

11. Код Хэмминга (ЛР)

11.1. Цель работы

Рассмотреть на примере и получить навыки в исследовании кодов Хэмминга с использованием системы компьютерной алгебры Octave.

11.2. Порядок выполнения задания

Задание выполняется каждым учащимся индивидуально. Результаты заданий лабораторного практикума и соответствующих заданий практикума должны совпадать.

Отчёт формируется в электронном виде в формате PDF и отправляется на электронную почту преподавателя.

11.2.1.

Для (n, k) кода Хэмминга $(15, 11)$ получить проверочную матрицу и порождающую матрицу. В системе Octave для этого используется функция `hammgen`, которая получает на вход число проверочных бит $r = n - k$, и вычисляет проверочную и порождающую матрицы, а также выводит n и k .

```
> [H, G, n, k] = hammgen(r)
```

11.2.2.

Закодировать заданный информационный вектор вначале встроенной функцией Octave, затем при помощи умножения на порождающую матрицу. Сравнить результаты. Информационный вектор берется из табл. 11.1 по предпоследней цифре зачетной книжки.

Таблица 11.1

Информационный вектор. По предпоследней цифре номера зачетной книжки

| Цифра | Вектор | Цифра | Вектор |
|-------|-----------------------|-------|-----------------------|
| 1 | 1 0 0 1 1 1 0 0 0 1 0 | 6 | 1 1 1 0 1 1 0 1 0 0 1 |
| 2 | 0 1 1 1 0 1 0 0 0 1 0 | 7 | 0 1 0 0 0 1 0 0 0 0 1 |
| 3 | 1 0 0 1 0 0 1 1 1 0 0 | 8 | 1 0 1 1 0 0 0 0 0 1 1 |
| 4 | 1 0 0 1 0 0 0 0 0 1 0 | 9 | 0 1 0 1 1 0 0 1 0 0 0 |
| 5 | 1 0 0 0 1 1 1 0 1 0 0 | 0 | 0 1 0 1 0 0 1 1 1 0 0 |

Для кодирования используется функция `encode`. В качестве параметров задаются исходное сообщение в двоичном виде, параметры кода n и k и указание использовать код Хэмминга.

```
> Mens = encode(Msg, n, k, "hamming")'
```

Оператор транспонирования «'» указывается, чтобы выводить сообщение строкой, а не столбцом.

Для умножения на порождающую матрицу предварительно необходимо задать информационный вектор и саму матрицу как структуры над простым полем Галуа $GF(2)$.

```
> G2=gf(G,1,3);
> Msg2=gf(Msg,1,3);
```

Далее можно умножать обычным способом.

11.2.3.

Последовательно наложить заданные векторы ошибки на кодовый вектор и декодировать полученные векторы с ошибкой вначале при помощи встро-енной функции Octave, затем посредством проверочной матрицы H по стандартному алгоритму для кодов Хэмминга. Сравнить результаты. Векторы ошибки берутся из табл. 11.2 по последней цифре зачетной книжки. Заданы векто-ры с одной, двумя и тремя ошибками.

Таблица 11.2

Вектор ошибки. По последней цифре номера зачетной книжки

| Цифра | Вектор | Цифра | Вектор |
|-------|------------------|-------|------------------|
| 1 | 0100000000000000 | 6 | 0000001000000000 |
| | 0100100000000000 | | 1000001000000000 |
| | 0100100100000000 | | 1000001000100000 |
| 2 | 0010000000000000 | 7 | 0000000100000000 |
| | 0010100000000000 | | 0100000100000000 |
| | 0010100100000000 | | 0100000100100000 |
| 3 | 0001000000000000 | 8 | 0000000010000000 |
| | 0001010000000000 | | 0010000010000000 |
| | 0001010100000000 | | 0010000010100000 |
| 4 | 0000100000000000 | 9 | 0000000001000000 |
| | 0000101000000000 | | 1000000001000000 |
| | 0000101010000000 | | 1000010001000000 |
| 5 | 0000010000000000 | 0 | 0000000000100000 |
| | 0000010100000000 | | 0001000000100000 |
| | 0000010101000000 | | 0001001000100000 |

Для наложения ошибки используется функция xor.

```
> Merr1=xor(Menc,Err1)
```

Для декодирования используется функция decode. В качестве парамет-ров задаются исходное сообщение в двоичном виде, параметры кода n и k и указание использовать код Хэмминга. На выходе функция возвращает деко-дированное сообщение Mdec и вектор ошибок err.

```
> Mdec=decode(Merr1,n,k,"hamming")'
```

Для декодирования по стандартному алгоритму для кодов Хэмминга необходимо произвести умножение на транспонированную проверочную матрицу H . Предварительно необходимо задать вектор с ошибкой и саму матрицу как структуры над простым полем Галуа GF(2).

```
> H2=gf(H,1,3);  
> Merr21=gf(Merr1,1,3);
```

Далее можно умножать обычным способом.

