Обработка и интерпретация сигналов
Конспект лекций

Александр Клыков

Лекция 1

Содержание

1 Упрощённая модель цифровой системы связи

Рассмотрим упрощённую модель цифровой системы связи. В системе есть:

- **источник данных** (например, флешка, компакт-диск, речь в микрофон и т. п.);
- **кодер источника** (ставит в соответствие информационным символам кодовые символы);
- **канал связи** (в канале действует шум, из-за чего возможны ошибки);
- **декодер источника** (по принятому сигналу восстанавливает переданные данные);
- **получатель**.

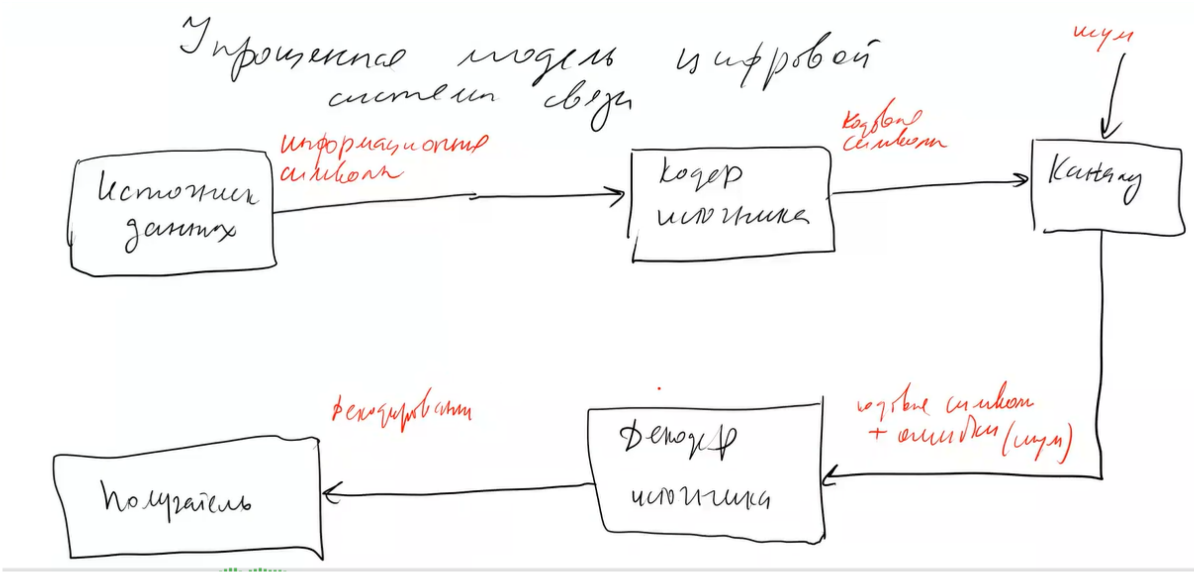

Рис. 1: Упрощённая модель цифровой системы связи.

В дальнейшем будем работать с **дискретными последовательностями**. Будем считать, что информационная последовательность состоит из элементов поля

$$\mathrm{GF}(2) = \{0, 1\}.$$

То есть источник генерирует последовательность нулей и единиц.

2 Двоичный симметричный канал

Рассмотрим **двоичный симметричный канал** (ДСК) с переходной вероятностью p. На вход канала последовательно подаются 0 и 1, а на выходе возможны ошибки:

$$\begin{array}{lll} 0 \to 0 & \text{с вероятностью} & 1-p \\ 0 \to 1 & \text{с вероятностью} & p \\ 1 \to 0 & \text{с вероятностью} & p \\ 1 \to 1 & \text{с вероятностью} & 1-p \end{array}$$

