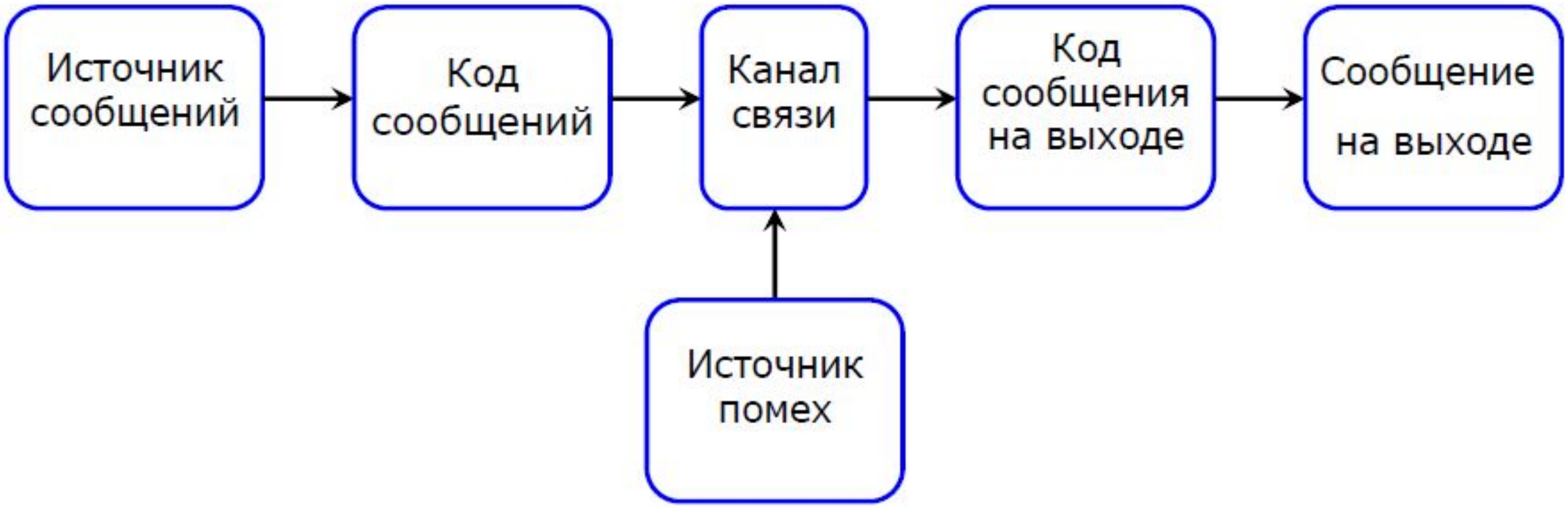
КОДИРОВАНИЕ



Основные понятия теории кодирования

Пусть $B=\{b_1,b_2,\ldots,b_m\}$ – алфавит. Конечная последовательность символов $\beta=b_{i_1}b_{i_2}\ldots b_{i_n}$ называется ***словом***, а число n – ***длиной слова*** (длину слова β будем обозначать через $|\beta|$). Через B^* обозначается множество всех слов в алфавите B. Слова из B^* будем называть также ***сообщениями***.

Под ***двоичным кодированием*** понимается инъективное отображение $f:L\to\{0,1\}^+$, где $\{0,1\}^+$ – множество всех непустых двоичных последовательностей. Далее под кодированием будем понимать двоичное кодирование.

Отображение f ставит в соответствие слову $\beta\in B^*$ слово $\alpha\in\{0,1\}^+$, α называется ***кодом сообщения*** β.

