

УДК 621.341

Оценка эффективности турбокода, построенного на основе блочного кода, при декодировании в соответствии с критерием максимума апостериорной вероятности

Сидоренко А. А.

Постановка задачи: для исправления ошибок, возникающих при передаче информации по каналам связи, в настоящее время широко используются корректирующие коды. В каналах передачи данных с высокой вероятностью появления ошибки наиболее эффективными являются турбокоды. При построении турбокодов на основе блочных и сверточных кодов применяются различные алгоритмы декодирования. При декодировании турбокода на основе блочного кода с использованием синдромной решетки возможна реализация алгоритмов декодирования, применяемых для декодирования турбокодов на основе сверточных кодов. **Целью работы** является оценка эффективности турбокода, построенного на основе блочного кода, при декодировании в соответствии с алгоритмом декодирования по максимуму апостериорной вероятности (далее – MAP), ранее применявшимся для декодирования турбокодов, построенных на основе сверточных кодов. Предлагается оценить корректирующую способность турбокода, и его возможности по адаптации к изменениям характеристик канала передачи данных. **Используемые методы:** декодирование турбокода осуществлено с использованием алгоритма MAP. Моделирование процесса декодирования произведено в MS Excel и на языке программирования Python. В тексте программы генерация массивов случайных чисел производилась с использованием библиотеки «scipy» и ее субпакета «stats». При выводе формулы расчета вероятности появления ошибочного бита в декодированном информационном сообщении использовались положения теории вероятностей. Положения теории вероятностей также использовались для формирования мягкого выхода детектора. **Новизна:** элементами новизны обладает как само использование алгоритма MAP для декодирования турбокода, созданного на основе блочного кода, так и полученные результаты исследования корректирующей способности турбокода и возможности по адаптации его параметров к изменению характеристик канала. **Результат:** проведенное исследование корректирующей способности показало одинаковую эффективность исследуемого турбокода при его сравнении с турбокодом, созданным на основе систематического сверточного кода при равном числе информационных и кодовых бит. Проведенное моделирование продемонстрировало возможность адаптации параметров исследуемого турбокода к изменениям характеристик канала передачи данных путем выкалывания кодовых бит в количестве не более трех бит для декодирования жесткого решения детектора и не более пяти бит для случая декодирования мягкого решения детектора. Исследована возможность адаптации параметров турбокода путем изменения числа итераций декодирования. **Практическая значимость:** представленные исследования показали высокую эффективность турбокода, созданного на основе блочного кода, с использованием при декодировании хорошо изученного алгоритма MAP, а также имеющиеся возможности такого турбокода по адаптации к изменениям характеристик канала передачи данных. С целью исследования эффективности турбокода разработан программный тренажер на языке программирования Python.

Ключевые слова: кодирование с исправлением ошибок, турбокод, алгоритм MAP, блочные коды, сверточные коды, синдромная решетка.

Библиографическая ссылка на статью:

Сидоренко А. А. Оценка эффективности турбокода, построенного на основе блочного кода, при декодировании в соответствии с критерием максимума апостериорной вероятности // Системы управления, связи и безопасности. 2023. № 3. С. 29-43. DOI: 10.24412/2410-9916-2023-3-29-43

Reference for citation:

Sidorenko A. A. Evaluation of the efficiency of a turbo code built on the basis of a block code when decoding in accordance with the criterion of maximum a posteriori probability. *Systems of Control, Communication and Security*, 2023, no. 3, pp. 29-43 (in Russian). DOI: 10.24412/2410-9916-2023-3-29-43

Введение

Система цифровой связи способна нормально функционировать при условии, что число ошибок в принятом информационном сообщении (пакете) не превышает определенного значения [1]. Актуальной задачей является исправление ошибок, возникающих при передаче информации по каналам связи [2]. Турбокоды обладают наилучшей корректирующей способностью при передаче данных по каналам с высокой вероятностью появления ошибочного бита [3, 4]. В настоящее время турбокоды применяются в разнообразных системах цифровой связи [5]. Построение турбокода возможно как на основе систематического сверточного кода (ССК), так и на основе систематического блочного кода (СБК) [3, 6]. Для варианта построения турбокода на основе ССК разработан и хорошо исследован целый ряд алгоритмов декодирования (SOVA, MAP, Log-MAP, Max-Log-MAP и др.) [3, 6, 7]. Для варианта построения турбокода на основе СБК структура реализуется в виде кода-произведения, при этом декодирование производится, как правило, в соответствии с алгоритмом Чейза-Пиндиаха [7]. Реализация декодирования СБК с использованием синдромной решетки позволяет строить турбокоды на основе СБК с применением при декодировании перечисленных выше алгоритмов, используемых при построении турбокода на основе ССК [8]. Предлагается произвести оценку эффективности такого турбокода, построенного на основе СБК. В известных источниках информация по данному вопросу отсутствует. Для оценки эффективности турбокода исследуем его корректирующую способность и возможности по адаптации к изменениям характеристик канала передачи данных.

При эксплуатации каналов передачи данных, качество которых может изменяться, корректирующая способность кода часто выбирается в соответствии с максимально допустимой вероятностью появления ошибочного бита в информационном сообщении (P_{DMAX}), исходя из наихудшего состояния канала, которому соответствует наибольшая вероятность битовой ошибки в канале передачи данных (P_{BMAX}). Это может привести как к введению значительной избыточности, так и к росту числа итераций декодирования, и, как следствие, к существенному снижению скорости передачи данных. Качество канала передачи данных может значительно варьироваться из-за разнообразных факторов, например, изменений характеристик среды распространения и/или расстояния между приемником и передатчиком [9]. Если в данный момент времени вероятность битовой ошибки в канале передачи данных меньше, чем P_{BMAX} , то для обеспечения P_{DMAX} допустима меньшая избыточность кода или меньшее число итераций декодирования. В таком случае способность кода к адаптации является его преимуществом.

