ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ФАКУЛЬТЕТ ТЕЛЕКОММУНИКАЦИЙ

ОСНОВЫ ПЕРЕДАЧИ ДАННЫХ

(часть 1)

Методические указания по выполнению практических работ

Ташкент 2013

). Методические
с. Ташкент, 2013

акультета и реко-

изовать процесс
передачи данных».

чебном процессе
никации». Продол-
указаниях, состав-

Ш. Ю. Джаббаров

Л. Х. Хамдам-Заде

ПРЕДИСЛОВИЕ

Современный этап развития мировой цивилизации характеризуется переходом от индустриального к информационному обществу, предполагающим наличие новых форм социальной и экономической деятельности, которые базируются на массовом использовании информационно-коммуникационных технологий (ИКТ). В современном динамично развивающемся обществе ИКТ являются движущей силой и основой экономического развития.

Внедрение ИКТ предусматривает создание такой телекоммуникационной инфраструктуры, которая смогла бы объединить в себе все возможные виды информации (речь, данные, мультимедиа) и удовлетворяла бы требованиям каждого из них к качеству обслуживания, а операторы заинтересованы в построении такой сети связи, которая бы поддерживала непрерывный контроль процессов обработки вызовов клиента и предоставления услуг по одним и тем же правилам, гарантирующим запрошенный уровень качества обслуживания, независимо от способов транспортировки данных и видов используемого оборудования. Учитывая, что в основе ИКТ лежит применение систем передачи данных и телекоммуникационных технологий, возникает необходимость изучения особенностей передачи данных в рамках таких технологий. В обществе, оснащённом ИКТ, грамотность в области этих технологий становится необходимым условием для учёбы и работы молодёжи.

Данное методическое пособие имеет целью обучение студентов бакалавриата направления «Телекоммуникации» практическому применению знаний по материалам основных разделов дисциплины «Основы передачи данных».

Практическая работа №1

РАСЧЕТ ОБЪЕМА ИНФОРМАЦИИ И ЭНТРОПИИ

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными количественными характеристиками сообщений;

- получения базовых практических навыков расчета количества информации и информационной энтропии сообщений.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Информация – это совокупность сведений о каком-нибудь явлении или объекте, увеличивающие наши знания об этом явлении или объекте и предназначена для передачи, распределения, преобразования, хранения или непосредственного использования. Это могут быть сведения о результатах измерения, наблюдения за каким-либо объектом и т.п.

Сообщение является формой представления информации. Одно и то же сведение может быть представлено в различной форме. Например, сведение о часе приезда вашего приятеля может быть передано по телефону или же в виде электронного сообщения. В первом случае мы имеем дело с информацией, представленной в непрерывном виде (непрерывное сообщение). Будем считать, что это сообщение вырабатывается некоторым источником – в данном случае источником непрерывных сообщений. Во втором случае – с информацией, представленной в дискретном виде (дискретное сообщение). Это сообщение вырабатывается источником дискретных сообщений.

Дискретными называются сообщения, составленные из заранее известного конечного набора символов. В отличие от дискретных, непрерывные сообщения (речь, музыка, телевидение) складываются из бесконечного количества элементов.

При передаче сведений по электронной почте информация заложена в буквах, из которых составлены слова, и цифрах. Очевидно, что на конечном отрезке времени число букв или цифр является конечным. Это и является отличительной особенностью дискретного сообщения. В то же время число различных возможных значений звукового давления, измеренное при разговоре, даже на конечном отрезке времени, будет бесконечным. В современных цифровых системах телефонной связи в канал связи передаются кодовые комбинации, несущие информацию об отсчетах квантованного аналогового сигнала. Следовательно, такой телефонный квантованный сигнал относится к классу дискретных.

К основным информационным характеристикам сообщений относятся такие характеристики, как количество информации в сообщении и информационная энтропия.

Чем меньше вероятность появления того или иного сообщения, тем большее количество информации мы извлекаем при его получении. Если в памяти источника имеется два независимых сообщения a_1 и a_2 и первое из них выдается с вероятностью $P(a_1) = 1$, то сообщение a_1 не несет информации, ибо оно заранее известно получателю.

Было предложено определять количество информации $I(a_i)$, которое приходится на одно сообщение a_i, следующим выражением:

$$I(a_i) = \log_2 \frac{1}{P(a_i)} = -\log_2 P(a_i),$$

где $P(a_i)$ – вероятность появления сообщения a_i.

Для примера рассмотрим систему, в которой источник сообщений генерирует с одинаковой вероятностью два сообщения a_1 и a_2, т.е. $P(a_1) =$

$P(a_2) = 0{,}5$. Тогда количество информации, переносимое каждым сообщением будет определено следующим образом:

$$I(a_1) = \log_2 \frac{1}{P(a_1)} = \log_2 \frac{1}{0{,}5} = \log_2 2 = 1 \text{ bit},$$

$$I(a_2) = \log_2 \frac{1}{P(a_2)} = \log_2 \frac{1}{0{,}5} = \log_2 2 = 1 \text{ bit}.$$

Рассмотренное соотношение справедливо для сообщений, появление которых равновероятно. Если же вероятности $P(a_i)$ появления различных сообщений a_i различны, то среднее количество информации в одном сообщении можно определить по следующей формуле:

$$H(a) = -\sum_{i=1}^{n} P(a_i) \cdot \log_2 P(a_i).$$

Последнее выражение известно как формула Шеннона для энтропии источника дискретных сообщений. Энтропия – мера неопределенности в поведении источника дискретных сообщений. Энтропия равна нулю, если с вероятностью единица источником выдается всегда одно и то же сообщение (в этом случае неопределенность в поведении источника сообщений отсутствует). Энтропия максимальна, если символы источника появляются независимо и с одинаковой вероятностью.

3. ЗАДАНИЕ

Задача 1

Источник генерирует с одинаковой вероятностью сообщения, состоящие из N символов алфавита, содержащего A символов (табл. 1.1). Найти количество информации, содержащееся в каждом сообщении.

Таблица 1.1

Варианты для задач 1 и 2

Вариант	группа 1		группа 2		группа 3	
	N	A	N	A	N	A
1	3	16	3	22	3	28
2	4	16	4	22	4	28
3	5	16	5	22	5	28
4	6	16	6	22	6	28
5	7	16	7	22	7	28
6	3	17	3	23	3	29
7	4	17	4	23	4	29
8	5	17	5	23	5	29
9	6	17	6	23	6	29
10	7	17	7	23	7	29
11	3	18	3	24	3	30
12	4	18	4	24	4	30
13	5	18	5	24	5	30
14	6	18	6	24	6	30
15	7	18	7	24	7	30
16	3	19	3	25	3	31

Продолжение табл. 1.1

Вариант	группа 1		группа 2		группа 3	
	N	*A*	*N*	*A*	*N*	*A*
17	4	19	4	25	4	31
18	5	19	5	25	5	31
19	6	19	6	25	6	31
20	7	19	7	25	7	31
21	3	20	3	26	3	32
22	4	20	4	26	4	32
23	5	20	5	26	5	32
24	6	20	6	26	6	32
25	7	20	7	26	7	32

Задача 2

Источник генерирует с одинаковой вероятностью сообщения, которые могут включать в себя от 1 до N символов алфавита, содержащего A символов (табл. 1.1). Найти количество информации, содержащееся в каждом сообщении.

Задача 3

Источник генерирует сообщения a_1, a_2, a_3 и a_4 с вероятностями P_1, P_2, P_3 и P_4 соответственно (табл. 1.2). Найти информационную энтропию данного источника сообщений.

Таблица 1.2

Варианты для задачи 3

Вариант	группа 1				группа 2				группа 3			
	P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4
1	0,44	0,35	0,15	0,06	0,40	0,21	0,24	0,15	0,45	0,30	0,17	0,08
2	0,43	0,34	0,16	0,07	0,27	0,08	0,37	0,28	0,44	0,29	0,18	0,09
3	0,42	0,33	0,17	0,08	0,39	0,20	0,25	0,16	0,41	0,26	0,21	0,12
4	0,41	0,32	0,18	0,09	0,26	0,07	0,38	0,29	0,40	0,25	0,22	0,13
5	0,40	0,31	0,19	0,10	0,38	0,19	0,26	0,17	0,37	0,22	0,25	0,16
6	0,39	0,30	0,20	0,11	0,25	0,06	0,39	0,30	0,36	0,21	0,26	0,17
7	0,38	0,29	0,21	0,12	0,37	0,18	0,27	0,18	0,33	0,18	0,29	0,20
8	0,37	0,28	0,22	0,13	0,49	0,30	0,15	0,06	0,32	0,17	0,30	0,21
9	0,36	0,27	0,23	0,14	0,36	0,17	0,28	0,19	0,29	0,14	0,33	0,24
10	0,35	0,26	0,24	0,15	0,48	0,29	0,16	0,07	0,28	0,13	0,34	0,25
11	0,34	0,25	0,25	0,16	0,35	0,16	0,29	0,20	0,25	0,10	0,37	0,28
12	0,33	0,24	0,26	0,17	0,47	0,28	0,17	0,08	0,24	0,09	0,38	0,29
13	0,32	0,23	0,27	0,18	0,34	0,15	0,30	0,21	0,47	0,32	0,15	0,06
14	0,31	0,22	0,28	0,19	0,46	0,27	0,18	0,09	0,46	0,31	0,16	0,07
15	0,30	0,21	0,29	0,20	0,33	0,14	0,31	0,22	0,43	0,28	0,19	0,10
16	0,29	0,20	0,30	0,21	0,45	0,26	0,19	0,10	0,42	0,27	0,20	0,11
17	0,28	0,19	0,31	0,22	0,32	0,13	0,32	0,23	0,39	0,24	0,23	0,14
18	0,27	0,18	0,32	0,23	0,44	0,25	0,20	0,11	0,38	0,23	0,24	0,15
19	0,26	0,17	0,33	0,24	0,31	0,12	0,33	0,24	0,35	0,20	0,27	0,18
20	0,25	0,16	0,34	0,25	0,43	0,24	0,21	0,12	0,34	0,19	0,28	0,19
21	0,24	0,15	0,35	0,26	0,30	0,11	0,34	0,25	0,31	0,16	0,31	0,22
22	0,23	0,14	0,36	0,27	0,42	0,23	0,22	0,13	0,30	0,15	0,32	0,23

Продолжение табл. 1.2

Вариант	группа 1				группа 2				группа 3			
	P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4	P_1	P_2	P_3	P_4
23	0,22	0,13	0,37	0,28	0,29	0,10	0,35	0,26	0,27	0,12	0,35	0,26
24	0,21	0,12	0,38	0,29	0,41	0,22	0,23	0,14	0,26	0,11	0,36	0,27
25	0,20	0,11	0,39	0,30	0,28	0,09	0,36	0,27	0,23	0,08	0,39	0,30

Задача 4

Источник генерирует сообщения, состоящие из двух символов алфавита, содержащего 2 символа. Вероятность P_1 появления одного из символов приведена в табл. 1.3. Найти информационную энтропию данного источника сообщений.

