Wykład 3.

3.1 Algorytm „dekodujący"

Niech H będzie macierzą o n wierszach i $2^n - 1$ kolumnach, i niech będzie to macierz układu równań definiującego kod Hamminga.

Rozważmy następujący algorytm:

Dane: słowo $\vec{x} \in \{0,1\}^{2^n-1}$;

1. $k := H\vec{x}^T$;
2. jeżeli $k = \vec{0}^T$, to zwróć dane słowo $\vec{x}$;
3. w przeciwnym razie ustal i takie, że i-ta kolumna macierzy H jest równa k i zwróć słowo $\vec{x} + \vec{e}_i$.

Nietrudno zauważyć, że podany algorytm zawsze zwraca słowo kodowe kodu Hamminga (pamiętajmy, że dodajemy w ciele Z_2, a więc $a + a = 0$ dla dowolnego a) oraz, że zwrócone słowo różni się od danego najwyżej na jednej pozycji. Tak więc jest to algorytm korygujący błędy pojawiające się podczas przesyłania słów kodowych tak, że zmianie ulega najwyżej jeden znak.

3.2 Przykład znanego kodu

Czasem przesyłając siedem bitów dołączmy do nich bit kontroli parzystości. Jeżeli przesyłamy bity $x_1, \ldots, x_7$, to dołączamy do nich bit $x_8 = x_1 + \ldots + x_7$ (dodawanie to dodawanie modulo 2). W ten sposób zamiast przesyłać siedem bitów przesyłamy słowo kodu definiowanego równaniem $x_1 + \ldots + x_8 = 0$, a więc pewnego kodu liniowego.

Zauważmy także, że wszystkie binarne kody liniowe definiowane układami równań liniowych są podobnej postaci. Jeżeli mamy dowolny jednorodny układ równań liniowych, to występujące w nim zmienne można podzielić na dwa rozłączne zbiory $x_1, \ldots, x_n$ i $y_1, \ldots, y_m$, i można sprowadzić go do postaci

$$x_i = \sum_{j=1}^{m} a_{i,j} y_j \tag{5}$$

dla $i = 1, \ldots, n$. Jeżeli układ ten definiuje pewien kod, to symbole słów kodowych możemy podzielić na dwa rodzaje: symbole znaczące, odpowiadające zmiennym y_j, i symbole kontrolne, odpowiadające zmiennym x_i (podział symboli na znaczące i kontrolne zależy od przedstawienia definicji kodu w postaci układu (5), a to przedstawienie nie jest jednoznaczne; widać to dobrze na przykładzie wspomnianego kodu). Słowa kodowe takiego kodu tworzymy wybierając w dowolny sposób symbole znaczące i uzupełniając je o symbole kontrolne zgodnie z wzorami (5). Zauważmy jeszcze, że wzory (5) mają szczególnie prostą postać wtedy, gdy rozważamy równania nad ciałem Z_2. Wtedy współczynniki a_j są równe 0 lub 1, a więc prawe strony tych wzorów są sumami symboli znaczących.

3.3 Trochę algebry liniowej

Będziemy rozważać tylko przestrzenie liniowe nad skończonym ciałem F. Założenie o skończoności F w wielu przypadkach można pominąć, ale wtedy dowody będą wymagały bardziej subtelnych metod. Zauważmy także, że ciała skończone różnią się istotnie np. od ciała liczb rzeczywistych. Ciało liczb rzeczywistych jest uporządkowane i zachodzi w nim nierówność $x < x + 1$. Tak więc $1 < 1 + 1 < 1 + 1 + 1 < 1 + 1 + 1 + 1 < \ldots$. Wobec tego w ciałach skończonych nie można określić porządku, który ma własności znane z teorii liczb rzeczywistych. Powoduje to, że przestrzeniach liniowych nad ciałami skończonymi są niezerowe wektory, których kwadrat skalarny jest równy 0, a także różne wektory, których odległość kartezjańska jest równa 0. Nie jest bowiem poprawne następujące rozumowanie: jeżeli $\sum_{i=1}^{n} x_i^2 = 0$, to $x_i^2 = 0$ dla wszystkich i. Dowód tej własności dla liczb rzeczywistych wykorzystuje prawdziwy w ciałach uprządkowany fakt, że jeżeli suma liczb nieujemnych jest równa 0, to są one równe 0.

