Санкт-Петербургский Политехнический Университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Телекоммуникационные технологии

Отчет по лабораторной работе №7
Помехоустойчивое кодирование

Работу выполнил:
Ерниязов Т.Е.
Группа: 33501/3
Преподаватель:
Богач Н.В.

Санкт-Петербург
2017

Содержание

1. Цель и задачи

1.1. Цель работы

Изучение методов помехоустойчивого кодирования и сравнения их свойств.

1.2. Постановка задачи

Провести кодирование/декодирование сигнала, полученного с помощью функции randerr кодом Хэмминга 2-мя способами: с помощью встроенных функций encode/decode, а также через создание проверочной и генераторной матриц и вычисление синдрома. Оценить корректирующую способность кода.

Выполнить кодирование/декодирование циклическим кодом, кодом БЧХ, кодом Рида-Соломона. Оценить корректирующую способность кода.

2. Теоретическая информация

2.1. Кодирование

Физическое кодирование — линейное преобразование двоичных данных, осуществляемое для их передачи по физическому каналу. Физическое кодирование может менять форму, ширину полосы частот и гармонический состав сигнала в целях осуществления синхронизации приёмника и передатчика, устранения постоянной составляющей или уменьшения аппаратных затрат передачи сигнала.

Обнаружение ошибок в технике связи — действие, направленное на контроль целостности данных при записи/воспроизведении информации или при её передаче по линиям связи. Исправление ошибок (коррекция ошибок) — процедура восстановления информации после чтения её из устройства хранения или канала связи.

Для обнаружения ошибок используют коды обнаружения ошибок, для исправления — корректирующие коды (коды, исправляющие ошибки, коды с коррекцией ошибок, помехоустойчивые коды).

Основой для обнаружения и исправления ошибок такими кодами является увеличение кодового расстояния между кодовыми комбинациями (кодовые комбинации различаются в двух и более разрядах).

Кодовое расстояние (Хемминга) - это минимальное число элементов, в которых одна кодовая комбинация отличается от другой (по всем парам кодовых слов). В корректирующих кодах комбинации построены по определенному правилу, например, содержат четное число единиц. Построение помехоустойчивого кода с недоиспользованием части кодовых комбинаций, приводящей к так называемой "избыточности". Избыточность означает, что из исходных символов можно построить больше комбинаций, чем предусмотрено при их приеме. Таким образом, уменьшение числа используемых комбинаций приводит к повышению помехоустойчивости кода, которая связана с увеличением кодового расстояния.

2.2. Типы помехоустойчивого кодирования

2.2.1. Кодирование Хэмминга

Коды Хемминга представляют собой линейные блочные коды, что означает, что операция кодирования может быть описана с помощью порождающей матрицы. Они позволяет исправлять одиночную ошибку (ошибка в одном бите) и находить двойную. Код Хэмминга

(7,4) дает 7 бит вывода для каждых 4 бит ввода. Кодовые слова получаются как линейная комбинация строк G, где все операции вычисляются по модулю 2 в каждом векторном элементе. То есть, код является пространством строк для G. Для Вектор сообщений m = [m l, m2, m3, m4] кодовое слово является

$$code = msg * G \tag{1}$$

При декодировании используется проверочная матрица H. Каждому (n, k) линейному блочному коду соответствует связанная с ним матрица H, называемая матрицей проверки на четность. Первый этап декодирования - это вычисление синдрома S.

$$S = code * H^T \tag{2}$$

Синдром позволяет определить в какой позиции произошла ошибка.

Коды Хэмминга являются самокорректирующимися кодами, то есть кодами, позволяющими автоматически обнаруживать ошибки при передаче данных и исправлять их.

Для построения самокорректирующегося кода, рассчитанного на исправление одиночных ошибок, одного контрольного разряда недостаточно. Как видно из дальнейшего, количество контрольных разрядов k должно быть выбрано так, чтобы удовлетворялось неравенство

$$2^k \geq k + m + 1 \tag{3}$$

или

$$k \geq \log_2(k + m + 1) \tag{4}$$

где m — количество основных двоичных разрядов кодового слова.

