ 100,\ 110$:

α	α^2	α^3	α^4	α^5	α^6	α^7	α^8
010	100	011	110	111	101	001	010
100	110	101	010	011	111	001	100
110	010	111	100	101	011	001	110

Weźmy $\alpha = [0,1,0]$. Łatwo policzyć z powyższego, że $\alpha^2 = [1,0,0]$ oraz $\alpha^3 = [0,1,1]$. Stąd

$$\alpha^3 + \alpha + 1 \;=\; [0,1,1] + [0,1,0] + [0,0,1] \;=\; [0,0,0] \pmod 2$$

a więc $w(\alpha) = 0$.

Wszystkie α^i, $i = 0, 1, ..., 6$ są różnymi pierwiastkami $y(X) = X^7 - 1$.

Konstrukcja kodu BCH

Znajdujemy wielomian $u(X)$ minimalnego stopnia k taki, że

$$u(\alpha) = u(\alpha^2) = \cdots u(\alpha^{d-1}) = 0$$

Jako przestrzeń słów kodowych wybieramy ideał generowany tym wielomianem

$$\mathcal{C} \;=\; \{p(x)u(x) \,(\mathrm{mod}\,(x^n-1)) : \quad p(x) \in \mathbb{Z}_2[x]\}$$

Czyli $w(x) \in \mathcal{C} \;\Leftrightarrow\; w(\alpha) = ... = w(\alpha^{d-1}) = 0$. Wtedy macierz syndromu $\boldsymbol{S}$ ma postać

$$\begin{bmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \vdots & & & \dots & \vdots \\ 1 & \alpha^{d-1} & \alpha^{2(d-1)} & \dots & \alpha^{(n-1)(d-1)} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{n-1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

bo w wierszu i-tym mamy $w(\alpha^i) = 0$. Jeśli otrzymamy $\tilde{w}(x)$ z błędami, wówczas nie wszystkie $\tilde{w}(\alpha^i) = 0$. Niech indeksy i, na których wystąpiło przekłamanie tworzą zbiór $\mathcal{E} = \{i_1, i_2, \dots, i_t\}$, przy czym $t \le \lfloor \frac{d-1}{2} \rfloor$. Wektor błędu odpowiada wielomianowi $e(X) = \sum_{i\in\mathcal{E}} X^i$ o własności $\tilde{w}(X) = w(X) + e(X)$.

Ponieważ $w \in \mathcal{C}$, dla $e(X)$ mamy

$$e(\alpha) = \tilde{w}(\alpha), \quad e(\alpha^2) = \tilde{w}(\alpha^2), \quad e(\alpha^{d-1}) = \tilde{w}(\alpha^{d-1})\,.$$

Jeśli więc wszystkie $e(\alpha^j) = 0$, to $\tilde{w}(\alpha^j) = w(\alpha^j) = 0$ i nie ma błędów, albo wystąpiło ich co najmniej d.

Konstruujemy *fuknc ję lokalizacji błędów* $(t = \lfloor \frac{d-1}{2} \rfloor)$

$$\sigma(x) \;=\; \prod_{i\in\mathcal{E}} (1 - \alpha^i x) \;=\; \sigma_0 + \sigma_1 x + \cdots + \sigma_t x^t$$

Jeśli poznamy współczynniki σ_j, odczytamy pozycje błędów z warunku, że jeśli $i \in \mathcal{E}$ to $\sigma(\alpha^{-i}) = 0$.

Rozważmy formalny szereg

$$\eta(X) \;=\; \sum_{j=1}^{\infty} e(\alpha^j) X^j\,.$$

Współczynniki powtarzają się cyklicznie, bo $\alpha^j = \alpha^{n+j}$ oraz $e(\alpha^j) = \tilde{w}(\alpha^j)$ dla $j = 1, 2, ..., d-1$. Okazuje się, że tych początkowych $d-1$ współczynników

wystarcza do wyznaczenia funkcji $\sigma(X)$.

$$\begin{aligned}\eta(X) &= \sum_{j=1}^{\infty} e(\alpha^j)X^j = \sum_{j=1}^{\infty}\sum_{i\in\mathcal{E}} \alpha^{ji}X^j \\ &= \sum_{i\in\mathcal{E}}\sum_{j=1}^{\infty}(\alpha^i X)^j = \sum_{i\in\mathcal{E}} \frac{\alpha^i X}{1-\alpha^i X} \\ &= \text{(wspólny mianownik)} = \frac{\omega(X)}{\sigma(X)}\,.\end{aligned}$$

Stąd

$$\omega(X) = \sum_{i\in\mathcal{E}} \alpha^i X \prod_{i\neq j\in\mathcal{E}} (1-\alpha^j X)\,.$$

Zauważmy, że zarówno $\sigma(X)$ jak i $\omega(X)$ mają stopień $|\mathcal{E}| \leq t$. Ponieważ $\sigma(X)\eta(X) = \omega(X)$,

$$\begin{aligned}&(\sigma_0+\sigma_1 X+\cdots+\sigma_t X^t)(\tilde{w}(\alpha)X+\tilde{w}(\alpha^2)X^2+\cdots+\tilde{w}(\alpha^{2t})X^{2t}+e(\alpha^d)X^d+\cdots) \\ &\qquad = \omega_0+\omega_1 X+\cdots\omega_t x^t\end{aligned}$$

Współczynniki przy X^k po lewej stronie dla $t < k \leq 2t = d-1$ muszą znikać

$$\sum_{j=0}^{t} \sigma_j \tilde{w}(\alpha^{k-j}) = 0$$

i nie zawierają w sobie członów $e(\alpha^k)X^k$ dla $k \geq d = 2t+1$. Mamy stąd układ t równań na $t+1$ poszukiwanych współczynników funkcji σ:

$$\begin{bmatrix} \tilde{w}(\alpha^{t+1}) & \tilde{w}(\alpha^t) & \dots & \tilde{w}(\alpha) \\ \tilde{w}(\alpha^{t+2}) & \tilde{w}(\alpha^{t+1}) & \dots & \tilde{w}(\alpha^2) \\ \vdots & & \dots & \vdots \\ \tilde{w}(\alpha^{2t}) & \tilde{w}(\alpha^{2t-1}) & \dots & \tilde{w}(\alpha^t) \end{bmatrix} \begin{bmatrix} \sigma_0 \\ \sigma_1 \\ \vdots \\ \sigma_t \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (*)$$

Algorytm jest więc następujący:

- odbieramy słowo $\tilde{w}$ i obliczamy $\tilde{w}(\alpha), ..., \tilde{w}(\alpha^{2t})$
- jeśli wszystkie są równe 0, akceptujemy $\tilde{w}$ jako słowo kodowe
- jeśli nie, rozwiązujemy układ $(*)$, tworząc funkcję $\sigma(X)$
- dla $i = 1, ..., t$ sprawdzamy, które z wartości $\sigma(\alpha^{-i}) \neq 0$ — to są pozycje błędnych bitów w $\tilde{w}$.