Постановка задачи

С целью оценки эффективности турбокода, построенного на основе СБК, будем осуществлять декодирование в соответствии с распространенным алгоритмом декодирования по максимуму апостериорной вероятности (MAP), подробно описанным в разнообразных источниках [3, 6, 10].

Оценка корректирующей способности будет осуществлена для случая наличия в канале независимых ошибок, путем нахождения вероятности появления ошибочного бита в декодированном информационном сообщении p_D . Значение p_D будет определяться по выведенной формуле и по результатам работы программного тренажера [11]. Найденное значение будет сравниваться со значением p_D для турбокода, построенного на основе ССК.

Одним из вариантов адаптации к изменению характеристик канала передачи данных является регулировка вносимой при кодировании избыточности. Снижение избыточности увеличивает реальную скорость передачи информации. Для снижения избыточности может потребоваться корректировка параметров применяемого помехоустойчивого кода или изменение типа кода. Одним из самых простых методов регулировки избыточности является перфорация. Перфорация заключается в исключении (выкалывании) определенных бит из потока передаваемых данных [10, 12]. Влияние перфорации на корректирующую способность турбокода будет исследовано путем оценки значения p_D при выкалывании разного количества бит из передаваемого кодового сообщения.

Скорость передачи данных можно варьировать, изменяя число итераций декодирования. Это достаточно простой вариант адаптации параметров помехоустойчивого кода к изменениям характеристик канала передачи данных. Исследование такого варианта адаптации будет произведено путем оценки значения p_D при разном числе итераций декодирования.

Методы

Рассмотрим более подробно процедуру построения и алгоритм декодирования исследуемого турбокода. Структурная схема N – размерного кодера турбокода изображена на рис. 1. Кодовое слово турбокода состоит из последовательности информационных бит и N последовательностей проверочных бит [10]. Построим турбокод на основе простейшего кода Хемминга. Кодовое слово такого кода Хемминга состоит из четырех информационных и трех проверочных бит. Примем $N=2$, тогда кодовое слово на выходе кодера турбокода состоит из информационных бит и двух блоков проверочных бит (см. рис. 2).

Опишем процесс создания кодовой последовательности турбокода на примере кодирования информационной последовательности 0100. В процессе кодирования информационная последовательность поступает на вход первого кодера Хемминга без перемежения. Кодирование может быть осуществлено с использованием синдромной решетки, изображенной на рис. 3 при записи информационных бит в правой части решетки [13]. На выходе первого кодера формируется первый блок проверочных бит 011. На вход второго кодера Хемминга поступает информационная последовательность после перемежения. Пусть при перемежении первый по счету бит информационной последовательности меняется с третьим, а второй с четвертым, тогда на вход второго кодера Хемминга поступает последовательность 0001, а на выходе формируется второй блок проверочных бит 101. Выход кодера турбокода состоит из информа-

ционных бит, проверочных бит с выхода первого кодера, проверочных бит с выхода второго кодера, и имеет вид: 0100 011 101.

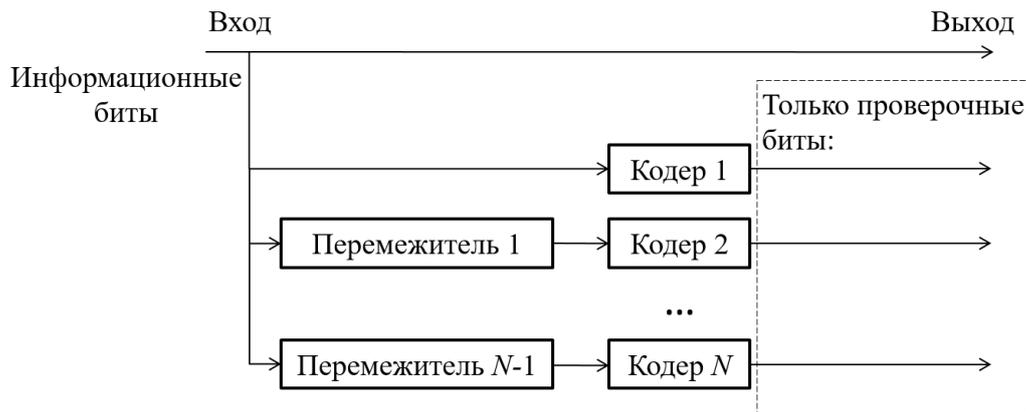


Рис. 1. Структурная схема N – размерного кодера турбокода

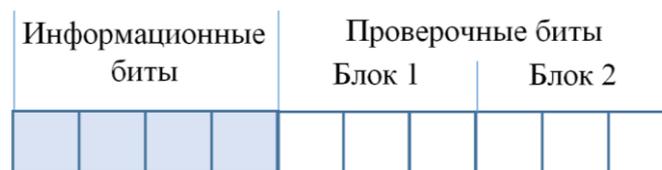


Рис. 2. Структура кодового слова турбокода

Перейдем к рассмотрению процесса декодирования турбокода. При декодировании будем применять алгоритм MAP с использованием синдромной решетки кода Хемминга [3, 13]. Использование синдромной решетки для кодирования и декодирования блочного кода рассмотрено в [8]. В качестве декодера турбокода используем распространенный параллельный декодер, структурная схема которого изображена на рис. 4 [3].

