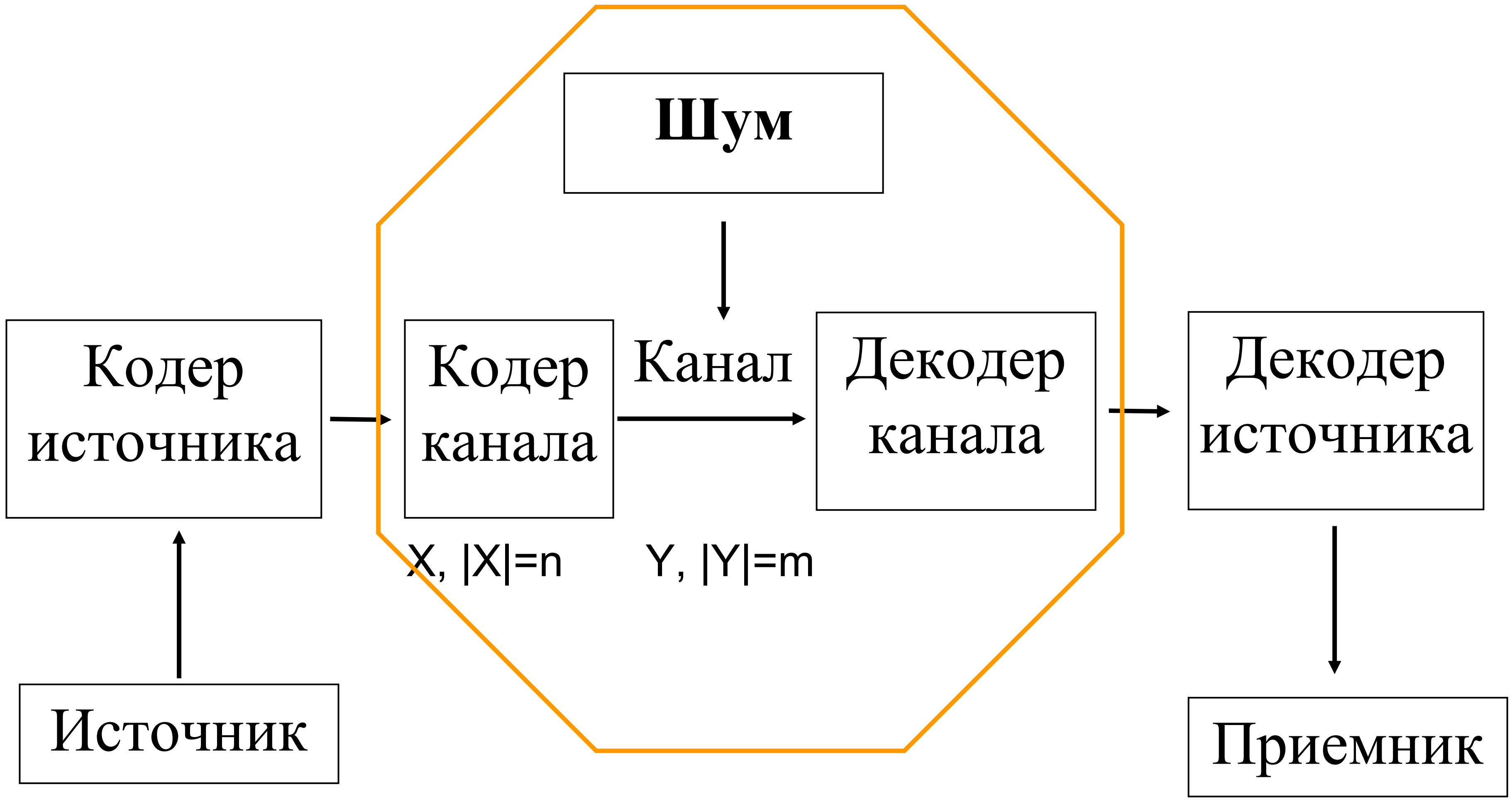
Помехоустойчивое кодирование

Основные понятия и оценки

Модель системы передачи сигналов



Теорема Шеннона для дискретного канала с шумом

Пусть дискретный канал обладает пропускной способностью C, дискретный источник – энтропией H(A) в единицу времени.

- **Если H(A)<C, то существует такая система кодирования, при котором сообщения источника могут быть переданы по каналу с произвольно малой ненадежностью.**
- **Если H(A)>C, то не существует способа кодирования, обеспечивающего ненадежность, меньшую чем H(A)-C**

- *Помехоустойчивое кодирование* вносит в данные специальным образом организованную избыточность, что

  в дальнейшем позволит обнаружить или исправить ошибки, внесенные в данные каналом связи.

