

Теория кодирования

МФТИ, осень 2013

Александр Дайняк

www.dainiak.com

Коммуникационная сложность

- У Аси есть слово X , а у Бори слово Y , где $X, Y \in \{0,1\}^k$.
(Ася не знает Y , а Боря не знает X .)
- Задана функция f , определённая на $\{0,1\}^k \times \{0,1\}^k$.
- Ася и Боря хотят вычислить значение $f(X, Y)$, переслав для этого друг другу минимум данных.
- $L_{\text{comm}}(f) :=$ число битов, которые в сумме Ася и Боря перешлют друг другу в худшем случае при использовании фиксированного алгоритма вычисления f .
- Пример: $L_{\text{comm}}(\mathbb{1}_{X=Y}) = k + 1$. (Доказательство: пр-п Дирихле.)

Коды в качестве усилителей различия

Построим *рандомизированный* алгоритм вычисления $\mathbb{1}_{X=Y}$, при котором Ася и Боря пересылаю друг другу всего $O(\log k)$ битов.

Идея: используем код, исправляющий ошибки, в качестве «усилителя различия» слов.

Ася и Боря кодируют X и Y в одном и том же $[n, k, d]_q$ -коде (кодируя биты 0 и 1 различными элементами из \mathbb{F}_q), получая слова $X', Y' \in \mathbb{F}_q^n$.

Ася выбирает *случайную* позицию в X' и пересылает её номер (в двоичной записи объёмом $\lceil \log_2 n \rceil$) и её значение (объёмом $\lceil \log_2 q \rceil$).

Боря проверяет, совпадает ли принятое от Аси значение с соответствующим значением в Y' , и затем результат сравнения он одним битом пересылает Асе.

Вероятность ошибки не превосходит $(n - d)/n$.

Коды в качестве усилителей различия

Пусть $\varepsilon \in (0,1)$.

Ася и Боря используют $\left[\frac{1}{\varepsilon} \cdot k, k, \frac{1-\varepsilon}{\varepsilon} \cdot k + 1\right]_q$ -код Рида—Соломона, где $q \in \left[\frac{k}{\varepsilon}, \frac{2k}{\varepsilon}\right]$.

Тогда вероятность ошибки будет не больше ε , а количество бит, которые Ася и Боря перешлют друг другу, равно $O\left(\log \frac{k}{\varepsilon}\right)$.

О криптографии с открытым ключом на одном слайде

- Ася вывешивает в интернете алгоритм с открытыми исходниками, который преобразует сообщения X в $\phi(X)$.
- У Аси есть алгоритм, *который знает только она*, позволяющий по коду вида $\phi(X)$ эффективно восстановить сам X .
- Никто, кроме Аси (т.е. у кого нет секретного алгоритма декодирования) не должен уметь эффективно восстанавливать X по $\phi(X)$. Так что собеседник Аси Боря может выкладывать в открытый доступ сообщения, к которым сможет обращаться кто угодно, но расшифровать (за приемлемое время) Борины послания сможет только Ася.

Криптосхема МакЭлиса (R. McEliece '1978)

Ася выбирает (не раскрывая никому)

- произвольный $[n, k, d]$ -код C , где $d \geq 2t + 1$; этот код должен обладать эффективными алгоритмами построения порождающей матрицы $G \in \mathbb{F}_2^{k \times n}$ и декодирования с исправлением не более t ошибок.
- *случайную* невырожденную матрицу S из $\mathbb{F}_2^{k \times k}$,
- *случайную* перестановочную матрицу P из $\mathbb{F}_2^{n \times n}$.

Затем Ася вычисляет матрицу $\hat{G} := SGP \in \mathbb{F}_2^{k \times n}$ и выкладывает в открытый доступ алгоритм, который

- по сообщению $X \in \mathbb{F}_2^k$ вычисляет вектор $X\hat{G} \in \mathbb{F}_2^n$ и искажает его в t случайных битах.

Криптосхема МакЭлиса (R. McEliece '1978)

- $[n, k, d]$ -код C , с порождающей матрицей $G \in \mathbb{F}_2^{k \times n}$
- *случайная* невырожденная матрица $S \in \mathbb{F}_2^{k \times k}$ (секрет Аси!),
- *случайная* перестановочная матрица $P \in \mathbb{F}_2^{n \times n}$ (секрет Аси!),
- $\hat{G} := SGP \in \mathbb{F}_2^{k \times n}$ — известная всем матрица,
- По сообщению $X \in \mathbb{F}_2^k$ вычисляется вектор $X\hat{G} \in \mathbb{F}_2^n$ и искажается в t случайных битах. Получается вектор \tilde{X} .

