

Системы, сети и устройства телекоммуникаций

УДК 621.341

МЕХАНИЗМ ИСПРАВЛЕНИЯ МНОЖЕСТВЕННЫХ ОШИБОК ПРИ ДЕКОДИРОВАНИИ СВЁРТОЧНОГО КОДА

Сидоренко Александр Анатольевич

кандидат технических наук; эксперт отделения пограничологии
Международной академии информатизации»

E-mail: likotasa@mail.ru.

Адрес: 125040, Россия, г. Москва, ул. Ленинградский проспект, д. 3.

Аннотация: В настоящее время свёрточные коды применяются в разнообразных системах передачи и хранения цифровых данных, в мобильной и спутниковой связи. Применение свёрточного кодирования позволяет эффективно восстанавливать потерянные данные при возникновении одиночных ошибок. В статье подробно рассмотрен процесс декодирования простейшего свёрточного кода с параметрами $k=1$, $n=2$, $K=3$. Данный код способен исправлять два любых, рядом расположенных ошибочных бита в бесконечной кодовой последовательности. Наличие трёх ошибочных бит при их близком расположении гарантированно приводит к появлению ошибочного бита. Ошибочное декодирование информационного бита в некоторых случаях может приводить к ошибочному декодированию последующих информационных бит. Вероятность ошибки декодирования значительно увеличивается при уменьшении длины информационной последовательности из-за слабой защищённости последних кодовых бит. Показана справедливость приведённой формулы для вычисления числа гарантированно исправимых ошибок.

Ключевые слова: кодирование с исправлением ошибок, свёрточные коды, одиночные и множественные ошибки.

Применение кодирования с исправлением ошибок позволяет восстанавливать данные, потерянные как при передаче, так и в процессе хранения. С целью исправления ошибок к информации добавляется некоторая избыточность. По способу введения избыточности в сообщении коды с исправлением ошибок разделяют на два больших класса: блочные коды и свёрточные коды.

При кодировании блочным кодом обработка информации происходит отдельными блоками определенной длины. В процессе кодирования блок информационных бит преобразуется в блок кодовых бит, обладающий структурной избыточностью. Кодирование линейного блокового кода, как правило, задается порождающей матрицей или вектором [1–5].

Свёрточные коды используют непрерывную обработку потока данных. Свёрточный кодер, в отличие от блочного, имеет память, и симво-

лы на его выходе зависят не только от очередного блока символов на входе, но и от предыдущих символов [6–10]. Свёрточное кодирование является отображением информационной последовательности символов в кодовую последовательность с помощью линейной схемы с параметрами, не меняющимися во времени. В настоящее время свёрточные коды применяются в разнообразных системах передачи и хранения цифровых данных, в мобильной и спутниковой связи [11].

При свёрточном кодировании очередная группа из k двоичных символов информационного сообщения путём линейного преобразования отображается в группу n двоичных кодовых символов. При этом в формировании кодовой группы принимают участие еще $K-1$ кодовых символов. Величину K называют длиной кодового ограничения. Как правило, свёрточный кодер реализуется kK -разрядным

регистром сдвига и n -сумматорами по модулю 2. В каждый шаг времени все символы в регистре смещаются на k -разрядов и на место первых k -разрядов регистра перемещаются k -новых символов, при этом на выходе n -сумматоров образуются кодовые символы. Широкое распространение получили двоичные свёрточные коды, для которых символ равен одному биту.

На примере двоичного свёрточного кода с параметрами $k=1$, $n=2$, $K=3$ рассмотрим процесс кодирования. Структурная схема кодера изображена на рис. 1.

