

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 621.391

АБРАМОВ
Игорь Олегович

**ДЕКОДИРОВАНИЕ КОДОВ С МАЛОЙ ПЛОТНОСТЬЮ ПРОВЕРОК
НА ЧЕТНОСТЬ**

Автореферат
на соискание степени магистра
по специальности 1–45 80 01 Системы и сети инфокоммуникаций

Научный руководитель
к.т.н., доцент
Астровский Иван Иванович

Минск 2023

ВВЕДЕНИЕ

Работа относится к теории и технике помехоустойчивого кодирования. Рассматриваются пути повышения эффективности практической реализации декодеров кодов с малой плотностью проверок на четность (LDPC от англ. Low-density parity-check). Разрабатываются и исследуются варианты реализации LDPC декодеров, обеспечивающих наилучшие показатели по критерию помехоустойчивость-сложность технической реализации. Результаты исследований апробируются на примере спутниковой телекоммуникационной подсистемы передачи эфемеридной и служебной информации наземным потребителям с использованием сигнала L1С.

Целью работы является разработка и исследование алгоритмов декодирования LDPC

кодов. Для достижения поставленной цели решаются следующие задачи:

1. Анализ существующих алгоритмов декодирования LDPC кодов.
2. Оценка вычислительной сложности декодирования LDPC кодов.
3. Оценка статистических характеристик декодирования (BER, число итераций, сходимость синдрома) LDPC кодов.
4. Разработка модификаций и методов, позволяющих повысить вычислительную эффективность декодирования и сэкономить используемые при декодировании ресурсы памяти.

В современных телекоммуникационных системах большое внимание уделяется помехозащищенности передаваемой информации. Помехозащищенность обеспечивается за счет применения помехоустойчивого кодирования информации. Примерно с начала 90-ых годов задачу помехоустойчивого кодирования решали с помощью турбо кодов. Однако с ростом объемов трафика и скоростей передачи информации возрос интерес к более эффективной технике помехоустойчивого кодирования – кодированию с помощью кодов с малой плотностью проверок на четность.

Коды с малой плотностью проверок на четность обладают эффективными алгоритмами декодирования, позволяющими быстро и надежно корректировать поврежденную при передаче в результате шумов информацию. Данные коды позволяют осуществлять работу цифровой линии связи при отношении сигнал/шум, близких к границе Шеннона, опережая по эффективности коррекции ошибок турбо коды на относительно больших длинах кодовых слов.

Коды с малой плотностью проверок на четность рекомендованы для коррекции ошибок в современных стандартах связи DVB-S2, DVB-C2, Wi-Fi

802.11n, WiMAX 802.16e.

На данный момент можно выделить два направления исследований в области кодов с малой плотностью проверок на четность. Первое из них относится к синтезу конструкций LDPC кодов. В этом направлении следует отметить труды Афанасьева В.Б., Воробьева К.А., Зигангирова Д.К., Зигангирова К.Ш., Зяблова В.В., Иванова Ф.И., Крука Е.А., Овчинникова А.А., Пацей Н.В., Потапова В.Г., Трухачева Д.В., Costello D., Johnson S.J., Kou Y., Luby M.G., Richardson J., Weller S.R.

Второе направление исследует алгоритмическую составляющую LDPC кодеков. В этом направлении следует отметить труды Башкирова А.В., Белоголового А.В., Витязева В.В., Владимирова С.М., Климова А.И., Козлова А.В., Кравченко А.Н., Лихобабина Е.А., Муратова А.В., Овечкина Г.В., Проскурина А.А., Солтанова А.Г., Chen J., Fossorier M., Kim N., Urbanke R.L.

Общий вклад в развитие теории внесли труды Акулинина А.С., Золотарева В.В., Зубарева Ю.Б., Колесника В.Д., Eckford A.W., MacKay D., Tanner M.

Последние достижения в области кодирования кодами с малой плотностью проверок на четность позволили практически вплотную приблизиться к границе Шеннона. Как следствие, актуальной задачей на сегодня является не увеличение исправляющей способности, а разработка методики выбора алгоритма декодирования, обеспечивающего наилучший компромисс по определенным критериям качества в рамках рассматриваемой системы связи, и модификация существующих алгоритмов с целью повышения вычислительной эффективности декодирования и экономии используемых ресурсов памяти.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Связь работы с крупными научными программами

На данный момент можно выделить два направления исследований в области кодов с малой плотностью проверок на четность. Первое из них относится к синтезу конструкций LDPC кодов. В этом направлении следует отметить труды Афанасьева В.Б., Воробьева К.А., Зигангирова Д.К., Зигангирова К.Ш., Зяблова В.В., Иванова Ф.И., Крука Е.А., Овчинникова А.А., Пацей Н.В., Потапова В.Г., Трухачева Д.В., Costello D., Johnson S.J., Kou Y., Luby M.G., Richardson J., Weller S.R.

Второе направление исследует алгоритмическую составляющую LDPC кодеков. В этом направлении следует отметить труды Башкирова А.В., Белоголового А.В., Витязева В.В., Владимирова С.М., Климова А.И., Козлова

А.В., Кравченко А.Н., Лихобабина Е.А., Муратова А.В., Овечкина Г.В., Проскурина А.А., Солтанова А.Г., Chen J., Fossorier M., Kim N., Urbanke R.L.

Общий вклад в развитие теории внесли труды Акулинина А.С., Золотарева В.В., Зубарева Ю.Б., Колесника В.Д., Eckford A.W., MacKay D., Tanner M.

Последние достижения в области кодирования кодами с малой плотностью проверок на четность позволили практически вплотную приблизиться к границе Шеннона. Как следствие, актуальной задачей на сегодня является не увеличение исправляющей способности, а разработка методики выбора алгоритма декодирования, обеспечивающего наилучший компромисс по определенным критериям качества в рамках рассматриваемой системы связи, и модификация существующих алгоритмов с целью повышения вычислительной эффективности декодирования и экономии используемых ресурсов памяти.

