Wykład 8.

8.1 Kodowanie i dekodowanie dla kodów cyklicznych

Niech $K \subseteq F^{n+1}$ będzie kodem cyklicznym wymiaru m generowanym przez wielomian g stopnia $k = n + 1 - m$.

Przypuśćmy, że literę a z alfabetu A przesyłamy używając kodu K w następujący sposób: najpierw zamieniamy a na słowo $f(a) \in F^m$, słowo $f(a)$ kodujemy jako słowo $\kappa(f(a)) \in K$ i przesłamy je odbiorcy. Odbiorca po otrzymaniu informacji powinien skorygować błędy, które wystąpiły podczas przesyłania informacji, a więc ustalić na podstawie otrzymanej informacji słowo $\kappa(f(a))$. Następnie powinien zdekodować otrzymane słowo obliczając $f(a)$ i na tej podstawie odtworzyć literę a.

Dla kodu cyklicznego K mamy dwa naturalne sposoby kodowanie słów (czyli definiowania funkcji κ).

Sposób 1. Dane słowo $a_0 \ldots a_{m-1}$ utożsamiamy z wielomianem $w(x) = a_0 + a_1 \cdot x + \ldots + a_{m-1} \cdot x^{m-1}$ i kodujemy je jako iloczyn wielomianów $w(x) \cdot g(x)$ (lub jako ciąg współczynników tego wielomianu. W tym przypadku dekodowanie polega na dzieleniu przez $g(x)$.

Sposób 2. Dane słowo $a_0 \ldots a_{m-1}$ utożsamiamy z wielomianem $w(x) = (a_0 + a_1 \cdot x + \ldots + a_{m-1} \cdot x^{m-1}) \cdot x^{n+1-m}$ i kodujemy je jako $w(x) - w(x) \bmod g(x)$. Zauważmy, że w tym przypadku $n + 1 - m$ pierwszych współczynników wielomianu $w(x) - w(x) \bmod g(x)$ to współczynniki reszty $w(x) \bmod g(x)$ z dzielenia wielomianu $w(x)$ przez $g(x)$, a pozostałymi współczynnikami są $a_0, \ldots, a_{m-1}$. Wobec tego, dekodowanie polega na usunięciu $n+1-m$ pierwszych liter danego słowa, albo – w języku wielomianów – na obliczeniu ilorazu danego wielomianu i wielomianu x^{n+1-m}.

8.2 Korygowanie błędów w przypadku kodów cyklicznych

.

Niech $g(x)$ będzie wielomianem stopnia k oraz niech

$$K = \{u(x) \cdot g(x) : \text{ stopnień } u \leqslant n - k\}$$

będzie kodem (niekoniecznie cyklicznym) korygującym t błędów. W takim przypadku, resztę z dzielenia danego wielomianu przez $g(x)$ będziemy nazywać syndromem.

Algorytm 1. Dane: wielomian $w \in F^n[x]$.

1. Dla każdego wielomianu $u \in F^n[x]$ o $\leqslant t$ niezerowych współczynnikach zapamiętujemy parę złożoną z wielomianu $u(x)$ oraz z jego syndromu $u(x) \bmod g(x)$;

2. obliczamy syndrom $w(x) \bmod g(x)$ wielomianu w;

3. przeglądamy obliczone pary w poszukiwaniu wielomianu u takiego, że $w(x) \bmod g(x) = u(x) \bmod g(x)$;

4. Jeżeli znajdziemy taki wielomian $u(x)$, to zwracamy wielomian $w(x) - u(x)$, w przeciwnym razie zwracamy 0.

Lemat 8.1 *Jeżeli K jest kodem generowanym przez wielomian $g(x)$, korygującym t błędów, to dla dowolnych w, v takich, że $w \in S_t(v)$ oraz $v \in K$, algorytm 1 uruchomiony z wielomianem w zwraca wielomian v (a więc algorytm 1 jest algorytmem korygującym błędy dla kodu K).*

