УДК 681.3.07

Н. Н. ТАШАТОВ

ИСПОЛЬЗОВАНИЕ ИЗБЫТОЧНОСТИ ДЛЯ ЗАЩИТЫ ОТ ОШИБОК ПРИ КАНАЛЬНОМ КОДИРОВАНИИ И СТРУКТУРИРОВАННЫЕ ПОСЛЕДОВАТЕЛЬНОСТИ

Рассматриваются два метода использования избыточности для защиты от ошибок. Для проверки на наличие ошибки, в первом методе, используется контрольный бит четности, присоединяемый к данным. Этот метод называется *обнаружение ошибок и повторная передача*. Второй метод называется *прямым исправлением ошибок*. Он требует односторонней линии связи. В этом случае контрольный бит четности служит как для обнаружения, так и исправления ошибок.

Рассмотрим два основных метода использования избыточности для защиты от ошибок. В первом методе, для проверки на наличие ошибки используется контрольный бит четности, т.е. дополнительный бит, присоединяемый к данным, и назовем ее *обнаружение ошибок и повторная передача*. При этом приемное оконечное устройство не предпринимает никаких попыток исправить ошибку, оно просто посылает передатчику запрос на повторную передачу данных. Между передатчиком и приемником для такого диалога необходима двухсторонняя связь. Второй метод, требует лишь односторонней линии связи, поскольку в этом случае контрольный бит четности служит как для обнаружения, так и исправления ошибок и назовем ее *прямое исправление*

ошибок (FEC). Заметим, что не все комбинации ошибок можно исправить, так что коды коррекции классифицируются в соответствии с их возможностями исправления ошибок.

Тип соединения оконечных устройств. Согласно типу их соединения оконечные устройства систем связи часто классифицируют с другими оконечными устройствами. Рассмотрим возможные типы соединения, которые показанные на рис. 1. Тип соединения а) называются *симплексными*, б) *полудуплексными* и в) *полнодуплексными*.

Симплексное соединение на рис. 1, а) – это односторонняя линия связи. Передача сигналов производится *только* от оконечного устройства А к оконечному устройству В. Полудуплексное соединение показано на рис. 1, б) – это линия связи, посредством которой можно осуществить передачи сигналов в обоих направлениях, но не одновременно. Полнодуплексное соединение показано на рис. 1, в) – это двусторонняя связь, где передача сигналов происходит одновременно в обоих направлениях.

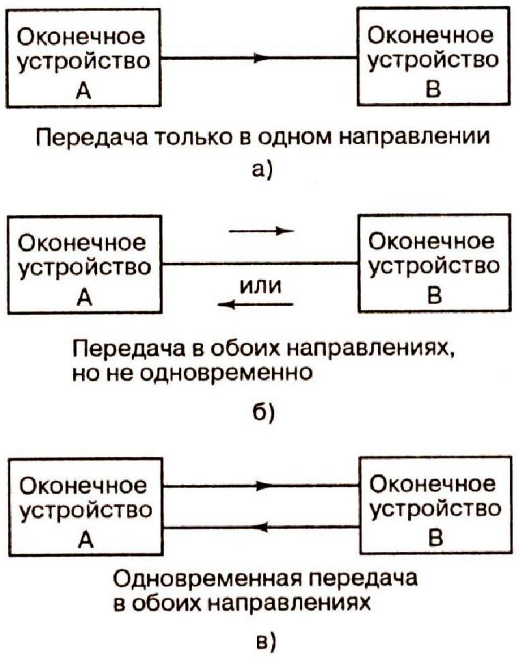

Рис. 1. Классификация типов соединения оконечных устройств: а) симплексное соединение; б) полудуплексное соединение; в) полнодуплексное соединение

Автоматический запрос повторной передачи. Система связи должна обеспечить средства предупреждения передатчика об опасности, если защита от ошибок заключается только в их обнаружении, сообщающие, что была обнаружена ошибка и требуется повторная передача. Такие процедуры защиты от ошибок называются методами *автоматического запроса повторной передачи* (ARQ). На рис. 2 показаны наиболее три распространенные процедуры ARQ. На каждой схеме ось времени направлена слева направо.