Ася может восстановить X , декодировав с исправлением ошибок вектор $\tilde{X}P^{-1}$ (ведь это искажённое слово кода C), и домножив результат на S^{-1} .

Криптосхема МакЭлиса (R. McEliece '1978)

- $G \in \mathbb{F}_2^{k \times n}$ — порождающая матрица «хорошего» $[n, k, d]$ -кода
- *случайная* невырожденная матрица $S \in \mathbb{F}_2^{k \times k}$ (секрет Аси!),
- *случайная* перестановочная матрица $P \in \mathbb{F}_2^{n \times n}$ (секрет Аси!),
- $X \rightarrow XSGP \rightarrow \tilde{X}$

Почему именно так:

- Предполагается, что задача NCP даже при известной порождающей матрице кода трудна для «почти всех» кодов.
Значит, даже зная хороший алгоритм декодирования кода с матрицей G , трудно декодировать код с матрицей GP .
- Домножение X на S перед кодированием призвано разрушить внутреннюю структуру X , чтобы трудно было «угадать» X .

l -однородные множества

Множество наборов $U \subseteq \{0,1\}^n$ называется l -однородным, если для любых $i_1, \dots, i_l \in \{1, \dots, n\}$ и любых $t_1, \dots, t_l \in \{0,1\}$ выполнено

$$\frac{|\{(a_1, \dots, a_n) \in U \mid a_{i_1} = t_1, a_{i_2} = t_2, \dots, a_{i_l} = t_l\}|}{|U|} = 2^{-l}$$

То есть при случайном равномерном выборе набора $\mathbf{a} \in U$ любые l бит в \mathbf{a} будут равны фиксированным значениям с той же вероятностью, что и при случайном выборе из «полного» множества $\{0,1\}^n$.

l -однородность и порождающая матрица

Лемма.

Пусть $C \subseteq \mathbb{F}_2^n$ — линейный $[n, k, \dots]$ -код. Множество C является l -однородным т. и т.т., когда любые l столбцов порождающей матрицы кода линейно независимы.

Доказательство:

Пусть $G_1, \dots, G_n \in \mathbb{F}_2^k$ — столбцы порождающей матрицы G кода C .

Пусть, например, $G_1 + G_2 + \dots + G_s = \mathbf{0}$, где $s \leq l$. Рассмотрим тогда любые t_1, \dots, t_l , такие, что $t_1 + \dots + t_s = 1$. Имеем

$$\frac{|\{(a_1, \dots, a_n) \in C \mid a_1 = t_1, \dots, a_l = t_l\}|}{|C|} = 0 \neq 2^{-l}$$

l -однородность и порождающая матрица

$G_1, \dots, G_n \in \mathbb{F}_2^k$ — столбцы порождающей матрицы G кода C .

Пусть теперь G_1, \dots, G_l линейно независимы. Тогда ранг матрицы $(G_1 | G_2 | \dots | G_l)$ равен l , и значит в G найдутся строки, — пусть это строки $\mathbf{g}_1, \dots, \mathbf{g}_l \in \mathbb{F}_2^n$, — такие, что их начальные куски длины l линейно независимы.

Обозначим $C_{t_1, \dots, t_l} := \{\mathbf{c} \in C \mid c_1 = t_1, \dots, c_l = t_l\}$.

Для любых $t'_1, \dots, t'_l, t''_1, \dots, t''_l$ найдётся кодовое слово \mathbf{a} (линейная комбинация строк $\mathbf{g}_1, \dots, \mathbf{g}_l$), для которого

$$a_1 = t'_1 + t''_1, \dots, a_l = t'_l + t''_l.$$

Тогда $C_{t'_1, \dots, t'_l} = C_{t''_1, \dots, t''_l} + \mathbf{a}$, отсюда $|C_{t'_1, \dots, t'_l}| = |C_{t''_1, \dots, t''_l}|$.

l -однородность и порождающая матрица

$G_1, \dots, G_n \in \mathbb{F}_2^k$ — столбцы порождающей матрицы G кода C .

$$C_{t_1, \dots, t_l} := \{c \in C \mid c_1 = t_1, \dots, c_l = t_l\}.$$

Если G_1, \dots, G_l линейно независимы, то для любых $t'_1, \dots, t'_l, t''_1, \dots, t''_l$

$$\text{имеем } |C_{t'_1, \dots, t'_l}| = |C_{t''_1, \dots, t''_l}|.$$

Отсюда $|C_{t_1, \dots, t_l}| = \frac{|C|}{2^l}$ для любых t_1, \dots, t_l .

Лемма доказана.

l -однородность и кодовое расстояние

Лемма.

Пусть $C \subseteq \mathbb{F}_2^n$ — линейный $[n, k, \dots]$ -код. Множество C является l -однородным т. и т.т., когда любые l столбцов порождающей матрицы кода линейно независимы.