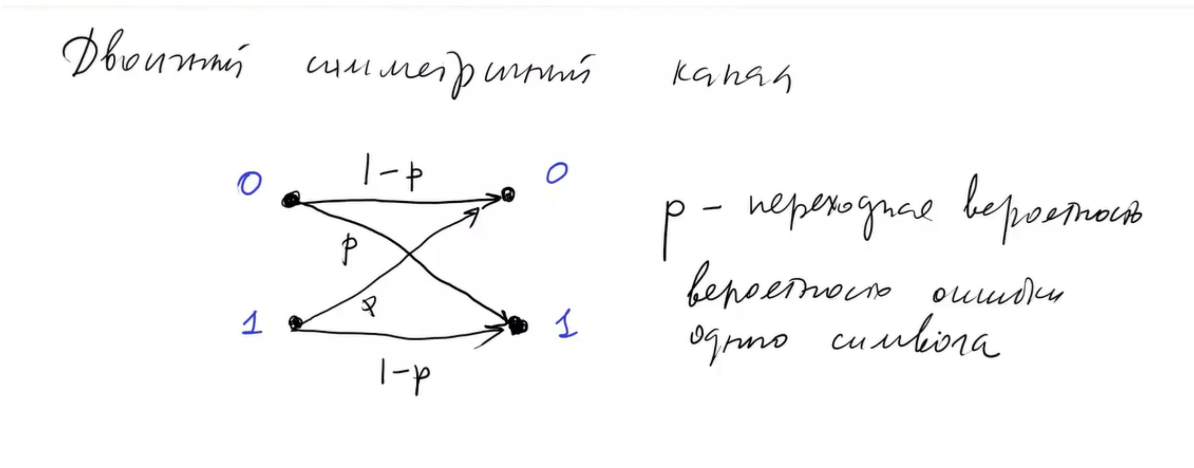

Рис. 2: Двоичный симметричный канал .

Пусть $p = 10^{-3}$. Возникает вопрос: **как уменьшить вероятность ошибки**? Один из простейших способов — добавить избыточность, например, **дублировать данные**.

3 Повторный код длины 3

Рассмотрим кодирование повторением по 3 раза:

$$0 \mapsto 000, \qquad 1 \mapsto 111.$$

Здесь $0, 1$ — **информационные символы**, а $000, 111$ — **кодовые слова** (кодовые символы). Процедура перехода от информационного слова к кодовому называется **кодированием**.

3.1 Декодирование и вероятность ошибки

Для принятого трёхбитового слова естественно использовать правило большинства:

- декодируем как 0, если принято 000, 001, 010, 100;
- декодируем как 1 в остальных случаях.

Если было передано 000, то ошибка декодирования произойдёт, когда в канале исказятся **как минимум два бита**, то есть события:

$$\text{ровно 2 ошибки} \quad \text{или} \quad \text{ровно 3 ошибки}.$$

Для ДСК это даёт

$$P_{\text{ош}} = \binom{3}{2} p^2(1-p) + p^3 = 3p^2(1-p) + p^3.$$

4 Пример кода $(n, k) = (5, 2)$

Рассмотрим другой пример: «склеим» биты парами, то есть $k = 2$, и будем кодировать в слова длины $n = 5$:

$$\begin{aligned}
00 &\mapsto 00000,\\
01 &\mapsto 10110,\\
10 &\mapsto 01011,\\
11 &\mapsto 11101.
\end{aligned}$$

4.1 Принцип декодирования по минимальному расстоянию

Если принято слово r, то выбираем то кодовое слово c, для которого расстояние Хемминга $d(r, c)$ минимально. Например:

- если получено 10000, то наиболее вероятно было передано 00000;
- если передавали 01011, а получено 01001, то естественно предположить, что передавали 01011 (искажён один бит).

Можно проверкой убедиться, что данный код исправляет **одну** ошибку.

4.2 Скорость кода

Пусть k — число информационных бит (символов) на слово, n — длина кодового слова. Тогда **скорость кода**:

$$R = \frac{k}{n}.$$

Для повторного кода $0 \mapsto 000$, $1 \mapsto 111$ имеем $k = 1$, $n = 3$, значит $R = \frac{1}{3}$.

Для кода длины 5 с информационными парами $k = 2$, $n = 5$, поэтому $R = \frac{2}{5}$.

$$\frac{2}{5} = 0.4 > \frac{1}{3} \approx 0.333.$$

То есть второй код передаёт информацию **быстрее** (при меньшей избыточности), чем повторение по 3 раза.

5 Историческая справка и теорема Шеннона

Клод Шеннон (1948) заложил основы теории информации. В **кодировании источника** обычно стремятся убирать избыточность, а в **канальном кодировании** — наоборот, добавляют избыточность, чтобы исправлять ошибки, возникающие из-за шума в канале.

5.1 Пропускная способность ДСК

Для двоичного симметричного канала с переходной вероятностью p вводится **пропускная способность**:

$$C = 1 - h(p),$$

где $h(p)$ — двоичная энтропия:

$$h(p) = -p \log_2 p - (1 - p) \log_2(1 - p).$$

Утверждение 1. *Если скорость передачи $R < C$, то можно обеспечить сколь угодно малую вероятность ошибки декодирования за счёт увеличения длины используемых кодов (что повышает сложность кодирования и декодирования). Если же $R > C$, то надёжная передача становится невозможной.*

Для $p = 10^{-3}$:

$$C = 1 - h(10^{-3}) \approx 0.988592.$$

То есть, чтобы добиться сколь угодно высокой надёжности, в принципе достаточно добавить порядка нескольких процентов избыточности (при достаточно больших длинах кодов).