Таблица 1.3

Варианты для задач 4 и 5

Вариант	группа 1	группа 2	группа 3
	P_1	P_1	P_1
1	0,490	0,466	0,445
2	0,480	0,456	0,435
3	0,470	0,446	0,425
4	0,460	0,436	0,415
5	0,450	0,426	0,405
6	0,440	0,416	0,395
7	0,430	0,406	0,385
8	0,420	0,396	0,375

Продолжение табл. 1.3

Вариант	группа 1	группа 2	группа 3
	P_1	P_1	P_1
9	0,410	0,386	0,365
10	0,400	0,376	0,355
11	0,390	0,366	0,345
12	0,380	0,356	0,335
13	0,370	0,346	0,325
14	0,360	0,336	0,315
15	0,350	0,326	0,305
16	0,340	0,316	0,295
17	0,330	0,306	0,285
18	0,320	0,296	0,275
19	0,310	0,286	0,265
20	0,300	0,276	0,255
21	0,290	0,266	0,245
22	0,280	0,256	0,235
23	0,270	0,246	0,225
24	0,260	0,236	0,215
25	0,250	0,226	0,205

Задача 5

Источник генерирует сообщения, состоящие из одного или двух символов алфавита, содержащего 2 символа. Вероятности появления 1- и 2-символьного сообщений одинаковы. Вероятность P_1 появления одного из символов приведена в табл. 1.3. Найти информационную энтропию данного источника сообщений.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое информация?

2) Что является формой представления информации?

3) Что такое сообщение?

4) Какие типы сообщений существуют?

5) Чем различаются типы сообщений между собой?

6) Какое сообщение называется непрерывным?

7) Какое сообщение называется дискретным?

8) Какие основные информационные характеристики сообщений существуют?

9) Как связано количество информации в сообщении с вероятностью появления сообщения?

10) Какое количество информации содержит сообщение, вероятность появления которого равна 1?

11) Какое количество информации содержит сообщение, вероятность появления которого равна P?

12) Что такое информационная энтропия?

13) Как вычисляется информационная энтропия?

14) Чему равна энтропия, если источник генерирует только одно сообщение с вероятностью 1?

15) Какое максимально возможное количество сообщений длиной N символов можно составить из алфавита, содержащего A символов?

16) Какое максимально возможное количество сообщений длиной от 1 до N символов можно составить из алфавита, содержащего A символов?

17) Что такое бит, трит, тетрит, нат, дит?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

Практическая работа №2

ЭФФЕКТИВНЫЕ МЕТОДЫ КОДИРОВАНИЯ ИНФОРМАЦИИ

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными принципами кодирования информации;
- получения базовых практических навыков построения эффективных кодов.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Сигнал – это физический процесс, параметры которого изменяются в соответствии с передаваемым сообщением.

Сигналы, формируемые на выходе преобразователя дискретного сообщения в сигнал, как правило, описываются функцией дискретного времени и конечным множеством возможных значений. В технике передачи данных такие сигналы называют цифровыми сигналами данных.

Параметр сигнала данных, изменение которого отображает изменение сообщения, называется информационным параметром сигнала данных.

Под кодированием понимается процесс преобразования дискретного сообщения в сигнал. Тогда кодом будет называться соответствие между алфавитом сообщения, т.е. набором символов, и значениями информационного параметра сигнала, формируемыми в процессе передачи.

Коды можно разделить на две самостоятельные группы. К первой относятся коды, использующие все возможные комбинации – неизбыточные коды. Ко второй группе относятся коды, использующие лишь определенную часть всех возможных комбинаций, такие коды называются избыточными. Оставшаяся часть комбинаций используется для обнаружения или исправления ошибок, возникающих при передаче сообщений.

Обе группы кодов, в свою очередь, подразделяются на равномерные и неравномерные. Равномерные коды – это коды, все кодовые комбинации которых содержат постоянное количество разрядов. Неравномерные коды содержат кодовые комбинации с различным числом разрядов.

При передаче сообщений, закодированных равномерным кодом, не учитывается статистическая структура передаваемых сообщений.

Из теоремы Шеннона о кодировании сообщений в каналах без шумов следует следующее:

если передача дискретных сообщений ведется при отсутствии помех, то всегда можно найти такой метод кодирования, при котором среднее число двоичных символов на одно сообщение будет сколь угодно близким к энтропии источника этих сообщений, но никогда не может быть меньше ее.

Учет статистики сообщений на основании теоремы Шеннона позволяет строить код, в котором часто встречающимся сообщением присваиваются более короткие кодовые комбинации, а редко встречающимся – более длинные.

2.1. Алгоритм Шеннона-Фано

Первым подобным кодом стал код Шеннона-Фано. Код Шеннона-Фано является префиксным. Префиксный код – это неравномерный код, обладающий следующим свойством: если в код входит слово *a*, то для любой непустой строки *b* слова *ab* в коде не существует. Хотя префиксный код состоит из слов разной длины, эти слова можно записывать без разделительного символа.

Код Шеннона-Фано строится следующим образом:

1. Символы (сообщения) выписывают в порядке убывания вероятностей.

2. Упорядоченные символы делят на две группы, суммарные вероятности которых максимально близки друг другу.

3. Первой группе присваивается «0», второй – «1».

4. Полученные группы рекурсивно делятся и их подгруппам назначаются соответствующие двоичные цифры.

Основной принцип, положенный в основу кодирования по методу Шеннона-Фано заключается в том, что при выборе каждой цифры кодового слова стремятся, чтобы содержащееся в ней количество информации было наибольшее.

Для примера рассмотрим сообщение, для которого справедливо следующее распределение количества появлений символов:

Символ	A	B	C	D	E	F	G
Количество появлений	5	4	4	4	3	2	1

Энтропия для описанного выше сообщения $H(x) = 2{,}682$ bit.

Так как используется всего 7 символов, то для кодирования каждого из них равномерным кодом достаточно будет использовать 3 bit.

Теперь рассмотрим кодирование методом Шеннона-Фано.

<table>
<tr><th>Символ</th><th>Количество появлений</th><th colspan="4">Вспомогательная таблица</th><th>Код</th></tr>
<tr><td>A</td><td>5</td><td rowspan="3">0</td><td colspan="3">0</td><td>00</td></tr>
<tr><td>B</td><td>4</td><td rowspan="2">1</td><td colspan="2">0</td><td>010</td></tr>
<tr><td>C</td><td>4</td><td colspan="2">1</td><td>011</td></tr>
<tr><td>D</td><td>4</td><td rowspan="4">1</td><td colspan="3">0</td><td>10</td></tr>
<tr><td>E</td><td>3</td><td rowspan="3">1</td><td colspan="2">0</td><td>110</td></tr>
<tr><td>F</td><td>2</td><td rowspan="2">1</td><td>0</td><td>1110</td></tr>
<tr><td>G</td><td>1</td><td>1</td><td>1111</td></tr>
</table>

Зная количество бит n_i в каждом символе a_i при использовании кода Шеннона-Фано, можно рассчитать среднее количество бит n_{cp}, приходящееся на каждый символ:

$$n_{cp} = \sum_{i=1}^{7} \left[P(a_i) \cdot n_i \right] = 2{,}739 \text{ bit}.$$

Таким образом, среднее количество бит, приходящееся на один символ кода, при использовании кода Шеннона-Фано меньше, чем при использовании равномерного кода, и ближе к значению энтропии.

2.2. Алгоритм Хаффмена

Еще одним неравномерным кодом, учитывающим статистическую структуру сообщений, является код Хаффмена.

Метод построения кода Хаффмена сводится к следующему:

1. Исходные сообщения записываются в порядке их убывания.

2. Выбирается две наименьших по вероятности сообщения, и создается их «родитель» – сумма двух этих сообщений, вероятность которой определяется как сумма вероятностей исходных сообщений.

3. Пункт 2 повторяется до тех пор, пока не будет найден главный «родитель».

4. Чтобы составить кодовую комбинацию, соответственно данному сообщению, необходимо проследить путь перехода сообщения от главного «родителя». Каждой ветви, выходящей из одного узла, присваивается двоичный символ. Причем ветви с большей вероятностью присваивается «1», а с меньшей – «0».

Для примера рассмотрим сообщение, приведенное в примере для кода Шеннона-Фано, и построим кодовое дерево (рис. 2.1).

Получается, что символы сообщения будут закодированы следующим образом:

Символ	A	B	C	D	E	F	G
Код	01	00	111	110	101	1001	1000

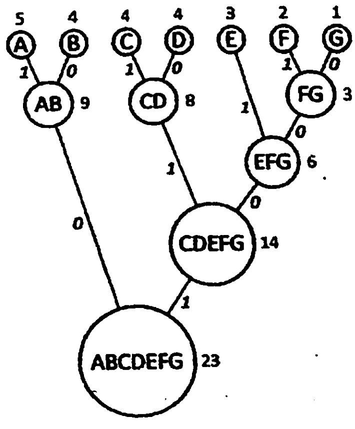

Рис. 2.1. Пример кодового дерева

Также как и для кода Шеннона-Фано, зная количество бит n_i в каждом символе a_i при использовании кода Хаффмена, можно рассчитать среднее количество бит $n_{ср}$, приходящееся на каждый символ:

$$n_{ср} = \sum_{i=1}^{7}\left[P(a_i)\cdot n_i\right] = 2{,}739 \text{ bit}.$$

Таким образом, среднее количество бит, приходящееся на один символ кода, при использовании кода Хаффмена, равно среднему количеству символов при использовании кода Шеннона-Фано и меньше, чем при использовании равномерного кода.

