Заметки к курсу „Теория информации“

А.В. Смаль

2 августа 2021 г.

Аннотация

Курс посвящён изучению подходов к определению понятия „количество информации“. Последовательность изложения материала данного курса основана на классической статье Колмогорова „Три подхода к определению понятия количества информации“ (1965).

В курсе будет рассмотрено три подхода к определению „количества информации“: комбинаторный (информация по Хартли), вероятностный (энтропия Шеннона) и алгоритмический (Колмогоровская сложность). Кроме этого мы поговорим про различные применения аппарата теории информации в различных областях компьютерных наук: в криптографии, в коммуникационной сложности, в теории кодирования, в теории конечных автоматов, в теории сложности вычислений и некоторых других.

Содержание

1. Комбинаторный подход

1.1. Информация по Хартли

Пусть задано некоторое конечное множество A — *множество исходов*.

Определение 1.1 (1928)**.** Определим *количество информации в A* как $\chi(A) = \log_2 |A|$ (мы будем измерять количество информации в битах, поэтому все логарифмы будут по основанию 2, для измерения в байтах нужно выбрать основание 256).

Если про некоторый $x \in A$ стало известно, что $x \in B$, то теперь для идентификации x нам достаточно $\chi(A \cap B) = \log |A \cap B|$ битов, т.е. нам сообщили $\chi(A) - \chi(A \cap B)$ битов информации.

Пример 1.1. Предположим, что мы хотим узнать некоторое неизвестное упорядочение множества $\{a_1, a_2, \ldots, a_5\}$. Нам стало известно, что $a_1 > a_2$ или $a_3 > a_4$. Сколько битов информации мы узнали? Множество A состоит из 5! перестановок, множество B — из перестановок, которые удовлетворяют новому условию. Легко проверить, что $|B| = 90$. Итого мы узнали $\log 120 - \log 90 = \log(4/3)$ битов.

Пусть $A \subset \{0,1\}^* \times \{0,1\}^*$. Обозначим через $\pi_1(A)$ и $\pi_2(A)$ проекции множества A на первую и вторую координату соответственно, а $\chi_1(A) = \log |\pi_1(A)|$ и $\chi_2(A) = \log |\pi_2(A)|$ — количество информации в них по Хартли.

Теорема 1.1. $\chi(A) \leq \chi_1(A) + \chi_2(A)$.

Определение 1.2. Количество информации в второй координате $A \subset \{0,1\}^* \times \{0,1\}^*$ при известной первой

$$\chi_{2|1} = \log \left(\max_{a \in \pi_1(A)} \left| \{x \mid (a,x) \in A\} \right| \right).$$

Теорема 1.2. $\chi(A) \leq \chi_1(A) + \chi_{2|1}(A)$.

Теорема 1.3. *Для $A \subset \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^*$*

$$2 \cdot \chi(A) \leq \chi_{12}(A) + \chi_{13}(A) + \chi_{23}(A).$$

Следствие 1.1. *Квадрат объёма трёхмерного тела не превосходит произведение площадей его проекций на координатные плоскости.*

Утверждение 1.1. *Если $f : X \to Y$*

1. *является сюръекцией, то $\chi(Y) \leq \chi(X)$,*

2. *является инъекцией, то $\chi(X) \leq \chi(Y)$.*

1.2. Применение: игра в 10 вопросов

Сколько вопросов на ДА/НЕТ нужно задать, чтобы определить загаданное число от 1 до N, если (а) можно задавать вопросы адаптивно; (б) вопросы нужно написать на бумажке заранее.

Оценка $\lceil \log N \rceil$ достигается в обоих случаях, если задавать вопросы про биты двоичного представления загаданного числа.

Докажем нижнюю оценку. Пусть $A = [N]$. Множество $Q = \{(q_1, q_2, \ldots, q_k)\}$ — множество протоколов (ответы на вопросы). Можно рассматривать A и Q как проекции некоторого множества исходов игры S на разные координаты. Тогда верны следующие неравенства:

- $\chi_Q(S) = \chi(Q) \leq \chi_1(Q) + \chi_2(Q) + \cdots + \chi_k(Q) \leq k$,
- $\chi_A(S) = \chi(A) \leq \chi(S) \leq \chi_Q(S) + \chi_{A|Q}(S) \leq k + 0 = k$.

Таким образом получаем, что $\log N = \chi(A) \leq k$.

1.3. Цена информации

Пусть загадано некоторое целое число от 1 до n (где $n \geq 2$). Разрешается задавать любые вопросы с ответами ДА/НЕТ. При ответе ДА мы заплатим 1 рубль, а при ответе НЕТ — два рубля. Сколько необходимо и достаточно заплатить для отгадывания числа?

Верхняя оценка. Будем задавать вопросы так, чтобы отрицательные ответы приносили в два раза больше информации, чем положительные. Тогда за каждый бит информации мы заплатим некоторое константное количество рублей c. Пусть все вопросы будут вида „$x \in T$?“. Потребуем, чтобы

$$2 \cdot (\log |X| - \log |X \cap T|) = \log |X| - \log |X \cap \overline{T}|.$$

Пусть $|X \cap T| = \alpha |X|$, тогда $|X \cap \overline{T}| = (1 - \alpha)|X|$, таким образом получается уравнение

$$2 \log(1/\alpha) = \log(1/(1 - \alpha)),$$

эквивалентное квадратному уравнению

$$\alpha^2 = 1 - \alpha.$$

Из двух корней нас интересует тот, что меньше 1, т.е. $\alpha = (\sqrt{5} - 1)/2$. Следовательно при любом ответе мы заплатим $c = 1/(-\log \alpha) \approx 1.44$ рублей за бит, а в целом — $c \log n$ рублей.

В этой оценке мы полностью проигнорировали вопросы округления. Действительно, у нас никогда получится разделить множество из n элементов на два в отношении

$\alpha : (1-\alpha)$, т.к. α — иррациональное. Поэтому на каждом вопросе будет накапливаться некоторая ошибка округления. Давайте вместо вопросов принадлежности некоторому подмножеству T множества X будем задавать вопрос о принадлежности отрезку с вещественными координатами. Начнём с отрезок $S = [1, n]$ и будем каждый раз уменьшать его в $1/\alpha$ раз, т.е. первым вопросом спросим, принадлежит ли x отрезку $S' = [1, 1 + \alpha(n-1)]$. Длина отрезка S' в $1/\alpha$ раз меньше длины отрезка S. Продолжим действовать так же до тех пор, пока длина отрезка не станет меньше 1 — в этом случае x определено однозначно. После каждого вопроса длина отрезка уменьшается максимум в $1/(1-\alpha) = 1/\alpha^2$, поэтому длина последнего отрезка не меньше α^2. Таким образом длина отрезка сократится не более, чем в $(n-1)/\alpha^2$ раз. Поскольку мы каждый раз выбирали отрезки так, чтобы платить c рублей за уменьшение $\log|S|$ на 1, то в сумме заплатим не более

$$c \log((n-1)/\alpha^2) = c \log(n-1) - 2c \log \alpha = c \log(n-1) + 2.$$

При любом исходе мы заплатим целое число рублей, поэтому эту оценку можно уточнить до $\lfloor c \log(n-1) \rfloor + 2$.

Нижняя оценка. Применим рассуждение про злонамеренного противника (adversary argument). Пусть противник выбирает ответ ДА/НЕТ в зависимости от того, какое из двух значений $1/(\log|X| - \log|X \cap T|)$ и $2/(\log|X| - \log|X \cap \overline{T}|)$ больше. При любых X, T одно из этих значений не меньше $c = 1/(-\log \alpha)$. Таким образом мы заставляем алгоритм платить не менее c рублей за бит, а значит любой алгоритм в худшем случае заплатит $\lceil c \log n \rceil$ рублей.

1.4. Применение: упорядочивание камней по весу

1.4.1. Верхняя и нижняя оценки для произвольного N

Сколько сравнений нужно сделать для того, чтобы упорядочить N камней по весу?

Нижняя оценка. Потребуется $\lceil \chi(S_N) \rceil = \lceil \log n! \rceil$ сравнений.

Верхняя оценка. Будем сортировать вставкой с бинарным поиском места вставки. Количество сравнений:

$$\lceil \log 2 \rceil + \lceil \log 3 \rceil + \cdots + \lceil \log n \rceil \leq \log n! + n - 1 = n \log n + O(n).$$

1.4.2. Точные оценки для маленьких N

Упражнение 1.1. Сколько нужно взвешиваний, чтобы упорядочить N камней по весу? Найдите точный ответ на этот вопрос для $N = 2, 3, 4, 5$. Указание: воспользуйтесь жадной стратегией, при которой каждое взвешивание приносит максимум информации.

1.5. Применение: поиск фальшивой монетки

Предлагается решить следующие несколько задач.

- Есть 20 с виду одинаковых монет, одна из которых фальшивая легче остальных. Как найти фальшивую монету используя чашечные весы? За какое минимально количество взвешиваний это можно сделать?

  Решение. Каждое взвешивание даёт не более $\log 3$ битов информации, следовательно число взвешиваний не меньше $\lceil \log 20/\log 3 \rceil = \lceil \log_3 20 \rceil = 3$.

- Есть 13 с виду одинаковых монет, одна фальшивая (с неизвестным относительным весом). Можно ли за три взвешивания найти фальшивую монету и узнать её относительный вес?

  Решение. Нужно рассмотреть два варианта первого шага:

  – если взвешиваем по 4, то при равенстве нельзя из 5 за два взвешивания найти фальшивую (остаётся 10 исходов),

  – если взвешиваем по 5, то при неравенстве остаётся 10 возможных исходов.

- Есть 15 монет с виду одинаковых монет, одна фальшивая. Можно ли за три взвешивания найти фальшивую, если не требуется узнавать её относительный вес?

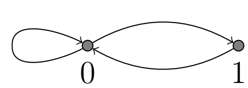
  Решение. Рассмотрим, сколько может быть различных исходов. Только в одном случае мы можем узнать, что монетка фальшивая и при этом не знать её относительный вес — если она не побывала на весах. Это значит, что в результате всех трёх взвешиваниях на весах будет равновесие. Поэтому такая монета может быть только одна, а в остальных случаях мы узнаем относительный вес. В таком случае различных исходов $2 \cdot 14 + 1 = 29$, что превышает 3^3 — число результатов трёх взвешиваний.

- Есть 14 монет с виду одинаковых монет, одна фальшивая. Можно ли за три взвешивания найти фальшивую, если не требуется узнавать её относительный вес?

  Вместо решения. Аналогичные рассуждения не позволяют доказать невозможность, т.к. $2 \cdot 13 + 1 = 3^3$. Тем не менее, решить эту задачу за три взвешивания нельзя. Для того, чтобы это доказать, нам недостаточно определения информации по Хартли.

Упражнение 1.2. За три взвешивания найти одну фальшивую монету из 12, если её относительный вес неизвестен. Указание: воспользуйтесь „жадной“ стратегией, при которой каждое взвешивание приносит максимум информации.

Упражнение 1.3. Пусть L_n — множество путей длины n в графе.



Чему равен предел $\lim_{n\to\infty} \frac{\chi(L_n)}{n}$?

Упражнение 1.4. Пусть загадано число от 1 до N. Можно задавать любые вопросы на ДА/НЕТ. Сколько вопросов потребуется, если на один ответ можно дать неверный ответ, а вопросы (а) можно задавать адаптивно; (б) нужно написать заранее?

1.6. Логика знаний

В этом разделе мы будем называть множество исходов A множеством *миров*. Пусть f — это некоторая функция из A в некоторое множество I (будем воспринимать это как информация о мире). Нам не важно какие значения принимает f, нам будут важны лишь классы эквивалентности, на которые f разбивает A: каждый класс эквивалентности будет состоять из миров A с одинаковым значением f.

Пример 1.2. Пусть $A = \{1, 2, 3, 4, 5\}$, а $f(x) = x \bmod 3$. Тогда f разбивает A на три класса эквивалентности $\{1, 4\}$, $\{2, 5\}$ и $\{3\}$.

Пусть $B \subset A$ — это некоторое *утверждение* о мирах. B *истинно* в мире x, если $x \in B$. В противном случае B *ложно* в x. В мире x мы *знаем, что B истинно*, если $y \in B$ для всех $y \sim x$.

Пример 1.3. Пусть $A = \{1, 2, 3, 4, 5\}$, а $f(x) = x \bmod 3$. Тогда в мирах 1, 4 и 3 мы знаем, что мир меньше 5. А в мирах 2 и 5 — не знаем.

Замечание 1.1. „Не знаем“ мы будем понимать в смысле „не верно, что знаем“.

К утверждениям о мирах можно применять обычные логические связки: «И» (пересечение), «ИЛИ» (объединение), «НЕ» (дополнение).

Утверждение 1.2. *Если в мире x мы знаем B, то в мире x мы знаем, что мы знаем B. Аналогично, если в мире x мы не знаем B, то в мире x мы знаем, что не знаем B.*

Пусть теперь у нас есть k человек со своими знаниями о мире. Они определяют k отношений эквивалентности $\sim_1, \sim_2, \dots, \sim_k$ и, соответственно, k разбиений на классы эквивалентности.

Пример 1.4. Пусть множество миров $A = \{1, 2, 3, 4, 5\}$ и есть два человека, Алиса и Боб. Алиса знает значения $f_A(x) = x \bmod 3$, а Боб знает $f_B(x) = x \bmod 2$. Тогда классы эквивалентности Алисы: $\{1, 4\}$, $\{2, 5\}$ и $\{3\}$, а классы эквивалентности Боба: $\{1, 3, 5\}$ и $\{2, 4\}$. В мире 1 Алиса знает, что мир меньше 5, а Боб не знает. В мире 4 они оба это знают. В мире 1 Алиса не знает, что Боб не знает, что мир меньше 5 (действительно, в мире 4, который с точки зрения Алисы эквивалентен 1, Боб это знает).

Задача 1.1. Пусть имеется некоторая карточка, про которую известно, что на одной её стороне написано целое неотрицательное число n, а на другой — целое число $n + 1$. Алиса и Боб сидят друг напротив друга смотрят на эту карточку с разных сторон и между ними происходит следующий разговор.

А: Я не знаю числа на стороне Боба.

Б: Я не знаю числа на стороне Алисы.

Это повторяется 10 раз и после этого Алиса говорит, что знает число на стороне Боба. Какие числа могли быть написаны на карточке?

Задача 1.2. В магазине имеется три красные шляпы и две белые. Три джентльмена поочереди покупают случайную шляпу и не глядя надевают её на себя (т.е. джентльмен не знает цвета шляпы, которую он купил). После этого джентльмены смотрят друг на друга и происходит следующий разговор.

1: Я не знаю цвета своей шляпы.

2: Я не знаю цвета своей шляпы.

3: Теперь я знаю цвет своей шляпы.

Какого цвета шляпа на третьем джентльмене?

Задача 1.3. У короля есть 9 бутылок вина. В одной бутылке вино отравленное. У короля есть две служанки. Каждый день любая служанка может намешать в свой стакан коктейль из разных бутылок и выпить, но служанке даётся только одна попытка в день, в фиксированное время, ровно в полдень (так что если обе служанки пробуют, одна из них не может учитывать результат второй в тот же день). Любое количество отравленного вина в стакане быстро убивает.

Как обнаружить, какая из бутылок отравлена, за два дня?

2. Вероятностный подход

2.1. Энтропия Ше́ннона

Энтропия Шеннона определяет количество информации $H(\alpha)$ в распределении вероятностей для некоторой случайной величины α. Пусть α принимает значения из множества $\{a_1, a_2, \dots, a_k\}$ с вероятностями $\{p_1, p_2, \dots, p_k\}$, $p_i \geq 0$, $\sum_i p_i = 1$.

Нам бы хотелось, чтобы это определение согласовывалось с определением Хартли, т.е. имеют место следующие „граничные условия“:

- если $p_1 = \dots = p_k$, то $H(\alpha) = \log k$,
- если $p_1 = 1$, $p_2 = \dots = p_k = 0$, то $H(\alpha) = 0$.

Будем искать $H(\alpha)$ в виде математического ожидания информации, которую мы получаем от каждого исхода.

$$H(\alpha) = \sum_i p_i \cdot (\text{информация в } a_i).$$

Как оценить, сколько информации в исходе a_i? Пусть U — всё пространство элементарных исходов, все исходы которого равновероятны. Тогда событию $\alpha = a_i$ соответствует множеству элементарных исходов меры p_i. Соответственно, если случилось событие

$\alpha = a_i$, то размер множества согласованных с этим событием исходов уменьшается с $|U|$ до $p_i|U|$, т.е. событие $\alpha = a_i$ сообщает нам $\log|U| - \log(p_i|U|) = \log\frac{1}{p_i}$ битов информации. Пусть теперь элементарные исходы не равновероятны. В этом случае событие $\alpha = a_i$ сообщает нам информацию, которая уменьшает *меру* множества возможных исходов в $1/p_i$ раз, т.е. опять получаем $\log 1 - \log p_i = \log\frac{1}{p_i}$. Это приводит нас к следующему определению.

Определение 2.1 (1948)**.** Энтропия Шеннона случайной величины α

$$H(\alpha) = \sum_{i=1}^{k} p_i \cdot \log\frac{1}{p_i}.$$

(По непрерывности доопределим $0 \cdot \log\frac{1}{0} = 0$.)

Можно вывести это соотношение из определения информации по Хартли другим способом. Пусть W_n — это множество всех слов длины n состоящих из букв $\{a_1, a_2, \ldots, a_k\}$, где каждая буква a_i встречается ровно $n_i = p_i \cdot n$ раз (будем считать, что вероятности p_i рациональны, и что множество W_n определено только тогда, когда все n_i целые). Информация по Хартли в W_n

$$\chi(W_n) = \log|W_n| = \log\frac{n!}{n_1!n_2!\cdots n_k!}.$$

Это выражение можно оценить при помощи формулы Стирлинга.

$$\begin{aligned}
\chi(W_n) &= \log\frac{\mathrm{poly}(n)\cdot(n/e)^n}{\mathrm{poly}(n)\cdot(n_1/e)^{n_1}\cdot(n_1/e)^{n_2}\cdots(n_k/e)^{n_k}} = \\
&= \log\left(\left(\frac{n}{n_1}\right)^{n_1}\cdot\left(\frac{n}{n_2}\right)^{n_2}\cdots\left(\frac{n}{n_k}\right)^{n_k}\right) + O(\log n) = \\
&= \log\left(\left(\frac{1}{p_1}\right)^{p_1\cdot n}\cdot\left(\frac{1}{p_2}\right)^{p_2\cdot n}\cdots\left(\frac{1}{p_k}\right)^{p_k\cdot n}\right) + O(\log n) = \\
&= n\cdot\sum_{i=1}^{k} p_i\cdot\log\frac{1}{p_i} + O(\log n).
\end{aligned}$$

В среднем на один символ приходится $\chi(W_n)/n$ битов информации. В пределе получаем

$$\lim_{n\to\infty}\frac{\chi(W_n)}{n} = \sum_{i=1}^{k} p_i\cdot\log\frac{1}{p_i} = H(\alpha)$$

(предел нужно брать по бесконечной подпоследовательности натуральных чисел n таких, для которых все $\{n_i\}$ — целые).

Лемма 2.1. *Для энтропии Шеннона выполняются следующие соотношения.*

- $H(\alpha) \geq 0$, *причём* $H(\alpha) = 0 \iff$ *распределение* α *вырождено.*

- $H(\alpha) \leq \log k$, *причём* $H(\alpha) = \log k \iff$ *величина* α *распределена равномерно.*

Для доказательства нам потребуется следующая теорема.

Теорема 2.1 (Неравенство Йенсена)**.** *Пусть функция* $f(x)$ *является вогнутой на некотором промежутке* $\mathcal{X}$ *и числа* $q_1, q_2, \ldots, q_n > 0$ *таковы, что* $q_1 + \cdots + q_n = 1$. *Тогда для любых* $x_1, x_2, \ldots, x_n$ *из промежутка* $\mathcal{X}$ *выполняется неравенство:*

$$\sum_{i=1}^{n} q_i f(x_i) \leq f\left(\sum_{i=1}^{n} q_i x_i\right).$$

Доказательство леммы 2.1. Первое свойство следует напрямую из определения: каждый член суммы $H(\alpha)$ неотрицателен и равен нулю только в случае, если $p_i = 0$ или $p_i = 1$.

Для доказательства второго неравенства перенесём всё в левую часть и применим неравенство Йенсена:

$$H(\alpha) - \log k = \sum_{i=1}^{k} p_k \cdot \log\frac{1}{p_i} - \sum_{i=1}^{k} p_i \cdot \log k = \sum_{i=1}^{k} p_k \cdot \log\frac{1}{p_i k} \leq \log\left(\sum_{i=1}^{k} p_i \frac{1}{p_i k}\right) = \log 1 = 0.$$

□

Энтропию совместного распределения пары случайных величин α и β будем обозначать $H(\alpha, \beta)$.

Лемма 2.2. *Выполняются следующие свойства:*

- $H(\alpha, \beta) \leq H(\alpha) + H(\beta)$, *причём равенство достигается тогда и только тогда, когда случайные величины независимы;*

- $H(\alpha) \leq H(\alpha, \beta)$, *причём равенство достигается тогда и только тогда, когда* β *полностью определяется значением* α, *т.е.* $\beta = f(\alpha)$.