В ответ на поступивший в кодер информационный бит формируется группа из двух кодовых бит. Значения кодовых бит являются линейной комбинацией входного бита и двух предыдущих информационных бит:

$$y_1 = u_i + u_{i+1} + u_{i+2}, \quad y_2 = u_i + u_{i+2}. \quad (1)$$

Очевидно, что содержимое ячеек памяти кодера может меняться лишь определенным образом. Например, какой бы ни был новый информационный бит, значение ячеек памяти не может измениться с 11 на 00 или с 00 на 11. Из каждого состояния ячеек памяти существует два перехода в новые состояния в зависимости от того, будет новый информационный бит нулем или единицей. Возможные варианты перехода из одной пары значений ячеек памяти в другую пару показаны на диаграмме состояний кодера (см. рис. 2).

Диаграмма состояний кодера представляет собой решётку с узлами, соответствующими состоянию ячеек памяти. Слева и справа диаграммы обозначены состояния ячеек памяти. Стрелками показаны возможные переходы между состояниями ячеек. При этом поступлению на вход кодера нулевого бита соответствует верхняя стрелка перехода, а нижняя стрелка перехода соответствует поступлению на вход кодера единицы. Пары кодовых бит, образующихся при переходах из одного состояния ячеек памяти в другое, показаны на соответствующих указателях переходов.

В процессе кодирования, шаг за шагом, пе-

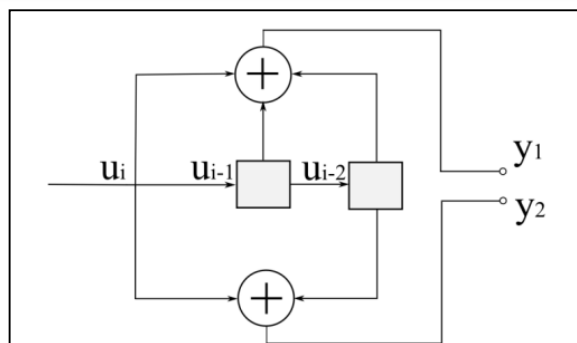


Рис.1. Структурная схема кодера

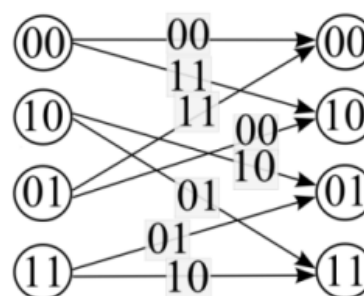


Рис.2. Диаграмма состояний кодера

реходя в соответствии с диаграммой состояний из одного состояния ячеек памяти в другое, формируется кодовая ветвь.

При декодировании свёрточных кодов наибольшее распространение получил алгоритм Витерби. В соответствии с данным алгоритмом, из всех ветвей декодером выбирается наиболее близкая к принятой кодовой последовательности. Для реализации процедуры выбора на каждом шаге декодирования для каждой ветви вычисляется метрика – число, показывающее, на сколько бит данная ветвь отличается от принятой кодовой последовательности. После каждого шага декодирования из двух ветвей, входящих в один узел решётки, отбрасывается ветвь, имеющая большую метрику. Оставшаяся ветвь называется выжившей.

Корректирующая способность свёрточных кодов определяется минимальным расстоянием между двумя кодовыми последовательностями, называемым также «свободным расстоянием» [12] и обозначаемым $d_{св}$. При определении минимального расстояния в качестве одной из последовательностей берут нулевую.

Покажем, как происходит определение «свободного расстояния» на примере рассматриваемого кода.

Как видно из рис. 3, нулевая кодовая ветвь имеет отличия от наиболее короткой расходящейся с ней ветви (111011) в пяти битах. Получаем для нашего кода $d_{CB}=5$. Максимальное количество гарантировано исправимых ошибок при декодировании равно [12, 13]:

$$t = \left\lfloor \frac{d_{CB} - 1}{2} \right\rfloor, \quad (2)$$

где $\lfloor \cdot \rfloor$ означают наибольшее целое, но не превышающее значение числа. Для рассматриваемого нами кода $t=2$. Это означает, что два ошибочных бита, возникшие в трёх подряд следующих парах кодовых бит, будут исправлены независимо от расположения в них ошибок. Далее будет произведена проверка справедливости данного утверждения путем рассмотрения механизма декодирования сверточного кода при наличии ошибок в трёх подряд следующих парах кодовых бит.