Для оценки корректирующей способности описанного турбокода, построенного на основе блочного кода Хемминга (ТСБК), произведена разработка в MS Excel реализации декодера, позволяющего осуществлять декодирование одного кодового слова и функционирующего на основе алгоритма MAP. Проанализированы результаты трех итераций декодирования жесткого решения детектора при разных вариантах расположения ошибочных бит в кодовом слове. Один ошибочный бит исправляется при любом расположении. Рассмотрим возможности декодера по исправлению двух ошибок при их произвольном расположении. Всего существует четыре случая расположения двух ошибочных бит в кодовом слове турбокода:

- 1 случай – обе ошибки располагаются в информационных битах (в этом случае существует 6 вариантов расположения ошибок);
- 2 случай – одна ошибка располагается в проверочных битах, а вторая в блоке информационных бит (24 варианта расположения ошибок);
- 3 случай – обе ошибки располагаются в одном блоке проверочных бит (6 вариантов расположения ошибок);
- 4 случай – ошибки располагаются в разных блоках проверочных бит (9 вариантов расположения ошибок).

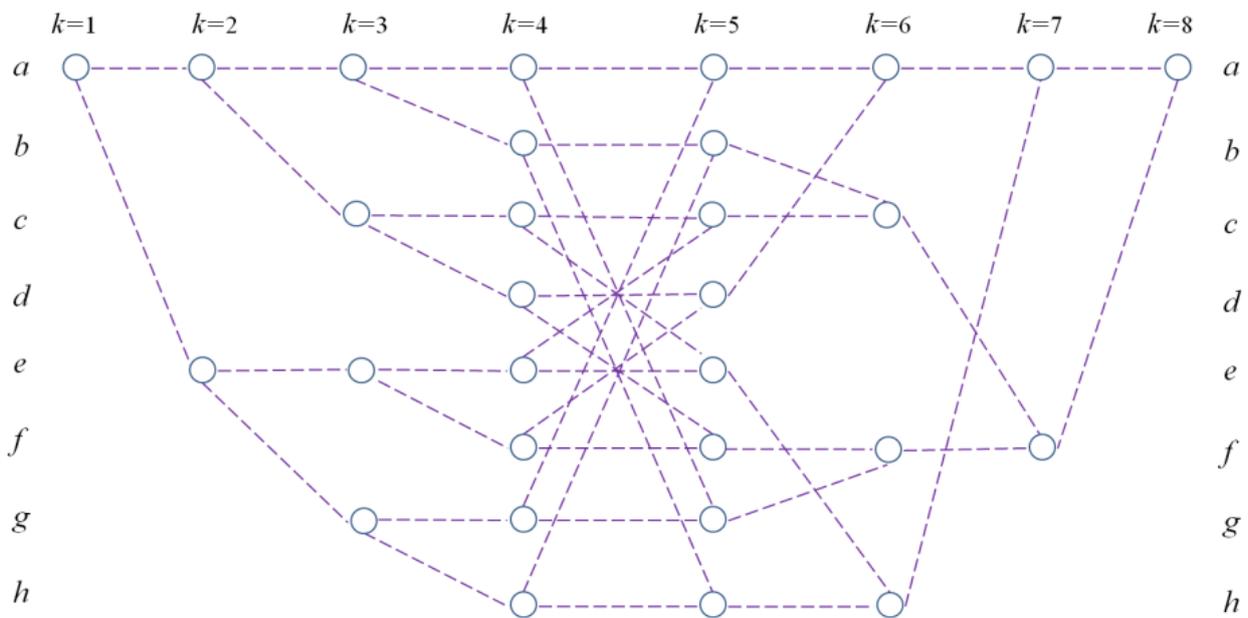


Рис. 3. Синдромная решетка кода Хемминга

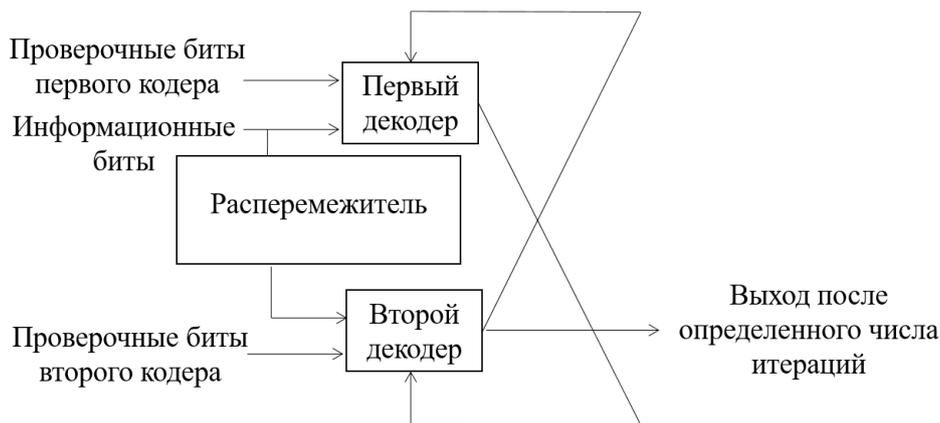


Рис. 4. Структурная схема параллельного декодера турбокода

Таблица 1 – Количество вариантов исправления двух ошибок в кодовом слове

Случаи расположения ошибок				Всего исправлено
1 случай	2 случай	3 случай	4 случай	
0	20	6	9	35

Рассмотрев приведенные в таблице 1 результаты декодирования при разном расположении двух ошибок, можно прийти к следующим заключениям. Две ошибки в проверочных битах исправляются при любом их расположении. Если хотя бы один ошибочный бит располагается в информационных битах, то может произойти ошибка декодирования. Ошибка декодирования кодового слова происходит в 10 случаях из 45 возможных, то есть в 22 % случаев. Вероятность появления 2 ошибочных бит из 10 равна:

$$P_{2uz10} = (10! / (2! \cdot (10 - 2)!)) \cdot p_B^2 \cdot (1 - p_B)^{10-2} = 45 \cdot p_B^2 \cdot (1 - p_B)^8,$$

где p_B вероятность появления ошибочного бита в канале передачи данных.