Доказательство. Введём обозначения для вероятностей событий совместного распределения вероятностей (α, β). Пусть пара (a_i, b_j) имеет вероятность $p_{i,j}$, событие $[\alpha = a_i]$ имеет вероятность $p_{i,*} = p_{i,1} + \cdots + p_{i,n}$, а событие $[\beta = b_j]$ — вероятность $p_{*,j} = p_{1,j} + \cdots + p_{k,j}$. В этих обозначениях неравенство $H(\alpha, \beta) \leq H(\alpha) + H(\beta)$ переписывается как

$$\sum_{i,j} p_{i,j} \cdot \log\frac{1}{p_{i,j}} \leq \sum_{i}\sum_{j} p_{i,j} \cdot \log\frac{1}{p_{i,*}} + \sum_{j}\sum_{i} p_{i,j} \cdot \log\frac{1}{p_{*,j}}.$$

Перенесём всё в левую часть и применим неравенство Йенсена.

$$\sum_{i,j} p_{i,j} \cdot \log \frac{p_{i,*} \cdot p_{*,j}}{p_{i,j}} \le \log \left(\sum_{i,j} p_{i,j} \cdot \frac{p_{i,*} \cdot p_{*,j}}{p_{i,j}} \right) = \log \left(\sum_{i,j} p_{i,*} \cdot p_{*,j} \right) = \\ = \log \left(\underbrace{\left(\sum_i p_{i,*} \right)}_{1} \cdot \underbrace{\left(\sum_j p_{*,j} \right)}_{1} \right) = 0.$$

Равенство в неравенстве Йенсена для $f(x) = \log(x)$ достигается только, если все точки равны, т.е. для любых i, j $\frac{p_{i,*} p_{*,j}}{p_{i,j}} = c$ для некоторой константы c. Несложно заметить, что $c = 1$, т.к. выполняется следующее равенство $\sum_{i,j} p_{i,*} p_{*,j} = c \sum_{i,j} p_{i,j}$ в котором обе суммы равны 1. Таким образом в случае равенства α и β независимы.

Доказательство второго свойства мы получим как следствие из свойств условной энтропии. □

Определение 2.2. Энтропия α при условии $\beta = b_j$

$$H(\alpha \mid \beta = b_j) = \sum_i \Pr[\alpha = a_i \mid \beta = b_j] \cdot \log \frac{1}{\Pr[\alpha = a_i \mid \beta = b_j]}.$$

Определение 2.3. *Условная (относительная) энтропия* α относительно β

$$H(\alpha \mid \beta) = \sum_j \Pr[\beta = b_j] \cdot H(\alpha \mid \beta = b_j).$$

Другими словами

$$H(\alpha \mid \beta) = \mathop{\mathbb{E}}_{b_j \leftarrow \beta} [H(\alpha \mid \beta = b_j)].$$

Если подставить определение 2.2, то можно получить выражение для условной энтропии через отдельные вероятности событий.

$$H(\alpha \mid \beta) = \sum_j \Pr[\beta = b_j] \cdot \sum_i \Pr[\alpha = a_i \mid \beta = b_j] \cdot \log \frac{1}{\Pr[\alpha = a_i \mid \beta = b_j]} = \sum_{i,j} p_{i,j} \cdot \log \frac{p_{*,j}}{p_{i,j}}.$$

Лемма 2.3. *Условная энтропия обладает следующими свойствами.*

- $H(\alpha \mid \beta) \ge 0$.
- $H(\alpha \mid \beta) = 0 \iff \alpha$ *однозначно определяется по* β.
- $H(\alpha, \beta) = H(\beta) + H(\alpha \mid \beta) = H(\alpha) + H(\beta \mid \alpha)$.

Доказательство. Первое свойство выполняется, т.к. условная энтропия это матожидание неотрицательной случайной величины. Второе свойство объясняется тем, что для любого j распределение $\langle \alpha \mid \beta = b_j \rangle$ имеет нулевую энтропию, т.е. распределение вырождено и каждому b_j соответствует ровно один a_i. Третье свойство следует из следующего равенства.

$$\sum_{i,j} p_{i,j} \cdot \log \frac{1}{p_{i,j}} = \sum_{i,j} p_{i,j} \cdot \log \frac{1}{p_{*,j}} + \sum_{i,j} p_{i,j} \cdot \log \frac{p_{*,j}}{p_{i,j}}.$$

(Нужна аккуратность, если есть строки, которые состоят из одних нулей, т.е $p_{*,j} = 0$ — такие строки не нужно включать в эти суммы.) □

Следствие 2.1. *$H(\alpha, \beta) \geq H(\alpha)$, причём равенство достигается тогда и только тогда, когда $\beta = f(\alpha)$.*

Доказательство. $H(\alpha, \beta) - H(\alpha) = H(\beta \mid \alpha) \geq 0$. По второму свойству условной энтропии равенство достигается тогда и только тогда, когда $\beta = f(\alpha)$. □

2.2. Взаимная информация

Определение 2.4. *Информация в α о величине β* определяется следующим соотношением:

$$I(\alpha : \beta) = H(\beta) - H(\beta \mid \alpha).$$

Эту величину так же называют *взаимной информацией случайных величин α и β.*

Лемма 2.4. *Для взаимной информации выполняются следующие соотношения.*

1. $I(\alpha : \beta) \leq H(\alpha)$.
2. $I(\alpha : \beta) \leq H(\beta)$.
3. $I(\alpha : \alpha) = H(\alpha)$.
4. $I(\alpha : \beta) = I(\beta : \alpha)$.
5. $I(\alpha : \beta) = H(\alpha) + H(\beta) - H(\alpha, \beta)$.

Определение 2.5. Пусть α, β, γ — случайные величины. Определим *взаимную информацию в α о β при условии γ.*

1. $I(\alpha : \beta \mid \gamma) = H(\beta \mid \gamma) - H(\beta \mid \alpha, \gamma)$.
2. $I(\alpha : \beta \mid \gamma) = \sum_\ell I(\alpha : \beta \mid \gamma = c_\ell) \cdot \Pr[\gamma = c_\ell]$.
3. $I(\alpha : \beta \mid \gamma) = H(\alpha \mid \gamma) + H(\beta \mid \gamma) - H(\alpha, \beta \mid \gamma)$.
4. $I(\alpha : \beta \mid \gamma) = H(\alpha, \gamma) + H(\beta, \gamma) - H(\alpha, \beta, \gamma) - H(\gamma)$.

Лемма 2.5. *Все определения условной взаимной информации эквивалентны.*

Доказательство. (3) $\iff$ (4).

$(3) = H(\alpha \mid \gamma) + H(\beta \mid \gamma) - H(\alpha, \beta \mid \gamma) = H(\alpha, \gamma) - H(\gamma) + H(\beta, \gamma) - H(\gamma) - H(\alpha, \beta, \gamma) + H(\gamma).$

□

Утверждение 2.1 (chain rule for mutual information)**.** *Имеют место следующие соотношения:*

1. $I((\alpha, \beta) : \gamma) = I(\alpha : \gamma) + I(\beta : \gamma \mid \alpha)$.

2. $I((\alpha, \beta) : \gamma \mid \delta) = I(\alpha : \gamma \mid \delta) + I(\beta : \gamma \mid \alpha, \delta)$.

2.3. Применение: опять о поиске фальшивой монетки

Теперь у нас достаточно знаний, чтобы доказать, что за три взвешивания нельзя найти одну фальшивую монету из 14, даже если не нужно определять её относительный вес.

Доказательство. Предположим, что существует способ найти фальшивую монету за три взвешивания. Тогда протокол взвешивания можно представить в виде полного троичного дерева, где каждый лист помечен номером монетки, которая оказалась фальшивой (у нас как раз ровно $3^3 = 27$ исходов).

Давайте введём следующее распределение вероятностей α. Пусть монета, номер которой находится в листе, соответствующем трём равенствам (такой лист только один), имеет номер i. В нашем распределении вероятностей монета с номером i будет фальшивой с вероятностью $1/27$. Оставшиеся монеты оказываются фальшивыми с вероятностями $2/27$, причём с вероятностью $1/27$ монета оказывается легче, чем настоящая, и с такой же вероятностью она оказывается тяжелее настоящей.

$$H(\alpha) = \log 27 = 3 \log 3.$$

Пусть случайные величины β_1, β_2, β_3 соответствуют результатам первого, второго и третьего взвешивания соответственно. Значение α однозначно определяется после трёх взвешиваний: $H(\alpha \mid \beta_1, \beta_2, \beta_3) = 0$, а следовательно

$$H(\alpha) \le H(\beta_1, \beta_2, \beta_3) \le H(\beta_1) + H(\beta_2) + H(\beta_3) \le 3 \log 3.$$

Таким образом каждое взвешивание должно иметь энтропию ровно $\log 3$. Рассмотрим первое взвешивание. Пусть на чашах весов лежит по k монет. Вероятность каждого исхода взвешивания ($<$, $>$, $=$) относительно распределения α должна быть ровно $1/3$.

$$\Pr[<] = \frac{k}{27} + \frac{k}{27} = \frac{1}{3}.$$

Таким образом $2k = 9$, а значит нет такого целого k. □

Упражнение 2.1. Пусть у нас есть N камней разного веса и чашечные весы. Сколько нужно взвешиваний, чтобы найти

1. самый тяжёлый и второй по тяжести камень,

2. самый тяжёлый и самый лёгкий камни.

3. Кодирование

3.1. Однозначно декодируемые коды

Определение 3.1. Будем называть *кодом* функцию $C : \{a_1, a_2, \ldots, a_n\} \to \{0,1\}^*$, сопоставляющую буквам некоторого алфавита *кодовые слова*. Если любое сообщение, которое получено применением кода C, декодируется однозначно (т.е. только единственным образом разрезается на образы C), то такой код называется *однозначно декодируемым.*

Определение 3.2. Код называется *префиксным (беспрефиксным, prefix-free)*, если никакое кодовое слово не является префиксом другого кодового слова.

Теорема 3.1 (Неравенство Крафта-Макмилана)**.** *Для любого однозначно декодируемого кода со множеством кодовых слов $\{c_1, c_2, \ldots, c_n\}$ выполняется следующее неравенство:*

$$\sum_{i=1}^{n} 2^{-|c_i|} \leq 1.$$

Лемма 3.1. *Для префиксных кодов верно неравенство Крафта-Макмилана.*

Доказательство. Рассмотрим дерево префиксного кода и посчитаем суммарную меру поддеревьев, которые соответствуют кодовым словам. □

Утверждение 3.1. *Для префиксных кодов верно и обратное: если есть набор целых чисел $\{\ell_1, \ell_2, \ldots, \ell_n\}$, удовлетворяющие неравенству Крафта-Макмилана*

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1,$$

то существует префиксный код с кодовыми словами $\{c_1, c_2, \ldots, c_n\}$, где $|c_i| = \ell_i$.

Доказательство. Отсортируем ℓ_i по возрастанию и будем развешивать их в бесконечном двоичном дереве, выбирая каждый раз самый левый свободный узел соответствующей меры. Можно заметить, что мы всегда сможем найти такой узел. □

Следствие 3.1. *Для любого однозначно декодируемого кода существует префиксный код с теми же длинами кодовых слов.*

Доказательства теоремы 3.1. Сопоставим кодовым словам $\{c_i\}$ мономы $\{p_i\}$ от переменных x и y таким образом, что каждый '0' в кодовом слове соответствует x, а каждая '1' — y:

$$c_i = 0110101 \implies p_i(x,y) = xyyxyxy.$$

Рассмотрим следующее выражение для некоторого L.

$$\left(\sum_{i=1}^{n} p_i(x,y)\right)^L = \sum_{\ell=L}^{\max|c_i|\cdot L} M_\ell(x,y),$$

где M_ℓ обозначает сумму всех получившихся мономов степени ℓ. Заметим, что в каждом M_ℓ не более 2^ℓ мономов: в противном случае код не был бы однозначно декодируемым — каждый моном (без учёта коммутативности и ассоциативности) мог получиться не более одного раза.

Теперь рассмотрим значение этого выражения при $x = y = \frac{1}{2}$.

$$\left(\sum_{i=1}^{n} p_i\left(\tfrac{1}{2}, \tfrac{1}{2}\right)\right)^L = \sum_{\ell=L}^{\max|c_i|\cdot L} M_\ell\left(\tfrac{1}{2}, \tfrac{1}{2}\right) \leq \sum_{\ell=L}^{\max|c_i|\cdot L} (2^{-\ell} \cdot 2^\ell) \leq L \cdot \max|c_i| = O(L). \qquad (1)$$

Предположим теперь, что неравенство Крафта-Макмилана не выполняется, т.е.

$$q = \sum_{i=1}^{n} p_i(1/2, 1/2) = \sum_{i=1}^{n} 2^{-|c_i|} > 1.$$

Сравнивая это с (1) получаем противоречие: $q^L = O(L)$ (левая часть растёт экспоненциально, а правая — линейно). $\square$

Пусть для каждого символа алфавита задана вероятность p_i. Нас будут интересовать самые короткие в среднем коды, т.е. такие, что

$$\sum_{i=1}^{n} p_i \cdot |c_i| \to \min.$$

Теорема 3.2 (Шеннон)**.** *Для любого однозначно декодируемого кода выполняется*

$$\sum_{i=1}^{n} p_i \cdot |c_i| \geq \sum_{i=1}^{n} p_i \cdot \log\frac{1}{p_i}.$$

Доказательство. Перенесём всё в правую часть и применим неравенство Йенсена:

$$\sum_{i=1}^{n} p_i \cdot \log\frac{2^{-|c_i|}}{p_i} \leq \log\sum_{i=1}^{n}\left(p_i\frac{2^{-|c_i|}}{p_i}\right) = \log\sum_{i=1}^{n} 2^{-|c_i|} \leq \log 1 = 0.$$

$\square$

Теорема 3.3 (Шеннон)**.** *Для любого распределения вероятностей* $\{p_1, p_2, \ldots, p_n\}$ *существует однозначно декодируемый/префиксный код* $\{c_1, c_2, \ldots, c_n\}$*, такой что*

$$\sum_{i=1}^{n} p_i \cdot |c_i| \leq \sum_{i=1}^{n} p_i \cdot \log\frac{1}{p_i} + 1.$$

Замечание 3.1. От '+1' в правой части никак не избавиться: например, если у нас только два символа в алфавите, то $\sum p_i \cdot |c_i| = 1$, в то время как $\sum p_i \log\frac{1}{p_i}$ может быть сколько угодно близко к нулю.

Доказательство. Покажем, что найдутся $\{c_1, c_2, \dots, c_n\}$ такие, что $|c_i| = \lceil \log \frac{1}{p_i} \rceil$. Код существует, т.к. для длин c_i выполняется неравенство Крафта-Макмилана:

$$\sum_{i=1}^{n} 2^{-|c_i|} = \sum_{i=1}^{n} 2^{-\lceil \log \frac{1}{p_i} \rceil} \le \sum_{i=1}^{n} 2^{-\log \frac{1}{p_i}} = \sum_{i=1}^{n} p_i = 1.$$

Теперь оценим среднюю длину кода:

$$\sum_{i=1}^{n} p_i \cdot |c_i| = \sum_{i=1}^{n} p_i \cdot \lceil \log \frac{1}{p_i} \rceil < \sum_{i=1}^{n} p_i \cdot \left(\log \frac{1}{p_i} + 1\right) = \left(\sum_{i=1}^{n} p_i \cdot \log \frac{1}{p_i}\right) + 1.$$

□

3.2. Код Шеннона-Фано

Упорядочим вероятности символов по убыванию: $p_1 \ge p_2 \ge \dots \ge p_n$. Уложим на прямой без пропусков отрезки длиной p_1, p_2,..., p_n и обозначим i-ый отрезок через S_i, а их объединение — через S. Коды тех букв a_i, для которых отрезок S_i попал в левую половину S, будут начинаться с '0', а коды тех букв, для которых отрезок S_i попал в правую часть S — с '1'. Центральный отрезок может не попасть целиком в одну из половин S. Если центральный отрезок является первым или последним, то начнём его код, соответственно, с '0' или '1'. В противном случае отнесём его в произвольную половину S. Далее применяем эту стратегию отдельно для букв из левой половины S и отдельно для правой половины S. Повторяем так пока не получим уникальные коды для всех символов.

Определение 3.3. Будем называть кодирование, при котором для некоторой константы c и для всех i выполняется $|c_i| \le -\log p_i + c$, *сбалансированным.*

Теорема 3.4 (Шеннон)**.** *Средняя длина кода Шеннона-Фано близка к энтропии, но не обязательно оптимальна:*

$$\sum_{i=1}^{n} p_i \cdot |c_i| = H + O(1).$$

3.3. Код Хаффмана

Определение 3.4. Будем строить *код Хаффмана* по индукции. При $n = 2$ коды $c_1 = \langle 0 \rangle$, $c_2 = \langle 1 \rangle$. При $n > 2$ будем предполагать, что вероятности упорядочены по убыванию $p_1 \ge p_2 \ge \dots \ge p_n$. Заменим символы a_{n-1} и a_n на символ a'_{n-1} с вероятностью $p'_{n-1} = p_{n-1} + p_n$. Построим код Хаффмана для $n-1$ символа. Для символов a_{n-1} и a_n возьмём коды $c_{n-1} = c'_{n-1}0$ и $c_n = c'_{n-1}1$.

Лемма 3.2. *Средняя длина кодового слова для кода Хаффмана оптимальна, т.е. не превосходит средней длины любого другого префиксного кода (а значит и любого однозначно декодируемого).*

Следствие 3.2. *Для кода Хаффмана выполняется неравенство из теоремы Шеннона 3.3.*

Замечание 3.2. На энтропию случайной величины иногда удобно смотреть как на среднюю длину кода Хаффмана.

3.4. Блоковое кодирование

Для того, чтобы нивелировать неустранимую '+1' в средней длине кода, мы будем кодировать не отдельные символы, а блоки символов. Пусть каждый блок состоит из k символов. Пусть случайные величины $\alpha_1, \alpha_2, \ldots, \alpha_k$ распределены как α и соответствуют буквам в блоке.

$$H(\alpha_1, \alpha_2, \ldots, \alpha_k) = \sum_{i=1}^{k} H(\alpha_i) = k \cdot H(\alpha).$$

Тогда по теоремам Шеннона получается следующее ограничение на среднюю длину кода символа в блоке:

$$H(\alpha) \leq (\text{средняя длина кода буквы в блоке}) \leq H(\alpha) + \frac{1}{k}.$$

При кодировании блоков длины 100 мы получаем отклонение от энтропии не более, чем на 0.01. Однако мы не можем применить код Хаффмана, т.к. на вход алгоритму его построения нужно было бы передать n^{100} частот символов.

3.5. Арифметическое кодирование

Мы построим код со следующим ограничением на среднюю длину:

$$\sum_{i=1}^{n} p_i \cdot |c_i| \leq \sum_{i=1}^{n} p_i \cdot \log \frac{1}{p_i} + 2,$$

что хуже, чем в теореме Шеннона.

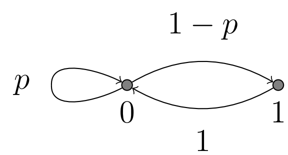
Определение 3.5. Будем называть полуинтервал *стандартным*, если он имеет вид $[0.v0_2, 0.v1_2)$, где v — это некоторая последовательность битов, а числа записаны в двоичной системе счисления. Будем сопоставлять каждому стандартному интервалу $[0.v0_2, 0.v1_2)$ код $v0$.

Для первой буквы кода на отрезке [0,1] мы отложим слева направо непересекающиеся интервалы длины p_i. Пусть первая буква блока — это a_{i_1}, тогда для второй буквы кода мы внутри интервала соответствующего p_{i_j} повторим эту операцию (отложим непересекающиеся интервалы), но длины интервалов будут уже масштабированы с коэффициентом p_i. Повторим эту операцию k раз. Получившемуся интервалу в качестве его кода сопоставим код наибольшего стандартного интервала, который полностью содержится внутри него.

Утверждение 3.2. *В интервале $[a,b)$ всегда найдётся стандартный интервал длины 2^{-k}, где $\frac{b-a}{4} < 2^{-k} \le \frac{b-a}{2}$, т.е. длина кода любого интервала при арифметическом кодировании не превосходит $\log\frac{4}{b-a} = \log\frac{1}{p} + 2$, где p — вероятность соответствующего блока.*

Замечание 3.3. В случае Марковской цепи можно строить код с соответствующими условными вероятностями.

Упражнение 3.1. Пусть Марковская цепь задана графом.



Определим $h_p = \lim_{n\to\infty}\frac{H(\alpha_1,\alpha_2,\ldots,\alpha_n)}{n}$. Найти $\max\limits_{p} h_p$.

3.6. Блоковые коды с ошибками

Пусть $\alpha_1, \alpha_2, \ldots, \alpha_n$ — независимые одинаково распределённые на $\{a_1, a_2, \ldots, a_k\}$ случайные величины с вероятностями $p_1, p_2, \ldots, p_k$. Рассмотрим блоковое кодирование, заданное функциями E_n и D_n:

$$E_n : \{a_1, a_2, \ldots, a_k\}^n \to \{0,1\}^{L_n},$$

$$D_n : \{0,1\}^{L_n} \to \{a_1, a_2, \ldots, a_k\}^n,$$

Определение 3.6. *Вероятность ошибки ε_n* — это вероятность следующего события: $[(\alpha_1, \alpha_2, \ldots, \alpha_n) = (a_{i_1}, a_{i_2}, \ldots, a_{i_n}) \mid D_n(E_n(a_{i_1}, a_{i_2}, \ldots, a_{i_n})) \ne (a_{i_1}, a_{i_2}, \ldots, a_{i_n})]$.