Dowód. Niech $e(x) = w(x) - v(x)$. Oczywiście, wielomian $e(x)$ ma $\leqslant t$ niezerowych współczynników oraz $e(x) \bmod g(x) = w(x) \bmod g(x)$. Po uruchomieniu algorytmu 1 z daną w zostanie więc znaleziony pewien wielomian $u(x)$ o $\leqslant t$ niezerowych współczynnikach spełniający równość $u(x) \bmod g(x) = w(x) \bmod g(x)$. Wobec tego, wielomian $w(x) - u(x)$ dzieli się $g(x)$, a więc $w(x) - u(x) \in K$. Zauważmy, że $w \in S_t(v) \cap S_t(w - u)$. Ponieważ K koryguje t błędów, jest to możliwe tylko wtedy, gdy $v = w - u$. □

Niech C oznacza operację cyklicznego przesunięcia. Tak więc $C(a_0 \ldots a_n) = a_n a_0 \ldots a_{n-1}$ lub $C(w(x)) = x \cdot w(x) - a_n \cdot (x^{n+1} - 1)$. Nietrudno zauważyć, że C jest operacją liniową, zachowuje liczbę niezerowych współczynników i dla $w \in F^n[x]$ zachodzi wzór $C^{n+1}(w) = w$. Jeżeli $w \in F^n[x]$ jest niezerowym wielomianem stopnia $< n$, to $C(w)$ jest wielomianem stopnia o 1 większego niż stopień w.

Algorytm 2. Dane: wielomian $w \in F^n[x]$.

1. Dla każdego wielomianu $u \in F^n[x]$ stopnia n o $\leqslant t$ niezerowych współczynnikach zapamiętujemy parę złożoną z wielomianu $u(x)$ oraz z jego syndromu $u(x) \bmod g(x)$;
2. $W := w$; $I := 0$;
3. **dopóki** $I < n + 1$:
   (a) oblicz syndrom $W \bmod g(x)$ wielomianu W;
   (b) przejrzyj obliczone pary w poszukiwaniu wielomianu u takiego, że $W \bmod g(x) = u(x) \bmod g(x)$;
   (c) jeżeli jest taki wielomian $u(x)$, to $W := W - u(x)$;
   (d) $W := C(W)$;
   (e) $I := I + 1$
4. zwróć W.

Lemat 8.2 *Jeżeli K jest kodem cyklicznym generowanym przez wielomian $g(x)$, korygującym t błędów, to dla dowolnych w, v takich, że $w \in S_t(v)$ oraz $v \in K$, algorytm 2 uruchomiony z wielomianem w zwraca wielomian v (a więc algorytm 2 jest algorytmem korygującym błędy dla kodu K).*

Dowód. Najpierw zauważmy, że własność $W = C^I(w)$, gdzie w jest dowolną stałą taką, że $w \in K$, jest niezmiennikiem tej pętli (niezmiennikiem instrukcji (a)-(e)). Wynika to stąd, że podczas wykonywania pętli w takiej sytuacji nie jest spełniony warunek w instrukcji warunkowej (c), a więc jednokrotne wykonanie pętli polega nadaniu zmiennej W wartości $C(W')$ oraz zmiennej I wartości $I' + 1$ (' oznacza początkowe wartości zmiennych). Natomiast wspomniany warunek jest fałszywy, ponieważ gdyby był prawdziwy, to oznaczałby, że $W' - u \in K$ dla pewnego niezerowego wielomianu o najwyżej t niezerowych współczynnikach. Oczywiście, $W' \in K$ na podstawie cykliczności K. Tak więc dwa wielomiany $W', W' - u \in K$ różniłyby się najwyżej t współczynnikami, co przeczy założeniu, że K koryguje t błędów.

Z powyższego wynika, że jeżeli zmienna W przyjmie w pewnym momencie wartość $w \in K$, a zmienna I przyjmuje w tym momencie wartość i, to algorytm 2 zwróci wartość $C^{n+1-i}(w) \in K$. W szczególności, algorytm 2 uruchomiony z daną $w \in K$ zwraca w.

