Wykład 11.

11.1 Kody BCH

Przypuśćmy, że mamy dane skończone ciało F, liczbę n względnie pierwszą z charakterystyką F, dodatnią liczbę t oraz element β rzędu n w pewnym rozszerzeniu F (czyli $\beta^n = 1$ i $\beta^i \neq 1$ dla i takich, że $0 < i < n$). Wtedy kodem BCH (od nazwisk Bose-Chaudhuri-Hocquenghem) nazywamy zbiór

$$\{w \in F^{n-1}[x] : \forall i = 1, \dots, 2 \cdot t \;\; w(\beta^i) = 0\}.$$

Jest oczywiste, że kody BCH są cykliczne. Zauważmy, że warunek „n jest względnie pierwsze z charakterystyką F" jest równoważny istnieniu elementu rzędu n w pewnym rozszerzeniu F. Fakt ten wynika z twierdzenia Eulera, którego dowód jest przedstawiony w początkowym fragmencie rozdziału 10.3.

Twierdzenie 11.1 *Jeżeli parametry F, n, t i β spełniają wyżej podane warunki, to minimalna odległość słów kodu BCH zdefiniowanego dla tych parametrów jest przynajmniej równa $2 \cdot t + 1$, a więc taki kod koryguje przynajmniej t błędów.*

Dowód. Ponieważ kody BCH są liniowe, wystarczy pokazać, że jeżeli słowo kodowe $w \in S_{2\cdot t}(0)$, to $w = 0$.

Przypuśćmy więc, że $w \in S_{2\cdot t}(0) \subseteq F^{n-1}[x]$. Taki wielomian w można przedstawić w postaci $w(x) = \sum_{i=1}^{m} w_i \cdot x^{k_i}$ dla pewnego $m \leqslant \min(2 \cdot t, n-1)$ oraz dla pewnych $k_1 < \ldots < k_m < n$. Jeżeli w jest słowem kodowym, to współczynniki $w_0, \ldots, w_m$ spełniają dla j takich, że $0 < j \leqslant m \leqslant 2 \cdot t$ równania liniowe

$$\sum_{i=0}^{m} w_i \cdot (\beta^j)^{k_i} = \sum_{i=0}^{m} w_i \cdot (\beta^{k_i})^j = 0.$$

Macierz główna tego układu ma postać

$$\begin{bmatrix} \beta^{k_1} & \beta^{k_2} & \ldots & \beta^{k_m} \\ (\beta^{k_1})^2 & (\beta^{k_2})^2 & \ldots & (\beta^{k_m})^2 \\ & & \ddots & \\ (\beta^{k_1})^m & (\beta^{k_2})^m & \ldots & (\beta^{k_m})^m \end{bmatrix}.$$

Wyznacznik tej macierzy jest tzw. wyznacznikiem Vandermonde'a i z głównej własności tych wyznaczników wynika, że jest różny od 0, gdyż elementy $\beta^{k_1}, \beta^{k_2}, \ldots, \beta^{k_m}$ są parami różne. Wobec tego, rozważany układ równań ma dokładnie jedno rozwiązanie i jest to rozwiązanie $w_1 = \ldots = w_m = 0$. Tak więc wielomian w ma wszystkie współczynniki równe 0 i jest wielomianem zerowym. □

Z podanego twierdzenia wynika, że jeżeli $n \leqslant 2 \cdot t$, to kod BCH składa się tylko z 0.

11.2 Algorytm Euklidesa

Rozważmy następującą procedurę $\mathcal{E}(a,b)$ z parametrami i zmiennymi, których wartościami są wielomiany:

1. $U_1 := 1$; $W_1 := 0$; $U_2 := 0$; $W_2 := 1$; $D_1 := a$; $D_2 := b$;
2. **dopóki** $D_2 \neq 0$
   (a) $Q := (-1) \cdot (D_1 \div D_2)$;
   (b) $X := U_2$; $U_2 := Q \cdot U_2 + U_1$; $U_1 := X$;
   (c) $Y := W_2$; $W_2 := Q \cdot W_2 + W_1$; $W_1 := Y$;
   (d) $D_1 := D_2$; $D_2 := U_2 \cdot a + W_2 \cdot b$

Przypuśćmy, że zmienna z indeksem ^{p} oznacza wartość tej zmiennej tuż przed pewnym wykonaniem instrukcji (a)-(d), a z indeksem ^{k} – tuż po wykonaniu tych instrukcji. Oczywiście, $D_1^k = D_2^p$, $U_1^k = U_2^p$ oraz $W_1^k = W_2^p$.