Теорема 3.5 (Шеннон)**.** *При блоковом кодировании допускающем ошибки выполняются следующие соотношения.*

1. *Если $h > H(\alpha) = \sum_{i=1}^{k} p_i \log\frac{1}{p_i}$, то существует функции (E_n, D_n) для $L_n = \lceil h\cdot n\rceil$, такие что $\varepsilon_n \to 0$ при $n \to \infty$.*

2. *Если $h < H(\alpha) = \sum_{i=1}^{k} p_i \log\frac{1}{p_i}$, то для любых функций (E_n, D_n) для $L_n = \lceil h\cdot n\rceil$ вероятность ошибки $\varepsilon_n \to 1$ при $n \to \infty$.*

Определение 3.7. Будем называть слово $w = \langle a_{i_1}, a_{i_2}, \ldots, a_{i_n}\rangle$ *δ-типичным*, если каждая буква a_j встречается в нём t_j раз, причём

$$\begin{cases} t_j \le (p_j + \delta)\cdot n, \\ t_j \ge (p_j - \delta)\cdot n. \end{cases}$$

Лемма 3.3. *Для $\delta = n^{-0.49} = \frac{n^{0.01}}{\sqrt{n}}$ вероятность не δ-типичного не превосходит ε_n, для $\varepsilon_n \to 0$.*

Доказательство. Применить неравенство Чебышева

$$\mathrm{P}[|X-\mu| \geq \delta n] \leq \frac{\sigma^2}{(\delta n)^2} = \frac{np_i(1-p_i)}{\delta^2 n^2} = O(n^{-0.02}).$$

□

Лемма 3.4. *Для $\delta = n^{-0.49}$ и $h > H(\alpha)$ количество δ-типичных слов не превосходит $2^{h\cdot n}$ (при достаточно больших n).*

Доказательство. Давайте для начала рассмотрим слова определённого *типа*, в которых буква i встречается n_i раз, $n_1+n_2+\cdots+n_k = n$. Сначала оценим количество слов типа, в котором $n_i = n \cdot p_i$. Таких слов

$$\frac{n!}{n_1! n_2! \cdots n_k!}.$$

По формуле Стирлинга $n! \sim \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \cdot (1+o(1))$.

$$\log \frac{n!}{n_1!n_2!\cdots n_k!} \approx \log \frac{\mathrm{poly}(n)\left(\frac{n}{e}\right)^n}{\mathrm{poly}(n)\left(\frac{n_1}{e}\right)^{n_1}\cdots\left(\frac{n_k}{e}\right)^{n_k}} =$$
$$= \log\left(\frac{n}{n_1}\right)^{n_1}\cdots\left(\frac{n}{n_k}\right)^{n_k} + O(\log n) = \sum_{i=1}^{k} \underbrace{np_i}_{n_i} \cdot \log \tfrac{1}{p_i} + O(\log n) < h \cdot n. \quad (2)$$

Последнее неравенство выполняется асимптотически, т.к. по предположению $h > H(\alpha)$. Мы оценили это только для конкретного типа слов. Давайте оценим для произвольного δ-типичного слова с $n_i = n \cdot (p_i + \Delta_i)$, где $|\Delta_i| \leq \delta$. Тогда (2) изменится следующим образом:

$$\cdots = \sum_{i=1}^{k} n(p_i+\Delta_i)\cdot \log \tfrac{1}{p_i+\Delta_i} + O(\log n) = n \cdot \sum_{i=1}^{k} p_i \cdot \log \tfrac{1}{p_i} + O(\log n) + n \cdot O(\delta) < h \cdot n.$$

(Действительно, энтропия — это непрерывная функция, а значит при небольшом отклонении она изменяется на $c \cdot \max_i \Delta_i$, где c зависит от производной функции энтропии.) Итого общее количество δ-типичных слов можно оценить как количество типов умноженное на количество δ-типичных слов одного типа:

$$\mathrm{poly}(n) \cdot 2^{n\cdot H(\alpha) + n \cdot O(\delta) + O(\log n)} < 2^{h\cdot n}.$$

□

Доказательство теоремы 3.5.

1. Если мы будем кодировать только δ-типичные слова, то по лемме 3.4 нам будет достаточно длины кода L_n, а вероятность всех не типичных слов будет стремиться к нулю.

2. Обозначим за $\hat{\varepsilon}_n$ вероятность ошибки при декодировании δ-типичных слов. Мы хотим показать, что $\hat{\varepsilon}_n \to 1$. Давайте рассмотрим конкретное δ-типичное слово $w = \langle a_{i_1}, a_{i_2}, \dots, a_{i_n} \rangle$. Пусть $p'_1, p'_2, \dots, p'_n$ — это частоты букв $a_1, a_2, \dots, a_n$ в слове w. Оценим вероятность появления w:

$$\Pr[\langle \alpha_{i_1}, \alpha_{i_2}, \dots, \alpha_{i_n} \rangle = w] = p_1^{p'_1 \cdot n} \cdot \ldots \cdot p_k^{p'_k \cdot n} = 2^{-(\sum_i p'_i \log \frac{1}{p_i}) \cdot n} \le 2^{-(\sum_i p_i \log \frac{1}{p_i}) \cdot n + O(\delta_n \cdot n)}.$$

Всего мы можем корректно закодировать не более 2^{L_n} δ-типичных слов, т.е. вероятность корректно декодировать δ-типичное слово

$$1 - \hat{\varepsilon}_n \le 2^{L_n} \cdot 2^{-H(\alpha) \cdot n + O(\delta_n \cdot n)} \le 2^{h \cdot n + 1} \cdot 2^{-H(\alpha) \cdot n + O(\delta_n \cdot n)} \to 0.$$

Таким образом $\hat{\varepsilon}_n \to 1$. Вместе с леммой 3.3 получаем, что $\varepsilon_n \to 1$.

□

Замечание 3.4. Используя предыдущую теорему можно, например, получить альтернативное доказательство неравенства $H(\alpha, \beta) \le H(\alpha) + H(\beta)$. В левой части стоит асимптотическая средняя длина кода при блоковом кодировании (α, β), а справа сумма средних длин кодов при блоковом кодировании α и β отдельно друг от друга. Т.к. мы можем рассмотреть кодирование (α, β) как конкатенацию кодов для α и β, то неравенство выполняется.

4. Свойства распределений

4.1. Энтропийные профили

Утверждение 4.1. *Для любого $h \ge 0$ существует распределение α: $H(\alpha) = h$.*

Доказательство. Возьмём некоторое целое n: $0 \le h \le \log n$. Искомое распределение — это линейная комбинация распределений с вероятностями $(1, 0, \dots, 0)$ и $(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})$. □

Каким может быть совместное распределение двух случайных величин α и β? Рассмотрим как может быть устроен *энтропийный профиль* $(H(\alpha), H(\beta), H(\alpha, \beta))$.

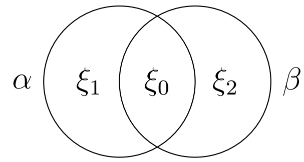
Утверждение 4.2. *Для любых чисел $h_1, h_2, h_{12} \ge 0$, которые удовлетворяют следующим соотношениям*

$$\begin{cases} h_{12} \le h_1 + h_2 & \iff \quad t_0 = I(\alpha : \beta) \ge 0, \\ h_2 \le h_{12} & \iff \quad t_1 = H(\alpha \mid \beta) \ge 0, \\ h_1 \le h_{12} & \iff \quad t_2 = H(\beta \mid \alpha) \ge 0. \end{cases}$$

существует пара случайных величин (α, β) с энтропийным профилем (h_1, h_2, h_{12}).

Доказательство. Пусть ξ_0, ξ_1, ξ_2 — независимые случайные величины с энтропиями t_0, t_1, t_2 соответственно. Тогда $\alpha = (\xi_0, \xi_1)$ и $\beta = (\xi_0, \xi_2)$ будут искомыми величинами.

$$\begin{cases} H(\xi_0) = t_0 = h_1 + h_2 - h_{12}, \\ H(\xi_1) = t_1 = h_{12} - h_2, \\ H(\xi_2) = t_2 = h_{12} - h_1. \end{cases}$$



□

Давайте попробуем разобраться с аналогичным вопросом для троек случайных величин. Энтропийный профиль для тройки (α, β, γ) будет задаваться 7 числами:

$$\big(H(\alpha), H(\beta), H(\gamma), H(\alpha, \beta), H(\alpha, \gamma), H(\beta, \gamma), H(\alpha, \beta, \gamma)\big).$$

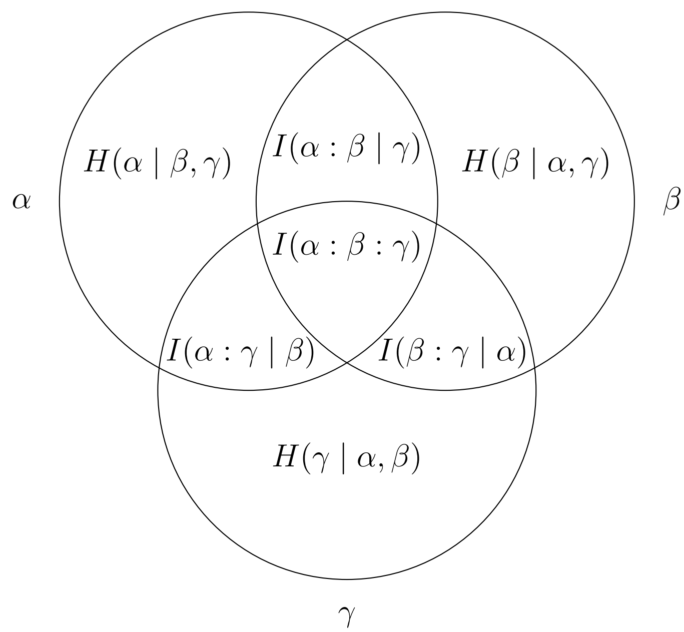
Для случайных величин (α, β, γ) можно записать 9 независимых неравенств.

$$\begin{array}{lll} H(\alpha \mid \beta, \gamma) \geq 0, & I(\alpha : \beta) \geq 0, & I(\alpha : \beta \mid \gamma) \geq 0, \\ H(\beta \mid \gamma, \alpha) \geq 0, & I(\beta : \gamma) \geq 0, & I(\beta : \gamma \mid \alpha) \geq 0, \\ H(\gamma \mid \alpha, \beta) \geq 0, & I(\gamma : \alpha) \geq 0, & I(\gamma : \alpha \mid \beta) \geq 0. \end{array}$$

Определение 4.1. Определим общую информацию трёх случайных величин

$$I(\alpha : \beta : \gamma) = I(\alpha : \beta) - I(\alpha : \beta \mid \gamma).$$

Соотношения на информационные величины имеют удобную геометрическую интерпретацию. Давайте нарисуем три круга Эйлера и сопоставим площади каждой из получившихся замкнутых области некоторую информационную величину.



Мы можем проверить, что в результате получится корректное представление. Так, например, площадь круга α будет соответствовать

$$H(\alpha) = H(\alpha \mid \beta, \gamma) + I(\alpha : \beta \mid \gamma) + I(\alpha : \gamma \mid \beta) + I(\alpha : \beta : \gamma),$$

а пересечение кругов α и β

$$I(\alpha : \beta) = I(\alpha : \beta \mid \gamma) + I(\alpha : \beta : \gamma).$$

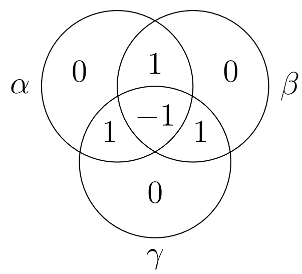
В дальнейшем мы будем использовать эту геометрической интерпретацию для доказательства соотношений на информационные величины.

Утверждение 4.3. *Общая информация трёх случайных величин может быть отрицательной.*

Доказательство. Пусть α и β будут независимыми равномерно распределёнными на $\{0,1\}$ случайными величинами. Случайная величина γ будет принимать значение из $\{0,1\}$ в соответствии со следующим соотношением:

$$\alpha \oplus \beta \oplus \gamma = 0.$$

Мы получим следующую картину:

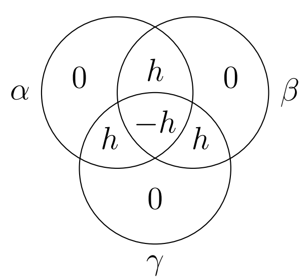


□

Утверждение 4.4. *Других неравенств для троек нет.*

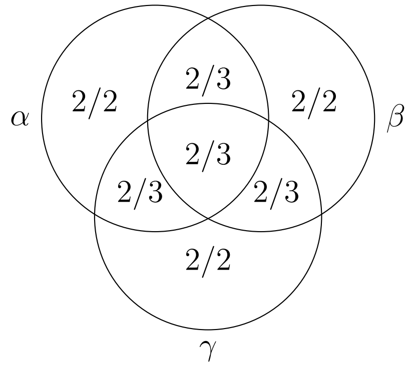
Утверждение 4.5. *Есть профили, которые не реализуются никакими распределениями, но их мера 0.*

Упражнение 4.1. Доказать, что следующий профиль реализуется только при $h = \log n$ для некоторого целого n.



Утверждение 4.6. $2H(\alpha,\beta,\gamma) \le H(\alpha,\beta) + H(\alpha,\gamma) + H(\beta,\gamma)$.

Доказательство. Отметим сколько раз каждая область входит в левую/в правую часть неравенства.



Таким образом утверждение упрощается до $0 \le I(\beta : \gamma) + I(\alpha : \beta \mid \gamma) + I(\alpha : \gamma \mid \beta)$. □

Следствие 4.1 (Теорема 1.3)**.** *Для* $A \subset \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^*$

$$2\chi(A) \le \chi_{12}(A) + \chi_{13}(A) + \chi_{23}(A).$$

Доказательство. Пусть (α, β, γ) равномерно распределены на A (т.е. случайные величины — это координаты точек в множестве A).

$$2\chi(A) = 2H(\alpha,\beta,\gamma) \le \underbrace{H(\alpha,\beta)}_{\le \chi_{12}(A)} + \underbrace{H(\alpha,\gamma)}_{\le \chi_{13}(A)} + \underbrace{H(\beta,\gamma)}_{\le \chi_{23}(A)}.$$

□

Можно рассмотреть обобщение этой теоремы на произвольное число координат.

Теорема 4.1 (Лемма Ширера)**.** *Пусть X — случайная величина, распределённая на $\{0,1\}^n$. Для любого распределения S на подмножествах $[n]$, при котором $\Pr[i \in S] \ge \mu$, выполняется $\mathbb{E}[H(X_S)] \ge \mu \cdot H(X)$.*

Доказательство. Для любого множества $T = \{i_1, i_2, \ldots, i_k\} \subset [n]$, $i_1 < i_2 < \cdots < i_k$ выполняется

$$H(X_T) = H(X_{i_1}) + H(X_{i_2} \mid X_{i_1}) + \cdots + H(X_{i_k} \mid X_{i_1}, \ldots, X_{i_{k-1}}).$$

Воспользуемся тем, что $H(X_{i_t} \mid X_{i_1}, \ldots, X_{i_{t-1}}) \ge H(X_{i_t} \mid X_{<i_t})$, тогда

$$H(X_T) \ge H(X_{i_1} \mid X_{<i_1}) + H(X_{i_2} \mid X_{<i_2}) + \cdots + H(X_{i_k} \mid X_{<i_k}).$$

Теперь применим этот факт к распределению S.

$$\mathop{\mathbb{E}}_S[H(X_S)] \ge \mathop{\mathbb{E}}_S\left[\sum_{i \in S} H(X_i \mid X_{<i})\right] = \sum_{i \in [n]} \Pr[i \in S] \cdot H(X_i \mid X_{<i})$$
$$\ge \mu \sum_{i \in [n]} H(X_i \mid X_{<i}) = \mu \cdot H(X).$$

□

У леммы Ширера имеется множество применений.

Пример 4.1 (Подсчёт треугольников в графе). Пусть $G = (V, E)$ — неориентированный граф с t треугольниками, и пусть $\ell = |E|$. Покажем, что $t \le (2\ell)^{3/2}/6$.

Доказательство. Пусть тройка случайных величин (α, β, γ) равномерно распределена на вершинах треугольников, и пусть $X = (\alpha, \beta, \gamma)$. Тогда $H(X) = H(\alpha, \beta, \gamma) = \log(6t)$, т.к. каждый треугольник случается шестью различными перестановками. Рассмотрим распределение S, равномерное на подмножествах $\{1, 2, 3\}$ размера 2. Тогда $\Pr[i \in S] = 2/3$. По лемме Ширера

$$\mathop{\mathbb{E}}_{S}[H(X_S)] \ge \frac{2}{3}\log(6t),$$

т.е. существует $T \subset \{1, 2, 3\}$, для которого $H(X_T) \ge \frac{2}{3}\log(6t)$. С другой стороны X_T — это распределение на рёбрах графа, то есть $\log(2\ell) \ge H(X_T)$. Из этого получаем, что $2\ell \ge (6t)^{2/3}$. □

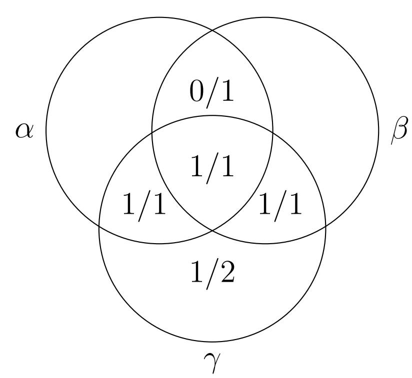
Обобщение для вложения произвольных графов см. в [11].

Утверждение 4.7. *Для любых α, β и γ выполняется следующее неравенство*

$$H(\gamma) \le H(\gamma \mid \alpha) + H(\gamma \mid \beta) + I(\alpha : \beta).$$

Если $H(\gamma \mid \alpha) = H(\gamma \mid \beta) = 0$ (т.е. γ однозначно определяется и по α и по β), то $H(\gamma) \le I(\alpha : \beta)$.

Доказательство. Отметим сколько раз каждая область входит в левую/в правую часть неравенства.

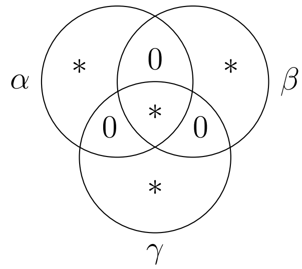


Таким образом неравенство упрощается до $0 \le H(\gamma \mid \alpha, \beta) + I(\alpha : \beta \mid \gamma)$. □

Упражнение 4.2. Пусть $\alpha \to \beta \to \gamma$ образуют Марковскую цепь, т.е. распределение $\langle \gamma \mid \beta \rangle = \langle \gamma \mid \alpha, \beta \rangle$. Докажите, что $I(\alpha : \gamma) \le I(\alpha : \beta)$ и $I(\alpha : \gamma) \le I(\beta : \gamma)$.

Упражнение 4.3. Пусть $\alpha \to \beta \to \gamma \to \delta$ образуют Марковскую цепь. Докажите, что $I(\alpha : \beta) \le I(\beta : \gamma)$.

Упражнение 4.4. Пусть α, β и γ имеют следующий профиль.



Докажите, что существует случайная величина δ, такая что

$$\begin{cases} H(\delta \mid \alpha) = 0, \\ H(\delta \mid \beta) = 0, \\ H(\delta \mid \gamma) = 0, \\ H(\delta) = I(\alpha : \beta : \gamma). \end{cases}$$

И при этом $I(\alpha : \beta \mid \delta) = I(\alpha : \gamma \mid \delta) = I(\beta : \gamma \mid \delta) = 0$.

Упражнение 4.5. Возьмём в качестве x, y, a, b случайные величины из предыдущего упражнения: $x = \alpha$, $y = \beta$, $a = \gamma$, $b = \delta$. Покажите, что для любых таких (a, b, x, y) из условия $I(x : y \mid a) = I(x : a \mid y) = I(y : a \mid x) = 0$ следует

$$I(a : b) \leq I(a : b \mid x) + I(a : b \mid y) + I(x : y).$$

[Указание: примените неравенство из утверждения 4.7.]

Упражнение 4.6. Возьмём в качестве x, y, a, b случайные величины из упражнения 4.4: $x = \alpha$, $y = \beta$, $a = \gamma$, $b = \delta$. Покажите, что существуют такие (a, b, x, y), для которых

$$I(a : b) \not\leq I(a : b \mid x) + I(a : b \mid y) + I(x : y).$$

(Т.е. условие в предыдущем упражнении было необходимо.)

Утверждение 4.8 (Неравенство для 5 случайных величин)**.**

$$I(a : b) \leq I(a : b \mid x) + I(a : b \mid y) + I(x : y) + I(a : b \mid z) + I(a : z \mid b) + I(b : z \mid a).$$

Следствие 4.2 (Zhang, Yeung, 1998)**.** *Неравенство для 4 случайных величин, которое не выражается через базисные неравенства.*

$$I(a : b) \leq 2I(a : b \mid x) + I(a : b \mid y) + I(x : y) + I(a : x \mid b) + I(b : x \mid a).$$

Утверждение 4.9. *Для 4 случайных величин существует бесконечно много неравенств, которые независимы в совокупности.*

4.2. Неравенства о тройках

Будем в различных предположениях доказывать следующее утверждение

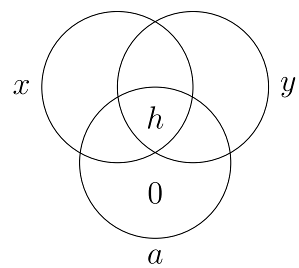
$$H(a \mid x) + H(a \mid y) \leq H(a).$$

Утверждение 4.10. *Если a, x, y такие, что*

$$\begin{cases} H(a \mid x, y) = 0, \\ I(x : y \mid a) = 0. \end{cases}$$

то $H(a \mid x) + H(a \mid y) \leq H(a)$.

Доказательство. Получается, что нам нужно доказать неотрицательность h.



