Теория кодирования

Михайлов Максим

17 января 2023 г.

Оглавление

Лекция 1

2 сентября

1 Введение

Определение. Передаваемый сигнал это

$$x(t) = \sum_i S_{x_i}(t - iT)$$

, где x_i — передаваемые символы, T — продолжительность передачи одного символа.

Пример (M-ичная амплитудно-импульсная модуляция).

$$S_i(t) = \alpha(2i + 1 - M)g(t)\sin(2\pi ft)$$

, где:

- $g(t)$ — сигнальный импульс
- f — несущая частота
- α — коэффициент энергии передаваемого сигнала

Модели канала:

1. В непрерывном времени: $y(t) = x(t) + \eta(t)$
2. В дискретном времени: $y_i = \alpha(2x_i + 1 - M) + \eta_i$

η — белый шум, обычно Гауссов $\mathcal{N}(0, \sigma^2)$.

У передаваемого сигнала обычно не должно быть постоянной компоненты (по причинам физики), поэтому сигнал симметричен. С точки зрения теории кодирования это несущественно.

Приемник наблюдает на выходе канала вектор $y = (y_0 \dots y_{n-1})$. Канал характеризуется условным распределением $P_{Y|X}(y \mid x)$, где X, Y — случайные величины, соответствующие векторам переданных и принятых символов.

Приемник разбивает векторное пространство на решающие области $R_x : y \in R_x \implies \hat{x} = x$, т.е. если сигнал попал в область R_x, то мы ему сопоставляем кодовый символ x. Тогда вероятность ошибки:

$$\begin{aligned} P_e &= \int_{\mathbb{R}^N} P_e(y) p_Y(y) dy \\ &= \sum_x \int_{R_x} p_e(y) p_Y(y) dy \\ &= \sum_x \int_{R_x} (1 - p_{X|Y}\{x \mid y\}) p_Y(y) dy \\ &= 1 - \sum_x \int_{R_x} p_{X|Y}\{x \mid y\} p_Y(y) dy \end{aligned}$$

Приемник должен найти оптимальные решающие области. Оптимальность определяется критерием, например:

1. Критерий максимума апостериорной вероятности (критерий идеального наблюдателя)

$$\begin{aligned} R_x &= \{y \mid p_{X|Y}(x \mid y) > p_{X|Y}(x' \mid y), x' \neq x\} \\ &= \{y \mid P_X(x) p_{Y|X}(y \mid x) > P_X(x') p_{Y|X}(y \mid x'), x' \neq x\} \end{aligned}$$

2. Критерий максимума правдоподобия

$$R_x = \{y \mid p_{Y|X}(y \mid x) > p_{Y|X}(y \mid x'), x' \neq x\}$$

   В случае равновероятных символов этот критерий совпадает с критерием идеального наблюдателя.

Пример (2-ичная амплитудно-импульсная модуляция (2-АМ)). Пусть $y_i = \alpha(2x_i - 1) + \eta_i, \eta_i \sim \mathcal{N}(0, \sigma^2), x_i \in \{0, 1\}$. Тогда:

$$p_{Y|X}(y \mid x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y - \alpha(2x-1))^2}{2\sigma^2}}$$

Применим критерий максимального правдоподобия:

$$R_0 = \{y \mid y < 0\}, R_1 = \{y \mid y \geq 0\}$$

Вычислим вероятность ошибки:

$$P_e = P_X(0) P\{Y \geq 0 \mid X = 0\} + P_X(1) P\{Y < 0 \mid X = 1\}$$

$$
\begin{aligned}
&= 0.5 \int_0^\infty p_{Y|X}(y \mid 0) dy + 0.5 \int_{-\infty}^0 p_{Y|X}(y \mid 1) dy \\
&= \int_0^\infty \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(y+\alpha)^2}{2\sigma^2}} dy \\
&= \int_\alpha^\infty \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{y^2}{2\sigma^2}} dy \\
&= \int_{\frac{\alpha}{\sigma}}^\infty \frac{1}{\sqrt{2\pi}} e^{-\frac{y^2}{2}} dy \\
&=: Q\left(\frac{\alpha}{\sigma}\right) \\
&= \frac{1}{2}\operatorname{erfc}\left(\frac{\alpha}{\sqrt{2}\sigma}\right)
\end{aligned}
$$

Значение сигнала это обычно уровень напряжения. Как мы знаем из школьной физики, мощность $P = \frac{U^2}{R}$. Мы хотим минимизировать мощность, чтобы экономить электроэнергию. Мощность сигнала суть случайная величина с матожиданием, пропорциональным $E_S = \alpha^2$. Мощность белого шума не зависит от частоты и пропорциональна $\sigma^2 = \frac{N_0}{2}$. Если же шум зависит от частоты, то он называется розовым или голубым.

Соотношение мощностей сигнал/шум на символ это $\frac{E_S}{N_0}$, обычно измеряемое в децибелах, т.е. $10 \log_{10} \frac{E_S}{N_0}$. Однако нас интересуют не символы, а биты и тогда соотношение сигнал/шум на бит это $\frac{E_S}{RN_0}$, где R — количество бит информации, представленных одним символом.

1.1 Способы снижения вероятности ошибки

1. Посимвольное повторение: будем передавать вместо каждого символа m копий того же символа.

   $$y_{mi+j} = \alpha(2x_i - 1) + \eta_{mi+j}, \quad 0 \le j < m$$

   Рассмотрим разные способы работы приемника:

   - Для каждого y_{mi+j} проведем голосование. Тогда вероятность ошибки:

     $$P_v(m) = \sum_{j=\lceil \frac{m}{2} \rceil}^{m-1} C_m^j P_e^j (1 - P_e)^{m-j}$$

   - Примем решение по

     $$\sum_{j=0}^{m-1} y_{mi+j} = m\alpha(2x_i - 1) + \sum_{j=0}^{m-1} \eta_{mi+j}$$

, т.е сложим наблюдения в одном блоке. Тогда вероятность ошибки:

$$P_a(m) = Q\left(\frac{m\alpha}{\sqrt{m}\sigma}\right) = Q\left(\sqrt{2\frac{mE_s}{N_0}}\right) = Q\left(\sqrt{2\frac{E_b}{N_0}}\right)$$

Выигрыша не будет, т.к. на один символ передается в m раз меньше бит (см. последний переход).

Второй метод лучше первого, т.к. мы не теряем информацию о нашей уверенности в каждом принятом символе.

Избыточность на уровне битов не дала улучшения, поэтому введем избыточность на уровне блоков.

1.2 Понятие кода

Определение. Код — множество допустимых последовательностей символов алфавита X.

Последовательности могут быть конечными или бесконечными, но на практике только конечными. Не каждая последовательность символов алфавита является кодовой.

Определение. Кодер — устройство, отображающее информационные последовательности символов алфавита $\mathcal{B}$ в кодовые.

Различным последовательностям алфавита $\mathcal{B}$ сопоставляются различные последовательности алфавита X для однозначности кодирования.

Определение. Скорость кода — отношение длин информационной и кодовой последовательностей.

Определение. Декодер — устройство, восстанавливающее по принятой последовательности символов наиболее вероятную кодовую последовательность.

1.3 Теоремы кодирования

Пусть для передачи используется код $\mathcal{C} \subset X^n$ длины n, состоящий из M кодовых слов, выбираемых с одинаковой вероятностью.

Теорема 1 (обратная)**.** Для дискретного постоянного канала с пропускной способностью C для любого $\delta > 0$ существует $\varepsilon > 0$ такое, что для любого кода со скоростью $R > C + \delta$ средняя вероятность ошибки $\overline{P_e} \geq \varepsilon$

Теорема 2 (прямая)**.** Для дискретного постоянного канала с пропускной способностью C для любых $\varepsilon, \delta > 0$ существует достаточно большое число $n_0 > 0$, такое что для всех натуральных $n \geq n_0$ существует код длиной n со скоростью $R \geq C - \delta$, средняя вероятность ошибки которого $\overline{P_e} \leq \varepsilon$

1.4 Пропускная способность некоторых каналов

1. Двоичный симметричный канал: $X, Y \in \{0, 1\}, p_{Y|X}(y \mid x) = \begin{cases} p, & y \neq x \\ 1-p, & y = x \end{cases}$. Из теории информации:

$$C_{BSC} = 1 + p \log_2 p + (1-p) \log_2 (1-p)$$

2. Идеальный частотно ограниченный гауссовский канал: $y(t) = x(t) + \eta(t), \eta(t) -$ гауссовский случайный процесс, спектральная плотность мощности которого равна $S(f) = \begin{cases} \frac{N_0}{2}, & -W < f < W \\ 0, & \text{иначе} \end{cases}$ Из теории случайных процессов:

$$C_{AWGN} = W \log_2 \left(1 + \frac{E_S}{W N_0}\right)$$

$$\lim_{W \to \infty} C_{AWGN} = \frac{E_S}{N_0 \ln 2}$$

Лекция 2

9 сентября

1.5 Энергетический выигрыш кодирования

Энергетический выигрыш кодирования показывает, во сколько раз использование кодирования позволяет уменьшить отношение сигнал/шум, необходимое для достижения заданной вероятности ошибки по сравнению со случаем отсутствия кодирования.

Выигрыш от кодирования неидеален, т.к.:

1. Оптимальный код сложно придумать (прямая теорема кодирования не конструктивна)
2. Используются коды конечной (и обычно малой) длины
3. Алгоритмы декодирования часто декодируют не по максимуму правдоподобия, а как получится
4. Дискретизация выхода канала

Теория кодирования занимается увеличением энергетического выигрыша кодирования.

2 Жесткое и мягкое декодирование

- При **мягком декодировании** декодер непосредственно использует y_i, т.е. есть информация о надежности символов.
- При **жестком декодировании** декодер использует только оценки $\hat{x_i}$.

Пример. Канал с аддитивным белым гауссовским шумом (АБГШ): $y_i = (2x_i - 1) + \eta_i, x_i \in \{0, 1\}$

В случае жесткого декодирования канал превращается в двойной симметричный канал с вероятностью ошибки $p = Q\left(\sqrt{2\frac{E_s}{N_0}}\right)$

Примечание. При малых x $Q(x) \approx \frac{1}{2} - \frac{x}{\sqrt{2\pi}}$

Достижимая скорость передачи данных:

$$R < \frac{2}{\ln 2}\left(\sqrt{\frac{E_s}{N_0\pi}}\right)^2$$

$$\frac{E_b}{N_0} = \frac{E_s}{RN_0} > \frac{\pi}{2}\ln 2 = 1.09 = 0.37$$

Как мы уже выяснили, для мягкого декодирования это значение -1.59, жесткое декодирование хуже.

Пример. Рассмотрим идеальный частотно ограниченный гауссовский канал, т.е. спектральная плотность мощности отличается от нуля только в некотором диапазоне частот.

Спектральная эффективность кодирования — $\beta = \frac{R}{W}$, скорость передачи данных, поделенная на ширину канала.

$$\frac{R}{W} < \log_2\left(1 + \frac{R}{W}\frac{E_b}{N_0}\right)$$

$$\frac{E_b}{N_0} > \frac{2^\beta - 1}{\beta}$$

2.1 Критерии декодирования

Оптимальные критерии декодирования сложны в реализации, сведем их к чему-нибудь попроще.

