Коды

Т. А. Новикова

Факультет ВМиК
Казахстанский филиал МГУ им.М. В. Ломоносова

5 мая 2016 г.

Пусть $A = \{a_1, a_2, \ldots, a_r\}$ — исходный алфавит,
$B = \{b_1, \ldots, b_m\}$ — кодирующий алфавит и

$$A^* = \emptyset \cup A \cup A^2 \cup \ldots A^n \cup \ldots,$$

$$B^* = \emptyset \cup B \cup B^2 \cup \ldots B^n \cup \ldots.$$

Под A^n будем понимать все слова длины n в алфавите A.

Пусть $A = \{a_1, a_2, \ldots, a_r\}$ — исходный алфавит,
$B = \{b_1, \ldots, b_m\}$ — кодирующий алфавит и

$$A^* = \emptyset \cup A \cup A^2 \cup \ldots A^n \cup \ldots,$$

$$B^* = \emptyset \cup B \cup B^2 \cup \ldots B^n \cup \ldots.$$

Под A^n будем понимать все слова длины n в алфавите A.

Definition

Алфавитным кодированием $A^* \to B^*$ назовем отображение $\phi : A \to B^*$ такое, что $a_i \to B_i$.

Пусть $A = \{a_1, a_2, \dots, a_r\}$ — исходный алфавит,
$B = \{b_1, \dots, b_m\}$ — кодирующий алфавит и

$$A^* = \emptyset \cup A \cup A^2 \cup \dots A^n \cup \dots,$$

$$B^* = \emptyset \cup B \cup B^2 \cup \dots B^n \cup \dots.$$

Под A^n будем понимать все слова длины n в алфавите A.

Definition

Алфавитным кодированием $A^* \to B^*$ назовем отображение $\phi : A \to B^*$ такое, что $a_i \to B_i$.

Множество $\{B_1, \dots, B_r\}$ называется множеством кодовых слов и считается, что $\phi : a_{i_1} a_{i_2} \dots a_{i_k} \to B_{i_1} B_{i_2} \dots B_{i_k}$.

Договоримся обозначать $\overline{a}$ последовательную конкатенацию символов.

Кодирование $A^* \to B^*$ называется взаимно однозначным (декодируемым, разделимым), если для любых слов $\overline{a_1} \in A^*, \overline{a_2} \in A^*$ верно, что $\overline{a_1} \neq \overline{a_2} \Rightarrow \phi(\overline{a_1}) \neq \phi(\overline{a_2})$.

Договоримся обозначать $\overline{a}$ последовательную конкатенацию символов.
Кодирование $A^* \to B^*$ называется взаимно однозначным (декодируемым, разделимым), если для любых слов $\overline{a_1} \in A^*, \overline{a_2} \in A^*$ верно, что $\overline{a_1} \neq \overline{a_2} \Rightarrow \phi(\overline{a_1}) \neq \phi(\overline{a_2})$.

Код называется равномерным, если длины всех его кодовых слов одинаковы.

Договоримся обозначать $\overline{a}$ последовательную конкатенацию символов.

Кодирование $A^* \to B^*$ называется взаимно однозначным (декодируемым, разделимым), если для любых слов $\overline{a_1} \in A^*, \overline{a_2} \in A^*$ верно, что $\overline{a_1} \neq \overline{a_2} \Rightarrow \phi(\overline{a_1}) \neq \phi(\overline{a_2})$.

Код называется равномерным, если длины всех его кодовых слов одинаковы.

Утв.1: любой равномерный код является взаимно однозначным.

Код называется префиксным, если никакое кодовое слово не является началом другого.

Код называется префиксным, если никакое кодовое слово не является началом другого.

Утв.2: любое префиксное кодирование является взаимно однозначным.

Код называется префиксным, если никакое кодовое слово не является началом другого.

Утв.2: любое префиксное кодирование является взаимно однозначным.

Код называется постфиксным (суффиксным), если никакое кодовое слово не является концом другого.

Код называется префиксным, если никакое кодовое слово не является началом другого.

Утв.2: любое префиксное кодирование является взаимно однозначным.

Код называется постфиксным (суффиксным), если никакое кодовое слово не является концом другого. Утв.3: любое постфиксное кодирование является взаимно однозначным.

Definition

Слово $\bar{b} \in B^*$ называется неприводимым, если $\bar{b}$ декодируется неоднозначно, однако, при выбрасывании из $\bar{b}$ любого связного непустого куска получается слово, которое декодируется не более, чем одним способом.

Theorem (Марков А.А.)