Przypuśćmy, że mamy wielomiany w, v takie, że $w \in S_t(v)$ oraz $v \in K$. Możemy dodatkowo założyć, że $w \neq v$. Wtedy $e(x) = w(x) - v(x)$ niezerowym wielomianem stopnia l dla pewnego $l \leqslant n$. $C^{n-l}(e)$ jest wielomianem stopnia n. Wielomiany e oraz $C^{n-l}(e)$ mają $\leqslant t$ niezerowych współczynników.

Algorytm 2 uruchomiony z daną $w \in S_t(v)$ dla pewnego $v \in K$ doprowadzi do sytuacji, w której warunek z instrukcji (c) będzie prawdziwy. Gdyby tak nie było, to podczas wykonywania algorytmu 2 z taką daną zmienna W będzie przyjmowała kolejno wartości $w, C(w), \ldots, C^{n+1}(w)$. Przyjmie też wartość $C^{n-l}(w)$. Dla wielomianu $C^{n-l}(w)$ mamy $C^{n-l}(w) \bmod g = C^{n-l}(e) \bmod g$ i okazuje się, że jeżeli W przyjmuje wartość $C^{n-l}(w)$, to wbrew założeniu warunek z instrukcji (c) jest prawdziwy.

Przypuśćmy więc, że algorytm 2 uruchomiony z daną w doprowadza do sytuacji, w której warunek z instrukcji warunkowej jest prawdziwy, i wtedy zmienna I przyjmuje wartość i. Zmienna W przyjmuje wtedy wartość $C^i(w)$ i podczas wykonywania instrukcji (c) wartość ta zostanie zmieniona na $C^i(w) - u$ dla pewnego u o najwyżej t współczynnikach $\neq 0$ takiego, że $C^i(w) - u \in K$. Stąd, na podstawie wcześniejszych ustaleń i cykliczności K stwierdzamy, że algorytm 2 zwraca $v' = C^{n+1-i}(C^i(w) - u) = w - C^{n+1-i}(u) \in K$. Tak więc $w \in S_t(v) \cap S_t(v')$. Ponieważ K koryguje t błędów, więc zwrócony wielomian v' jest równy v. $\square$

Algorytm 3. Dane: wielomian $w \in Z_2^n[x]$.

1. Znajdź zbiór

$$\mathcal{S} = \{u(x) \bmod g(x) : u \in Z_2^n[x] \text{ ma stopień } n \text{ i } \leqslant t \text{ współczynników } \neq 0\};$$

2. $W := w$; $I := 0$;
3. **dopóki** $I < n + 1$:
   (a) jeżeli $W \bmod g(x) \in \mathcal{S}$, to $W := W + x^n$;
   (b) $W := C(W)$;
   (c) $I := I + 1$
4. zwróć W.

Lemat 8.3 *Jeżeli K jest binarnym kodem cyklicznym generowanym przez wielomian $g(x)$, korygującym t błędów, to dla dowolnych w, v takich, że $w \in S_t(v)$ oraz $v \in K$, algorytm 3 uruchomiony z wielomianem w zwraca wielomian v (a więc algorytm 3 jest algorytmem korygującym błędy dla kodu K).*

Dowód. Niech $\mathcal{S} = \{u(x) \bmod g(x) : u \in Z_2^n[x]$ ma stopień n i $\leqslant t$ współczynników $\neq 0\}$. Najpierw zauważmy, że jeżeli mamy wielomiany $v \in K$ i w ze zbioru $Z_2^n[x]$ takie, że $w \in S_t(v)$, to współczynniki v_n i w_n tych wielomianów przy potędze x^n są różne wtedy i tylko wtedy, gdy $w(x) \bmod g(x) \in \mathcal{S}$. Jest bowiem oczywiste, że jeżeli $w_n \neq v_n$, to $e(x) = w(x) - v(x)$ jest wielomianem stopnia n o najwyżej t niezerowych współczynnikach, a więc $e(x) \bmod g(x) \in \mathcal{S}$. Ponieważ $v \in K$, więc także $w(x) \bmod g(x) = e(x) \bmod g(x) \in \mathcal{S}$. Jeżeli natomiast $w(x) \bmod g(x) \in \mathcal{S}$, to $w(x) \bmod g(x) = e(x) \bmod g(x)$ dla pewnego wielomianu $e(x)$ stopnia n o najwyżej t niezerowych współczynnikach. Dla tego wielomianu e wielomian $w - e \in K$ oraz $w \in S_t(w - e)$. Ponieważ także $w \in S_t(v)$ i kod K koryguje t błędów, więc $w - e = v$. Mamy $w_n \neq v_n$, gdyż wielomian e jest stopnia n.