Так как вероятность наличия двух ошибочных бит в кодовом слове значительно превышает вероятность возникновения большего числа ошибок, то с небольшой долей приближения можно считать, что ошибочное декодирование слова вызвано наличием двух ошибок. С учетом того, что при наличии двух ошибок в кодовом слове его ошибочное декодирование осуществляется в 22 % случаев, вероятность ошибочного декодирования кодового слова p_W вычисляется по следующей формуле:

$$p_W \approx 0,22 \cdot P_{2\text{ош}} = 9,9 \cdot p_B^2 \cdot (1 - p_B)^8.$$

Исследования показали, что при ошибочном декодировании кодового слова ошибка возникает в среднем в 2,25 битах из четырех информационных бит. Тогда вероятность появления ошибочного бита в декодированном информационном сообщении p_D равна:

$$p_D = (2,25 / 4) \cdot p_W = 5,6 \cdot p_B^2 \cdot (1 - p_B)^8. \quad (1)$$

Предлагается произвести оценку корректирующей способности ТСБК, исходя из полученного значения p_D , сравнив его со значением p_D турбокода, построенного на основе ССК (ТССК), имеющего такую же структурную схему кодера турбокода (см. рис. 1 при $N=2$) и декодера турбокода (см. рис. 4). Структурная схема кодера ССК изображена на рис. 5. Структурная схема кодового слова ТССК изображена на рис. 6.

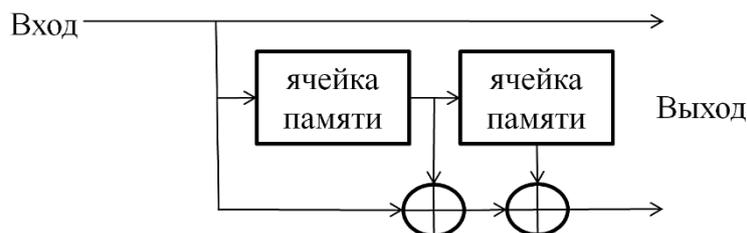


Рис. 5. Структурная схема кодера ССК



Рис. 6. Структура кодового слова турбокода

Для оценки корректирующей способности ТСБК при разных значениях p_B произведен расчет p_D по формуле (1). Также для исследования корректирующей способности ТСБК и ТССК созданы программные тренажеры на языке программирования *Python* [14]. Каждый программный тренажер ТСБК и ТССК состоит из описанного кодера турбокода, блока моделирования выхода детектора приемника с заданной вероятностью появления ошибочного бита и описанного декодера турбокода. Вначале работы программного тренажера осуществляется генерация массива информационных бит, с равной вероятностью принимающих значение ноль или единица. Генерация массива осуществляется с использованием библиотеки «*scipy*» и ее субпакета «*stats*», содержащего статистические функции [15]. Листинг программы приведен в приложении 1. Сгенериро-

ванный массив информационных бит разбивается на блоки и кодируется. Далее опишем блок моделирования выхода детектора приемника с заданной вероятностью появления ошибочного бита в канале передачи данных.

Рассмотрим моделирование выхода детектора при формировании им жесткого решения. Вначале с использованием библиотеки «*scipy*» производится генерация массива ошибок - массива двоичных чисел в котором единичные биты встречаются с вероятностью появления ошибочного бита в канале передачи данных. Листинг программы приведен в приложении 2. Сгенерированный массив ошибок побитово складывается с массивом кодовых бит в соответствии с правилом сложения по модулю 2. В результате чего в кодовом массиве возникают ошибки с заданной вероятностью. Сформированный таким образом массив является моделью жесткого выхода детектора.

Опишем формирование модели мягкого выхода детектора с заданной вероятностью появления ошибочного бита. Более подробно данный вопрос изложен в источнике [16]. Сделаем ряд необходимых допущений. Канал передачи информации является двоичным симметричным каналом. Шум в канале является аддитивным белым гауссовским шумом. Передача бит по каналу ведется противоположными сигналами. Отдельный бит кодового слова «1» или «-1» после прохождения канала без шума и детектирования принимает аналоговое значение r_0 или $-r_0$. Из-за наличия шума в канале детектированный сигнал является непрерывной стохастической переменной с плотностями условных вероятностей, приведенными на рис. 7.

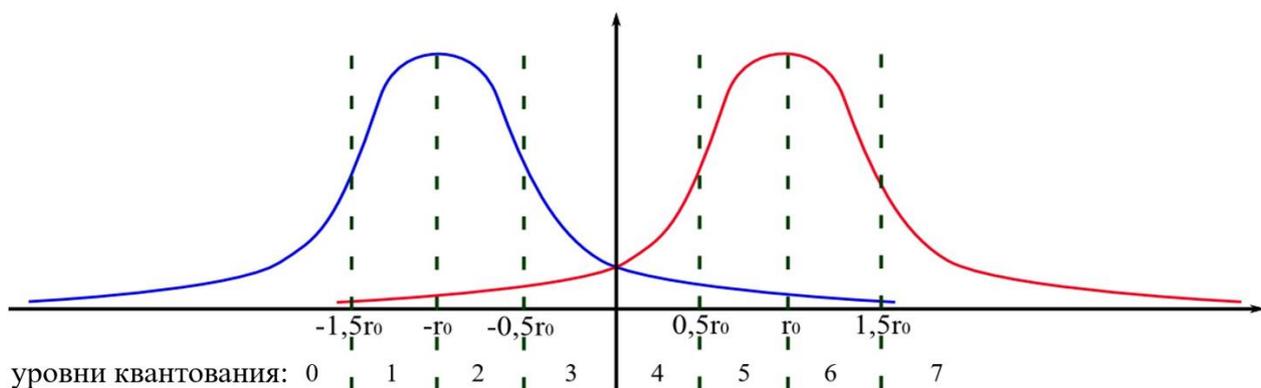


Рис. 7. Правило квантования принятого сигнала на 8 уровней

Формируя мягкое решение детектора необходимо значение каждого элемента массива кодовых бит заменить одним из K символов, где K – число уровней квантования. Для этого необходимо сформировать вспомогательный массив уровней квантования. Пусть $K = 8$, а правило квантования принятого сигнала приведено на рис. 7. Правило генерации элементов массива уровней квантования при $p_B=0,1$ приведено в таблице 2. Листинг программы, осуществляющей генерацию массива уровней, приведен в приложении 3. После формирования массива уровней производится его сложение с массивом кодовых бит по правилу, представленному в таблице 3. На выходе процедуры сложения получается массив, моделирующий мягкий выход детектора.