1. **Критерий минимального расстояния**: кодовое слово $c = \operatorname{argmin}_{c\in C} d(c, y)$
2. **Списочное декодирование**: ищем все кодовые слова, находящиеся в сфере заданного радиуса вокруг полученного вектора.
3. **Побитовое декодирование**: используется критерий идеального наблюдателя для отдельных символов кодового слова. Часто также дополняется вычислением логарифма отношений правдоподобия для отдельных символов.

$$L_i = \ln \frac{\sum_{\substack{c\in C\\ c_i=0}} P\{c \mid y\}}{\sum_{\substack{c\in C\\ c_i=1}} P\{c \mid y\}}$$

aaa

Не дописано

Лекция 3

16 сентября

Определение. Для группы $\mathcal{G} = (G, +)$ с подгруппой $\mathcal{H} = (H, +)$ назовём **смежным классом** для $x \in G$ множество:

$$x + H = \{x + h \mid h \in H\}$$

Пример. Для группы $\mathbb{Z}$ группа чётных чисел $2\mathbb{Z}$ является подгруппой. Для 1 смежный класс — все нечётные числа, для 2 — все чётные.

Не дописано

Иногда демодулятор может сообщить, что некоторый полученный символ ненадежен. Это называется **стиранием**. Также стиранием считаются потери пакетов в компьютерных сетях (UDP).

Стирания исправлять проще, чем ошибки, т.к. мы знаем, где они произошли.

Определение. Весовой спектр кода это $A_i = |\{c \in C \mid \mathrm{wt}(c) = i\}|$

Для двоичного симметричного канала с переходной вероятностью p вероятность необнаружения ошибки это:

$$P\{S = 0\} = \sum_{i=d}^{n} A_i p^i (1-p)^{n-i} \leq \sum_{i=d}^{n} C_n^i p^i (1-p)^{n-1}$$

Не дописано

Для аддитивного гауссовского канала с двоичной модуляцией вероятность ошибки мягкого декодирования по максимуму правдоподобия линейного блокового кода:

$$P \leq \sum_{i=d}^{n} A_i Q\left(\sqrt{2i\frac{E_s}{N_0}}\right) = \sum_{i=d}^{n} A_i Q\left(\sqrt{2iR\frac{E_b}{N_0}}\right) = \frac{1}{2}\sum_{i=d}^{n} A_i \mathrm{erfc}\left(\sqrt{iR\frac{E_s}{N_0}}\right)$$

Определение. Полное декодирование по минимальному расстоянию — нахождение по y ближайшего кодового слова $c = \text{argmin}_{c \in C} d(c, y)$

Определение. Информационная совокупность *(ИС)* — множество из k позиций в кодовом слове, значения которых однозначно определяют значения символов на остальных позициях кодового слова.

Определение. Если $\gamma = \{j_1 \dots j_k\}$ — информационная совокупность, то остальные позиции кодового слова называются **проверочной совокупностью**.

Если γ — информационная совокупность, то матрица из столбцов порождающей матрицы с номерами $j_1 \dots j_k$, обратима. Почему? Не дописаноБудем обозначать такую матрицу $M(\gamma)$.

Примечание. Эта матрица не единственна.

Пусть $G(\gamma) = M(\gamma)G$ — порождающая матрица, содержащая единичную подматрицу на столбцах $j_1 \dots j_k$.

Информационные совокупности позволяют более эффективно проверять коды, т.к. стандартный метод требует таблицу размера 2^k.

Теорема 3. Алгоритм декодирования по информационной совокупности обеспечивает полное декодирование по минимальному расстоянию.

Доказательство. Необходимо доказать, что для всякого исправимого вектора ошибки r существует информационная совокупность, свободная от ошибок.

Пусть c — единственное решение задачи декодирования по минимальному расстоянию. Тогда $e = r - c$ — вектор ошибки, $E = \text{supp}(e)$ — множество позиций ненулевых элементов e и $|E| \leq n - k$.

Пусть $N = \{1 \dots n\}$. Предположим, что $N \setminus E$ не содержит информационных совокупностей, тогда существуют различные кодовые слова, отличающиеся от r в позициях E. Но тогда c не единственно, противоречие. □

Сложность декодирования для $(n, k)_q$ кода экспоненциальная.

2.2 Покрытия

Определение. Покрытием $M(n, m, t)$ называется такой набор $F \subset 2^{N_n}$, где $N_n = \{1 \dots n\}, \forall f \in F \ \ |f| = m$ и любое t-элементное подмножество N_n содержится в одном из $f \in F$.

Для декодирования по информационной совокупности с исправлением не более t ошибок необходимо покрыть все исправимые конфигурации ошибок. Элементы такого покрытия — проверочные совокупности.

Пример. Не дописано

Построение минимального покрытия — NP-полная задача. Но есть итеративный приблизительный алгоритм.

2.3 Дуальные коды

Не дописано

2.4 Граница Хэмминга

Теорема 4. Для любого q-ичного кода с минимальным расстоянием $d = 2t + 1$ число кодовых слов удовлетворяет

$$A_q(n, d) \leq \frac{q^n}{\sum_{i=0}^{t} C_n^i (q-1)^i}$$

Доказательство. Если код способен исправить t ошибок, то вокруг всех кодовых слов можно описать хэмминговы шары радиуса t, не пересекающиеся друг с другом. □

2.4.1 Асимптотическая оценка

$$\begin{aligned} A_q(n, d) &\leq \frac{q^n}{\sum_{i=0}^{t} C_n^i (q-1)^i} \\ &\leq \frac{q^n}{C_n^{\frac{d-1}{2}} (q-1)^{\frac{d-1}{2}}} \\ &= \frac{q^n \left(\frac{d-1}{2}\right)! \left(n - \frac{d-1}{2}\right)!}{n! (q-1)^{\frac{d-1}{2}}} \\ &= ??? \end{aligned}$$

2.5 Граница Варшамова–Гилберта

Теорема 5. Существует q-ичный код длины n с минимальным расстоянием d, число слов которого удовлетворяет:

$$A_q(n, d) \geq \frac{q^n}{\sum_{i=0}^{d-1} C_n^i (q-1)^i}$$

Доказательство. Если код C имеет максимальную мощность, для любого вектора $x \notin C$ существует кодовое слово c такое, что $d(x, c) \leq d - 1$. Не дописано □

2.6 Граница Варшамова–Гилберта для линейных кодов

Теорема 6. Если

$$q^r > \sum_{i=0}^{d-2} C_{n-1}^i (q-1)^i$$

, то существует линейный код над $GF(q)$ длины n с минимальным расстоянием не менее d и не более чем $r = n - k$ проверочными символами.

Доказательство. Построим матрицу H размера $(n-k) \times n$ такую, что любые $d-1$ её столбцов линейно независимы.

Первый столбец пусть будет произвольным ненулевым вектором. Если уже выбраны j столбцов, то $j+1$-й столбец не может быть никакой линейной комбинацией любых $d-2$ выбранных столбцов. Таких столбцов $\sum_{i=0}^{d-2} C_j^i (q-1)^i$. Пока не запрещены все q^{n-k} столбцов, то можно выбрать ещё хотя бы один столбец. □

Не дописано

2.7 Граница Грайсмера

Обозначение. $N(k,d)$ — минимальная длина двоичного линейного кода размерности k с минимальным расстоянием d.

Теорема 7. $N(k,d) \geq d + N(k-1, \lceil d/2 \rceil)$

Доказательство. Пусть порождающая матрица (n,k,d) кода C наименьшей длины $n = N(k,d)$ имеет вид:

$$G = \begin{pmatrix} 0 & \dots & 0 & 1 & \dots & 1 \\ & G' & & & * & \end{pmatrix}$$

Не дописано □

Лекция 4

23 сентября

3 Универсальные методы декодирования линейных кодов

Мы изучали методы жесткого декодирования, но на практике чаще применяются методы мягкого декодирования, которые учитывают надежность символов.

Мы уже знаем, что декодирование кода по критерию максимального правдоподобия в канале с АБГШ эквивалентно декодированию по критерию минимального расстояния Евклида.

Несложными преобразованиями получим:

???

Пусть тогда $\hat{c}_i = \begin{cases} 0, & y_i > 0 \\ 1, & y_i \leq 0 \end{cases}$ — жесткие решения.

Тогда:

$$\text{argmax}_{c \in C} \sum_{i=0}^{n-1} (-1)^{c_i} y_i = \text{argmax}_{c \in C} \sum_{i=0}^{n-1} (-1)^{c_i} y_i$$

$$= ???$$

Не дописано

3.1 Метод порядковых статистик

Рассмотрим передачу кодовых слов $c_0 \dots c_{n-1}$ двоичного (n, k) кода с помощью символов 2-АМ, например $y_i = (-1)^{c_i} + \eta_i$ — с АБГШ.

Пусть тогда $L_i = \log \frac{P\{c_i=0|y_i\}}{P\{c_i=1|y_i\}}$ — логарифмические отношения правдоподобия. Вероятность ошибки в i-том жестком решении убывает с увеличением $|L_i|$. Тогда выберем информационную совокупность J для кода, соответствующего наибольшим значениям $|L_i|$. Приведем порождающую матрицу кода к виду G_J с единичной подматрицей в столбцах J.

С большой вероятностью число неверных решений $\hat{c}_j$, где $j \in J$, мало. Переберем все конфигурации ошибок e веса не более t на J и построим кодовые слова $c_e = (\hat{c}_J + e)G_J$. Из всех полученных кодовых слов выберем наиболее правдоподобное.

Сложность алгоритма $\mathcal{O}\left(k^2 n + \sum_{i=0}^{t} \text{in} C_k^i\right)$.

Есть два[1] способа ускорить этот алгоритм:

1. Обменять память на скорость каким-то образом (придумать дома).
2. Реализовать раннюю остановку.

Рассмотрим двоичный (n, k, d) код C с порождающей матрицей G, пусть $D(x, y)$ — функция расстояния Хемминга.

Существует много кодов, содержащих одинаковые префиксы некоторой длины a, поэтому можно не пересчитывать $\sum_{i=0}^{a} D(c_i, y_i)$ заново.

Не дописано

3.2 Декодирование по решеткам

Определение. Решетка ***(англ. ???)*** — граф, обладающий следующими свойствами:

1. Вершины графа разбиты на непересекающиеся подмножества, называемые **уровнями** или **ярусами**.
2. Нулевой и последний ярусы содержат по одной вершине, называемой **терминальной**.
3. Граф направленный и допускается движение только от уровня с меньшим номером к уровню с большим номером.
4. Ребрам графа сопоставлены метки, соответствующие символам кодовых слов, а также метрики, называемые весами.

На таком графе можно запустить алгоритм Дейкстры или алгоритм Витерби

[1] Не взаимозаменяющих

Определение. Профиль сложности решетки это $\xi_0 \dots \xi_n$, где $\xi_i = |V_i|$.

Определение. Решетка называется **минимальной**, если профиль сложности решетки минимален среди всех решеток с заданным количеством ярусов.

Рассмотрим все кодовые слова $c_m = c_{m,0} \dots c_{m,n-1}$ кода.

Для любого i определим префикс длины i, называемый **прошлым** и обозначим его c_m^p, а оставшийся суффикс длины $n - i$ — **будущим** и обозначим его c_m^f.

Очевидно, что в произвольной решетке пути, входящие в фиксированную вершину, имеют общее будущее, а пути, исходящие из фиксированной вершины, имеют общее прошлое.

Не дописано

Докажем, что построенная решетка минимальна.

Доказательство. Рассмотрим произвольную решетку T' этого кода.

