Оптимальное кодирование.

$\exists\ \varphi: \mathcal{A} \to \mathcal{B}^*$ — алфавитное кодирование с длинами код. слов $\mathrm{len}(\varphi(a_i)) = l_i$, $1 \le i \le m$,

φ^* кодирует ДИС без памяти $(\mathcal{A}, \vec{p})$, $\vec{p} = (p_1, \dots, p_m)$

Хотим: $M(\mathrm{len}(\varphi^*(A^n))) \longrightarrow \min$. при $\forall n \ge 1$

т.к. все A_i одинаково распределены,

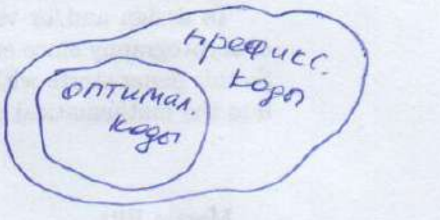
$$M\big(\mathrm{len}(\varphi^*(A^n))\big) = n \cdot M\big(\mathrm{len}(\varphi^*(A_1))\big) = n \cdot \sum_{i=1}^{m} p_i l_i$$

ОПР. Пусть $\varphi: \mathcal{A} \to \mathcal{B}^*$ – алф. кодир. из $\mathcal{A} = \{a_1, \dots, a_m\}$ в $\mathcal{B} = \{b_1, \dots, b_q\}$, $\underbrace{\vec{p} = (p_1, \dots, p_m)}_{\text{ДИС}}$

Средней длиной кода $\varphi(\mathcal{A})$ назов. $l^\varphi := \sum_{i=1}^{m} p_i l_i$

ОПР. Алфавитное кодир $\varphi: \mathcal{A} \to \mathcal{B}^*$ и код $\varphi(\mathcal{A})$ наз. оптимальным, если φ – однозначно декодируемо и средняя длина l^φ минимальна

Замеч. 1) Если оптимальное кодир существует $\Rightarrow$ $\Rightarrow$ существ. префиксное кодир с таким же набором код. слов и это префикс. кодир-е тоже оптимальное.



2) т.к. $l_i \ge 1\ \forall i \Rightarrow l^\varphi \ge 1$

Так, при $m \le q$ оптимальный код трив.

$\varphi(a_i) = b_i$; $1 \le i \le m$.

УТВ. Оптимальное кодир. φ – существует.

Дано: ДИС $(\mathcal{A}, \vec{p}) \Rightarrow l^\varphi = \sum_{i=1}^{m} p_i l_i$ – функция от арг. $(\mathcal{A}, \vec{p})$

$\Rightarrow \exists\ \inf_\varphi l^\varphi$ этой функции

Покажем, что $\exists\ \varphi: \mathcal{A} \to \mathcal{B}^*$, т.ч. $l^\varphi = \inf_\varphi l^\varphi$.

Д-во. $\exists\ L \in \mathbb{N}$ – наименьшее, т.ч. $m \le q^L$. Тогда $\exists\ \varphi$ – равномер: $\mathrm{len}\,\varphi(a_i) = L\ \forall 1 \le i \le m$.

$\Rightarrow \inf_\varphi l^\varphi \le L$

Не будем рассматрив. кодир-я φ, для кот. $l^\varphi > L$.

$l^\varphi = \sum_{i=1}^{m} p_i l_i \le L$ Удаляем эти слагаемые.

Если $p_i = 0 \Rightarrow l_i$ – не влияет на l^φ

Если $p_i > 0 \Rightarrow l_i \le \frac{L}{p_i} \le \frac{L}{p}$, где $p = \min\{p_1, \dots, p_m\}$.

Нам нужно найти кодирование, у кот. все $l_i \le \frac{L}{p}$ (при $p_i > 0$)

а l_i при $p_i = 0$ не важны

l^φ при таких ограничениях может прин. лишь конеч. число различных знач.

$\Rightarrow$ при некотором φ достигает min.