Таблица 2 – Вероятности генерации уровней квантования

Элемент массива	«0»	«1»	«2»	«3»	«4»	«5»	«6»	«7»
Вероятность	$6,9 \cdot 10^{-4}$	0,0045	0,02211	0,0727	0,1608	0,2392	0,239	0,261

Таблица 3 – Правило сложения

Уровни квантования		0	1	2	3	4	5	6	7
Биты кодового массива	0	7	6	5	4	3	2	1	0
	1	0	1	2	3	4	5	6	7

Далее перейдем к рассмотрению результатов исследования эффективности турбокода, полученных с использованием формулы (1) и программного тренажера.

Результаты

Произведем оценку корректирующей способности ТСБК. Результаты расчетов по формуле (1), а также результаты работы программных тренажеров ТСБК и ТССК для случая декодирования жесткого решения приведены в таблице 4.

Таблица 4 – Значение p_D при декодировании жесткого решения детектора

Применяемый код	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
ТСБК по формуле	$2,4 \cdot 10^{-2}$	$5,2 \cdot 10^{-4}$	$5,5 \cdot 10^{-6}$	$5,6 \cdot 10^{-8}$
ТСБК (результаты работы программного тренажера)	$5,2 \cdot 10^{-2}$	$5,2 \cdot 10^{-4}$	$5,5 \cdot 10^{-6}$	$5,6 \cdot 10^{-8}$
ТССК (результаты работы программного тренажера)	$3,1 \cdot 10^{-2}$	$1,3 \cdot 10^{-4}$	$8,5 \cdot 10^{-7}$	$4,7 \cdot 10^{-9}$
ТССК с выкалыванием двух бит (результаты работы программного тренажера)	$4,9 \cdot 10^{-2}$	$5,4 \cdot 10^{-4}$	$6,1 \cdot 10^{-6}$	$6,0 \cdot 10^{-8}$

В связи с тем, что при одинаковой длине информационной последовательности (4 бита), длина кодового слова ТССК (12 бит) на 2 бита превышает длину кодового слова ТСБК (10 бит), вызывает интерес сравнение корректирующей способности ТСБК с ТССК при одинаковой длине кодового слова. В связи с этим в нижней строке таблицы 4 приведены результаты работы программного тренажера ТССК при выкалывании двух бит в кодовом слове турбокода. Проанализировав полученные данные, можно сделать следующие выводы:

- расчеты по формуле (1) для значений $p_B \leq 1 \cdot 10^{-2}$ совпадают с результатами работы программы;
- корректирующая способность ТСБК ниже, чем у ТССК;
- при наличии в кодовом слове ТССК двух выколотых бит, корректирующие способности турбокодов отличаются незначительно.

В дополнение к сделанным выводам следует отметить существенную разницу значений p_D , полученных по формуле (3) и по результатам работы программного тренажера при $p_B=1 \cdot 10^{-1}$. По всей видимости, это вызвано тем, что формула (1) не учитывает возникновение ошибки декодирования при наличии в кодовом слове более двух ошибок, что при $p_B=1 \cdot 10^{-1}$ достаточно вероятно.

Известно, что корректирующая способность турбокодов значительно возрастает при декодировании мягких решений детектора. Результаты работы программных тренажеров ТСБК и ТССК для случая декодирования мягкого решения детектора, квантованного на 8 уровней, приведены в таблице 5. Рассмотрев полученные результаты, можно сделать следующие выводы:

- корректирующая способность ТСБК значительно ниже, чем у ТССК;
- при наличии в кодовом слове ТССК двух выколотых бит, корректирующие способности турбокодов отличаются незначительно.

Таблица 5 – Значение p_D при декодировании мягкого решения детектора

Применяемый код	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
ТСБК (результаты работы программного тренажера)	$2,50 \cdot 10^{-2}$	$5,47 \cdot 10^{-5}$	$1,40 \cdot 10^{-7}$	$3,79 \cdot 10^{-10}$
ТССК (результаты работы программного тренажера)	$9,55 \cdot 10^{-3}$	$2,81 \cdot 10^{-6}$	$3,50 \cdot 10^{-9}$	$2,91 \cdot 10^{-12}$
ТССК с выкалыванием двух бит (результаты работы программного тренажера)	$1,96 \cdot 10^{-2}$	$2,90 \cdot 10^{-5}$	$1,01 \cdot 10^{-7}$	$2,4 \cdot 10^{-10}$

Перейдем к изучению возможности ТСБК по адаптации к изменениям характеристик канала передачи данных. Рассмотрим влияние перфорации на корректирующую способность ТСБК. Результаты работы программного тренажера для случая декодирования жесткого решения детектора и наличия выколотых бит приведены в таблице 6. Проведя анализ полученных значений можно сделать вывод, что с ростом числа выколотых бит происходит плавное увеличение p_D при числе итераций от 1 до 3. При выкалывании четырех бит корректирующая способность резко снижается. При числе выколотых бит более трех $p_D > p_B$, и процедура кодирования не имеет смысла.