Цель и задачи исследования

Целью работы является разработка и исследование алгоритмов декодирования LDPC

кодов. Для достижения поставленной цели решаются следующие задачи:

1. Анализ существующих алгоритмов декодирования LDPC кодов.
2. Оценка вычислительной сложности декодирования LDPC кодов.
3. Оценка статистических характеристик декодирования (BER, число итераций, сходимость синдрома) LDPC кодов.
4. Разработка модификаций и методов, позволяющих повысить вычислительную эффективность декодирования и сэкономить используемые при декодировании ресурсы памяти.

Методы исследования

В работе использовался аппарат теории вероятностей, теории электрической связи, дискретной математики и математического анализа.

Для проведения моделирования использовалась среда имитационного моделирования MATLAB с пакетом Simulink и среда разработки программных продуктов Microsoft Visual Studio 2010.

Научная новизна

1. Получены и проанализированы соотношения для расчета сложности итерации декодирования LDPC кодов для различных алгоритмов коррекции ошибок.

2. Получены и исследованы статистические характеристики декодирования (BER, число итераций, сходимость синдрома) для различных алгоритмов коррекции ошибок в рамках рассматриваемого LDPC кода.

3. Предложена методика выбора алгоритма декодирования, обеспечивающего заданную вероятность ошибки при наименьшей сложности декодирования.

4. Предложена методика компактного представления разреженной проверочной матрицы LDPC кода, позволяющая экономить ресурсы памяти для ее хранения.

5. Предложены модификации алгоритмов, позволяющие повысить вычислительную эффективность декодирования без потери исправляющей способности при незначительном увеличении требований к памяти для хранения внутренних переменных декодера.

6. Предложен способ идентификации инверсии битового потока за счет внутренних ресурсов LDPC декодера и исследована его работа на реальном сигнале.

Практическая ценность диссертационной работы

Предложенная методика выбора алгоритма декодирования LDPC может использоваться при проектировании современных цифровых телекоммуникационных систем.

Компактное представление проверочной матрицы позволяет сэкономить в 2 раза ресурсы памяти, выделенные на ее хранение.

Применение разработанных модификаций к алгоритмам коррекции ошибок позволяет без потери в качестве декодирования повысить скорость декодирования в 3 раза при незначительном увеличении требований к памяти для хранения внутренних переменных декодера.

Предложенный способ идентификации инверсии битового потока на входе LDPC декодера может применяться в системах связи, не имеющих в принимаемом сигнале фиксированную преамбулу для решения этой задачи.

Разработанные программные реализации LDPC и BCH кодеров могут быть внедрены в системы связи, использующие соответствующее помехоустойчивое кодирование информации.

Достоверность

Достоверность полученных результатов обусловлена сопоставлением результатов моделирования на имитационной модели с теорией, а также экспериментами других авторов и результатами декодирования и расшифровки реального сигнала.

Положения, выносимые на защиту

1. Предложенная методика выбора алгоритма декодирования LDPC кодов позволяет определить алгоритм, обеспечивающий заданную вероятность ошибки на выходе декодера при наименьшей сложности декодирования.
2. Предложенная методика представления разряженной проверочной матрицы LDPC кода позволяет уменьшить в 2 раза требуемые ресурсы памяти, предназначенные для ее хранения.
3. Разработанные модификации алгоритмов декодирования LDPC кодов позволяют повысить скорость работы декодера в 3 раза при незначительном увеличении требований к памяти для хранения внутренних переменных декодера.
4. Предложенный способ идентификации инверсии битового потока позволяет определять инверсию за счет внутренних ресурсов LDPC декодера.

Личный вклад автора

Все результаты, полученные в данной работе, являются личными достижениями автора.

Структура и объем диссертации

Работа состоит из введения, пяти разделов, заключения, содержит 77 рисунков, 21 таблицу и 151 формулу. Работа размещена на 131 страницах.

Проверка на уникальность

Проведена экспертиза диссертации *Абрамова Игоря Олеговича «ДЕКОДИРОВАНИЕ КОДОВ С МАЛОЙ ПЛОТНОСТЬЮ ПРОВЕРОК НА ЧЕТНОСТЬ»* на корректность использования заимствованных материалов с применением сетевого ресурса «Антиплагиат» (адрес доступа: <https://antiplagiat.ru>) в on-line режиме 27.03.2023 г. В результате проверки установлена корректность использования заимствованных материалов (оригинальность диссертационной работы составляет 85,1%).

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во введении дана общая характеристика работы, отмечена ее актуальность, степень проработанности темы и сформулирована цель. Обоснована практическая ценность проводимой работы, а также отмечена научная новизна и сформулированы положения, выносимые на защиту.

В первой разделе поставлена задача декодирования сигнала LDPC. Введены основные понятия и локализованы существующие алгоритмы декодирования кодов с малой плотностью проверок на четность. Приведено их описание и группировка по некоторым признакам. Отмечены плюсы и минусы алгоритмов декодирования. Приведены существующие результаты исследований других авторов.