Niech K będzie podprzestrzenią liniową przestrzeni F^n. Dopełnieniem ortogonalnym przestrzeni K nazywamy zbiór

$$K^{\perp} = \{\vec{y} \in F^n : \forall \vec{x} \in K \; \vec{x}\vec{y} = \vec{0}\}.$$

Nietrudno zauważyć, że zbiór $K^{\perp}$ jest podprzestrzenią liniową, a jeżeli X jest zbiorem generatorów K, to w definicji $K^{\perp}$ kwantyfikator „dla każdego elementu K" można zastąpić przez „dla każdego elementu X". Także jest oczywiste, że jeżeli K jest przestrzenią generowaną przez zbiór wektorów będących wierszami macierzy H, to $K^{\perp} = \{\vec{x} : H\vec{x}^T = \vec{0}^T\}$.

Lemat 3.1 *Niech $X = \{\vec{x}_1, \ldots, \vec{x}_n\}$ będzie zbiorem generatorów przestrzeni K. Zbiór X jest minimalnym zbiorem generatorów wtedy i tylko wtedy, gdy jest zbiorem wektorów liniowo niezależnych (gdy warunek $\sum_{i=1}^{n} a_i\vec{x}_i = \vec{0}$ implikuje, że $a_i = 0$ dla $i = 1, \ldots, n$).*

Dowód. Załóżmy, że X jest minimalnym zbiorem generatorów oraz $\sum_{i=1}^{n} a_i\vec{x}_i = \vec{0}$. Gdyby np. współczynnik a_n był różny od 0, to

$$\vec{x}_n = \sum_{i=1}^{n-1} \frac{a_i}{a_n}\vec{x}_i$$

i zbiór $\{\vec{x}_1, \ldots, \vec{x}_{n-1}\}$ także generowałby K. Oznaczałoby to, że X nie jest minimalnym zbiorem generatorów.

Jeżeli X nie jest minimalnym zbiorem generatorów, to np. zbiór $\{\vec{x}_1, \ldots, \vec{x}_{n-1}\}$ także generuje K. Tak więc

$$\vec{x}_n = \sum_{i=1}^{n-1} a_i\vec{x}_i$$

dla pewnych współczynników a_i. Równość ta może zostać zapisana równoważnie jako

$$\sum_{i=1}^{n-1} a_i\vec{x}_i + \vec{x}_n = \vec{0}$$

i oznacza, że X nie jest zbiorem liniowo niezależnym. □

Lemat 3.2 *Przypuśćmy, że $X = \{\vec{x}_1, \ldots, \vec{x}_m\}$ jest minimalnym zbiorem generatorów przestrzeni $K \subseteq F^m$. Wtedy funkcja $f : F^m \to K$ zdefiniowana wzorem*

$$f(a_1, \ldots, a_m) = \sum_{i=1}^{m} a_i \vec{x}_i$$

jest różnowartościowa i typu „na”.

Dowód. Jest to konsekwencja lematu 3.1. □

Przestrzeń liniową, która jest generowana przez n elementowy zbiór wektorów i nie jest generowana przez jakikolwiek podzbiór tego zbioru, nazywamy przestrzenią wymiaru n. Z lematu 3.2 wynika, że wszystkie minimalne zbiory generatorów są równoliczne, a więc, że jest to poprawna definicja.