(Т) Если алф. кодир. φ — однозначно декодируемо, то

$$\ell^{\varphi} \geq \frac{H(\vec{p})}{\log_2 D},$$ причем равенство достигается тогда и т. тогда, когда все $\begin{cases} p_i = D^{-\ell_i} > 0 \\ p_i = 0 \end{cases}$

Д-во. $H(\vec{p}) - \ell^{\varphi}\log_2 D =$

$$= -\sum_{i=1}^{m} p_i \log_2 p_i - \log_2 D \sum_{i=1}^{m} p_i \ell_i = \left\{ \begin{array}{l} \text{раскр. скобки,} \\ \ell_i \text{ — под логарифм} \end{array} \right\}$$

$$= -\sum_{i=1}^{m} p_i \log_2 p_i + \sum_{i=1}^{m} p_i \log_2 D^{-\ell_i} = \sum_{i=1}^{m} p_i \log_2 \frac{D^{-\ell_i}}{p_i} \leq \left\{ \begin{array}{l} \log_2 p \leq \log_2 e\,(p-1) \\ \text{McMillan} \end{array} \right\} \leq$$

$$\leq \log_2 e \cdot \sum_{i=1}^{m} p_i \left(\frac{D^{-\ell_i}}{p_i} - 1 \right) = \log_2 e \left(\sum_{i=1}^{m} D^{-\ell_i} - \sum_{i=1}^{m} p_i \right) \leq 0$$

(Т) Если распред. $\vec{p}$ — невырожденное, то существует такое префикс. кодир. φ, для кот. справедливо:

$$\ell^{\varphi} < 1 + \frac{H(\vec{p})}{\log_2 D}$$

Следствие (2-х теорем). Средняя длина оптимального алф. кодирования φ удовл. нерав-вам:

$$\frac{H(\vec{p})}{\log_2 D} \leq \ell^{\varphi} \leq 1 + \frac{H(\vec{p})}{\log_2 D}$$

(Без д-ва)

Осн. алгоритмы для построения префиксных кодов: алг. Фано и Хаффмана

алг. Фано — строит код малой средней длины (не обяз. оптимальный)

Хаффмана — строит оптимальный код

Алгоритм Фано ($D=2$)

Вход: ДИС $\mathcal{A} = \{a_1, \ldots, a_m\}$, $\vec{p} = (p_1, \ldots, p_m)$

— алф. упорядочен так, чтобы $p_1 \geq p_2 \geq \ldots \geq p_m$ (!)

Кодовый алфавит: $\mathcal{B} = \{0, 1\}$ $D = \#\mathcal{B} = 2$.

Выход: Кодирование $\varphi: \mathcal{A} \to \mathcal{B}^*$.

Алгоритм: Шаг 1. Выберем $k \in [1, m]$:

$$\left| \sum_{i=1}^{k} p_i - \sum_{i=k+1}^{m} p_i \right| \longrightarrow \min$$

Шаг 2. Разбиваем A на подмнож.:

$$\underbrace{\boxed{a_0}\boxed{a_1}\boxed{\ldots}\boxed{a_k}}_{A_0}\underbrace{\boxed{\ldots}\boxed{a_m}}_{A_1}$$

$A = A_0 \cup A_1$

$A_0 := \{a_1, \ldots a_k\}$ $A_1 := \{a_{k+1}, \ldots, a_m\}$

Шаг s. Имеем множ. $A_{i_1, \ldots, i_s} \subseteq A$; $i_1, \ldots, i_s \in B$

если $A_{i_1 \ldots i_s} = \{a_j\} \Rightarrow$ присвоим симв. a_j код $i_1 \ldots i_s$:
$\varphi(a_j) := i_1 \ldots i_s$.