Równości $D_1 = U_1 \cdot a + W_1 \cdot b$ oraz $D_2 = U_2 \cdot a + W_2 \cdot b$ są spełnione przed wykonaniem instrukcji „dopóki". Równość $D_2 = U_2 \cdot a + W_2 \cdot b$ jest spełniona zawsze po wykonaniu instrukcji (d). Wobec tego, $D_2^p = U_2^p \cdot a + W_2^p \cdot b$. Z równości podanych w poprzednim akapicie wynika, że $D_1^k = U_1^k \cdot a + W_1^k \cdot b$. Ostatecznie otrzymujemy, że równości $D_1 = U_1 \cdot a + W_1 \cdot b$ oraz $D_2 = U_2 \cdot a + W_2 \cdot b$ są spełnione zawsze przed wykonaniem instrukcji (a) i po wykonaniu instrukcji (d).

Zauważmy też, że

$$\begin{aligned} D_2^k &= U_2^k \cdot a + W_2^k \cdot b = (Q^k \cdot U_2^p + U_1^p) \cdot a + (Q^k \cdot W_2^p + W_1^p) \cdot b \\ &= Q^k \cdot (U_2^p \cdot a + W_2^p \cdot b) + (U_1^p \cdot a + W_1^p \cdot b) = Q^k \cdot D_2^p + D_1^p \\ &= D_1^p - (D_1^p \div D_2^p) \cdot D_2^p = D_1^p \bmod D_2^p. \end{aligned} \tag{14}$$

Z wzoru (14) wynika, że procedura $\mathcal{E}$ zawsze kończy pracę. Wzór ten implikuje, że podczas wykonywania procedury $\mathcal{E}$ stale maleje stopień wartości zmiennej D_2. Wobec tego, w pewnym momencie stopień D_2 stanie się mniejszy od 1 (stopień wielomianu wygodnie zdefiniować tak, aby wielomian zerowy miał stopień -1, a pozostałe wielomiany będące jednocześnie elementami ciała współczynników – stopień 0). Wtedy procedura $\mathcal{E}$ zakończy pracę, gdyż albo $D_2 = 0$, albo D_2 jest różnym od zera elementem ciała, a więc wielomianem, który dzieli każdy inny wielomian. W tym ostatnim przypadku, po wykonaniu jeszcze raz instrukcji (a)-(d) zmienna D_2 przyjmie wartość 0.

Oczywiście, pary wielomianów (w,v) oraz $(v, w \bmod v)$ mają ten sam zbiór dzielników. Na podstawie wzoru (14) to samo można twierdzić o parach (D_1^p, D_2^p) oraz $(D_1^k, D_2^k) = (D_2^p, D_1^p \bmod D_2^p)$. Niech $X(a,b)$ będzie zbiorem wszystkich dzielników wielomianów a i b. Wobec tego, własność „$X(a,b)$ jest zbiorem dzielników D_1 i D_2" jest niezmiennikiem pętli „dopóki". Jest też prawdziwa przed wykonaniem pętli i – w konsekwencji – po jej wykonaniu. Wtedy jednak wartością D_2 jest zero. Oznacza to, że $X(a,b)$ jest zbiorem dzielników D_1. W zbiorze $X(a,b)$ jest więc największy dzielnik i jest nim wartość zmiennej D_1 po zakończeniu procedury $\mathcal{E}(a,b)$. Dodatkowo, wtedy $D_1 = U_1 \cdot a + W_1 \cdot b$. Procedura $\mathcal{E}(a,b)$ znajduje więc także przedstawienie największego wspólnego dzielnika a i b w postaci kombinacji liniowej a i b.