Lemat 3.3 *Dla przestrzeni $K \subseteq F^m$ definiujemy w zbiorze F^m relację R_K przyjmując, że*

$$\vec{x} R_K \vec{y} \Leftrightarrow \vec{x} - \vec{y} \in K.$$

Relacja R_K jest relacją typu równoważności. Klasami abstrakcji tej relacji są zbiory

$$\vec{a} + K = \{\vec{a} + \vec{x} : \vec{x} \in K\}$$

(i tylko te zbiory). Każde dwie klasy abstrakcji relacji R_K są równoliczne.

Dowód. Zwrotność relacji R_K wynika stąd, że $\vec{0} \in K$, symetryczność jest konsekwencją zamkniętości K ze względu na mnożenie przez -1, a przechodniość – zamkniętości ze względu na dodawanie.

Zauważmy, że klasa abstrakcji wektora $\vec{x}$, czyli $\{\vec{y} : \vec{x} R_K \vec{y}\}$, jest równa $\vec{x} + K = \{\vec{x} + \vec{z} : \vec{z} \in K\}$. Niech $\vec{y}$ spełnia warunek $\vec{x} R_K \vec{y}$. Wtedy $\vec{z} = \vec{y} - \vec{x} \in K$ oraz $\vec{y} = \vec{x} + \vec{z} \in \vec{x} + K$. Z drugiej strony, jeżeli $\vec{z} \in K$, to $\vec{x} R_K (\vec{x} + \vec{z})$. Dowodzi to, że zbiór $\vec{x} + K$ jest zawarty w klasie abstrakcji wektora $\vec{x}$.

Oczywiście, funkcja $f : K \to \vec{x} + K$ zdefiniowana wzorem $f(\vec{y}) = \vec{x} + \vec{y}$ ustala równoliczność zbiorów K i $\vec{x} + K$. Stąd wszystkie klasy abstrakcji relacji R_K są równoliczne z K. □

Lemat 3.4 *Niech $f : F^n \to F^m$ będzie odwzorowaniem liniowym typu „na”. Wtedy $K = \{\vec{x} \in F^n : f(\vec{x}) = \vec{0}\}$ jest podprzestrzenią liniową przestrzeni F^n oraz relacja R_K (patrz lemat 3.3) spełnia równoważność*

$$\vec{x} R_K \vec{y} \Leftrightarrow f(\vec{x}) = f(\vec{y}).$$

Klasami abstrakcji relacji R_K są dokładnie zbiory $\{\vec{x} \in F^n : f(\vec{x}) = \vec{y}\}$ dla $\vec{y} \in F^m$. Relacja R_K ma tyle klas abstrakcji, ile elementów ma zbiór F^m.

Wniosek 3.5 *Jeżeli F jest ciałem skończonym i $f : F^n \to F^m$ jest odwzorowaniem liniowym typu „na”, to podprzestrzeń $\{\vec{x} \in F^n : f(\vec{x}) = \vec{0}\}$ jest równoliczna ze zbiorem F^{n-m}.*

Dowód. Wystarczy zauważyć, że zbiory postaci $\{\vec{x} \in F^n : f(\vec{x}) = \vec{y}\}$ dla $\vec{y} \in F^m$ są parami rozłączne, równoliczne i w sumie dają zbiór F^n. □

Lemat 3.6 *Niech H będzie macierzą o m wierszach i n kolumnach. Przypuśćmy, że wiersze macierzy H tworzą minimalny zbiór generujący pewną podprzestrzeń K. Wtedy funkcja $f : F^n \to F^m$ zdefiniowana wzorem*

$$f(\vec{x}) = H\vec{x}^T$$

jest odwzorowaniem liniowym typu „na”.

Dowód. Niech

$$H = \begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,n} \\ a_{2,1} & a_{2,2} & \dots & a_{2,n} \\ & & \vdots & \\ a_{m,1} & a_{m,2} & \dots & a_{m,n} \end{bmatrix}.$$

Poszczególne elementy kolumny $H\vec{x}^T$ są iloczynami skalarnymi wierszy macierzy H i $\vec{x}$. Wobec tego, liniowość funkcji f wynika z liniowości iloczynu skalarnego.