Пусть $\phi : a_i \to B_i$ — некоторое кодирование. Пусть W — максимальное число кодовых слов, которые помещаются подряд внутри кодового слова. Пусть l_i — длина слова B_i, $L = \sum_{i=1}^{r} l_i$. Тогда если кодирование ϕ не взаимно однозначно, то существуют два различных слова $a' \in A^*, a'' \in A^*$,

$$len(a') \leq \left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor,$$

$$len(a'') \leq \left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor$$

и при этом $\phi(a') = \phi(a'')$.

Доказательство. Пусть ϕ не является взаимно однозначным. Тогда существует некоторое слово $\overline{b_1}$, которое допускает две расшифровки. Если слово $\overline{b_1}$ не является неприводимым, то выбрасывая из $\overline{b_1}$ куски несколько раз, получим неприводимое слово $\overline{b}$ иначе, положим $\overline{b} = \overline{b_1}$. Это всегда можно сделать.

Доказательство. Пусть ϕ не является взаимно однозначным. Тогда существует некоторое слово $\overline{b_1}$, которое допускает две расшифровки. Если слово $\overline{b_1}$ не является неприводимым, то выбрасывая из $\overline{b_1}$ куски несколько раз, получим неприводимое слово $\overline{b}$ иначе, положим $\overline{b} = \overline{b_1}$. Это всегда можно сделать.

Рассмотрим любые две декодировки слова $\overline{b}$. Разрежем слово $\overline{b}$ в концевых точках кодовых слов каждого из разбиений. Слова нового разбиения разделим на два класса: к I классу отнесём слова, являющиеся элементарными кодами, а ко II классу — все слова, являющиеся началами кодовых слов одного разбиения и концами слов второго разбиения.

Lemma

Если $\overline{b}$ — неприводимое слово, то все слова $\beta_1, \beta_2, \ldots, \beta_m$ II класса различны.

Lemma

Если $\overline{b}$ — неприводимое слово, то все слова $\beta_1, \beta_2, \ldots, \beta_m$ II класса различны.

Доказательство. Пусть $\beta' = \beta''$. Тогда очевидно, что слово $\overline{b}$ не будет неприводимым, поскольку тогда при выкидывании отрезка между β', β'' вместе с любым из этих слов, получим снова две различные расшифровки этого слова.

Продолжим доказательство теоремы Маркова. Получается, что все $\beta_1, \ldots, \beta_m$ разные. Тогда число слов второго класса не превосходит числа непустых начал элементарных кодов, то есть не превосходит

$$(l_1 - 1) + (l_2 - 1) + \ldots + (l_r - 1) = L - r.$$

Продолжим доказательство теоремы Маркова.
Получается, что все $\beta_1, \ldots, \beta_m$ разные. Тогда число слов второго класса не превосходит числа непустых начал элементарных кодов, то есть не превосходит

$$(l_1 - 1) + (l_2 - 1) + \ldots + (l_r - 1) = L - r.$$

Слова из второго класса разбивают слово не более чем на $L - r + 1$ кусков. Рассмотрим пары соседних кусков. Тогда согласно одному разбиению в одной половинке уложится не более одного кодового слова, а в другой — не более W (согласно второму разбиению ситуация симметрична).

Всего пар кусков не больше, чем

$$\left\lceil \frac{L-r+1}{2} \right\rceil \le \frac{L-r+2}{2},$$

а в каждом из них укладывается слов не более чем $W+1$. Отсюда число кодовых слов в любом разбиении не превосходит $\frac{L-r+2}{2}(W+1)$, а т.к. число целое, то не превосходит целой части $\left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor$.

$\square$

Theorem (Неравенство Макмиллана.)

Пусть задано кодирование $\varphi : a_i \to B_i, i = 1, \ldots, r$ и пусть в кодирующем алфавите $B - q$ букв и $len(B_i) = l_i$. Тогда если φ взаимно однозначно, то

$$\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1.$$

Theorem (Неравенство Макмиллана.)

Пусть задано кодирование $\varphi : a_i \to B_i, i = 1, \dots, r$ и пусть в кодирующем алфавите $B - q$ букв и $len(B_i) = l_i$. Тогда если φ взаимно однозначно, то

$$\sum_{i-1}^{r} \frac{1}{q^{l_i}} \le 1.$$

Доказательство. Положим $x = \sum_{i-1}^{r} \frac{1}{q^{l_i}}$. Тогда для любого натурального n:

$$x^n = (\sum_{i_1=1}^{r} \frac{1}{q^{i_1}})(\sum_{i_2=1}^{r} \frac{1}{q^{i_2}}) \dots (\sum_{i_n=1}^{r} \frac{1}{q^{i_n}}) = \sum_{i_1=1}^{r} \sum_{i_2=1}^{r} \dots \sum_{i_n=1}^{r} \frac{1}{q^{l_{i_1}+l_{i_2}+\dots+l_{i_n}}}.$$

Обозначая $l_{max} = max_{1 \le i \le r} l_i$, получим, что эта сумма равна $\sum_{k=1}^{n \cdot l_{max}} \frac{c_k}{q^k}$ для некоторых c_k.