блочные помехоустойчивые коды

- Входная последовательность двоичных символов делится на блоки одинаковой длины n (количество информационных сиволов)
- Каждому блоку сопоставляется двоичное кодовое слово длины m (длина кода)
- Очевидно, что n<m (m-n число проверочных символов)

Пример

Код трехкратного повторения ставит в соответствие любому биту входных данных этот же бит, но повторенный трижды.

n=1	код m=3	искажения
1	111	011,101,110
0	000	100, 010,001

Множества не пересекаются

Если в кодовом слове произойдет две ошибки, то код не сможет их исправить: «1» кодируем в «111», вносим две ошибки – «100», декодируем – «0».

Очевидно, что для случая, когда необходимо исправлять большее число ошибок, можно увеличить число повторения бита в кодовом слове, так, код пятикратного повторения будет гарантированно исправлять все одиночные и двойные ошибки.

Однако это сильно снижает скорость передачи информации

Пример

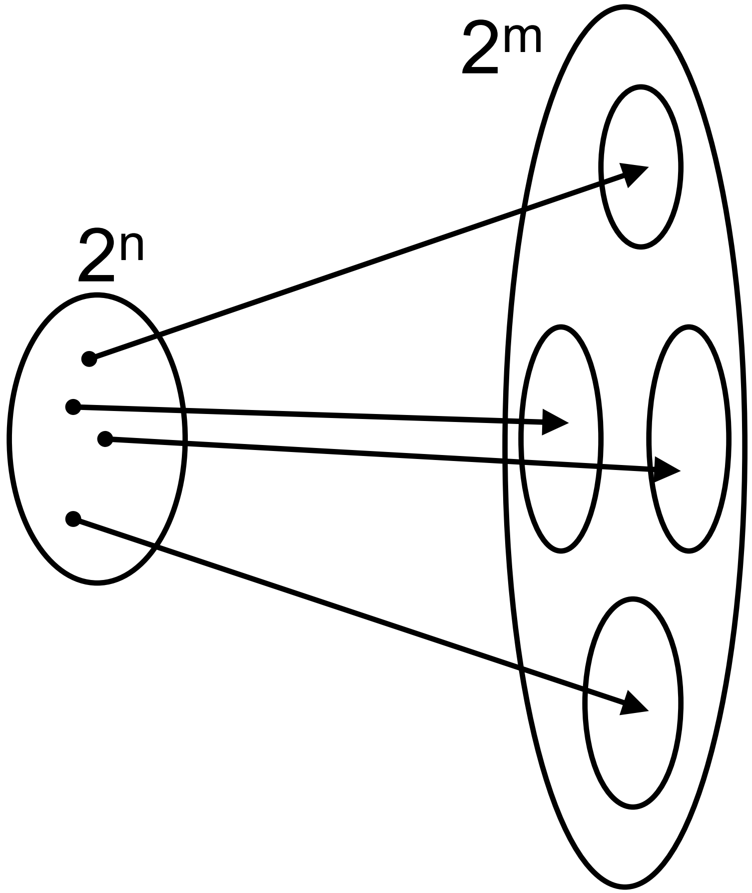
Код проверки на четность добавляет к строке входных данных фиксированного размера *k* один (*k*+1)й бит как сумму предыдущих *k* бит. Таким образом, число единиц в кодовом слове всегда четное

Этот код не исправляет ошибки, а умеет только **обнаруживать** одиночные ошибки.

Если декодер получил из канала связи с нечетном числом единичных бит, то обнаруживает, что слово искажено ошибкой,

Однако указать ее местоположение декодер не может.

- Кодирование – это отображение двоичных наборов
- Искаженные наборы должны группироваться около кодовых и не пересекаться