При построении кода Шеннона-Фано разбиение множества элементов может быть произведено несколькими способами. Выбор разбиения на каждом уровне может ухудшить варианты разбиения на следующем уровне и привести к неоптимальности кода в целом. Поэтому код Шеннона-Фано не является оптимальным в общем смысле, хотя и дает оптимальные результаты при некоторых распределениях вероятностей. Коды Хаффмена данным недостатком не обладают. Для одного и того же распределения вероятностей можно построить несколько кодов Шеннона-Фано, и все они могут дать различные результаты. Если построить все возможные коды Шеннона-Фано для данного распределения вероятностей, то среди них будут находиться и все коды Хаффмана, то есть оптимальные коды.

3. ЗАДАНИЕ

1. Определить частоту появления каждого символа (включая знаки препинания и пробелы) в сообщении.

2. Найти информационную энтропию исходного сообщения.

3. Определить минимальное количество бит, необходимое для кодирования всех символов исходного сообщения равномерным кодом.

4. Закодировать все символы исходного сообщения (включая знаки препинания и пробелы), используя алгоритм Шеннона-Фано. Определить среднее количество бит, приходящееся на каждый символ сообщения.

5. Закодировать все символы исходного сообщения (включая знаки препинания и пробелы), используя алгоритм Хаффмена. Определить среднее количество бит, приходящееся на каждый символ сообщения.

6. Сделать выводы по полученным результатам.

Отчет о данной работе должен содержать:

- номер и название работы;
- задание в соответствии с вариантом;
- минимальное количество бит, необходимое для кодирования всех символов исходного сообщения равномерным кодом;
- вспомогательная таблица кодирования по алгоритму Шеннона-Фано;
- среднее количество бит, приходящееся на каждый символ сообщения, при кодировании с использованием алгоритма Шеннона-Фано;
- кодовое дерево, использовавшееся при кодировании по алгоритму Хаффмена;
- среднее количество бит, приходящееся на каждый символ сообщения, при кодировании с использованием алгоритма Хаффмена;
- выводы по полученным результатам.

Таблица 2.1

Варианты для выполнения практической работы №2

Вариант	Исходное сообщение
1	система передачи данных, обеспечивающая двухстороннюю передачу информации, может содержать большое число источников и получателей сообщений, передатчиков, приемников и линий связи.
2	требования, предъявляемые к системам передачи данных, удобнее рассматривать на примере системы одностороннего действия с одним получателем сообщений. кодовые комбинации первичного
3	кода поступают от источника информации в передатчик, который состоит из устройства защиты от ошибок и устройства преобразования сигналов. принятые по каналу связи сигналы поступают
4	к получателю информации. важнейшими требованиями, которые получатель информации предъявляет к системе передачи данных, являются требования к верности передачи, надежности функционирования
5	системы передачи данных и сроку доставки информации или скорости передачи информации. в систему передачи дискретной информации может входить дискретный канал, либо канал передачи данных.
6	источник информации и получатель информации в состав системы передачи данных не входят. дискретный канал представляет собой совокупность устройства преобразования сигнала и канала связи
7	и предназначен для передачи дискретных сигналов. устройство преобразования сигналов предназначено для спектрального согласования, а также для согласования по амплитуде источника информации
8	с каналом связи. в пункте приема согласуется по амплитуде выход канала связи с входом получателя информации. дискретный канал характеризуется максимально допустимой скоростью модуляции,
9	вероятностью ошибки и надежностью. никаких специальных мер для повышения верности передачи и надежности дискретного канала не предусмотрено. следовательно, качество передачи информации

Продолжение табл. 2.1

Вариант	Исходное сообщение
10	в системе передачи данных целиком зависит от реально существующих характеристик канала и устройства преобразования сигнала. канал передачи данных представляет собой совокупность дискретного
11	канала и устройств защиты от ошибок. наличие устройств защиты от ошибок гарантирует заданную верность передачи. с целью повышения верности передачи требуется водить избыточность в
12	сообщение, что снижает пропускную способность системы. канал передачи данных характеризуется заданной верностью. эффективной скоростью передачи информации, и надежностью. тракт передачи
13	данных представляет собой совокупность взаимно резервированных каналов передачи данных (двух или более) и групповых устройств. тракт обеспечивает передачу информации с заданными эффективной
14	скоростью, верностью и надежностью. приведённые требования являются основными и далеко не исчерпывают всех требований, предъявляемых к системе передачи данных. любая система передачи
15	данных может быть описана через три основные свои компоненты. такими компонентами являются передатчик, канал передачи данных и приёмник. при двухсторонней передаче источник и получатель
16	передача данных является качественно новым видом телекоммуникаций, и поэтому её основные технические характеристики отличаются следующими: верность или безошибочность передачи данных
17	характеризуется коэффициентом ошибок, численно равным отношению количества ошибочно принятых знаков к общему количеству переданных знаков. обычно при передаче данных коэффициент ошибок

Продолжение табл. 2.1

Вариант	Исходное сообщение
18	не должен превышать одной миллионной. это требование определяется важностью передаваемых данных. для повышения верности приема информации принимает специальные меры. так, например, можно
19	одну и ту же информацию передавать по нескольким независимым путям, а на приеме производить выборку голосованием. чем больше независимых путей, тем меньше вероятность неправильного приема.
20	аналогичную операцию можно произвести, если одно и то же сообщение передавать по одному каналу несколько раз. верность приема можно увеличить, использовав помехоустойчивые коды. пропускная
21	способность при передаче данных должна быть как можно высокой, поскольку требуется передавать больше объема информации в жестко заданный отрезок времени. увеличение пропускной способности
22	достигается за счет значительного увеличения скорости передачи сигналов данных, а также за счет применения более широкополосных каналов связи. пропускная способность системы передачи
23	дискретных сообщений или скорость передачи информации, зависит в основном от ширины спектра частот, отводимого для работы системы, а также от вида модуляции и интенсивности помех.
24	ширина канала определяет длительность нарастания переходного процесса при модуляции какого-либо параметра несущих частоты. надежность работы элементов устройств передачи данных должна
25	быть достаточно высокой, чтобы обеспечить бесперебойное функционирование системы, для которой передаются эти данные. надёжность системы передачи данных характеризуется средним временем

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое сигнал?

2) Какие сигналы называют непрерывными?

3) Какие сигналы называют дискретными?

4) Что такое информационный параметр сигнала?

5) Что такое код?

6) Что такое кодирование?

7) Какие коды называются избыточными?

8) Какие коды называются неизбыточными?

9) Какие коды называются равномерными?

10) Какие коды называются неравномерными?

11) Каковы преимущества использования равномерных кодов перед использованием неравномерных?

12) Каковы преимущества использования неравномерных кодов перед использованием равномерных?

13) Теорема Шеннона о кодировании сообщений в каналах без шумов.

14) Что такое префиксный код?

15) Каковы преимущества использования префиксного кода перед использованием непрефиксного кода?

16) Алгоритм Шеннона-Фано.

17) Алгоритм Хаффмена.

18) Каковы недостатки алгоритма Шеннона-Фано?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

Практическая работа №3

ИЗУЧЕНИЕ ПРИНЦИПОВ ФОРМИРОВАНИЯ ПОМЕХОУСТОЙЧИВЫХ КОДОВ

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными принципами помехоустойчивого кодирования;
- получения базовых практических навыков по формирования помехоустойчивых кодов.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Под кодированием понимается процесс преобразования дискретного сообщения в сигнал. Тогда кодом будет называться соответствие между алфавитом сообщения, т.е. набором символов, и значениями информационного параметра сигнала, формируемыми в процессе передачи.

Любой код характеризуется следующими параметрами:

- значность кода n – количество элементов в кодовой комбинации;
- кодовое расстояние d – количество элементов, которые отличают одну кодовую комбинацию от другой;

- минимальное кодовое расстояние (расстояние Хэмминга) d_0 – минимальное количество элементов, которые отличают одну кодовую комбинацию от другой.

Значность кода n выбирается таким образом, чтобы обеспечить возможность передачи всех символов алфавита сообщения. Например, если нам нужно закодировать двоичным кодом сообщения, алфавит которых включает в себя A символа, то значность кода можно найти из следующего соотношения:

$$n \geq \log_2 A.$$

Если $A = 32$, то минимальное (достаточное) значение $n = 5$. При этом каждый символ алфавита будет закодирован 5 значениями информационного параметра сигнала и все возможные комбинации значений информационного параметра сигнала будут задействованы. В данном случае минимальное кодовое расстояние $d_0 = 1$.

Однако, в общем случае, для рассматриваемого алфавита ($A = 32$) может выть выбрано значение $n \geq 5$. Допустим, что было выбрано $n = 6$, т.е. каждый символ алфавита будет закодирован комбинацией из 6 значений информационного параметра сигнала. Всего таких комбинаций может быть составлено $2^6 = 64$. Однако нам требуются только 32 комбинации для кодирования всех символов алфавита. Следовательно, для кодирования алфавита будет использована только половина всех доступных комбинаций. Те комбинации, которые будут использованы для кодирования символов алфавита, называются разрешенными кодовыми комбинациями, а все остальные – запрещенными.

Отношение количества запрещенных кодовых комбинаций к общему количеству комбинаций называется избыточностью кода.

Очевидно, что передатчик будет передавать в канал связи только разрешенные кодовые комбинации. Поэтому прием запрещенной комбинации будет свидетельствовать о произошедшей при передаче ошибке.

Вывод о том какая кодовая комбинация передавалась, делается на основании сравнения принятой запрещенной комбинации со всеми разрешенными. Принятая комбинация отождествляется с той комбинацией, от которой она отличается меньше всего. Таким образом, ошибка может быть исправлена. Если же принятая запрещенная комбинация отличается хотя бы от двух разрешенных комбинаций на одинаковое количество символов, то исправить ошибку будет невозможно.

