15

Лекция 15–16 декабря

Коды, исправляющие ошибки

Константин Чухарев

Дискретная математика — Осень 2026

Коды, исправляющие ошибки

> *Цель вычислений — проникновение в суть, а не числа.*
>
> ***— Ричард Хэмминг***

Канал шумит

Канал передачи данных искажает биты.

Провод, радиоволна, дорожка диска — всё это каналы, и все они ошибаются.

Запись на диск — тоже передача, только из прошлого в будущее.

- В памяти с коррекцией бит может перевернуться от космической частицы.
- Царапина на диске портит дорожку не по одному биту, а целым участком.

Защита и сжатие

- **Защита**: добавить избыточность, чтобы исправить ошибки.
- **Сжатие**: убрать избыточность, чтобы уменьшить размер.

Коды, исправляющие ошибки, работают в QR-кодах, на дисках и в спутниковой связи.

Сжатие без потерь живёт в zip и PNG.

Плата за защиту — полезная скорость: чем больше избыточности, тем меньше данных проходит через канал за единицу времени.

Чётность: первая защита

Пример

К двум битам данных добавим третий — их XOR:

$$00 \mapsto 000, \quad 01 \mapsto 011, \quad 10 \mapsto 101, \quad 11 \mapsto 110.$$

Одна ошибка ломает чётность — её видно.

Но какой бит испорчен, не узнать: от принятого слова три кодовых слова отличаются ровно одним битом.

Код чётности имеет кодовое расстояние $d = 2$.

Он обнаруживает одну ошибку и не исправляет ни одной.

Повторение: дорого

Пример

Код-повторение передаёт каждый бит трижды: слова 000 и 111.

Принятый бит решается голосованием по большинству.

Одна ошибка исправляется.

Но два из трёх битов канала потрачены на защиту, а не на данные.

Защита оплачивается скоростью.

Код-повторение сжигает две трети канала, чтобы исправить один бит.

Расстояние Хэмминга

Определение 1 (Расстояние Хэмминга)

Число позиций, в которых строки различаются:

$$d(x, y) = |\{i \mid x_i \neq y_i\}|.$$

Пример

$d(10110, 10001) = 3$: различаются позиции 3, 4 и 5.

$d(0000, 1111) = 4$, $d(x, x) = 0$.

Метрика

Расстояние Хэмминга — метрика на строках.

- Оно неотрицательно и равно нулю только для совпадающих строк.
- Оно симметрично: $d(x, y) = d(y, x)$.
- Выполняется неравенство треугольника: $d(x, z) \leq d(x, y) + d(y, z)$.

Чтобы превратить x в y, нужно перевернуть ровно $d(x, y)$ битов.

Неравенство треугольника гарантирует: ошибки канала не сблизят два кодовых слова.

Вес слова

Определение 2 (Вес слова)

Весом слова x называется число его единичных позиций, обозначается $\mathrm{wt}(x)$.

Пример

$\mathrm{wt}(10110) = 3$, $\mathrm{wt}(111) = 3$, $\mathrm{wt}(000) = 0$.

Расстояние и вес связаны: $d(x, y) = \mathrm{wt}(x + y)$, где $+$ — побитовый XOR.

Разошедшиеся позиции — это ровно единицы суммы $x + y$.

Кодовое расстояние

Определение 3 (Кодовое расстояние)

Минимум расстояний между разными кодовыми словами:

$$d = \min(\{d(x, y) \mid x, y \in C, x \neq y\}).$$

Пример

У кода-повторения $\{000, 111\}$: $d = 3$.

У кода чётности: $d = 2$.

Параметры кода

Определение 4 (Параметры кода)

Код характеризуется тройкой $(n, |C|, d)$: длина, число слов, кодовое расстояние.

Для линейных кодов пишут (n, k, d), где k — число информационных битов, и тогда $|C| = 2^k$.

Пример

Код Хэмминга: $(7, 4, 3)$ — четыре бита данных, $2^4 = 16$ слов, расстояние 3.

Код-повторение: $(3, 1, 3)$.

Скорость и избыточность

Определение 5 (Скорость кода)

Скорость $R = \frac{k}{n}$ — доля информационных битов.

Избыточность $1 - R$ — доля служебных.

Код	Скорость	Исправляет
Повторение $(3, 1)$	$1/3 \approx 0.33$	1 бит
Хэмминг $(7, 4)$	$4/7 \approx 0.57$	1 бит

При одинаковой надёжности код Хэмминга передаёт примерно в $\frac{12}{7} \approx 1.7$ раза больше данных, чем повторение.

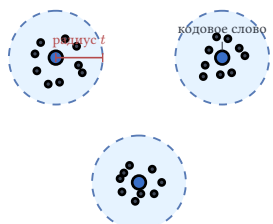
Повторение — простая и дорогая форма избыточности.

Шары вокруг слов

Кодовые слова — точки в пространстве всех двоичных строк длины n.

Декодер возвращает принятую строку к ближайшему кодовому слову — как текстовый редактор исправляет опечатку к ближайшему правильному слову.