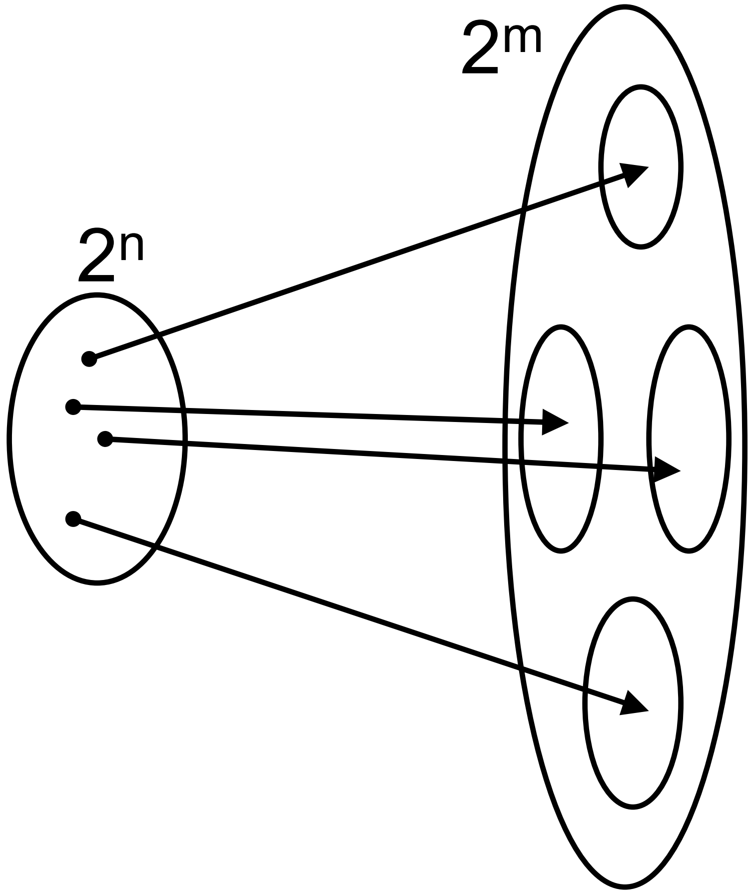
- **Вес набора x w(x) – количество единиц в x**
- ***Расстоянием* (метрикой) *Хемминга* $d(x,y)$ между двоичными наборами *x*,*y* называется число несовпадающих в *x* и *y* позиций.**
- **Очевидно, что $d(x,y)= w(x+y)$**
- ***Пример* *Р*асстояние Хемминга между 1011,1111 $d(1011,1111)=w(1011+1111)=w(0100)=1$;**

- **Минимальное значение $d(c_i, c_j)$ между всеми парами кодовых слов $c_i \neq c_j$ *минимальным расстоянием кода*, далее будем обозначать его $d_{\min}$.**
- **Если канал связи вносит не более t ошибок в кодовое слово, то для однозначного восстановления необходимо и достаточно, чтобы для расстояния между кодовыми словами $c_i \neq c_j$ выполнялось неравенство:**

$$d(c_i, c_j) \geq 2t+1.$$

- *Код* называют *совершенным*, если для него сферы некоторого одинакового радиуса вокруг кодовых слов, не пересекаясь, покрывают все пространство.

- *Квазисовершенный код* – это код, у которого сферы радиусом *t* вокруг каждого кодового слова не пересекаются и все слова, не лежащие внутри сфер, лежат на расстоянии *t*+1 хотя бы от одного кодового слова.



Основные параметры блочного кода

- *размерность кода* n (т.е. длина информационных слов),
- *длина кода* m (длина кодовых слов)
- Минимальное кодовое расстояние d_{min}.
- Код с указанными параметрами кратко называют: (m,n)-код или (m,n,d_{min})-код.
- Отношение m/n называют *избыточностью блочного кода*, а отношение n/m – *скоростью* кода.

Граница Хэмминга

- Для (m,n)-кода с минимальным кодовым расстоянием 2t+1 верно соотношение

$$2^m \geq 2^n (C_n^t + C_n^{t-1} + \ldots + C_n^1 + 1)$$

- Граница Хэмминга – необходимое условие для помехозащитного кода с заданными характеристиками
- Для совершенных кодов выполняется равенство

Оценка Варшамова-Гилберта

- (m,n)-код с минимальным кодовым расстоянием d существует, если

$$2^{m-n} - 1 > (C_{n-1}^{d-2} + C_{n-1}^{d-3} + \ldots + C_{n-1}^{1} + 1)$$

Линейные коды

- Если любая линейная комбинация кодовых слов также является кодовым словом, то такой код называется линейным
- Если рассматривать линейный код длины m как множество векторов, то линейный код образует линейное подпространство в E^m

- Линейное подпространство имеет набор базисных векторов, и любой вектор подпространства может быть представлен как линейная комбинация базисных векторов.
- Таким образом, линейный (m,n)-код C может быть задан порождающей матрицей G размерности n x m
- Строками порождающей матрицы являются базисные вектора

- Другой способ задания линейного кода – проверочная матрица H размерности (m-n) x m, составленная из базисных векторов ортогонального дополнения
- Тогда любое кодовое слово с удовлетворяет соотношению $\vec{c}H^T = 0$

- Если для линейного (m,n)-кода С заданы порождающая матрица G и проверочная матрица H, то кодовое слово для блока v будет определяться как $\vec{v}G$
- При декодировании проверка на ошибки происходит при помощи проверочной матрицы.

