УДК 681.391
DOI 10.24411/2409-3203-2019-12034

ЗАЩИТА ИНФОРМАЦИИ МЕТОДАМИ ИЗБЫТОЧНОГО КОДИРОВАНИЯ НА ОСНОВЕ КАСКАДНЫХ СХЕМ

Романенко Дмитрий Михайлович
к.т.н., заведующий кафедрой информатики и веб-дизайна
Белорусского государственного технологического университета
Республика Беларусь, г. Минск
Пацей Наталья Владимировна
к.т.н., заведующий кафедрой программной инженерии
Белорусского государственного технологического университета
Республика Беларусь, г. Минск

Аннотация: в данной статье рассматриваются принципы построения кодеков на основе последовательных каскадных и многомерных схем для исправления многократных, в том числе и пакетных ошибок.

Ключевые слова: избыточный код, многомерная схема кодирования, каскадная схема кодирования, ошибка, пакет, паритеты, декодер.

PROTECTION OF INFORMATION BY REDUNDANT CODING METHODS BASED ON CASCADE SCHEMES

Dmitri M. Romanenko
PhD, Head of the Informatics and Web-Design Department
Belarusian State Technological University
Republic of Belarus, Minsk
Natalia V. Patsei
PhD, Head of the Program Engineering Department
Belarusian State Technological University
Republic of Belarus, Minsk

Abstract: This article discusses the principles of constructing codecs based on sequential cascade and multidimensional schemes for correcting multiple and packet errors.

Keywords: redundant code, multidimensional coding scheme, cascade coding scheme, error, packet, parities, decoder.

В настоящее время широкое распространение получили и продолжают быстро развиваться области, связанные с передачей и соответственно защитой информации в беспроводных (спутниковых) сетях, системах хранения данных. В целом под защитой информации понимается деятельность, направленная на предотвращение утечки защищаемой информации, несанкционированных и непреднамеренных воздействий на защищаемую информацию. Для защиты от непреднамеренных воздействий передаваемые сигналы подвергаются специальной обработке с помощью методов избыточного кодирования для эффективного обнаружения изменений данных в условиях помех без потери информации.

В последние годы самым эффективным направлением в теории избыточного кодирования является использование методов комбинирования известных кодов, что позволяет приблизиться к оптимальной пропускной способности канала. Для этого необходимы компонентные коды с широким спектром скоростей, корректирующих

возможностей и эффективные алгоритмы декодирования: Так для нейтрализации пакетов ошибок высокой кратности интересными являются коды Рида-Соломона или Файра, а также сверточные коды, для исправления одиночных ошибок можно использовать коды Хэмминга, простые циклические или итеративные коды. Последние уже сами по себе являются примером комбинирования простых сверток по модулю 2 на основе кронекеровского произведение кодов [1].

Классический итеративный код [1] по сути и является прямым произведением двух сверток по модулю два. Развитием идеи комбинирования известных кодов стал трехмерный линейный итеративный код (ТЛИК) – код, полученный прямым произведением линейного итеративного кода и кода с простой проверкой четности [1]. При использовании трех и более кодов можно получить многомерные схемы кодирования, т.е. многомерные коды. Многомерные схемы итеративных кодов с числом проверок 5 (ТЛИК5), 7 (ТЛИК7) и 9 (ТЛИК9) описаны [2]. Необходимо отметить, что наилучшем для многомерных итеративных кодов является многопороговый метод декодирования [2] который включает несколько стадий (итераций) с различными пороговыми значениями. Стадии декодирования выполняются последовательно друг за другом, а, следовательно, обнаружение и исправление ошибок в кодовой последовательности выполняется несколько раз при различных пороговых значениях.

В теории избыточного кодирования также хорошо известна и широко применима на практике последовательная каскадная схема кодирования/декодирования. Каскадные схемы практически всегда обеспечивают гораздо более высокий энергетический выигрыш кодирования, чем исходные базовые кодеки, из которых формируются сами каскадные коды. Принципиальная схема использования каскадного кода, состоящего из двух составляющих кодов, показан на рисунке 1 [3].