Lemma

Для любого k верно $c_k \leq q^k$.

Доказательство. За c_k в предыдущей формуле фактически обозначено число наборов $(i_1, \ldots, i_n)$, для которых $l_{i_1} + l_{i_2} + \ldots + l_{i_n} = k$. Но такой сумме соответствует слово $B_{i_1} B_{i_2} \ldots B_{i_n}$ и $len(B_{i_1} B_{i_2} \ldots B_{i_n}) = l_{i_1} + l_{i_2} + \ldots + l_{i_n} = k$.

Lemma

Для любого k верно $c_k \leq q^k$.

Доказательство. За c_k в предыдущей формуле фактически обозначено число наборов $(i_1, \ldots, i_n)$, для которых $l_{i_1} + l_{i_2} + \ldots + l_{i_n} = k$. Но такой сумме соответствует слово $B_{i_1} B_{i_2} \ldots B_{i_n}$ и $len(B_{i_1} B_{i_2} \ldots B_{i_n}) = l_{i_1} + l_{i_2} + \ldots + l_{i_n} = k$. В силу однозначности кодирования различным наборам соответствуют различные сообщения, а различных сообщений длины k в алфавите из q букв не более q^k, значит, $c_k \leq q^k$.

Согласно лемме, $x^n = \sum_{k=1}^{n \cdot l_{max}} \frac{c_k}{q^k} \leq \sum_{k=1}^{n \cdot l_{max}} 1 = nl_{max}$, но это равнозначно тому, что $x \leq \sqrt[n]{nl_{max}}, \forall n$. Устремим n к бесконечности, получим, что $x \leq 1$. $\square$

Theorem

Если $|B| = q$ и натуральные числа $l_1, l_2, \ldots, l_r$ удовлетворяют неравенству

$$\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1,$$

то существует префиксный код $B_1, B_2, \ldots, B_r$ такой, что $len(B_i) = l_i$.

Theorem

Если $|B| = q$ и натуральные числа $l_1, l_2, \ldots, l_r$ удовлетворяют неравенству

$$\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1,$$

то существует префиксный код $B_1, B_2, \ldots, B_r$ такой, что $len(B_i) = l_i$.

Доказательство. Пусть $\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1$ и для любого k существует ровно d_k таких i, что $l_i = k$, то есть $\sum_{k=1}^{l_{max}} \frac{d_k}{q^k} \leq 1$.

Theorem

Если $|B| = q$ и натуральные числа $l_1, l_2, \ldots, l_r$ удовлетворяют неравенству

$$\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1,$$

то существует префиксный код $B_1, B_2, \ldots, B_r$ такой, что $len(B_i) = l_i$.

Доказательство. Пусть $\sum_{i-1}^{r} \frac{1}{q^{l_i}} \leq 1$ и для любого k существует ровно d_k таких i, что $l_i = k$, то есть $\sum_{k=1}^{l_{max}} \frac{d_k}{q^k} \leq 1$.

Тогда надо построить префиксный код, в котором ровно d_1 слов длины 1, d_2 слов длины 2 и т.д. Имеем, что для любого m: $\sum_{k=1}^{m} \frac{d_k}{q^k} \leq 1$, а это означает:

$$\frac{d_1}{q} + \frac{d_2}{q^2} + \ldots + \frac{d_{m-1}}{q^{m-1}} + \frac{d_m}{q^m} \leq 1$$

$$\Leftrightarrow d_m \leq q^m - (d_1 q^{m-1} + d_2 q^{m-2} + \ldots + d_{m-1} q).$$

Рассмотрим это неравенство для $m = 1 : d_1 \leq q$. Для слов длины 1 всего предоставляется ровно q вариантов в алфавите мощности q.

Рассмотрим это неравенство для $m = 1 : d_1 \leq q$. Для слов длины 1 всего предоставляется ровно q вариантов в алфавите мощности q.

После выбора d_1 слов длины 1 рассмотрим неравенство для $m = 2 : d_2 \leq q^2 - d_1 q$. Всего слов длины 2 существует q^2, но все они могут начинаться лишь с тех букв, которые не были выбраны в качестве слов длины 1, значит, остается ровно $q^2 - d_1 q$ возможностей выбрать слова длины 2, что удовлетворяет $d_2 \leq q^2 - d_1 q$.