Рассмотрим более подробно алгоритм Витерби [14, 15] на примере декодирования кодовой последовательности, состоящей из нулей (соответствующей нулевой ветви решетки) при отсутствии ошибок (см. рис. 4). Пусть начальное состояние ячеек памяти: 00. На первом из

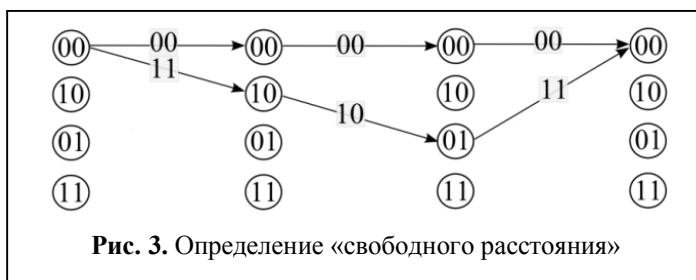


Рис. 3. Определение «свободного расстояния»

дальнейших шагов декодирования существует 2 разрешённых ветви: ветвь, идущая в узел 00, и ветвь, идущая в узел 10. Как видно из диаграммы состояний кодера, ветвь, идущая в узел 00, соответствует кодовым битам 00, а ветвь, идущая в узел 10, соответствует кодовым битам 11. Тогда метрика верхней ветви равна 0, а нижней ветви 2.

На следующем шаге декодирования образуется уже 4 ветви. Нулевая ветвь соответствует принятой последовательности, и ее метрика на каждом шаге равна 0. Метрики других ветвей равны соответственно 2, 3, 3. На третьем шаге декодирования и на всех последующих шагах присутствует 8 ветвей. В соответствии с алгоритмом Витерби необходимо оставить по одной ветви на каждый узел решетки, отбросив ветвь с большей метрикой. Выжившие после третьего шага декодирования ветви, показаны на рис. 5.

В нашем примере, на третьем шаге выжившие ветви имеют метрики: 0, 2, 3, 3. Необходимо отметить, что при отсутствии ошибок для всех вариантов кодовой последовательности на любом шаге декодирования, начиная со второго шага, метрики выживших ветвей будут иметь вид одной из 4 комбинаций: 1) 0, 2, 3, 3; 2) 2, 0, 3, 3; 3) 3, 3, 0, 2; 4) 3, 3, 2, 0.

Когда после очередного отбрасывания ветвей между некоторыми более ранними состояниями ячеек памяти остается один переход, можно однозначно определить соответствующий данному переходу информационный символ.

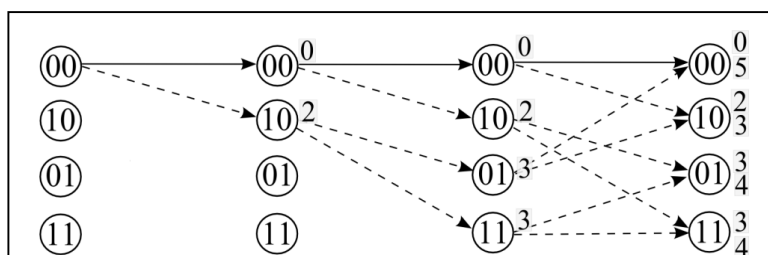


Рис. 4. Декодирование при отсутствии ошибок

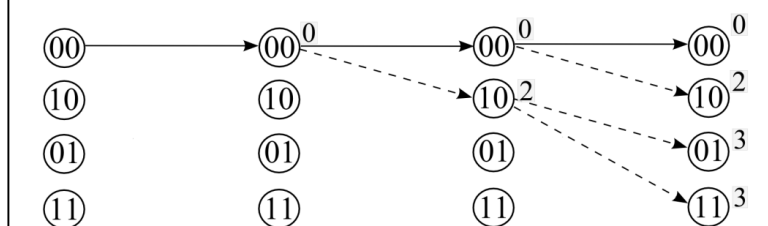


Рис. 5. Метрики выживших ветвей

Рассмотрим работу алгоритма Витерби на примере декодирования нулевой кодовой последовательности при наличии одиночной ошибки в любом из пары кодовых бит, поступивших в очередной шаг на вход декодера.