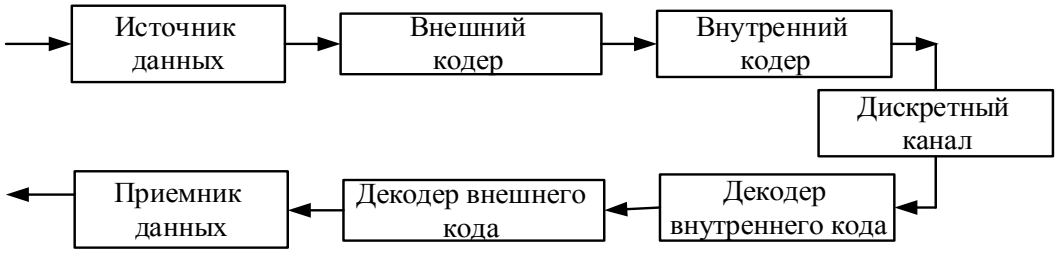

Рисунок 1 – Принципиальная схема последовательного каскадного кода с двумя компонентными кодами

На представленной схеме данные источника сначала кодируются внешним блочным (n_1, k_1) кодом. Затем закодированные символы внешнего кода кодируются кодером внутреннего (n_2, k_2) кода. Общая длина кодового слова каскадного кода оказывается равной $N = n_1 \cdot n_2$ двоичных символов, $K = k_1 \cdot k_2$, из них являются информационными. Результирующая кодовая скорость полученного каскадного кода будет равна:

$$\mathrm{R} = \frac{K}{N} = \frac{k_1 k_2}{n_1 n_2} = r_1 r_2 . \qquad (1.1)$$

где r_1, r_2 - кодовые скорости компонентных кодеров.

В ходе имитационного моделирования [4] каскадной схемы были использованы следующие элементарные коды: 40 циклических, 15 сверточных с различными комбинациями генераторных полиномов, 10 фонтанных LT–кодов и др. В таблице 1 приведены варианты циклических кодов, условно обозначенные как Су $_{n,k}$, где n и k размеры соответственно кодового и информационного слова, которые по результатам моделирования определены как наиболее перспективные в каскадной схеме. В столбце «Узлы $g(x)$» в таблице 1 представлены показатели корней порождающего полинома исключая сопряженные корни;

примитивный полином *m(x)*, определяющий поле, представлен в шестнадцатеричном виде).

Таблица 1 – Параметры и обозначения некоторых бинарных циклических кодов, использованных в имитационной модели каскадной схемы

Условное обозначение	*n*	*k*	*d*	*узлы g(x)*	*R*	*m(x)*
1	2	3	4	5	6	7
Cy $_{129,112}$	129	112	6	0,1,43	0.87	7EBF
Cy $_{129,86}$		86	14	0,1,19,21	0.66	
Cy $_{129,71}$		71	17	1,3,7,19,43	0.55	
Cy $_{129,44}$		44	30	0,1,3,7,9,11,19	0.34	
Cy $_{133,126}$	133	126	2	0,19,57	0.95	5B8D5
Cy $_{133,111}$		111	6	0,31,57	0.83	
Cy $_{133,75}$		75	16	1,7,19,31,57	0.56	
Cy $_{133,60}$		60	24	0,1,3,7,9	0,45	
Cy $_{133,42}$		42	28	0,1,5,7,9,31	0,32	

Важной моментом, характерным для используемых в каскадной схеме сверточных кодов является то, что код, исправляющий *t* ошибок, будет исправлять любой пакет ошибок длины *t*, т.е. данные коды особенно полезны для исправления группирующихся ошибок.

Используемые генераторные полиномы представлены в таблице 2.

