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:
Шустенков О.А.
Группа: 33501/1
Преподаватель:
Богач Н.В.

Санкт-Петербург
2018

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 (такому как оптическое волокно или витая пара). 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.

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).

2.2. Типы помехоустойчивого кодирования

2.2.1. Кодирование Хэмминга

Коды Хемминга — простейшие линейные коды с минимальным расстоянием 3, то есть способные исправить одну ошибку. Код Хемминга может быть представлен в таком виде, что синдром

$$\vec{s} = \vec{r}H^T \tag{1}$$

Это принятый вектор, будет равен номеру позиции, в которой произошла ошибка. Это свойство позволяет сделать декодирование очень простым.

Коды Хэмминга являются самоконтролирующимися кодами, то есть кодами, позволяющими автоматически обнаруживать ошибки при передаче данных.

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

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

или

$$k \geq \log_2(k+m+1) \tag{3}$$

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.

$$r_1 = i_1 \oplus i_2 \oplus ... \oplus i_k \tag{4}$$

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

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

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

Для примера рассмотрим классический код Хемминга $(7, 4)$. Сгруппируем проверочные символы следующим образом:

$$r_1 = i_1 \oplus i_2 \oplus i_3 r_2 = i_2 \oplus i_3 \oplus i_4 r_3 = i_1 \oplus i_2 \oplus i_4 \tag{6}$$

Получение кодового слова выглядит следующим образом:

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

На вход декодера поступает кодовое слово $V = (i'_1, i'_2, i'_3, i'_4, r'_1, r'_2, r'_3)$ где штрихом помечены символы, которые могут исказиться в результате помехи. В декодере в режиме исправления ошибок строится последовательность синдромов:

$S_1 = r_1 \oplus i_1 \oplus i_2 \oplus i_3$

$S_2 = r_2 \oplus i_2 \oplus i_3 \oplus i_4$

$S_3 = r_3 \oplus i_1 \oplus i_2 \oplus i_4$

$S = (S_1, S_2, S_3)$ называется синдромом последовательности.

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 Боуза-Чоудхури-Хоквингема (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

3.1. Коды Хэмминга

Реализация коды Хэмминга с помощью MATLAB:

Листинг 1: Код в программе MatLab

```
out = randerr(1,4) + randerr(1,4);
disp(out);
code = encode (out, 7, 4, 'hamming/binary');
disp(code);
dcode = decode (code, 7, 4, 'hamming/binary');
if (dcode == out) disp('dcode=out -its Hamming');
end;
```

Представлены сообщение и код, полученный стандартной функцией encode с параметром 'hamming/binary' (использовался стандартный код (7,4)).

```
     0     1     1     0

     1     0     0     0     1     1     0

dcode=out -its Hamming
```

Рис. 3.1.1. Исходное сообщение и его код Хэмминга.

При кодировании сообщений с кодовым расстоянием, равным 1, получали, как пример, закодированные сообщения с кодовым расстоянием равным 3.

3.2. Циклические коды

Реализация циклического кода с помощью MATLAB:

Листинг 2: Код в программе MatLab

```
out = randerr(1,4) + randerr(1,4);
```

```
disp(out);
code = encode (out, 7, 4, 'cyclic/binary');
disp (code);
dcode = decode (code, 7, 4, 'cyclic/binary');
if (dcode == out) disp('dcode=out -its cyclic');
end;
```

Представлено сообщение, закодированное циклическим кодом, полученным стандартной функцией encode с параметром 'cyclic/binary' (использовался стандартный код (7,4)).

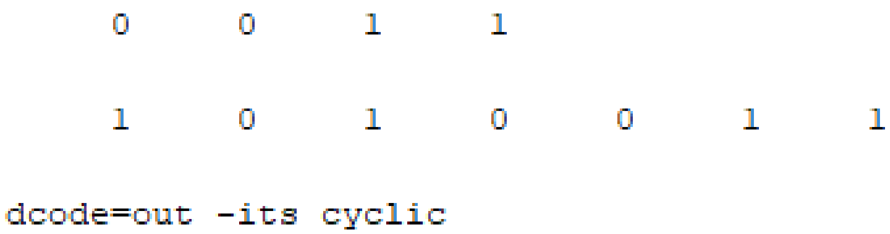

Рис. 3.2.1. Исходное сообщение и его циклический код.

При кодировании сообщений с кодовым расстоянием, равным 1, получали, как пример, закодированные сообщения с кодовым расстоянием равным 3.

3.3. Коды Боуза-Чоудхури-Хоквингема (БЧХ)

Реализация БЧХ с помощью MATLAB:

Листинг 3: Код в программе MatLab

```
m = 4;        % Number of bits per symbol
n = 2^m-1;    % Codeword length
k = 5;        % Message length
nwords = 10; % Number of words to encode

code = gf(randi([0 1],nwords,k));
[~,t] = bchgenpoly(n,k);
enc = bchenc(code,n,k);
noisycode = enc + randerr(nwords,n,1:t);
dcode = bchdec(noisycode,n,k);

code
dcode

if (code == dcode) disp ('BCH');
end;
```

Для кодирования/декодирования с помощью кодов БЧХ использовались, соответственно, функции bchenc/bchdec. При кодировании сообщений с кодовым расстоянием, равным 1, получали, как пример, закодированные сообщения с кодовым расстоянием равным 3, или 4.

Массивы после кодирования и декодирования представлены на рис. 3.3.1 и 3.3.2.

```
code = GF(2) array.

Array elements =

     0     0     0     0     0
     1     1     1     0     1
     1     0     1     1     0
     1     1     0     0     0
     0     1     1     0     1
     0     1     1     0     0
     0     1     1     1     1
     1     0     1     1     1
     0     1     1     1     1
     1     0     1     0     0
```

Рис. 3.3.1. Массив после кодирования

```
dcode = GF(2) array.

Array elements =

     0     0     0     0     0
     1     1     1     0     1
     1     0     1     1     0
     1     1     0     0     0
     0     1     1     0     1
     0     1     1     0     0
     0     1     1     1     1
     1     0     1     1     1
     0     1     1     1     1
     1     0     1     0     0
```

Рис. 3.3.2. Массив после декодирования

3.4. Коды Рида-Соломона

Реализация кода Ричи-Соломона с помощью MATLAB:

Листинг 4: Код в программе MatLab

```
m = 3;              % Number of bits per symbol
n = 2^m - 1;        % Codeword length
k = 3;              % Message length

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);

cnumerr
```

При использовании кодов Рида-Соломона в виде стандартной функции rsenc можно наблюдать вектор cnumerr, который содержит количества исправляемых ошибок.

```
cnumerr =

     1
     2
    -1
```

Рис. 3.4.1. Количество исправляемых ошибок cnumerr.

При кодировании сообщений с кодовым расстоянием, равным 1, получали, как пример, закодированные сообщения с кодовым расстоянием равным 3, или 4.

4. Выводы

В данной работе были рассмотрены различные методы кодирования, которые являются самокорректирующимися: коды Хэмминга, циклические коды, коды Боуза-Чоудхури-Хоквингема и коды Рида-Соломона.

Кодирование - важный процесс при передаче сигналов по каналам связи. Методы кодирования дополняют методы модуляции для обеспечения улучшения качества передачи, для предотвращения ошибок при передаче, а также защищенности данных от получения их другими лицами.