Co więcej, współczynniki tej kombinacji są względnie pierwsze. Wynika to z równości

$$\begin{aligned}&U_1^k \cdot W_2^k - U_2^k \cdot W_1^k = U_2^p \cdot (Q \cdot W_2^p + W_1^p) - (Q \cdot U_2^p + U_1^p) \cdot W_2^p\\&= -(U_1^p \cdot W_2^p - U_2^p \cdot W_1^p).\end{aligned}$$

Po wykonaniu początkowych podstawień wartość $U_1 \cdot W_2 - U_2 \cdot W_1 = 1$. Podczas wykonywania procedury $\mathcal{E}$ wartość ta jest równa ± 1. Konsekwencją tej równości jest to, że wielomiany U_1 i W_1 oraz wielomiany U_2 i W_2 są względnie pierwsze.

Dalszą analizę algorytmu poprowadzimy przy dodatkowym założeniu, że stopień wielomianu a jest większy od stopnia wielomianu b, a więc $st(a) > st(b) \geqslant 0$ (założenie to nie jest potrzebne, ale znacznie ułatwia analizę algorytmu). Wtedy na początku i na końcu każdej pętli zachodzi wzór $st(D_1) > st(D_2)$. Po wykonaniu pętli, na podstawie wzoru (14) mamy bowiem $st(D_2^k) < st(D_2^p) = st(D_1^k)$. Stąd i z założenia $st(a) > st(b)$ dowodzona nierówność jest prawdziwa także przed wykonaniem pętli. Konsekwencją tego wzoru jest m.in. fakt, że $st(Q) > 0$, jeżeli tylko wartość Q jest określona. Jest oczywiste, że $st(Q) = st(D_1^p) - st(D_2^p)$. Zawsze zachodzi także wzór $st(W_1) < st(W_2)$. Jest on bowiem prawdziwy na początku programu i po każdym wykonaniu pętli:

$$\begin{aligned}&st(W_2^k) = st(Q \cdot W_2^p + W_1^p) = st(Q \cdot W_2^p) = st(Q) + st(W_2^p)\\&= st(Q) + st(W_1^k) > st(W_1^k).\end{aligned}$$

Niezmiennikiem pętli wykonywanej w procedurze $\mathcal{E}$ jest także własność $st(W_2) + st(D_1) = st(a)$, ponieważ

$$\begin{aligned}&st(W_2^k) + st(D_1^k) = st(W_2^p) + st(Q) + st(D_1^k)\\&= st(W_2^p) + st(Q) + st(D_2^p) = st(W_2^p) + st(D_1^p) = st(a).\end{aligned}$$

Z analizy algorytmu $\mathcal{E}$ wynika także, że po zakończeniu tego algorytmu

$$\begin{aligned}&st(W_1) = st(W_1^k) = st(W_2^p) = st(a) - st(D_1^p) \leqslant st(a) - st(D_2^p)\\&= st(a) - st(D_1^k) = st(a) - st(NWD(a,b)),\end{aligned}$$

a także

$$\begin{aligned}&st(U_1) = st(U_1^k) = st(U_1^k \cdot b) - st(b) \leqslant st(D_1^k - W_1^k \cdot a) - st(b)\\&\leqslant \max(st(NWD(a,b)), 2 \cdot st(a) - st(NWD(a,b))) - st(b)\\&< st(a) - st(NWD(a,b)).\end{aligned}$$

11.3 Algorytm korygujący błędy dla kodów BCH

Formalną pochodną wielomianu

$$w(x) = \sum_{i=0}^{n} a_i \cdot x^i$$

nazywamy wielomian

$$w'(x) = \sum_{i=0}^{n-1} a_{i+1} \cdot (i+1) \cdot x^i.$$

Można dowieść, że formalna pochodna ma różne własności znane z rachunku różniczkowego, w szczególności prawdziwy jest wzór

$$(u(x) \cdot w(x))' = u'(x) \cdot w(x) + u(x) \cdot w'(x).$$

Przyjmijmy, że procedura $\mathcal{E}'$ powstaje z procedury $\mathcal{E}$ z poprzedniego rozdziału (z algorytmu Euklidesa) przez usunięcie warunku „$D_2 \neq 0$” i wpisanie w jego miejsce warunku „stopień D_2 jest $\geqslant t$”.