Т.к. $I(x : y \mid a) = 0$, то $h = I(x : y) \geq 0$. □

Утверждение 4.11. *Если a, x, y такие, что $H(a \mid x, y) = 0$ и*

$$\begin{cases} A_i \sim X_j \\ A_i \sim Y_k \end{cases} \implies A_i \sim (X_j, Y_k),$$

то $H(a \mid x) + H(a \mid y) \leq H(a)$. (Обозначение $A_i \sim X_j \iff \Pr[a = A_i \wedge x = X_j] > 0$.)

Замечание 4.1. Условие $H(a \mid x, y) = 0$ можно интерпретировать так: $a = f(x, y)$.

Доказательство. Построим новое распределение (a', x', y'):

- a' имеет то же распределение, что и a,
- условное распределение x' при условии a' совпадает с условным распределением x при условии a,
- условное распределение y' при условии a' совпадает с условным распределением y при условии a,
- x' и y' независимы.

$$\Pr[a' = A_i, x' = X_j, y' = Y_k] = \Pr[a' = A_i] \cdot \Pr[x' = X_j \mid a' = A_i] \cdot \Pr[y' = Y_k \mid a' = A_i].$$

Таким образом

$$H(a', x', y') = H(a') + H(x' \mid a') + H(y' \mid a') - \underbrace{I(x' : y' \mid a')}_{0}.$$

С другой стороны

$$H(a', x', y') \leq H(x') + H(y') + H(a' \mid x', y').$$

Кроме того, мы может стереть штрихи почти везде.

$$H(x) + H(y) + H(a' \mid x', y') \geq H(a', x', y') = H(a) + H(x \mid a) + H(y \mid a).$$

Покажем, что $H(a' \mid x', y') = 0$, т.е. $a' = f(x', y')$. Действительно: если тройка (A_i, X_j, Y_k) в новом распределении встречается с положительной вероятностью, то и в исходном распределении она так же встречалась с положительной вероятностью, следовательно $a' = f(x', y')$. Получаем: $H(a) + H(x \mid a) + H(y \mid a) \leq H(x) + H(y)$. Прибавим $H(a)$ к обеим частям неравенства:

$$H(x, a) + H(y, a) \leq H(x) + H(y) + H(a) \implies H(a \mid x) + H(a \mid y) \leq H(a).$$

□

Задача 4.1 (Верещагин [9]). Рассмотрим двудольный граф с вершинами (L, R) с цветными рёбрами. Все рёбра инцидентные одной вершине разноцветные, степень в левой доле не меньше n, в правой — не меньше m. Пусть известно, что для пары вершин $(x \in L, y \in R)$ есть не более одного общего цвета. Докажите, что количество цветов хотя бы $n \cdot m$.

Заметим, что одноцветные рёбра образуют паросочетания. Для каждого цвета c соединим все согласованные с c вершины слева с согласованными с c вершинами справа. Получим биклику из рёбер цвета c.

Рассмотрим распределение на тройках (a, x, y) (цвет, вершина из левой доли, вершина из правой доли): выбираем цвет пропорционально размеру (количеству рёбер) соответствующей биклики и выбираем случайное ребро этого цвета. Можно проверить, что выполняется следующее соотношение:

$$\begin{cases} A_i \sim X_j, \\ A_i \sim Y_k, \end{cases} \implies A_i \sim (X_j, Y_k).$$

Теперь применим: $\underbrace{H(a \mid x)}_{\geq \log n} + \underbrace{H(a \mid y)}_{\geq \log m} \leq H(a) \leq \log(\#\text{ цветов})$.

4.3. Условное неравенство о четвёрке

Утверждение 4.12. *Если для случайных величин a, b, x, y выполняется*

$$\begin{cases} I(x : y \mid a) = 0, \\ H(a \mid x, y) = 0, \end{cases}$$

то $I(a : b) \leq I(a : b \mid x) + I(a : b \mid y) + I(x : y)$.

Доказательство. Построим новое распределение (a', b', x', y'): сначала выберем значение $(a', b') \sim (a, b)$. При фиксированном значении (a', b') выбираем независимо x' и y' так, чтобы условные распределения вероятностей относительно a' были такими же, как у x и y относительно a.

$$H(a', b', x', y') = H(a', b') + H(x \mid a', b') + H(y \mid a', b') - \underbrace{I(x' : y' \mid a', b')}_{0} =$$
$$= H(a, b) + H(x \mid a, b) + H(y \mid a, b).$$

С другой стороны

$$H(a', b', x', y') \leq H(b') + H(x' \mid b') + H(y' \mid b') + H(a' \mid x', y') =$$
$$= H(b) + H(x \mid b) + H(y \mid b) + H(a' \mid x', y').$$

Покажем, что $H(a' \mid x', y') = 0$. В исходном распределении это выполнялось по условию. Пусть $[a' = A_i, x' = X_j, y' = Y_k]$ в новом распределении случается с положительной вероятностью. Следовательно и в исходном распределении это случается с положительной вероятностью (при фиксированном a' величины x' и y' независимы), а значит сохраняется соответствующее свойство функциональной зависимости a' от (x', y').

В результате получаем

$$H(a, b) + H(x \mid a, b) + H(y \mid a, b) \leq H(b) + H(x \mid b) + H(y \mid b).$$

Распишем это неравенство в безусловных энтропиях:

$$H(a, b) + H(x, a, b) - H(a, b) + H(y, a, b) - H(a, b) \leq H(b) + H(x, b) - H(b) + H(y, b) - H(b).$$

Упрощаем и получаем:

$$H(x, a, b) + H(y, a, b) + H(b) \leq H(x, b) + H(y, b) + H(a, b). \tag{3}$$

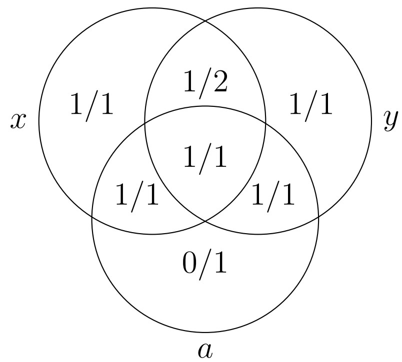
Проделаем то же самое с $I(a : b) \leq I(a : b \mid x) + I(a : b \mid y) + I(x : y)$.

$$\begin{aligned} H(a) + H(b) - H(a, b) \leq \quad & H(a, x) + H(b, x) - H(a, b, x) - H(x) + \\ & H(a, y) + H(b, y) - H(a, b, y) - H(y) + \\ & H(x) + H(y) - H(x, y). \end{aligned}$$

Упрощаем и получаем:

$$H(a,b,x)+H(a,b,y)+H(b)+H(x,y) \le \begin{aligned}[t] &H(b,x)+H(b,y)+H(a,b)+\\ &H(a,x)+H(a,y)-H(a). \end{aligned} \tag{4}$$

Заметим, что нам осталось доказать лишь $H(x,y) \le H(a) + H(x \mid a) + H(y \mid a)$. Сложив это неравенство с (3) мы получим (4). Отметим сколько раз каждая область входит в левую/в правую часть неравенства.



Т.е. оно эквивалентно $H(a \mid x,y) + I(x:y \mid a) \ge 0$. $\square$

Вопросы на подумать. Придумать интерпретацию для этого неравенства. Zhang и Yeung в 97 году доказали это же неравенство в предположении $I(x:y) = I(x:y \mid a) = 0$. Есть ли комбинаторная интерпретация у этого утверждения?

Упражнение 4.7. Прямоугольная таблица разбита на (комбинаторные) прямоугольники таким образом, что каждая строка пересекает не менее n прямоугольников, а каждый столбец — не менее m прямоугольников. Докажите, что общее число прямоугольников не менее nm.

5. Криптография

5.1. Шифрования с закрытым ключом

Рассмотрим задачу кодирования сообщения при помощи симметричного шифрования. Будем считать, что вычислительные ресурсы противника неограниченны. Предположим, что мы шифруем сообщение m с ключом шифрования k. При шифровании сообщения мы получаем *шифрограмму* $c = E(k,m)$. Получатель шифрограммы тоже знает ключ k и может узнать исходное сообщение $m = D(k,c)$.

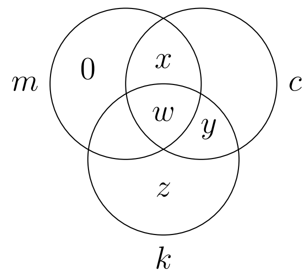
Будем предполагать, что m и k являются случайными величинами. Противник не знает m и k, но знает c. Для *совершенной* схемы шифрования должны выполняться следующие соотношения:

$$\begin{cases} H(c \mid k,m) = 0, \\ H(m \mid k,c) = 0, \\ I(c:m) = 0. \end{cases}$$

Теорема 5.1 (Шеннон). *$H(k) \geq H(m)$, даже если условие $H(c \mid k, m) = 0$ нарушается (т.е. алгоритм E использует случайные биты).*

Замечание 5.1. Одноразовый блокнот (one-time notepad) обладает этим свойством.

Доказательство. По условию $x + w = 0$, т.е. $x = -w$.



Т.к. взаимная информация неотрицательна, то $w + y \geq 0$, т.е. $y \geq -w = x$. Теперь из $y \geq x$ и $z \geq 0$ следует $H(k) \geq H(m)$. □

5.2. Схемы разделения секрета

Пусть у нас есть некоторый секрет S_0 и n участников и мы хотим разделить между ними этот секрет так, чтобы они могли им воспользоваться только все вместе, а любое подмножество участников — не могло.

Определение 5.1. *Совершенная схема разделения секрета* — это совместное распределение вероятностей $(S_0, S_1, S_2, \dots, S_n)$, такое что

$$\begin{cases} H(S_0 \mid S_1, S_2, \dots, S_n) = 0, \\ H(S_0 \mid S_{i_1}, S_{i_2}, \dots, S_{i_k}) = H(S_0), \quad k < n. \end{cases}$$

Второе условие можно переписать как $I(S_0 : S_{i_1}, S_{i_2}, \dots, S_{i_k}) = 0$.

Для совершенной схемы разделения секрета есть простая конструкция. Будем считать, что S_0 записан (закодирован) при помощи ℓ бит. Выберем независимо и равномерно $S_1, \dots, S_{n-1} \in \{0,1\}^\ell$. S_n определяется из условия $S_0 \oplus S_1 \oplus S_2 \oplus \dots \oplus S_n = \vec{0}$ (покоординатная сумма по модулю 2).

Утверждение 5.1. *Предложенная схема разделения секрета является совершенной.*

Определение 5.2. *Пороговая совершенная схема разделения секрета* — это совместное распределение вероятностей $(S_0, S_1, S_2, \dots, S_n)$, такое что

$$\begin{cases} H(S_0 \mid S_{i_1}, S_{i_2}, \dots, S_{i_t}) = 0, \\ H(S_0 \mid S_{i_1}, S_{i_2}, \dots, S_{i_k}) = H(S_0), \quad k < t. \end{cases}$$

Пороговая схема Шамира. Будем считать, что секрет S_0 — это элемент некоторого конечного поля $\mathbb{F}_q$. Выберем случайный многочлен p над полем $\mathbb{F}_q$ степени не более $t-1$: выберем $t-1$ коэффициент независимо и равномерно, а последний (свободный) коэффициент определим из соотношения $p(0) = S_0$. Выберем произвольным образом и сообщим всем участникам некоторый набор различных ненулевых элементов поля $a_1, a_2, \ldots, a_n \in \mathbb{F}_q$ и вычислим секреты участников как значение полинома в соответствующих точках $S_i = p(a_i)$. Теперь любые t участником могут собраться, воспользоваться формулой для интерполяции многочлена и вычислить $S_0 = p(0)$. Если же соберётся меньше участников, то у них не будет никакой информации об S_0.

Утверждение 5.2. *Пороговая схема Шамира является совершенной.*

Доказательство. Любой полином степени меньше $t-1$ можно дополнить до полинома большей степени с любым значением в точке 0. □

Определение 5.3. *Совершенная схема разделения секрета для структуры доступа* $\Gamma \subset 2^{[n]}$ (Γ должно быть замкнуто вверх) — это совместное распределение вероятностей $(S_0, S_1, S_2, \ldots, S_n)$, такое что

$$\begin{cases} H(S_0 \mid S_{i_1}, S_{i_2}, \ldots, S_{i_m}) = 0, & \{i_1, i_2, \ldots, i_m\} \in \Gamma, \\ H(S_0 \mid S_{i_1}, S_{i_2}, \ldots, S_{i_m}) = H(S_0), & \{i_1, i_2, \ldots, i_m\} \notin \Gamma. \end{cases}$$

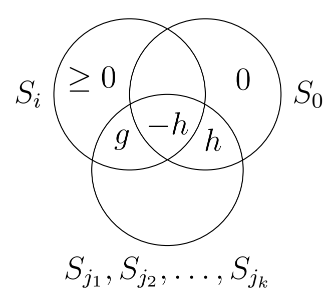
Определение 5.4. *Идеальная схема разделения секрета* — это совершенная схема разделения секрета с дополнительным требованием „экономности“.

$$\forall i \in \{1, 2, \ldots, n\},\ H(S_i) \leq H(S_0).$$

Утверждение 5.3. *Если участник i является* существенным *в структуре доступа* Γ *(т.е. существует такое $s \in \Gamma$, что $s \setminus \{i\} \notin \Gamma$), то $H(S_i) \geq H(S_0)$.*

Замечание 5.2. Схема Шамира является идеальной.

Доказательство. Пусть $s = \{i, j_1, j_2, \ldots, j_k\} \in \Gamma$, а $s \setminus \{i\} \notin \Gamma$. Обозначим взаимную информацию $I(S_0 : S_{j_1}, S_{j_2}, \ldots, S_{j_k} \mid S_i)$ за h, а $I(S_i : S_{j_1}, S_{j_2}, \ldots, S_{j_k} \mid S_0)$ за g. Из условия $I(S_0 : S_{j_1}, S_{j_2}, \ldots, S_{j_k}) = 0$ получаем, что $I(S_0 : S_i : S_{j_1}, S_{j_2}, \ldots, S_{j_k}) = -h$, аналогичным образом из $I(S_i : S_{j_1}, S_{j_2}, \ldots, S_{j_k}) \geq 0$ получаем, что $g \geq h$.



Таким образом $H(S_i) \geq H(S_0)$. □

Замечание 5.3. Это утверждение показывает, что не бывает более „экономной“ схемы разделения секрета, чем идеальная.

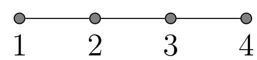
Утверждение 5.4. *Для любой системы доступа* Γ *существует совершенная схема разделения секрета.*

Доказательство. Давайте для каждого подмножества $A = \{i_1, i_2, \ldots, i_k\} \in \Gamma$ создадим собственный набор секретов $S^A_{i_1}, S^A_{i_2}, \ldots, S^A_{i_k}$: $S^A_{i_1} \oplus S^A_{i_2} \oplus \cdots \oplus S^A_{i_k} = S_0$. (Достаточно рассматривать только минимальные множества A.) $\square$

Замечание 5.4. Предложенная схема не является идеальной.

Утверждение 5.5. *Существуют структуры доступа, для которых не существует идеальной схемы разделения секрета.*

Доказательство. Рассмотрим структуру доступа, заданную следующим графом (рёбра соответствуют авторизованным множествам).

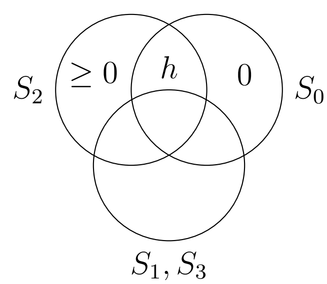


Покажем, что для этой структуры доступа $H(S_2) + H(S_3) \geq 3H(S_0)$, другими словами $\max_i \frac{H(S_i)}{H(S_0)} \geq 3/2$.

Для доказательства нам потребуются три леммы. Будем обозначать $h = H(S_0)$.

Лемма 5.1. $H(S_2 \mid S_1, S_3) \geq h$.

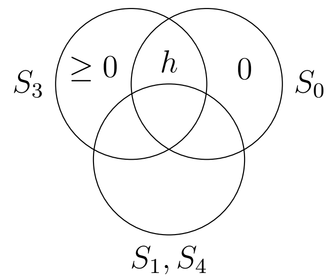
Доказательство. Второй участник может восстановить секрет, воспользовавшись либо секретом первого или секретом третьего участника, т.е. $I(S_2 : S_0 \mid S_1, S_3) = h$.



Таким образом $H(S_2 \mid S_1, S_3) \geq I(S_2 : S_0 \mid S_1, S_3) = h$. $\square$

Лемма 5.2. $H(S_3 \mid S_1) \geq h$.

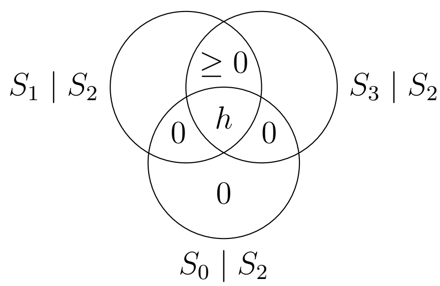
Доказательство. Аналогично предыдущей лемме получаем, что $H(S_3 \mid S_1, S_4) \geq h$, и как следствие $H(S_3 \mid S_1) \geq h$.



$\square$

Лемма 5.3. $I(S_1 : S_3 \mid S_2) \geq h$.

Доказательство. Следующую схему следует интерпретировать как энтропия при условии S_2.



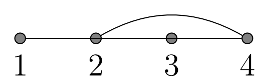
Заметим, что $I(S_1 : S_0 \mid S_2) = h$ и $I(S_3 : S_0 \mid S_2) = h$ в то время, как $I(S_1 : S_0 \mid S_2, S_3) = 0$ и $I(S_3 : S_0 \mid S_1, S_2) = 0$. Т.е. $I(S_1 : S_3 : S_0 \mid S_2) = h$, следовательно $I(S_1 : S_3 \mid S_2) \geq h$. $\square$

Теперь осталось сложить результаты трёх лемм:

$$H(S_2)+H(S_3) \geq H(S_2, S_3) = H(S_2 \mid S_1, S_3)+H(S_3 \mid S_1)+I(S_1 : S_3 \mid S_2)+I(S_2 : S_1) \geq 3h.$$

$\square$

Упражнение 5.1. Доказать, что для любой схемы разделения секреты для этой структуры $\max_i \frac{H(S_i)}{H(S_0)} \geq 3/2$.



Теорема 5.2 (Csirmaz'94)**.** *Существуют такие структуры доступа* Γ *на* n *участниках, что для любой схемы разделения секрета* $\max_i \frac{H(S_i)}{H(S_0)} \geq \Omega(n/\log n)$.

Доказательство. Выберем n и k такие, что $n = 2^k + k - 1$, и два множества участников

$$A = \{a_1, a_2, \ldots, a_k\},$$
$$B = \{b_1, b_2, \ldots, b_{2^k-1}\}.$$

Для определения структуры доступа нам потребуются два семейства множеств. Пусть $\{A_0, A_1, A_2, \ldots, A_{2^k-1}\}$ — это все подмножества A, причём $A_0 = A$ и для любых $i < j$ выполняется $A_i \not\subseteq A_j$ (например, можно их упорядочить по уменьшению размера). Построим множества $\{B_0, B_1, B_2, \ldots, B_{2^k-1}\}$ следующим образом: $B_0 = \emptyset$, $B_i = \{b_1, b_2, \ldots, b_i\}$. Теперь мы готовы определить структуру доступа Γ: $\Gamma = \{U_i\}_{i=0}^{2^k-1}$, где $U_i = A_i \cup B_i$.

Как и в предыдущих утверждениях обозначим $H(S_0)$ за h. В дальнейших рассуждениях мы будем использовать следующую нотацию: под энтропией некоторого множества участников $X = \{x_1, x_2, \ldots, x_t\} \subset A \cup B$, мы будем понимать энтропию секретов, которые принадлежат участникам этого множества, т.е. $H(X) = H(S_{x_1}, S_{x_2}, \ldots, S_{x_t})$.

Лемма 5.4. *Для* $i = \{0, 1, 2, \ldots, 2^k - 2\}$

$$H(A \cup B_i) - H(B_i) \geq H(A \cup B_{i+1}) - H(B_{i+1}) + h.$$

Из этой леммы следует, что

$$H(A) = H(A \cup B_0) - H(B_0) \geq H(A \cup B_1) - H(B_1) + h \geq \cdots \geq \\ \geq \underbrace{H(A \cup B_{2^k-1}) - H(B_{2^k-1})}_{\geq 0} + (2^k - 1) \cdot h.$$

Получаем, что $H(A) = H(S_{a_1}, S_{a_2}, \ldots, S_{a_k}) \geq (2^k - 1) \cdot h$. Следовательно есть i такое, что $H(S_{a_i}) \geq \frac{2^k-1}{k} \cdot h$. Вспомним, что мы выбрали $n = 2^k + k - 1$, т.е. $H(S_{a_i}) \geq \Omega(n/\log n) \cdot h$. Осталось доказать лемму.

Доказательство леммы 5.4. Докажем два неравенства:

1. $H(A_{i+1} \cup B_i) + H(B_{i+1}) \geq H(A_{i+1} \cup B_{i+1}) + H(B_i)$.

2. $H(A \cup B_i) + H(A_{i+1} \cup B_{i+1}) \geq H(A \cup B_{i+1}) + H(A_{i+1} \cup B_i) + h$.