6 Вес и расстояние Хемминга

Определение 1 (Вес Хемминга)**.** *Пусть x — кодовое слово.* ***Вес Хемминга*** $\omega(x)$ *— это число ненулевых элементов в x. В двоичном случае это просто число единиц.*

Определение 2 (Расстояние Хемминга)**.** ***Расстояние Хемминга*** $d(x, y)$ *между словами x и y — это количество позиций, в которых они различаются.*

Пример:

$$x = 001101, \quad \omega(x) = 3; \qquad y = 101001, \quad \omega(y) = 3; \qquad d(x, y) = 2.$$

Утверждение 2. *В двоичном случае справедливо:*

$$d(x, y) = \omega(x + y),$$

где сложение $x + y$ выполняется побитово по модулю 2.

7 Минимальное расстояние кода и исправление ошибок

Для кода C (множества кодовых слов) определим **минимальное расстояние**:

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

Для кода

$$\{00000,\ 10110,\ 01011,\ 11101\}$$

можно выписать матрицу расстояний (нумеруем слова как c_1, c_2, c_3, c_4 в порядке сверху):

	c_1	c_2	c_3	c_4
c_1	0	3	3	4
c_2	3	0	4	3
c_3	3	4	0	3
c_4	4	3	3	0

Отсюда $d_{\min} = 3$.