Любой линейный блочный код, в том числе и с малой плотностью проверок на четность, можно описать порождающей и проверочной матрицей. Проверочная матрица состоит из «0» и «1» и представляет собой систему проверочных уравнений. В случае LDPC кодов матрица проверки на четность будет содержать малое число «1». Пример такой матрицы приведен на рисунке 1.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
1	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0
2	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0
3	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0
4	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0
5	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	1
6	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0
7	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
8	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0
9	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0
10	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1
11	0	1	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	1
12	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0
13	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	0	1	0	0	0
14	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0
15	1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	1	0	0

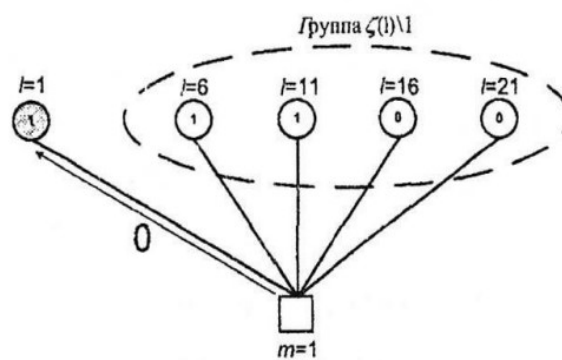


Рисунок 1 - Пример проверочной матрицы LDPC кода

Рисунок 2 - Фрагмент графа Таннера для первой проверки на четность

В данном случае матрица содержит 15 проверочных уравнений и каждый из 25 принятых символов кодового слова участвует в трех проверках на четность. Число «1» пришедшего кодового слова в рамках проверки на четность должно быть четным. Невыполнение проверки на четность свидетельствует о наличии ошибок в принятом кодовом слове.

При описании алгоритмов декодирования кодов с малой плотностью проверок на четность удобно переходить от матрицы к двудольному графу, с одной стороны которого находятся символьные узлы i (variable nodes), а с другой стороны проверочные узлы m (check nodes). Количество символьных узлов соответствует количеству символов принятого кодового слова, а количество проверочных узлов соответствует количеству проверочных уравнений в матрице проверки на четность. Символьные и проверочные узлы

соединяются ребрами в соответствии с расположением «1» в матрице. Фрагмент графа для первой проверки на четность приведен на рисунке 2.

В соответствии с первой строкой матрицы на рисунке 1, в первой проверке на четность $m=1$ участвуют символы пришедшего кодового слова с номерами 1, 6, 11, 16 и 21. Предположим, что каждому из этих символов кодового слова были приписаны «жесткие» решения, приведенные в окружностях, и первый символ был принят с ошибкой. В этом случае проверка на четность не будет выполняться. Каждому символу в рамках рассматриваемой проверки на четность выносится поправка. Например, для расчета поправки к первому символу необходимо рассчитать количество «1» в проверке на четность $m=1$ без учета первого символа. Для этого формируется локализованная группа символьных узлов $4 \times 1 \setminus 1$ (выделена пунктиром). Число в скобках показывает номер проверки на четность, а число после черты показывает номер символа, которому рассчитывается поправка относительно локализованной группы. В конкретном примере в группе $f(1) \setminus 1$ две «1». Следовательно, для удовлетворения проверки на четность первый символ должен быть «0», о чем проверочный узел $m=1$ «сообщает» первому символу.

В работе большинства алгоритмов декодирования LDPC кодов лежит принцип, описанный выше. Отличие заключается лишь в том, что работа ведется не с «жесткими» решениями «1» и «0», а с «мягкими» решениями (надежности принятых символов). В этом случае поправка к априорному «мягкому» решению демодулятора, рассчитанная относительно локализованной группы, будет представлять собой вещественное число.

Алгоритмы декодирования LDPC кодов представлены на рисунке 3 и отличаются друг от друга способом расчета поправок к априорным «мягким» решениям демодулятора. Базовым алгоритмом декодирования является алгоритм с распространением доверия «Belief propagation». Его недостатком является высокая сложность в силу необходимости использования гиперболических функций тангенса и арктангенса при расчете поправок. На практике используют упрощенные алгоритмы декодирования из семейств «Min-sum» и «UMP BP» или различные аппроксимации гиперболических функций.

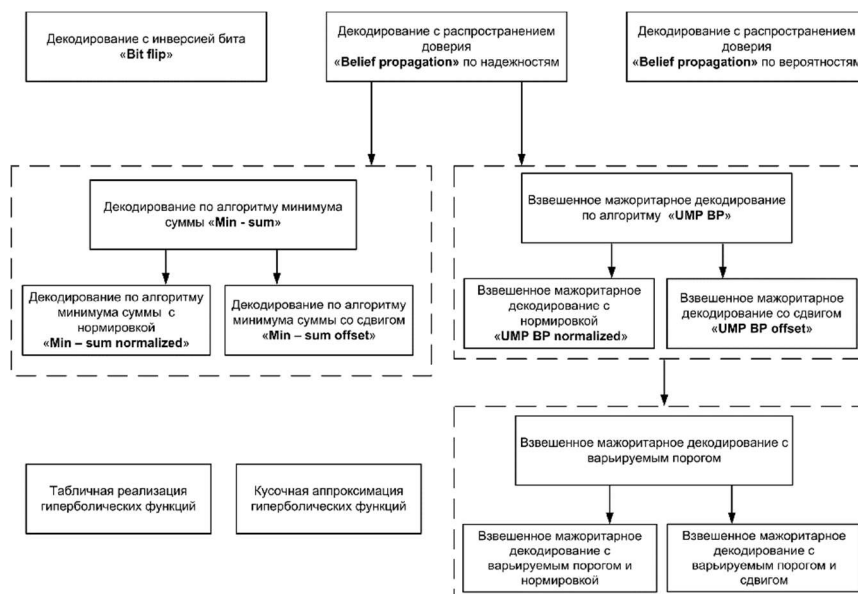


Рисунок 3 - Алгоритмы декодирования LDPC кодов

В данной работе LDPC коды исследуются на примере спутниковой телекоммуникационной подсистемы передачи эфемеридной и служебной информации наземным потребителям с использованием сигнала L1C. Структура кадра сигнала L1C на входе декодера представлена на рисунке 4.