Рассмотрим это неравенство для $m = 1 : d_1 \leq q$. Для слов длины 1 всего предоставляется ровно q вариантов в алфавите мощности q.

После выбора d_1 слов длины 1 рассмотрим неравенство для $m = 2 : d_2 \leq q^2 - d_1 q$. Всего слов длины 2 существует q^2, но все они могут начинаться лишь с тех букв, которые не были выбраны в качестве слов длины 1, значит, остается ровно $q^2 - d_1 q$ возможностей выбрать слова длины 2, что удовлетворяет $d_2 \leq q^2 - d_1 q$.

Если мы таким образом выберем необходимое количество слов длины 1, 2 и т.д., до слов длин $m - 1$. Тогда для слов длины m разрешено возможностей не меньше, чем $q^m - d_{m-1} q - d_{m-2} q^2 - \ldots - d_2 q^{m-2} - d_1 q^{m-1}$, что удовлетворяет условию. $\square$

Оптимальные коды

Будем рассматривать кодирование в алфавит $\{0, 1\}$. Пусть известны некоторые частоты $a_1 : p_1, a_2 : p_2, \ldots, a_k : p_k$ появления символов кодируемого алфавита в тексте.

Оптимальные коды

Будем рассматривать кодирование в алфавит $\{0, 1\}$. Пусть известны некоторые частоты $a_1 : p_1, a_2 : p_2, \ldots, a_k : p_k$ появления символов кодируемого алфавита в тексте.

Definition

Ценой (стоимостью, избыточностью) кодирования φ называется функция $c(\varphi) = \sum_{i=1}^{k} p_i l_i$. При кодировании текста длины N его длина становится примерно равной

$$\sum_{i=1}^{k} (Np_i) l_i = N \sum_{i=1}^{k} p_i l_i .$$

Definition

Взаимно однозначное кодирование φ называется оптимальным, если на нем достигается $inf_{\varphi} c(\varphi)$, где грань достигается по всем взаимно однозначным кодированиям φ.

Definition

Взаимно однозначное кодирование φ называется оптимальным, если на нем достигается $inf_{\varphi} c(\varphi)$, где грань достигается по всем взаимно однозначным кодированиям φ.

Справедливо утверждение: если существует оптимальный код, то существует оптимальный префиксный код с тем же спектром длин слов.

Lemma

Если φ — оптимальное кодирование и $p_i > p_j$, то $l_i \leq l_j$.

Доказательство. Допустим, что $p_i > p_j$ и $l_i > l_j$. Рассмотрим кодирование φ и рассмотрим кодирование φ', в котором переставим кодовые слова B_i, B_j: если раньше $a_i \to B_i, a_j \to B_j$, то теперь $a_i \to B_j, a_j \to B_i$.

Lemma

Если φ — оптимальное кодирование и $p_i > p_j$, то $l_i \leq l_j$.

Доказательство. Допустим, что $p_i > p_j$ и $l_i > l_j$. Рассмотрим кодирование φ и рассмотрим кодирование φ', в котором переставим кодовые слова B_i, B_j: если раньше $a_i \to B_i, a_j \to B_j$, то теперь $a_i \to B_j, a_j \to B_i$.

Тогда
$c(\varphi) - c(\varphi') = (p_i l_i + p_j l_j) - (p_i l_j + p_j l_i) = (p_i - p_j)(l_i - l_j) > 0$, значит, $c(\varphi') < c(\varphi)$, то есть φ не является оптимальным $\Rightarrow$?!.

Lemma

Если φ — оптимальное префиксное кодирование и $l_{max} = max l_i$, $len(B_j) = l_{max}$, $B_j = B'_j\alpha$, где $\alpha \in \{0, 1\}$, то в коде существует слово B_r такое, что $B_r = B'_j\bar{\alpha}$.

Lemma

Если φ — оптимальное префиксное кодирование и $l_{max} = max l_i$, $len(B_j) = l_{max}$, $B_j = B_j'\alpha$, где $\alpha \in \{0, 1\}$, то в коде существует слово B_r такое, что $B_r = B_j'\bar{\alpha}$.

Доказательство. Допустим, что в φ нет слова $B_j'\bar{\alpha}$. Тогда заменим в φ $B_j'\alpha$ на B_j'. Получим код φ', который является префиксным, но

$$c(\varphi) - c(\varphi') = p_j len(B_j'\alpha) - p_j len(B_j') = p_j,$$

но тогда снова код φ не является оптимальным — ?!.