В T' два слова $c_1 = (c_1^p, c_1^f)$ и $c_2 = (c_2^p, c_2^f)$ могут иметь общую вершину на ярусе i только если $F_i(c_1^p) = F_i(c_2^p)$. По построению два пути, проходящие через общую вершину в T', проходят также через общую вершину в T.

Таким образом, число вершин на ярусе i в T' не меньше числа вершин на ярусе i в T. □

Теорема 8. Любой код имеет минимальную решетку, и все минимальные решетки совпадают с точностью до нумерации вершин яруса.

Не дописано

Теорема 9. Решетка, получаемая по порождающей матрице в минимальной спановой форме, минимальна.

Доказательство. Докажем, что для любого $l \in \mathbb{N}$ пути, определяющие слова с одинаковыми c^f длины $n - l$, не проходят через различные узлы на ярусе с номером l.

Узел, через который проходит путь на ярусе l, определяется значениями информационных символов, которые соответствуют активным на этом ярусе строкам. Т.к. эти строки линейно независимы и заканчиваются на ярусах с номерами $> l$, следовательно, нетривиальные линейные комбинации этих строк отличаются хотя бы на одной позиции $> l$.

Предположим, что есть два слова с одинаковым будущим, проходящие через разные узлы на ярусе l. Их сумма образует слово, активное на ярусе l и равное 0 на позициях правее l. Но из соображений выше таких слов быть не может, а следовательно слова с одинаковым будущим проходят через одни и те же узлы, а следовательно решетка минимальна. □

Не дописано

Теорема 10. Решетка, построенная по проверочной матрице, минимальна.

Доказательство. Докажем, что пути с одинаковыми c^f не проходят через разные узлы.

Для кодового слова $c = (c^p, c^f)$ частичные синдромы, вычисленные по c^p и c^f, совпадают. Следовательно, все совпадающие c^f исходят из одного и того же узла, определенного частичным синдромом c^p. □

4 Декодирование с мягким выходом

Длинные коды можно строить путём комбинирования более коротких кодов. Как — мы узнаем позже. Декодеры таких кодов могут быть построены из декодеров кодов-компонент. Такие декодеры могут взаимодействовать путем обмена апостериорными вероятностями:

$$p\{c_i = a \mid y_0^{n-1}\} = \sum_{c \in C_i(a)} p\{c \mid y_0^{n-1}\}$$

???

4.1 Алгоритм Бала–Коке–Елинека–Равива

Нужно вычислить

$$L_i = \ln \frac{P\{c_i = 0 \mid y_0^{n-1}\}}{P\{c_i = 1 \mid y_0^{n-1}\}} = \ln \frac{\sum_{(s',s)\in S_0} \dfrac{p(s_i = s', s_{i+1}, y_0^{n-1})}{p(y_0^{n-1})}}{\sum_{(s',s)\in S_0} \dfrac{p(s_i = s', s_{i+1}, y_0^{n-1})}{p(y_0^{n-1})}}$$

, где S_0 и S_1 — множества пар состояний $s' \in V_i, s \in V_{i+1}$, переход между которыми помечен 0 и 1 соответственно, $p(y_0^{n-1})$ — совместная плотность распределения принятых сигналов, $p(s_i = s', s_{i+1} = s, y_0^{n-1})$ — совместная плотность распределения принятых сигналов и состояний кодера на ярусах i и $i+1$.

Поведение кодера при обработке i-го информационного бита определяется только его состоянием s' на предыдущем шаге и канал не имеет памяти:

$$p(s_i = s', s_{i+1} = s, y_0^{n-1}) = p(s_i = s', y_0^{n-1}) p(s_{i+1} = s, y_i \mid s_i = s', y_0^{y-1}) p(y_{i+1}^{n-1} \mid s_{i+1} = s, s_i = s', y_0^i)$$

$$= \underbrace{p(s_i = s', y_0^{n-1})}_{\alpha_i(s')} \underbrace{p(s_{i+1} = s, y_i \mid s_i = s')}_{\gamma_{i+1}(s',s)} \underbrace{p(y_{i+1}^{n-1} \mid s_{i+1} = s)}_{\beta_{i+1}(s)}$$

Примечание. Предыдущая строка — определения α, β, γ

Лекция 5

30 сентября

По формуле Байеса:

$$\alpha_i(s) = \sum_{\tilde{s} \in V_{i-1}} \alpha_{i-1}(\tilde{s})\gamma_i(\tilde{s}, s), s \in V_i$$

$$\beta_i(\tilde{s}) = \sum_{s \in V_{i+1}} \gamma_{i+1}(\tilde{s}, s)\beta_{i+1}(\tilde{s})$$

Не дописано

Начальные значения для рекуррентной формулы:

$$\beta'_n(s) = \beta_n(s) = \begin{cases} 1, & s = 0 \\ 0, & s \neq 0 \end{cases}$$

$$\gamma_{i+1}(s', s) = ???$$

Не дописано

4.2 Выводы

Метод порядковых статистик позволяет выполнить мягкое декодирование произвольного линейного блокового кода. Увеличение сложности позволяет приблизиться к декодированию по максимум правдоподобия.

Всякий линейный блоковый код может быть представлен в виде решетки.

Алгоритм Витерби реализует декодирование линейного кода по максимуму правдоподобия.

Алгоритм Бала-Коке-Елинека-Равива реализует посимвольное декодирование линейного кода по максимуму апостериорной вероятности.

5 Сверточные коды

Задача кодера — сделать передаваемые символы статистически зависимыми. Сверточный код отображает автоматом блоки данных в кадры кодового слова.

Пример. Простейший автомат — регистр сдвига.

Не дописано

Лекция 6

7 октября

6 Методы модификации и комбинирования кодов

Мы не хотим строить сложные коды целиком, проще комбинировать коды поменьше.

6.1 Конструкция Плоткина

Пусть дано два кода C_1 и $C_2 : (n, k_i, d_i)$. Построим тогда код $C = \{(c_1, c_1 + c_2 \mid c_i \in C_i, i \in \{1, 2\})\}$. Тогда этот код $(2n, k_1 + k_2, \min(2d_1, d_2))$.

Порождающая матрица этого кода, если G_1 — порождающая матрица для C_1 и G_2 — для C_2:

$$G = \begin{pmatrix} G_1 & G_1 \\ 0 & G_2 \end{pmatrix}$$

Доказательство. C содержит кодовые слова вида $(c_1, c_1) \mid c_1 \in C_1$ и вида $(0, c_2) \mid c_2 \in C_2$, следовательно, $d \leq 2d_1$ и $d \leq d_2$

Пусть $c_1, c_1' \in C_1 \setminus \{0\}$ и $c_2, c_2' \in C_2 \setminus \{0\}$ — ненулевые слова компонентных кодов.

$$d((c_1, c_1 + c_2), (c_1', c_1' + c_2')) = d(c_1, c_1') + d(c_1 + c_2, c_1' + c_2')$$

???

□

Будем декодировать $(y', y'') = (c_1, c_1 + c_2) + (e', e'')$ в метрике Хэмминга.

$$y''' = y'' - y' = c_1 + c_2 + e'' - c_1 - e' = c_2 + e'''$$

Продекодируем получившийся y''' в коде C_2. Если $\mathrm{wt}((e', e'')) \leq \lfloor (d-1)/2 \rfloor$, то $\mathrm{wt}(e''') \leq \mathrm{wt}(e') + \mathrm{wt}(e'') \leq \lfloor (d-1)/2 \rfloor \leq \lfloor (d_2 - 1)/2 \rfloor$ и декодирование успешно, т.е. мы получим c_2.

Предположим, что c_2 получено правильно. Продекодируем в C_1 вектора $y' = c_1 + e'$ и $y'' - c_2 = c_1 + e$. Если $\mathrm{wt}((e', e'')) \leq$??? , то декодирование y' или y'' даст правильный результат.

6.1.1 Коды Рида-Маллера

Это семейство кодов определяется рекурсивным применением конструкции Плоткина. Код Рида-Маллера порядка r длины 2^m обозначается как $RM(r, m)$.

Код $RM(0, m)$ — код $(2^m, 1, 2^m)$. Код $RM(m, m)$ — $(2^m, 2^m, 1)$.

Код $RM(r+1, m+1)$ получается применением конструкции Плоткина к $RM(r+1, m)$ и $RM(r, m)$.

Недавно было доказано, что коды Рида-Маллера оптимальны в том смысле, что он достигает границы Шеннона для двоичного канала с АБГШ и некоторых других каналов. Проблема этих кодов в том, что их сложно эффективно декодировать.

6.2 Модификации кодов

6.2.1 Укорочение

Укороченный код получается путем выбора кодовых слов исходного кода, содержащий нули на заданных позициях. Эти нули удаляются.

Если дан (n, k, d) код с $G = (I \mid A)$, то мы удаляем из порождающей матрицы m столбцов единичной подматрицы и соответствующие им m строк. Если появятся нулевые столбцы, то мы их тоже удаляем. Таким образом получится порождающая матрица $(\leq n - m, k - m, \geq d)$ кода.

6.2.2 Выкалывание

Удалим из всех кодовых слов заданные символы (обычно проверочных ???).

Если дан (n, k, d) код с $H = (A \mid -I)$, то удалим из H m столбцов единичной подматрицы и соответствующие им m строк. Если появятся линейно зависимые строки, то мы их тоже удаляем. Тогда мы получим $(n - m, \leq k, \geq d - m)$ код.

Пример: таким образом можно построить оптимальный код $(10, 3, 5)$.

6.2.3 Расширение

Наиболее распространенный способ — проверки на четность, так мы получаем из (n, k, d) кода $(n+1, k, d')$ код.

Если d нечетно, то $d' = d + 1$, иначе[1] $d' = d$.

6.3 Каскадные коды

Перемежитель переставляет символы таким образом, чтобы последствия ошибочного декодирования одного кода могут быть легко ликвидированы декодером другого кода.

6.4 Прямое произведение кодов

Для двух заданных кодов (n_1, k_1, d_1) и (n_2, k_2, d_2) построим прямое произведение этих двух кодов — один из кодов будет декодировать по строчкам, а другой — по столбцам.

Порождающая матрица такого кода будет иметь вид

$$G' \oplus G'' = \begin{pmatrix} G'_{11}G'' & G'_{12}G'' & \dots & G'_{1n_1}G'' \\ G'_{21}G'' & G'_{22}G'' & \dots & G'_{2n_1}G'' \\ \vdots & \vdots & \ddots & \vdots \\ G'_{k_11}G'' & G'_{k_12}G'' & \dots & G'_{k_1n_1}G'' \end{pmatrix}$$

Пропускная способность такого кода будет равна

$$R = \frac{k_1 k_2}{n_1 n_2} < \frac{k_1}{n_1}, \frac{k_2}{n_2}$$

Код способен исправить многие *(но не все)* ошибки веса большего, чем $d_1 d_2/2$.

Алгоритм декодирования можно паралеллизировать.

6.4.1 Пример: код Рао-Редди $(48, 31, 8)$

РЖД использует этот код для передачи данных машинисту.

Рассмотрим прямое произведение расширенного $(16, 11, 4)$ кода Хемминга C_1 и $(3, 2, 2)$ кода C_2 с проверкой на четность. Получится код $(48, 22, 8)$.

Дополним этот код кодовыми словами кода Рида-Маллера $(16, 5, 8)$ C_3, к которым дописали 32 нуля в конец для размерности.