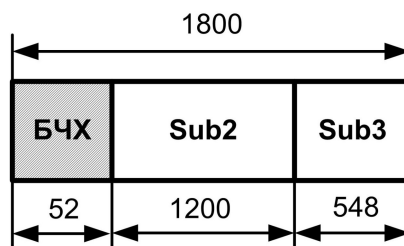


Рисунок 4 - Структура кадра сигнала L1C на входе декодера

Кадр состоит из 1800 отсчетов и делится на 3 части. Первая часть представляет собой закодированный БЧХ кодом номер кадра L1C. Информация в части Sub2 (Subframe2) закодирована LDPC кодом длиной 1200 символов. Информация в части Sub3 (Subframe3) закодирована LDPC кодом длиной 548 символов. Матрицы проверки на четность, соответствующие кодам из частей Sub2 и Sub3, представлены на рисунке 5. Точки соответствуют расположению «1».

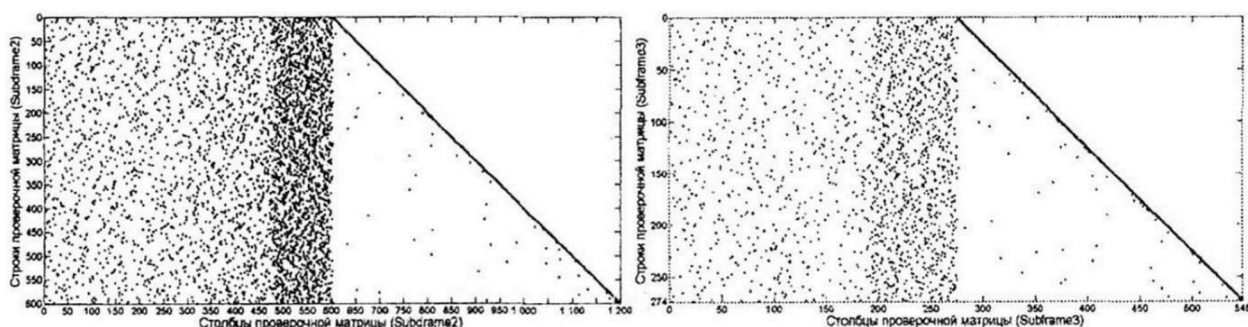


Рисунок 5 - Структуры матриц проверки на четность из Sub2 и Sub3

При анализе и реализации алгоритмов декодирования было выявлено, что существующие алгоритмы слишком «тяжелы» с точки зрения вычислительной сложности и используемых ресурсов памяти для платформы, на базе которой реализуется декодер. Целью работы является разработка и исследование алгоритмов декодирования LDPC кодов для их дальнейшего применения при декодировании составных частей сигнала L1C на базе выбранной платформы. Для достижения цели в работе анализируются существующие алгоритмы декодирования, оцениваются их сложность и статистические характеристики декодирования, предлагается методика выбора алгоритма декодирования, а также разрабатываются методы, позволяющие повысить вычислительную эффективность декодирования и сэкономить используемые при декодировании ресурсы памяти.

Предлагаемая методика выбора алгоритма декодирования заключается в локализации алгоритмов декодирования, обеспечивающих требование по вероятности ошибки на выходе декодера при фиксированном отношении сигнал/шум и определении среди локализованных алгоритмов того, который обеспечивает это требование с минимальной сложностью. Как следствие, необходимо иметь реестр всех существующих алгоритмов декодирования LDPC (Раздел 1), из которых будет осуществляться выбор, а также оценить сложность (Раздел 2) и эффективность (Раздел 3) декодирования.

Во втором разделе проведена оценка сложности декодирования LDPC кодов. Получены соотношения для оценки вычислительной сложности для случая регулярных и нерегулярных матриц для различных алгоритмов декодирования. На основе анализа работы алгоритмов декодирования разработаны две модификации, позволяющие использовать меньшее число элементарных операций без потери в качестве декодирования при незначительном увеличении требований к памяти для хранения внутренних переменных декодера.

Для оценки сложности алгоритмов декодирования в качестве базовой используется методика, учитывающая операции сложения ($\{N_+\}$), умножения (A^\wedge), сравнения ($N_,$), взятия модуля числа ($M|$) и сложения по модулю 2 (A^\otimes) на одну итерацию декодирования.

Операция вычитания приравнивается к операции сложения. Число операций различного типа будет зависеть от «весов» строк (k) и столбцов (l) проверочной матрицы, а также от длины кодового слова (l), количества уравнений в матрице проверки на четность (t) и алгоритма декодирования. Работу любого алгоритма декодирования LDPC можно разделить на несколько этапов, представленных на рисунке 6. Подсчет числа элементарных операций будет вестись для этапов 1-4, выполняемых итеративно.

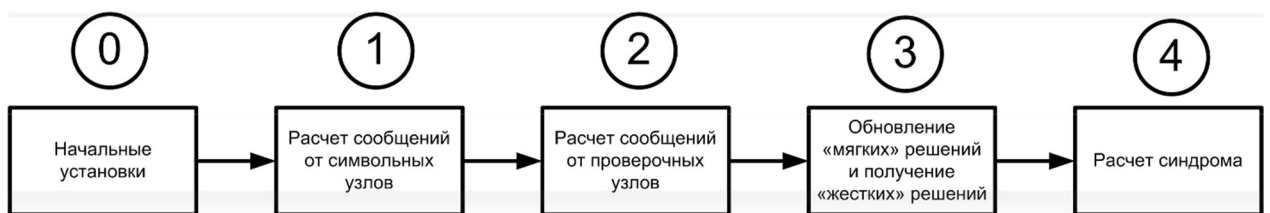


Рисунок 6 - Этапы декодирования LDPC кода

В работе получены аналитические соотношения, позволяющие оценить вычислительную сложность для различных алгоритмов декодирования LDPC кодов (число операций различного типа на итерацию декодирования) для любой матрицы проверки на четность. По полученным соотношениям проведен расчет суммарной вычислительной сложности алгоритмов декодирования в рамках LDPC кода, применяемого в Subframe2. Под суммарной сложностью в данной работе понимается сумма операций различного типа, выполняемых декодером за одну итерацию декодирования. Приведенные значения на рисунке 7 получены при суммировании различных типов операций с «весом» 1. В общем случае суммирование различных типов операций необходимо производить с разными «весами», которые будут зависеть от аппаратуры, на которой реализуется декодирование.