Таблица 2 – Определение и обозначения генераторных полиномов

Полином	*r(x)*
R_2	$1 + x^2$
$R_{2'}$	$1 + x + x^2$
R_3	$1 + x + x^3$
$R_{3'}$	$1 + x + x^2 + x^3$
R_4	$1 + x^3 + x^4$
$R_{4'}$	$1 + x + x^3 + x^4$
$R_{4''}$	$1 + x + x^2 + x^3 + x^4$
R_5	$1 + x^2 + x^4 + x^5$
$R_{5'}$	$1 + x + x^2 + x^4 + x^5$
R_6	$1 + x^2 + x^3 + x^5 + x^6$
R_7	$1 + x^2 + x^5 + x^6 + x^7$
$R_{7'}$	$1 + x + x^2 + x^5 + x^6 + x^7$
$R_{7''}$	$1 + x^2 + x^3 + x^4 + x^5 + x^6 + x^7$
R_8	$1 + x^2 + x^3 + x^4 + x^8$
$R_{8'}$	$1 + x^2 + x^3 + x^4 + x^5 + x^8$
R_{11}	$1 + x^2 + x^4 + x^7 + x^{11}$
$R_{11'}$	$1 + x^2 + x^4 + x^7 + x^8 + x^9 + x^{10} + x^{11}$
R_{20}	$1 + x + x^2 + x^5 + x^6 + x^8 + x^9 + x^{12} + x^{13} + x^{14} + x^{16} + x^{17} + x^{18} + x^{19} + x^{20}$
R_{23}	$1 + x^2 + x^4 + x^5 + x^7 + x^8 + x^9 + x^{11} + x^{13} + x^{14} + x^{16} + x^{17} + x^{18} + x^{19} + x^{21} + x^{22} + x^{23}$

Сверточные коды, использованные при имитационном моделировании кодеков на основе каскадной схемы, описывались несколькими порождающими многочленами: они перечисляются через запятую, например C5,7 означает что в кодере будет два порождающих многочлена R_5 и R_7. Количество многочленов, определяется количеством выходных символов n_0. В целом необходимо отметить, что при использования исключительно сверточных кодов в каскаде без перемежителей достагется высокая

эффективность исправления ошибок (уменьшается BER), но при этом уменьшится и скорость кода R (рисунок 1).

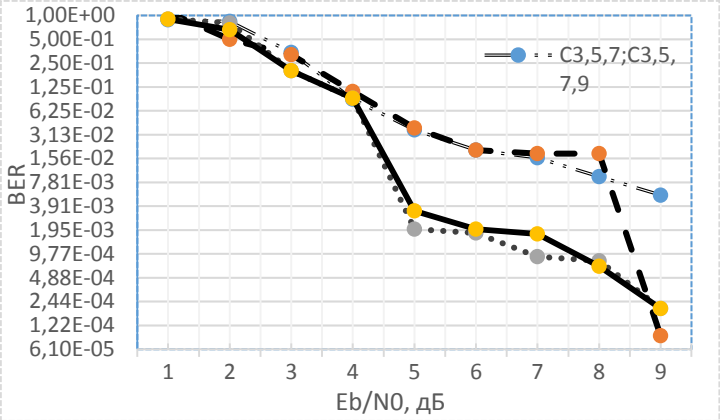

Рисунок 1 – График зависимости BER от отношения сигнал/шум для одиночных и каскадов сверточных кодов со скоростью от 0,2 до 0,33 без перемежителей

Скорость кода в тестируемых комбинациях изменялась от 0,2 до 0,33. Как видно отдельные каскады, состоящие из сверточных кодов, на два порядка улучшают BER по отношению к одиночному сверточному коду.

Фонтанные коды коррекции, необходимые для исправления ошибок типа «стирание», определяются размером блока, количества блоков и размером пакета передачи. Оценка вероятности декодирования была обеспечена на уровне 0,95. Результаты исследований представлены в таблице 3.

Таблица 3 – Параметры и характеристики фонтанных LT кодов

Размер пакета N, бит	Размер блока L, бит	% стираний в пакете	Вероятность правильного декодирования	Средняя избыточность,%
100	10	0	0,95	40
500	50			27
1000	100			25
100	10	5		50
500	50			30
1000	100			25
100	10	20		55
500	50			23
1000	100			25

Как видно при средней величине избыточности от 20-60% фонтанные LT коды могут исправлять до 20% стираний в пакете. Другие типы ошибок (единичные и модульные) фонтанные коды не могут исправлять, поэтому их применение оправдано только в составе каскадного кодека.