Теорема.

Двоичный линейный код C является l -однородным множеством т. и т.т., когда $d(C^\perp) > l$.

Доказательство: применяем лемму, заметив, что порождающая матрица C является проверочной для C^\perp , и используем утверждение о связи кодового расстояния с проверочной матрицей линейного кода.

q -ичные l -однородные множества

Теорема.

Двоичный линейный код C образует l -однородное множество т. и т.т., когда $d(C^\perp) > l$.

Замечание.

Аналогично можно вести понятие q -ичного l -однородного множества и доказать похожую теорему: линейный код $C \subseteq \mathbb{F}_q^n$ образует l -однородное множество т. и т.т., когда $d(C^\perp) > l$.

l -однородные множества на основе РМ-кодов

Интересны l -однородные множества малой (полиномиальной по n) мощности.

Для того, чтобы получить такое множество, нужно взять линейный код C у которого

- $d(C) > l$
- $\dim C$ велико

и затем рассмотреть C^\perp .

Возьмём в качестве C код Рида—Маллера с параметрами

$$m := \lceil \log_2 n \rceil, \quad r := m - 2$$

Для такого C имеем $d(C) = 2^{m-r} = 4$.

Код C^\perp тоже является РМ-кодом, с параметром $r' = m - r - 1 = 1$.

При этом $|C^\perp| = 2^{1+m} \leq 4n$.

l -однородные множества на основе РМ-кодов

Доказанное утверждение:

Код Рида—Маллера с параметрами $m = \lceil \log_2 n \rceil$ и $r = 1$ образует 3-однородное множество наборов длины n , мощность которого линейна по n .

Задача $3_{\geq \gamma}$ -SAT

Задача $3_{\geq \gamma}$ -SAT: для заданной 3-КНФ найти набор, на котором не менее чем γ -я доля всех скобок обращается в единицу.

Обычная задача 3-SAT — это $3_{\geq 1}$ -SAT.

Теорема.

Задача $3_{\geq 7/8}$ -SAT полиномиально разрешима.

Полиномиальность $3_{\geq 7/8}$ -SAT

Теорема.

Задача $3_{\geq 7/8}$ -SAT полиномиально разрешима.

Доказательство:

Пусть 3-КНФ содержит n переменных и m скобок.

При случайном выборе набора из $\{0,1\}^n$ имеем

$$\Pr[\text{фиксированная скобка равна нулю}] = \frac{1}{8}$$

Отсюда

$$\mathbb{E} \# \text{скобок, равных нулю} = \frac{m}{8}$$

Заметим, что $\Pr[\text{фикс. ск.} = 0] = \frac{1}{8}$ и в том случае, когда берётся случайный набор из произвольного 3-однородного множества.

Полиномиальность $3_{\geq 7/8}$ -SAT

Получается, что в любом 3-однородном множестве найдётся набор, на котором $\leq \frac{m}{8}$ скобок равны нулю.

Получается простой алгоритм:

- Перебираем всевозможные наборы РМ-кода (с нужными параметрами) и подставляем их в 3-КНФ. Хотя бы один из наборов должен сгодиться.

Задача о разделении секрета

Задача:

Есть несколько человек и *секрет*.

Нужно сообщить людям некоторую информацию, так, чтобы

- все вместе они могли бы восстановить секрет
- никакая компания из меньшего числа человек не могла бы восстановить секрет

Задача о разделении секрета

Пусть нужно разделить секрет между m людьми.

- Сопоставляем секрету элемент $s \in \mathbb{F}_q$.
- Берём q -ичный код C , такой, что $d(C^\perp) = m + 1$.
- В порождающей матрице G кода C найдутся $(m + 1)$ линейно независимых столбцов, пусть это первые $(m + 1)$ столбцов. Тогда найдутся такие $\alpha_1, \dots, \alpha_{m+1} \in \mathbb{F}_q \setminus \{0\}$, что в любом кодовом слове c первые $(m + 1)$ разрядов удовлетворяют соотношению $\alpha_1 c_1 + \dots + \alpha_{m+1} c_{m+1} = 0$.
Т.е. c_{m+1} всегда можно однозначно определить по c_1, \dots, c_m .

Задача о разделении секрета

Секрет $s \in \mathbb{F}_q$ разделяем между m людьми.

- Берём q -ичный код C , такой, что $d(C^\perp) = m + 1$.
Пусть в порождающей матрице G кода C первые $(m + 1)$ столбцов линейно зависимы.
- Выбираем *случайные* элементы t_1, \dots, t_{m-1} .
- Элемент t_m однозначно выбираем так, чтобы в коде C нашлось слово вида

$$(t_1, \dots, t_{m-1}, t_m, s, \dots)$$

Поскольку C является q -ичным m -однородным множеством, зная любые $(m - 1)$ из чисел t_1, \dots, t_m , об s ничего нельзя сказать.