Тогда кодовые слова будут иметь вид $(c_1 + c_3, c_2, c_1 + c_2) \mid c_1, c_2 \in C_1, c_3 \in C_3$.

Вес кодового слова ≥ 8, т.к.:

- ???
- ???

[1] Мне так кажется

Теперь получен код $(48, 27, 8)$.

Пусть C_4 — код с порождающей матрицей ???.

Итого код $(48, 31, 8)$ Рао-Редди состоит из кодовых слов вида:

$$(c_1 + c_3 + c_4, c_2 + c_4, c_1 + c_2 + c_4) \mid c_1, c_2 \in C_1, c_3 \in C_3, c_4 \in C_4$$

Вывод: создание кодов больше похоже на искусство, непонятно что лучше сработает.

6.5 Лестничные коды

???

6.6 Граница Зяблова

Выберем внутренний (n, k, d) код на границе Варшамова–Гилберта с $r = k/n \geq 1 - h(d/n) = 1 - \delta$.

???

6.7 Обобщенные каскадные коды (Блох, Зяблов, Зиновьев)

??? внешние (N, K_i, D_i) коды $\mathcal{A}_i$ над $GF(q^{m_i}) \mid 1 \leq i \leq s$.

??? вложенные внутренние (n, k_i, d_i) коды $\mathcal{B}_i : \mathcal{B}_1 \supset \mathcal{B}_2 \supset \cdots \supset \mathcal{B}_s$ над $GF(q)$, при этом $k_i - k_{i+1} = m_i$ и код $\mathcal{B}_i$ порождается последними k_i строками $k_1 \times m$ матрицы B.

Кодирование: ???

???

Декодирование: ???

6.8 Турбо–коды

Будем кодировать данные сразу двумя сверточными кодами.

???

Минимальное расстояние растет с увеличением длины крайне медленно и начиная с некоторого места перестает расти. Поэтому использовать длинные турбо–коды неразумно.

???

6.9 Построение перемежителей

Давайте найдем, как перемежитель должен переставлять символы.

Будем переставлять близкие позиции во входной последовательности в максимально удаленные позиции в выходной последовательности:

$$0 < |i - j| < d \Rightarrow |\pi(i) - \pi(j)| \geq S$$

- Можно генерировать случайные перестановки и отбрасывать неэффективные. Но это долго.
- Табличный перемежитель: запишем все символы в квадратную матрицу по строкам, а прочитаем ее по столбцам.
- Перестановочный полином: если ???

Перемежитель в целом нужен когда мы делаем композицию декодеров, т.к. канал с декодером на нем уже не является каналом с АБГШ.

???

Лекция 7

14 октября

7 Полярные коды

7.1 Некоторые определения

7.1.1 Функция переходных вероятностей канала

Определение. Рассмотрим канал без памяти с входным алфавитом $\mathcal{X} = \{0, 1\}$ и выходным алфавитом $\mathcal{Y}$.

Если $\mathcal{Y}$ дискретен, то **функция переходных вероятностей канала** $W(y \mid c)$ — вероятность наблюдения на выходе $y \in \mathcal{Y}$ при условии подачи на его вход $c \in \mathcal{X}$.

Если $\mathcal{Y}$ непрерывен, то **функция переходных вероятностей канала** $W(y \mid c)$ — плотность распределения выходного символа при подаче c на его вход

Пример (Двоичный симметричный канал).

$$\mathcal{Y} = \mathcal{X} \quad W(y \mid c) = \begin{cases} p, & y \neq x \\ 1 - p, & y = x \end{cases}$$

Пример (Двоичный стирающий канал).

$$\mathcal{Y} = \{0, 1, \varepsilon\} \quad W(y \mid c) = \begin{cases} p, & y = \varepsilon \\ 1 - p, & y = x \in \{0, 1\} \end{cases}$$

Пример (Двоичный симметричный канал со стираниями).

$$\mathcal{Y} = \{0, 1, \varepsilon\} \quad W(y \mid c) = \begin{cases} 1 - p - s, & y = x \\ s, & y = \varepsilon \\ p, & y \neq x, y \neq \varepsilon \end{cases}$$

Пример (Аддитивный гауссовский канал).

$$\mathcal{Y} = \mathbb{R}, y = (-1)^c + \eta, \eta \sim \mathcal{N}(0, \sigma^2) \quad W(y \mid c) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y-(-1)^c)^2}{2\sigma^2}\right)$$

7.1.2 Параметр Бхаттачарьи

Рассмотрим канал без памяти с двоичным кодом, приемник по максимуму правдоподобия.

Если передаваемые символы равновероятны, то вероятность ошибки:

$$\begin{aligned} P_e &= P\{c=0\}P\{\text{err} \mid c=0\} + P\{c=1\}P\{\text{err} \mid c=1\} \\ &= \frac{1}{2} \sum_{y:W(y|0)<W(y|1)} W(y \mid 0) + \frac{1}{2} \sum_{y:W(y|1)<W(y|0)} W(y \mid 1) \\ &= \frac{1}{2} \sum_{y:\frac{W(y|1)}{W(y|0)}>1} W(y \mid 0) + \frac{1}{2} \sum_{y:\frac{W(y|0)}{W(y|1)}>1} W(y \mid 1) \\ &= \frac{1}{2} \sum_{y \in \mathcal{Y}} \sum_{c \in \{0,1\}} \left(W(y \mid c) \chi\left(\frac{W(y \mid 1-c)}{W(y \mid c)}\right)\right) \end{aligned}$$

, где χ — индикаторная функция:

$$\chi(z) = \begin{cases} 1, z \geq 1 \\ 0, z < 1 \end{cases}$$

Заметим, что $\chi(z) \leq \sqrt{z}$ для всех $z \in \mathbb{N}$. Таким образом, можно оценить вероятность ошибки:

$$\begin{aligned} P_e &\leq \frac{1}{2} \sum_{y \in \mathcal{Y}} \sum_{c \in \{0,1\}} W(y \mid c) \sqrt{\frac{W(y \mid 1-c)}{W(y \mid c)}} \\ &= \frac{1}{2} \sum_{y \in \mathcal{Y}} \sum_{c \in \{0,1\}} \sqrt{W(y \mid 1-c) W(y \mid c)} \\ &= \sum_{y \in \mathcal{Y}} \sqrt{W(y \mid 1) W(y \mid 0)} \\ &=: Z(W) \end{aligned}$$

Определение. $Z(W)$ — **параметр Бхаттачарьи** ФПВК W.

Пример (Двоичный стирающий канал).

$$Z(BEC(p)) = \sqrt{W(0 \mid 0)W(0 \mid 1)} + \sqrt{W(1 \mid 0)W(1 \mid 1)} + \sqrt{W(\varepsilon \mid 0)W(\varepsilon \mid 1)} = 0 + 0 + p = p$$

Пример (Аддитивный гауссовский канал).

$$\begin{aligned} Z(\mathcal{G}(\sigma)) &= \int_{-\infty}^{\infty} \sqrt{W(y \mid 0)W(y \mid 1)}dy \\ &= \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \exp\left(-\frac{(y-1)^2 + (y+1)^2}{4\sigma^2}\right)dy \\ &= \exp\left(-\frac{1}{2\sigma^2}\right) \end{aligned}$$

Пропускную способность канала можно вычислить по формуле:

$$I(W) = \max_{\{p(x)\}} \sum_{x\in\mathcal{X}} \sum_{y\in\mathcal{Y}} W(y \mid x)P\{x\} \log \frac{W(y \mid x)}{W(y)}$$

Для многих каналов оптимальным распределением символов на входе $P\{x\}$ является равномерное, что мы и будем рассматривать дальше.

7.2 Поляризация канала

Рассмотрим следующее линейное преобразование:

$$\begin{pmatrix} c_0 & c_1 \end{pmatrix} = \begin{pmatrix} u_0 & u_1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

Пропустим полученные символы через двоичный стирающий канал с вероятностью стирания p, получим y_0, y_1.

Если оба символа не стерты, то u_0 можно восстановить из y_0, y_1 как $y_0 \oplus y_1$. Вероятность того, что этого не произойдет — $1 - (1-p)^2 = 2p - p^2 \geq p$

u_1 восстанавливается либо как y_1, либо как $y_0 \oplus u_0$ и тогда нам нужна подсказка свыше о u_0. Вероятность того, что восстановить не получится — вероятность того, что оба символа стерты, т.е. $p^2 \leq p$.

Таким образом, мы с помощью **поляризации** получили два виртуальных канала — один чуть получше, другой чуть похуже, чем исходный канал.

Пусть $n = 2^m$, $A_m = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}^{\oplus m}$, где $\oplus m$ — m-кратное произведение Кронекера матрицы с собой. Т.к. канал без памяти, выполняется $W_m(y_0^{n-1} \mid c_0^{n-1}) = \prod_{i=0}^{n-1} W(y_i \mid c_i)$.

Вход канала — u_i, выход — реальный выход канала y_0^{n-1} и подсказка свыше о предыдущих символах u_0^{i-1}.

Определим ФПВ для синтетических каналов:

$$\begin{aligned}
W_m^{(i)}(y_0^{n-1}, u_0^{i-1} \mid u_i) &\stackrel{\text{def}}{=} \frac{W_m^{(i)}(y_0^{n-1}, u_0^i)}{P\{u_i\}} \\
&= \frac{\sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} W_m^{(i)}(y_0^{n-1} \mid u_0^{n-1}) P\{u_0^{n-1}\}}{\frac{1}{2}} \\
&= 2 \sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} W_m^{(i)}(y_0^{n-1} \mid u_0^{n-1}) P\{u_0^{n-1}\} \\
&= \frac{2}{2^n} \sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} W_m^{(i)}(y_0^{n-1} \mid u_0^{n-1}) \\
&= \frac{2}{2^n} \sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} W_m^{(i)}(y_0^{n-1} \mid u_0^{n-1}) \\
&= \frac{2}{2^n} \sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} W_m(y_0^{n-1} \mid u_0^{n-1} A_m) \\
&= 2^{1-n} \sum_{u_{i+1}^{n-1} \in \mathbb{F}_2^{n-i}} \prod_{j=0}^{n-1} W(y_j \mid (u_0^{n-1} A_m)_j)
\end{aligned}$$

Вычислим ФПВ битовых подканалов, для начала для простейшего случая $n = 0$:

$$\begin{aligned}
W_1^{(0)}(y_0, y_1 \mid u_0) &= \frac{W_1^{(0)}(y_0, y_1, u_0)}{P\{u_0\}} \\
&= 2 \sum_{u_1=0}^{1} W_1^{(1)}(y_0, y_1, u_0, u_1) \\
&= 2 \sum_{u_1=0}^{1} W_1^{(1)}(y_0, y_1 \mid u_0, u_1) P\{u_0, u_1\} \\
&= \frac{1}{2} \sum_{u_1=0}^{1} W(y_0 \mid u_0 + u_1) + W(y_1 \mid u_1)
\end{aligned}$$

$$W_1^{(0)}(y_0, y_1 \mid u_0) = \frac{1}{2} \sum_{u_1=0}^{1} W(y_0 \mid u_0 + u_1) W(y_1 \mid u_1)$$

$$W_1(y_0, y_1, u_0 \mid u_1) = \frac{1}{2} W(y_0 \mid u_0 + u_1) W(y_1 \mid u_1)$$

Из тех же самых соображений получаем рекурсивное определение:

$$W_\lambda^{(2i)}(y_0^{2^\lambda - 1}, u_0^{2i-1} \mid u_{2i}) = \frac{1}{2} \sum_{u_{2i+1}=0}^{1} W_{\lambda-1}^{(i)}(y_{0,even}^{2^\lambda - 1}, u_{0,even}^{2i-1} + u_{0,odd}^{2i-1} \mid u_{2i} + u_{2i+1}) W_{\lambda-1}^{(i)}(y_{0,odd}^{2^\lambda - 1}, u_{0,odd})$$

$$W_\lambda^{(2i+1)}(y_0^{2^\lambda-1}, u_0^{2i} \mid u_{2i+1}) = \frac{1}{2} W_{\lambda-1}^{(i)}(y_{0,even}^{2^\lambda-1}, u_{0,even}^{2i-1} + u_{0,odd}^{2i-1} \mid u_{2i} + u_{2i+1}) W_{\lambda-1}^{(i)}(y_{0,odd}^{2^\lambda-1}, u_{0,odd}^{2i-1} \mid u_{2i+1})$$

7.2.1 Параметры подканалов

Параметры Бхаттачарьи для битовых подканалов $Z_{m,i} = Z(W_m^{(i)})$ (без доказательства):

$$Z_{m,2i+1} \le Z_{m,2i} \le 2Z_{m-1,i} - Z_{m-1,i}^2$$

$$Z_{m,2i+1} = Z_{m-1,i}^2$$

При этом в случае двоичного стирающего канала равенство, а не $\le$.