Построение кодов Хэмминга основано на принципе проверки на четность числа единичных символов: к последовательности добавляется такой элемент, чтобы число единичных символов в получившейся последовательности было четным.

$$r_1 = i_1 \oplus i_2 \oplus ... \oplus i_k \tag{5}$$

$$S = i_1 \oplus i_2 \oplus ... \oplus i_n \oplus r_1 \tag{6}$$

Тогда если $S = 0$ - ошибки нет, иначе есть однократная ошибка.

Такой код называется $(k + 1, k)$. Первое число — количество элементов последовательности, второе — количество информационных символов.

Получение кодового слова выглядит следующим образом:

$$(i_1\; i_2\; i_3\; i_4) \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} = (i_1\; i_2\; i_3\; i_4\; r_1\; r_2\; r_3) \tag{7}$$

Получение синдрома выглядит следующим образом:

$$(i_1\; i_2\; i_3\; i_4\; r_1\; r_2\; r_3) \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} S_1 & S_2 & S_3 \end{pmatrix} \tag{8}$$

2.2.2. Циклические коды

Циклический код — линейный код, обладающий свойством цикличности, то есть каждая циклическая перестановка кодового слова также является кодовым словом. Используется для преобразования информации для защиты её от ошибок.

2.2.3. Коды БЧХ

Коды Боуза — Чоудхури — Хоквингема (БЧХ-коды) — в теории кодирования это широкий класс циклических кодов, применяемых для защиты информации от ошибок. Отличается возможностью построения кода с заранее определёнными корректирующими свойствами, а именно, минимальным кодовым расстоянием. Частным случаем БЧХ-кодов является код Рида — Соломона.

2.2.4. Коды Рида-Соломона

Коды Рида—Соломона (англ. Reed–Solomon codes) — недвоичные циклические коды, позволяющие исправлять ошибки в блоках данных. Элементами кодового вектора являются не биты, а группы битов (блоки). Код Рида—Соломона является частным случаем БЧХ-кода.

3. Ход работы

Листинг 1: Код MATLAB

```
function Code()
close all
format loose
clc
out = randerr(1,4);
disp('Hamming')
fprintf('Исходное сообщение');
disp(out);
code = encode (out, 7, 4, 'hamming/binary');
fprintf('Кодированное сообщение\n');
disp(code);
dcode = decode (code, 7, 4, 'hamming/binary');
fprintf('Декодированное сообщение\n');
disp(dcode)

msg = out;
[h, gen] = hammgen(3);
% gen = gen2par(h);
code = rem(msg * gen, 2);
code(3) = 0;
trt = syndtable(h);
syndrome = rem(code * h', 2);

% error location:
err = bi2de(fliplr(syndrome));
err_loc = trt(err + 1, :);

% corrected code
ccode = rem(err_loc + code, 2);

dcode = ccode(7 - 4 + 1:7);
```

```


fprintf('Порождающая матрица');
disp(gen)
fprintf('Проверочная матрица\n');
disp(h)

fprintf('Исходное сообщение\n');
disp(msg)
fprintf('Кодированное сообщение\n');
disp(code)
fprintf('Синдром\n');
disp(syndrome)
fprintf('Скорректированное сообщение\n');
disp(ccode)
fprintf('Декодированное сообщение\n');
disp(dcode)

disp('Cyclic')
out = randerr(1,4);
fprintf('Исходное сообщение\n');
disp(out);
code = encode (out, 7, 4, 'cyclic/binary');
fprintf('Кодированное сообщение\n');
disp (code);
dcode = decode (code, 7, 4, 'cyclic/binary');
fprintf('Декодированное сообщение\n');
disp(dcode)


m = 4;
n = 2^m-1;
k = 5;
nwords = 5;
msg = randi([0 1],nwords,k);
code = gf(msg);
[poly,t] = bchgenpoly(n,k);
enc = bchenc(code,n,k);
noisycode = enc + randerr(nwords,n,1:t);
dcode = bchdec(noisycode,n,k);

fprintf('Исходное сообщение\n');
disp(msg)
fprintf('Полином\n');
disp(poly.x)
fprintf('Кодированное сообщение\n');
disp(enc.x)
fprintf('Декодированное сообщение\n');
disp(dcode.x)
disp('Rid - Salomon')

m = 3;
n = 2^m - 1;
k = 3;

msg = gf([2 7 3; 4 0 6; 5 1 1],m);
code = rsenc(msg,n,k);

errors = gf([2 0 0 0 0 0 0; 3 4 0 0 0 0 0; 5 6 7 0 0 0 0],m);
noisycode = code + errors;
```

```

[dcode, cnumerr] = rsdec(noisycode,n,k);

fprintf('Исходное сообщение\n');
disp(msg.x)
fprintf('Кодированное сообщение\n');
disp(code.x)
fprintf('Декодированное сообщение\n');
disp(dcode.x)
fprintf('Число исправленных ошибок\n');
disp(cnumerr)
end
```