Teraz udowodnimy, że własność

$$W \bmod x^I = C^I(v) \bmod x^I \wedge W \in S_t(C^I(v)) \wedge C^I(v) \in K \tag{12}$$

jest niezmiennikiem pętli (a)-(c). Przypuśćmy więc, że własność ta jest prawdziwa przed wykonaniem instrukcji (a). Będziemy analizować dwa przypadki. Jeżeli $W \bmod g \notin \mathcal{S}$, to na podstawie udowodnionej równoważności wiemy, że współczynniki W i $C^I(v)$ przy x^n są równe i wykonanie pętli polega na wykonaniu podstawień $W := C(W)$ oraz $I := I + 1$. Mamy też

$$C(W) \bmod x^{I+1} = (W \cdot x - w_n \cdot (x^{n+1} - 1)) \bmod x^{I+1} = (W \bmod x^I) \cdot x + w_n.$$

Jeżeli analogicznie obliczymy $C(C^I(v)) \bmod x^{I+1}$ i porównamy otrzymane wzory, to okaże się, że $C(W) \bmod x^{I+1} = C^{I+1}(v) \bmod x^{I+1}$. Oczywiste własności C i założenia o kodzie K pociągają, że $C(W) \in S_t(C^{I+1}(v)) \wedge C^{I+1}(v) \in K$. Dowiedliśmy więc, że końcowe wartości zmiennych W oraz I mają własność (12).

W przypadku, gdy $W \bmod g \in \mathcal{S}$, to współczynniki W i $C^I(v)$ przy x^n są różne, ale tym razem instrukcja (a) zmienia najstarszy współczynnik W. Ponieważ rozważamy wielomiany o współczynnikach równych 0 lub 1, więc najstarsze współczynniki wielomianów $W + x^n$ i $C^I(v)$ są równe. Dalsze rozumowanie prowadzimy podobnie, jak w poprzednim akapicie. Zauważmy, że

$$C(W + x^n) \bmod x^{I+1} = (W \cdot x + x^{n+1} - w_n \cdot (x^{n+1} - 1)) \bmod x^{I+1} = (W \bmod x^I) \cdot x + w_n + 1.$$

Załóżmy, że $v \in K$ i $w \in S_t(v)$. Aby dowieść lemat wystarczy zauważyć, że niezmiennik (12) jest prawdziwy przed wykonaniem instrukcji „dopóki”, ponieważ każdy wielomian jest podzielny przez niezerowy wielomian stopnia 0, czyli niezerowy element ciała, oraz v ma własności gwarantujące prawdziwość ostatnich członów niezmiennika. Wobec tego, ten niezmiennik jest prawdziwy po wykonaniu instrukcji „dopóki”. Wtedy zmienna I ma wartość $n + 1$, a więc $W = W \bmod x^{n+1} = C^{n+1}(v) \bmod x^{n+1} = v \bmod x^{n+1} = v$. Udowodniona równość oznacza, że algorytm 5 zwrócił wielomian v. $\square$

Algorytm 4 (pomocniczy). Zdefiniujemy funkcję $D(a)$ (procedurę z jednym parametrem, zwracającą wartość) korzystającą z dwóch (globalnych) tablic G i R indeksowanych liczbami od 0 do k.

1. $R[0] := a$; $B := R[k]$;
2. dla i od k do 1 wykonaj $R[i] := R[i-1] - B * G[i-1]$;
3. zwróć B.