Będziemy rozważać następujący algorytm korygujący błędy dla kodu BCH zdefiniowanego wzorem

$$K = \{w \in F^{n-1}[x] : \forall i = 1, \ldots, 2 \cdot t \;\; w(\beta^i) = 0\} :$$

Dane: słowo w,

1. **oblicz** wielomian
$$s(x) = \sum_{i=0}^{2\cdot t-1} w(\beta^{i+1}) \cdot x^i;$$

2. **jeżeli** $s(x) = 0$**, to przerwij obliczenia i zwróć** w**;**

3. **wykonaj** zmodyfikowany algorytm Euklidesa $\mathcal{E}'$ z parametrami $x^{2\cdot t}$ i $s(x)$; niech $\bar{\omega}(x)$ i $\bar{\sigma}(x)$ będą końcowymi wartościami zmiennych D_2 i W_2;

4. $e(x) := 0$

5. **dla** $i := 0$ **do** $n-1$ **wykonaj**

   jeżeli $\bar{\sigma}(\beta^{-i}) = 0$**, to**
$$e(x) := e(x) - \frac{\bar{\omega}(\beta^{-i})}{\bar{\sigma}'(\beta^{-i})} \cdot x^i;$$

6. **zwróć** $w(x) - e(x)$.

Nietrudno zauważyć, że jeżeli podany algorytm uruchomimy ze słowem $w \in K$, to zostanie stwierdzone, że $s(x) = 0$ i algorytm zwróci słowo w. Przypuśćmy więc, że mamy wielomiany $v \in K$ oraz $w \in F^{n-1}[x]$ takie, że $e(x) = w(x) - v(x)$ jest niezerowym wielomianem o $p \leqslant t$ niezerowych współczynnikach. Przyjmijmy, że

$$e(x) = \sum_{i=1}^{p} e_i \cdot x^{k_i}.$$

Zdefiniujmy dwa wielomiany

$$\sigma(x) = \prod_{i=1}^{p}(1 - \beta^{k_i} \cdot x)$$

(wielomian „błędnych pozycji") oraz

$$\omega(x) = \sum_{i=1}^{p} e_i \cdot \beta^{k_i} \cdot \frac{\sigma(x)}{1 - \beta^{k_i} \cdot x}$$

(wielomian „błędnych wartości"). Znając te dwa wielomiany potrafimy obliczyć wielomian e.

Aby to zrobić, warto znaleźć formalna pochodną wielomianu σ. Jest to wielomian wyrażający się wzorem

$$\sigma'(x) = -\sum_{i=1}^{p} \beta^{k_i} \cdot \frac{\sigma(x)}{1 - \beta^{k_i} \cdot x}.$$

Wobec tego,

$$\sigma'(\beta^{-k_i}) = -\beta^{k_i} \cdot (\frac{\sigma(x)}{1 - \beta^{k_i} \cdot x})_{x=\beta^{-k_i}}.$$

Zachodzi także wzór

$$\omega(\beta^{-k_i}) = e_i \cdot \beta^{k_i} \cdot (\frac{\sigma(x)}{1 - \beta^{k_i} \cdot x})_{x=\beta^{-k_i}}.$$

Stąd otrzymujemy, że

$$e_i = -\frac{\omega(\beta^{-k_i})}{\sigma'(\beta^{-k_i})}.$$

Aby obliczyć wielomian $e(x)$ mając dane wielomiany $\sigma(x)$ i $\omega(x)$, postępujemy zgodnie z punktem 5 podanego algorytmu i analizujemy rozwiązania postaci β^j równania $\sigma(x) = 0$. Wielomian e jest sumą jednomianów postaci

$$-\frac{\omega(\beta^j)}{\sigma'(\beta^j)} \cdot x^j$$

wyznaczonych przez wszystkie rozwiązania tej postaci równania $\sigma(x) = 0$.

Mając dane słowo w (przysłane z nieznanym błędem $e(x)$), algorytm korygujący błędy najpierw oblicza wielomian $s(x)$, a następnie wielomiany $\bar{\sigma}(x)$ oraz $\bar{\omega}(x)$. Zauważmy, że gdyby te wielomiany okazały się równe odpowiednio $c\cdot\sigma(x)$ oraz $c\cdot\omega(x)$ dla pewnej niezerowej stałej c, to rozważany algorytm obliczyłby błąd $e(x)$. Będziemy więc dowodzić podane przedstawienia wielomianów $\bar{\sigma}(x)$ oraz $\bar{\omega}(x)$.