Требование инъективности отображения f означает, что различные сообщения должны кодироваться разными двоичными последовательностями. При выполнении этого требования обеспечивается однозначность декодирования сообщений. Такое кодирование будем называть ***взаимно-однозначным.***

Пример 3.1.1. Рассмотрим схему алфавитного кодирования

$$f_1 : \begin{cases} b_1 \to 0, \\ b_2 \to 01. \end{cases}$$

Схема f_1 задает взаимно-однозначное кодирование. Достаточно заметить, что перед каждым вхождением символа 1 стоит символ 0, поэтому каждое вхождение 1 вместе с предшествующим нулем кодирует букву b_2. Символ 0, за которым не следует 1, кодирует букву b_1. Например, последовательность $\beta = 001010001$ имеет единственную расшифровку $\alpha = b_1 b_2 b_2 b_1 b_1 b_2$.

Пример 3.1.2. Рассмотрим схему

$$f_2:\begin{cases} b_1 \to 0, \\ b_2 \to 01, \\ b_3 \to 001. \end{cases}$$

Кодирование, задаваемое схемой f_2, не является взаимно-однозначным. Последовательность $\beta = 001$ допускает две расшифровки $\alpha_1 = b_1 b_2$ и $\alpha_2 = b_3$.

Алфавитное кодирование задается схемой, в которой каждой букве алфавита ставится в соответствие двоичная последовательность символов:

$$f:\begin{cases} b_1 \to v_1, \\ b_2 \to v_2, \\ \quad \dots \\ b_m \to v_m, \end{cases} \qquad v_i \in \{0,1\}^* \text{ для всех } i = \overline{1, m}.$$

Коды v_i называются ***элементарными***, а их набор $V = (v_1, v_2, \dots v_m)$ − ***кодом***.

Через l_i будем обозначать длину элементарного кода буквы b_i. Набор длин элементарных кодов $L = (l_1, l_2, \dots l_m)$ называется ***спектром кода***.

Проблема распознавания взаимной однозначности кодирования

Пусть слово α имеет вид $\alpha_1\alpha_2$. Тогда α_1 называется ***префиксом*** слова α, а α_2 – ***суффиксом*** α. Если $0 < |\alpha_1| < |\alpha|$, то α_1 называется ***собственным префиксом*** α, если $0 < |\alpha_2| < |\alpha|$, то α_2 – ***собственный суффикс*** α (здесь через $|x|$ обозначается длина слова x).

Схема алфавитного кодирования f обладает ***свойством префикса***, если для любых i и j $(1 \le i, j \le m, i \ne j)$ слово v_i не является префиксом слова v_j.

Алфавитное кодирование, схема которого обладает свойством префикса, называется ***префиксным***.

Теорема 3.2.1. *Если схема обладает свойством префикса, то алфавитное кодирование является взаимно-однозначным.*

Таким образом, свойство префикса является достаточным условием взаимной однозначности.

Теорема 3.2.2. *Если алфавитное кодирование со схемой f обладает свойством взаимной однозначности, то длины элементарных кодов $l_i = |v_i| \; (i = \overline{1, m})$ удовлетворяют неравенству Мак-Миллана:* $\sum_{i=1}^{m} 2^{-l_i} \le 1.$

Неравенство Мак-Миллана является необходимым условием взаимной однозначности кода со схемой f, но не достаточным.

Для схемы f_2, рассмотренной в примере 3.1.2, имеем $l_1 = 1,\ l_2 = 2,\ l_3 = 3$, и неравенство Мак-Миллана выполняется: $2^{-1} + 2^{-2} + 2^{-3} = 7/8 < 1$. Однако задаваемое схемой f_2, кодирование не является взаимно-однозначным.

Пусть дана схема кодирования Σ и l_i — длина слова B_i, $L = l_1 + \ldots + l_r$.

Назовем *нетривиальным разложением* слова B_i его представление в виде $B_i = \beta' B_{j_1} \ldots B_{j_w} \beta''$, где $B_{j_1} \neq B_i$, β'' является началом какого-нибудь элементарного кода, а β' является концом какого-нибудь элементарного кода. Слова β' и β'' могут быть пустыми.

Пример.

