

# Kody korekcyjne: Lista 11

12 grudnia 2019

**Zadanie 1.** Pokaż, że dla rodziny kodów Wozencrafta  $\{C_{\text{in}}^\alpha\}_{\alpha \in \mathbb{F}_Q \setminus \{0\}}$  zachodzi następujący fakt:

Niezerowy wektor  $\vec{y} \in \mathbb{F}_Q^2$  należy do najwyżej jednego kodu  $C_{\text{in}}^\alpha$ .

Dla przypomnienia,  $C_{\text{in}}^\alpha = \{(x, \alpha x) : x \in \mathbb{F}_Q\}$ .

**Zadanie 2.** Pokaż, jak poprawić błędy i wymazania w kodzie RS (o parametrach  $n, k, d = n - k + 1$ ), jeśli liczba błędów ( $e$ ) i wymazań ( $s$ ) spełnia warunek

$$2e + s < d .$$

*Wskazówka:* Co jeśli  $e = 0$ ? W pozostającym przypadku postuż się algorytmem BW poprawiającym błędy jako czarną skrzynką. Usuń na początek wymazania, jak to zmienić kod?

**Zadanie 3.** Rozważ korekcję błędów dla kodów skonkatenowanych, w których wymazujemy słowa mające odległość od najbliższego słowa kodowego  $> \alpha d$  oraz poprawiamy te mającą odległość  $\leq \alpha d$  (możesz zamienić nierówności słabą i ostrą miejscami, jeśli to pomoże). Jak dużo błędów potrafi poprawić ten algorytm? Jaka jest optymalna wartość  $\alpha$ ?

**Zadanie 4** (Z gwiazdką, sam jeszcze nie wiem, co wychodzi). W algorytmie powyżej algorytm nie bierze pod uwagę dwóch parametrów: ile słów nie jest słowami kodowymi (kodu wewnętrznego) oraz jaka jest suma odległości od słów kodowych (kodu wewnętrznego)

Czy potrafisz poprawić algorytm z Zadania 3 używając tych parametrów?

*Wskazówka:* Jeśli suma odległości jest duża, np. blisko  $Dd/2$ , to w bardzo niewielu miejscach możemy mieć błąd  $\geq d/2$ , czyli należy więcej poprawiać. Analogicznie, jeśli jest dużo słów z błędami, to mało z nich może mieć błąd  $\geq d/2$  i też należy więcej poprawiać.

**Zadanie 5.** Analiza poprawności algorytmu poprawiania błędów dla kodów skonkatenowanych pokazuje, że dla pewnej wartości progu  $\theta$  algorytm, który poprawia błędy, jeśli odległość do najbliższego słowa kodowego jest  $\leq \theta$  oraz wymazuje, jeśli jest  $> \theta$ , potrafi poprawić  $e < Dd/2$  błędów. Czy umiesz to pokazać nie odwołując się do losowania?

*Wskazówka:* Użyj wagi: waga dla elementu kodu wewnętrznego to 0, gdy jest zgodny z przesłaną wiadomością,  $d/2$  jeśli jest wymazany i  $d$ , jeśli źle poprawiliśmy błąd. Rozpatrz, ile jest słów kodowych w odległości  $i$  oraz  $i$  od złego słowa kodowego i jaka jest waga, jeśli wszystkie te słowa zostaną wymazane lub poprawione.