Twierdzenie 11.2 *Jeżeli $w(x) - e(x) \in K$, to wielomian $\omega(x) - \sigma(x) \cdot s(x)$ dzieli się przez $x^{2\cdot t}$.*

Dowód. Zauważmy, że

$$1 - y^{2\cdot t} = (1 - y) \cdot \sum_{i=0}^{2\cdot t-1} y^i.$$

Podstawiając $\beta^{k_j} \cdot x$ zamiast y otrzymujemy, że

$$1 - (\beta^{k_j} \cdot x)^{2\cdot t} = (1 - \beta^{k_j} \cdot x) \cdot \sum_{i=0}^{2\cdot t-1} (\beta^{k_j} \cdot x)^i.$$

Stąd wynika, że

$$\frac{\sigma(x)}{1 - \beta^{k_j} \cdot x} \cdot (1 - (\beta^{k_j} \cdot x)^{2\cdot t}) = \sigma(x) \cdot \sum_{i=0}^{2\cdot t-1} (\beta^{k_j} \cdot x)^i$$

oraz (dla odpowiedniego wielomianu $p(x)$)

$$\begin{aligned}
\omega(x) &= \sum_{j=1}^{p} e_j \cdot \beta^{k_j} \cdot \frac{\sigma(x)}{1 - \beta^{k_j} \cdot x} \\
&= \sum_{j=1}^{p} e_j \cdot \beta^{k_j} \cdot \frac{\sigma(x)}{1 - \beta^{k_j} \cdot x} \cdot (1 - (\beta^{k_j} \cdot x)^{2\cdot t}) + \sum_{j=1}^{p} e_j \cdot \beta^{k_j} \cdot \frac{\sigma(x)}{1 - \beta^{k_j} \cdot x} \cdot (\beta^{k_j} \cdot x)^{2\cdot t} \\
&= \sum_{j=1}^{p} e_j \cdot \beta^{k_j} \cdot \sigma(x) \cdot \sum_{i=0}^{2\cdot t-1} (\beta^{k_j} \cdot x)^i + p(x) \cdot x^{2\cdot t} \\
&= \sigma(x) \cdot \sum_{i=0}^{2\cdot t-1} (\sum_{j=1}^{p} e_j \cdot (\beta^{i+1})^{k_j}) \cdot x^i + p(x) \cdot x^{2\cdot t} \\
&= \sigma(x) \cdot \sum_{i=0}^{2\cdot t-1} e(\beta^{i+1}) \cdot x^i + p(x) \cdot x^{2\cdot t}.
\end{aligned}$$

Z tego, że $v \in K$ i z definicji kodu K wynika, że dla $i = 0, \ldots, 2 \cdot t - 1$ zachodzi równość

$$e(\beta^{i+1}) = w(\beta^{i+1}).$$

Stąd otrzymujemy, że

$$\omega(x) = \sigma(x) \cdot \sum_{i=0}^{2\cdot t-1} w(\beta^{i+1}) \cdot x^i + p(x) \cdot x^{2\cdot t} = \sigma(x) \cdot s(x) + p(x) \cdot x^{2\cdot t}$$

dla pewnego wielomianu $p(x)$. $\square$

Z analizy algorytmu Euklidesa otrzymujemy, że

$$\bar{\omega}(x) = u(x) \cdot x^{2\cdot t} + \bar{\sigma}(x) \cdot s(x) \tag{15}$$

dla pewnego wielomianu u.

Przyjmijmy, że podczas wykonywania procedury $\mathcal{E}'$ zostały wykonane instrukcje (a)-(d). Będziemy analizować ostatnie wykonanie tych instrukcji. Wtedy mamy $st(D_2^p) \geqslant t$ oraz $st(\bar{\omega}) = st(D_2^k) < t$. Ponadto,

$$st(\bar{\sigma}) = st(W_2^k) = st(W_2^p) + st(Q) = st(W_2^p) + st(D_1^p) - st(D_2^p) = 2 \cdot t - st(D_2^p) \leqslant t.$$

Zauważmy, że nierówności $st(\bar{\sigma}) \leqslant t$ oraz $st(\bar{\omega}) < t$ są prawdziwe także wtedy, gdy instrukcje (a)-(d) nie zostały wykonane.