Σ:

$A_1 = (1\ 0\ 0\ 1)$ $\quad l_1 = 4$

$A_2 = (0)$ $\quad l_2 = 1$

$A_3 = (0\ 1\ 0)$ $\quad l_3 = 3$

Рассмотрим слово $B = 0\ 1\ 0\ 0\ 1\ 0 = A_2\, A_1\, A_2 = A_3\, A_3$

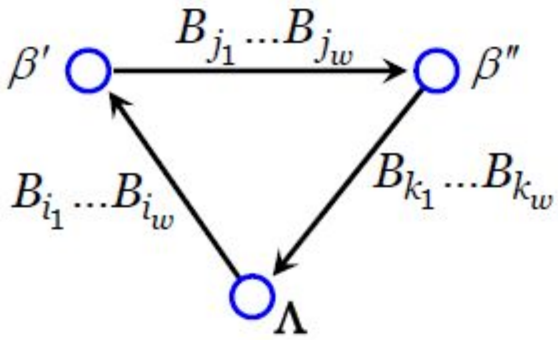
Нет однозначности декодирования!

Алгоритм проверки однозначности кодирования

Пусть дана схема кодирования Σ. Для каждого элементарного кода B_i рассмотрим все его нетривиальные разложения

$$B_i = \beta' B_{j_1} \dots B_{j_w} \beta''. \qquad (1)$$

Обозначим через $V = V(\Sigma)$ множество, содержащее пустое слово Λ и слова β, встречающиеся в разложениях вида (1) как в виде начал, так и в виде окончаний. Построим далее помеченный ориентированный граф $\Gamma = \Gamma(\Sigma)$ по следующим правилам. Множеством вершин графа Γ является $V = V(\Sigma)$. Проводим дугу из вершины $\beta' \in V$ в вершину $\beta'' \in V$, если и только если в некотором разложении вида (1) β' является началом, а β'' — концом. При этом дуга (β', β'') помечается словом $B_{j_1} \dots B_{j_w}$.



$$B_1 = \beta' B_{j_1} \dots B_{j_w} \beta''$$

$$B_2 = B_{i_1} \dots B_{i_w} \beta'$$

$$B_3 = \beta'' B_{k_1} \dots B_{k_w}$$

Теорема 2. Схема кодирования Σ не обладает свойством однозначности декодирования тогда и только тогда, когда граф $\Gamma(\Sigma)$ содержит контур, проходящий через вершину Λ.

Доказательство. Допустим, что Σ не обладает свойством однозначности декодирования. Тогда, как следует из доказательства теоремы 1, кратчайшее слово, имеющее две расшифровки в схеме Σ, имеет вид

$$B = B_{i_{1,1}} \dots B_{i_{1,k(1)}} \beta_1 B_{i_{2,1}} \dots B_{i_{2,k(2)}} \beta_2 \dots \beta_{s-1} B_{i_{s,1}} \dots B_{i_{s,k(s)}},$$

где все β_i различны и слова $B_{i_{1,1}} \dots B_{i_{1,k(1)}}$, β_1, $\beta_1 B_{i_{2,1}} \dots B_{i_{2,k(2)}} \beta_2, \dots$ $\dots, \beta_{s-1} B_{i_{s,1}} \dots B_{i_{s,k(s)}}$ являются элементарными кодами. Это значит, что в $\Gamma(\Sigma)$ есть контур, проходящий через вершины Λ, $\beta_1, \dots, \beta_{s-1}$.
Обратно, пусть в $\Gamma(\Sigma)$ существует контур, проходящий через вершины β_0, $\beta_1, \dots, \beta_{s-1}$, где $\beta_0 = \Lambda$ и дуга (β_j, β_{j+1}), $j = 0, 1, \dots, s-1$, $((s-1)+1=0)$, помечена словом $B_{i_{j+1,1}} \dots B_{i_{j+1,k(j+1)}}$. Тогда слово

$$B = B_{i_{1,1}} \dots B_{i_{1,k(1)}} \beta_1 B_{i_{2,1}} \dots B_{i_{2,k(2)}} \beta_2 \dots \beta_{s-1} B_{i_{s,1}} \dots B_{i_{s,k(s)}},$$