Код, для которого $d_0 = 1$, называется простым. В таком коде невозможно ни обнаружить, ни исправить возникшую ошибку, т.к. при возникновении ошибки одна разрешенная комбинация переходит в другую разрешенную комбинацию, Очевидно, что чем больше значение d_0, тем лучше код обнаруживает и исправляет ошибки. Но при больших значениях n перебор кодовых комбинаций для сравнения может потребовать значительных ресурсов. Кроме того, это приводит к уменьшению скорости передачи. Поэтому, основное направление помехоустойчивого кодирования заключается в поисках таких кодов, для которых кодирование и декодирование осуществляются не перебором кодовых комбинаций для сравнения, а с помощью некоторых математических операций.

Вся теория помехоустойчивого кодирования строится на основной теореме Шеннона о кодировании для канала с помехами:

1) при любой производительности источника сообщений меньше, чем пропускная способность канала, существует такой способ кодирования, который позволяет обеспечить передачу всей информации, создаваемой источником сообщения со сколь угодно малой вероятностью ошибки;

2) <u>не существует способа кодирования, позволяющего вести передачу информации со сколь угодно малой вероятностью ошибки, если производительность источника сообщений больше пропускной способности канала.</u>

Код, способный обнаружить или исправить ошибки, называют помехоустойчивым или корректирующим.

Помехоустойчивые коды строятся так, что для передачи сообщения используются не все кодовые комбинации, а лишь некоторая их часть (разрешенные кодовые комбинации). Тем самым создается возможность обнаружения и исправления ошибки при неправильном воспроизведении некоторого числа символов. Корректирующие свойства кодов достигаются введением в кодовые комбинации дополнительных (избыточных) символов.

Помехоустойчивые коды подразделяются на два основных вида:

- блочные;
- непрерывные.

К блочным относятся коды, в которых каждому символу алфавита сообщений соответствует кодовая комбинация (блок) из n_i элементов, где i – номер сообщения. Если длины блоков всех сообщений имеют одинаковую длину, то код называется равномерным. Если блоки отличаются длиной, то блочный код называется неравномерным.

В непрерывных кодах передаваемая информационная последовательность не разбивается на блоки, а проверочные элементы размещаются в определенном порядке между информационными.

Равномерные блочные коды делятся на разделимые и неразделимые. В разделимых кодах элементы разделяются на информационные и проверочные, занимающие определенные места в кодовой комбинации. В неразделимых кодах отсутствует деление элементов кодовых комбинаций на информационные и проверочные.

Разделимые коды подразделяются на систематические и несистематические. Систематическими называют коды, у которых сумма по модулю два двух разрешенных кодовых комбинаций дает разрешенную кодовую комбинацию того же кода. Особенностью этих кодов является то, что информационные и проверочные элементы связаны между собой зависимостями, которые описываются линейными уравнениями.

Циклические коды обеспечивают обнаружение и исправление, как независимых ошибок (одиночных и многократных), так и групповых ошибок. Циклические коды являются блочными систематическими кодами. Кроме того, если в разрешенной кодовой комбинации циклического кода осуществить циклическую перестановку элементов, то в результате получится другая разрешенная кодовая комбинация, принадлежащая данному коду.

Циклический код может быть описан с помощью следующих параметров:

- количество информационных разрядов – k;
- количество проверочных разрядов – r;
- длина кода – $n = k + r$;
- общее количество кодовых комбинаций – $N = 2^n$;
- количество разрешенных кодовых комбинаций – $N_P = 2^k$;
- избыточность кода – r/n;
- скорость кода – k/n;
- кодовое расстояние – d;
- минимальное кодовое расстояние – d_0;
- вероятность необнаружения ошибки – $P_{но}$;
- кратность обнаруживаемой ошибки – δ;
- кратность исправляемой ошибки – t_u.

Кратность исправляемых ошибок связана с минимальным кодовым расстоянием следующим соотношением:

$$d_0 \geq 2 \cdot t_u + 1. \qquad (3.1)$$

При построении циклических кодов кодовые комбинации принято представлять в полиномиальной форме:

$$F(x) = a_{m-1} \cdot x^{m-1} + a_{m-2} \cdot x^{m-2} + \ldots + a_2 \cdot x^2 + a_1 \cdot x^1 + a_0 \cdot x^0,$$

где m – количество разрядов в кодовой комбинации, a_i – коэффициенты, принимающие значения 0 или 1 в зависимости от значений разрядов кодовой комбинации.

Например, кодовая комбинация 110101 может быть представлена в виде следующего полинома:

$$F(x) = 1 \cdot x^5 + 1 \cdot x^4 + 0 \cdot x^3 + 1 \cdot x^2 + 0 \cdot x^1 + 1 \cdot x^0 = x^5 + x^4 + x^2 + 1.$$

Введем следующие обозначения:

- $G(x)$ – полином степени k-1, соответствующий информационной кодовой комбинации (информационным разрядам);
- $R(x)$ – полином степени r-1, соответствующий проверочной кодовой комбинации (проверочным разрядом);
- $F(x)$ – полином степени n-1, соответствующий кодовой комбинации циклического кода.

Комбинация циклического кода $F(x)$ образуется из исходной комбинации информационных разрядов $G(x)$ путем умножения на полином $P(x)$. Поэтому полином $P(x)$ называется образующим. Разрешенные кодовые комбинации циклического кода характеризуются тем, что все они делятся без остатка на образующий полином $P(x)$. Это условие выполняется, если $R(x)$ равен остатку от деления $x^r \cdot G(x)$ на $P(x)$. В качестве образующих полиномов выступают неприводимые полиномы. Их степень выбирается равной количеству проверочных разрядов.

Способность циклического кода обнаруживать ошибки основана на том, что разрешенные комбинации кода $F(x)$ делятся на образующий полином $P(x)$ без остатка, а запрещенные – с остатком.

Комбинации циклического кода формируются следующим образом:

1) полином $G(x)$ умножается на x^r, т.е. производим сдвиг k-разрядной кодовой комбинации на r разрядов влево;

2) произведение $x^r \cdot G(x)$ делится на образующий полином $P(x)$, а полученный от деления остаток $R(x)$ определяет r проверочных разрядов кодовой комбинации;

3) кодовая комбинация $F(x) = x^r \cdot G(x) + R(x)$ является разрешенной, т.к. она делится без остатка на $P(x)$.

Рассмотрим следующий пример. Пусть необходимо закодировать циклическим кодом информационную комбинацию 1011, таким образом, чтобы обеспечить возможность исправления однократных ошибок, т.е. $t_u = 1$.

Для определения необходимого количества проверочных разрядов воспользуемся следующим выражением:

$$r \geq \log_2\left(\sum_{i=0}^{\frac{d_0-1}{2}} C_n^i\right). \qquad (3.2)$$

Так как известно $t_u = 1$, то, учитывая (3.1), справедливым будет принять $d_0 = 3$. Тогда выражение (3.2) примет следующий вид:

$$r \geq \log_2\left(\sum_{i=0}^{1} C_n^i\right) = \log_2(C_n^0 + C_n^1) = \log_2(1+n) = \log_2(1+k+r).$$

Решить неравенство $r \geq \log_2(k + r + 1)$, зная только количество информационных разрядов $k = 4$, можно методом подбора. В результате получим, что $r \geq 3$. Следовательно, количество проверочных разрядов можно принять равным 3 (т.е. $r = 3$).

Таким образом, кодовая комбинация циклического кода будет иметь длину n = 7 разрядов, из которых информационных разрядов будет $k = 4$, а проверочных – $r = 3$.

Информационная комбинация 1011 в полиномиальной форме будет выглядеть следующим образом:

$$G(x) = x^3 + x + 1.$$

Тогда $x^r \cdot G(x) = x^3 \cdot (x^3 + x + 1) = x^6 + x^4 + x^3$.

В качестве образующего полинома выберем полином $P(x) = x^3 + x^2 + 1$. Тогда остаток от деления $x^r \cdot G(x)$ на $P(x)$ будет равен $R(x) = x^2$.

В результате кодовая комбинация циклического кода в полиномиальной форме будет выглядеть следующим образом:

$$F(x) = x^r \cdot G(x) + R(x) = x^6 + x^4 + x^3 + x^2.$$

В результате передачи кодовой комбинации на выходе канала связи будет получена кодовая комбинация

$$H(x) = F(x) + E(x),$$

где $E(x)$ – вектор ошибки, в котором разряды, соответствующие ошибочным, равны 1, а остальные разряды – 0.

Для обнаружения ошибок на приемном конце следует полином $H(x)$ разделить на образующий полином $P(x)$. Ненулевой остаток, полученный в результате деления, будет свидетельствовать о наличии ошибок в принятой комбинации.

Так как $H(x) = F(x) + E(x)$, а $F(x)$ делится на $P(x)$ без остатка, то остаток $S(x)$, получаемый при делении $H(x)$ на $P(x)$, будет равен остатку от деления $E(x)$ на $P(x)$. Полученный остаток $S(x)$ принято называть «синдромом ошибки». Каждому вектору ошибки $E(x)$ соответствует определенный синдром $S(x)$. Следовательно, зная синдром ошибки $S(x)$, можно определить вектор ошибки $E(x)$ и исправить ее при условии, что кратность ошибки не превышает $t_и$.

Найти все синдромы ошибки достаточно просто. Для этого следует найти остатки от деления векторов всех ошибок, кратность которых не превышает $t_и$, на образующий полином $P(x)$. Указанные остатки и будут являться синдромами соответствующих ошибок.

Рассмотрим следующий пример. Пусть на передаваемую кодовую комбинацию из первого примера $F(x) = x^6 + x^4 + x^3 + x^2$ воздействовала ошибка $E(x) = x^2$. В результате на выходе канала связи будет получена кодовая комбинация $H(x) = x^6 + x^4 + x^3$.

Сначала найдем синдром ошибки $S(x)$ соответствующий вектору ошибки $E(x) = x^2$. Для этого определим остаток от деления $E(x) = x^2$ на $P(x) = x^3 + x^2 + 1$, который будет равен x^2. Следовательно, $S(x) = x^2$.

Далее разделим $H(x)$ на $P(x)$. В результате деления будет получен остаток, равный x^2. Как можно заметить остаток от деления $H(x)$ на образующий полином $P(x)$, равен полученному ранее синдрому $S(x)$, который соответствует вектору ошибки $E(x)$.