Пропускные способности:

$$I_{m,2i} + I_{m,2i+1} = 2I_{m-1,i}$$

$$I_{m,2i} \le I_{m,2i+1}$$

И еще пропускные способности связаны с параметрами Бхаттачарьи следующим образом:

$$\sqrt{1 - Z(W)^2} \ge I(W) \ge \log \frac{2}{1 + Z(W)}$$

Теорема 11. Для любого $\delta \in (0, 1)$ при $m \to \infty$ доля подканалов с $I(W_m^{(i)}) \in (1 - \delta, 1]$ стремится к $I(W_0^{(0)}) = I(W)$, т.е. пропускной способности исходного канала, а доля подканалов с $I(W_m^{(i)}) \in [0, \delta)$ стремится к $1 - I(W)$.

Т.е. синтетические подканалы сходятся или к очень хорошим, или очень плохим каналам.

Это верно не только для двоичного стирающего канала.

7.3 Полярный код

Будем посылать по плохим каналам предопределенные символы, например нули. По остальным каналам посылаем полезные данные. Тогда кодирование имеет вид $c_0^{n-1} = u_0^{n-1} A_m$, где $u_i = 0$ при $i \in \mathcal{F}$ — множество плохих каналов или соответственно замороженных символов.

Таким образом мы получаем линейный блоковый код длины 2^m с размерностью $2^m - |\mathcal{F}|$.

Декодирование выполняется алгоритмом последовательного исключения:

1. Для замороженных символов значения известны.
2. Для не замороженных символов используется оценка $\hat{u}_i = \operatorname{argmax}_{u_i} W_m^{(i)}(y_0^{n-1}, \hat{u}_0^{i-1} \mid u_i)$. В формулу надо бы подставить истинные значения u_0^{i-1}, но мы их не знаем. Если u_0^{i-1} было найдено верно, то мы получим верное значение u_i, а если мы ошиблись, то мы и так уже проиграли.

Вероятность ошибки можно оценить как:

$$P \leq \sum_{i \notin F} Z_{m,i} \leq 2^{-n^\beta}, \beta < 0.5$$

Указать точное число исправимых ошибок тяжело, и не очень полезно, потому что бывает такое: гарантированно исправляем три ошибки, но с высокой вероятностью исправим и 10.

Чтобы кодировать, нужно умножать u на A_m, т.е.:

$$u_0^{n-1} A_m = \begin{pmatrix} u_0^{n/2-1} & u_{n/2}^{n-1} \end{pmatrix} \begin{pmatrix} A_{m-1} & 0 \\ A_{m-1} & A_{m-1} \end{pmatrix} = \begin{pmatrix} (u_0^{n/2-1} + u_{n/2}^{n-1}) A_{m-1} & u_{n/2}^{n-1} A_{m-1} \end{pmatrix}$$

Тогда сложность кодирования:

$$T(n) = \underbrace{2T\left(\frac{n}{2}\right)}_{\text{Умножение на матрицу}} + \underbrace{\frac{n}{2}}_{\text{Сложение векторов}} = \frac{1}{2} n \log_2 n$$

Определим логарифмическое отношение правдоподобия:

$$L_m^{(i)}(y_0^{n-1}, u_0^{i-1}) := \ln \frac{W_m^{(i)}(y_0^{n-1}, u_0^{i-1} \mid 0)}{W_m^{(i)}(y_0^{n-1}, u_0^{i-1} \mid 1)}$$

Тогда подстановкой рекурсивного определения $W_m^{(i)}$ можно получить для нечетных каналов:

$$L_\lambda^{(2i+1)}(y_0^{n-1}, u_0^{2i}) = ??? = (-1)^{u_{2i}} L_{\lambda-1}(y_{0,even}^{2^\lambda-1}, u_{0,even}^{2i-1} + u_{0,odd}^{2i-1}) + L_{\lambda-1}(y_{0,odd}^{2^\lambda-1}, u_{0,odd}^{2i-1})$$

Пусть при этом:

$$\begin{aligned} p_s &= W_\lambda^{(2i)}(s \mid y_0^{2^\lambda-1}, u_0^{2i-1}) = \frac{W_\lambda^{(2i)}(y_0^{2^\lambda-1}, u_0^{2i-1} \mid s)}{2 W_\lambda^{(2i)}(y_0^{2^\lambda-1}, u_0^{2i-1})} \\ p_{0s} &= W_{\lambda-1}^{(i)}(s \mid y_{0,even}^{2^\lambda-1}, u_{0,even}^{2i-1} + u_{0,odd}^{2i-1}) \\ p_{1s} &= W_{\lambda-1}^{(i)}(s \mid y_{0,odd}^{2^\lambda-1}, u_{0,odd}^{2i-1}) \end{aligned}$$

Тогда рекурсивные формулы записываются как:

$$\begin{aligned} p_0 &= p_{00} p_{10} + p_{01} p_{11} \\ p_1 &= p_{01} p_{10} + p_{00} p_{11} \end{aligned}$$

, причем:

$$p_0 + p_1 = 1$$
$$p_{i0} + p_{i1} = 1$$

Забавный факт:

$$\begin{aligned}
\tanh\left(\frac{1}{2}\ln\frac{p_0}{p_1}\right) &\stackrel{\text{def}}{=} \frac{\exp\left(\ln\left(\frac{p_0}{p_1}\right)\right) - 1}{\exp\left(\ln\left(\frac{p_0}{p_1}\right)\right) + 1} \\
&= \frac{\frac{p_0}{p_1} - 1}{\frac{p_0}{p_1} + 1} \\
&= \frac{p_0 - p_1}{p_0 + p_1} \\
&= p_0 - p_1 \\
&= 1 - 2p_1 \\
&= (1 - 2p_{01})(1 - 2p_{11}) \\
&= 1 - 2(p_{01} + p_{11} - 2p_{11}p_{01}) \\
&= 1 - 2(p_{01}(1 - p_{11}) + (1 - p_{01})p_{11})
\end{aligned}$$

???

Алгоритм последовательного исключения:

$$\hat{u}_i = \begin{cases} 0, & i \in \mathcal{F} \\ 0, & L_m^{(i)}(y_0^{n-1}, \hat{u}_0^{i-1}) > 0, i \notin \mathcal{F} \\ 1, & L_m^{(i)}(y_0^{n-1}, \hat{u}_0^{i-1}) \leq 0, i \notin \mathcal{F} \end{cases}$$

Есть также другой вариант алгоритма, где поменяны местами u_0^i и y_0^{n-1}:

$$W_m^{(i)}\left(u_0^i \mid y_0^{n-1}\right) = \frac{W_m^{(i)}\left(y_0^{n-1}, \widehat{u}_0^{i-1} \mid u_i\right)}{2W\left(y_0^{n-1}\right)} = \sum_{u_{i+1}^{n-1}} \prod_{j=0}^{n-1} W\left(\left(u_0^{n-1}A_m\right)_j \mid y_j\right)$$

$$W_\lambda^{(2i)}\left(u_0^{2i} \mid y_0^{n-1}\right) = \sum_{u_{2i+1}} W_{\lambda-1}^{(i)}\left(u_{2t} + u_{2t+1}, 0 \leq t \leq i \mid y_{0,\,\text{even}}^{n-1}\right) W_{\lambda-1}^{(i)}\left(u_{2t+1}, 0 \leq t \leq i \mid y_{0,\,\text{odd}}^{n-1}\right)$$

$$W_\mu^{(2i+1)}\left(u_0^{2i+1} \mid y_0^{n-1}\right) = W_{\lambda-1}^{(i)}\left(u_{2t} + u_{2t+1}, 0 \leq t \leq i \mid y_{0,\,\text{even}}^{n-1}\right) W_{\lambda-1}^{(i)}\left(u_{2t+1}, 0 \leq t \leq i \mid y_{0,\,\text{odd}}^{n-1}\right)$$

Если мы хотим получить код размерности k, то необходимо заморозить $2^m - k$ наименн надежных символов, например с наибольшим $Z_{m,i}$.

В случае двоичного стирающего канала $Z_{m,i}$ вычисляется просто — рекурсией с мемоизацией, со сложностью $\mathcal{O}(n)$.

В общем случае выходной алфавит канала $W_m^{(i)}(y_0^{n-1}, u_0^{i-1} \mid u_i)$ имеет мощность $|\mathcal{Y}|^n 2^i$, а поэтому выписывать (даже не вычислять) все вероятности нереализуемо за адекватное время. Однако можно аппроксисмировать $W_m^{(i)}$ каналом с выходным алфавитом фикированной мощности, который чуть лучше или хуже, чем настоящий канал. С помощью этого метода $Z_{m,i}$ можно вычислить за $\mathcal{O}(n\mu^2 \log \mu)$, где μ — мощность выходного алфавита канала.

Можно поступить проще, заметив, что для симметричных каналов вероятность ошибки не зависит от того, какое кодовое слово передавалось, а т.к. полярные коды линейные, то нулевое слово является кодовым. Тогда будем считать, что передавалось нулевое слово.

Пример. При передаче кодовых слов по аддитивному гауссовскому каналу выполнено $L_0^{(0)}(y_i) = \frac{2y_i}{\sigma^2}$. Т.к. мы рассматриваем нулевое кодовое слово, то получается, что $M[L_0^{(0)}(y_i)] = \mu_{00} = \frac{2}{\sigma^2}$ и $D[L_0^{(0)}(y_i)] = \frac{4}{\sigma^2} = 2M$.

Предположим, что все логарифмические отношения правдподобий имеют нормальное распределение[1]

$$\mathcal{L}_\lambda^{(i)} \sim \mathcal{N}(\mu_{\lambda,i}, 2\mu_{\lambda,i}), \quad 0 \le i < 2^\lambda, 0 \le \lambda \le m$$

Можем получить рекурсивную формулу для матожиданий (без доказательства):

$$\begin{aligned} \mu_{\lambda,2i} &= \Theta(\mu_{\lambda-1,i}) = \phi(1 - (1 - \phi(\mu_{\lambda-1,i}))^2) \\ \mu_{\lambda,2i+1} &= 2\mu_{\lambda-1,i} \end{aligned}$$

, где:

$$\phi(x) = 1 - \frac{1}{\sqrt{4\pi x}} \int_{-\infty}^{\infty} \tanh \frac{u}{2} \exp\left(-\frac{(u-x)^2}{4x}\right) \mathrm{d}u$$

Замораживаются символы с наименьшим $\mu_{m,i}$.