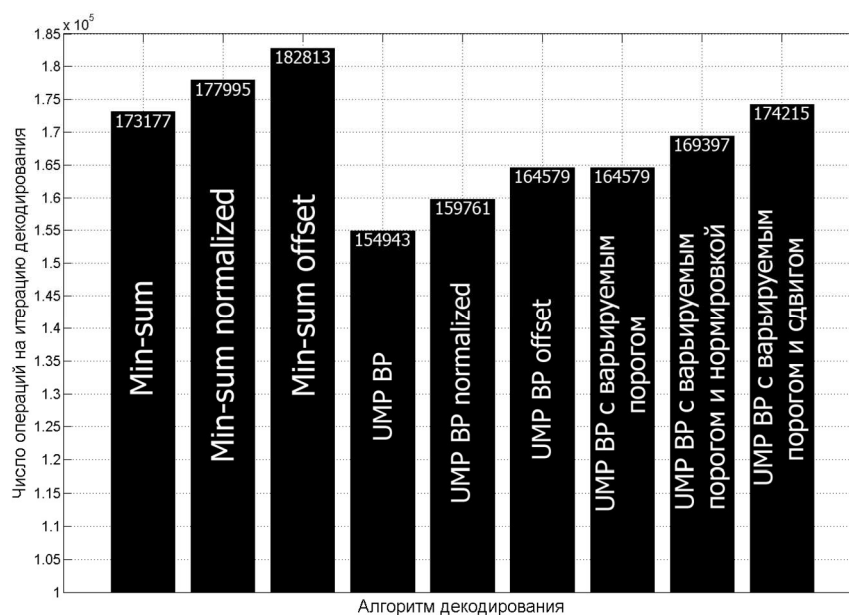


Рисунок 7 — Суммарная вычислительная сложность декодирования LDPC кода в Subframe2

Следует отметить, что суммарную сложность алгоритма декодирования необходимо учитывать в комплексе со средним числом итераций, которое использует декодер при работе по этому алгоритму декодирования для обеспечения заданных требований по BER.

Алгоритмы декодирования из семейств «Min-sum» и «UMP BP» могут быть реализованы таким образом, что их вычислительная сложность по сравнению с представленной на рисунке 7 будет значительно ниже. Предложенная модификация декодирования основана на том, что поправки могут быть рассчитаны сразу для всех символов, входящих в соответствующую проверку на четность m . Для этого достаточно определить два минимальных абсолютных значения среди сообщений, пришедших от символьных узлов l в проверочный узел m , и разослать эти сообщения в качестве поправок к априорным «мягким» решениям демодулятора с учетом знака.

Сравнение вычислительной сложности классического и модернизированного алгоритмов «Min-sum» (MS) приведено в таблице 1. При реализации выигрыш по скорости работы декодера составил в 3 раза.

Таблица 1 - Сравнение вычислительной сложности для MS

Операция	Количество на итерацию декодирования	
	Классический MS	Модернизированный MS

1	2	3
Сложение	37024	37024
Умножение	33888	10236
Сравнение	64159	13855
Взятие модуля	33888	4818
Сложение по модулю 2	4218	4218

Сравнение вычислительной сложности классического и модернизированного алгоритма «UMP BP» приведено в таблице 2.

Таблица 2 - Сравнение вычислительной сложности для «UMP BP»

Операция	Количество на итерацию декодирования	
	Классический «UMP BP»	Модернизированный «UMP BP»
Сложение	37024	37024
Сравнение	39907	18673
Взятие модуля	33888	4818
Сложение по модулю 2	44124	24090

Следует подчеркнуть, что платой за снижение вычислительной сложности классических алгоритмов декодирования «Min-sum» и «UMP BP» является незначительное увеличение памяти для хранения внутренних переменных декодера. Помехоустойчивость алгоритмов в результате модификаций не изменяется.

В третьем разделе на имитационной модели получены и проанализированы статистические характеристики декодирования кодов с малой плотностью проверок на четность на примере одного из двух кодов, используемых для кодирования информации, которую несет сигнал L1C.

Обратимся к схематично изображенному на рисунке 8 примеру низкоплотностной матрицы проверки на четность размерностью 600 на 1200, содержащей 4818 ненулевых элементов. Здесь поля с цифрами соответствуют ненулевым элементам, а без цифр - нулевым.

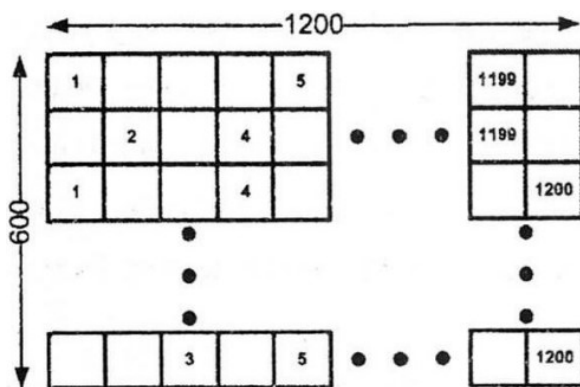


Рисунок 8 - Схематичное изображение матрицы проверки на четность

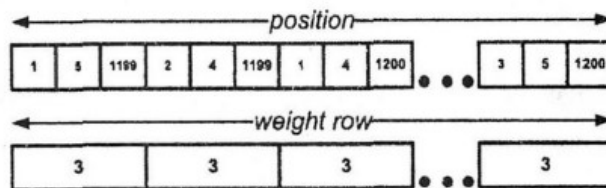


Рисунок 9 - Компактное представление матрицы проверки на четность

Низкоплотная матрица проверки на четность представляет собой двумерный массив. Чем больше длина кодового слова, тем больше матрица проверки на четность. Этот факт делает затруднительным процедуру хранения матриц в памяти декодера, особенно для больших длин кодовых слов. Кроме того, низкоплотная матрица содержит в основном нулевые элементы.