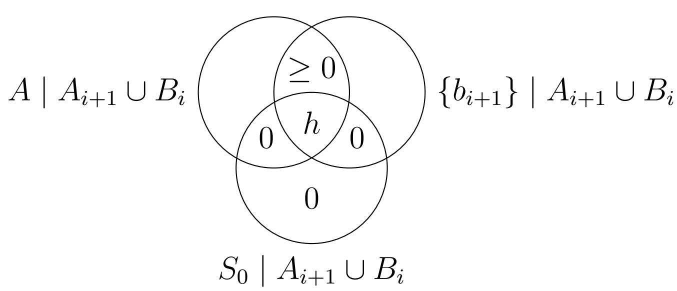
Заметим, что если сложить эти два неравенства, то мы получим утверждение леммы.

Первое неравенство говорит о неотрицательности условной совместной информации. Действительно, давайте вспомним формулу для условной совместной информации:

$$I(x : y \mid z) \geq 0 \iff H(x, z) + H(y, z) \geq H(x, y, z) + H(z).$$

Таким образом первое неравенство утверждает $I(A_{i+1} : \{b_{i+1}\} \mid B_i) \geq 0$.

Аналогично второе неравенство утверждает, $I(A : \{b_{i+1}\} \mid A_{i+1} \cup B_i) \geq h$. Доказательство этого утверждения аналогично лемме 5.3 — нужно рассмотреть условное распределение при известном $A_{i+1} \cup B_i$.



□

Эта лемма завершает доказательство теоремы. □

Замечание 5.5. Нижние оценки на избыточную сложность совершенных схем разделения секрета влекут нижние оценки на схемную сложность монотонных функций.

6. Коммуникационная сложность

Пусть X, Y и Z — это три конечных множества, и пусть задана некоторая функция $f : X \times Y \to Z$. Два игрока, будем называть их Алиса и Боб, решают *коммуникационную задачу для функции* f, если:

1. множества X, Y, Z и функция f известны обоим игрокам,
2. Алиса знает некоторое $x \in X$,
3. Боб знает некоторое $y \in Y$,
4. Алиса и Боб стремятся вычислить $f(x, y)$.

Для решения этой коммуникационной задачи Алиса и Боб могут пересылать друг другу сообщения. Задача считается решённой, если оба игрока знают $f(x, y)$. Нас интересует минимальное количество битов, которое необходимо и достаточно переслать для вычисления $f(x, y)$.

Определение 6.1. *Коммуникационный протокол* для функции $f : X \times Y \to Z$ — это корневое двоичное дерево, которое описывает совместное вычисление Алисой и Бобом функции f. В этом дереве каждая внутренняя вершина v помечена меткой А или Б, означающей очередь хода Алисы или Боба соответственно. Для каждой вершины, помеченной А, определена функция $g_v : X \to \{0, 1\}$, которая говорит Алисе, какой бит нужно послать, если вычисление находится в этой вершине. Аналогично, для каждой вершины v с пометкой Б определена функция $h_v : Y \to \{0, 1\}$, которая определяет бит, который Боб должен отослать в этой вершине. Каждая внутренняя вершина имеет двух потомков, ребро к первому потомку помечено 0, а ребро ко второму потомку помечено 1. Каждый лист помечен значением из множества Z.

Вычисление по такому протоколу на конкретной паре входов (x, y) устроено так: изначально вычисление находится в корне. В каждой внутренней вершине v в зависимости от пометки либо Алиса, либо Боб пересылают один бит (он определяется соответствующей функцией g_v или h_v). После этого вычисление переходит в один из потомков вершины v по ребру, пометка которого совпадает с битом, переданным в вершине v. Когда вычисление приходит в лист, то оно завершается. Результат вычисления — это пометка в листе.

Будем говорить, что коммуникационный протокол *вычисляет функцию* f, если для всех пар $(x, y) \in X \times Y$ вычисление приходит в лист с пометкой $f(x, y)$. Теперь можно дать формальное определение *коммуникационной сложности функции* f.

Аналогичным образом можно определить *коммуникационный протокол, вычисляющий отношение* $R \subset (X \times Y) \times Z$ — нужно только дополнительно потребовать, чтобы ответы Алисы и Боба были согласованы.

Определение 6.2. *Коммуникационная сложность* функции f определяется как наименьшая глубина протокола (максимальная рёберная длина пути от корня до листа), вычисляющего функцию f. Обозначается $D(f)$.

Утверждение 6.1. *Для любой* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, $D(f) \leq n + 1$.

Доказательство. Алиса посылает Бобу свой вход, а Боб посылает Алисе значение f. $\square$

Пример 6.1. Примеры функций с нетривиальной верхней оценкой на коммуникационную сложность.

1. (Pointer Chaising) $D(\mathrm{PC}) \leq k \log n$, где $PC(x, y) = \underbrace{x(y(x(y(x(y(x(y(x}_{k \text{ раундов}}(0)))))))))$.

   У игроков есть двудольный ориентированный граф на $2n$ вершинах, у которого исходящая степень каждой вершины равна 1. Алиса знает левую долю, Боб — правую. В начале они кладут фишку на вершину с номером 0 из доли Алисы и начинают передвигать её по рёбрам. Всего они должны сделать k переходов по рёбрам графа. Ответ — номер финальной вершины.

2. $D(\mathrm{MED}) = O(\log^2 n)$, где x и y интерпретируются как характеристические функции подмножеств $[n]$, а $\mathrm{MED}(x, y)$ — медиана их объединения. (Можно показать, что $D(\mathrm{MED}) = \Theta(\log n)$.)

3. $D(CIS_G) = O(\log^2 n)$, где x интерпретируется как характеристическая функция некоторой клики в графе G, а y — как характеристическая функция некоторого независимого множества в графе G. $\mathrm{CIS}(x, y) = 1$, если клика и независимое множество имеют общую вершину. (Замечание: не известно графов G, для которых нельзя решить эту задачу за $O(\log n)$.)

6.1. Нижние оценки

Рассмотрим коммуникационный протокол для некоторой функции $f : X \times Y \to Z$. Для каждой вершины v определим множество $R_v \subset X \times Y$ — множество всех пар $(x, y) \in X \times Y$, для которых вычисление приходит в вершину v.

Утверждение 6.2. *Для всех вершин v множество R_v является комбинаторным прямоугольником, т.е. существуют такие $X_v \subset X$ и $Y_v \subset Y$, что $R_v = X_v \times Y_v$.*

Доказательство. Покажем по индукции. Это верно для корня. Если это верно для какой-то вершины v с пометкой А: $R_v = X_v \times Y_v$. Если Алиса пересылает бит b и вычисление переходит в вершину u, то $R_u = X_u \times Y_u$, где $X_u = \{x \in X_v \mid g_v(x) = b\}$, а $Y_u = Y_v$. Аналогично, если Боб посылает бит b и вычисление переходит в вершину u, то $R_u = X_u \times Y_u$, где $X_u = X_v$, а $Y_u = \{y \in Y_v \mid h_v(y) = b\}$. □

Следствие 6.1. *Листья коммуникационного протокола для функции f задают разбиение множества $X \times Y$ на одноцветные прямоугольники.*

Будем обозначать $C^R(f)$ — минимальное количество *одноцветных* прямоугольников, покрывающих $X \times Y$.

Утверждение 6.3. $D(f) \geq \log C^R(f)$.

Доказательство. $D(f) \geq \log(\#\text{ листьев}) \geq \log C^R(f)$. □

Метод размера прямоугольников. Определим некоторую полуаддитивную меру на подпрямоугольниках $X \times Y$. Тогда верна следующая оценка

$$C^R(f) \geq \frac{w(X \times Y)}{\max\limits_{\text{одноцв. } A \times B} w(A \times B)}.$$

Метод трудного множества (fooling set). Это частный случай метода размера прямоугольников, при котором фиксируется некоторое множество $F \subset X \times Y$, а мера w определяется следующим образом:

$$w(A \times B) = |(A \times B) \cap F|.$$

Если при этом никакой одноцветный прямоугольник не содержит более одного элемента из F, тогда $C^R(f) \geq |F|$.

Метод ранга матрицы. Рассмотрим *матрицу функции* f — матрицу, в которой строки индексированы элементами X, столбцы — элементами Y, а в ячейке (x, y) стоит $f(x, y)$. Если мы рассмотрим эту матрицу функции как матрицу M над некоторым довольно большим полем, то можно показать, что $C^R(f) \geq \operatorname{rank} M$.

Упражнение 6.1. Докажите предыдущие утверждения.

Утверждение 6.4. $D(\mathrm{EQ}) = n + 1$, *где* $\mathrm{EQ}(x, y) = 1 \iff x = y$.

Утверждение 6.5. $D(\mathrm{GE}) = n + 1$, *где* $\mathrm{GE}(x, y) = 1 \iff x \geq y$.

6.2. Вероятностные протоколы

Можно рассмотреть коммуникационную игру, в которой у участников есть возможность использовать случайные биты. Можно формализовать это следующим образом: Алиса на вход получает пару (x, r), где $x \in X$, а r — случайная строка, аналогично, Боб получает пару (y, s), где $y \in Y$, а s — случайная строка. Функции g_v и h_v, записанные в вершинах протокола для такой игры, будут принимать два аргумента — вход и случайную строку, т.е. пересылаемые сообщения могут зависеть от случайных битов. Соответственно, результат игры будет зависеть от x, y, r, s.

Определение 6.3. Будем говорить, что вероятностный протокол ϵ-вычисляет f, если для любой пары x, y с вероятностью (по выбору (r, s)) не менее $1-\epsilon$ результат протокола равен $f(x, y)$ (с точки зрения обоих игроков). Через $R^\epsilon(f)$ обозначается минимальная высота вероятностного протокола ϵ-вычисляющего f.

Упражнение 6.2. Докажите, что $R^\epsilon(\mathrm{EQ}_n) = O(\log n + \log(1/\epsilon))$.

Упражнение 6.3. Докажите, что $R^\epsilon(\mathrm{GE}_n) = O(\log n(\log n + \log(1/\epsilon)))$.

Упражнение 6.4. Докажите, что если Алиса и Боб имеют доступ к общему источнику случайности (то есть $r = s$), то они могут ϵ-вычислить предикат EQ_n, передав $O(log(1/\epsilon))$ бит.

Упражнение 6.5. Докажите, что если Алиса и Боб имеют доступ к общему источнику случайности, то для любого фиксированного положительного ϵ они могут ϵ-вычислить предикат GE_n с ошибкой не более ϵ, передав $O(\log n)$ бит.

6.3. Связь протоколов и формул

Определение 6.4. *Игра Карчмера — Вигдерсона для функции* $f : \{0,1\}^n \to \{0,1\}$ — это следующая коммуникационная игра: Алиса получает $x \in f^{-1}(0)$, Боб получает $y \in f^{-1}(1)$, и они вместе пытаются найти такое $i \in [n]$, что $x_i \neq y_i$. Другими словами, игра Карчмера — Вигдерсона — это коммуникационная задача для *отношения*

$$R_f = \{((x, y), i) \mid x \in f^{-1}(0), y \in f^{-1}(1), x_i \neq y_i\}.$$

Отношение R_f будем называть *отношением Карчмера — Вигдерсона* для функции f.

Определение 6.5. *Формула в базисе Де Моргана для функции* $f : \{0,1\}^n \to \{0,1\}$ — это булевая формула с переменными $\{x_1, x_2, \ldots, x_n\}$, соответствующим отдельным битам входа f, и со связками $\{\wedge, \vee, \neg\}$, вычисляющая функцию f. Законы Де Моргана позволяют нам предполагать, что все $\neg$ находятся непосредственно перед переменными. Заметим, что структура формулы Де Моргана представляет собой корневое дерево (листья соответствуют переменным, а внутренние вершина — логическим связкам).

Будем называть *формульной сложностью* $L(f)$ функции $f : \{0,1\}^n \to \{0,1\}$ — это размер (количество вхождений переменных) минимальной формулы вычисляющей f. Если говорить более формально, то нужно говорить не о конкретной функции, а о последовательности функций.

Определение 6.6. Для функции $f : \{0,1\}^* \to \{0,1\}$ определим последовательность функций $\{f_1, f_2, \ldots, f_n, \ldots\}$, где $f_i : \{0,1\}^i \to \{0,1\}$ и $\forall x \in \{0,1\}^i, f(x) = f_i(x)$. Тогда формульная сложность $L(f)$ функции f ограничена $g(n)$, если для любого n существует формула ϕ_n размера не более $g(n)$, вычисляющая функцию f_n.

Теорема 6.1 (Шеннон)**.** *Существует f: $L(f) = \Omega(2^n/n)$.*

Доказательство. Пусть $n \geq 2$. Посчитаем количество формул размера не более s (здесь под размером формулы будем понимать количество вершин в дереве, соответствующем формуле). Пронумеруем вершины дерева по уровням от корня к листьям (корень будет иметь номер 1, потомки корня — номера 2 и 3, и т.д.). Теперь для каждой вершины в этом порядке запишем её краткое описание: для внутренних вершин описание будет операция в вершине (либо $\wedge$, либо $\vee$), для листьев с пометкой x_i запишем $(i, +)$, для листьев с пометкой $\neg x_i$ запишем $(i, -)$. В результате получится последовательность из s элементов, по которой можно восстановить исходную формулу. Различных последовательностей такого вида не более $(3n)^s$. В то же время число всех функций $f : \{0,1\}^n \to \{0,1\}$ ровно 2^{2^n}. Каким должно быть s, чтобы количество различных формул было достаточным, чтобы вычислить все функции на n битах?

$$(3n)^s \geq 2^{2^n} \implies s \cdot \log(3n) \geq 2^n \implies s = \Omega(2^n/n).$$

Так как формула задаёт двоичное дерево, то количество вершин и количество листьев (количество вхождений переменных) отличаются только в два раза. □

Замечание 6.1. Этот подсчёт показывает, что существуют функции с экспоненциальной формульной сложностью. Более того, любая случайная функция с большой вероятностью имеет такую сложность. Однако не известно *явных* функций большой сложности. Лучшая известная на данный момент нижняя оценка на формульную сложность явной функции это $\Omega(n^3)$ (оценка для функции Андреева, доказана Хостадом).

Теорема 6.2 (Карчмера, Вигдерсон, 1988)**.** *Для каждой формулы ϕ вычисляющей f, существует такой протокол Π_ϕ для отношения Карчмера — Вигдерсона R_f, что его дерево совпадает с деревом, описывающим структуру формулы ϕ. Верно и обратное: если есть протокол для R_f, то есть и формула для f с такой же структурой.*

Доказательство. Ход Алисы будет соответствовать связке $\wedge$, ход Боба — связке $\vee$.

- **формула $\to$ протокол**
  Каждая внутренняя вершина протокола соответствует некоторой подформуле исходной формулы ϕ. Будем поддерживать следующий инвариант: пусть ϕ_v — подформула, соответствующая текущей вершине протокола v, тогда $\phi_v(x) = 0$, а $\phi_v(y) = 1$. Это верно для начальной вершины (т.к. верно для ϕ). Если для текущей вершины это верно, и $\phi_v = \phi_{v0} \wedge \phi_{v1}$, то Алиса пересылает бит b такой, что $\phi_{vb}(x) = 0$ (такой бит должен быть по свойствам $\wedge$, т.к. $\phi_v(x) = 0$). При этом мы знаем, что $\phi_v(y) = \phi_{v0}(y) = \phi_{v1}(y) = 1$, т.е. инвариант сохраняется. Аналогично,

если $\phi_v = \phi_{v0} \vee \phi_{v1}$, то Боб пересылает бит b такой, что $\phi_{vb}(y) = 1$ (мы соответственно знаем, что $\phi_v(x) = \phi_{v0}(x) = \phi_{v1}(x) = 0$). Когда Алиса и Боб придут в некоторый лист, то по индукции получается, что значение в этом листе на входе Алисы отличается от значения в листе на входе Боба, а значит номер переменной в листе соответствует номеру бита различия.

- **протокол → формула**

  Будем последовательно строить формулы для внутренних вершин протокола от листьев к корню. При этом будем поддерживать следующий инвариант: пусть v — вершина протокола, $X_v \times Y_v$ — соответствующий прямоугольник, тогда формула ϕ_v для вершины v такая, что для всех $x \in X_v$, $\phi_v(x) = 0$ и для всех $y \in Y_v$, $\phi_v(y) = 1$. Пусть мы построили формулы ϕ_{v0} и ϕ_{v1} для сыновей некоторой вершины v. Если вершина v соответствовала ходу Алисы, то для всех входов Алисы из множества X_v формула ϕ_v должна быть равна 0. При этом по индукционному предположению мы знаем, что для некоторых входов Алисы (на которых Алиса посылает 0) $\phi_{v0} = 0$, а для остальных обязательно $\phi_{v1} = 0$. С другой стороны для всех входов Боба $y \in Y_v$, $\phi_{v0}(y) = \phi_{v1}(y) = 1$. Поэтому, если мы положим $\phi_v = \phi_{v0} \wedge \phi_{v1}$, то инвариант сохранится. Аналогично, если вершина соответствовала ходу Боба, то следует положить $\phi_v = \phi_{v0} \vee \phi_{v1}$.

  Осталось объяснить, что мы будем делать с листьями. Заметим, что если в листе протокола написан некоторый индекс i, то в него могут попадать либо пары входов, для которых $(x_i = 0,\ y_i = 1)$, либо входы, для которых $(x_i = 1,\ y_i = 0)$, но не могут попадать одновременно. В противном случае можно было бы воспользоваться свойствами комбинаторных прямоугольников и дать Алисе и Бобу входы с одинаковыми i-ми битами, которые привели бы в этот же лист.

$$\begin{cases} (x, y) \in R_\ell, & x_i = 0, y_i = 1, \\ (x', y') \in R_\ell, & x'_i = 1, y'_i = 0. \end{cases} \implies (x', y) \in R_\ell.$$

  Таким образом можно считать, что в каждом листе кроме номера бита различия записаны также значения этого бита у Алисы и у Боба. Если в листе ℓ с номером бита различия i записаны $(x_i = 0,\ y_i = 1)$, то $\phi_\ell = x_i$, в обратном случае $\phi_\ell = \neg x_i$.

$\square$

Таким образом мы получили соответствие между протоколами и формулами, сохраняющее структуру. Проблема в том, что сложность протоколов мы до этого измеряли в терминах максимальной глубины, а сложность формул — в терминах количества листьев. Давайте определим сложность протокола в терминах количества листьев.

Определение 6.7. Для отношения R_f будем обозначать через $L(R_f)$ минимальное количество листьев в коммуникационном протоколе для R_f.

Следствие 6.2. *Для любой функции f, $L(f) = L(R_f)$.*

С некоторыми потерями можно связать минимальный размер формулы для f с минимальной глубиной формулы для f.

Утверждение 6.6 ([7]). *Для любой $\alpha > 1$ такая, что для любой формулы ϕ размера s существует эквивалентная формула ϕ' размера s^α и глубины $O(\log s)$ (константа зависит от α).*

Мы докажем более слабое утверждение для конкретного $\alpha \approx 4$.

Доказательство. Определим рекурсивный алгоритм $A(\phi)$: найдём в ϕ подформулу ψ размера от $s/3$ до $2s/3$. Вернём $\phi' = (A(\psi) \wedge A(\phi|_{\psi=1})) \vee (\neg A(\psi) \wedge A(\phi|_{\psi=0}))$. Глубина рекурсии получится $\log_{3/2}(s)$, на каждой итерации глубина увеличивается на два. Суммарная глубина $2 \cdot \log_{3/2}(s)$. Таким образом размер формулы ϕ' не более $2^{2\cdot\log_{3/2}(s)} = O(s^4)$. $\square$

Определение 6.8. Пусть μ это некоторое распределение на входах Алисы и Боба, а X, Y — соответствующие случайные величины. *Внешнее информационное разглашение* протокола Π на распределении μ:

$$\mathrm{IC}_\mu^{ext}(\Pi) = I(\Pi(X,Y) : X, Y).$$

Внутреннее информационное разглашение протокола Π на распределении μ:

$$\mathrm{IC}_\mu^{int}(\Pi) = I(\Pi(X,Y) : X \mid Y) + I(\Pi(X,Y) : Y \mid X).$$

Лемма 6.1. *Для любого протокола Π и любого распределения μ*

$$D(\Pi) \geq \mathrm{IC}_\mu^{ext}(\Pi) \geq \mathrm{IC}_\mu^{int}(\Pi).$$

Доказательство. Первое неравенство тривиально (нельзя раскрыть больше информации, чем количество переданных битов).