Θ считать долго, поэтому есть аппроксимация:

$$\Theta(x) \approx \begin{cases} 0.9861x - 2.3152, & x > 12 \\ x(9.005 \cdot 10^{-3}x + 0.7694) - 0.9507, & x \in (3.5, 12] \\ x(0.062883x + 0.3678) - 0.1627, & x \in (1, 3.5] \\ x(0.2202x + 0.06448), & \text{otherwise} \end{cases}$$

Несмотря на свое определение, код Рида-Маллера $RM(r, m)$ длины 2^m порядка r — полярный код с

$$\mathcal{F} = \{i \mid 0 \le i < 2^m, \mathrm{wt}(i) < m - r\}$$

[1] Это неверно в общем случае, но аппроксимация все равно получается неплохая.

Доказательство. Рассмотрим $m = 1$. Тогда $RM(0,1)$ — код $(2,1,2)$ и $RM(1,1)$ — код $(2,2,1)$ с порождающими матрицами $\begin{pmatrix}1 & 1\end{pmatrix}$ и $\begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}$ соответственно. Несложно заметить, что искомое выполнено.

Индукицонный переход: пусть искомое верно для $RM(r, m-1)$ и $RM(r-1, m)$. Тогда пусть

$$\mathcal{F}_{r,m} = F_{r,m-1} \cup \{2^{m-1} + i \mid i \in \mathcal{F}_{r-1,m}\}$$

Тогда $wt(i) < m - 1 - r + 1 = m - 1$. □

Лекция 8

21 октября

Теорема 12. Минимальное расстояние полярного кода длины $n = 2^m$ с замороженным множеством $\mathcal{F}$ равно $\min_{i \notin \mathcal{F}} 2^{\mathrm{wt}(i)} = \min_{i \notin \mathcal{F}} \mathrm{wt}(A_m^i)$

, где A_m^i — i–тая строка матрицы A_m.

Минимальное расстояние полярного кода сильно хуже, чем минимальное расстояние расширенного БЧХ или кода Рида-Маллера, например. Но для этих кодов нет быстрого алгоритма декодирования, а для полярного кода есть.

Когда алгоритм последовательного исключения доходит до какого-то информационного символа, то при принятии решения об этом символе никак не учитываются последующие замороженные символы. Кроме того, если алгоритм допустил ошибку, то он не может ее исправить. Будем эти проблемы чинить.

7.3.1 Списочное декодирование

Не будем принимать окончательное решение о u_i на фазе i, вместо этого на каждой фазе i будем рассматривать L (константа) векторов u_0^{i-1}, строить возможные продолжения для них и выбирать из них L наиболее вероятных.

Тогда вектора u_0^i задают пути в кодовом дереве.

Не дописано

Лекция 9

28 октября

8 Циклические коды

Не дописано

Теорема 13. Подмножество $\mathcal{C} \in {}^{\mathbb{F}[x]}\!/_{(x^n-1)}$ образует циклический код тогда и только тогда, когда:

1. $\mathcal{C}$ образует группу по сложению
2. Если $c(x) \in \mathcal{C}$ и Не дописано

Определение. Порождающий многочлен циклического кода — ненулевой кодовый многочлен $g(x) \in \mathcal{C}$ наименьшей степени с коэффициентом при старшем члене 1.

Лемма 1. Все кодовые слова $c(x)$ в циклическом коде делятся на $g(x)$.

Доказательство. Предположим противное, т.е. $c(x) = a(x)g(x) + r(x), r(x) \in \mathcal{C}$. Но $\deg r(x) < \deg g(x)$, что противоречит предположению о минимальности степени $g(x)$. □

Порождающий многочлен циклического кода единственнен.

Лемма 2. Циклический код длины n с порождающим многочленом $g(x)$ существует тогда и только тогда, когда $g(x) \mid (x^n - 1)$

Доказательство.

“⇒” Пусть $g(x)$ не делится на $(x^n - 1)$. Тогда $x^n - 1 = a(x)g(x) + r(x), \deg r < \deg g$ Не дописано

“⇐” В качестве порождающего многочлена можно выбрать любой делитель $x^n - 1$.

□

Определение. Т.к. $g(x) \mid (x^n - 1)$, то $(x^n - 1) = g(x)h(x)$. $h(x)$ называется **проверочным многочленом**.

Несистематическое кодирование Не дописано

Определение. Систематическое кодирование информационных символов $a_0 \dots a_{k-1}$ в $c_{n-k} \dots c_{n-1}$:

$$c(x) = x^{n-k}a(x) - r(x)$$

Определим $r(x)$ как:

$$r(x) \equiv x^{n-k}a(x) \mod g(x), \quad \deg r < \deg g$$

Каждый метод кодирования можно представить в матричном виде, и каждому из них соответствует своя порождающая матрица. Матрицы различных методов выражаются друг через друга как $G' = QG$, где Q — некоторая обратимая матрица.

Какой бы метод кодирования не использовался, корректирующая способность остается одинаковой.

9 Элементы общей алгебры

Определение. Поле — Не дописано.

Определение. Поля, содержащие конечное число элементов q называются **конечными полями** или **полями Галуа** $GF(q)$.

Определение. Бесконечные поля называются полями **характеристики 0**.

Определение. Поле является **областью целостности**, т.к. если допустить $\exists a, b \neq 0 : ab = 0$, то $0 = (((ab)b^{-1})a^{-1}) = 1$.

Поле $GF(p)$, где p простое, суть арифметика по модулю p.

Характеристика поля — Не дописано.

Определение. Группа называется **циклической**, если $\exists y \in G$ такой, что любой ее элемент может быть получен как $x = y^i$ для некоторого $i \in \mathbb{N}$.

Определение. Порядок элемента группы $x \in G$ — минимальное число $i \in \mathbb{N}_+$, такое что $x^i = 1$.

Определение. Порядок группы — порядок образующего элемента группы.

Теорема 14 (Лагранж)**.** Порядок любой конечной группы делится на порядок любой ее подгруппы.

Теорема 15. Пусть $\mathcal{G} = (G, \cdot)$ — конечная группа и элементы $g, h \in G$ имеют порядок r, s соответственно, причем $\gcd(r, s) = 1$. Тогда gh имеет порядок rs.

Доказательство. Очевидно, что $(gh)^{rs} = 1$. Следовательно, порядок p элемента gh — делитель числа rs. Таким образом, $p \mid rs$ и $(gh)^p = 1$, следовательно, $(gh)^{pr} = 1 \cdot h^{pr} = 1$. Следовательно, $s \mid ps \Rightarrow s \mid p$. Аналогично $r \mid p$.

Т.к. $\gcd(r, s) = 1$, получаем $rs \mid p \Rightarrow p = rs$ □

Теорема 16. Пусть $\mathbb{F}$ — поле из q элементов. Тогда $q = p^m$, где p — простое, а $m \in \mathbb{N}$.

Доказательство. Элемент $1 \in \mathbb{F}$ образует аддитивную циклическую подгруппу простого порядка поля $\mathbb{F} \Rightarrow p \mid q \xRightarrow{\text{Лагранж}} GF(p) \subset \mathbb{F}$.

Будем называть элементы $\alpha_1 \dots \alpha_m$ линейно независимыми с коэффициентами из $GF(p)$, если:

$$\left\{ (x_1 \dots x_m) \in GF(p^m) \mid \sum_{i=1}^{m} x_i \alpha_i = 0 \right\} = \{(0 \dots 0)\}$$

Среди всех ЛНЗ подмножеств $\mathbb{F}$ выделим подмножество $\{\alpha_1 \dots \alpha_m\} \subset \mathbb{F}$ с максимальным числом элементов, тогда:

$$\forall \alpha_0 \in \mathbb{F} \;\; \exists x_1 \dots x_m \in GF(p) : \alpha_0 = \sum_{i=1}^{m} x_i \alpha_i$$

При этом различные $x_1 \dots x_m$ приводят к различным $\alpha_0 \in \mathbb{F} \Rightarrow |\mathbb{F}| = p^m$ □

- $GF(q^m)$ образует m-мерное линейное пространство над полем $GF(q)$
- $GF(p^m), m > 1$ — расширенное конечное поле, где m — степень расширение.
- $GF(p^m) \neq \mathbb{Z}_{p^m}$

Теорема 17. Ненулевые элементы $GF(q)$ образуют конечную циклическую группу по умножению.

Доказательство. По аксиомам поля $\mathbb{F} \setminus \{0\}$ образует конечную группу по умножению.

Выберем в $\mathbb{F} \setminus \{0\}$ элемент α, называемый **примитивным**, с наибольшим порядком r. Пусть l — порядок некоторого элемента $\beta \neq 0$.

Не дописано □

Свойства (Конечные поля).

- Для всякого ненулевого $\beta \in GF(q)$ выполняется $\beta^{q-1} = 1$
- Все элементы поля $GF(q)$ удовлетворяют уравнению $x^q - q = 0$
- Порядок любого ненулевого $\beta \in GF(q)$ делит $q - 1$.

- В поле характеристики $p \geq 1$ справедливо

$$(x+y)^p = x^p + y^p$$

$$(x+y)^p = \sum_{i=0}^{p} C_p^i x^i y^{p-i} \quad C_p^i = \frac{p!}{i!(p-i)!} \equiv 0 \mod p, 0 < i < p$$

9.1 Минимальные многочлены

Определение. Минимальным многочленом элемента $\beta \in GF(p^m)$ над $GF(p)$ называется нормированный многочлен $M_\beta(x) \in GF(p)[x]$ наименьшей степени такой, что $M_\beta(\beta) = 0$

Теорема 18. $M_\beta(x)$ неприводим над $GF(p)$.

Доказательство. Пусть $M_\beta(x)$ приводим, т.е. $M_\beta(x) = M_1(x)M_2(x)$ и $\deg M_i < \deg M$, Не дописано □

Теорема 19. Не дописано

Теорема 20. Не дописано

Теорема 21. Если α — примитивный элемент $GF(p^m)$, то степень его минимального многочлена равна m.

Доказательство. Не дописано □

Теорема 22. Все конечные поля $GF(p^m)$ изоморфны.

Доказательство. Пусть F и G — поля, содержащие p^m элементов, пусть α — примитивный элемент F с минимальным многочленом $\pi(x)$.

$$\pi(x) \mid (x^{p^m} - x) \Rightarrow \exists \beta \in G : \pi(\beta) = 0$$

Тогда можно рассматривать F как множество многочленов от α степени не более $m-1$, а G — как множество многочленов от β степени не более $m-1$. Тогда соответствие $\alpha \leftrightarrow \beta$ задает изоморфизм полей F и G. □

Не дописано

9.2 Построение минимальных многочленов

Теорема 23.

$$\forall \beta \in GF(p^m) \;\; M_\beta(x) = M_{\beta^p}(x)$$

Доказательство.

$$0 = M_\beta(\beta) = \sum_{i=0}^{d} M_{\beta,i}\beta^i, M_{\beta,i} \in GF(p)$$

$$0 = (M_\beta(\beta))^p = \sum_{i=0}^{d} M_{\beta,i}^p \beta^{pi} = \sum_{i=0}^{d} M_{\beta,i}\beta^{pi} = M_\beta(\beta^p) \Rightarrow M_{\beta^p}(x) \mid M_\beta(x)$$

Т.к. минимальные многочлены неприводимы, из делимости следует $M_{\beta^p}(x) = M_\beta(x)$. □

Не дописано

Лекция 10

11 ноября

9.3 Идеалы

Определение. Подмножество I кольца R называется правым *(или левым)* **идеалом**, если:

1. $(I, \{+\})$ является подгруппой $(R, \{+\})$
2. $\forall r \in R, x \in I : xr \in I$ *(или $rx \in I$) для правого идеала*

Двусторонний идеал — левый и правый идеал.