При поступлении на вход декодера пары информационных бит, содержащих ошибку, нулевая ветвь, которая до появления ошибки имела нулевую метрику, расходится на две ветви с единичной метрикой. Одну из ветвей, соответствующую информационной последовательности, назовем истинной (в данном случае нулевую ветвь), а вторую ветвь - конкурирующей (см. рис. 6). Будем отображать истинную ветвь сплошной линией, а её ответвления - длинным пунктиром. Коротким пунктиром обозначим конкурирующую ветвь и её ответвления. Декодированная информационная последовательность не будет содержать ошибок, если все ветви, кроме истинной, будут отброшены при формировании выживших ветвей.

Для удобства, здесь и далее, начнём условный отсчёт шагов декодирования с момента поступления пары кодовых бит, один из которых ошибочный. Вначале, при нулевом состоянии ячеек памяти, на вход декодера поступает пара бит: 01 или 10. Вычисляемые на первом шаге декодирования метрики истинной и конкурирующей ветви в любом случае равны 1. На следующем шаге истинная ветвь породит две ветви с метриками 1 и 3, а конкурирующая ветвь породит две ветви с метриками 2 и 2. Хочется отметить, что это справедливо и при всех прочих вариантах кодовой последовательности. То есть, на данном шаге декодирования существует всего 4 варианта метрик выживших ветвей (порождаемых истинной и

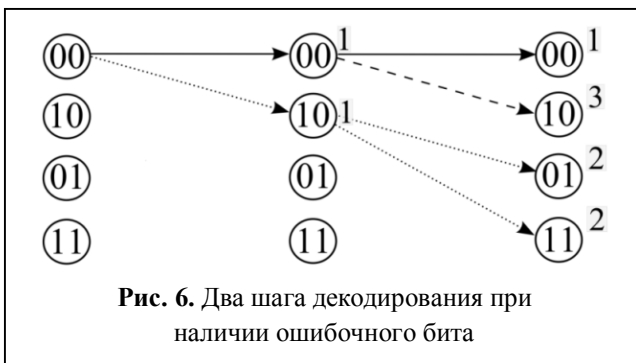


Рис. 6. Два шага декодирования при наличии ошибочного бита

конкурирующей ветвями): 1) 1, 3, 2, 2; 2) 3, 1, 2, 2; 3) 2, 2, 1, 3; 4) 2, 2, 3, 1.

С каждым дальнейшим шагом истинная ветвь продолжит порождать ветви с метриками 1 и 3, в то время как метрики ветвей, порождённых конкурирующей ветвью, будут увеличиваться, а значит, через некоторое время все ветви, порождённые конкурирующей ветвью, будут отброшены. Следует отметить, что при равенстве значений метрик ветвей, входящих в узел, выжившая ветвь выбирается произвольным образом.

Рассмотрим вариант декодирования, в котором при равенстве метрик ветвей, входящих в узел, будет отбрасываться ветвь, порождаемая конкурирующей ветвью. В этом случае отбрасывание всех ветвей, порождаемых конкурирующей ветвью, происходит за минимальное число шагов. Как показано на рис. 7, все ветви, порождённые конкурирующей ветвью, будут отброшены за 5 шагов.