В работе предложено представлять матрицу (рисунок 9) в виде двух целочисленных одномерных массивов: массива *position* координат ненулевых элементов матрицы по горизонтали и массива *weight row* «весов» строк матрицы. Под «весом» строки понимается количество единиц в строке. При этом массив *position* будет иметь размер, равный числу ненулевых элементов матрицы, а массив *weight row* будет иметь размер, равный количеству строк матрицы. Такой подход позволяет в 2 раза сократить требуемую память для хранения матрицы по сравнению с существующими методами, в которых для представления используются координаты ненулевых элементов матрицы по вертикали и горизонтали.

Целью моделирования являлось получение ряда статистических характеристик низкоплотного декодера (BER, число итераций, сходимость синдрома). Моделирование проводилось для LDPC кода из части Subframe2. Структурная схема имитационной модели односторонней линии цифровой радиосвязи приведена на рисунке 10.



Рисунок 10 - Структурная схема имитационной модели

Для каждого из алгоритмов декодирования был разработан C++ код, внедряемый в сегмент декодера имитационной модели. Для различных алгоритмов декодирования были получены кривые помехоустойчивости BER (рисунок 11), зависимости среднего числа использованных декодером итераций от отношения сигнал/шум (рисунок 12) и характеристики сходимости среднего «веса» синдрома от номера итерации при битовом отношении сигнал/шум 3 дБ (рисунок 13 и рисунок 14).

Алгоритм «Belief propagation» демонстрирует наилучшие характеристики помехоустойчивости, наименьшее среднее число использованных итераций и наиболее быстрое схождение среднего «веса» синдрома к нулю.

Версии алгоритмов с нормировкой (normalized) и сдвигом (offset) из семейств «Min-sum» и «UMP BP», а также мажоритарные алгоритмы с варьируемым порогом и нормировкой и сдвигом проигрывают алгоритму «Belief propagation» не более -0.1 дБ.

Причем во всех случаях алгоритмы со сдвигом демонстрируют лучшие на -0.05 дБ характеристики помехоустойчивости, чем алгоритмы с нормировкой. Более того, в среднем версии алгоритмов со сдвигом требуют меньшее число итераций на декодирование, а их средние «веса» синдромов сходятся быстрее средних «весов» алгоритмов с нормировкой.

Мажоритарные алгоритмы с варьируемым порогом демонстрируют энергетический выигрыш -0.3-0.4 дБ относительно декодирования с нулевым порогом (стандартный «UMP BP»), но проигрывают кластеру из алгоритма «Belief propagation» и алгоритмов с нормировкой и сдвигом порядка -0.3-0.4 дБ, тем самым занимая промежуточный сегмент. Следует отметить, что в силу своей специфики эти алгоритмы отличаются медленным сходимостью среднего

«веса» синдрома к нулю и большим средним числом итераций для декодирования по отношению к алгоритмам, не использующим варьирование порога.