Каждое слово окружено шаром радиуса t: все строки внутри шара декодируются в его центр.



Способность кода

Теорема 1 (Способность кода)

Если $d \geq s + 1$, код обнаруживает до s ошибок.

Если $d \geq 2t + 1$, код исправляет до t ошибок, то есть

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor.$$

Пример

Код с $d = 3$ исправляет одну ошибку.

Код с $d = 4$ исправляет по-прежнему одну, зато отличает двойную ошибку от одиночной.

Декодирование по ближайшему

Определение 6 (Декодирование по ближайшему)

Принятое слово r декодируется в кодовое слово c, для которого $d(r, c)$ минимально.

Пока ошибок не больше t, ближайшее слово — единственное и правильное.

При большем числе ошибок декодер всё равно вернёт ближайшее, но это уже совпадение, а не гарантия.

Доказательство

Доказательство

Докажем разбором двух утверждений.

Пусть $d \geq s + 1$, и принятое слово отличается от переданного не более чем в s позициях. Тогда оно отстоит от переданного слова меньше чем на d, поэтому не может совпасть с другим кодовым словом, — ошибка обнаружена.

Пусть теперь $d \geq 2t + 1$. Шары радиуса t вокруг кодовых слов не пересекаются: если бы точка z лежала в двух шарах с центрами x и y, получилось бы $d(x, y) \leq d(x, z) + d(z, y) \leq 2t$ — противоречие.

Принятое слово с не более чем t ошибками попадает ровно в один шар и декодируется в его центр.

Код Хэмминга (7, 4)

Определение 7 (Код Хэмминга)

Параметры: $n = 7$, $k = 4$ бита данных, $d = 3$.

Позиции $1, 2, 4$ — степени двойки — занимают проверочные биты, позиции $3, 5, 6, 7$ — данные:

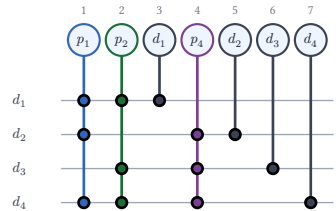
$$p_1, p_2, d_1, p_4, d_2, d_3, d_4.$$

Хэмминг придумал код от раздражения: машина с перфокартами останавливалась на каждой ошибке чтения, и выходные проходили впустую.

Его статья 1950 года дала первый явный код, исправляющий одну ошибку.

Контрольные группы

Проверочный бит — XOR данных своей группы.



- p_1 контролирует d_1, d_2, d_4.
- p_2 контролирует d_1, d_3, d_4.
- p_4 контролирует d_2, d_3, d_4.

Почему такие группы?

Бит p_i контролирует позиции, в двоичной записи которых i-й бит равен 1.

Позиция $3 = 011$ входит в первые две группы, позиция $7 = 111$ — во все три.

Три проверки работают как вопросы с ответом «да-нет»: каждая делит позиции пополам и отдаёт один бит номера позиции.

Кодирование

Пример

Данные $d_1 = 1, d_2 = 0, d_3 = 1, d_4 = 1$.

$$p_1 = 1 \oplus 0 \oplus 1 = 0, \quad p_2 = 1 \oplus 1 \oplus 1 = 1, \quad p_4 = 0 \oplus 1 \oplus 1 = 0.$$

Кодовое слово: $(0, 1, 1, 0, 0, 1, 1)$.

Задача «двадцать вопросов» решена: три XOR при кодировании собирают три бита защиты.

Синдром

Определение 8 (Синдром)

При приёме каждый бит синдрома — XOR своей контрольной группы:

$$s_1 = p_1 \oplus d_1 \oplus d_2 \oplus d_4, \quad s_2 = p_2 \oplus d_1 \oplus d_3 \oplus d_4, \quad s_4 = p_4 \oplus d_2 \oplus d_3 \oplus d_4.$$

Нулевой синдром означает, что ошибок нет.

Ненулевой синдром читается как двоичное число $(s_4 s_2 s_1)_2$.

Оно равно номеру искажённой позиции — надо лишь перевернуть этот бит.

Таблица синдромов

s_1	s_2	s_4	$(s_4 s_2 s_1)_2$	Позиция
1	0	0	1	1
0	1	0	2	2
1	1	0	3	3
0	0	1	4	4
1	0	1	5	5
0	1	1	6	6
1	1	1	7	7

Совпадение синдрома с номером позиции — следствие выбора контрольных групп.

Столбцы проверочной матрицы — двоичные записи чисел от 1 до 7, поэтому синдром складывает столбцы искажённых позиций.

Принять и исправить

Пример

Передано $(0,1,1,0,0,1,1)$, ошибка в позиции 5: принято $(0,1,1,0,1,1,1)$.

$$s_1 = 0 \oplus 1 \oplus 1 \oplus 1 = 1, \quad s_2 = 1 \oplus 1 \oplus 1 \oplus 1 = 0, \quad s_4 = 0 \oplus 1 \oplus 1 \oplus 1 = 1.$$

Синдром $(1,0,1) = 5$: переворачиваем пятый бит — слово восстановлено.