Рассмотрим вариант декодирования, в котором при равенстве метрик ветвей, входящих в узел, будет отбрасываться ветвь, порождаемая истинной ветвью. В этом случае отбрасывание всех ветвей, порождаемых конкурирующей ветвью, происходит за максимальное число шагов. Как показано на рис. 8, все ветви,

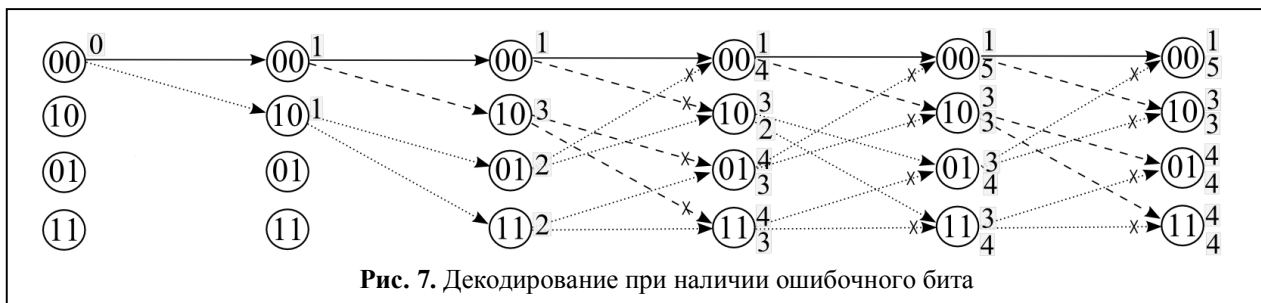


Рис. 7. Декодирование при наличии ошибочного бита

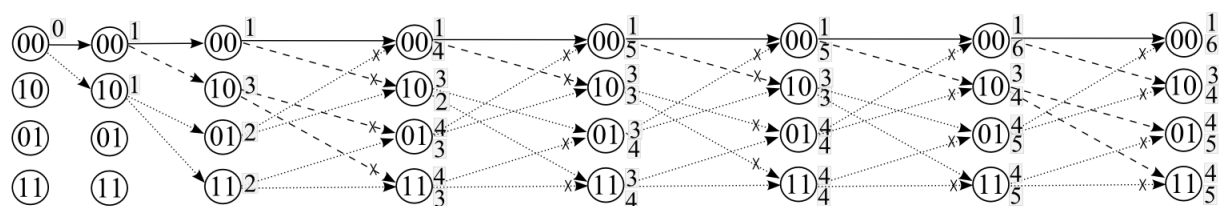


Рис. 8. Декодирование при наличии ошибочного бита

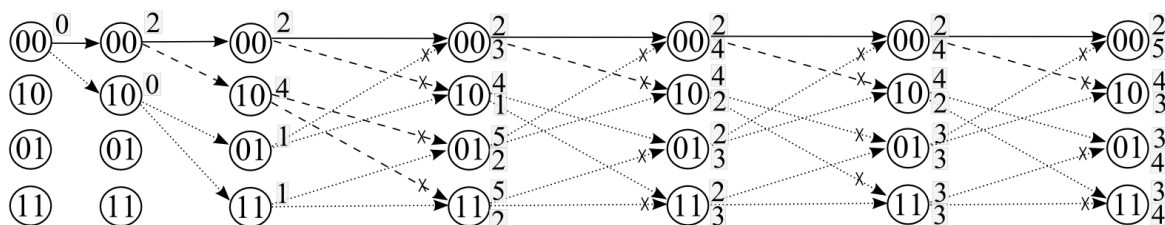


Рис. 9. Декодирование при наличии двух ошибочных бит

порождённые конкурирующей ветвью, будут отброшены за 7 шагов.

Несмотря на то, что на нейтрализацию последствий появления ошибочного бита уходит от 5 до 7 шагов декодирования, уже на 2 шаге метрика истинной ветви является наименьшей.