Таблица 6 – Значение p_D при разном количестве выколотых бит для случая декодирования жесткого решения детектора

Число выкал	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
1	$6,0 \cdot 10^{-2}$	$7,3 \cdot 10^{-4}$	$6,9 \cdot 10^{-6}$	$7,2 \cdot 10^{-8}$
2	$6,1 \cdot 10^{-2}$	$7,4 \cdot 10^{-4}$	$7,2 \cdot 10^{-6}$	$7,3 \cdot 10^{-8}$
3	$6,6 \cdot 10^{-2}$	$8,6 \cdot 10^{-4}$	$8,4 \cdot 10^{-6}$	$9,6 \cdot 10^{-8}$
4	$6,7 \cdot 10^{-2}$	$1,0 \cdot 10^{-2}$	$1,0 \cdot 10^{-3}$	$1,0 \cdot 10^{-4}$

Рассмотрим влияние перфорации на успешность декодирования мягкого решения детектора. Результаты работы программного тренажера приведены в таблице 7. Проведя анализ полученных значений можно сделать вывод, что с увеличением числа выколотых бит повышение p_D происходит более резко, чем для случая декодирования жесткого решения. При числе выколотых бит более пяти $p_D > p_B$, и процедура кодирования не имеет смысла.

Таблица 7 – Значение p_D при разном количестве выколотых бит для случая декодирования мягкого решения детектора

Число выкол	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
1	$3,2 \cdot 10^{-2}$	$1,0 \cdot 10^{-4}$	$4,6 \cdot 10^{-7}$	$2,4 \cdot 10^{-9}$
2	$3,4 \cdot 10^{-2}$	$1,8 \cdot 10^{-4}$	$9,7 \cdot 10^{-7}$	$4,9 \cdot 10^{-9}$
3	$4,3 \cdot 10^{-2}$	$3,8 \cdot 10^{-4}$	$3,7 \cdot 10^{-6}$	$4,2 \cdot 10^{-8}$
4	$6,7 \cdot 10^{-2}$	$2,6 \cdot 10^{-3}$	$1,4 \cdot 10^{-4}$	$9,3 \cdot 10^{-6}$
5	$8,2 \cdot 10^{-2}$	$5,5 \cdot 10^{-3}$	$4,4 \cdot 10^{-4}$	$3,3 \cdot 10^{-5}$
6	$1,0 \cdot 10^{-1}$	$1,0 \cdot 10^{-2}$	$1,0 \cdot 10^{-3}$	$1,0 \cdot 10^{-4}$

Рассмотрим возможности по адаптации ТСБК к изменениям характеристик канала передачи данных путем изменения числа итераций декодирования. Результаты работы программного тренажера для случая декодирования жесткого решения детектора приведены в таблице 8. Проведя анализ полученных значений можно сделать вывод, что с увеличением числа итераций вероятность появления ошибочного бита в декодированном информационном сообщении снижается, однако при числе итераций более трех это снижение крайне незначительно. Результаты работы программного тренажера для случая декодирования мягкого решения детектора приведены в таблице 9. По полученным данным можно сделать вывод, что с увеличением числа итерации с одной до трех происходит резкое снижение вероятности появления ошибочного бита в декодированном информационном сообщении (более резкое, чем для случая декодирования жесткого решения детектора). При числе итераций более трех снижение p_D крайне незначительно.

Таблица 8 – Значение p_D при разном количестве итераций декодирования жесткого решения детектора

Число итераций декодирования	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
1	$6,21 \cdot 10^{-2}$	$8,49 \cdot 10^{-4}$	$8,41 \cdot 10^{-6}$	$7,22 \cdot 10^{-8}$
2	$5,67 \cdot 10^{-2}$	$7,51 \cdot 10^{-4}$	$8,12 \cdot 10^{-6}$	$6,31 \cdot 10^{-8}$
3	$5,30 \cdot 10^{-2}$	$5,21 \cdot 10^{-4}$	$5,31 \cdot 10^{-6}$	$5,69 \cdot 10^{-8}$
4	$5,29 \cdot 10^{-2}$	$5,02 \cdot 10^{-4}$	$5,03 \cdot 10^{-6}$	$5,42 \cdot 10^{-8}$
5	$5,30 \cdot 10^{-2}$	$5,01 \cdot 10^{-4}$	$4,81 \cdot 10^{-6}$	$5,21 \cdot 10^{-8}$
6	$5,30 \cdot 10^{-2}$	$4,99 \cdot 10^{-4}$	$4,39 \cdot 10^{-6}$	$4,90 \cdot 10^{-8}$

Таблица 9 – Значение p_D при разном количестве итераций декодирования мягкого решения детектора

Число итераций декодирования	$p_B=1 \cdot 10^{-1}$	$p_B=1 \cdot 10^{-2}$	$p_B=1 \cdot 10^{-3}$	$p_B=1 \cdot 10^{-4}$
1	$4,74 \cdot 10^{-2}$	$7,10 \cdot 10^{-4}$	$1,31 \cdot 10^{-5}$	$2,95 \cdot 10^{-7}$
2	$2,93 \cdot 10^{-2}$	$8,84 \cdot 10^{-5}$	$2,87 \cdot 10^{-7}$	$6,80 \cdot 10^{-10}$
3	$2,55 \cdot 10^{-2}$	$5,47 \cdot 10^{-5}$	$1,40 \cdot 10^{-7}$	$3,79 \cdot 10^{-10}$
4	$2,43 \cdot 10^{-2}$	$4,11 \cdot 10^{-5}$	$1,08 \cdot 10^{-7}$	$2,51 \cdot 10^{-10}$
5	$2,44 \cdot 10^{-2}$	$4,18 \cdot 10^{-5}$	$1,26 \cdot 10^{-7}$	$2,33 \cdot 10^{-10}$
6	$2,42 \cdot 10^{-2}$	$4,48 \cdot 10^{-5}$	$1,18 \cdot 10^{-7}$	$2,24 \cdot 10^{-10}$

С учетом того, что код Хемминга обладает минимальной корректирующей способностью из всех СБК, можно предположить, что эффективность турбокодов на основе блочных кодов Голя и БЧХ будет значительно выше, при этом следует учесть значительное возрастание сложности синдромной решетки, необходимой для декодирования в соответствии с алгоритмом MAP. Хранение в памяти нерегулярной структуры синдромной решетки СБК требует более значительных объемов памяти, чем хранение регулярной структуры решетки ССК, однако в связи с постоянным ростом объемов хранилищ памяти во всех цифровых устройствах этот недостаток становится все менее значительным.