Для исправления обнаруженной ошибки и восстановления исходной кодовой комбинации можно воспользоваться следующим выражением:

$$F(x) = H(x) + E(x).$$

3. ЗАДАНИЕ

1. Используя циклический код закодировать в соответствии с вариантом (табл. 3.1) информационную последовательность таким образом, чтобы обеспечить возможность исправления однократных ошибок.

Таблица 3.1

Варианты для выполнения практической работы

Вариант	Информационная последовательность	Ошибочный разряд	Ошибочные разряды
1	00001110010	4	14; 0
2	00010110001	5	14; 1
3	00011110000	6	14; 2
4	00100101111	7	14; 3
5	00101101110	8	14; 4
6	00110101101	9	14; 5
7	00111101100	10	14; 6
8	01000101011	11	14; 7
9	01001101010	12	14; 8
10	01010101001	13	14; 9
11	01011101000	14	14; 10
12	01100100111	0	14; 11
13	01101100110	1	14; 12
14	01110100101	2	14; 13
15	01111100100	3	13; 0
16	10000100011	4	13; 1
17	10001100010	5	13; 2
18	10010100001	6	13; 3
19	10011100000	7	13; 4
20	10100011111	8	13; 5
21	10101011110	9	13; 6
22	10110011101	10	13; 7
23	10111011100	11	13; 8
24	11000011011	12	13; 9
25	11001011010	13	13; 10

2. Определить все возможные синдромы ошибок для данного кода.

3. В соответствии с вариантом изменить один разряд в полученной кодовой комбинации $F(x)$.

4. Определить ошибочный разряд, сравнив остаток от деления измененной кодовой комбинации $H_1(x)$ на образующий полином с синдромами ошибки.

5. В соответствии с вариантом изменить два разряда в полученной кодовой комбинации $F(x)$.

6. Определить ошибочный разряд, сравнив остаток от деления измененной кодовой комбинации $H_2(x)$ на образующий полином с синдромами ошибки.

7. Сделать выводы на основании полученных результатов.

В табл. 3.2 приведены неприводимые полиномы, которые могут быть использованы при выполнении данной работы.

Таблица 3.2

Неприводимые полиномы

Степень полинома	Полином
1	$x+1$
2	x^2+x+1
3	x^3+x+1
	x^3+x^2+1
4	x^4+x+1
	x^4+x^3+1
	x^4+x^2+x+1
	$x^4+x^3+x^2+1$
5	x^5+x^2+1
	x^5+x^3+1

Отчет о данной работе должен содержать:

- номер и название работы;
- задание в соответствии с вариантом;

- полиномиальное представление информационной последовательности;

- расчет необходимого количества проверочных разрядов;

- выбранный образующий полином;

- таблицу, содержащую все возможные синдромы ошибки;

- кодовую комбинацию $F(x)$ после кодирования циклическим кодом;

- кодовую комбинацию $H_1(x)$, представляющую собой исходную комбинацию $F(x)$ с одним измененным разрядом;

- остаток от деления $H_1(x)$ на образующий полином $P(x)$;

- результат сравнения остатка от деления $H_1(x)$ на образующий полином $P(x)$;

- кодовую комбинацию $H_2(x)$, представляющую собой исходную комбинацию $F(x)$ с одним измененным разрядом;

- остаток от деления $H_2(x)$ на образующий полином $P(x)$;

- результат сравнения остатка от деления $H_2(x)$ на образующий полином $P(x)$;

- выводы по полученным результатам.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое кодирование?
2) Что такое код?
3) Какие параметры кодов существуют?
4) Что такое кодовое расстояние?
5) Что такое минимальное кодовое расстояние?
6) Что такое расстояние Хэмминга?
7) Что такое избыточность кода?
8) Какие кодовые комбинации называются разрешенными?
9) Какие кодовые комбинации называются запрещенными?

10) Какие коды называются простыми?

11) Теорема Шеннона о кодировании для канала с помехами.

12) Какой код называют помехоустойчивым?

13) Какие коды называются разделимыми?

14) Какие коды называются неразделимыми?

15) Какие коды называются систематическими?

16) Какие коды называются циклическими?

17) Что такое кратность ошибки?

18) Как соотносится кратность обнаруживаемой ошибки с минимальным кодовым расстоянием циклического кода?

19) Как соотносится кратность исправляемой ошибки с минимальным кодовым расстоянием циклического кода?

20) Что такое образующий полином в циклическом коде?

21) Какие полиномы выступают в качестве образующих в циклических кодах?

22) Принцип формирования циклического кода.

23) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности обнаруживаемой ошибки?

24) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности исправляемой ошибки?

25) Что такое вектор ошибки?

26) Что такое синдром ошибки?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

Практическая работа №4

РАСЧЕТ ОПТИМАЛЬНЫХ ПАРАМЕТРОВ ПОМЕХОУСТОЙЧИВОГО КОДА

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными принципами подбора параметров помехоустойчивых кодов;
- получения базовых практических навыков расчета оптимальных параметров циклического кода.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Как известно, при использовании помехоустойчивых кодов каждая кодовая комбинация значности n включает в себя k информационных и r проверочных разрядов.

При этом значение k выбирается таким образом, чтобы обеспечить возможность передачи всех символов алфавита сообщения, т.е. если нужно закодировать двоичным кодом сообщения, алфавит которых включает в себя A символа, то значение k можно найти из соотношения

$$k \geq \log_2 A. \tag{4.1}$$

Необходимое количество проверочных разрядов r зависит от требуемой исправляющей способности используемого помехоустойчивого кода. Для циклического кода в целях определения значения r можно воспользоваться следующим выражением:

$$r \geq \log_2\left(\sum_{i=0}^{\frac{d_0-1}{2}} C_n^i\right), \tag{4.2}$$

где d_0 – расстояние Хэмминга.

Количество изменений информационного параметра передаваемого сигнала в 1 секунду называется скоростью модуляции или символьной скоростью. Скорость модуляции измеряется в бодах (бод; baud).

Если в системе связи используется двоичный сигнал, и каждый единичный элемент сигнала несет не более одного бита информации, то между скоростью передачи информации C и скоростью модуляции B при использовании циклического кода справедливым будет следующее соотношение:

$$C = \frac{k}{n}B = \frac{k}{k+r}B. \tag{4.3}$$

Можно заметить, что скорость передачи информации в системе связи будет тем выше, чем меньше будет значение r.

Как известно, соотношение расстояния Хэмминга и исправляющей способности кода можно описать неравенством $d_0 \geq 2t_u + 1$. В тоже время расстояние Хэмминга связано со способностью помехоустойчивого кода обнаруживать ошибки неравенством $d_0 \geq t_o + 1$. Следовательно, помехоустойчивый код способен обнаружить ошибки большей кратности, чем он способен исправить. Однако помехоустойчивый код при обнаружении ошибки не позволяет определить ее кратность. Вследствие этого в системах связи, как правило, реализуются два варианта действий при обнаружении ошибок:

- приемная сторона пытается исправить обнаруженную ошибку;
- приемная сторона посылает запрос отправителю на повторную отправку комбинации, в которой была обнаружена ошибка (система с обратной связью).

Если предположить, что в кодовой комбинации произошла ошибка, кратность которой меньше t_o, но больше t_u, то она будет обнаружена. Но попытка исправить эту ошибку не увенчается успехом. Однако приемная сторона будет считать, что ошибка исправлена.

При втором же варианте любая ошибка будет вызывать повторный запрос передачи кодовой комбинации, что будет снижать скорость передачи информации в системе. Кроме того, отправитель должен определенное время хранить у себя M ранее отправленных кодовых комбинаций на случай повторной отправки, и в случае обнаружения ошибки на приемной стороне, повторно будет отправлена группа из M кодовых комбинаций.

Предположим, что количество ошибок, возникающих при передаче по каналу в кодовой комбинации длинной n, распределено по биномиальному закону, т.е.

$$P_{ош}(l) = C_n^l \cdot p^l \cdot (1-p)^{n-l}, \tag{4.4}$$

где l – количество ошибок, p – вероятность неправильного приема одного бита.

Тогда вероятность безошибочного приема кодовой комбинации будет равна

$$P_{бо} = P_{ош}(l=0) = (1-p)^n. \tag{4.5}$$

Вероятность необнаруженной ошибки в кадре данных определяется по приближенной формуле:

$$P_{но} \approx \frac{1}{2^r} \cdot \sum_{i=d_0}^{n} C_n^i \cdot p^i \cdot (1-p)^{n-i}. \qquad (4.6)$$

Тогда, учитывая (4.5) и (4.6), вероятность обнаружения ошибки можно найти как

$$P_{оо} = 1 - P_{но} - P_{бо}. \qquad (4.7)$$

Повторная передача группы из M кодовых комбинаций будет происходить только в том случае, если в такой группе произойдет как минимум одна ошибка в кодовой комбинации длинной n, которую декодер сможет обнаружить. Следовательно, вероятность повторной передачи можно найти как

$$P_{пот} = 1 - (1 - P_{оо})^M, \qquad (4.8)$$

т.е. как вероятность того, что хотя бы в одной из M кодовых комбинаций длинной n произойдет ошибка, которая может быть обнаружена.

Тогда вероятность того, что одна группа из M кодовых комбинаций будет передана i раз, можно найти следующим образом:

$$P_{пот}(i) = P_{пот}^{i-1} \cdot (1 - P_{пот}), \qquad (4.9)$$

т.е. как вероятность того, что одна группа из M кодовых комбинаций будет i-1 раз передана с обнаруживаемыми ошибками, и один раз ошибка либо не произойдет, либо не будет обнаружена.

Обычно количество повторных передач ограничивается определенным максимальным количеством N_{max}, при достижении которого группа из M кодовых комбинации больше не передается.