Niech $\vec{e}_j = (0, \dots, 1, \dots, 0) \in F^n$ będzie wektorem z dokładnie jedną jedynką na j-tej pozycji. Zauważmy, że $H\vec{e}_j^T$ jest równe j-tej kolumnie macierzy H. Stąd i z liniowości funkcji f wynika, że zbiorem wartości funkcji f jest podprzestrzeń F^m generowana przez kolumny macierzy H. Spośród kolumn H wybierzmy minimalny zbiór generujący zbiór wartości funkcji f. Aby uprościć zapis przyjmijmy, że składa się on z kolumn o numerach od 1 do k. Zbiór wartości f jest więc przestrzenią k wymiarową zawartą w F^m. Wobec tego, $k \leqslant m$. Przyjmijmy też, że $\vec{a}_i = (a_{i,1}, a_{i,2}, \dots, a_{i,k})$.

Ustalmy $j > k$. j-ta kolumna macierzy H jest kombinacją liniową kolumn o numerach od 1 do k. Przyjmijmy, że współczynnikami tej kombinacji są liczby $b_{j,1}, \dots, b_{j,k}$. Niech $\vec{b}_j = (b_{j,1}, \dots, b_{j,k})$. i-ty wyraz j-tej kolumny jest kombinacją liniową i-tych wyrazów kolumn o numerach od 1 do k ze współczynnikami $b_{j,1}, \dots, b_{j,k}$, a więc jest równy $\vec{b}_i\vec{a}_j$. Wynika stąd, że macierz H ma postać

$$\begin{bmatrix} a_{1,1} & \dots & a_{1,k} & \vec{b}_{k+1}\vec{a}_1 & \dots & \vec{b}_n\vec{a}_1 \\ a_{2,1} & \dots & a_{2,k} & \vec{b}_{k+1}\vec{a}_2 & \dots & \vec{b}_n\vec{a}_2 \\ & \vdots & & & & \\ a_{m,1} & \dots & a_{m,k} & \vec{b}_{k+1}\vec{a}_m & \dots & \vec{b}_n\vec{a}_m \end{bmatrix}.$$

Na podstawie lematu 3.1 stwierdzamy, że wiersze H są liniowo niezależne. Stąd wynika, że także wektory $\vec{a}_1, \dots, \vec{a}_m$ są liniowo niezależne. Weźmy współczynniki α_i takie, że

$$\sum_{i=1}^{m} \alpha_i \vec{a}_i = \vec{0}.$$

Pokażemy, że kombinacja liniowa wierszy macierzy H o tych samych współczynnikach jest wektorem zerowym, a więc wszystkie jej wyrazy są zerami. Najpierw zajmiemy się wyrazami o numerach $j \leqslant k$. j-ty wyraz kombinacji liniowej wierszy macierzy H jest dla takich j równy j-temu wyrazowi wektora $\sum_{i=1}^{m} \alpha_i \vec{a}_i$, a więc jest równy 0. Jeżeli $j > k$, to j-ty wyraz kombinacji liniowej wierszy H jest równy

$$\sum_{i=1}^{m} \alpha_i (\vec{b}_j \vec{a}_i) = \vec{b}_j (\sum_{i=1}^{m} \alpha_i \vec{a}_i) = 0.$$

Liniowa niezależność wierszy H implikuje, że wszystkie współczynniki α_i są równe 0. Dowodzi to liniowej niezależności wektorów $\vec{a}_1, \ldots, \vec{a}_m$.