Lemma

Если φ — оптимальное префиксное кодирование и $p_1 \geq p_2 \geq \ldots \geq p_{k-1} \geq p_k$, то можно так переставить в коде φ, что получится оптимальное префиксное кодирование φ' такое, что слова B'_{k-1}, B'_k в нем будут отличаться только в последнем разряде.

Lemma

Если φ — оптимальное префиксное кодирование и $p_1 \geq p_2 \geq \ldots \geq p_{k-1} \geq p_k$, то можно так переставить в коде φ, что получится оптимальное префиксное кодирование φ' такое, что слова B'_{k-1}, B'_k в нем будут отличаться только в последнем разряде.

Доказательство. Пусть $p_1 \geq p_2 \geq \ldots \geq p_{k-1} \geq p_k$ю По предыдущей лемме в коде φ есть слова $B'0, B'1$ максимальной длины. Поменяем их местами с B_{k-1}, B_k. Так как $p_{k-1} \leq p_i$, $p_k \leq p_i$ для $1 \leq i \leq k_2$, то цена кодирования не увеличится и код останется оптимальным.

Lemma

Рассмотрим кодирования $\varphi : B_1(p_1), B_2(p_2), \ldots, B_k(p_k)$ и $\varphi' : B_1(p_1), \ldots, B_{k-1}(p_{k-1}), B_k0(p'), B_k1(p'')$, где $p' + p'' = p_k$. Если один из этих наборов префиксный, то второй также префиксный и $c(\varphi') = c(\varphi) + p_k$.

Lemma

Рассмотрим кодирования $\varphi: B_1(p_1), B_2(p_2), \ldots, B_k(p_k)$ и $\varphi': B_1(p_1), \ldots, B_{k-1}(p_{k-1}), B_k0(p'), B_k1(p'')$, где $p' + p'' = p_k$. Если один из этих наборов префиксный, то второй также префиксный и $c(\varphi') = c(\varphi) + p_k$.

Доказательство. Рассмотрим

$$c(\varphi') - c(\varphi) = p' len(B_k0) + p'' len(B_k1) - p_k len(B_k) =$$
$$= p'(l_k + 1) + p''(l_k + 1) - p_k l_k =$$
$$= (p' + p'')l_k + (p' + p'') - p_k l_k = p_k.$$

Theorem (Теорема редукции.)

Пусть заданы два набора частот и два набора слов:
$\varphi : B_1(p_1), B_2(p_2), \ldots, B_k(p_k)$ и
$\varphi' : B_1(p_1), \ldots, B_{k-1}(p_{k-1}), B_k0(p'), B_k1(p'')$.

1. Тогда если φ' — оптимальное префиксное кодирование, то и φ — оптимальное префиксное кодирование.
2. Если же φ — оптимальное префиксное кодирование и $p_1 \geq p_2 \geq \ldots \geq p_{k-1} \geq p' \geq p''$, то φ' – также оптимальное префиксное кодирование.

Доказательство. 1). Префиксность исходного кода очевидна. Покажем его оптимальность. Пусть это не так, φ — не оптимально. Тогда существует префискный код $\varphi_1 : c(\varphi_1) < c(\varphi)$ для тех же распределений частот.

Доказательство. 1). Префиксность исходного кода очевидна. Покажем его оптимальность. Пусть это не так, φ — не оптимально. Тогда существует префискный код $\varphi_1 : c(\varphi_1) < c(\varphi)$ для тех же распределений частот.

Пусть $\varphi_1 : D_1(p_1), D_2(p_2), \ldots, D_k(p_k)$. Рассмотрим новое кодирование: $\varphi_1' : D_1(p_1), D_2(p_2), \ldots, D_k0(p'), D_k1(p'')$. Вновь полученное кодирование также является префиксным и из того, что $c(\varphi') = c(\varphi) + p_k, c(\varphi_1') = c(\varphi_1) + p_k$ следует, что $c(\varphi_1) < c(\varphi)$, а это означает, что
$c(\varphi_1') = c(\varphi_1) + p_k < c(\varphi) + p_k = c(\varphi')$.
Но тогда φ' не является оптимальным кодированием, что противоречит условию $\Rightarrow \varphi$ оптимально.