Второе неравенство можно свести к утверждению 4.11. Для начала распишем взаимную информацию через энтропию.

$$\mathrm{IC}_\mu^{ext}(\Pi) = I(\Pi(X,Y) : X, Y) = H(\Pi(X,Y)) - H(\Pi(X,Y) \mid X, Y) = H(\Pi(X,Y)).$$

Последнее равенство имеет место, т.к. протокол детерминированный и $\Pi(X,Y)$ полностью определяется значениями X и Y. Аналогично, получаем

$$\mathrm{IC}_\mu^{int}(\Pi) = H(\Pi(X,Y) \mid Y) + H(\Pi(X,Y) \mid X).$$

Осталось убедиться, что $a = \Pi(X,Y)$, $x = X$, $y = Y$ удовлетворяют условиям утверждения 4.11, а следовательно

$$H(\Pi(X,Y)) \geq H(\Pi(X,Y) \mid Y) + H(\Pi(X,Y) \mid X).$$

$\square$

Теорема 6.3 ([8])**.** *Пусть* Π *коммуникационный протокол. Для любого распределения* μ*:* $\log L(\Pi) \geq \mathrm{IC}_\mu^{ext}(\Pi)$*. Кроме того существует такое распределение* μ^* *для которого* $\log L(\Pi) = \mathrm{IC}_{\mu^*}^{ext}(\Pi)$*. Будем называть* μ^* труднейшим *распределением для* Π*.*

Доказательство. Для детерминированных протоколов $\mathrm{IC}^{ext}(\Pi) = H_\mu(\Pi)$. Первое утверждение теоремы следует из верхней оценки на энтропию (энтропия случайной величины не превосходит логарифм числа исходов):

$$\mathrm{IC}_\mu^{ext}(\Pi) = H_\mu(\Pi) \leq \log L(\Pi).$$

Для доказательства второго утверждения мы предъявим распределение μ^*: выберем (равномерно) случайный лист l протокола Π и в соответствующем прямоугольнике R_l выберем произвольную пару (x, y). Полученное распределение μ^* равномерно на листьях Π, поэтому

$$\mathrm{IC}_{\mu^*}^{ext}(\Pi) = H_{\mu^*}(\Pi) = \log L(\Pi).$$

$\square$

Следствие 6.3. *Пусть* f *— булевая функция,* $s \in \mathbb{N}$*.* $L(f) \geq s$ *тогда и только тогда, когда для любого протокола* Π *для* R_f *существует распределение* μ*:* $\mathrm{IC}_\mu^{ext}(\Pi) \geq \log s$*.*

Теорема 6.4 (Храпченко)**.** $L(\oplus_n) \geq n^2$*.*

Доказательство. Покажем, что для любого протокола существует распределение μ: $\mathrm{IC}_\mu^{ext}(\Pi) \geq 2\log n$. Из этого напрямую следует, что $L(\oplus_n) \geq n^2$. Распределение μ будет равномерным распределением на парах вида $(x, x \oplus e_i)$, где $\oplus_n(x) = 0$, а строка e_i имеет единицу в позиции i и нули во всех остальных. Т.е., пары входов из распределения μ всегда будут отличаться только в одом бите.

$$\mathrm{IC}_\mu^{ext}(\Pi) \geq \mathrm{IC}_\mu^{int}(\Pi) = I(\Pi : X \mid Y) + I(\Pi : Y \mid X).$$

Рассмотрим однои из слагаемых $I(\Pi : X \mid Y)$.

$$\begin{aligned} I(\Pi : X \mid Y) &= H(X \mid Y) - H(X \mid Y, \Pi) \\ &= H(i \mid Y) - H(i \mid Y, \Pi) \\ &= \log n - 0. \end{aligned}$$

Таким образом $\mathrm{IC}_\mu^{ext}(\Pi) \geq 2\log n$. $\square$

Упражнение 6.6. Докажите, что для любой булевой функции f и любого распределения μ существует протокол Π для R_f: $\mathrm{IC}_\mu^{int}(\Pi) \leq 2\log n$.

Упражнение 6.7. Будем называть *универсальным отношением* для строк длины n отношение $U_n = \{(x, y, i) \mid x, y \in \{0,1\}^n, x_i \neq y_i\}$ (это обобщение понятия отношения Карчмера — Вигдерсона). Будем называть *расширенным универсальным отношением* для строк длины n отношение $U_n' = U_n \cup \{(x, x, \perp) \mid x \in \{0,1\}^n\}$ (решая коммуникационную задачу для расширенного универсального отношения Алиса и Боб могут получить *одинаковые* строки и тогда они должны ответить $\perp$).

Докажите следующие утверждения:

1. $4 \cdot L(U_n) \geq L(U'_n) \geq L(U_n)$.
2. $L(U'_n) \geq 2^n$.

Упражнение 6.8. Пусть $f : \{0,1\}^n \to \{0,1\}$ некоторая булева функция. Определим функцию $(\vee_m \circ f) : \{0,1\}^{m \times n} \to \{0,1\}$ следующим образом:

$$(\vee_m \circ f)(x_1, x_2, \ldots, x_m) = f(x_1) \vee f(x_2) \vee \cdots \vee f(x_m),$$

где $x_i \in \{0,1\}^n$ (т.е. мы определили композицию функция $\vee_m$ и f). Докажите, что $L(\vee_m \circ f) = m \cdot L(f)$.

7. Алгоритмический подход

7.1. Колмогоровская сложность

Сколько информации в первых 10^{10} знаках числа π? Её довольно мало, но сжать такое количество цифр, например, кодированием Хаффмана, не получится.

Определение 7.1. Частичная функция $f : \{0,1\}^* \to \{0,1\}^*$ называется *вычислимой*, если существует программа P:

- для $\forall x \in \operatorname{dom} f$: $P(x)$ печатает $f(x)$,
- для $\forall x \notin \operatorname{dom} f$: $P(x)$ не останавливается.

Определение 7.2. Пусть $F : \{0,1\}^* \to \{0,1\}^*$ — вычислимая функция. *Сложность описания относительно* F определяется как

$$K_F(x) = \min\{|p| : F(p) = x\}.$$

Определение 7.3. Будем говорить, что способ описания F *не хуже* G, обозначается $F \prec G$, если существует константа c_G такая, что для $\forall x \in \{0,1\}^*$

$$K_F(x) \leq K_G(x) + c_G.$$

Теорема 7.1 (Соломонова-Колмогорова)**.** *Существует способ описания (вычислимая функция) F такой, что для любого другого способа описания G выполняется $F \prec G$.*

Докажем сначала более простое утверждение.

Утверждение 7.1. *Пусть F и G — два способа описания. Тогда существует способ описания H такой, что $H \prec F$ и $H \prec G$.*

Доказательство. Определим H следующим образом: $H(0x) = F(x)$, $H(1x) = G(x)$ (если на каком-то входе x значение $F(x)$ или $G(x)$ не определено, то и H не определено на соответствующем входе $0x$ или $1x$). Тогда легко проверить, что для любых x верно $K_H(x) \leq K_F(x) + 1$ и $K_H(x) \leq K_G(x) + 1$. □

Доказательство теоремы 7.1. Пронумеруем все программы натуральными числами (программ счётное число). Пусть F_N — это программа с номером N (для машин Тьюринга N называется *номером Гёделя*). Рассмотрим функцию $U(\langle N, x\rangle) = F_N(x)$, где пара $\langle N, x\rangle$ закодирована следующим образом $\underbrace{11\dots1}_{N}0x$. Тогда

$$K_U(x) \leq K_{F_N}(x) + N + 1.$$

(Для машин Тьюринга U — это универсальная машина Тьюринга.) □

Определение 7.4. Будем называть $K(x) = K_U(x)$ *Колмогоровской сложностью* x.

Лемма 7.1. *Колмогоровская сложность обладает следующими свойствами.*

1. *Существует c такая, что для всех x $K(x) \leq |x| + c$.*
2. *Существует c такая, что для всех x $K(xx) \leq |x| + c$.*
3. *Для любых оптимальных F_1 и F_2 выполняется $F_1 \prec F_2$ и $F_2 \prec F_1$, т.е. существует такая константа c, что $|K_{F_1} - K_{F_2}| \leq c$.*

Доказательство. Третье свойство следует из определения. Докажем первые два.

1. Рассмотрим $H(x) = x$. Тогда $K(x) \leq K_H(x) + c = |x| + c$.
2. Рассмотрим $H(p) = pp$. Тогда $K(xx) \leq K_H(xx) + c = |x| + c$.

□

Вопрос: может быть такая длина n, что для всех $x \in \{0,1\}^n$ $K(x) < n$.

Утверждение 7.2. *Для любого n существует $x \in \{0,1\}^n$ такой, что $K(x) \geq n$ (т.е. x — несжимаемый).*

Доказательство. Слов длины n всего 2^n. Слов сложности меньше n не больше, чем программ длины меньше n: $1 + 2 + \dots + 2^{n-1} = 2^n - 1 < 2^n$. □

Утверждение 7.3. *Существует $c > 0$ такое, что для 99% слов длины n:*

$$n - c \leq K(x) \leq n + c = |x| + c.$$

Доказательство. Второе неравенство мы уже доказали. Первое неравенство следует из того, что программ длины не более $n - c$ всего $1 + 2 + \dots + 2^{n-c} \leq 2^{n-c+1}$, т.е. доля слов такой сложности не может быть больше 2^{-c+1}. При $c = 11$ эта доля меньше 0.1%. □

Утверждение 7.4. *Не существует вычислимой функции $f : \{0,1\}^* \to \{0,1\}^*$, которая была бы всюду определена и $f(\bar{n}) = x_n$, где $K(x_n) \geq n$ ($\bar{n}$ означает двоичную запись числа n).*

Доказательство. С одной стороны сложность x_n большая, с другой стороны мы можем описать x_n при помощи $\log n$ битов.

$$n \le K(x_n) \le K_f(x_n) + O(1) \le \log n + O(1).$$

□

Замечание 7.1. Это утверждение можно усилить, заменив „всюду определена“ на „определена для бесконечного числа входов“. Доказательство останется тем же.

Следствие 7.1. *Отображение $x \to K(x)$ не является вычислимым.*

Замечание 7.2. У этого факта есть довольно простое доказательство основанное на парадоксе Берри. Этот парадокс состоит в предложении рассмотреть

наименьшее натуральное число, которое нельзя определить
фразой из не более чем четырнадцати русских слов.

Эта фраза содержит четырнадцать слов и определяет то самое наименьшее число, отсюда получаем противоречие. Аналогично, в предположении, что такое отображение является вычислимым, первую строку x для которой $K(x) \ge n$ мы можем описать при помощи $\log n$ битов.

Следствие 7.2. *Оптимальный способ описания не является всюду определённой функцией.*

Следствие 7.3. *Пусть есть некоторая формальная теория, т.ч. в ней можно записать ‘$K(x) > c$’. Для всех достаточно больших c и для всех x формулы ‘$K(x) > c$’ недоказуемы (и при этом почти все эти утверждения истины).*

Доказательство. Если для любого c существует x такое, что ‘$K(x) > c$’ доказуемо, тогда перебирая все доказательства мы сможем по c построить x. □

Следствие 7.4. *Первая теорема Гёделя о неполноте.*

Замечание 7.3. Это кроме всего прочего даёт способ с хорошей вероятностью порождать недоказуемые утверждения.

Утверждение 7.5. *Пусть $x = \langle 011010010\ldots10110\rangle$ длины n содержит $p \cdot n$ единиц и $(1-p) \cdot n$ нулей, тогда*

$$K(x) \le \left(p \cdot \log\frac{1}{p} + (1-p)\cdot\log\frac{1}{1-p}\right)\cdot n + O(\log n).$$

Доказательство. Рассмотрим следующее описание:

$$\langle \text{количество '1', количество '0', номер перестановки с данным числом '1' и '0'} \rangle.$$

Всего перестановок

$$\binom{n}{pn} = 2^{\left(p\cdot\log\frac{1}{p}+(1-p)\cdot\log\frac{1}{1-p}\right)\cdot n+O(\log n)}.$$

Т.е. $K(x) \leq \left(p\cdot\log\frac{1}{p}+(1-p)\cdot\log\frac{1}{1-p}\right)\cdot n+O(\log n) = H(p)\cdot n+O(\log n)$. □

Замечание 7.4. В доказательстве важно кодировать эту тройку так, чтобы она однозначно разрезалась на три части. Можно, например, удвоить все биты первых компонент и добавить разделитель '01'.

7.2. Условная Колмогоровская сложность

Определение 7.5. Сложность *условного описания* x при условии y относительно F:

$$K_F(x \mid y) = \min\{|p| : F(p,y) = x\}.$$

Определение 7.6. Условное описание F *не хуже*, чем условное описание G, $F \prec G$, если существует c такая, что для любый x и y

$$K_F(x \mid y) \leq K_G(x \mid y) + c.$$

Теорема 7.2. *Существует оптимальный способ описания условного описания F такой, что для любого другого способа условного описания G выполняется $F \prec G$.*

Определение 7.7. Сложность оптимального описание x при условии y относительно оптимального способа условного описания $K(x \mid y)$ называется *условной Колмогоровской сложностью x при условии y.*

Утверждение 7.6. *Условная Колмогоровская сложность обладает следующими свойствами.*

1. $K(x \mid y) \leq K(x) + O(1)$.
2. $K(x \mid y) \leq |x| + O(1)$.
3. *Существует такая константа c, что для всех n, всех y для 99% слов x длины n выполняется $|K(x \mid y) - n| \leq c$.*
4. $K(x \mid x) = O(1)$.
5. *Пусть f — вычислимая функция. Тогда существует c_f такая, что для всех x $K(f(x) \mid x) \leq c_f$.*

7.3. Сложность пары

Будем обозначать сложность пары $K(x,y) = K(\langle x, y\rangle)$, где $\langle\cdot,\cdot\rangle$ — это произвольный вычислимый способ кодирования пар.

Утверждение 7.7. *Следующее утверждение* неверно*:*

$$\exists c\ \forall x, y\ K(x,y) \le K(x) + K(y \mid x) + c.$$

Доказательство. Докажем от обратного. Пусть $|x| + |y| = n$. Тогда

$$K(x,y) \le K(x) + K(y \mid x) + c \le |x| + |y| + 2\cdot O(1) + c = n + O(1).$$

С одной стороны различных пар всего $(n+1)\cdot 2^n$. С другой стороны из оценки на сложность следует, что различных описаний пар не может быть больше $2^{n+O(1)}$. $\square$

Теорема 7.3. $\forall x, y\ K(x,y) \le K(x) + K(y \mid x) + O(\log K(x,y))$.

Доказательство. Рассмотрим следующий способ кодирования пар: $\langle\overline{\overline{|p|}}01pq\rangle$, где $\overline{\overline{|p|}}$ — это двоичная запись $|p|$, в которой удвоен каждый бит. $\square$

Теорема 7.4 (Колмогорова-Левина)**.** $K(x,y) = K(x) + K(y \mid x) + O(\log K(x,y))$.

Определение 7.8. *Взаимная информация* x *и* y*:*

$$I(x:y) = K(y) - K(y \mid x),$$

$$I(y:x) = K(x) - K(x \mid y).$$

Таким образом теорема Колмогорова-Левина — это теорема о симметрии взаимной информации.

$$I(x:y) = K(x) + K(y) - K(x,y) + O(\log K(x,y)) = I(y:x).$$

Доказательство теоремы 7.4. Неравенство '$\le$' уже доказано. Осталось доказать '$\ge$'.

$$\underbrace{K(x)}_{m} + \underbrace{K(y \mid x)}_{l} \le \underbrace{K(x,y)}_{n} + \underbrace{O(\log K(x,y))}_{O(\log n)}.$$

Пусть $S = \{(a,b) \mid K(a,b) \le n\}$. Заметим, что $(x,y) \in S$ и $|S| \le 2^{n+1}$. Рассмотрим $S_x = \{(x,b) \mid (x,b) \in S\}$. По определению $(x,y) \in S_x$. Покажем, что

$$l = K(y \mid x) \le \log|S_x| + O(\log n).$$

Будем перечислять множество S. В процессе этого перечисления мы будем получать точки из S_x. Для того, чтобы задать y, нам нужно указать номер (x,y) в этом перечислении. Кроме того, чтобы такое перечисление запустить, нам нужно знать число n. Получается, что

$$|S_x| \ge 2^{l - c\cdot\log n} \ge 2^{l'},$$

где l' — ближайшее снизу целое, т.е. $l' = \lfloor l - c \cdot \log n \rfloor$.

Посмотрим ещё раз на перечисление S. В процессе перечисления у нас возникают „тяжёлые сечения“ — те, в которых число элементов хотя бы $2^{l'}$. Для того, чтобы задать сечение S_x, нам нужно задать его порядковый номер в перечислении S среди всех „тяжёлых сечений“. Таким образом

$$m = K(x) \leq \log(\#\ \text{тяжёлых сечений}) + O(\log n) + O(\log l').$$

Тяжёлых сечений не больше, чем $|S|/2^{l'}$.

$$m = K(x) \leq \log \frac{|S|}{2^{l'}} + O(\log n) = n - l + O(\log n).$$

Таким образом получаем утверждение теоремы: $m + l \leq n + O(\log n)$. □

Следствие 7.5. $|I(x:y) - I(y:x)| \leq O(\log K(x,y))$.

Замечание 7.5. Выберем n такое, что его двоичная запись несжимаема, т.е. $K(\bar{n}) = \log n + O(1)$. Возьмём $x \in \{0,1\}^n$ такой, что $K(x \mid \bar{n}) = n + O(1)$. Тогда

- $I(\bar{n} : x) = K(x) - K(x \mid \bar{n}) = n + O(1) - (n + O(1)) = O(1)$,
- $I(x : \bar{n}) = K(\bar{n}) - K(\bar{n} \mid x) = (\log n + O(1)) - O(1) = \log n + O(1)$.

Т.е. нельзя уменьшить логарифмический зазор в теореме Колмогорова-Левина.

Упражнение 7.1. $2K(x,y,z) \leq K(x,y) + K(x,z) + K(y,z) + O(\log n)$, при $n = |x| + |y| + |z|$.

Упражнение 7.2. $K(x,y,z) + K(z) \leq K(x,z) + K(y,z) + O(\log n)$, при $n = |x| + |y| + |z|$.

Упражнение 7.3. $K(z) \leq K(z \mid x) + K(z \mid y) + I(x:y) + O(\log n)$, при $n = |x| + |y| + |z|$.

7.4. Метод несжимаемых объектов

Определение 7.9. *Конечный автомат с несколькими головками* — это конечный автомат, у которого на каждом шаге функция перехода по внутреннему состоянию автомата и по символам, на которых находятся головки, возвращает состояние на следующем шаге и номера головок, которые нужно сдвинуть, и при этом на каждом шаге сдвигается хотя бы одна головка.

Определим класс $\mathcal{L}_k$ — класс языков, которые распознаются конечными автоматами с k головками.

Теорема 7.5. $\mathcal{L}_k \subsetneq \mathcal{L}_{k+1}$.

Определим следующее семейство языков над алфавитом $\{0, 1, \#\}$

$$A_n = \{w_1 \# w_2 \# \cdots \# w_n \# w_n \# \cdots \# w_1 \mid w_i \in \{0,1\}^*\},$$

где $w_i \in \{0,1\}^*$, $\forall i \in \{1, 2, \ldots, n\}$.

При $n = 1$ для языка $A_1 = \{w_1\#w_1\}$ нужно две головки (по лемме о накачке конечный автомат с одной головкой этот язык не распознать).

При $n = 3$ можно распознать с четырьмя головками:

$$\underset{\boxed{1}}{w_1}\#w_2\#w_3\#\underset{\boxed{2}}{w_3}\#\underset{\boxed{3}}{w_2}\#\underset{\boxed{4}}{w_1}.$$

Но можно обойтись и тремя головками (придумайте трюк):

$$\underset{\boxed{1}}{w_1}\#\underset{\boxed{2}}{w_2}\#w_3\#w_3\#\underset{\boxed{3}}{w_2}\#w_1.$$

Если использовать этот трюк для k головок, то можно было бы распознать язык A_n для $n \leq (k-1) + (k-2) + \cdots + 1$, т.е. $n = \frac{k\cdot(k-1)}{2}$. Таким образом конечный автомат с k головками распознаёт язык A_n для $n \leq \frac{k\cdot(k-1)}{2}$.

Лемма 7.2. *A_n не распознаётся конечным автоматом с k головками, если $n > \frac{k\cdot(k-1)}{2}$.*

Доказательство. Будем говорить, что пара головок (i, j) *инспектирует* w_ℓ, если найдётся шаг работы конечного автомата, когда i-ая головка читает символ левой копии w_ℓ, а j-ая головка читает символ правой копии w_ℓ.

Для любого $x \in A_n$ и для любой пары (i, j) существует не более одного блока w_ℓ такого, что пара (i, j) инспектирует w_ℓ. Если $n > k\cdot(k-1)/2$, то найдётся блок, который не инспектируется ни одной парой головок. Будем рассматривать некоторый $x \in A_n$ и предположим, что блок w_ℓ не инспектируется.

Замечание 7.6. Блок w_ℓ не инспектируется, поэтому, пока как какие-то головки находятся в левой копии w_ℓ, то в правой копии никакие головки находится не могут.

Запишем *протокол работы автомата на слове x с выделенным ℓ*. Будем записывать состояние автомата каждый раз, когда происходят следующие события:

- вход головки в копию w_ℓ,
- выход головки из копии w_ℓ.

Состояние автомата будет описываться внутренним состоянием автомата и позициями всех головок. Будем обозначать такой протокол $\pi(x, \ell)$.

Предположим, что для конкретного x

$$x = w_1\#w_2\#\cdots\#w_\ell\#\cdots\#w_n\#w_n\#\cdots\#w_\ell\#\cdots\#w_1,$$

конечный автомат не инспектирует блок ℓ. Рассмотрим вход x' с другим блоком w'_ℓ:

$$x' = w_1\#w_2\#\cdots\#w'_\ell\#\cdots\#w_n\#w_n\#\cdots\#w'_\ell\#\cdots\#w_1.$$

Утверждение 7.8. *Невозможно, что для x' блок ℓ тоже не инспектируется, и при этом протоколы равны $\pi(x, \ell) = \pi(x', \ell)$.*

Доказательство. Если протоколы равны, то автомат должен и допускать вход

$$x'' = w_1 \# w_2 \# \cdots \# w_\ell \# \cdots \# w_n \# w_n \# \cdots \# w'_\ell \# \cdots \# w_1.$$

Если какие-то головки находятся в w_ℓ, то автомат на x'' работает как на входе x. Если какие-то головки находятся в w'_ℓ, то автомат работает как на входе x'. Следовательно он должен принимать $x'' \notin A_n$. Таким образом мы пришли к противоречию. $\square$

Мы показали, что для разных x у нас должны быть разные протоколы. Таким образом зная ℓ и зная протокол мы можем восстановить w_ℓ — для этого нужно знать все остальные блоки и протокол. Наше наблюдение можно переписать следующим образом:

$$K(w_\ell \mid w_1, \ldots, w_{\ell-1}, w_{\ell+1}, \ldots, w_n, \ell, \pi(x, \ell)) = O(1).$$

Будем считать, что все блоки имеют длину N. Кроме того мы изначально потребуем, чтобы x был несжимаемым, т.е. $K(x) = K(w_1, w_2, \ldots, w_n) \geq n \cdot N$. Тогда

$$n \cdot N \leq K(w_1, \ldots, w_n) \leq \underbrace{(n-1) \cdot N}_{\{w_i\}_{i \neq \ell}} + \underbrace{O(\log n)}_{\ell} + \underbrace{4 \cdot k \cdot O(k \log nN)}_{\text{сложность } \pi(x,\ell)}.$$

При $N \to \infty$ мы получаем противоречие: $n \cdot N \leq (n-1)N + O(k^2 \log nN)$. $\square$

Доказательство теоремы 7.5. Язык $A_{\frac{k \cdot (k+1)}{2}}$ лежит в $\mathcal{L}_{k+1}$ и не лежит в $\mathcal{L}_k$. $\square$

7.5. Определение случайности

Если говорить о конечных последовательностях, то совершенно непонятно как провести границу между случайными и неслучайными последовательностями. Давайте попробуем дать формальное определение случайной бесконечной последовательности на языке Колмогоровской сложности. Какие свойства мы хотим от этого определения?