Для оценки производительности каскадного кодека был проведен ряд испытаний, позволяющий оценить корректирующую способность комбинации кодов. На рисунке 2 приведены исследования экспериментального вычисления скорости битовой ошибки (BER) для вектора нормализованного отношения Eb/No от 1 до 35 dB. При моделировании (рисунок 2) во всех случая использовался стандартный перемежитель между первым и вторым составными кодерами длины 3 для (Cy120,112; Cy133,122) и (Cy120,112; C3,5,7) и длины 4 для (C3,5,7,11; C3,5,7) и (H120; C3,5,7,11). Как видно из графика, коды с

максимальной избыточностью и низкой скоростью обеспечивают наивысшую корректирующую способность. Можно отметить, что ключевую роль в исправлении каскадным кодом ошибок, особенно пакетных, сыграли сверточные коды, которые сопоставимы с лучшими кодами, известными сегодня.

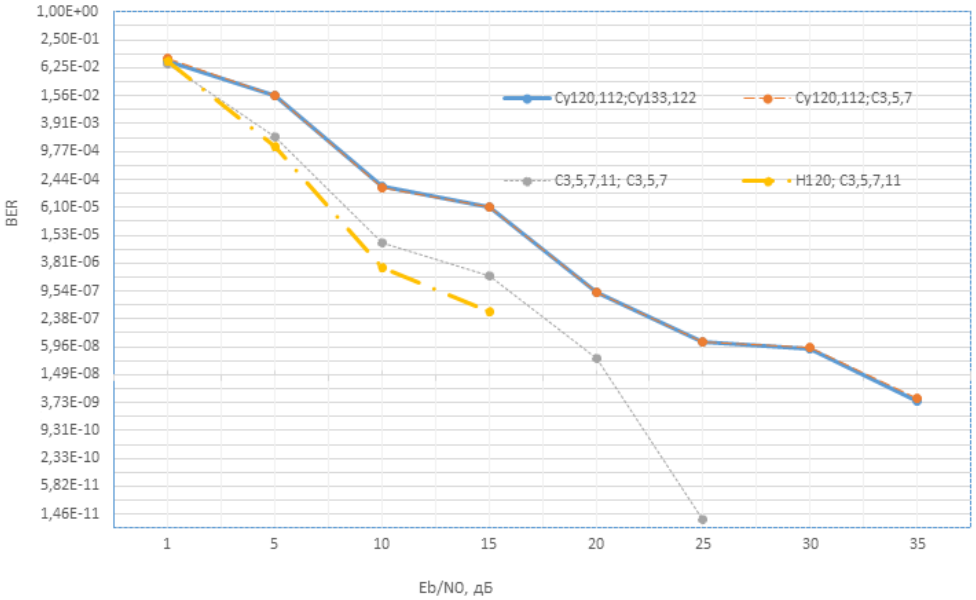

Рисунок 2 – График зависимости BER от отношения сигнал/шум для каскадного кода

Каскадная схема со сверточным кодом исправляет пакеты ошибок длины 4-26 бит при условии, что пакеты ошибок находятся достаточно далеко. Но при их использовании скорость снижается до значений от ¼ до ½., что потребует существенных временных затрат. Решением может выступать реализация «многопоточности» через использование модифицированой каскадной схемы кодирования, которая по сути соединена с многомерной схемой (представлена в [1] в виде многомерных итеративных кодов). Получим своего рода последовательно-параллельную схему кодирования/декодирования (рис. 2).