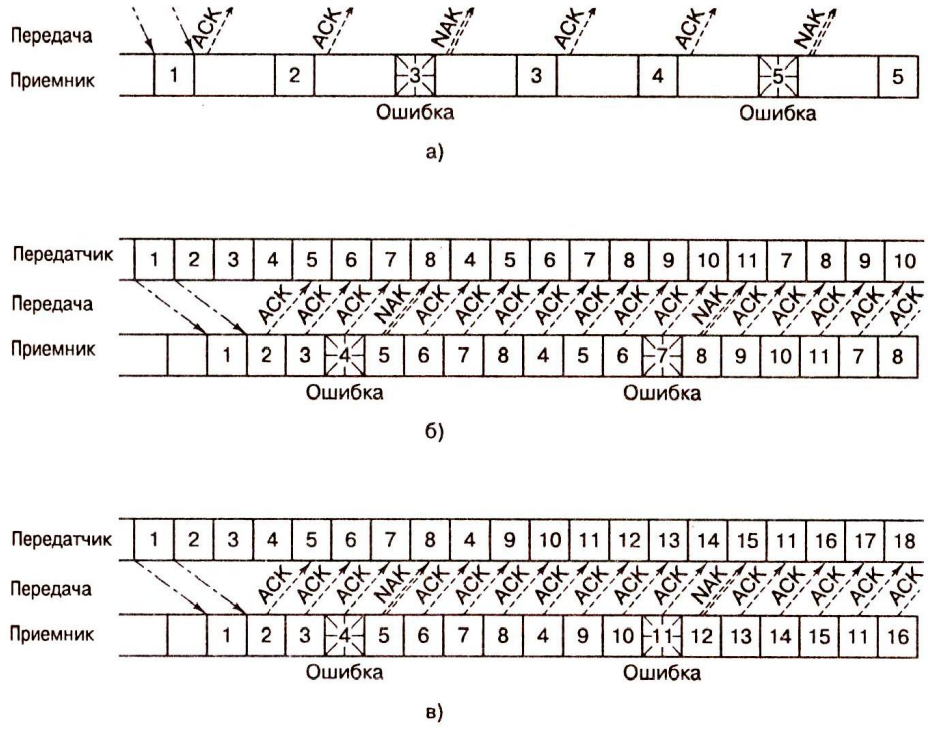

Рис. 2. Автоматический запрос повторной передачи (ARQ): а) запрос ARQ с остановками (полудуплексная связь); б) непрерывный запрос ARQ с возвратом (полнодуплексная связь); в) непрерывный запрос ARQ с выборочным повторением (полнодуплексная связь)

Запрос ARQ *с остановками*, первая процедура ARQ, показана на рис. 2, а). Так как передатчик перед началом очередной передачи ожидает подтверждения об успешном приеме (ACK) предыдущей, то ее реализация требует только полудуплексного соединения. На рисунке, третий блок передаваемых данных принят с ошибкой. Следовательно, приемник передает отрицательное подтверждение приема (NAK). Передатчик начинает повторять передачу третьего блока сообщения и только после этого передает следующий по очередности блок.

Непрерывный запрос ARQ *с возвратом*, вторая процедура ARQ, показана на рис. 2, б). Здесь требуется полнодуплексное соединение. Оба оконечных устройства начинают передачу одновременно: передатчик отправляет информацию, а приемник передает подтверждение о приеме данных. Каждому блоку передаваемых данных присваивается порядковый номер. Кроме того, номера кадров ACK и NAK должны быть согласованы; т.е., задержка распространения сигнала должна быть известна априори, чтобы передатчик знал, к какому блоку сообщения относится данный кадр подтверждения приема. На рис. 2, б) время подобрано так, что между отправленным блоком сообщений и полученным подтверждением о приеме существует постоянный интервал в четыре блока. Например, после отправки сообщения 8, приходит сигнал NAK, сообщающий об ошибке в блоке 4. При использовании процедуры ARQ передатчик «возвращается» к сообщению с ошибкой и снова передает всю информацию, начиная с поврежденного сообщения.

Третья процедура показана на рис. 2, в), называемая *непрерывным запросом* ARQ *с выборочным повторением.* Здесь, как и во второй процедуре, требуется полнодуплексное соединение. В этой процедуре повторно передается только искаженное сообщение, а затем передатчик продолжает передачу с того места, где она прервалась, не выполняя повторной передачи правильно принятых сообщений.