Jeżeli wektory $\vec{a}_1, \ldots, \vec{a}_m \in F^k$ są liniowo niezależne, to na podstawie lematów 3.1 i 3.2 otrzymujemy, że przestrzeń F^k zawiera podprzestrzeń równoliczną ze zbiorem F^m. Jest to możliwe tylko wtedy, gdy $m \leqslant k$. Wobec tego, $m = k$ i podprzestrzeń F^m generowana przez k pierwszych kolumn macierzy H, czyli zbiór wartości f, jest równa F^m. A to z kolei oznacza, że funkcja f jest typu „na”. □

Wniosek 3.7 *Jeżeli K jest podprzestrzenią liniową wymiaru m zawartą w przestrzeni F^n nad skończonym ciałem F, to $K^\perp$ jest przestrzenią wymiaru $n - m$.* □

Wniosek 3.8 *Dla dowolnej podprzestrzeni K zawartej w przestrzeni F^n nad skończonym ciałem F zachodzi równość $(K^\perp)^\perp = K$.*

Dowód. Z definicji przestrzeni $K^\perp$ wynika, że

$$\forall \vec{x} \in K^\perp \forall \vec{y} \in K \vec{x}\vec{y} = \vec{0}.$$

Po przestawieniu kwantyfikatorów, zdanie to stwierdza, że $K \subseteq (K^\perp)^\perp$. Z poprzedniego wniosku otrzymujemy, że rozważane przestrzenie są tego samego wymiaru, a więc są równoliczne. Z podanego zawierania i skończoności tych przestrzeni wynika ich równość. □

Wniosek 3.9 *Dowolna podprzestrzeń K zawarta w przestrzeni F^n nad skończonym ciałem F jest zbiorem rozwiązań pewnego jednorodnego układu równań liniowych.*

3.4 Obliczanie zbioru generatorów

Niech I_n oznacza macierz jednostkową o n wierszach i n kolumnach. Będziemy rozważać macierz $H = [I_n, A]$ złożoną z dwóch macierzy: I_n oraz A o n wierszach i $m - n$ kolumnach. Macierz H ma więc n wierszy i m kolumn.

Lemat 3.10 *Kod $K = \{\vec{x} \in \{0,1\}^m : H\vec{x}^T = \vec{0}^T\}$ jest generowany (jako przestrzeń liniowa) przez wiersze macierzy $\left[A^T, I_{m-n}\right]$.*

Dowód. Niech $a_{i,j}$ oznacza wyraz znajdujący się w i-tym wierszu i j-tej kolumnie macierzy A. Weźmy i-ty wiersz $\vec{x}$ macierzy $\left[A^T, I_{m-n}\right]$. jest to ciąg postaci $(a_{1,i}, \ldots, a_{n,i}, 0, \ldots, 1, \ldots, 0)$ z jedynką ma miejscu o numerze $n + i$. Podobnie, j-ty wyraz $\vec{y}$ macierzy H ma postać $(0, \ldots, 1, \ldots, 0, a_{j,1}, \ldots, a_{j,m-n})$, przy czym jedynka jest j-tym wyrazem tego ciągu. Iloczyn skalarny $\vec{x}\vec{y}$ jest dany wzorem

$$\vec{x}\vec{y} = a_{j,i} + a_{j,i} = 0.$$

Wobec tego $\vec{x} \in K$.

Niech K' będzie przestrzenią generowaną przez wiersze macierzy $\left[A^T, I_{m-n}\right]$. Łatwo sprawdzić, że K' ma wymiar $m - n$. Pokazaliśmy, że jest to podprzestrzeń przestrzeni K. Wniosek 3.7 stwierdza, że wymiar K jest także równy $m - n$. Tak więc $K' = K$. □

3.5 Algorytm dekodujący dla dowolnego kodu liniowego

Będziemy rozważać kod liniowy $K \subseteq F^n$.

Przypuśćmy, że po wysłaniu słowa kodowego x odbiorca otrzymał słowo y. Podczas przesyłania wystąpił więc błąd $e = y - x$. Tak więc $y - e = x \in K$.

Zdefiniujmy w zbiorze F^n relację R_K żądając, aby

$$aR_Kb \Leftrightarrow a - b \in K$$

dla dowolnych a i b. Nietrudno zauważyć, że R_K jest relacją równoważności. Rozbija więc zbiór F^n na rozłączne klasy abstrakcji. Dowodzi się klasy abstrakcji tej relacji są równoliczne.