На практике свёрточный декодер обычно имеет дело с конечной последовательностью кодовых бит. После поступления на вход декодера последней пары кодовых бит отбрасываются все ветви, кроме имеющей наименьшую метрику. В случае, если окончание кодовой последовательности наступит уже после первого шага декодирования ошибочного бита (см. рис. 7, 8), вероятность ошибки декодирования будет равна 0,5 из-за равенства метрики истинной и конкурирующей ветви. В случае, если окончание декодируемой последовательности наступит после второго шага декодирования ошибочного бита или позднее, декодирование завершится без ошибок.

Рассмотрим, аналогично предыдущему случаю, процесс декодирования при наличии в принятой последовательности 2 ошибочных бит.

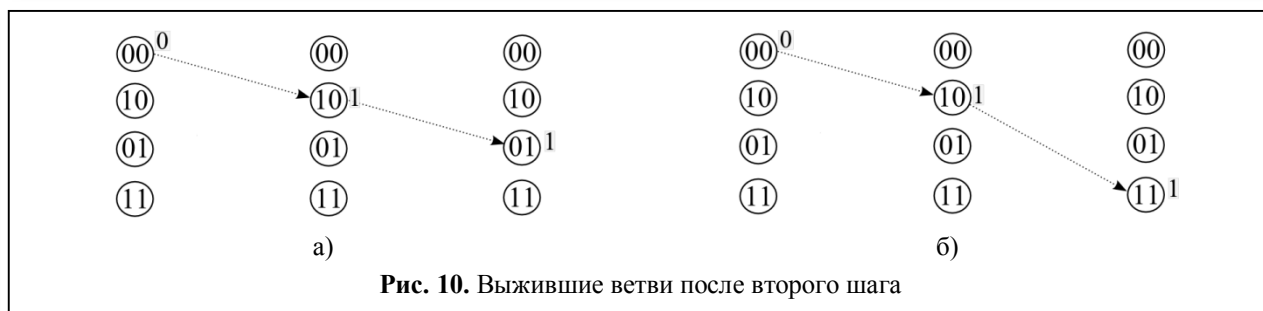
Пусть на очередном шаге декодирования из пары поступивших на вход декодера бит оба бита ошибочные. Метрика истинной ветви станет наименьшей на 6 шаге декодирования (см. рис. 9). Декодирование будет произведено без

ошибок, за исключением рассмотренных ниже случаев.

После первого шага декодирования метрики истинной и конкурирующей ветви равны 2 и 0 соответственно. Если окончание кодовой последовательности наступит уже после первого шага, то информационный бит будет декодирован как 1, то есть произойдет ошибка декодирования. Если окончание кодовой последовательности наступит после 2 шага, то выжившей ветвью будет одна из двух конкурирующих ветвей с метрикой равной 1, что приведет к ошибке декодирования одного (см. рис. 10,а) или двух информационных бит (см. рис. 10,б).

Если окончание кодовой последовательности наступит в диапазоне от 3 до 5 шага, будет ошибочно декодировано, как минимум, два информационных бита.

Результаты декодирования при другом расположении двух ошибочных бит в трёх следующих подряд парах кодовых бит будут схожи с рассмотренным выше случаем наличия двух ошибочных бит в одной паре кодовых символов. Метрика истинной ветви станет наименьшей на 6 шаге декодирования. Декодирование также будет произведено без ошибок, за исключением случаев, когда окончание кодовой последовательности наступит в диапазоне от 1 до 5 шага.



В случае декодирования бесконечной последовательности декодер исправляет два ошибочных бита при любом их расположении в трёх подряд следующих парах кодовых бит.

Рассмотрим процесс декодирования при наличии в принятой последовательности трёх ошибочных бит. Пусть нулевая кодовая последовательность в каждой из трёх подряд следующих пар кодовых бит содержит по одному ошибочному биту и имеет вид 011010. Процесс декодирования показан на рис. 11. Выжившие после сравнения метрик на третьем шаге кодирования ветви изображены на рис. 12. В процессе декодирования истинная ветвь отбрасывается, что приведёт к ошибке декодирования информационного бита. При другом расположении ошибок в кодовых битах, ошибок при декодировании также избежать невозможно.