Среднее количество передач одной группы из M кодовых комбинаций можно найти как

$$N_{ср} = \sum_{i=1}^{N_{max}} i \cdot P_{повт}(i). \tag{4.10}$$

Каждая кодовая комбинация длиной n содержит в себе k бит информации. Тогда в группе из M кодовых комбинаций, дина которой равна $M{\cdot}n$, будет содержаться $M{\cdot}k$ бит информации. Другими словами в $M{\cdot}n$ кодовых разрядах содержится $M{\cdot}k$ бит информации. Однако каждая группа из M кодовых комбинаций передается в среднем $N_{ср}$ раз. Получается, что для передачи $M{\cdot}k$ бит информации приходится передавать $N_{ср}{\cdot}M{\cdot}n$ разрядов (или единичных элементов). Следовательно, время, затрачиваемое на передачу $M{\cdot}k$ бит информации, можно найти следующим образом:

$$T = \frac{N_{ср} \cdot M \cdot n}{B}. \tag{4.11}$$

В тоже время скорость передачи $M{\cdot}k$ бит информации рассчитывается как

$$C = \frac{M \cdot k}{T}. \tag{4.12}$$

Подставив (4.11) в (4.12), получим

$$C=\frac{M\cdot k\cdot B}{N_{cp}\cdot M\cdot n}=\frac{k\cdot B}{N_{cp}\cdot n}. \qquad (4.13)$$

Тогда относительную пропускную способность R можно выразить следующим образом:

$$R=\frac{C}{B}=\frac{k}{N_{cp}\cdot n}=\frac{k}{N_{cp}\cdot (k+r)}. \qquad (4.14)$$

Следует заметить, что N_{cp} зависит от размера группы кодовых комбинаций M. Количество кодовых комбинаций длиной n в группе выбирается исходя из следующего соотношения:

$$M=\left\lfloor 3+2\cdot\frac{T_p}{T_\kappa}\right\rfloor, \qquad (4.15)$$

где T_p – время распространения сигнала по линии связи, T_κ – длительность одной кодовой комбинации длиной n, а оператор $\lfloor\ \rfloor$ – округление к ближайшему большему целому.

Время распространения сигнала t_p можно найти следующим образом:

$$T_p=\frac{L}{v}, \qquad (4.16)$$

где L – длина линии связи, v – скорость распространения сигнала по линии связи.

Длительность одной кодовой комбинации $t_к$ определяется как

$$T_к = \frac{n}{B}. \qquad (4.17)$$

Тогда, подставив (4.16) и (4.17) в (4.15), получим следующее:

$$M = \left\lfloor 3 + 2 \cdot \frac{L \cdot B}{v \cdot n} \right\rfloor. \qquad (4.18)$$

Под оптимальными параметрами будем понимать такие параметры помехоустойчивого кода, при которых значение относительной пропускной способности R наиболее близко к 1. При этом значения параметров L, v, B, p, t_o и N_{max} заранее известны.

3. ЗАДАНИЕ

В соответствии с вариантом (табл. 4.1) найти оптимальные значения параметров n, k и r циклического кода при заданных значениях L, v, B, p, t_o и N_{max}. Для упрощения считать, что $d_0 = t_o + 1$.

Таблица 4.1

Варианты для выполнения практической работы

Вариант	L, км	v, $\cdot 10^8$ км/с	B, бод	p, $\cdot 10^{-5}$	t_o	N_{max}
1	50	2,999976	10240	25	2	3
2	52	2,999977	11264	24	3	4
3	54	2,999978	12288	23	4	5
4	56	2,999979	13312	22	2	6
5	58	2,999980	14336	21	3	7
6	60	2,999981	15360	20	4	8
7	62	2,999982	16384	19	2	3
8	64	2,999983	17408	18	3	4

Продолжение табл. 4.1

Варианты для выполнения практической работы

Вариант	L, км	v, $\cdot 10^8$ км/с	B, бод	p, $\cdot 10^{-9}$	t_o	N_{max}
9	66	2,999984	18432	17	4	5
10	68	2,999985	19456	16	2	6
11	70	2,999986	20480	15	3	7
12	72	2,999987	21504	14	4	8
13	74	2,999988	22528	13	2	3
14	76	2,999989	23552	12	3	4
15	78	2,999990	24576	11	4	5
16	80	2,999991	25600	10	2	6
17	82	2,999992	26624	9	3	7
18	84	2,999993	27648	8	4	8
19	86	2,999994	28672	7	2	3
20	88	2,999995	29696	6	3	4
21	90	2,999996	30720	5	4	5
22	92	2,999997	31744	4	2	6
23	94	2,999998	32768	3	3	7
24	96	2,999999	33792	2	4	8
25	98	3,000000	34816	1	2	3

Результаты расчетов представить в табличной форме:

n	k	r	R
n_1	k_1	r_1	R_1
n_2	k_2	r_2	R_2
...	...	...	...

Отчет о данной работе должен содержать:

- номер и название работы;
- задание в соответствии с вариантом;
- расчеты;
- таблицу с полученными результатами;
- выбранные оптимальные значения параметров кода;
- выводы по полученным результатам.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое кодирование?

2) Что такое код?

3) Какие параметры кодов существуют?

4) Что такое кодовое расстояние?

5) Что такое минимальное кодовое расстояние?

6) Что такое расстояние Хэмминга?

7) Что такое избыточность кода?

8) Теорема Шеннона о кодировании для канала с помехами.

9) Что такое кратность ошибки?

10) Как соотносится кратность обнаруживаемой ошибки с минимальным кодовым расстоянием циклического кода?

11) Как соотносится кратность исправляемой ошибки с минимальным кодовым расстоянием циклического кода?

12) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности обнаруживаемой ошибки?

13) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности исправляемой ошибки?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

Практическая работа №5

ПОСТРОЕНИЕ УСТРОЙСТВ ПОМЕХОУСТОЙЧИВОГО КОДИРОВАНИЯ

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными принципами построения устройств помехоустойчивого кодирования;
- получения базовых практических навыков построения кодирующих устройств циклического кода.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Построение кодера циклического кода, выполняющего кодирование в соответствии с алгоритмом, описанным в практической работе №4, основывается на следующих принципах:

- кодер строится в соответствии с видом образующего полинома $P(x)$ и представляет собой регистр сдвига с обратными логическими связями через сумматоры по модулю 2;
- число разрядов (ячеек памяти) в регистре равно степени образующего полинома;
- число сумматоров по модулю 2 на единицу меньше веса образующего полинома $P(x)$;
- сумматоры по модулю 2 ставятся перед ячейками памяти, соответствующим ненулевым членам образующего полинома $P(x)$, исключая его старшую степень.

Предположим, что необходимо построить структурную схему кодера циклического кода (7, 4) на основе образующего полинома $P(x) = x^3 + x^2 + 1$.

Тогда количество разрядов в регистре сдвига будет равно 3, а количество сумматоров по модулю 2 будет равно 2, и они будут располагаться перед нулевой и второй ячейками. Структурная схема данного кодера приведена на рис. 5.1.

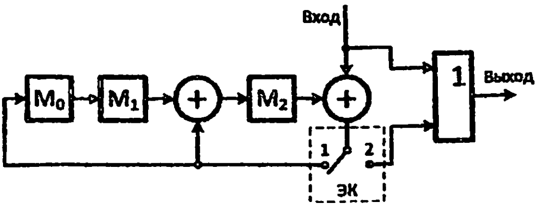

Рис. 5.1. Структурная схема кодера циклического кода

В табл. 7.1 приведен пример работы рассмотренного кодера (рис. 5.1) при поступлении на его вход информационной последовательности 1011.

Таблица 5.1

Пример работы кодера

Номер такта	Вход	Положение ЭК	Ячейки регистра сдвига			Выход
			M_0	M_1	M_2	
0	–	1	0	0	0	–
1	1	1	1	0	1	1
2	0	1	1	1	1	0
3	1	1	0	1	1	1
4	1	1	0	0	1	1
5	–	2	–	0	0	1
6	–	2	–	–	0	0
7	–	2	–	–	–	0

В исходном состоянии электронный ключ ЭК находится в положении 1, т.е. обратная связь замкнута. На вход последовательно, начиная со старшего разряда, подаются символы информационной последовательности $G(x)$, одновременно поступающие через схему ИЛИ на выход кодера.

Через k тактов в регистре сдвига вырабатывается остаток $R(x)$ от деления произведения $G(x)\cdot x^r$ на образующий полином $P(x)$. Затем на такте (k+1) ЭК переводится в положение 2. Обратная связь при этом разрывается и на выход кодера из регистра в течение r тактов поступают проверочные разряды. Через n тактов с начала работы кодера на выходе получается кодовая комбинация, закодированная циклическим кодом.

Таким образом, на выходе кодера будет получена последовательность 1011100.

3. ЗАДАНИЕ

1. В соответствии с вариантом (табл. 5.2) разработать схему кодера циклического кода.

2. Рассчитать проверочные разряды для заданной в соответствии с вариантом информационной последовательности.

3. Составить описание работы кодера в виде таблицы, как показано выше в примере (табл. 5.1).

Таблица 5.2

Варианты для выполнения практической работы

Вариант	Информационная последовательность	Образующий полином
1	0 0 0 0 1 1 1 0 0 1 0	x^4+x+1
2	0 0 0 1 0 1 1 0 0 0 1	x^4+x^3+1
3	0 0 0 1 1 1 1 0 0 0 0	x^4+x^2+x+1
4	0 0 1 0 0 1 0 1 1 1 1	$x^4+x^3+x^2+1$
5	0 0 1 0 1 1 0 1 1 1 0	x^4+x+1
6	0 0 1 1 0 1 0 1 1 0 1	x^4+x^3+1
7	0 0 1 1 1 1 0 1 1 0 0	x^4+x^2+x+1
8	0 1 0 0 0 1 0 1 0 1 1	$x^4+x^3+x^2+1$
9	0 1 0 0 1 1 0 1 0 1 0	x^4+x+1

Продолжение табл. 5.2

Вариант	Информационная последовательность	Образующий полином
10	01010101001	x^4+x^3+1
11	01011101000	x^4+x^2+x+1
12	01100100111	$x^4+x^3+x^2+1$
13	01101100110	x^4+x+1
14	01110100101	x^4+x^3+1
15	01111100100	x^4+x^2+x+1
16	10000100011	$x^4+x^3+x^2+1$
17	10001100010	x^4+x+1
18	10010100001	x^4+x^3+1
19	10011100000	x^4+x^2+x+1
20	10100011111	$x^4+x^3+x^2+1$
21	10101011110	x^4+x+1
22	10110011101	x^4+x^3+1
23	10111011100	x^4+x^2+x+1
24	11000011011	$x^4+x^3+x^2+1$
25	11001011010	x^4+x+1

Отчет о данной работе должен содержать:

- номер и название работы;
- задание в соответствии с вариантом;
- разработанную схему кодера;
- результаты расчета проверочных разрядов для заданной информационной последовательности;
- описание работы кодера в виде таблицы.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое кодирование?
2) Что такое код?
3) Какие параметры кодов существуют?
4) Что такое кодовое расстояние?