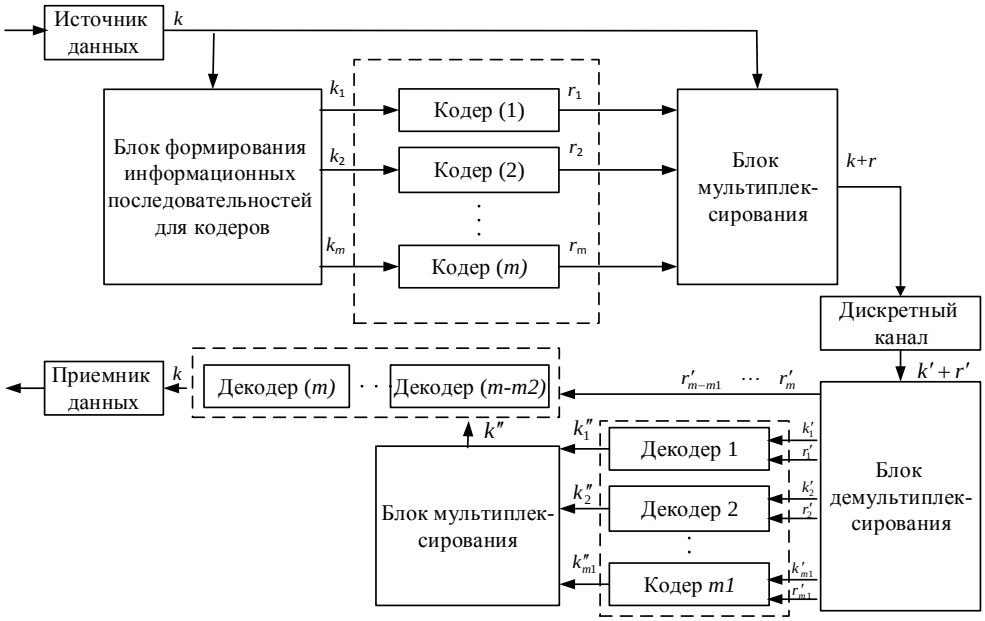

Рисунок 2 – Структурная схема последовательно-параллельной схемы кодирования с *m* компонентными кодами

Поступающая на этап кодирования информационная последовательность (*k*) записывается в трехмерную структуру (куб или параллелограмм), при этом линейный адрес

каждого информационного бита преобразуется в адрес с тремя координатами: номер плоскости, номер строки в плоскости, номер столбца в плоскости (рисунок 3). Так, например, для информационной последовательности длиной 64 бита 20-й бит (выделен полужирным начертанием и курсивом) получит адрес (2, 1, 4), т.е. вторая плоскость, 1 строка, 4 столбец, а 58-й бит получит адрес (4, 3, 2).

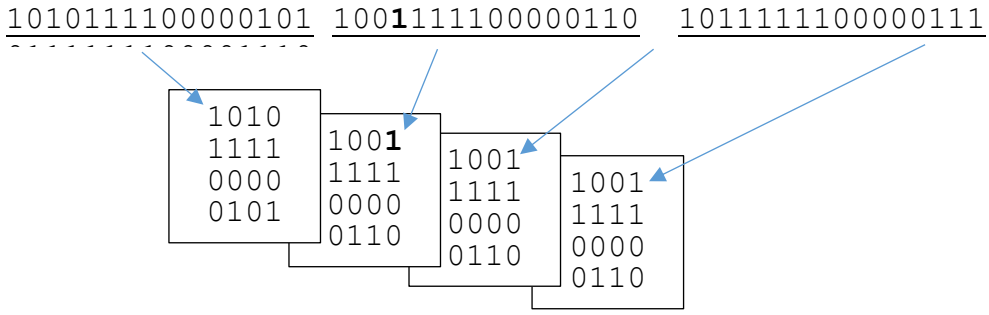

Рисунок 3 – Принцип формирования трехмерной структуры кода

На данном же этапе можно при необходимости осуществлять перемежение путем изменения последовательности записи бит. Далее из информационных бит в соответствии с новыми трехмерными адресами формируется набор информационных последовательностей (k_1, k_2 … k_m), каждая из которых подается на блок кодирования, состоящий из m кодеров. Примером последовательности могут быть как строка или столбец в плоскости, так и все плоскость в целом. Используемые коды могут быть как одинаковые, так и отличаться. В блоке мультиплексирования осуществляется формирование итоговой кодовой последовательности (k+r, где $r = r_1 + r_2 + \ldots + r_m$) путем объединения информационных бит (k) и полученных корректирующих символов (r_1, r_2 … r_m). После передачи данных на принимающей стороне осуществляется многостадийное декодирования принятой кодовой последовательности k'+r', причем на первой стадии некоторое число декодеров (m_1) выполняют операции параллельно, а декодированная информационная последовательность (k'') отправляется на следующие стадии декодирования, количество которых равно m_2, выполняемые последовательно, как в классической каскадной схеме.