В коммутативных кольцах все идеалы являются двусторонними.

Определение. Для $A \subset R$ **идеалом, порождаемым** A, называется наименьший идеал, содержащий A:

$$\langle A \rangle \coloneqq \sum_i r_i a_i, a_i \in A, r_i \in R$$

Определение. Идеал I называется **конечно порожденным**, если существует конечное множество A такое, что $I = \langle A \rangle$.

Определение. Идеал называется **главным**, если он порождается единственным элементом.

Пример.

- Множество четных чисел — идеал в кольце $\mathbb{Z}$, порожденный 2.
- Множество многочленов с вещественными коэффициентами, делящихся на $x^2 + 1$ — идеал в кольце $\mathbb{R}[x]$.
- Множество квадратных матриц с нулевым последним столбцом — левый идеал в кольце квадратных матриц, но не правый идеал.

- Множество функций таких, что $f(1) = 0$ — идеал в кольце непрерывных функций $C(\mathbb{R})$.
- $\{0\}, R$ — тривиальные идеалы в любом кольце R.

Определение. Область целостности, в которой все идеалы являются главными, называется **кольцом главных идеалов**.

Определение. Идеал I кольца R называется **максимальным**, если $I \neq R$ и всякий прочий идеал J, содержащий I, является I или R.

Если $f_1 \dots f_n$ — система образующих максимального идеала I, то добавление в нее любого $f_0 \notin I$ приведет к $\langle f_0, f_1 \dots f_n \rangle = R$.

Пример. Идеал $\langle 7 \rangle \subset \mathbb{Z}$ является максимальным, так как числа, кратные 7, невозможно получить иначе как умножением 7 на целые числа.

Определение. Бинарное отношение $\sim$ на группе G называется отношением **конгруэнтности**, если оно является отношением эквивалентности и $\forall a, b, c \in G : (a \sim b) \Rightarrow a \cdot c \sim b \cdot c$.

Пример. $x \equiv y \mod n$ — отношение конгруэнтности.

Пример. Если I — идеал в кольце R, то бинарное отношение $\sim := \{(a, b) \in R^2 \mid b - a \in I\}$ — отношение конгруэнтности.

Различные обозначения для классов эквивалентности:

$$[a] = a + I = a \mod I = \{a + r \mid r \in I\}$$

Определение. Множество классов эквивалентности по отношению $\sim$ называется **факторкольцом** или **кольцом вычетов** R по модулю I и обозначается R/I.

Факторкольцо действительно является кольцом, если определить операции как:

1. $(a + I) + (b + I) = (a + b) + I$
2. $-(a + I) = (-a) + I$
3. $(a + I)(b + I) = ab + I$
4. Нулевой элемент $0 + I$
5. Единичный элемент $1 + I$

Если R — кольцо главных идеалов и $\langle a \rangle \in R$, то соответствующее факторкольцо обозначают R/aR

Теорема 24. Если $I \subset R$ является максимальным идеалом, то R/I является полем.

Доказательство. Необходимо показать, что для всякого ненулевого $a + I \in {}^{R}\!/_{I}$ существует обратный $b + I : (a + I)(b + I) = 1 + I$.

Если $a + I \neq 0$, то $a \notin I$.

Множество $J = \{ax + m \mid x \in R, m \in I\}$ является идеалом, т.к.:

1. $(ax_1 + m_1) \pm (ax_2 + m_2) = a(x_1 \pm x_2) + (m_1 \pm m_2)$ и $(x_1 \pm x_2) \in R, (m_1 \pm m_2) \in I$
2. $(ax + m)y = a(xy) + (my), xy \in R, my \in I$ при любом $y \in R$.

Если $x = 0$, то мы получим все элементы I, следовательно, $I \subset J$.

С другой стороны, т.к. $a \in J$ и $a \notin I$, а I — максимальный идеал, то $J = R$, а следовательно $1 \in J \Rightarrow \forall a \neq 0 \ \exists b, m : 1 = ab + m \Rightarrow ab - 1 \in I \Rightarrow (a + I)(b + I) = 1 + I$ □

Теорема 25. Пусть R — кольцо главных идеалов, p — его неприводимый элемент. Тогда факторкольцо ${}^{R}\!/_{pR}$ является полем.

Доказательство. Докажем, что $I = \langle p \rangle$ является максимальным. Предположим обратное, т.е. что существует идеал $J \neq R : I \subset J$.

Т.к. R — кольцо главных идеалов, то J является главным идеалом и $\exists q \in R : J = \langle q \rangle$.

Рассмотрим два случая:

1. q обратим. Тогда по предположению $J \neq R$, т.е. $qR \neq R$. Тогда умножение на q не инъективно, т.е. $\exists r_1 \neq r_2 : qr_1 = qr_2$. Но мы можем домножить слева на q^{-1} и получим что $r_1 = r_2$ — противоречие.
2. q не обратим. Тогда, т.к. $I \subset J$ и $J = \langle q \rangle$, $\exists s \in R : p = qs$. Но p неприводим по условию, следовательно s — обратимый элемент, а тогда $s^{-1}p = q \Rightarrow q \in I \Rightarrow J \subset I$.

Таким образом, $I \subset J$ и $J \subset I$, т.е. $I = J$, т.е. I — максимальный идеал. □

В дальнейшем такие поля будут обозначаться просто ${}^{R}\!/_{p}$

Пример.

- $GF(2) = {}^{\mathbb{Z}}\!/_{2}$
- Пусть p — простое число. $\mathbb{Z}$ — кольцо главных идеалов, следовательно, ${}^{\mathbb{Z}}\!/_{p}$ является полем и кроме того, это $GF(p)$.
- Многочлен x^2+1 неприводим над $\mathbb{R}$, $\mathbb{R}[x]$ — кольцо главных идеалов, следовательно, ${}^{\mathbb{R}[x]}\!/_{\langle x^2 + 1 \rangle}$ Не дописано

Не дописано

Если мы хотим построить циклический код длины n, то нам нужен $g(x) : g(x) \mid (x^n - 1) \Rightarrow \prod_{i \in J} f_i(x), J \subset \{0 \dots l-1\}$. Если все $f_i(x)$ различны, то есть $2^l - 2$ нетривиальных циклических кода.

Циклические коды над $GF(q)$ длины $n = q^m - 1$ называются примитивными.

Теорема 26. Пусть $\beta_1 \dots \beta_r \in GF(q^m)$ — корни порождающего многочлена $g(x)$ примитивного циклического кода $\mathcal{C}$ длины n над полем $GF(q)$.

Многочлен $c(x) \in GF(q)[x]$ является кодовым тогда и только тогда, когда $c(\beta_1) = \dots = c(\beta_r) = 0$.

Доказательство. Не дописано □

Не дописано

Пример. Рассмотрим поле $GF(2^3)$. В нем есть элементы $0, 1, \alpha$ и следующие элементы:

$$\begin{aligned} 1 &\quad \alpha^0 \\ \alpha &\quad \alpha^1 \\ \alpha^2 &\quad \alpha^2 \\ \alpha + 1 &\quad \alpha^3 \\ \alpha^2 + \alpha &\quad \alpha^4 \\ \alpha^2 + \alpha + 1 &\quad \alpha^5 \\ \alpha^2 + 1 &\quad \alpha^6 \\ 1 &\quad \alpha^7 \end{aligned}$$

Рассмотрим многочлен $g(x) = x^3 + x + 1$ и его корни: $0, \alpha^2, \alpha^4$.

Тогда проверочная матрица над $GF(2^3)$ это:

$$H = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha+1 & \alpha^2+\alpha & \alpha^2+\alpha+1 & \alpha^2+1 \\ 1 & \alpha^2 & \alpha^2+\alpha & \alpha^2+1 & \alpha & \alpha+1 & \alpha^2+\alpha+1 \\ 1 & \alpha^2+\alpha & \alpha & \alpha^2+\alpha+1 & \alpha^2+\alpha^2+1 & \alpha^2+1 & \alpha+1 \end{pmatrix}$$

Не дописано

10 Коды Боуза-Чоудхури-Хоквингема

Определение. Кодом **БЧХ** над $GF(q)$ длины n с конструктивным расстоянием δ называется циклический код наибольшей возможной размерности, порождающий многочлен которого имеет корни $a^b \dots a^{b+\delta-2}$, где $\alpha \in GF(q^m)$ — примитивный корень степени n из 1.

Примечание.

- По теореме Лагранжа $n \mid (q^m - 1)$. Если невозможно подобрать такое m, то кода БЧХ не существует.
- Если $n = q^m - 1$, то код БЧХ называется примитивным.
- Если $b = 1$, то это код БЧХ в узком смысле.
- Если $m = 1$, то это код Рида-Соломона.

Определение. Если порождающий многочлен циклического кода длины n над $GF(q)$ имеет корни $\alpha^b \dots \alpha^{b+\delta-2}$, где $\alpha \in GF(q^m)$ — примитивный корень степени n из 1, то минимальное расстояние этого кода $d \geq \delta$.

Доказательство. Не дописано □

Не дописано

Как правило, используют коды БЧХ в узком смысле, но иногда коды БЧХ в широком смысле позволяют выиграть в размерности.

До недавнего времени использовались в основном коды БЧХ, только недавно его стал вытеснять код LDPC, но внезапно оказалось, что на достаточно высоких скоростях его не успевают декодировать.

10.1 Коды Рида-Соломона

Определение. Код Рида-Соломона — код БЧХ длины $q - 1$ над $GF(q)$.

Минимальный многочлен $\beta \in GF(q)$ над $GF(q)$ это $M_\beta(x) = x - \beta$.

Порождающий многочлен кода Рида-Соломона имеет вид $g(x) = \prod_{i=0}^{\delta-2}(x - \alpha^{b+i})$.

Размерность: $k = n - \delta + 1$

Минимальное расстояние по теореме $d \geq \delta$. С другой стороны, граница Синглтона: $d \leq n-k+1 = \delta \Rightarrow d = n-k+1$. Таким образом, код Рида-Соломона имеет максимальное достижимое расстояние.

10.2 Декодирование кодов БЧХ

Рассмотрим исправление ошибок в векторе $y = c + e$, $y(x) = a(x)g(x) + e(x)$.

Синдром: $S_i = y(\alpha^{b+i}) = a(\alpha^{b+i})g(\alpha^{b+i}) + e(\alpha^{b+i}) = e(\alpha^{b+i}), 0 \leq i < \delta - 1$.