11.2.4.

Сравнить по методу Монте-Карло вероятностные характеристики двух кодов Хэмминга согласно варианту. Для этого воспользоваться написанной в листинге 11.1 программой, изменив ее для своих нужд, подставив необходимые параметры кодов. В результате выполнения будет получен график, который необходимо проанализировать. График и выводы должны быть представлены в отчете. Также необходимо проанализировать текст самой программы и разобраться в ее работе.

Листинг 11.1

Листинг программы для сравнения двух кодов Хэмминга по вероятности битовой ошибки в канале ДСК

```
1 r1=3;  
2 r2=4;  
3 %  
4 [H1,G1,n1,k1]=hammgen(r1);  
5 [H2,G2,n2,k2]=hammgen(r2);  
6 s1=sprintf("Hamming code (%d,%d)",n1,k1);  
7 s2=sprintf("Hamming code (%d,%d)",n2,k2);  
8 %  
9 p0=[5e-4 1e-3 5e-3 1e-2 5e-2 1e-1];  
10 stat=zeros(2,6);  
11 %  
12 msg1=randi([0 1],1e5,k1);  
13 msg2=randi([0 1],1e5,k2);  
14 %  
15 menc1=encode(msg1,n1,k1,"hamming");  
16 menc2=encode(msg2,n2,k2,"hamming");  
17 %  
18 for i=1:1:6  
19     mrec1=bsc(menc1,p0(i));  
20     mdec1=decode(mrec1,n1,k1,"hamming");  
21     [num,rate]=biterr(msg1,mdec1);  
22     stat(1,i)=rate;  
23     mrec2=bsc(menc2,p0(i));  
24     mdec2=decode(mrec2,n2,k2,"hamming");  
25     [num,rate]=biterr(msg2,mdec2);  
26     stat(2,i)=rate;  
27 end
```

```

28 %
29 format long;
30 %
31 stat
32 %
33 mfig=figure;
34 L1=loglog(p0,stat(1,:));
35 set(L1,"LineWidth",3,"Color","k");
36 hold on;
37 L2=loglog(p0,stat(2,:));
38 set(L2,"LineWidth",3,"Color","b");
39 hold on;
40 title(sprintf("Hamming codes (%d,%d) and (%d,%d) in BSC
    channel",n1,k1,n2,k2));
41 xlabel("BER in BSC channel, p0");
42 ylabel("Error rate after decoding");
43 legend(s1,s2,3);
44 legend("show");
45 grid on;
46 print(mfig,'-dpng',sprintf("ham-%d-%d_ham-%d-%d_bsc_err-rate"
    ,n1,k1,n2,k2));

```

Программу необходимо сохранить в виде файла с расширением *.m и запускать из командной строки (из каталога, в котором лежит программа) так, как указано ниже.

```
user@name:[~]$ octave -q hamming_compar.m
```

Параметры кодов указаны в табл. 11.3. Выбор производится по последней цифре номера зачетной книжки.

Таблица 11.3

Параметры кодов Хэмминга для сравнения.
(По последней цифре номера зачетной книжки)

| Цифра | Код 1 (n,k,r) | Код 2 (n,k,r) | Цифра | Код 1 (n,k,r) | Код 2 (n,k,r) |
|-------|----------------------|----------------------|-------|----------------------|----------------------|
| 1 | (7,4,3) | (31,26,5) | 6 | (15,11,4) | (127,120,7) |
| 2 | (7,4,3) | (63,57,6) | 7 | (15,11,4) | (255,247,8) |
| 3 | (7,4,3) | (127,120,7) | 8 | (31,26,5) | (127,120,7) |
| 4 | (7,4,3) | (255,247,8) | 9 | (31,26,5) | (255,247,8) |
| 5 | (15,11,4) | (63,57,6) | 0 | (63,57,6) | (255,247,8) |

11.3. Пример выполнения работы для кода (7,4) (только основные команды)

```

> [H,G,n,k]= hamngen(3)
> Msg=[1 0 1 0]
> Menc=encode(Msg,n,k,"hamming")
> G2=gf(G,1,3);

```

```
> Msg2=gf(Msg,1,3);
> Menc2=Msg2*G2
> Err1=[0 1 0 0 0 0 0]
> Merr1=xor(Menc,Err1)
> Mdec1=decode(Merr1,n,k,"hamming")'
> H2=gf(H,1,3);
> Merr21=gf(Merr1,1,3);
> S1=Merr21*H2'
> Err2=[0 1 0 1 0 0 0]
...
> Err3=[0 1 0 1 0 1 0]
...
> exit
```

11.4. Порядок защиты практической работы

Защита работы может осуществляться одним из нижеперечисленных способов или их сочетанием на усмотрение преподавателя.

1. Устный ответ по теме работы.
2. Тестирование по теме работы
3. Задача по теме работы.
4. Иные варианты на усмотрение преподавателя.