Давайте рассмотрим последовательность $\bar{x} = x_1 x_2 x_3 \ldots x_n \ldots$. Естественным было бы получить определение вида $\forall n\ K(x_1 x_2 x_3 \ldots x_n) \geq n - c$. Покажем, что для обычного определения Колмогоровской сложности такое определение не имеет смысла.

Утверждение 7.9. *Для любой последовательности $\bar{x} = x_1 x_2 x_3 \ldots x_n \ldots$ и существует n такое, что*

$$K(x_1, \ldots, x_n) \leq n - \log n + O(1).$$

(т.е. для любой c существует префикс, такой что $K(x_1, \ldots, x_n) \leq n - c$).

Доказательство. Возьмём некоторый префикс длины k и интерпретируем его как двоичную запись некоторого числа m (добавим ведущую единицу), и рассмотрим его продолжение длины m:

$$\underbrace{1x_1 x_2 x_3 \ldots x_k}_{\overline{m}} \underbrace{x_{k+1} \ldots x_{k+m}}_{y},$$

где $|y| = m$. Пусть $n = m + k$. Тогда утверждается, что

$$K(x_1 \dots x_{m+k}) \le K(y) + O(1) \le m + O(1) \le n - k + O(1) \le n - \log n + O(1).$$

Действительно, зная строку y можно определить $m = |y|$ и приписать $\overline{m}$ в начало без ведущей единицы. □

Определение 7.10. *Префиксная сложность x относительно F*:

$$KP_F(x) = \min\{|p| : F(p) = x\},$$

где F — это функция с (бес)префиксной областью определения, т.е. если определены $F(p_1)$ и $F(p_2)$, то $p_1 \not\sqsubseteq p_2$.

Определение 7.11. Беспрефиксный способ описания F *не хуже* беспрефиксного способа описания G, $F \prec G$, если $\exists c\ \forall x\ KP_F(x) \le KP_G(x) + c$.

Теорема 7.6. *Существует оптимальный способ беспрефиксного описания.*

Доказательство. Проблема: не все программы имеют беспрефиксную область определения. Можно преобразовать любую программу π_i в программу с беспрефиксной областью определения π_i' таким образом, чтобы

- если π_i вычисляла функцию F_i с беспрефиксной областью определения, то π_i' тоже вычисляет F_i,
- если π_i вычисляла что-то другое, то π_i' вычисляет некоторую функцию с беспрефиксной областью определения (область может быть пустой).

После этого воспользуемся конструкцией аналогичной теореме 7.1 (Соломонова-Колмогорова): $UP(\underbrace{11\dots1}_{n}0p) = \pi_n'(p)$.

Определим работу программы π_n': на входе p программа π_n' запускает параллельно программу π_n на всех входах:

$$\pi_n(0), \pi_n(1), \pi_n(00), \pi_n(01), \dots, \pi_n(p), \dots$$

Если в какой-то момент обнаруживается, что π_n имеет не беспрефиксную область определения, то $\pi'(p)$ зацикливается. Если же в какой-то момент $\pi(p)$ завершается и до этого не было обнаружено нарушение беспрефиксности, то $\pi'(p) = \pi(p)$. □

Определение 7.12. $KP(x) = KP_{UP}(x)$, префиксная сложность относительно UP, называется *префиксной Колмогоровской сложностью x*.

Упражнение 7.4. $KP(x, y) \le KP(x) + KP(y) + O(1)$.

Определение 7.13. Последовательность $\bar{x} = x_1 x_2 \dots x_n \dots$ называется *случайной по Мартин-Лёфу*, если $\exists c\ \forall n\ KP(x_1 \dots x_n) \ge n - c$.

Утверждение 7.10. *Префиксная Колмогоровская сложность обладает следующими свойствами*

- $KP(x) \leq K(x) + 2\log K(x) + O(1)$.
- $\sum_{x\in\{0,1\}^k} 2^{-KP(x)} \leq 1$.

Доказательство.

- $2\log K(x)$ возникает из-за преобразования строки p в беспрефиксную $p' = \overline{\overline{|p|}}01p$.
- Аналогично неравенству Крафта-Макмилана для префиксных кодов.

$\square$

Теорема 7.7. *Почти все последовательности $\bar{x} = x_1x_2\ldots x_n\ldots$ являются случайными по Мартину-Лёфу, т.е. неслучайные имеют меру 0 по мере Бернулли.*

Доказательство. Построим покрывающее множество

$$A_c = \bigcup_{KP(x_1\ldots x_n)\leq n-c} \Omega_{x_1\ldots x_n},$$

где $\Omega_p = \{$все последовательности продолжающие $p\}$. A_c покрывает все неслучайные по Мартину-Лёфу последовательности. Действительно, у любой неслучайной последовательности есть начало, задающее такое Ω_p. Какова мера A_c?

$$\mu(A_c) \leq \sum_{KP(x_1\ldots x_n)\leq n-c} 2^{-n} \leq \sum_{KP(x_1\ldots x_n)\leq n-c} 2^{-KP(x_1\ldots x_n)-c} \leq$$
$$\leq \sum_{x_1\ldots x_n} 2^{-KP(x_1\ldots x_n)-c} = 2^{-c}\cdot \sum_{x_1\ldots x_n} 2^{-KP(x_1\ldots x_n)} \leq 2^{-c}.$$

Таким образом по любому ε мы можем предъявить покрытие неслучайных по Мартин-Лёфу последовательностей счётным числом конусов. $\square$

Утверждение 7.11. *Выполняются следующие свойства случайных по Мартину-Лёфу последовательностей.*

- *Всякая случайная по Мартину-Лёфу последовательность невычислима.*
- *Если $\bar{x}$ случайная по Мартин-Лёфу, то*

$$\lim_{n\to\infty} \frac{\text{число единиц}}{n} = \frac{1}{2}.$$

Доказательство.

- Если $\bar{x}$ вычислима, то

$$KP(x_1 \dots x_n) \leq K(x_1 \dots x_n) + 2\log K(x_1 \dots x_n) \leq \log n + 2\log\log n + O(1).$$

- Используется оценка $K(x_1 \dots x_n) \leq H(p) \cdot n + O(\log n)$ из утверждения 7.5.

$\square$

Упражнение 7.5. Докажите, что следующие последовательности не являются случайными по Мартину-Лёфу:

- $x_1 0 x_3 0 x_5 0 \dots x_{2n+1} 0 x_{2n+3} \dots$,
- $x_1 x_1 x_2 x_2 x_3 x_3 \dots x_n x_n \dots$.

Теорема 7.8 (Закон больших чисел в форма Харди-Литлвуда)**.** *Для почти всех последовательностей $\bar{x} = x_1 x_2 \dots x_n \dots$ (с вероятностью 1)*

$$\left| \frac{x_1 + x_2 + \dots + x_n}{n} - \frac{1}{2} \right| = O\left(\sqrt{\frac{\ln n}{n}} \right).$$

Доказательство. Пусть в $x_1 \dots x_n$ всего $p_n \cdot n$ единиц и $(1 - p_n) \cdot n$ нулей.

$$KP(x_1 \dots x_n) \leq K(x_1 \dots x_n) + O(\log n) \leq H(p) \cdot n + O(\log n).$$

Пусть $p = \frac{1}{2} + \delta_n$. Разложим $H(p)$ в ряд в окрестности $\frac{1}{2}$:

$$H(1/2 + \delta_n) \cdot n = (1 - c_H \cdot \delta_n^2 + o(\delta_n^2)) \cdot n \leq (1 - c'_H \cdot \delta_n^2) \cdot n.$$

Таким образом для случайно последовательности (т.е. с вероятностью 1):

$$n - c \leq KP(x_1 \dots x_n) \leq n - c'_H \cdot \delta_n^2 \cdot n + O(\log n).$$

Получаем, что $\delta_n^2 \leq O\left(\frac{\log n}{n}\right)$. $\square$

Замечание 7.7. Более сильный закон больших чисел имеет оценку $(1+\varepsilon)\sqrt{\frac{2\log\log n}{n}}$.

8. Приложения Колмогоровской сложности

8.1. Бесконечность множества простых чисел

Теорема 8.1. *Простых чисел бесконечно много.*

Доказательство. Докажем от обратного. Пусть простых чисел всего m: $p_1, p_2, \ldots, p_m$. Тогда любое целое разлагается на степени этих простых:

$$x = p_1^{k_1} \cdot p_2^{k_2} \cdot \cdots \cdot p_m^{k_m},$$

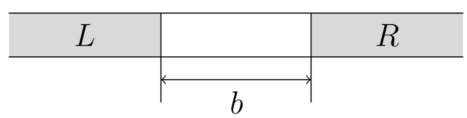
и тем самым определяется набором степеней $k_1, k_2, \ldots, k_m$. Каждое $k_i \leq \log x$, а значит записывается при помощи $O(\log\log x)$ битов. Число m является абсолютной константой, поэтому любое x задаётся при помощи $O(\log\log x)$ битов. В то же время случайное существуют n битовые числа x сложности не менее n. Отсюда получаем противоречие:

$$n \leq K(x) \leq O(\log\log x) = O(\log n).$$

□

8.2. Перенос информации по ленте

Мы докажем, что для копирования слова длины n на одноленточной машине Тьюринга необходимо $\Omega(n^2)$ шагов. Для этого давайте рассмотрим более общую задачу — задачу о переносе информации по ленте. Пусть на ленте выделена некоторая „буферная“ область ширины b.



Нас будет интересовать скорость переноса информации через „буферную зону“ слева направо, т.е. из области L в область R. Пусть изначально область R пуста. Какова может быть сложность строки R через t шагов после начала работы? Мы покажем, что сложность R не более $(t\log m)/b + O(\log t)$, где m — число состояний машины Тьюринга. Это можно объяснить из неформальных соображений: каждое состояние „несёт“ не более $\log m$ битов информации, за один шаг информация переносится на одну клетку, т.е. всего за t шагов мы перенесём не более $t\log m$ битов информации. Нам же нужно перенести информацию на расстояние b, отсюда получаем $(t\log m)/b$.

Теорема 8.2. *Пусть зафиксирована машина Тьюринга m состояниями. Тогда существует такая константа c, что для любого b и для любого вычисления с буферной зоной b (вначале эта зона и лента справа от неё пусты, головка машины находится слева от зоны) сложность правой части ленты $R(t)$ после t шагов вычисления не превосходит*

$$\frac{t\log m}{b} + 4\log t + c.$$

Доказательство. Проведём границу где-нибудь внутри буферной зоны, и при каждом пересечении головкой машины Тьюринга границы слева направо будем записывать, в каком состоянии она её пересекла. Будем называть такой протокол работы *следом* машины. Заметим, что по следу можно восстановить работу машины справа от границы —

действительно, поведение машины Тьюринга справа от границы зависит только от того состояния, в котором машина пересекла границу и от того, что уже к этому моменту записано на ленте справа от границы.

Для того, чтобы восстановить $R(t)$ нам потребуется указать след S, количество шагов $t' \leq t$, которое машина Тьюринга сделала справа от границы, а так же расстояние $b' < b$ от границы до R. Получаем

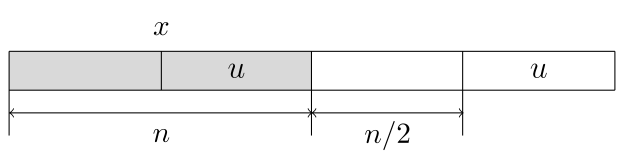
$$K(R(t)) \leq |\langle S, b', t' \rangle| + c \leq |S| \cdot \log m + 2\log b + 2\log t + c \leq |S| \cdot \log m + 4\log t + c.$$

Это неравенство верно для любого начального состояния и положения границы. Если для данного L мы выберем *самый короткий* из следов для всех возможных положений границ, то его заведомо длина меньше t/b (различных положений границы $b+1$, на каждом шаге пересекается только одна из возможных границ, таким образом сумма длин следов не превосходит t). Следовательно, наша оценка верна для $|S| < t/b$. □

Отсюда сразу же получается квадратичная нижняя оценка на копирование на одноленточной машине Тьюринга. Под копированием будем понимать следующий процесс: изначально на ленте написано некоторое слово $x \in \{0,1\}^*$, а справа от него лента пуста. В конце работы машины Тьюринга на ленте должно быть написано xx.

Теорема 8.3. *Существует такая константа $\epsilon > 0$, что для любого n существует слово длины n, копирование которого с помощью машины M занимает не менее ϵn^2 шагов.*

Доказательство. Для простоты будем предполагать, что n чётно. Возьмём в качестве x слово, у которого вторая половина u несжимаема (т.е. имеет сложность $\geq n/2$). Применим теорему о скорости переноса информации, считая буферной зоной участок длины $n/2$ справа от x.



Пусть копирование заняло t шагов, тогда сложность зоны R не меньшь $n/2$. С другой стороны, по предыдущей теореме сложность R не превосходит $(t \log m)/b + 4\log t + c$, где $b = n/2$. Получаем, что

$$\frac{n}{2} \leq \frac{t \log m}{b} + 4\log t + c.$$

Предположим, что $t < n^2$ (иначе нам нечего доказывать), а следовательно $4\log t < 8\log n$. Отсюда получаем, что

$$t \geq \frac{n^2}{4\log m} - O(n\log n).$$

От второго слагаемого можно избавиться, если взять ϵ немного меньше $1/(4\log m)$. □

8.3. Алгоритм сложения битовых чисел

Пусть $\bar{x} = \overline{x_{n-1}\dots x_0}$ и $\bar{y} = \overline{y_{n-1}\dots y_0}$ — это два n-битных числа. Предложим алгоритм сложения $\bar{x}$ и $\bar{y}$, который делает $\log n$ операций в среднем (предполагается, что побитовые операции с n-битными числами выполняются за O(1)).

Алгоритм будет устроен следующим образом.

- Первая итерация.
  Вычисляем $\bar{z}^{(1)}$: $z_i^{(1)} = x_i \oplus y_i$.
  Вычисляем $\bar{c}^{(1)}$: $c_i^{(1)} = x_{i-1} \wedge y_{i-1}$ (вектор переносов).
- Итерация $k+1$.
  Вычисляем $\bar{z}^{(k+1)}$: $z_i^{(k+1)} = z_i^{(k)} \oplus c_i^{(k)}$.
  Вычисляем $\bar{c}^{(k+1)}$: $c_i^{(k+1)} = z_{i-1}^{(k)} \wedge c_{i-1}^{(k)}$.

Итерации заканчиваются, если $\bar{c}^{(k)} = 0$.

На каком входе мы можем сделать t итераций? Утверждается, что это может произойти только в том случае, если в $\bar{x}$ и $\bar{y}$ есть непрерывные блоки длины t, соответствующие биты в которых противоположны, а после них стоит ’1’.

$$\begin{array}{cccccc} \bar{x}: & \boxed{} & b & \boxed{v_t\dots v_1} & 1 & \boxed{} \\ \bar{y}: & \boxed{} & b & \boxed{\bar{v}_t\dots \bar{v}_1} & 1 & \boxed{} \\ & & & & j & \end{array}$$

Так как у $\bar{x}$ и $\bar{y}$ есть общий блок битов длины t, то

$$K(\bar{x} \mid \bar{y}) \le (n-t-2) + \underbrace{\log n}_{t} + \underbrace{\log n}_{j} + O(\log\log n).$$

Отсюда $t \le n - K(\bar{x} \mid \bar{y}) + 2\log n + O(\log\log n)$. Среднее количество итераций в алгоритме можно оценить как

$$\sum_t t \cdot [\text{доля пар } (x,y) \text{ с общим блоком длины } t]$$

Введём обозначение $K(\bar{x} \mid \bar{y}) \le n - \underbrace{(t - 2\log n - O(\log\log n))}_{s}$ и будем называть s *дефектом случайности*, т.е. $s = t - 2\log n + O(\log\log n)$ и $t \le s + 2\log n + O(\log\log n)$.

Доля пар $(\bar{x}, \bar{y})$ таких, что $K(\bar{x} \mid \bar{y}) = n - s$ не больше 2^{-s+1}. Нас интересует асимптотическая оценка, поэтому мы можем считать, что

$$t \le s + 2\log n + O(\log\log n) \le s + 3\log n.$$

Тогда среднее количество итераций в алгоритме не больше, чем

$$\sum_s (s + 3\log n) \cdot 2^{-s+1} = 6\log n \sum_s 2^{-s} + 2\sum_s \frac{s}{2^s} = O(\log n) + O(1).$$

Тут важно заметить, что из $n - s \le n + O(1)$ следует $-s \le O(1)$.

8.4. Локальная лемма Ловаса

Пусть задан некоторый набор событий $\{A_1, A_2, \ldots, A_n\}$, и про каждое событие известна его вероятность $\Pr[A_i] = \varepsilon_i$. Какова вероятность того, что ни одно из этих событий не произойдёт? Есть два крайних случая.

- Если про природу событий $\{A_i\}_{i=1}^n$ ничего не известно, то
$$\Pr[\overline{A}_1 \wedge \overline{A}_2 \wedge \cdots \wedge \overline{A}_n] \geq 1 - \varepsilon_1 - \varepsilon_2 - \cdots - \varepsilon_n.$$

- Если все $\{A_i\}_{i=1}^n$ независимы в совокупности, то
$$\Pr[\overline{A}_1 \wedge \overline{A}_2 \wedge \cdots \wedge \overline{A}_n] = (1 - \varepsilon_1) \cdot (1 - \varepsilon_2) \cdot \ldots \cdot (1 - \varepsilon_n).$$

Локальная лемма Ловаса даёт оценку в промежуточном случае, когда зависимость между событиями *локальна*: каждое A_i зависимо только с относительно небольшим количеством *соседей*. Будем обозначать $N(i)$ — множество соседей события i.

Теорема 8.4 (Локальная лемма Ловаса)**.** *Пусть задано множество из n событий $\{A_1, A_2, \ldots, A_n\}$, в котором каждое событие A_i независимо со всеми событиями A_j, $j \notin N(i)$. Если для каждого i выбрано $\varepsilon_i < 1$ так, что*
$$\Pr[A_i] \leq \varepsilon_i \cdot \prod_{j \in N(i)} (1 - \varepsilon_j),$$
то вероятность того, что не произойдёт ни одного из событий
$$\Pr[\overline{A}_1 \wedge \overline{A}_2 \wedge \cdots \wedge \overline{A}_n] \geq (1 - \varepsilon_1) \cdot (1 - \varepsilon_2) \cdot \ldots \cdot (1 - \varepsilon_n).$$

Доказательство. Начнём с пары простых утверждений. По определению условной вероятности
$$\Pr[A \mid B] = \frac{\Pr[A \wedge B]}{\Pr[B]} \leq \frac{\Pr[A]}{\Pr[B]}.$$
Данное утверждение можно *„релятивизировать“*, т.е. добавить во все вероятности дополнительное условие C:
$$\Pr[A \mid B \wedge C] \leq \frac{\Pr[A \mid C]}{\Pr[B \mid C]}. \tag{5}$$

Будем доказывать теорему по индукции. Доказательство индукционного перехода будет состоять из доказательства двух утверждений.

1. Для любого i и множества $J = \{j_1, j_2, \ldots, j_k\} \subset [n]$, $i \notin J$:
$$\Pr[A_i \mid \overline{A}_{j_1} \wedge \overline{A}_{j_2} \wedge \cdots \wedge \overline{A}_{j_k}] \leq \varepsilon_i. \tag{6}$$

2. Для любых непересекающихся $I, J \subset [n]$, $I = \{i_1, i_2, \ldots, i_\ell\}$, $J = \{j_1, j_2, \ldots, j_m\}$
$$\Pr[\overline{A}_{i_1} \wedge \overline{A}_{i_2} \wedge \cdots \wedge \overline{A}_{i_\ell} \mid \overline{A}_{j_1} \wedge \overline{A}_{j_2} \wedge \cdots \wedge \overline{A}_{j_m}] \geq (1 - \varepsilon_{i_1}) \cdot (1 - \varepsilon_{i_2}) \cdot \ldots \cdot (1 - \varepsilon_{i_\ell}). \tag{7}$$

Связь этих двух утверждений мы сформулируем в виде следующих двух лемм.

Лемма 8.1. *Если неравенство* (6) *верно всех* $k \le t$, *то неравенство* (7) *верно для* $\ell + m \le t + 1$.