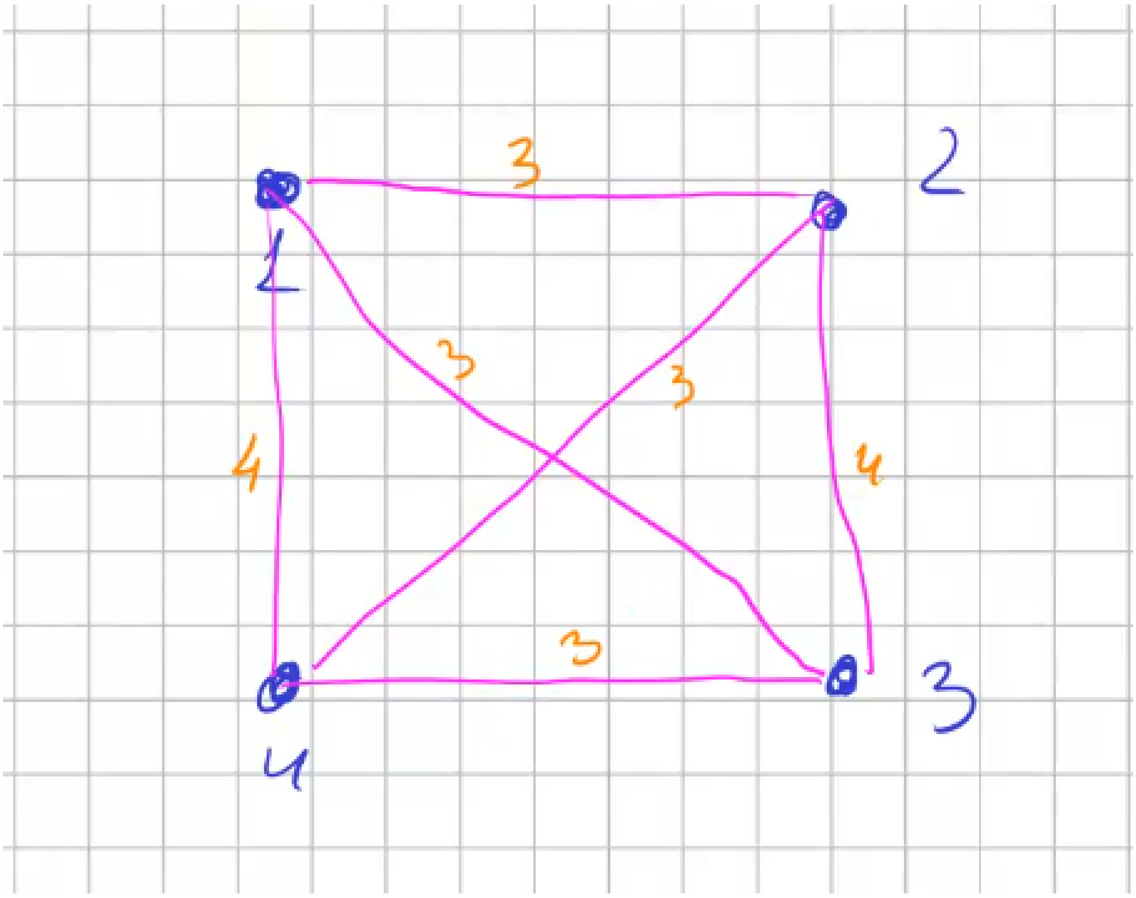

Рис. 3: Схема декодирования по минимальному расстоянию.

Утверждение 3. *Если код исправляет ошибки кратности t, то выполняется оценка:*

$$t \le \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor .$$

8 Линейные коды

Определение 3. *Двоичный код называется **линейным**, если сумма двух любых кодовых слов (по модулю 2) снова является кодовым словом.*

Обычно множество кодовых слов обозначают C (не путать с пропускной способностью канала, которая тоже часто обозначается C). Для линейного кода:

$$\forall x, y \in C : \quad (x + y) \in C.$$

Замечание 1. *В двоичном случае*

$$d(x, y) = \omega(x + y).$$

Если код линейный, то можно переписать минимальное расстояние как

$$d_{\min} = \min_{\substack{z \in C \\ z \neq 0}} \omega(z),$$

то есть минимальное расстояние равно минимальному весу среди всех ненулевых кодовых слов.

8.1 Общее определение q-ичного линейного кода

Определение 4. *Линейный q-ичный (n,k)-код — это любое k-мерное подпространство пространства $\mathbb{F}_q^n$ всех векторов длины n над полем $\mathbb{F}_q$. Здесь n — длина кодового слова, k — длина информационного слова.*

Пример для $q=3$, $k=2$, $n=5$: информационных слов $q^k = 3^2 = 9$ (элементы из $\mathbb{F}_3 = \{0,1,2\}$), а всего слов длины 5 — $q^n = 3^5$. Линейный код — это выбор подпространства размерности $k=2$, то есть множества из $q^k = 9$ кодовых слов, замкнутого относительно сложения.

9 Порождающая матрица

Так как код является линейным подпространством, у него существует базис. Для двоичного линейного (n,k)-кода удобно собрать базисные векторы в строки матрицы.

Определение 5. ***Порождающей матрицей*** *(generator matrix) (n,k)-кода называется матрица G размера $k \times n$, строки которой образуют базис кода. Любое кодовое слово является линейной комбинацией строк G.*

Пусть $m = (m_1, \ldots, m_k)$ — информационное слово, а $c = (c_1, \ldots, c_n)$ — кодовое слово. Тогда

$$c = mG.$$

Подставляя разные m, получаем все возможные кодовые слова.

9.1 Пример базиса для кода $(5,2)$

Для кода

$$\begin{aligned} 00 &\mapsto 00000, \\ 01 &\mapsto 10110, \\ 10 &\mapsto 01011, \\ 11 &\mapsto 11101, \end{aligned}$$

можно взять базисные векторы, например:

$$e_1 = 10110, \qquad e_2 = 01011.$$

Тогда:

$$\begin{aligned} 0 \cdot e_1 + 0 \cdot e_2 &= 00000, \\ 1 \cdot e_1 + 0 \cdot e_2 &= 10110, \\ 0 \cdot e_1 + 1 \cdot e_2 &= 01011, \\ 1 \cdot e_1 + 1 \cdot e_2 &= 11101. \end{aligned}$$

Соответственно можно записать порождающую матрицу:

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

10 Проверочная матрица

Иногда удобно задавать код не порождающей матрицей, а системой проверок.

Пусть существует вектор $h = (h_1, \dots, h_n)$ такой, что для любого кодового слова $c = (c_1, \dots, c_n)$ выполняется

$$(c, h) = c_1 h_1 + \cdots + c_n h_n = 0 \quad \text{(все операции в GF(2))}.$$

Тогда говорят, что h ортогонален коду и задаёт **проверку**.

Если собрать $n - k$ независимых проверок в строки матрицы H размера $(n - k) \times n$, то получим **проверочную матрицу** (parity-check matrix), для которой выполняется:

$$GH^T = 0, \qquad cH^T = 0 \ \text{ для любого кодового слова } c.$$