2). Пусть φ — оптимальное префиксное кодирование и $p_1 \geq p_2 \geq \ldots \geq p_{k-1} \geq p' \geq p''$ю Допустим, что φ' не оптимально. Тогда для частот $p_1, p_2, \ldots, p_{k-1}, p', p''$ существует оптимальное префиксное кодирование $\varphi_1' : D_1, \ldots, D_{k-1}, D_k0, D_k1$ и $c(\varphi_1') < c(\varphi)$. Тогда для частот $p_1, p_2, \ldots, p_k$ рассмотрим кодирование $\varphi_1 : D_1, \ldots, D_{k-1}, D_k$. Получим

$$c(\varphi_1) = c(\varphi_1') - p_k < c(\varphi') - p_k = c(\varphi) \Rightarrow c(\varphi_1) < c(\varphi)$$

и φ не оптимально, что противоречит условию. $\square$

В этом разделе будем рассматривать равномерные коды с длиной кодового слова n, а также ошибки замещения, когда бит α заменяется на бит $\overline{\alpha}$.

В этом разделе будем рассматривать равномерные коды с длиной кодового слова n, а также ошибки замещения, когда бит α заменяется на бит $\overline{\alpha}$.

Definition

Код называется исправляющим r ошибок, если при наличии в любом кодовом слове не более r ошибок типа замещения можно восстановить исходное кодовое слово.

Definition

Расстоянием Хэмминга $\rho(\widetilde{\alpha_1}, \widetilde{\alpha_2})$ между 2 наборами длины n называется число разрядов, в которых эти наборы различаются.

Definition

Шаром (сферой) радиуса r с центром в точке $\widetilde{\alpha} = (\alpha_1, \ldots, \alpha_n)$ называется множество всех наборов длины n, расстояние от которых до $\widetilde{\alpha}$ не превосходит r (в точности равно r).

Definition

Кодовым расстоянием называется расстояние Хэмминга: $\rho_{min} = min_{\alpha_i,\alpha_j}\rho(\widetilde{\alpha}_i, \widetilde{\alpha}_j)$.

Утверждение. Код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \ldots, \widetilde{\alpha}_m\}$ исправляет r ошибок тогда и только тогда, когда $\rho_{min}(K) \geq 2r + 1$.

Definition

Кодовым расстоянием называется расстояние Хэмминга: $\rho_{min} = min_{\alpha_i,\alpha_j}\rho(\widetilde{\alpha}_i, \widetilde{\alpha}_j)$.

Утверждение. Код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \ldots, \widetilde{\alpha}_m\}$ исправляет r ошибок тогда и только тогда, когда $\rho_{min}(K) \geq 2r + 1$.

Идея доказательства. Рассмотрим шары, соответствующие наборам, их радиус должен совпадать с r. Чтобы эти шары не пересекались, расстояние должно быть не меньше, чем $2r + 1$.

Definition

Код обнаруживает r ошибок, если при наличии в нём не более r ошибок типа замещения можно сказать, были ошибки, или их не было.

Утверждение. Код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \dots, \widetilde{\alpha}_m\}$ обнаруживает r ошибок тогда и только тогда, когда $\rho_{min}(K) \geq r + 1$.

Definition

Код обнаруживает r ошибок, если при наличии в нём не более r ошибок типа замещения можно сказать, были ошибки, или их не было.

Утверждение. Код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \ldots, \widetilde{\alpha}_m\}$ обнаруживает r ошибок тогда и только тогда, когда $\rho_{min}(K) \geq r + 1$.

Идея доказательства. Условие утверждения эквивалентно тому, что ни один из центров шаров (кодовое слово) не содержится в каком-либо другом шаре.

Definition

Функция $M_r(n)$ есть есть максимальное число слов длины n, образующих код, исправляющий r ошибок. $S_r(n)$ — число точек (наборов длины n) в шаре радиуса r.

Утверждение. $S_r(n) = 1 + C_n^1 + C_n^2 + \ldots + C_n^r$.

Definition

Функция $M_r(n)$ есть есть максимальное число слов длины n, образующих код, исправляющий r ошибок. $S_r(n)$ — число точек (наборов длины n) в шаре радиуса r.

Утверждение. $S_r(n) = 1 + C_n^1 + C_n^2 + \ldots + C_n^r$.

Доказательство.Точки шара радиуса r — это его центр, множество наборов, отличающихся от центра в одной координате — C_n^1, множество наборов, отличающихся в двух координатах — C_n^2 и т.д. Получаем исходное утверждение.