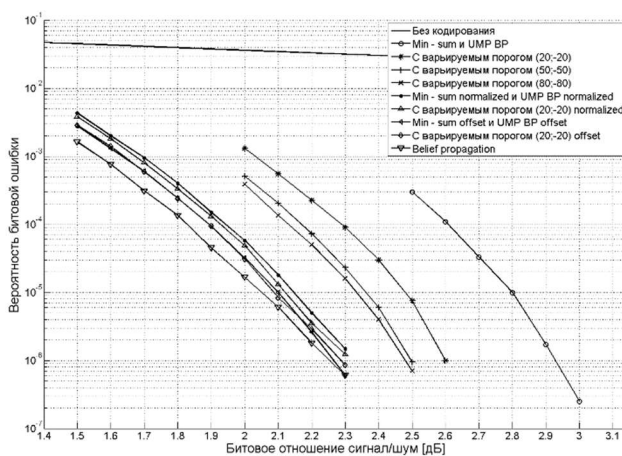


Рисунок 11 - Зависимость BER от ОСШ

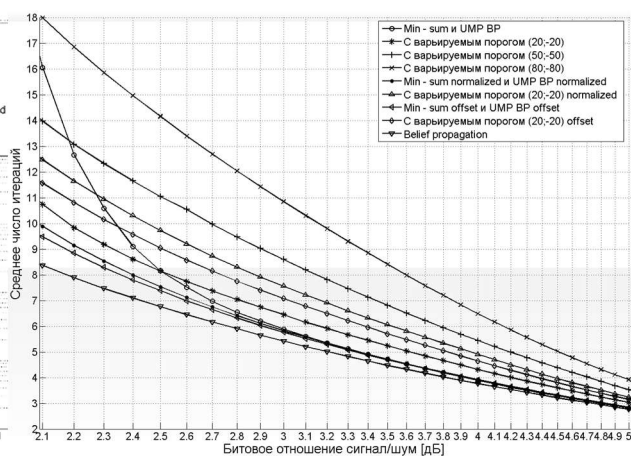


Рисунок 12 - Зависимость среднего для различных алгоритмов числа итераций от ОСШ

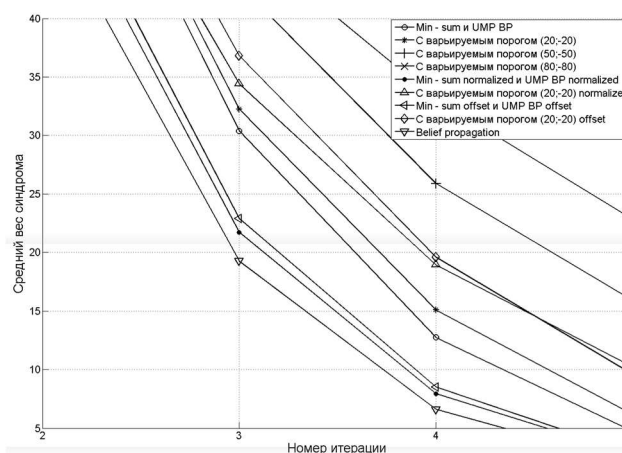


Рисунок 13 - Сходимость среднего веса синдрома (нач. участок)

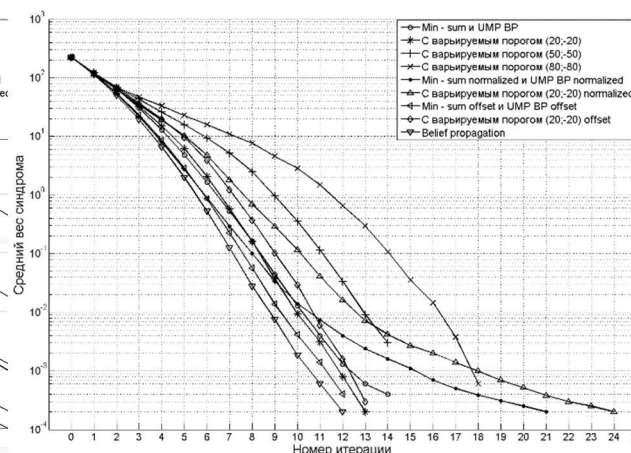


Рисунок 14 – Сходимость среднего веса синдрома

В четвертом разделе приведен пример применения предлагаемой методики выбора алгоритма декодирования LDPC кодов, а также проанализированы характеристики декодирования LDPC кодов на основе результатов обработки сигнала L1C. Кроме того, предложен способ идентификации инверсии битового потока сигнала L1C за счет внутренних ресурсов LDPC декодера.

В качестве примера задается требование обеспечения вероятности ошибки на выходе декодера 10^{-5} при битовом отношении сигнал/шум 2.2 дБ.

Согласно предлагаемой методике, вначале происходит локализация алгоритмов, которые могут обеспечить заданную вероятность ошибки при указанном отношении сигнал/шум. В конкретном случае на рисунке 11 таких алгоритмов 7: «Min-sum normalized», «UMP BP normalized», «С варьируемым порогом (20;-20) normalized», «Min-sum offset», «UMP BP offset», «С варьируемым порогом (20;-20) offset», «Belief propagation».

Как отмечалось выше, алгоритм с распространением доверия «Belief propagation» редко реализуется на практике в чистом виде и будет вычеркнут из рассмотрения. Для остальных алгоритмов необходимо рассчитать комплексный показатель сложности, учитывающий суммарную сложность декодирования на итерацию (рисунок 7) и среднее число итераций (рисунок 12), которое используется алгоритмами при фиксированном отношении сигнал/шум для обеспечения заданного BER. При учете суммарной сложности алгоритма на итерацию декодирования, как уже отмечалось выше, следует учитывать «веса» операций различного типа при их суммировании. Здесь, в качестве примера, «веса» приятны равными 1. Разработчики, используя полученные во втором разделе значения сложности по каждому типу операции, могут провести суммирование с соответствующими для их аппаратуры «весами».

Согласно представленным на рисунке 15 значениям комплексного показателя сложности, для удовлетворения сформулированного выше требования с наименьшими «затратами» следует отдать предпочтение алгоритмам «UMP BP offset» и «UMP BP normalized».

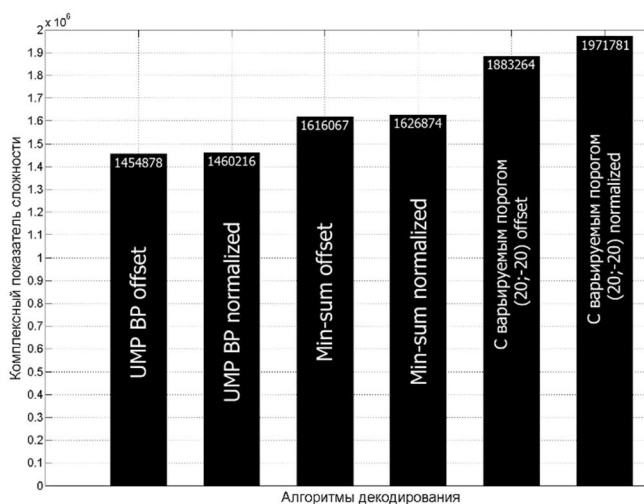


Рисунок 15 - Значения комплексных показателей сложности

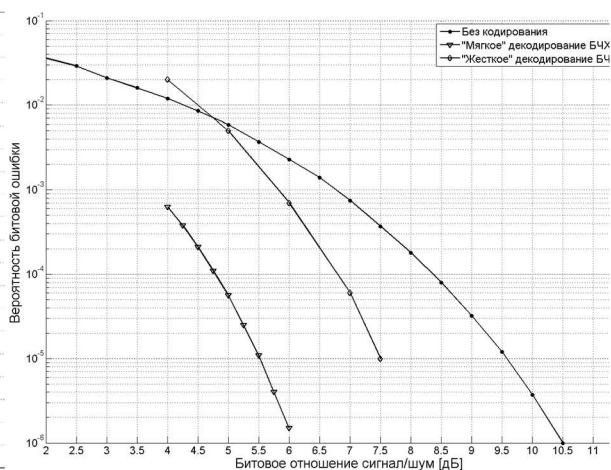


Рисунок 16-Характеристика помехоустойчивости БЧХ кода (52,9)

Каждый новый кадр сигнала L1C сопровождается 9-ти разрядным номером эпохи, закодированным БЧХ кодом (52,9). В составе имитационной модели был реализован БЧХ кодек (52,9) и получены характеристики BER для «мягких» и «жестких» решений (рисунок 16). Выигрыш от применения «мягких» решений по сравнению с «жесткими» решениями составил 2 дБ по уровню 10^5 .