Przekształcimy teraz wyrażenie $\omega(x) \cdot s(x) \cdot \bar{\sigma}(x)$. Zauważmy, że

$$\begin{aligned}
&\omega(x) \cdot \bar{\sigma}(x) \cdot s(x) \\
&= (\sigma(x) \cdot s(x) + p(x) \cdot x^{2 \cdot t}) \cdot \bar{\sigma}(x) \cdot s(x) \\
&= \sigma(x) \cdot s(x) \cdot \bar{\sigma}(x) \cdot s(x) + p(x) \cdot x^{2 \cdot t} \cdot \bar{\sigma}(x) \cdot s(x) \\
&= \sigma(x) \cdot s(x) \cdot (\bar{\omega}(x) - u(x) \cdot x^{2 \cdot t}) + p(x) \cdot x^{2 \cdot t} \cdot \bar{\sigma}(x) \cdot s(x) \\
&= \sigma(x) \cdot s(x) \cdot \bar{\omega}(x) - \sigma(x) \cdot s(x) \cdot u(x) \cdot x^{2 \cdot t} + p(x) \cdot x^{2 \cdot t} \cdot \bar{\sigma}(x) \cdot s(x) \\
&= \sigma(x) \cdot s(x) \cdot \bar{\omega}(x) + s(x) \cdot x^{2 \cdot t} \cdot (p(x) \cdot \bar{\sigma}(x) - \sigma(x) \cdot u(x)).
\end{aligned}$$

Jeżeli udowodnioną równość podzielimy przez $s(x)$, to otrzymamy

$$\omega(x) \cdot \bar{\sigma}(x) = \sigma(x) \cdot \bar{\omega}(x) + x^{2 \cdot t} \cdot (p(x) \cdot \bar{\sigma}(x) - \sigma(x) \cdot u(x)).$$

Wobec tego, wielomian $\omega(x) \cdot \bar{\sigma}(x) - \sigma(x) \cdot \bar{\omega}(x)$ jest podzielny przez $x^{2 \cdot t}$, a ponieważ jest to wielomian stopnia $\leqslant 2 \cdot t - 1$, więc jest równy 0. Oznacza to, że

$$\omega(x) \cdot \bar{\sigma}(x) = \sigma(x) \cdot \bar{\omega}(x). \tag{16}$$

Wobec tego, mamy także

$$p(x) \cdot \bar{\sigma}(x) = \sigma(x) \cdot u(x). \tag{17}$$

Znamy pierwiastki równania $\sigma(x) = 0$ i wiemy, że nie są pierwiastkami równania $\omega(x) = 0$. Z równości (16) wynika więc, że są pierwiastkami równania $\bar{\sigma}(x) = 0$. To z kolei implikuje, że $\bar{\sigma}(x) = h(x) \cdot \sigma(x)$ dla pewnego wielomianu $h(x)$. Podstawiając w równości (16) za $\bar{\sigma}(x)$ wielomian $h(x) \cdot \sigma(x)$ i dzieląc obustronnie przez $\sigma(x)$ otrzymujemy, że

$$\omega(x) \cdot h(x) = \bar{\omega}(x).$$

Wykonując analogiczne podstawienia w równości (17) otrzymujemy, że

$$p(x) \cdot h(x) = u(x).$$

Z analizy algorytmu Euklidesa $\mathcal{E}$ (patrz wzór (15)) wynika, że wielomiany $u(x)$ i $\bar{\sigma}(x)$ są względnie pierwsze. Pokazaliśmy też, że wielomiany te są podzielne przez $h(x)$. Wielomian $h(x)$ ma więc stopień 0. Wielomian $h(x)$ można znaleźć korzystając z tego, że współczynnik przy najwyższej potędze wielomianu $\sigma(x)$ jest równy 1. Jest on równy współczynnikowi d przy najwyższej potędze wielomianu $\bar{\sigma}(x)$. Tak więc zachodzą wzory

$$\sigma(x) = \frac{\bar{\sigma}(x)}{d} \text{ oraz } \omega(x) = \frac{\bar{\omega}(x)}{d}.$$