имеет две различные расшифровки. ■

Пример. Σ: $a_1 — b_1\, b_2$

$a_2 — b_1\, b_3\, b_2$

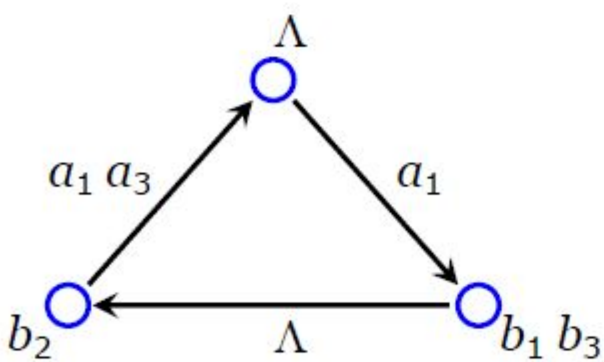
$a_3 — b_2\, b_3$

$a_4 — b_1\, b_2\, b_1\, b_3$

$a_5 — b_2\, b_1\, b_2\, b_2\, b_3$

Находим все префиксы, которые одновременно являются суффиксами и не являются кодовыми словами:

$\{\Lambda,\ b_2,\ b_1\, b_3\}$, то есть три вершины в графе



$a_1\, a_2\, a_1\, a_3 =$

$= b_1\, b_2\, b_1\, b_3\, b_2\, b_1\, b_2\, b_2\, b_3 =$

$= a_4\, a_5$

Пример.

$$\Sigma:\quad a_1 - b_1$$

$$a_2 - b_2\, b_1$$

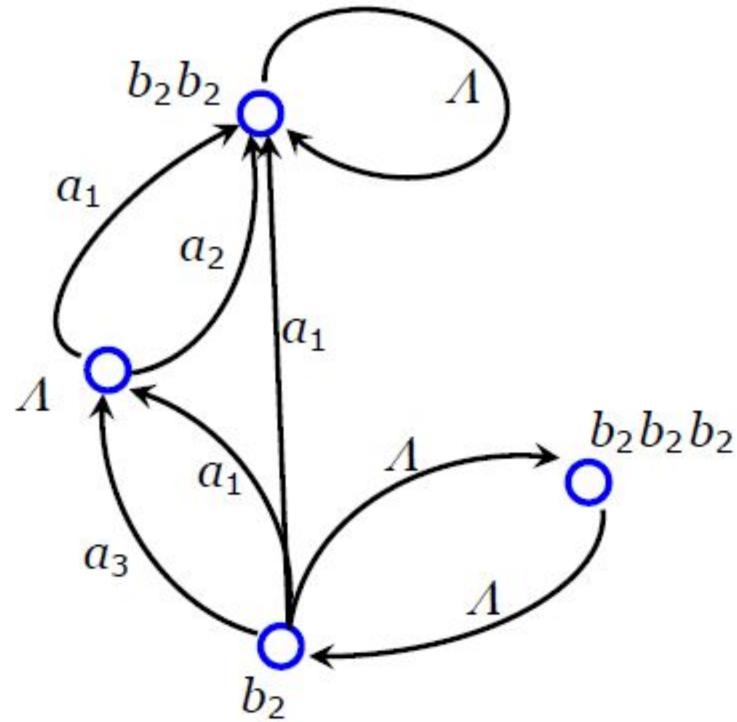
$$a_3 - b_1 b_2\, b_2$$

$$a_4 - b_2\, b_1\, b_2\, b_2$$

$$a_5 - b_2\, b_2\, b_2\, b_2$$

Находим все β: $\{\Lambda, b_2,\ b_2\, b_2,\ b_2\, b_2\, b_2\}$

Тогда получаем граф:



Нет цикла через вершину Λ.