5) Что такое минимальное кодовое расстояние?

6) Что такое расстояние Хэмминга?

7) Что такое избыточность кода?

8) Теорема Шеннона о кодировании для канала с помехами.

9) Что такое кратность ошибки?

10) Как соотносится кратность обнаруживаемой ошибки с минимальным кодовым расстоянием циклического кода?

11) Как соотносится кратность исправляемой ошибки с минимальным кодовым расстоянием циклического кода?

12) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности обнаруживаемой ошибки?

13) Как определить необходимое количество проверочных разрядов циклического кода при заданной кратности исправляемой ошибки?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

Практическая работа №6

ИЗУЧЕНИЕ АЛГОРИТМОВ СЖАТИЯ ИНФОРМАЦИИ

1. ЦЕЛЬ РАБОТЫ

Данная практическая работа предназначена для:

- ознакомления с основными принципами сжатия информации;
- получения базовых практических навыков применения словарных алгоритмов сжатия информации.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. Общие сведения

Сжатие информации – алгоритмическое преобразование сообщений, производимое с целью уменьшения занимаемого ими объёма. Применяется для более рационального использования устройств хранения и передачи данных. Обратная процедура называется восстановлением информации.

Сжатие основано на устранении избыточности, содержащейся в исходном сообщении. Подобная избыточность обычно устраняется заменой повторяющейся последовательности ссылкой на уже закодированный фрагмент с указанием его длины. Другой вид избыточности связан с тем, что некоторые значения в сжимаемых сообщениях встречаются чаще других. Сокращение объёма сообщения достигается за счёт замены часто встречающихся символов короткими кодовыми словами, а редких – длинными. Сжатие сообщений, не обладающих свойством избыточности, невозможно осуществить без потерь информации.

Все методы сжатия данных делятся на два основных класса:

- сжатие без потерь;
- сжатие с потерями.

Сжатие без потерь – метод сжатия, при использовании которого сообщения могут быть восстановлены с точностью до бита. При этом оригинальные сообщения полностью восстанавливаются из сжатого состояния. Для каждого из типов информации, как правило, существуют свои оптимальные алгоритмы сжатия без потерь.

Сжатие с потерями – метод сжатия, при использовании которого восстановленные сообщения отличаются от исходных сообщений, но степень их отличия не является существенной с точки зрения их дальнейшего использования.

Коэффициент сжатия – это основная характеристика алгоритма сжатия. Она определяется как отношение объёма исходных несжатых данных к объёму сжатых, т.е.

$$K = \frac{S_u}{S_c},$$

где S_u – объём исходных данных, а S_c – объём сжатых.

Если $K = 1$, то алгоритм не производит сжатия, т.е. выходное сообщение оказывается по объёму равным исходному. Если же $K < 1$, то алгоритм порождает сообщение большего размера, нежели исходное.

Ситуация с $K < 1$ вполне возможна при сжатии, т.к. невозможно получить алгоритм сжатия без потерь, который для любых исходных сообщений образовывал бы на выходе сообщения меньшей или равной длины. Обоснование этого факта заключается в том, что поскольку число различных сообщений длиной n бит составляет 2^n, число различных сообщений с длиной меньшей или равной n бит будет меньше 2^n. Это значит, что невозможно однозначно сопоставить все исходные сообщения сжатым. Однако даже когда алгоритм сжатия увеличивает размер исходных данных, легко добиться того, чтобы их объём гарантировано не мог увеличиться более чем на 1 бит. Делается это следующим образом: если объём сжатых данных меньше объёма исходных, возвращаем сжатые данные, добавив к ним «1», иначе возвращаем исходные данные, добавив к ним «0»).

2.2. Словарные методы сжатия

Словарные методы сжатия основываются на том, что входную последовательность символов можно рассматривать как последовательность строк, содержащих произвольное количество символов. Эти строки символов можно заменить кодами, которые будут соответствовать индексу строк в некотором словаре. Строки, образующие словарь, будем называть фразами. При восстановлении производится замена индекса на соответствующую ему фразу из словаря.

В данном случае словарь будет представлять собой набор таких фраз, которые предположительно будут встречаться в обрабатываемом сообщении. Индекс фраз словаря должен быть построен таким образом, чтобы в среднем количество содержащихся в нем символов было меньше, чем количество символов, содержащееся во фразах словаря. Именно это позволяет осуществить сжатие исходного сообщения.

2.3. Алгоритмы сжатия Лемпеля-Зива

Классические алгоритмы Лемпеля-Зива представляют собой универсальные алгоритмы словарного сжатия, в которых словарь формируется на основании уже обработанной части входного потока символов исходного сообщения, т.е. данные алгоритмы являются адаптивными. Отличия данных алгоритмов заключаются лишь в способе формирования фраз словаря.

2.3.1. Алгоритм LZ77

Алгоритм LZ77 является первым алгоритмом со скользящим словарем (скользящим окном). В данном алгоритме в качестве словаря используется блок уже закодированной последовательности. По мере выполнения обработки положение этого блока относительно начала последовательности изменяется.

Скользящее окно имеет длину N символов и состоит из двух частей:

- последовательность длины $W = N\text{-}n$ уже закодированных символов, которая является словарем;
- упреждающий буфер длины n.

Пусть к определенному моменту уже закодировано t символов исходного сообщения $s_1, s_2, \ldots, s_t$. Тогда словарем будет W предшествующих символов $s_{t-(W-1)}, s_{t-(W-1)+1}, s_{t-(W-1)+2}, \ldots, s_t$. При этом в упреждающем буфере находится n символов $s_{t+1}, s_{t+2}, \ldots, s_{t+n}$.

Сущность алгоритма заключается в поиске самого длинного совпадения между строкой буфера, начинающейся с символа s_{t+1}, и всеми фразами словаря. Эти фразы могут начинаться с любого символа $s_{t-(W-1)}, s_{t-(W-1)+1}, s_{t-(W-1)+2}, \ldots, s_t$, но не должны выходить за пределы скользящего окна. Кроме того длина совпадения не должна превышать размер буфера. Полученная в результате поиска фраза $s_{t-(i-1)}, s_{t-(i-1)+1}, \ldots, s_{t-(i-1)+(j-1)}$ кодируется с помощью двух чисел:

- смещения от начала буфера – i;
- длины совпадения – j.

Смещение и длина совпадения выполняют функцию указателя, однозначно определяющего фразу словаря. Также в выходной поток записывается один символ, непосредственно следующий за совпавшей строкой буфера.

Таким образом, на каждом этапе кодер выдает значения трех элементов: смещение i, длина совпадения j и один символ. После этого окно смещается на j+1 символов вправо и начинается следующий этап сжатия.

Пример работы алгоритма LZ77 для сжатия исходного сообщения «ABAADDDDDBAACCCCEAFFFFDAAAA» представлен в табл. 6.1.

В исходном сообщении содержится 27 символов алфавита, включающего 26 символов. Для кодирования каждого символа s_k исходного сообщения достаточно 5 бит. Следовательно, для кодирования данного сообщения равномерным двоичным кодом без сжатия потребуется не менее 27·5 = 135 бит.

Так как количество символов в сообщении равно 27, то максимальное смещение относительно начала буфера будет равно i_{max} = 25 символам. Размер упреждающего буфера равен 7 символам, поэтому максимальная длина совпадения будет равна j_{max} = 7. Следовательно, для кодирования i будет достаточно 5 бит, а для кодирования j – 3 бит. Так как вместо символов исходного сообщения будут передаваться тройки (i; j; s), то после сжатия сообщение будет занимать 14·(5+3+5) = 182 бит.

Таблица 6.1

Пример работы алгоритма LZ77

Такт	Скользящее окно		Совпадающая фраза	Код		
	Словарь	Буфер (7)		i	j	s
1	–	ABAADDD	–	1	0	A
2	A	BAADDDD	–	1	0	B
3	AB	AADDDDD	A	2	1	A
4	ABAA	DDDDDBA	–	1	0	D
5	ABAAD	DDDDBAA	D	1	1	D
6	ABAADDD	DDBAACC	DD	2	2	B
7	ABAADDDDDB	AACCCCE	AA	8	2	C
8	ABAADDDDDBAAC	CCCEAFF	C	1	1	C
9	ABAADDDDDBAACCC	CEAFFFF	C	1	1	E
10	ABAADDDDDBAACCCCE	AFFFFDA	A	6	1	F
11	ABAADDDDDBAACCCCEAF	FFFDAAA	F	1	1	F
12	ABAADDDDDBAACCCCEAFFF	FDAAAA	F	1	1	D
13	ABAADDDDDBAACCCCEAFFFFD	AAAA	AA	13	2	A
14	ABAADDDDDBAACCCCEAFFFFDAAA	A	–	1	0	A

В данном случае коэффициент сжатия

$$K_{LZ77} = \frac{S_u}{S_c} = \frac{135}{195} = 0{,}742.$$

Следовательно, в данном случае сжатие было выполнено неэффективно. Данный пример показывает, что алгоритм LZ77 неэффективен для относительно коротких сообщений.

Восстановление сжатого сообщения осуществляется путем простой замены кода на блок символов, состоящий из фразы словаря и явно передаваемого символа. Фраза словаря определяется по смещению i и длине совпадения j.

2.3.2. Алгоритм LZSS

Алгоритм LZSS позволяет сочетать в выходной последовательности символы и указатели, что в некоторых случаях позволяет повысить эффективность сжатия.