Выводы

Созданный на базе простейшего кода Хемминга турбокод, в сравнении с ТССК при одинаковой длине кодового слова, имеет такую же корректирующую способность, как для случая декодирования жесткого решения детектора, так и для случая декодирования мягкого решения детектора. Предполагается рост эффективности турбокодов, созданных на основе блочных кодов БЧХ и Голя, при их декодировании с использованием синдромной решетки по максимуму апостериорной информации. Параметры ТСБК в широких пределах можно адаптировать к характеристикам канала передачи данных. Избыточность кода можно изменять путем выкалывания кодовых бит в количестве не более трех бит для декодирования жесткого решения детектора, и не более пяти бит для случая декодирования мягкого решения детектора. Имеется возможность регулировки быстродействия декодера ТСБК путем изменения числа итераций декодирования.

Приложение 1.

```

Программа на языке Python, реализующая генерацию массива информации
infmassiv=[0 for val in range(0, loc)]# массив информации размера loc обнуллим
from scipy import stats as st
chance = 0.5 # вероятность генерации «1»
iter = 1 # количество итераций
count = st.rv_discrete(values=([0,1], [1-chance, chance]))
sample = [count.rvs(size=loc) for val in range(0,iter)]
for value in sample:
    infmassiv=value # присваиваем значение очередному элементу массива
    
```

Приложение 2.

```
Программа на языке Python, реализующая генерацию массива ошибок  
errormassiv=[0 for val in range(0, loc)]# массив ошибок errormassiv размера loc обнуллим  
from scipy import stats as st  
chancerr = 0.1 # вероятность генерации «1»  
iter = 1 # количество итераций  
count = st.rv_discrete(values=([0,1], [1-chancerr, chancerr]))  
sample = [count.rvs(size=loc) for val in range(0,iter)]  
for value in sample:  
    errormassiv=value # присваиваем значение очередному элементу массива
```

Приложение 3.

```
Программа на языке Python, реализующая генерацию массива уровней  
level=[0 for val in range(0, dlin)]# массив информации level размера dlin обнуллим  
from scipy import stats as st  
iter = 1 # количество итераций  
count = st.rv_discrete(values=([0,1,2,3,4,5,6,7],[0.00069,0.0045,0.02211,0.0727,0.1608,0.2392,0.239,  
0.261])  
sample = [count.rvs(size= dlin) for val in range(0,iter)]  
for value in sample:  
    level =value # присваиваем значение очередному элементу массива
```

Литература

1. Аджемов А. С., Санников В. Г. Общая теория связи: учебник для вузов. – М.: Горячая линия – Телеком, 2023. – 624 с.
2. Васильев К. К., Глушков В. А., Нестеренко А. Г. Теория электрической связи. – Вологда: Инфра-Инженерия, 2021. – 468 с.
3. Кудряшев Б. Д. Основы теории кодирования: учебное пособие. – СПб.: БХВ-Петербург, 2016. – 400 с.
4. Золоторев В. В. Кодирование для цифровой связи и систем памяти: монография. – М.: Горячая линия – Телеком, 2023. – 192 с.
5. Ключев Л. Л. Теория электрической связи: учебник. – М.: ИНФРА-М, 2019. – 448 с.
6. Варгузин В. А., Цикин И. А. Методы повышения энергетической и спектральной эффективности цифровой радиосвязи. – СПб.: БХВ-Петербург, 2013. – 352 с.
7. Голиков А. М. Модуляция, кодирование и моделирование в телекоммуникационных системах. Теория и практика: учебное пособие для вузов. – СПб.: Лань, 2022. – 452 с.
8. Сидоренко А. А. Декодирование турбокода, созданного на основе блочного кода, с использованием синдромной решетки // Проектирование и технология электронных средств. 2022. № 4. С. 46-52.
9. Харкевич А. А. Борьба с помехами. – М.: Ленанд, 2018. – 280 с.
10. Скляр Б. Цифровая связь. Теоретические основы и практическое применение. Пер с англ. – М.: Вильямс, 2016. – 1104 с.

11. Сидоренко А. А. Программный тренажер исследования эффективности турбокода, построенного на основе кода Хемминга // Свидетельство о государственной регистрации программы для ЭВМ. № 2023614197. 2023.
12. Березкин Е. Ф. Основы теории кодирования: учебное пособие. – СПб.: Лань, 2019. – 320 с.
13. Морелос-Сарагоса Р. Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. Пер с англ. – М.: Техносфера, 2006. – 319 с.
14. Кольцов Д. М. Python. Полное руководство. – М.: Наука и техника, 2022. – 480 с.
15. Нуньес-Иглесиас Х., Уолт Ш., Дэшноу Х. Элегантный SciPy. Пер с англ. – ДМК Пресс, 2018. – 266 с.
16. Сидоренко А. А. Разработка программной эмуляция мягкого решения на выходе детектора, для сравнительного исследования эффективности декодеров, работающих с мягким и жестким решением детектора // Проектирование и технология электронных средств. 2018. № 3. С. 44-50.