Пусть ошибки произошли в неизвестных нам позициях $j_1 \dots j_t, t \leq \lfloor(\delta - 1)/2\rfloor$

Не дописано

Лекция 11

18 ноября

Не дописано

Теорема 27.

$$E_i = \frac{X_i^{-b}\Gamma(X_i^{-1})}{\prod_{j\neq i}(1 - X_jX_i^{-1})}, \quad 0 \leq i < t$$

Доказательство.

$$\begin{aligned}\frac{X_i^{-b}\Gamma(X_i^{-1})}{\prod_{j\neq i}(1 - X_jX_i^{-1})} &= \frac{X_i^{-b}\sum_{l=1}^{t} E_lX_l^b\prod_{j\neq l}(1 - X_jX_i^{-1})}{\prod_{j\neq i}(1 - X_jX_i^{-1})} \\ &= \frac{E_i\prod_{j\neq l}(1 - X_jX_i^{-1})}{\prod_{j\neq i}(1 - X_jX_i^{-1})} \\ &= E_i\end{aligned}$$

□

Не дописано

10.3 Расширенный алгоритм Евклида

Не дописано

Пусть

$$U_j(x) = \prod_{i=0}^{j}\begin{pmatrix}-q_i(x), 1\\ 1, 0\end{pmatrix} = U_{j-1}\underbrace{\begin{pmatrix}-q_j(x), 1\\ 1, 0\end{pmatrix}}_{Q_j(x)} = \begin{pmatrix}u_{j,0}(x) & u_{j-1,0}(x)\\ u_{j,1}(x) & u_{j-1,1}(x)\end{pmatrix}, \quad U_{-1}(x) = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$$

$$\begin{pmatrix} r_0(x), r_{-1}(x) \end{pmatrix} \begin{pmatrix} u_{j,0}(x) & u_{j-1,0}(x) \\ u_{j,1}(x) & u_{j-1,1}(x) \end{pmatrix} = \begin{pmatrix} r_{j+1}(x), r_j(x) \end{pmatrix}$$

Свойства алгоритма: Не дописано

10.4 Алгоритм Сугиямы

Пусть $\delta = 2\tau + 1$.

1. Пусть $r_{-1}(x) = x^{2\tau}, r_0(x) = S(x)$
2. Выполнять $r_{i-1}(x) = q_i(x)r_i(x)+r_{i+1}(x)$, пока не получится $\deg r_i(x) \geq \tau, \deg r_{i+1}(x) < \tau$.
3. $\Gamma(x) = r_{i+1}(x), \Lambda(x) = u_{i,0}(x)$

Корректность алгоритма:

- Алгоритм конечен, т.к. степени $r_i(x)$ монотонно убывают с каждым шагом алгоритма. Т.к. каждый шаг понижает степень хотя бы на 1, то алгоритм имеет сложность $\mathcal{O}(n)$.
- $\Gamma(x) = r_{i+1}(x) = r_0(x)u_{i,0}(x) + r_{-1}u_{i,1}(x) = S(x)\Lambda(x) + x^{2\tau}u_{i,1}(x) \equiv S(x)\Lambda(x) \mod x^{2\tau}$.
- $\deg u_{i,0}(x) = \deg r_{-1}(x) - \deg r_i(x) \leq 2\tau - \tau = \tau$.
- Пусть у нас есть некоторое альтернативное решение Не дописано

10.5 Сложность декодирования кодов БЧХ и Рида-Соломона

1. Вычисление синдрома:
   - Схема Горнера: $S_i = y(\alpha^{b+i}) = y_0 + \alpha^{b+i}(y_1 + \alpha^{b+i}(y_2 + \dots))$, сложность – $\mathcal{O}(n\delta)$ операций.
   - Метод Герцеля: $r_i(x) \equiv y(x) \mod M_{\alpha^i}(x), \quad S_i = r_i(\alpha^i), \alpha \in GF(p^m)$, где $M_{\alpha^i}(x) \in GF(p)[x]$ – минимальный многочлен α^i. Деление на него использует только сложения, при этом минимальные многочлены многих α^i совпадают.

   Примечание. Это не единственное упрощение, но у нас нет времени на остальные.
2. Решение ключевого уравнения $\Gamma(x) \equiv S(x)\Lambda(x) \mod x^{\delta-1}$. Расширенный алгоритм Евклида это делает за $\mathcal{O}(\delta^2)$ операций.
3. Поиск корней X_i^{-1} многочлена локаторов ошибок $\Lambda(x)$ выполняется процедурой Ченя, т.е. перебором $\alpha^i, 0 \leq i < n$ и проверкой $\Lambda(\alpha^i) = 0$, за $\mathcal{O}(n\delta/2)$. Это можно делать асимптотически быстрее как факторизацию многочлена, но так не делают

из-за большой константы. В реальности используют быстрое преобразование Фурье, т.к. Не дописано.

4. Поиск значений ошибок выполняется методом Форни со сложностью $\mathcal{O}(\delta^2)$.

10.6 Синтез регистров сдвига с линейной обратной связью

Определение. РСЛОС — регистр, в котором новое значение порождается по формуле

$$-S_{j+t} = \Lambda_1 S_{j+t-1} + \cdots + \Lambda_t S_j$$

, Λ_i — константы, значение с наименьшим индексом выбрасывается.

Не дописано

Лемма 3. Пусть фильтры $(L_{n-1}, \Lambda^{(n-1)}(x))$ и $(L_n, \Lambda^{(n)}(x))$ порождают последовательности S_0^{n-2} и S_0^{n-1} соответственно, причем $(L_{n-1}, \Lambda^{(n-1)}(x))$ не порождает S_0^{n-1}, и величины L_{n-1} и L_n являются наименьшими возможными.

Тогда $L_n \geq \max(L_{n-1}, n - L_{n-1})$.

Доказательство. Не дописано □

Не дописано

Лекция 12

25 ноября

Не дописано

10.7 QR–коды

Определение. Число y называется **квадратичным вычетом** ***(quadratic residue)*** по модулю n, если существует число x такое, что $y = x^2 \mod n$.

Свойства.

- $(n-x)^2 \equiv x^2 \mod n \Rightarrow 1^2, 2^2 \dots ((n-1)/2)^2$ являются Не дописано
- Не дописано

Это не те QR–коды, про которые вы подумали, черно-белые квадраты это Quick Response code, где используется код Рида-Соломона.

В Blu-Ray дисках данные кодируются так называемым **пикетным** кодом. Не дописано. Оптические диски имеют свойство повреждаться, при этом повреждения состоят из большого числа последовательных ошибок. Пикетный код состоит из двух кодов — один кодирует данные, другой служебные данные. Не дописано

CRC *(Cyclic Redundancy Check)* — циклический код. Контрольная сумма, т.е. многочлен проверочных символов $r(x)$ для многочлена данных $a(x)$ вычисляется с помощью формулы систематического кодирования $r(x) \equiv x^{N-K} a(x) \mod g(x), \deg a(x) \leq K-1$. Не дописано.

Выводы:

- Циклические коды допускают более компактное задание по сравнению с линейными блоковыми кодами.
- Коды БЧХ можно строить с заданным минимальным расстоянием.

- Коды Рида-Соломона — коды БЧХ на границе Синглтона.
- Существуют алгоритмы декодирования кодов БЧХ с исправлением $\lfloor(\delta-1)/2\rfloor$ ошибок со сложностью $\mathcal{O}(n\delta+\delta^2)$.
- Не дописано

11 Альтернантные коды и криптосистема Мак-Элиса

Теорема 28. Если $c=(c_0\dots c_{n-1})$ — кодовое слово кода Рида-Соломона над $GF(q)$ в узком смысле тогда и только тогда, когда $c_i=f(\alpha^i), 0\le i<n$, т.е. $c=\mathrm{ev}^{[1]}(f)$, где $\deg f(x)<k, f(x)\in GF(q)[x]$.

Доказательство.

$$\alpha^n \overset{\text{def}}{=} 1 \Rightarrow 0=\alpha^{in}-1=(\alpha^i-1)\sum_{j=0}^{n-1}\alpha^{ij},\quad 0<i<n$$

Т.к. по определению БЧХ $\alpha^i\neq 1$, то $\sum_{j=0}^{n-1}\alpha^{ij}=0$.

c — кодовое слово тогда и только тогда, когда $cH^{\mathsf{T}}=0$, т.е. $\sum_{i=0}^{n-1}c_i\alpha^{ij}=0, 1\le j\le n-k$.

$$\begin{aligned}\sum_{i=0}^{n-1}f(\alpha^i)\alpha^{ij}&=\sum_{i=0}^{n-1}\left(\sum_{t=0}^{k-1}f_t\alpha^{it}\right)\alpha^{ij}\\&=\sum_{t=0}^{k-1}f_t\underbrace{\sum_{i=0}^{n-1}\alpha^{i(j+t)}}_{0}\\&=0\end{aligned}$$

Не дописано □

Из этого результата можно получить альтернативное определение кода Рида-Соломона:

Определение. $(n,k,n-k+1)$ кодом **Рида-Соломона** называется множество векторов $c=(c_0\dots c_{n-1})$, где $c_i=f(\alpha^i), \deg f(x)<k, f(x)\in GF(q)[x], \alpha^i\in GF(q)$ — различные значения, называемые **локаторами**.

Не дописано

Определение (обобщенные коды Рида-Соломона)**.** Не дописано

[1] Evaluate.

Определение. Альтернантным кодом длины n над полем $GF(q)$ называется код $\mathcal{A}(n, r, a, u)$ с проверочной матрицей

$$H = \begin{pmatrix} \alpha_0^0 & \alpha_1^0 & \dots & \alpha_{n-1}^0 \\ \alpha_0^1 & \alpha_1^1 & \dots & \alpha_{n-1}^1 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_0^{n-1} & \alpha_1^{n-1} & \dots & \alpha_{n-1}^{n-1} \end{pmatrix} \operatorname{diag}(u_0 \dots u_{n-1})$$

, где $\alpha_i \in GF(q^m)$ — различные элементы и $u_i \in GF(q^m) \setminus \{0\}$.

- Не дописано

Теорема 29. Пусть $m \mid (n-h)$. Существует альтернантный $(n, k \geq h, d \geq \delta)$ код над $GF(q)$ такой, что

$$\sum_{i=1}^{\delta-1} (q-1)^i \binom{i}{n} < (q^m - 1)^{(n-h)/m}$$

Доказательство. Пусть вектор локаторов фиксирован.

Рассмотрим вектор без элементов–нулей $y \in GF(q)^n \setminus \{0\}$. Оценим число обобщенных кодов Рида–Соломона, содержащих этот вектор, т.е. $|\{GRS(n, k', a, v) \mid y \in GRS(n, k', a, v)\}|$.

Не дописано □

Не дописано

11.1 Криптосистема Мак–Элиса

Пусть дана порождающая матрица G кода, для которого известен эффективный алгоритм исправления t ошибок.

Будем использовать как секретный ключ Не дописано, а как открытый ключ пару $\Gamma = QGP, t$. Сообщение x шифруется как $y = x\Gamma + e$, где e — случайный вектор веса не более t.

Дешифрование y происходит следующим образом: $y' = yP^{-1} = xQG + eP^{-1} = xQG + e^t$. Вектор e' можно найти из $y' - e' = (xQ \mid xQA)$.

Поиск секретного ключа по открытому — задача эквивалентности кодов, которая просто решается для кодов Рида-Соломона, но не для кодов Гоппы.

Восстановление сообщения по криптограмме — задача исправления декодирования линейного кода.

Недостаток: большой размер открытого ключа: ≈ 512 Кбит.

11.2 Криптосистема Нидеррайтера

Не дописано