Принцип работы этого алгоритма заключается в добавлении к каждому указателю и символу одного бита f, который позволяет отличить указатель от символа. Символ подается на выход в явном виде, если текущая длина максимального совпадения буфера и какой-либо фразы словаря меньше или равна 1. В данном случае значение бита $f = 0$. Во всех остальных случаях на выход подается указатель, а значение бита $f = 1$.

Пример работы алгоритма LZSS для сжатия исходного сообщения «ABAADDDDDBAACCCCEAFFFFDAAAA» представлен в табл. 6.2.

Как было указано выше для кодирования исходного сообщения равномерным двоичным кодом без сжатия потребуется 135 бит, для кодирования i будет достаточно 5 бит, а для кодирования j – 3 бит. Тогда количество бит, полученных в результате сжатия сообщения можно найти как $5 \cdot (1+5+3+5) + 11 \cdot (1+5) = 136$ бит.

Таблица 6.2

Пример работы алгоритма LZSS

Такт	Скользящее окно		Совпадающая фраза	Код			
	Словарь	Буфер (7)		*f*	*i*	*j*	*s*
1	–	ABAADDD	–	0	–	–	A
2	A	BAADDDD	–	0	–	–	B
3	AB	AADDDDD	A	0	–	–	A
4	ABA	ADDDDDB	A	0	–	–	A
5	ABAA	DDDDDBA	–	0	–	–	B
6	ABAAD	DDDDBAA	D	0	–	–	D
7	ABAADD	DDDBAAC	DD	1	2	2	D
8	ABAADDDDD	BAACCCC	BAA	1	8	3	C
9	ABAADDDDDBAAC	CCCEAFF	C	0	–	–	C
10	ABAADDDDDBAACC	CCEAFFF	CC	1	2	2	E
11	ABAADDDDDBAACCCCE	AFFFFDA	A	0	–	–	A
12	ABAADDDDDBAACCCCEA	FFFFDAA	–	0	–	–	F
13	ABAADDDDDBAACCCCEAF	FFFDAAA	F	0	–	–	F
14	ABAADDDDDBAACCCCEAFF	FFDAAAA	FF	1	2	2	D
15	ABAADDDDDBAACCCCEAFFFFD	AAAA	AA	1	13	2	A
16	ABAADDDDDBAACCCCEAFFFFDAAA	A	A	0	–	–	A

В данном случае коэффициент сжатия

$$K_{LZSS} = \frac{S_u}{S_c} = \frac{135}{136} = 0{,}993.$$

Следовательно, в данном случае сжатие также было выполнено неэффективно. Однако в данном случае алгоритм LZSS оказался эффективнее, чем алгоритм LZ77.

2.3.3. Алгоритм LZ78

Алгоритм LZ78 не использует скользящего окна и в словарь заносятся не все встречаемые при кодировании строки, а лишь те, которые имеют большую вероятность последующего использования. На каждом этапе в словарь записывается новая фраза, которая представляет собой конкатенацию фразы *S* словаря, имеющей самое длинное совпадение со строкой буфера, и символа *s*. Символ *s* является символом, следующим за строкой буфера, для которой найдена совпадающая фраза *S*.

Каждая кодовая последовательность после сжатия будет состоять из индекса *n* «родительской» фразы *S* и символа *s*.

Пример работы алгоритма LZ78 для сжатия исходного сообщения «ABAADDDDDBAACCCCCEAFFFFFDAAAA» представлен в табл. 6.3.

Таблица 6.3

Пример работы алгоритма LZ78

Такт	Буфер (6)	Код		Совпадающая фраза	Словарь	
		n	s		индекс	фраза
0	–	–	–	–	1	–
1	ABAADD	1	A	–	2	A
2	BAADDD	1	B	–	3	B
3	AADDDD	2	A	A	4	AA
4	DDDDDB	1	D	–	5	D
5	DDDDBA	5	D	D	6	DD
6	DDBAAC	6	B	DD	7	DDB
7	AACCCC	4	C	AA	8	AAC
8	CCCEAF	1	C	–	9	C
9	CCEAFF	9	C	C	10	CC
10	EAFFFF	1	E	–	11	E
11	AFFFFD	2	F	A	12	AF
12	FFFDAA	1	F	–	13	F
13	FFDAAA	13	F	F	14	FF
14	DAAAA	5	A	–	15	DA
15	AAA	4	A	AA	16	AAA

Для кодирования исходного сообщения равномерным двоичным кодом без сжатия потребуется 135 бит.

Так как количество символов в сообщении равно 27, то максимальное значение индекса словаря может быть равно $n_{max} = 28$ (с учетом пустой строки). Следовательно, для кодирования индекса n достаточно будет 5 бит. Так как вместо символов исходного сообщения будут передаваться пары (n; s), то после сжатия сообщение будет занимать 15·(5+5) = 150 бит.

В данном случае коэффициент сжатия

$$K_{LZ78} = \frac{S_u}{S_c} = \frac{135}{150} = 0{,}9.$$

Следовательно, в данном случае сжатие было выполнено неэффективно, однако алгоритм LZ78 оказался эффективнее LZ77, т.к. $K_{LZ78} > K_{LZ77}$. Но при этом для данного примера алгоритм LZ78 проигрывает в эффективности сжатия алгоритму LZSS.

3. ЗАДАНИЕ

1. В соответствии с вариантом (табл. 6.4) произвести сжатие исходной информационной последовательности с помощью алгоритма LZ77.

2. Определить коэффициент сжатия с помощью алгоритма LZ77.

3. В соответствии с вариантом (табл. 6.4) произвести сжатие исходной информационной последовательности с помощью алгоритма LZSS.

4. Определить коэффициент сжатия с помощью алгоритма LZSS.

5. В соответствии с вариантом (табл. 6.4) произвести сжатие исходной информационной последовательности с помощью алгоритма LZ78.

6. Определить коэффициент сжатия с помощью алгоритма LZ78.

Таблица 6.4

Варианты для выполнения практической работы №8

Вариант	Информационная последовательность
1	BBCCCCDEEEGGGGFFBAFFFEFFFFGGCCDEGGFFFFDD
2	DFABBBEEFAEBCEDDABCBBBEFFFAEBBCBBBABBBBC
3	CCCCCBBBCCCCCCCBBBBBAAAADDDDEEBBAAEFDEEF
4	BCDEEFAACCAAAEEEDEEDDBBCCDEDCCADBAAEEEDD
5	AAABBCCCAAAAAABBAAAAABBAFFFEBCCCCCDDDDEEE
6	CEDDBCEFABBBFAABBADDFAEBAADDEGGFDEEAAEFA
7	BBCBBCCAADBBDEEFDDDEGABBDABBBBACACCEGEDE
8	DDFEBBCAADAEFFBBCCBDDCEDDAEGHAABFBBAAFBD
9	BBCBBZZBCZDDEDAAYYYFADDFZXFGGHXCXXZZDHEE
10	DDECZZZPOODGHHEKKLBFFEELLLLBBEEEAAAAABBB
11	CAADSSAASSSSSAEVVVBGBUUUEUFBEUUUDEFHGEEA
12	FFFBBCOOOOXODEEAWETWAFDQQQFFAQQAABFFZXXF
13	CEEFKLLLFAEDAALLKKKEDAARRRTECBAATTTDOODD
14	EZZZXXXZXZEEKLMRRREBRRRBBERRREAASETRRROP
15	GBGBGBGZZDBAFZZZFFRETEDSGHRTBBBFFFDEEDER
16	BBSABAASABEDDAAAEDAAGFAAAZXCWJERZXESSSQW
17	FFEAAZAAZSADZZZSADFDFFGTOOTTYYYTTFAAGFAA
18	GFFFFBBSDSDSDBBAAAACCFFFAWETWDSRRRFCCDDE
19	FFFFFFECCDEEEGGGFGGCCDEGGAGFFBFFFFDDBBCC
20	FABBCEHABCDEDAAYYYFAAEGAAAFAAADFADDFADDD
21	AAADAAABBCCZSADZZDDEEFFGGHHZSADZZHHAADAA
22	ABBBCEBBFAABBDDEFFFAEBBCCBBFAABBCCADABBB
23	DDDDBAFZZDDZDAACCBABBBCCCDDDEDBAFZZEEFFF
24	AFFFEFFFFGGCCDEGGFFFFDDBBCCCCDEEEGGGGFFB
25	BCBBBEFFFAEBBCBBBABBBBCDFABBBEEFAEBCEDDA

7. Сделать выводы по полученным результатам.

Отчет о данной практической работе должен содержать:

- **номер и название практической работы;**
- **задание в соответствии с вариантом;**

- таблицу, описывающую процесс сжатия исходной последовательности алгоритмом LZ77;

- коэффициент сжатия с помощью алгоритма LZ77;

- таблицу, описывающую процесс сжатия исходной последовательности алгоритмом LZSS;

- коэффициент сжатия с помощью алгоритма LZSS;

- таблицу, описывающую процесс сжатия исходной последовательности алгоритмом LZ78;

- коэффициент сжатия с помощью алгоритма LZ78;

- выводы по полученным результатам.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1) Что такое сжатие информации?

2) На каком свойстве сообщений основаны все алгоритмы сжатия?

3) Какие классы алгоритмов сжатия существуют?

4) Какие алгоритмы сжатия называют алгоритмами сжатия без потерь?

5) Какие алгоритмы сжатия называют алгоритмами сжатия с потерями?

6) Что такое коэффициент сжатия?

7) В чем заключается основной принцип работы словарных методов сжатия?

7) В чем заключается основной принцип работы алгоритма LZ77?

8) В чем заключается основной принцип работы алгоритма LZSS?

9) В чем заключается основной принцип работы алгоритма LZ78?

СПИСОК ЛИТЕРАТУРЫ

1) Передача дискретных сообщений. Под ред. Шувалова В. П. – М:. Радио и связь, 1991

2) Финк. Л. М. Теория передачи дискретных сообщений. М:. Радио и связь, 1970

3) Фано Р. Передача информации. Статистическая теория связи. Перевод с английского. Под ред. Добрушина Р. Л. – М.: Мир, 1965

ОГЛАВЛЕНИЕ

Формат 60x84 1/16
Заказ № - 119 . Тираж - 40

Отпечатано в Издательско полиграфическом
центре «ALOQACHI» при ТУИТ
Ташкент ул. Амир Темура, 108