Jeżeli $y - e = x \in K$, to y i e należą do tej samej klasy abstrakcji relacji R_K. Mając dane słowo y możemy ustalić przesłane słowo x szukając w klasie abstrakcji słowa y błędu e. Zauważmy, że w każdej klasie abstrakcji relacji R_K może być najwyżej jeden błąd, który może zostać skorygowany. Gdyby znalazły się tam dwa błędy e_1 i e_2, to słowo $x = e_1 - e_2$ należałoby do K. Wtedy zarówno po wysłaniu wektora zerowego i wystąpienia błędu e_1, jak i po wysłaniu słowa x i wystąpieniu błędu e_2 otrzymalibyśmy słowo e_1 i – wobec tego – tych błędów nie potrafilibyśmy skorygować. Jest oczywiste, że jeżeli potrafimy skorygować błąd e_1, to potrafimy także skorygować błąd mniej odległy (w sensie odległości Hamminga) od zera niż e_1. Stąd wynika, że błędem w dowolnej klasie abstrakcji może być tylko wektor leżący najbliżej wektora zerowego. Sugeruje to następujący algorytm dekodujący (korygujący błędy):

Dane: słowo y;

1. znajdujemy klasę abstrakcji X relacji R_K wyznaczoną przez y;

2. znajdujemy w zbiorze X (którykolwiek) wektor e najmniej odległy od wektora zerowego;

3. zwracamy $y - e$.

Lemat 3.11 *Jeżeli kod K koryguje t błędów i $y \in S_t(x)$ dla pewnego $x \in K$, to podany algorytm po uruchomieniu ze słowem y zwraca x.*

Dowód. Przypuśćmy, że podany algorytm po uruchomieniu ze słowem y znajdzie klasę abstrakcji X wyznaczoną przez y, a w niej najmniej odległy od zerowego wektor e. Wobec tego $y - e = z$ dla pewnego $z \in K$. Wektor z jest tym, który został zwrócony przez rozważany algorytm.

Niech $e' = y - x$. Oczywiście $e' \in X$ i, w konsekwencji, $d(0, e) \leqslant d(0, e') \leqslant t$. Mamy więc, że $y \in S_t(z)$. Ponieważ kod K koryguje t błędów, $y \in S_t(x)$ oraz $x, z \in K$, więc $x = z$. Oznacza to, że rozważany algorytm po uruchomieniu ze słowem y zwrócił słowo x. □

Niech $a_{1,1}, \ldots, a_{1,m}$ będzie ciągiem złożonym dokładnie ze słów należących do K. Załóżmy dodatkowo, że $a_{1,1}$ jest słowem zerowym. Z każdej klasy abstrakcji relacji R_K wybierzmy słowo najmniej odległe od słowa zerowego. Niech $a_{2,1}, \ldots, a_{k,1}$ będą wybranymi słowami. Słowami $a_{1,1}, \ldots, a_{1,m}$ oraz $a_{2,1}, \ldots, a_{k,1}$ wypełniamy odpowiednio pierwszy wiersz i pierwszą kolumnę macierzy A. Pozostałe elementy tej macierzy definiujemy zgodnie z wzorem $a_{i,j} = a_{i,1} + a_{1,j}$. Tak zdefiniowaną macierz nazywamy macierzą standardową kodu K. Nietrudno zauważyć, że w macierzy A znajdują się wszystkie elementy F^n, a zbiory elementów poszczególnych wierszy tej macierzy są wszystkimi możliwymi klasami abstrakcji relacji R_K.

Mając macierz standardową $A = (a_{i,j})_{i,j}$ wyżej podany algorytm można implementować w następujący sposób:

Dane: słowo y;

1. znajdujemy w macierzy A element $a_{i,j}$ równy y.
2. zwracamy $a_{1,j}$.