- Если порождающая матрица n x m содержит единичную подматрицу в первых n столбцах, то ее называют порождающей матрицей в систематическом виде.

$$G = \left(E_n \middle| P \right)$$

- Проверочную матрицу можно привести к систематическому виду равносильными преобразованиями. Линейный код будет эквивалентным

- Проверочная матрица (m-n) x m будет иметь вид

$$H = \left(-P^T \middle| E_{m-n} \right)$$

Пример

- Зададим линейный (5,3)-код
- 3-мерное подпространство в E^5 можно задать базисом из 3 линейно независимых векторов

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

Блок v n=3	Код $\vec{v}G$ m=5
000	00000
001	00111
010	01001
011	01110
100	10010
101	10101
110	11011
111	11100

- Проверочная матрица (5-3)x5

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

Транспонируем и дополняем единичной подматрицей

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

- Если сообщение было передано верно, то при умножение на проверочную матрицу даст нулевой вектор.

Теорема

- Минимальное расстояние линейного кода равно минимальному из весов ненулевых кодовых слов
- Если любые $r \le d-1$ столбцов проверочной матрицы H линейного (m,n)-кода линейно независимы, то минимальное расстояние кода равно d
- Если минимальное расстояние линейного (m,n)-кода равно d, то любые $r \le d-1$ столбцов проверочной матрицы H линейно независимы и найдутся d линейно зависимых столбцов.

Оценка Синглтона

- Для линейного (m,n, d)-кода

$$d \le m - n + 1$$

Декодирование и исправление ошибок

- Сообщение c $\Rightarrow$ v=c+e, где e – вектор ошибки
- Тогда $s = vH^T = (c+e)H^T = eH^T$
- Вектор s называется синдромом, который позволяет найти ошибку.
- Кодовые слова, искаженные одинаковым вектором ошибок, образуют смежный класс и имеют одинаковый синдром
- Вектор наименьшего веса в смежном классе называют лидером класса

Пример

- e=(10000)

Код	Искаженный код	синдром
00000	10000	10 лидер
00111	10111	10
01001	11001	10
01110	11110	10
10010	00010	10
10101	00101	10
11011	01011	10
11100	01100	10

Декодирование и исправление ошибок слова v

1. Вычислить синдром $s = vH^T$

Если s=0, ошибок нет c=v

Если s≠0, то определяется подходящий смежный класс и лидер этого класса e

c=v+e

2. Исправленное кодовое слово декодируем по таблице кодовых слов

- Линейный код задан порождающей матрицей

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

- Определить основные параметры эквивалентного систематического кода и декодировать слово 0101

- Размеры матрицы 2x4, поэтому $n=2, m=4$.
- Скорость кода $n/m=0.5$

- Преобразуем матрицу G в систематический вид, переставив столбцы

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

- Тогда проверочная матрица имеет вид

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

Для построения декодера составим таблицы кодовых слов и синдромов

Блоки n=2	Коды сист.
00	0000
01	0110
10	1011
11	1101

синдромы	лидеры
00	0000
01	0001
10	0100
11	1000

- Получено слово v=(0101)
- Вычислим синдром s=(11)
- Ему соответствует вектор ошибок e=(1000)
- Исправление v+e=1101
- По таблице кодов определяем исходное сообщение 11

Код Хэмминга

Построить блочный код с простым декодером таким, чтобы значение синдрома было равно номеру позиции, в которой произошла ошибка.

коды Хемминга (1950 г.)

- блочные линейные коды с минимальным расстоянием $d=3$, т. е. можно исправить одну или обнаруживать две ошибки
- Длина кода $m=2^r-1$, число информационных символов $n=2^r-1-r$, число проверочных $m-n=r$, $r=2,3,\ldots$
- Коды Хемминга являются совершенными

- Проверочная матрица H(r,m) представляет собой матрицу, столбцами которой являются все ненулевые двоичные векторы длины n, записанные в естественном лексикографическом порядке

$$H(2,3) = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$$

$$H(3,7) = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

- Кодовое слово состоит из информационных и проверочных бит
- Проверочными являются биты с номерами-степенями двойки, т.е. 2^0, 2^1, 2^2, 2^3 ,… 2^r
- Остальные биты соответствуют порядку символов с исходном сообщении
- Проверочные биты определяются из системы линейных уравнений.

Пример

- r=3, m=7, n=4

$$H(3,7)=\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

- Закодировать слово a=(1010)

$$a = (1010)$$

$$b = (b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ b_7)$$

- Соотношение $\vec{b}H^T(3,7)=0$

дает систему уравнений для проверочных символов

$$\begin{cases} b_4+b_5+b_6+b_7=0 \\ b_2+b_3+b_6+b_7=0 \\ b_1+b_3+b_5+b_7=0 \end{cases} \qquad \begin{cases} b_4=0+1+0=1 \\ b_2=1+1+0=0 \\ b_1=1+0+0=1 \end{cases}$$

- Код для 1010 будет 1011010

- Декодировать слово c=(1010010)
- Вычисляем синдром $\vec{c}H^T(3,7) = e$

$$(1010010)\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} = (100)$$

Ошибка произошла в 4 разряде

Исправленное сообщение (**1011010**)

Исходное сообщение 1010