Как кодируются данные на CD

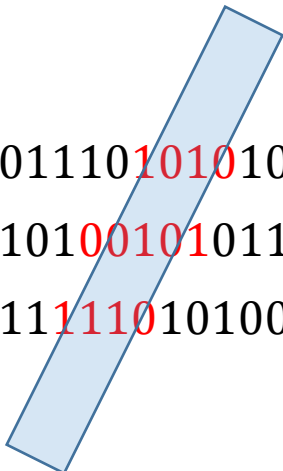
Параметры CD-ROM:

- Длина дорожки ≈ 5 км
- Ширина дорожки ≈ 6 мкм
- Высота углублений ≈ 1.2 мкм

Задача: обеспечить сохранность данных при наличии царапин/пыли/грязи.

Как кодируются данные на CD

Одна царапина может затрагивать много последовательных битов:



... 0010101110101010101 ...
... 101001010010101110100 ...
... 101001111101010000100 ...

В этом случае говорят о наличии *пакетов ошибок*.

Как кодируются данные на CD

Одна царапина может затрагивать много последовательных битов.

Выход: кодировать данные *с перемежением (interleaving)*, так, чтобы последовательные биты на дорожке не отвечали одному и тому же кодовому слову.

Добавляем помехоустойчивое кодирование Рида—Соломона и получаем технологию:

CIRC = Cross-interleaved Reed—Solomon Codes

Кодирование CIRC

- Каждый отсчёт одного канала звукозаписи занимает 16 бит
- Разбиваем 16 бит на две восьмёрки, и считаем каждую из них элементом \mathbb{F}_{256} . Пару элементов \mathbb{F}_{256} будем обозначать одной буквой.
- Запись на CD двухканальная (стерео), так что данные выглядят так:

$$L_1 R_1 L_2 R_2 L_3 R_3 \dots$$

— где $L_i, R_i \in \mathbb{F}_{256}^2$.

Кодирование CIRC

Последовательность данных

$$L_1 R_1 L_2 R_2 L_3 R_3 \dots$$

разбивается на *кадры (фреймы)*:

$$L_1 R_1 \dots L_6 R_6 \mid L_7 R_7 \dots L_{12} R_{12} \mid L_{13} R_{13} \dots$$

Затем в каждой паре последовательных фреймов перемежаем данные так (на примере первой пары фреймов):

$$L_1 L_3 L_5 R_1 R_3 R_5 L_8 L_{10} L_{12} R_8 R_{10} R_{12} L_7 L_9 L_{11} R_7 R_9 R_{11} L_2 L_4 L_6 R_2 R_4 R_6$$

Кодирование CIRC

После перемежения фреймов получается последовательность:

$$L_1 L_3 L_5 R_1 R_3 R_5 L_8 L_{10} L_{12} R_8 R_{10} R_{12} L_7 L_9 L_{11} R_7 R_9 R_{11} L_2 L_4 L_6 R_2 R_4 R_6 \dots$$

Далее к каждому 24 последовательным байтам применяется *систематический* $[28,24,5]_{256}$ -код Рида—Соломона.

Так, например, к последовательности

$$L_1 L_3 L_5 R_1 R_3 R_5 L_8 L_{10} L_{12} R_8 R_{10} R_{12}$$

добавятся 4 проверочных байта, которые вставляются в середину:

$$L_1 L_3 L_5 R_1 R_3 R_5 \mathbf{P_1 P_2} L_8 L_{10} L_{12} R_8 R_{10} R_{12}$$

Кодирование CIRC

Получается последовательность 28-байтных слов.

Каждый блок из 28 таких слов записывается в виде матрицы 28×136 :

$c_{1,1}$	$c_{2,1}$	$c_{3,1}$	$c_{4,1}$	$c_{5,1}$	$c_{6,1}$	$c_{7,1}$	$c_{8,1}$	$c_{9,1}$	$c_{10,1}$...
0	0	0	0	$c_{1,2}$	$c_{2,2}$	$c_{3,2}$	$c_{4,2}$	$c_{5,2}$	$c_{6,2}$...
0	0	0	0	0	0	0	0	$c_{1,3}$	$c_{2,3}$...
				\vdots						

Здесь $c_{i,j}$ — это j -й байт i -го слова.

Затем каждый столбец этой матрицы кодируется с помощью $[32,28,5]_{256}$ -кода Рида—Соломона.

Получается последовательность 32-байтных слов, она и записывается на CD.