References

1. Ajemov A. S., Sannikov V. G. *General theory of communication*. Moscow, Hotline – Telecom Publ., 2023. 624 p. (in Russian).
2. Vasiliev K. K., Glushkov V. A., Nesterenko A. G. *Theory of electrical communication*. Vologda, Infra-Engineering Publ., 2021. 468 p. (in Russian).
3. Kudryashev B. D. *Fundamentals of coding theory*. Saint-Petersburg, BHV-Petersburg Publ., 2016. 400 p. (in Russian).
4. Zolotarev V. V. *Coding for digital communication and memory systems*. Moscow, Hotline – Telecom Publ., 2023. 192 p. (in Russian).
5. Klyuev L. L. *Theory of electrical communication*. Moscow, INFRA-M Publ., 2019. 448 p. (in Russian).
6. Varguzin V. A., Tsikin I. A. *Methods for improving the energy and spectral efficiency of digital radio communications*. Saint-Petersburg, BHV-Petersburg Publ., 2013. 352 p. (in Russian).
7. Golikov A. M. *Modulation, coding and modeling in telecommunication systems. Theory and practice*. Saint-Petersburg, Lan Publ., 2022. 452 p. (in Russian).
8. Sidorenko A. A. Decoding of a turbo code based on a block code using a syndrome grid. *Design and technology of electronic means*, 2022, no. 4, pp. 46–52 (in Russia).
9. Harkevich A. A. *Anti-interference*. Moscow, Lenand Publ., 2018. 280 p. (in Russian).
10. Sklyar B. *Digital communication. Theoretical foundations and practical application*. New Jersey, Prentice Hall, 2016. 1104 p.
11. Sidorenko A. A. *Programmnyi trenager issledovaniya effektivnosti turbokoda, postroennogo na osnove koda Hemminga* [Program simulator for the study of the efficiency of a turbo code built on the basis of a Hemming code]. The

Certificate on Official Registration of the Computer Program in Russia. No. 2023614197, 2023.

12. Berezkin E. F. *Fundamentals of coding theory*. Saint-Petersburg, Lan Publ., 2019. 320 p. (in Russian).

13. Morelos-Zaragoza R. *The art of noise-resistant coding. Methods, algorithms, application*. Chichester, Jon Wiley and Sons, 2006. 319 p.

14. Koltsov D. M. *Python. Complete guide*. Moscow, Science and Technology Publ., 2022. 480 p. (in Russian).

15. Nunez-Iglesias H., Walt S., Dashnow H. *Elegant SciPy*. Sebastopol, O'Reilly Media, 2018. 266 p.

16. Sidorenko A. A. Development of software emulation of a soft solution at the output of the detector, for a comparative study of the efficiency of decoders working with a soft and hard detector solution. *Design and technology of electronic means*, 2018, no. 3, pp. 44–50 (in Russia).

Статья поступила 13 апреля 2023 г.

Информация об авторе

Сидоренко Александр Анатольевич – кандидат технических наук. Эксперт отделения погранологии. Международная академия информатизации. Область научных интересов: системы, сети и устройства телекоммуникаций. E-mail: kotofeykin@mail.ru

Адрес: 125284, Россия, г. Москва, Ленинградский пр., д. 3.

Evaluation of the efficiency of a turbo code built on the basis of a block code when decoding in accordance with the criterion of maximum a posteriori probability

A. A. Sidorenko

Problem statement: correction codes are now widely used to correct errors that occur when transmitting information via communication channels. In data transmission channels with a high probability of error, turbo codes are the most effective. When constructing turbo codes based on block and convolutional codes, various decoding algorithms are used. When decoding turbo codes based on a block code using a syndrome grid, it is possible to use decoding algorithms used to decode turbo codes based on convolutional codes. **The aim of the work** is to evaluate the efficiency of a turbo code built on the basis of a block code when decoding in accordance with the MAP algorithm, previously used for decoding turbo codes built on the basis of convolutional codes. It is proposed to evaluate the correcting ability of the turbo code, and its ability to adapt to changes in the characteristics of the data transmission channel. **Methods used:** turbo code decoding was carried out using the MAP algorithm. The decoding process is modeled in MS Excel and in the Python programming language. In the program text, random number arrays were generated using the «scipy» library and its «stats» sub-package. When deriving the formula for calculating the probability of an erroneous bit appearing in a decoded information message, the provisions of probability theory were used. The provisions of probability theory were also used to form a soft detector output. **Novelty:** both the use of the MAP algorithm itself for decoding the turbo code created on the basis of the block code, and the obtained results of the study of the correcting ability of the turbo code and the possibility of adapting its parameters to changes in the characteristics of the channel, have elements of novelty. **Result:** the conducted

*study of the corrective ability showed the same efficiency of the studied turbo code when compared with a turbo code created on the basis of a systematic convolutional code with the same number of information and code bits. The simulation showed the possibility of adapting the parameters of the turbo code under study to changes in the characteristics of the data transmission channel by splitting out code bits in the amount of no more than three bits to decode the hard solution of the detector, and no more than five bits for the case of decoding the soft solution of the detector. The possibility of adapting turbo code parameters by changing the number of decoding iterations is also investigated. **Practical significance:** the presented studies have shown good efficiency of the turbo code created on the basis of block code, using the well-studied MAP algorithm when decoding, as well as the available capabilities of such a turbo code to adapt to changes in the characteristics of the data transmission channel. In order to study the effectiveness of the turbo code built on the basis of the Hemming code, a software simulator in the Python programming language was developed.*

Key words: error correction code, turbo code, MAP algorithm, block codes, convolutional codes syndrome grid.

Information about Author

Aleksandr Anatolevich Sidorenko – Ph.D. of Technical Sciences. Expert of the Department of Foreign Studies. International Academy of Informatization. Field of research: telecommunication systems, networks and devices. E-mail: kotofeykin@mail.ru

Address: Russia, 125284, Moscow, Leningradsky prospekt, 3.