Theorem

$$\frac{2^n}{S_{2r}(n)} \le M_r(n) \le \frac{2^n}{S_r(n)}.$$

Доказательство. Рассмотрим произвольный код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \dots, \widetilde{\alpha}_m\}$, исправляющий r ошибок. Из утверждения 1 следует, что шары радиуса r не могут пересекаться, следовательно, число всех точек всех шаров не превосходит числа точек n-мерного куба и

$$m \cdot S_r(n) \le 2^n \Leftrightarrow m \le \frac{2^n}{S_r(n)} \Rightarrow M_r(n) \le \frac{2^n}{S_r(n)}.$$

Theorem

$$\frac{2^n}{S_{2r}(n)} \le M_r(n) \le \frac{2^n}{S_r(n)}.$$

Доказательство. Рассмотрим произвольный код $K = \{\widetilde{\alpha}_1, \widetilde{\alpha}_2, \ldots, \widetilde{\alpha}_m\}$, исправляющий r ошибок. Из утверждения 1 следует, что шары радиуса r не могут пересекаться, следовательно, число всех точек всех шаров не превосходит числа точек n-мерного куба и

$$m \cdot S_r(n) \le 2^n \Leftrightarrow m \le \frac{2^n}{S_r(n)} \Rightarrow M_r(n) \le \frac{2^n}{S_r(n)}.$$

Будем строить сам код K. Выберем произвольную точку $\widetilde{\alpha}_1$. В качестве точки $\widetilde{\alpha_2}$ мы не можем взять точки шара радиуса $2r$ с центром в точке $\widetilde{\alpha}_1$.

Пусть уже выбраны k наборов. Для выбора набора $\widetilde{\alpha}_{k+1}$ запрещено точек не больше, чем $k \cdot S_{2r}(n)$, то есть, если $k \cdot S_{2r}(n) < 2^n$, мы можем выбрать следующий набор. Рано или поздно встретится элемент m такой, что

$$m \cdot S_{2r}(n) \geq 2^n \Leftrightarrow m \geq \frac{2^n}{S_{2r}(n)} \Rightarrow M_r(n) \geq \frac{2^n}{S_{2r}(n)}.$$

$\square$

Рассмотрим коды, исправляющие одну ошибку типа замещения в словах длины n. Выберем натуральное k таким, что

$$2^{k-1} \leq n \leq 2^k - 1 \Leftrightarrow (k \leq log_2 n + 1)\&(k \geq log_2(n+1)) \\ \Leftrightarrow k = \lfloor log_2 n + 1 \rfloor = \lceil log_2(n+1) \rceil .$$

Рассмотрим коды, исправляющие одну ошибку типа замещения в словах длины n. Выберем натуральное k таким, что

$$2^{k-1} \leq n \leq 2^k - 1 \Leftrightarrow (k \leq log_2 n + 1) \& (k \geq log_2(n+1)) \\ \Leftrightarrow k = \lfloor log_2 n + 1 \rfloor = \lceil log_2(n+1) \rceil .$$

Разобьем номера всех разрядов исходного слова на k классов:
$D_i = \{m | m = (m_{k-1} m_{k-2} \dots m_0)_2, m_i = 1\}, 1 \leq m \leq n.$

Definition

Кодом Хэмминга порядка n называется множество наборов $\widetilde{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_n) \in E_2^k$, удовлетворяющих системе уравнений(суммы — по модулю 2):

$$\begin{cases} \sum_{j \in D_0} \alpha_j & = & 0 \\ \sum_{j \in D_1} \alpha_j & = & 0 \\ & \ldots & \\ \sum_{j \in D_{k-1}} \alpha_j & = & 0 \end{cases}$$

Theorem

Код Хэмминга порядка n содержит 2^{n-k} наборов, где $k = \lfloor \log_2 n \rfloor + 1$ и исправляет одну ошибку.

Theorem

Код Хэмминга порядка n содержит 2^{n-k} наборов, где $k = \lfloor \log_2 n \rfloor + 1$ и исправляет одну ошибку.

Доказательство. Рассмотрим систему уравнений из определения кода Хэмминга:

$$\begin{cases} \alpha_1 \oplus (\alpha_3 \oplus \ldots) & = & 0 \\ \alpha_2 \oplus (\ldots) & = & 0 \\ & \ldots & \\ \alpha_{2^{k-1}}(\ldots) & = & 0 \end{cases}$$

Заметим, что $\alpha_1, \alpha_2, \ldots, \alpha_{2^{k-1}}$, вынесенные нами за скобки, в скобках уже не встречаются. Значит, если значения всех остальных α_j мы зададим произвольно, то $\alpha_1, \alpha_2, \ldots, \alpha_{2^{k-1}}$ однозначно определятся из системы.

Пусть передавалось кодовое слово $\widetilde{\alpha} = (\alpha_1\alpha_2 \dots \alpha_n)$ и ошибка произошла в разряде $d = (\gamma_{k-1}\gamma_{k-2} \dots \gamma_1\gamma_0)_2$. Пусть на выходе получено слово $\widetilde{\beta} = (\beta_1\beta_2 \dots \beta_n)$, при этом $\beta_i = \alpha_i$ при $i \neq d, \beta_d = \alpha_d \oplus 1$.