Lemat 8.4 *Przypuśćmy, że po zapamiętaniu w tablicy G ciągu $g_0, \ldots, g_k$ takiego, że $g_k = 1$ i wyzerowaniu tablicy R została wykonana n razy procedura D z parametrami $a_1, \ldots, a_n$ Wtedy, jeżeli $b_1, \ldots, b_n$ są wartościami zwróconymi przez procedurę D, to*

$$\sum_{i=1}^{n} a_i \cdot x^{n-i} = \sum_{i=1}^{k} R[i] \cdot x^{i-1} + (\sum_{i=1}^{n} g_i \cdot x^i) \cdot (\sum_{i=1}^{n} b_i \cdot x^{n-i}).$$

Dowód. Lemat ten dowodzimy przez prostą indukcję ze względu na n. □

Z podanego lematu wynika, że iterując procedura D z parametrami $a_1, \ldots, a_n$ dzielimy wielomian $\sum_{i=1}^{n} a_i \cdot x^{n-i}$ przez wielomian $g(x) = \sum_{i=1}^{n} g_i \cdot x^i$.

Algorytm 5. Dane: wielomian $w(x) = \sum_{i=1}^{n} a_i \cdot x^{n-i} \in F^n[x]$.

1. Znajdź zbiór
$$\mathcal{S} = \{u(x) \bmod g(x) : u \in Z_2^n[x] \text{ ma stopień } n \text{ i} \leqslant t \text{ współczynników} \neq 0\};$$
2. dla I od 1 do k wykonaj $G[I] := g_I$ oraz $R[I] := 0$;
3. dla I od 1 do n wykonaj $D(a_I)$;
4. $W := w$;
5. $I := 0$;
6. **dopóki** $I < n + 1$
   (a) jeżeli $\sum_{i=1}^{k} R[i] \cdot x^{i-1} \in \mathcal{S}$, to wykonaj $W := W + x^n$ i $D(1)$, a w przeciwnym razie wykonaj $D(0)$;
   (b) $W := C(W)$;
   (c) $I := I + 1$
7. zwróć W.

Lemat 8.5 *Jeżeli K jest binarnym kodem cyklicznym generowanym przez wielomian $g(x) = \sum_{i=1}^{n} g_i \cdot x^i$, korygującym t błędów, to dla dowolnych w, v takich, że $w \in S_t(v)$ oraz $v \in K$, algorytm 5 uruchomiony z wielomianem w zwraca wielomian v (a więc algorytm 5 jest algorytmem korygującym błędy dla kodu K).*

Dowód. Algorytmy 3 i 5 różnią się tylko sposobem obliczania reszty W mod $g(x)$. Zgodnie z lematem 8.4, jeżeli $w(x) = \sum_{i=1}^{m} a_i \cdot x^{m-i}$, to po wykonaniu procedur $D(a_1), \ldots, D(a_m), D(0)$ w tablicy R są współczynniki reszty z dzielenia wielomianu $w(x) \cdot x$ przez wielomian $g(x) = \sum_{i=1}^{n} g_i \cdot x^i$, a po wykonaniu procedur $D(a_1), \ldots, D(a_m), D(1)$ w tablicy R są współczynniki reszty z dzielenia wielomianu $w(x) \cdot x + 1$ przez wielomian $g(x)$. Aby przekonać się o poprawności algorytmu 5, wystarczy sprawdzić, że

$$C(W + x^n) \bmod g(x) = (W \cdot x + 1) \bmod g(x)$$

oraz

$$C(W) \bmod g(x) = W \cdot x \bmod g(x).$$

Zauważmy, że ponieważ $g(x)$ dzieli $x^{n+1} - 1$, więc

$$\begin{aligned}
&C(W + x^n) \bmod g(x) = ((W + x^n) \cdot x - (w_n + 1) \cdot (x^{n+1} - 1)) \bmod g(x) \\
&= (W + x^n) \cdot x \bmod g(x) = (W \cdot x + (x^{n+1} - 1) + 1) \bmod g(x) \\
&= (W \cdot x + 1) \bmod g(x),
\end{aligned}$$

a także

$$C(W) \bmod g(x) = (W \cdot x - w_n \cdot (x^{n+1} - 1)) \bmod g(x) = W \cdot x \bmod g(x). \quad \square$$

Algorytmy 3 i 5 są dostosowane do cyklicznych kodów binarnych, ale trochę je komplikując można im nadać postać dającą się wykorzystać w przypadku dowolnych kodów cyklicznych.