Компромиссом между требованиями эффективности применения ресурсов связи и необходимостью полнодуплексной связи является выбор конкретной процедуры ARQ. Полудуплексная связь (рис. 1, а) требует меньших затрат, чем полнодуплексная, но то же время она менее эффективна, что можно определить по количеству пустых временных интервалов. Более эффективная работа, показанная на рис. 2, б), требует более дорогой полнодуплексной связи.

Главное преимущество схем ARQ перед схемами прямого исправления ошибок (FEC) заключается в том, что обнаружение ошибок требует более простого декодирующего оборудования и меньшей избыточности, чем коррекция ошибок. Кроме того, она гибче: информация передается повторно только при обнаружении ошибки. С другой стороны, метод FEC может оказаться более приемлемым по одной из следующих причин [1]:

1. Обратный канал недоступен или слишком велика задержка при использовании ARQ.

2. Алгоритм повторной передачи нельзя реализовать удобным образом.

3. При ожидаемом количестве ошибок потребуется слишком много повторных передач.

Рассмотрим два метода из класса процедур кодирования, известных как коды с контролем четности. Эти процедуры канального кодирования относятся к *структурированным последовательностям*, т.к. они представляют методы введения в исходные данные структурированной избыточности таким образом, что это позволяет обнаруживать или исправлять ошибки. Структурированные последовательности делятся на три категории: *блочные*, *сверточные* и *турбокоды*. Будем рассматривать блочное кодирование. Пусть мы имеем двоичный симметричный канал и, следовательно, декодер использует жесткое решение.

Двоичный симметричный канал.

Частным случаем дискретного канала без памяти является *двоичный симметричный канал*, у которого входной и выходной алфавиты состоят из двоичных элементов 0 и 1. Условные вероятности имеют симметричный вид:

$$P(0/1) = P(1/0) = p,$$
$$P(1/1) = P(0/0) = 1 - p. \qquad (1)$$

Уравнение (1) называется *вероятностями перехода*. Канал, в котором вероятность ложных переходов равны друг другу и вероятность правильного приема одного сигнала равна вероятности правильного приема другого сигнала, будем называть *симметричным*. На рис. 3 дана модель такого канала.

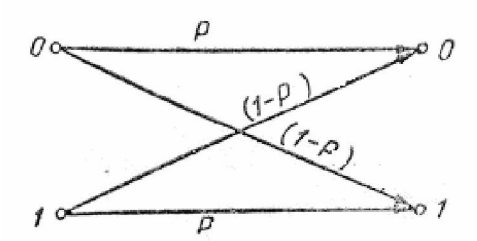

Рис. 3. Модель двоичного симметричного канала

Степень кодирования и избыточность.

При использовании блочных кодов исходные данные делятся на блоки из k бит, которые называются *информационными битами*, или битами сообщения. Каждый такой блок может представлять любое из 2^k отдельных сообщений. В процессе кодирования каждый k-битовый блок данных преобразуется в больший блок из n бит, который называется кодовым битом, или канальным символом. К каждому блоку данных кодирующее устройство прибавляет $(n - k)$ бит, которые называются либо *избыточными битами*, либо *битами четности*, либо *контрольными битами*. Эти биты новой информации не несут. Для обозначения кода используется запись (n, k). Отношение числа избыточных бит к числу информационных бит, $\frac{n-k}{k}$, называется *избыточностью* кода, а отношение числа бит данных к общему числу бит, $\frac{k}{n}$, будем называть *степенью кодирования*. Под степенью кодирования подразумевается доля кода, которая приходится на полезную информацию.

Код с одним контрольным битом.

Коды с контролем четности для обнаружения или исправления ошибок используют линейные суммы информационных битов, которые называются *символами* или *битами четности*. Код с одним контрольным битом – это прибавление к блоку информационных битов одного контрольного бита. Этот бит (бит четности) может быть равен нулю или единице, причем его значение выбирается так, чтобы сумма всех битов в кодовом слове была четной или нечетной. В операции суммирования используется арифметика по модулю 2. Данный код позволяет обнаруживать только однократные ошибки. Если бит четности выбран так, что результат четный, то говорят, что схема имеет *положительную четность*; если при добавлении бита четности результирующий блок данных является нечетным, то говорят, что он имеет *отрицательную четность*. На рис. 4 показана последовательная передача данных (первым является крайний справа бит). К каждому блоку добавляется один бит четности (крайний слева бит в каждом блоке), дающий положительную четность.