Иначе: $A_{i_1 \ldots i_s} = \{a_j \ldots a_l\}$

Выберем $k \in [j..l]$:

$$\left| \sum_{i=j}^{k} p_i - \sum_{i=k+1}^{l} p_i \right| \longrightarrow \min$$

Шаг s+1. Разбиваем $A_{i_1 \ldots i_s} := A_{i_1 \ldots i_s, 0} \cup A_{i_1 \ldots i_s, 1}$

Пример. $A = \{a, b, c, d, e\}$ $D = 2$

$\vec{p} = \{0{,}3;\ 0{,}2;\ 0{,}2;\ 0{,}2;\ 0{,}1\}$ $B = \{0, 1\}$

(a) 0,3	0	0		
(b) 0,2	0	1		
(c) 0,2	1	0		
(d) 0,2	1	1	0	
(e) 0,1	1	1	1	

$\varphi(a) = 00$

$\varphi(b) = 01$

$\varphi(c) = 10$

$\varphi(d) = 110$

$\varphi(e) = 111$

Средняя длина кода:

$$l^{\varphi} = (0{,}3 + 0{,}2 + 0{,}2) \cdot 2 + (0{,}2 + 0{,}1) \cdot 3 = 2{,}3 \text{ (бит)}.$$

Алгоритм Хаффмана ($D=2$)

Вход: ДИС $\mathcal{A}=\{a_1\dots a_m\}$ $\vec{P}=(p_1\dots p_m)$

- алф. упорядочен так, чтобы $p_1\geq p_2\geq\dots\geq p_m$ (!)

Кодовый алфавит: $\mathcal{B}=\{0;1\}$ $D=\#\mathcal{B}=2$.

Выход: Кодирование $\varphi: \mathcal{A}\to\mathcal{B}^*$.

Алгоритм: (I) Построим корневое размеч. дерево $G=(\cup\mathcal{A}^{(i)}, U)$

$\cup\mathcal{A}^{(i)}$ – множ. вершин, U – множ. ребер.

Положим $\mathcal{A}^{(0)}:=\mathcal{A}$ — листьями в дереве G (самый низкий уровень).

$\mathcal{A}^{(1)}$ – вершины предпоследнего уровня. Для их построения выполняем редукцию $\mathcal{A}^{(0)}$:

$\mathcal{A}^{(1)}:=\{a_1,\dots,a_{m-2},a_{m-1}\cup a_m\}$ – две вершины с min. вероятностями заменены на одну.

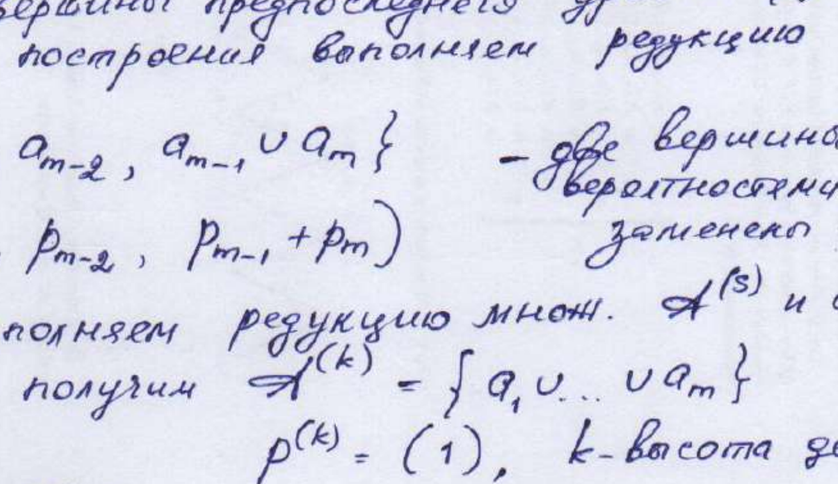
$\vec{P}^{(1)}=(p_1,\dots,p_{m-2},p_{m-1}+p_m)$

~~Вып~~ Выполняем редукцию множ. $\mathcal{A}^{(s)}$ и строим $\mathcal{A}^{(s+1)}$, пока не получим $\mathcal{A}^{(k)}=\{a_1\cup\dots\cup a_m\}$

$p^{(k)}=(1)$, k – высота дерева

(II) Код. слово $\varphi(a_i)$ – запис. последовательно метки ребер на пути от корня $a_1\cup\dots\cup a_m$ к листу a_i.

Пример.



$\varphi(a)=00$

$\varphi(b)=10$

$\varphi(c)=11$

$\varphi(d)=010$

$\varphi(e)=011$

Сред. длина $l^{\varphi}=2{,}3$ бит.