При любом расположении трёх ошибок в трёх подряд следующих парах кодовых бит

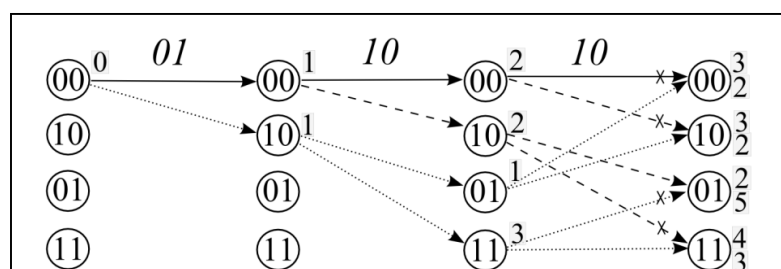


Рис. 11. Декодирование при наличии трех ошибочных бит

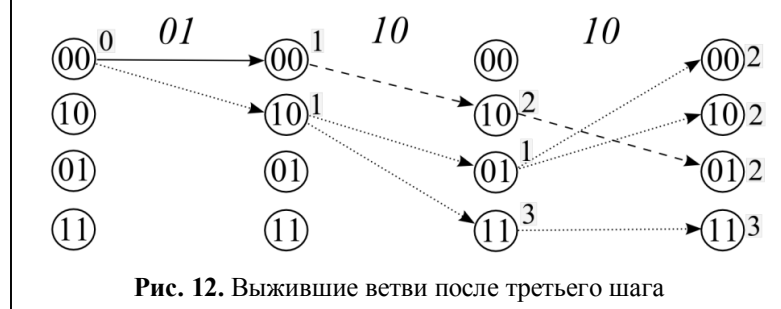


Рис. 12. Выжившие ветви после третьего шага

возникают ошибки декодирования. Справедливость формулы (2) подтверждена.

Литература

1. Питерсон У., Уэлдон Э. Коды, исправляющие ошибки. М. : Мир, 1976. 593 с.
2. Berlekamp E.R. Algebraic Coding Theory, rev. ed. Aegean Park Press, 1984.
3. Massey J.L. Shift Register and BCH Decoding // IEEE Trans. Info. Theory. Jan. 1969. Vol. IT-15, No. 1. Pp. 122-127.
4. Блейхут Р. Теория и практика кодов, контролируемых ошибки. М. : Мир, 1986. 576 с.
5. Berrou C., Glavieux A., Thitimajshima P. Near Shannon Limit Error – Correcting Coding and Decoding: Turbo-Codes, Proc. Intern. Conf. On Communication – ICC-93, Geneva, Switzerland, 1993. – Pp. 1064–1070.
6. Вернер М. Основы кодирования. М. : Техносфера, 2006. 286 с.
7. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М. : Техносфера, 2006. 319 с.
8. Мак-Вильямс Ф. Дж., Слоэн Н. Дж. А. Теория кодов, исправляющих ошибки. М.: Связь, 1979. 744 с.

9. Кларк Дж. мл., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи. М. : Радио и связь, 1987. 392 с.

10. Moon T. K. Error Correction Coding. Mathematical Methods and Algorithms. A John Wiley and Sons, Inc., 2005. 800 p.

11. Волков Л.Н., Немировский М.С., Шинаков Ю.С. Системы цифровой радиосвязи: базовые методы и характеристики : учебное пособие. М. : Эко-Трендз, 2005. 392 с.

12. Варгузин В.А. Методы повышения энергетической и спектральной эффективности цифровой радиосвязи. СПб. : БХВ-Петербург, 2013. 352 с.