Кодирование и декодирование зеркальны: те же три группы, тот же XOR.

Проверочная матрица

Три контрольные группы собираются в одну матрицу:

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Слово кодовое, если $Hx^T = 0$ над $\mathrm{GF}(2)$.

Пример

Для слова $(0,1,1,0,0,1,1)$ проверка даёт $Hx^T = (0,0,0)^T$.

Для принятого слова синдром — это Hr: он называет позицию ошибки.

Две ошибки

Пример

Передано $(0, 1, 1, 0, 0, 1, 1)$, ошибки в позициях 4 и 6: принято $(0, 1, 1, 1, 0, 0, 1)$.

Синдром $(0, 1, 0)$ имеет значение 2.

Декодер переворачивает бит 2 — здоровую позицию.

Две ошибки маскируются под одну.

Код с $d = 3$ их замечает, но исправить не может.

Три ошибки

Пример

Ошибки в позициях 1, 2 и 3: вектор ошибок $(1, 1, 1, 0, 0, 0, 0)$ сам является кодовым словом.

Принятое слово кодовое, синдром нулевой — декодер сообщает «ошибок нет», хотя данные испорчены.

За пределами $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ исправлений код за результат не отвечает.

Расширенный код (8, 4, 4)

Один общий бит чётности всего слова увеличивает расстояние до 4.

Теорема 2 (Расширенный код)

Код $(8, 4, 4)$ по-прежнему исправляет одну ошибку.

Зато он отличает двойную ошибку от одиночной.

Синдром	Чётность	Что делать
0	сошлась	ошибок нет
0	не сошлась	испорчен бит чётности — исправляем
не 0	не сошлась	одна ошибка — исправляем
не 0	сошлась	две ошибки — не трогаем

Линейные коды

Определение 9 (Линейный код)

Множество $C \subseteq \mathrm{GF}(2)^n$ называется **линейным кодом**, если оно содержит нулевое слово и замкнуто относительно покоординатного XOR.

Сумма двух кодовых слов — снова кодовое слово.

Все двоичные строки длины n — векторное пространство $\mathrm{GF}(2)^n$ над полем из двух элементов: XOR — сложение, AND — умножение.

Линейный код — подпространство в $\mathrm{GF}(2)^n$.

Порождающая матрица

Определение 10 (Порождающая матрица)

Матрица G размера $k \times n$, строки которой образуют базис кода.

Кодирование сообщения $m \in \mathrm{GF}(2)^k$ — умножение $c = mG$.

Пример

Для кода Хэмминга:

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

Сообщение $(1, 0, 1, 1)$: складываем строки 1, 3 и 4 — получаем $(0, 1, 1, 0, 0, 1, 1)$.

Расстояние линейного кода

Теорема 3 (Кодовое расстояние линейного кода)

$$d = \min(\{\mathrm{wt}(c) \mid c \in C, c \neq 0\}).$$

Доказательство

Докажем двумя неравенствами.

Пусть c — ненулевое слово наименьшего веса. Нулевое слово принадлежит коду, и пара c и 0 даёт $d \leq \mathrm{wt}(c)$.

Обратно, для любых разных $x, y \in C$ сумма $x + y$ — ненулевое кодовое слово, и $\mathrm{wt}(x + y) = d(x, y)$.

Перебор сокращается с $\binom{16}{2} = 120$ пар до 15 ненулевых слов.

Лекция 15: что мы поняли?

- Расстояние Хэмминга — метрика, а кодовое расстояние d обнаруживает до $d-1$ и исправляет до $\left\lfloor \frac{d-1}{2} \right\rfloor$ ошибок.
- Код Хэмминга $(7,4)$: проверочные биты — XOR контрольных групп, синдром называет позицию ошибки.
- Проверочная матрица собирает группы: кодовые слова — решения $Hx^T = 0$.
- За пределами гарантий код за результат не отвечает: две ошибки маскируются под одну, три могут пройти незаметно.
- Линейный код — подпространство в $\mathrm{GF}(2)^n$: порождающая матрица кодирует, расстояние равно минимальному весу.

Вопросы для самопроверки

1. Вычислите $d(110010, 010101)$.
2. Код чётности обнаруживает одну ошибку. Почему он не исправляет ни одной?
3. Закодируйте данные $(0, 1, 1, 0)$ кодом Хэмминга $(7, 4)$.
4. Принято $(0, 1, 1, 0, 1, 1, 1)$. Вычислите синдром и исправьте ошибку.
5. Почему для линейного кода достаточно найти минимальный вес ненулевого слова?
6. Дан код $\{000, 111\}$. Найдите его параметры $(n, |C|, d)$ и скорость.

Чтение к следующей паре

- m13 — «Расстояние Хэмминга и геометрия кода», «Код Хэмминга», «Линейные коды и $\mathrm{GF}(2)$».
- Флеш-опросник — в начале лекции 16.
- Теормин 2 (кодирование и булева алгебра, устно) — на следующей неделе, на отдельной паре.