3.1. Коды Хэмминга

Результат кодирования и декодирования сигнала кодом Хэмминга (7, 4).

Декодированное сообщение совпадает с исходным.

```
Исходное сообщение
     1     1     0     0

Кодированное сообщение
     1     0     1     1     1     0     0

Декодированное сообщение
     1     1     0     0
```

Допустим ошибку в кодированном сообщении и проведем самокоррекцию. Сначала построим проверочную и порождающую матрицу.

```
Порождающая матрица
     1     1     0     1     0     0     0
     0     1     1     0     1     0     0
     1     1     1     0     0     1     0
     1     0     1     0     0     0     1

Проверочная матрица
     1     0     0     1     0     1     1
     0     1     0     1     1     1     0
     0     0     1     0     1     1     1
```

Результаты кодирования, определения синдрома и декодирования приведены в тексте

```
Исходное сообщение
     0     1     0     0

Кодированное сообщение
     0     1     0     0     1     0     0

Синдром
     0     0     1

Скорректированное сообщение
```

```
    0     1     1     0     1     0     1

Декодированное сообщение
    0     1     0     0
```

Ошибка была найдена и скорректирована, исходное сообщение совпадает с декодированным.

3.2. Циклические коды

Результат кодирования циклическим кодом

```
Исходное сообщение
    1     1     0     0

Кодированное сообщение
    0     1     0     1     1     0     0

Декодированное сообщение
    1     1     0     0
```

При кодировании сообщений с кодовым расстоянием 1, получаются закодированные сообщения с кодовым расстоянием 3.

3.3. Коды БЧХ

Результат кодирования кодом БЧХ. При кодировании сообщений с кодовым расстоянием 1, получаются закодированные сообщения с кодовым расстоянием 3 или 4.

```
Исходное сообщение
    0     1     1     1     0
    0     0     1     1     1
    1     1     1     0     1
    0     0     0     0     1
    1     0     0     1     0


Полином
  1   0   1   0   0   1   1   0   1   1   1




Кодированное сообщение
  0   1   1   1   0   0   0   0   1   0   1   0   0   1   1
  0   0   1   1   1   1   0   1   0   1   1   0   0   1   0
  1   1   1   0   1   0   1   1   0   0   1   0   0   0   1
  0   0   0   0   1   0   1   0   0   1   1   0   1   1   1
  1   0   0   1   0   0   0   1   1   1   1   0   1   0   1
```

```
Декодированное сообщение
   0   1   1   1   0
   0   0   1   1   1
   1   1   1   0   1
   0   0   0   0   1
   1   0   0   1   0
```

3.4. Коды Рида-Соломона

Код Рида-Соломона позволяет кодировать недвоичные сообщения.

При кодировании сообщений с кодовым расстоянием 1, получаются закодированные сообщения с кодовым расстоянием 3 или 4.

```
Исходное сообщение
           2           7           3
           4           0           6
           5           1           1

Кодированное сообщение

  2   7   3   3   6   7   6
  4   0   6   4   2   2   0
  5   1   1   4   5   4   0


Декодированное сообщение
 2   7   3
 4   0   6
 0   7   6

Число исправленных ошибок
     1
     2
    -1
```

4. Выводы

В данной работе были рассмотрены помехозащищенные коды, позволяющие исправить 1 или более ошибок. На примере Кода Хэмминга была протестирована возможность исправления ошибки. Рассмотренные коды Хэмминга и Циклические позволяют устранить только одну ошибку, а рассмотренные коды БЧХ и Рида-Соломона позволяют устранить до двух ошибок.