13. Скляр Б. Цифровая связь. Теоретические основы и практическое

применение. М. : Издательский дом «Вильямс», 2007. 1104 с.

14. Viterbi A.J. Error bounds for convolutional codes and an asymptotically optimum decoding algo-

rithm // IEEE Transactions on Information Theory. 1967, April. Vol. 13, No. 2. Pp. 260–269.

15. Forney G. D. Jr. The Viterbi algorithm // Proceedings of the IEEE. 1973, March. Vol. 61. Pp. 268–278.

Поступила 24 октября 2017 г.

English

CORRECTION PROCEDURE OF MULTIPLE ERRORS WHEN DECODING THE CONVOLUTIONAL CODE

Alexander Anatolyevich Sidorenko – Candidate of Technical Sciences; Expert of the Frontier Issues Department of the International Academy of Informatization.

E-mail: likotasa@mail.ru.

Address: 125040, Russia, Moscow, Leningrad ave., 3.

Abstract: Currently, convolution codes are used in a variety of digital data transmission and storage systems, in mobile and satellite communications. The use of convolutional encoding enables the efficient lost data recovery in case of single errors. The decoding process of the simplest convolutional code with parameters $k=1$, $n=2$, $K=3$ is described in detail in the article. This code can correct any two nearly adjacent error bits in an infinite code sequence. The occurrence of three nearly adjacent erroneous bits surely causes the erroneous bit. Erroneous decoding of information bits in some cases may lead to erroneous decoding of subsequent information bits. The decoding error probability significantly grows as the information sequence lessens due to the weak last code bits protection. The validity of the given formula for calculating the number of surely correctable errors is shown.

Key words: error correction coding, convolutional codes, single and multiple errors.

References

1. Peterson W., Weldon E. Error-correcting Codes. Moscow : Mir, 1976. 593 p.
2. Berlekamp E.R. Algebraic Coding Theory, Rev. ed., Aegean Park Press, 1984.
3. Massey J. L. Shift Register and BCH Decoding // IEEE Trans. Info. Theory. Jan. 1969. Vol. IT-15. No. 1. Pp. 122–127.
4. Blahut R. Theory and practice of error control codes. Moscow : Mir, 1986. 576 p.
5. Berrou C., Glavieux A., Thitimajshima P. Near Shannon Limit Error – Correcting Coding and Decoding: Turbo-Codes, Proc. Intern. Conf. On Communication – ICC-93, Geneva, Switzerland, 1993. – Pp. 1064–1070.
6. Werner M. Coding: Fundamentals. M : Tekhnosfera, 2006. 286 p.
7. Morelos-Zaragoza R. The Art of error correcting coding. Methods, algorithms, application. M : Tekhnosfera, 2006. 319 p.
8. MacWilliams F.J., Sloane N.J.A. The Theory of Error-Correcting Codes. Moscow : Svyaz, 1979. 744 p.
9. George C. Clark Jr., Bibb Cain J. Error-Correction Coding for Digital Communications. Moscow : Radio i Svyaz, 1987. 392 p.
10. Moon T.K. Error Correction Coding. Mathematical Methods and Algorithms. A John Wiley and Sons, Inc., 2005. 800 p.
11. Volkov L.N., Nemirovsky M.S., Shinakov Y.S. Digital radio communication systems: Basic methods and properties: Textbook. Moscow : Eco-Trends, 2005. 392 p.
12. Varguzin V.A. Enhancing Methods of power and spectral efficiency of digital radio communication. SPb. : Bkhv-Petersburg, 2013. 352 p.
13. Sklyar B. Digital communication. Fundamentals and application. M. : publishing house "Williams", 2007. 1104 p.
14. Viterbi A.J. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm // IEEE Transactions on Information Theory. 1967, April. Vol. 13, No. 2. Pp. 260–269.
15. Forney G.D.Jr. The Viterbi algorithm // Proceedings of the IEEE. 1973, March. Vol. 61. Pp. 268–278.