Таким образом в результате компьютерного моделирования были определены комбинации составных кодов для каскадных кодеков. Установлено, что наибольший эффект с точки зрения корректирующей способности представляют каскады сверточных кодов (позволяют уменьшить BER на два порядка), однако скорость кода при этом уменьшается в 2-3 раза. Более практические в плане отношения корректирующая способность/скорость кода представляют каскады с первым каким-либо блочным кодом и вторым сверточным. Для уменьшения времени кодирования/декодирования при использовании сверточных кодов предлагается модифицировать последовательную каскадную схему до последовательно-параллельной на основе многомерной структуры, при этом интересным будет использование в качестве одного из компонентных кодов какой-либо из известных двумерный или трехмерный итеративный код.

Список литературы:

1. Multithreshold majority decoding of LDPC-codes / P. Urbanovich, D. Romanenko, D. Shiman, M. Vitkova // Informatyka Automatyka Pomiary. – Poland, Lublinie. – R. 84, № 4a/2012. – 2012. – P. 22–24.

2. Виткова, М.Ф. Адаптивное многопороговое декодирование многомерных итеративных кодов / М.В. Виткова, Д.М. Романенко // Труды БГТУ. Сер. VI. Физ.-мат. науки и информ. – Минск. – Вып. XX. – 2012. – С. 134–138.

3. Золотарёв В. В., Овечкин Г. В. Помехоустойчивое кодирование. Методы и алгоритмы: Справочник / Под. ред. чл.-кор. РАН Ю. Б. Зубарева. - М.: Горячая линия-Телеком, 2004. - 126 с.

4. Горовенко, Л.А. Математические методы компьютерного моделирования физических процессов / Л.А. Горовенко // Международный журнал экспериментального образования. – 2017. № 2. С. 92-93.

УДК 691.3
DOI 10.24411/2409-3203-2019-12035

ИЗВЕСТКОВО-ШЛАКОВОЕ ВЯЖУЩЕЕ КАК ЗАМЕНИТЕЛЬ ЦЕМЕНТА

Садирбаева Акмарал Махмутовна
Старший преподаватель кафедры Строительные материалы и технологии
Карагандинский Государственный технический университет
Казахстан, г. Караганда
Икишева Акнур Отановна
Старший преподаватель кафедры Строительные материалы и технологии
Карагандинский Государственный технический университет
Казахстан, г. Караганда
Дадиева Манара Кайриддиновна
Старший преподаватель кафедры Строительные материалы и технологии
Карагандинский Государственный технический университет
Казахстан, г. Караганда
Сыздықова Салтанат Қуатқызы
Ассистент кафедры Строительные материалы и технологии
Карагандинский Государственный технический университет
Казахстан, г. Караганда
Хан Максим Александрович
Преподаватель кафедры Строительные материалы и технологии
Карагандинский Государственный технический университет
Казахстан, г. Караганда

Аннотация: В области получения новых эффективных строительных материалов приоритетным направлением на сегодняшний день является рациональное использование вторичных сырьевых ресурсов. Одним из перспективных направлений по увеличению переработки техногенных отходов, как вторичного сырья для новых строительных материалов, является производство вяжущих веществ.

Технология производства вяжущих веществ с заданными специальными свойствами при максимальном использовании для их получения отходов производства имеют большую научную и практическую значимость.

Ключевые слова: строительные материалы, производство, отходы, промышленность, вяжущие вещества, сырье, бетон, технология.

LIME-SLAG BINDER AS A SUBSTITUTE FOR CEMENT

Sadyrbaeva Akmaral Mahmutovna