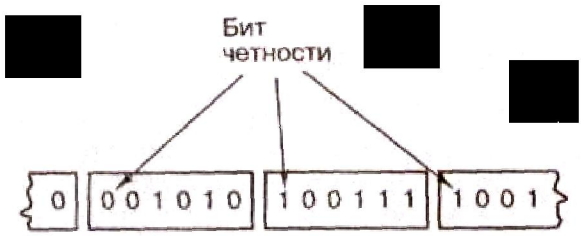

Рис. 4. Проверка четности для последовательной структуры кода

Декодирование производится в приемном оконечном устройстве, заключающееся в проверке, дают ли нуль суммы принятых битов кодового слова по модулю 2 (положительная четность). Если полученный результат равен 1, то кодовое слово заведомо содержит ошибки. Степень кодирования такого кода равен $\frac{k}{k+1}$. Декодер автоматически *исправить* цифру, полученную с ошибкой, не может. Только можно *обнаружить*, что в кодовом символе присутствует нечетное количество ошибок. (Если ошибка была внесена в четное число битов, то проверка четности покажет отсутствие ошибок; данный случай – это пример *необнаруженной ошибки.*) Предположим, что ошибки во всех разрядах равновероятны и появляются независимо, тогда можно записать вероятность появления j ошибок в блоке, состоящем из n символов:

$$P(n, j) = C_n^j p^j (1 - p)^{n-j}, \qquad (2)$$

где p – это вероятность получения *канального символа* с ошибкой, а через

$$C_n^j = \frac{n!}{j!(n-j)!} \qquad (3)$$

обозначается число различных способов выбора из n бит j ошибочных. Таким образом, для кода с одним битом четности вероятность необнаруженной ошибки $P_{\text{н.о.}}$ в блоке из n бит вычисляется следующим образом [1, 2]:

$$P_{\text{н.о.}} = \sum_{j=1}^{\substack{n/2 \text{ при } n=\text{четное} \\ (n-1)/2 \text{ при } n=\text{нечетное}}} C_n^{2j} p^{2j} (1-p)^{n-2j}. \quad (4)$$

Прямоугольный код.

Прямоугольный (*композиционный*) *код*, можно представить в виде параллельной структуры кода, которая показана на рис. 5.

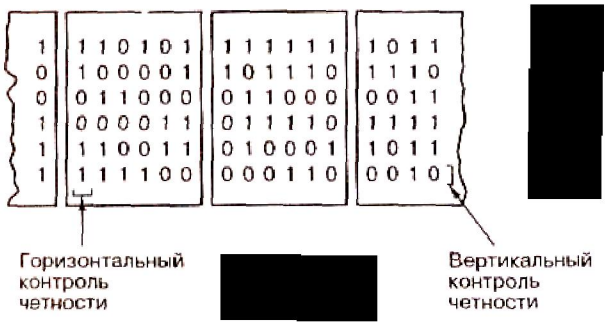

Рис. 5. Проверка четности для параллельной структуры кода

Код создается следующим образом. Вначале из битов сообщения строятся прямоугольники, состоящие из M строк и N столбцов. Затем к каждой строке и каждому столбцу прибавляется бит четности, в результате получаем матрицу размером $(M+1)\times(N+1)$. Степень кодирования прямоугольного кода k/n, запишем следующим образом:

$$\frac{k}{n} = \frac{M \cdot N}{(M+1)(N+1)}. \quad (5)$$

Зададимся вопросом: насколько прямоугольный код мощнее кода, который имеет один контрольный бит и предоставляет только возможность обнаружить одну ошибку? Заметим, что любая отдельная ошибка в разряде приведет к нарушению четности в одном столбце u в одной из строк матрицы. Тогда, прямоугольный код может исправить любую единичную ошибку, в которых была нарушена четность, так как расположение такой ошибки однозначно определяется пересечением строки и столбца. На рис. 5 размеры матрицы равны $M = N = 5$, следовательно, на рисунке отображен код (36, 25), способный исправлять единичные ошибки, расположенные в любом из 36 двоичных разрядов. Вычислим для такого блочного кода с коррекцией ошибок вероятность появления неисправленной ошибки, учитывая все способы появления *ошибки сообщения.* Для одинаковых, равноэнергетических ортогональных сигналов вероятность ошибки в кодовом слове (символе) P_E, можно оценить сверху [3]

$$P_E(M) \le (M-1) \cdot Q\left(\sqrt{\frac{E_s}{N_0}}\right), \quad (6)$$

где $M = 2^k$ – размер кодовых слов; k – число информационных бит в кодовом слове; $E_s = kE_b$ – энергия кодового слова;