Код однозначно декодируется.

Теорема Маркова о взаимной однозначности алфавитного кодирования:

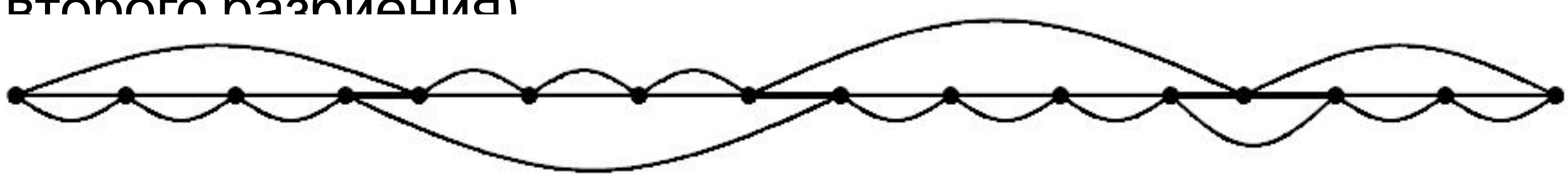
Пусть $\varphi: a_i \to B_i \ (i = 1, 2, \ldots, r)$ — некоторое кодирование. Пусть *W* — максимальное число кодовых слов, которые «помещаются» подряд внутри кодового слова. Пусть l_i — длина слова B_i и $L = \sum_{i=1}^{r} l_i$.

Тогда если кодирование φ не взаимно однозначно, то существуют два слова $a' \in A^*$, $a'' \in A^*$

$$\text{длина}(a') \leq \left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor, \ \text{длина}(a'') \leq \left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor \text{ и } \varphi(a') = \varphi(a'').$$

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 $\overline{b}=\overline{b_1}$. Очевидно, 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 остальные слова (то есть слова, являющиеся началами кодовых слов одного разбиения и концами слов второго разбиения).



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

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

Таким образом, все $\beta_1, \beta_2, \ldots, \beta_m$ разные. Тогда число слов второго класса не превосходит числа непустых начал элементарных кодов, то есть не пре

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

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

а в каждом из них укладывается слов не более чем W + 1. Отсюда число кодовых слов в любом разбиении не превосхо $\frac{L-r+2}{2}(W+1)$

а поскольку число целое. то не превосходит и целой части $\left\lfloor \frac{(W+1)(L-r+2)}{2} \right\rfloor$.

Теорема доказана

Теорема 2 (неравенство Макмиллана). Пусть задано кодирование $\varphi : a_i \to B_i$ $(i = 1, 2, \ldots, r)$ и пусть в кодирующем алфавите B — q букв и *длина* $(B_i) = l_i$ $(i = 1, 2, \ldots, r)$. 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

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

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

Лемма. $c_k \le q^k$ $(\forall k)$.

Доказательство. За c_k обозначено, очевидно, число наборов $(i_1, \ldots, i_n)$ $(1 \le i_j \le r)$, 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

$$\textit{длина}\left(B_{i_1} B_{i_2} \ldots B_{i_n}\right) = l_{i_1} + l_{i_2} + \ldots + l_{i_n} = k.$$

В силу того, что кодирование взаимно однозначно, различным наборам, соответствуют различные сообщения, а различных сообщений длины k в алфавите из q букв не более $q^k \Rightarrow c_k \le q^k$ $(\forall k)$.

Лемма доказана.

Согласно лемме $x^n = \sum_{k=1}^{nl_{\max}} \frac{c_k}{q^k} \le \sum_{k=1}^{nl_{\max}} 1 = nl_{\max} \Leftrightarrow x \le \sqrt[n]{nl_{\max}}, \forall n$. Устремляя n к бесконечности, получаем $x \le 1$. Теорема доказана.