Доказательство. Раскроем вероятность в левой части неравенства (7) по „релятивизированному“ (т.е. с дополнительным условием) определению условной вероятности $\Pr[A \wedge B \mid C] = \Pr[A \mid B \wedge C] \cdot \Pr[B \mid C]$ (это формулу нужно будет применить k раз):

$$\begin{aligned}
\Pr[\overline{A}_{i_1} \wedge \overline{A}_{i_2} \wedge \cdots \wedge \overline{A}_{i_\ell} \mid \overline{A}_{j_1} \wedge \overline{A}_{j_2} \wedge \cdots \wedge \overline{A}_{j_m}] = {} & \Pr[\overline{A}_{i_1} \mid \overline{A}_{i_2} \wedge \cdots \wedge \overline{A}_{i_\ell} \wedge \overline{A}_{j_1} \wedge \cdots \wedge \overline{A}_{j_m}] \\
& \cdot \Pr[\overline{A}_{i_2} \mid \overline{A}_{i_3} \wedge \cdots \wedge \overline{A}_{i_\ell} \wedge \overline{A}_{j_1} \wedge \cdots \wedge \overline{A}_{j_m}] \\
& \quad \vdots \\
& \cdot \Pr[\overline{A}_{i_\ell} \mid \overline{A}_{j_1} \wedge \cdots \wedge \overline{A}_{j_m}] \\
\ge {} & (1 - \varepsilon_{i_i}) \cdot (1 - \varepsilon_{i_2}) \cdot \ldots \cdot (1 - \varepsilon_{i_\ell}).
\end{aligned}$$

□

Лемма 8.2. *Если неравенство* (7) *верно для всех* ℓ *и* m, *таких что* $\ell + m = t$, *то неравенство* (6) *верно для* $k = t$.

Доказательство. Если $J \cap N(i) = \emptyset$, то неравенство (6) выполняется, т.к. A_i независимо в совокупности с $A_{i_1}, A_{i_2}, \ldots, A_{i_k}$. Иначе введём следующие обозначения:

$$N = \bigwedge_{j \in J \cap N(i)} \overline{A}_j, \quad F = \bigwedge_{j \in J \setminus N(i)} \overline{A}_j.$$

В этих обозначениях левая часть неравенства (6) переписывается следующим образом:

$$\Pr[A_i \mid N \wedge F] \le \frac{\Pr[A_i \mid F]}{\Pr[N \mid F]} = \frac{\Pr[A_i]}{\Pr[N \mid F]} \le \frac{\varepsilon_i \cdot \prod_{j \in N(i)} (1 - \varepsilon_j)}{\prod_{j \in J \cap N(i)} (1 - \varepsilon_j)} \le \varepsilon_i.$$

Тут первое неравенство является применением неравенства (5), равенство выполняется, т.к. A_i независимо от F (A_i независимо от не соседей A_i), а второе неравенство следует непосредственно из условия теоремы (числитель) и неравенства (7) для $\ell + m = k$ (знаменатель). □

Теперь можно описать, как будет устроена индукция. База индукции — это неравенство (6) для $k = 0$ (следует из условия теоремы), что то же самое, что и неравенство (7) для $\ell = 1$ и $m = 0$. Теперь предположим, что мы уже доказали неравенства (6) для $k < t$ и неравенства (7) для $\ell + m \le t$. Применим сначала лемму 8.2 и получим неравенство (6) для $k = t$. Затем при помощи леммы 8.1 мы получим неравенство (7) для $\ell + m = t + 1$.

Завершает доказательство следующее наблюдение: локальная лемма Ловаса является частным случаем неравенства (7) при $\ell = n$ и $m = 0$. □

Следствие 8.1 (Локальная лемма Ловаса для симметричного случая)**.** *Пусть в условии локальной леммы Ловаса дополнительно известно, что каждое событие A_i имеет вероятность не более p и число соседей не более d. Тогда, если*

$$ep(d+1) \leq 1,$$

то с положительной вероятностью не произойдёт ни одного события A_i.

Доказательство. По лемме Ловаса нам нужно подобрать ε_i такие, что

$$\Pr[A_i] \leq p \leq \varepsilon_i \cdot \prod_{j \in N(i)} (1 - \varepsilon_j).$$

Давайте для всех событий возьмём один и тот же ε. Тогда нам достаточно найти ε, удовлетворяющий условию

$$p \leq \varepsilon \cdot (1-\varepsilon)^d.$$

Рассмотрим выражение $(d\varepsilon) \cdot (1-\varepsilon)^d$. Если сложить все $d+1$ сомножителей (скобок), то в сумме получится d. Таким образом нам нужно максимизировать произведение при известной сумме множителей. Максимум достигается, когда все множители равны, т.е. $d\varepsilon = 1 - \varepsilon$, а следовательно $\varepsilon = 1/(d+1)$. При этом же значении достигается максимум исходного выражения $\varepsilon \cdot (1-\varepsilon)^d$. Получаем

$$p \leq \frac{1}{d+1} \cdot \left(1 - \frac{1}{d+1}\right)^d$$

Осталось заметить, что если $(1 + \frac{1}{d})^d < e$, то $(1 - \frac{1}{d+1})^d > 1/e$ при $d \geq 1$. □

Упражнение 8.1. В каждой клетке конечной ленты мы хотим написать число от 1 до N. При этом для каждой границы между клетками некоторые пары числе (l, r) запрещены, т.е. нельзя, чтобы слева от границы стояло l, а справа r. Докажите, что если для каждой границы доля запрещённых пар среди всех пар не больше $4/27$, то заполнение возможно.

Упражнение 8.2. Докажите аналогичный результат конструктивно и без использования локальной леммы Ловаса, если множество плохих пар имеет меру меньше $1/4$. (Это показывает, что в данной задаче локальная лемма Ловаса не даёт оптимального ответа.)

Теорема 8.5. *Пусть $\alpha < 1$ — некоторое положительное вещественное число. Пусть для каждого n некоторые двоичные слова, общим числом не более $2^{\alpha n}$, объявлены запрещёнными. Тогда существует число N и бесконечно большая последовательность нулей и единиц, не содержащая запрещённых подслов длиннее N.*

Доказательство. По соображениям компактности достаточно доказать существование сколь угодно длинных конечных последовательностей без запрещённых подслов.

Будем считать, что биты последовательности равновероятны и независимы. Появление запрещённой последовательности длинны n в данной позиции (на данном интервале

I) имеет вероятность $2^{(\alpha-1)n}$, где n — длина интервала. В качестве оценки в лемме Ловаса для этого события возьмём $2^{(\beta-1)n}$ для некоторого $\beta \in (0, \alpha)$. Нужно подобрать β так, чтобы выполнялось условие леммы Ловаса.

Соседями события на интервале I являются события на интервалах J, которые перекрываются с I. Поскольку вероятности событий зависят от длины, при подсчётах удобно группировать интервалы по длинам. Имеется $n+k-1$ интервал длины k, перекрывающихся с данным интервалом I длины n. Для каждого из них в правой части оценки леммы Ловаса появляется сомножитель $(1-2^{(\beta-1)k})$, и всего получается

$$\left(1-2^{(\beta-1)k}\right)^{n+k-1}.$$

Теперь перемножим это по всем k начиная с некоторого N. Таким образом, для применения леммы Ловаса нам необходимо, чтобы

$$2^{(\alpha-1)n} \leq 2^{(\beta-1)n} \cdot \prod_{k \geq N} \left(1-2^{(\beta-1)k}\right)^{n+k-1}.$$

(В данной формуле справа учитывается и сам интервал I при $n = k$, но это только уменьшает правую часть.) Покажем, что выполняется более сильное неравенство: грубо оценим $n+k-1 \leq nk$ (т.е. мы уменьшим числа в произведении), извлечём корень степени n и перенесём $2^{\beta-1}$ влево.

$$2^{(\alpha-\beta)} \leq \prod_{k \geq N} \left(1-2^{(\beta-1)k}\right)^{k}.$$

Применим к правой части неравенство Бернулли ($(1-x)^k \geq 1-kt$) — это ещё усилит неравенство. Получаем

$$2^{(\alpha-\beta)} \leq 1 - \sum_{k \geq N} k 2^{(\beta-1)k}.$$

Ряд в правой части сходится при $\beta < 1$, левая часть меньше 1 при $\alpha < \beta$. Таким образом по α можно выбрать $\beta < \alpha$ и достаточно большое N, для которого это неравенство выполняется, а значит выполняется и более слабое исходное неравенство. И раз так, то можно применить локальную лемму Ловаса для слов длины не меньше N.

Для получения бесконечного хорошего слова будем строить его итеративно. Начнём с некоторого хорошего слова длины 1, у которого есть бесконечное количество хороших продолжений (легко видеть, что такое есть). И будем постепенно добавлять к нему по одному символу так, чтобы и у полученного слова так же было бесконечное количество хороших продолжений (легко видеть, что на каждом шаге хотя бы один из символов нам подойдёт). $\square$

Упражнение 8.3 (Лемма Левина). Пусть $\alpha < 1$ — некоторое положительное вещественное число. Тогда существует бесконечная последовательность нулей и единиц, в которой все подслова достаточно большой длины n имеют сложность не менее αn.

Упражнение 8.4. Докажите двумерный аналог предыдущей теоремы: можно заполнить бесконечную клеточную бумагу нулями и единицами так, чтобы любой прямоугольник достаточно большой площади не был запрещённым, если для каждого прямоугольника площади k выбрано не более $2^{\alpha k}$ запрещённых, где $\alpha < 1$.

Пусть $w = w_0 w_1 w_2 \ldots$ — бесконечная последовательностью. Для любого конечного множества $F \subset \mathbb{N}$ индексов через $w(F)$ обозначим слово, составленное из символов w с номерами из F (в порядке возрастания номеров). Рассмотрим пару (F, X), где F — конечное множество индексов, а X — слово длины $|F|$. Будем говорить, что последовательность w запрещается парой (F, X), если $w(F) = X$. Пары такого вида будем называть *запрещениями*, а число элементов в F *размером запрещения*. Запрещение *покрывает* индексы, входящие в F.

Теорема 8.6. *Пусть задано некоторое положительное вещественное число $\alpha < 1$ и множество запрещений $\{(F, X)\}$, в котором для любого индекса i и числа n имеется не более $2^{\alpha n}$ запрещений размера n, которые покрывают i. Тогда существует число N и последовательность, не запрещённая ни одним из запрещений размера больше N.*

Доказательство. Аналогично предыдущей теореме будем доказывать это утверждение для конечных последовательностей, а потом воспользуемся компактностью.

Применим лемму Ловаса к нарушениям запрещений. Вероятность нарушения запрещения для запрещения размера n равна 2^{-n}. Для запрещения размера n в качестве ε_i возьмём $2^{-\beta n}$, где β — некоторая константа больше α.

Соседями запрещениями будут запрещения, пересекающиеся с ним (покрывающие общий индекс). Для леммы Ловаса надо взять запрещение размера n и проверить, что 2^{-n} не больше $2^{-\beta n}$, умноженного на произведение множителей $(1 - 2^{-\beta m})$ по всем запрещениям, пересекающимся с данным.

Разделим произведение на части, соответствующие различным точкам пересечения. Всего таких возможных точек n. Кроме того, в каждой точке сгруппируем сомножители по размерам. Тогда для данной точки и данного размера запрещения m получим не более $2^{\alpha m}$ сомножителей вида $(1 - 2^{-\beta m})$. Таким образом, нам нужно проверить неравенство

$$2^{-n} \le 2^{-\beta n} \cdot \prod_{m \ge N} (1 - 2^{-\beta m})^{2^{\alpha m} n}.$$

Возьмём корень степени n

$$2^{\beta - 1} \le \prod_{m \ge N} (1 - 2^{-\beta m})^{2^{\alpha m}}.$$

По неравенству Бернулли это выполняется, если

$$2^{\beta - 1} \le 1 - \sum_{m \ge N} 2^{\alpha m} \cdot 2^{-\beta m}.$$

Так как левая часть меньше 1, а ряд в правой части сходится, то для достаточно большого N это неравенство выполняется.

Аналогично предыдущей теореме можно построить бесконечную последовательность, не нарушающую запрещений большого размера (нужно сначала выбрать β, потом достаточно большое N и применить лемму Ловаса к последовательностям произвольной длины). $\square$

Следствие 8.2. *Пусть $\alpha < 1$ — некоторое положительное вещественное число. Существует последовательность w и константа N, для которых*

$$\max_{t \in F} K(F, w(F) \mid t) \geq \alpha |F|$$

при всех $F \subset \mathbb{N}$, содержащих не менее N элементов.

Замечание 8.1. В этом следствии в левой части неравенства в колмогоровской сложности к $w(F)$ почему-то добавилось F. Дело в том, что сложность подпоследовательности можно „закодировать“ в множестве индексов: например, в F можно записать индексы нулей внутри w и тогда $K(w(F)) = O(1)$. Максимум условной сложности в левой части возникает, т.к. мы хотели бы уметь „сдвигать“ множество индексов F вдоль последовательности, т.е. t в данном случае является чем-то вроде „точки привязки“. В такой формулировке, например, множество индексов $\{2, 4, 6\}$ будет иметь такую же сложность как $\{i, i+2, i+4\}$ для любых i, даже если сложность i большая.

Заметим, что отсюда в частности следует такое свойство w: всякое конечное множество F размера не менее N имеет элемент t, для которого

$$K(w(F) \mid F, t) \geq \alpha |F| - 2K(F \mid t).$$

Если опустить t в левой части, то получаем, что для любого достаточно большого F

$$K(w(F) \mid F) \geq \alpha |F| - 2 \max_{t \in F} K(F \mid t).$$

Если считать, что индексы расположены в плоскости, а в качестве F брать прямоугольники, то вычитаемое в правой части будет логарифмическим и его можно будет скомпенсировать за счёт α. Получается утверждение упражнения 8.4.

Доказательство. Применим теорему 8.6, где для запрещений (F, Z) выполняется неравенство $K(F, Z \mid t) < \alpha |F|$ для всех $t \in F$. Таким образом для каждого индекса t количество запрещений размера k, в которых содержится t, не превосходит $2^{\alpha k}$. □

8.4.1. „Эффективное“ доказательство леммы Ловаса

Все предыдущие рассуждения о лемме Ловаса были неконструктивными — мы доказывали существование некоторых объектов, но не говорили, как их найти. Более того, в наших доказательствах „хороший“ объект мог иметь экспоненциально маленькую вероятность, так что не было никакой надежды случайно „угадать“ его за какое-то разумное (полиномиальное) число попыток.

В данном разделе мы предложим очень простой вероятностный алгоритм, который с константной вероятностью за полиномиальное время позволит найти „хороший“ объект. Будем рассматривать применение алгоритма для конкретной задачи. Пусть нам нужно найти двоичное слово, имея некоторый набор запрещений, другими словами нам нужно найти выполняющий набор для КНФ. Выберем случайный набор значений переменных.

Некоторая доля запрещение при этом будет нарушена. Возьмём какое-нибудь запрещение и снова выберем биты для переменных, которые в нём участвуют. Скорей всего это запрещение на новых значениях переменных выполнится, но могут нарушиться какие-то другие. Если так, то повторим процесс для новых нарушенных запрещений. И так далее. Удивительным образом этот простой алгоритм приводит к цели.

Для простоты мы будем считать, что все запрещения одинакового размера. Пусть КНФ содержит n переменных и N дизъюнктов размера m. Будем считать соседями дизъюнкты, имеющие общую переменную с данным. Пусть у каждого дизъюнкта не более t соседей. Если t невелико, то по лемме Ловаса эта формула выполнима.

Поскольку все дизъюнкты одного размера, то разумно в лемме Ловаса выбрать один и тот же ε для всех событий. Тогда мы должны выполнить следующее неравенство:

$$2^{-m} \leq \varepsilon(1-\varepsilon)^t.$$

Правая часть максимальна при $\varepsilon = 1/(t+1)$, но для простоты положим $\varepsilon = 1/t$. Тогда $2^{-m} \leq (1-1/t)^t/t \approx 1/et$, т.е. выполнимость гарантируется при $t \leq 2^m/e$. В конструктивном доказательстве нам потребуется более сильное условие $t \leq 2^m/8$.

Теорема 8.7 (Moser, Tardos, 2009)**.** *Существует вероятностный алгоритм, который ищет выполняющий набор любой формулы в КНФ с n переменными и N дизъюнктами размера m, у каждого из которых не более $2^m/8$ соседей, за полиномиальное от $n+N$ время с вероятностью не менее $1/2$.*

Доказательство. Будем считать, что на дизъюнктах задан порядок. Алгоритм будет использовать рекурсивную процедуру $\mathrm{Fix}(d)$:

выберем случайные значения переменных x;
для всех дизъюнктов d формулы:
  если $d(x) = 0$, то $\mathrm{Fix}(d)$.

Для того, что бы алгоритм был корректным, процедура Fix не должна нарушать выполнимость дизъюнктов, которые мы уже просмотрели (т.е. к моменту после вызова $\mathrm{Fix}(d)$ все дизъюнкты левее d уже выполнены и должны оставаться выполненными после вызова $\mathrm{Fix}(d)$).

Текст процедуры $\mathrm{Fix}(d)$:

обновляем x, выбирая случайные значения для всех переменных в d;
для всех соседей d' дизъюнкта d:
  если $d'(x) = 0$, то $\mathrm{Fix}(d')$.

Будем считать, что дизъюнкт является собственным соседом, тогда нам не нужно отдельно рассматривать случай, при котором новые значения переменных для d совпали с предыдущими (т.е. дизъюнкт снова оказался не выполнен). Корректность Fix следует из определения — если уж она завершилась, то она не могла сделать другие дизъюнкты невыполненными, т.к. изменяла только те переменные, которые есть в d.

Остаётся показать, что с большой вероятностью процесс завершится за полиномиальное время. Заметим, что случайные биты в данном алгоритме используются только для выбора исходных значений переменных (n битов) и на каждой итерации Fix — для выбора новых значений переменных дизъюнкта (m битов).

Воспользуемся следующим наблюдением: все случайные биты, использованные к данному моменту работы алгоритма, можно восстановить по текущим значениям переменных и по списку дизъюнктов, для которых вызывалась процедура Fix. Действительно, если в какой-то моменты времени произошёл вызов процедуры Fix(d), то можно восстановить значения переменных входящих в d, т.к. дизъюнкт не выполнен только при одном значении переменных. Тогда можно начать восстанавливать события с конца и восстановить все биты, использованные алгоритмом.

Пусть к данному моменту произошло k вызовов Fix, т.е. алгоритм использовал $n+km$ случайных битов. Для их восстановления достаточно знать:

1. текущие значения переменных,
2. номера дизъюнктов, для которых Fix вызывалась из основного алгоритма,
3. какие вызовы Fix были сделаны рекурсивно для каждого из дизъюнктов.

Для первого потребуется n битов, для второго не более N битов (номера дизъюнктов можно задать битовой маской). Для третьего потребуется описать дерево рекурсивных вызовов. Закодируем дерево следующим способом: при рекурсивном вывозе из d достаточно указывать номер соседа, для которого мы вызываемся, что требует $\log t$ битов. Кроме того нам нужно как-то разделить ситуации, когда в Fix происходит рекурсивный вызов (переход вниз по дереву вызовов), а когда выход из процедуры (переход вверх по дереву вызовов). Поэтому на каждый переход добавим один бит: 0, если мы выходим из процедуры, 1, если мы переходим к соседу, и тогда следующие $\log t$ битов — это номер соседа.

Итого на каждую вершину дерева вызовов мы потратим $\log t + 2$ битов. В сумме получится $N+n+k\cdot(\log t+2)$. С вероятность $1/2$ случайные биты имеют максимальную колмогоровскую сложность, т.е.

$$n + k \cdot m - 1 \leq K(\text{случайные биты}) \leq N + n + k \cdot (\log t + 2) + c,$$

что даёт ограничение сверху на k, т.к. $\log t + 2 = \log(2^m/8) + 2 = m - 1$. Следовательно $k \leq N + c + 1$, где c — константа. □

Список литературы

[1] Н.К. Верещагин, Е.В. Щепин. *Информация, кодирование, предсказание*, МЦНМО, 2012.

[2] Н.К. Верещагин. *Коммуникационная сложность*, Computer Science клуб, 2017. http://compsciclub.ru/courses/communicationcomplexity/2017-spring/

[3] А.Е. Ромащенко. *Введение в теорию информации,* Computer Science клуб, 2015. http://compsciclub.ru/courses/informationtheory/2015-spring/

[4] А.Е. Ромащенко. *Краткий конспект лекций курса "Введение в теорию информации",* 2014. http://www.mccme.ru/~anromash/courses/lecture-notes-it-2014.pdf

[5] В.А. Успенский, Н.К. Верещагин, А.Шень. *Введение в колмогоровскую сложность.* МЦНМО, 2012.

[6] А. Шень. *Алгоритмическая теория информации,* Computer Science клуб, 2008. http://compsciclub.ru/courses/algo-information-theory/2008-autumn/

[7] M. L. Bonet, S. R. Buss. *Size-depth tradeoffs for Boolean formulae.* Information Processing Letters, 49(3), 151-155, 1994.

[8] D. Gavinsky, O. Meir, O. Weinstein, A. Wigderson. *Toward better formula lower bounds: an information complexity approach to the KRW composition conjecture.* STOC 2014.

[9] T.Kaced, A.E. Romashchenko, N.K.Vereshchagin, *A Conditional Information Inequality and Its Combinatorial Applications*. IEEE Trans. Information Theory, 2018.

[10] E. Nisan, N. Kushilevitz. *Communication complexity*, 1997.

[11] A. Rao. *Notes for CSE533: Information Theory in Computer Science*, 2010. https://homes.cs.washington.edu/~anuprao/pubs/CSE533Autumn2010/