$Q(x)$ – называется *гауссовым интегралом ошибок*, которая определяется следующим образом $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\left(-\frac{u^2}{2}\right) du$.

При фиксированном M с ростом E_b/N_0 оценка становится более точной, уже при $P_E(M) \le 10^{-3}$ уравнение (5) является довольно хорошим приближением. Используем связь между P_B и P_E для определения вероятности появления ошибочного бита, которое дается уравнением

$$\frac{P_B(k)}{P_E(k)} = \frac{2^{k-1}}{2^k - 1} \text{ или}$$

$$\frac{P_B(M)}{P_E(M)} = \frac{M/2}{M-1} = \frac{M}{2(M-1)}. \quad (7)$$

Объединяя (6) и (7), получим оценку вероятности появления ошибочного бита

$$P_B(k) \le 2^{k-1} \cdot Q\left(\sqrt{\frac{kE_B}{N_0}}\right) \text{ или}$$

$$P_B(M) \le \frac{M}{2} \cdot Q\left(\sqrt{\frac{kE_s}{N_0}}\right). \quad (8)$$

Из вероятности наличия j ошибок в блоке из n символов, записанной в выражении (6), можно записать вероятность ошибки сообщения, называемой также *блочной ошибкой* или *ошибочным словом*, для кода, который может исправить модели ошибок, состоящие из t или менее ошибочных битов:

$$P_M = \sum_{j=t+1}^{n} C_n^j p^j (1-p)^{n-j}, \quad (9)$$

где p – вероятность получения ошибочного *канального символа.* На рис. 5 при $t = 1$ код может исправить все однобитовые ошибки в прямо-

угольном блоке, состоящем из $n = 36$, бит. Суммирование в уравнении (9) начинается с $j = 2$:

$$P_M = \sum_{j=2}^{n} C_{36}^{j} p^j (1-p)^{36-j} . \qquad (10)$$

При достаточно малом p, наибольший вклад дает первое слагаемое суммы. Тогда, для примера с прямоугольным кодом (36,25) можно записать:

$$P_M \approx C_{36}^{2} p^2 (1-p)^{34} .$$

Точная *вероятность битовой ошибки* P_B зависит от конкретного кода и используемого декодера.

ЛИТЕРАТУРА

1. *Скляр Б.* Цифровая связь. Теоретические основы и практическое применение. Изд. 2-е, испр.: Пер. с англ. Издательский дом «Вильямс», 2004. 1104 с. ил.

2. *Березюк Н.Т., Андрущенко А.Г., Мощицкий С.С.* Кодирование информации (двоичные коды). Харьков: издательское объединение «Вища школа», 1978. 252 с.

3. *Lidsey W.C., Simon M.K.* Telecommunication System Engineering. Prentice-Hall, Inc., Englewood Cliffs, N. J. 1973.

Резюме

Мақалада қателіктерден сақтау үшін артығымен қолдану әдісінің екі әдісі қаралады. Қателігін тексеру үшін, бірінші әдісте деректерге қосылған қадағалау жұптық бит қолданылады. Бұл әдіс *қайта табыстау және қателіктерді табу* деп аталады. Екінші әдіс қателіктерді тікелей түзету әдісі. Ол бірбеткей байланыс желілігін талап етеді. Бұл жағдайда жұптық қадағалау биті қателікті түзету және оны табуға қызмет етеді.

Summary

Two methods of using of abundances for protection from mistakes are considered in the article. To check up the number of mistakes, control bit of evenness is used in the first method, connected to the given ones. This method is called *displaying of mistakes* and *repeated transmission.* The second method is called *direct correction of mistakes.* It requires one-sided lines of connection. In this case control bit of evenness serves as for displaying so for the correction of mistakes.

Поступила 2.09.07г.