Пусть передавалось кодовое слово $\widetilde{\alpha} = (\alpha_1\alpha_2\dots\alpha_n)$ и ошибка произошла в разряде $d = (\gamma_{k-1}\gamma_{k-2}\dots\gamma_1\gamma_0)_2$. Пусть на выходе получено слово $\widetilde{\beta} = (\beta_1\beta_2\dots\beta_n)$, при этом $\beta_i = \alpha_i$ при $i \neq d, \beta_d = \alpha_d \oplus 1$.

Обозначим $\delta_0 = \sum_{j\in D_0} \beta_j, \delta_1 = \sum_{j\in D_1} \beta_j, \dots, \delta_{k-1} = \sum_{j\in D_{k-1}} \beta_j$. Покажем, что $(\delta_{k-1}\delta_{k-2}\dots\delta_1\delta_0)_2 = d$.

Пусть передавалось кодовое слово $\widetilde{\alpha} = (\alpha_1\alpha_2\ldots\alpha_n)$ и ошибка произошла в разряде $d = (\gamma_{k-1}\gamma_{k-2}\ldots\gamma_1\gamma_0)_2$. Пусть на выходе получено слово $\widetilde{\beta} = (\beta_1\beta_2\ldots\beta_n)$, при этом $\beta_i = \alpha_i$ при $i \neq d, \beta_d = \alpha_d \oplus 1$.

Обозначим $\delta_0 = \sum_{j\in D_0}\beta_j, \delta_1 = \sum_{j\in D_1}\beta_j, \ldots, \delta_{k-1} = \sum_{j\in D_{k-1}}\beta_j$. Покажем, что $(\delta_{k-1}\delta_{k-2}\ldots\delta_1\delta_0)_2 = d$.

Пусть $\gamma_i = 0 \Rightarrow d \notin D_i$, тогда $\sum_{j\in D_i}\beta_j = \sum_{j\in D_i}\alpha_j$, значит, $\delta_i = 0, \delta_i = \gamma_i$.

Пусть передавалось кодовое слово $\widetilde{\alpha} = (\alpha_1\alpha_2\dots\alpha_n)$ и ошибка произошла в разряде $d = (\gamma_{k-1}\gamma_{k-2}\dots\gamma_1\gamma_0)_2$. Пусть на выходе получено слово $\widetilde{\beta} = (\beta_1\beta_2\dots\beta_n)$, при этом $\beta_i = \alpha_i$ при $i \neq d, \beta_d = \alpha_d \oplus 1$.

Обозначим $\delta_0 = \sum_{j\in D_0}\beta_j, \delta_1 = \sum_{j\in D_1}\beta_j, \dots, \delta_{k-1} = \sum_{j\in D_{k-1}}\beta_j$. Покажем, что $(\delta_{k-1}\delta_{k-2}\dots\delta_1\delta_0)_2 = d$.

Пусть $\gamma_i = 0 \Rightarrow d \notin D_i$, тогда $\sum_{j\in D_i}\beta_j = \sum_{j\in D_i}\alpha_j$, значит, $\delta_i = 0, \delta_i = \gamma_i$.
Пусть теперь $\gamma_i = 1, d \in D_i$. Тогда
$\sum_{j\in D_i}\beta_i = \sum_{j\in D_i}\alpha_i \oplus 1 = 1 \Rightarrow \delta_i = 1 \Rightarrow \delta_i = \gamma_i$.

$\square$

Theorem

$$\frac{2^n}{2n} \leq M_1(n) \leq \frac{2^n}{n+1}.$$

Доказательство. По одной из предыдущих теорем: $\frac{2^n}{S_{2r}(n)} \leq M_r(n) \leq \frac{2^n}{S_r(n)}$. Тогда правое неравенство явно следует из $S_1(n) = n + 1$.

Theorem

$$\frac{2^n}{2n} \le M_1(n) \le \frac{2^n}{n+1}.$$

Доказательство. По одной из предыдущих теорем: $\frac{2^n}{S_{2r}(n)} \le M_r(n) \le \frac{2^n}{S_r(n)}$. Тогда правое неравенство явно следует из $S_1(n) = n+1$.

В коде, исправляющем одну ошибку, различных слов ровно $2^{n-k} = m$. Поскольку $k = \lfloor \log_2 n \rfloor + 1$, тогда

$$k \le \log_2 n + 1 \Rightarrow m \ge 2^{n-\log_2 n - 1} = \frac{2^n}{2n} \Rightarrow M_1(n) \ge m \ge \frac{2^n}{2n}.$$

□