В качестве «мягких» решений на входах LDPC и БЧХ декодеров используются значения синфазной (I) составляющей радиосигнала L1C. Выборка, полученная при низком энергетическом потенциале (битовое отношение сигнал/шум $\sim 3-5$ дБ) и содержащая 3646800 отсчетов (ровно 2026 кадров), представлена на рисунке 17.

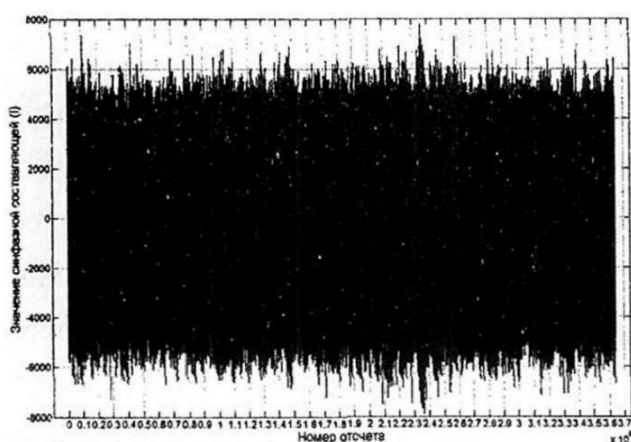


Рисунок 17 – Выборка отсчетов

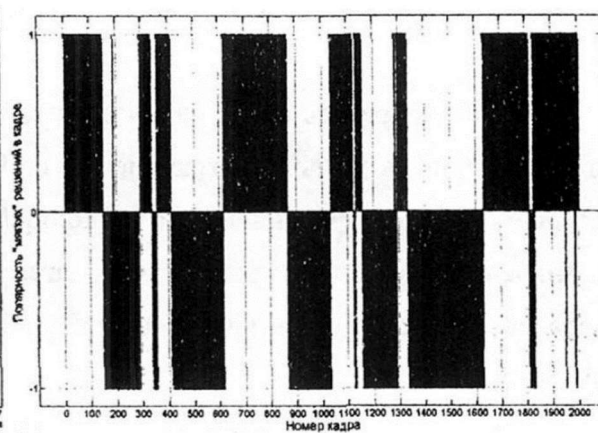


Рисунок 18 – Полярность кадров

На выборке было декодировано 1012 кадров из 2026 (-50%). Потери обусловлены тем, что при обработке выборки сигнала с низким энергетическим потенциалом в тракте приемника есть вероятность срыва слежения за фазой в схеме Костаса. Это может приводить к инверсии «мягких» решений на входе декодера, что равносильно внесению ошибок в принимаемый кадр сигнала L1C.

На рисунке 18 показано изменение полярности кадров в составе обрабатываемой выборки. Прямую полярность имеют 1012 кадров (декодированы успешно), а обратную - 988 (потеряны при декодировании). Еще 26 кадров находятся на «стыках» между сменой полярности. Такие кадры инвертированы частично и чаще всего не поддаются декодированию. В отличие от кадров, находящихся на стыках, полностью инвертированные кадры можно декодировать, умножив «мягкие» решения по этим кадрам на -1. Для этого необходимо идентифицировать факт произошедшей инверсии.

В структуре сигнала L1C не предусмотрено фиксированной преамбулы,

по которой можно было бы принимать решение об инверсии принятого кадра. В данной работе идентифицировать факт инверсии предлагается за счет внутренних ресурсов LDPC декодера, обрабатывающего Subframe2. Идея идентификации инверсии заключается в подаче синхронизированного кадра сигнала L1C на вход декодера и последующем декодировании его составной части Subframe2 в течение большого, но ограниченного числа итераций (до 100 итераций в данной работе). Если в течение 100 итераций не удалось декодировать Subframe2, то принимается решение о полной или частичной инверсии всего кадра. С первого взгляда может показаться некорректным вынесение решения по всему кадру на основе анализа только одной его части. Однако следует учитывать, что после внесения инверсии, имеющей характер пакета ошибок, и до декодирования происходит блоковое деперемежение содержимого в кадре. Это означает, что в какой бы части принятого кадра не произошла инверсия, после деперемежения она будет распространена по всему кадру. Недостатком данного способа является долгое время для принятия решения об инверсии кадра.

Чтобы сократить время на принятие решения о полной или частичной инверсии кадра можно выставить меньшее ограничение на число итераций, однако в этом случае будут потеряны кадры, требующие на декодирование больше итераций, чем выставленное ограничение. В данной работе предлагается не ограничивать декодер в проведении итеративного декодирования, а идентифицировать факт инверсии по анализу сходимости синдрома.

Перед декодированием и после каждой новой проделанной итерации декодер рассчитывает уравнения на четность (синдром), в которых участвуют полученные в результате проведения итерации декодирования «жесткие» символы кодового слова. Моделирование показало (рисунок 13 и рисунок 14), что для случая нормального хода декодирования с каждой новой проделанной итерацией «вес» синдрома (число несошедшихся уравнений на четность), в среднем, уменьшается. Пример сходимости синдрома при декодировании одного из Subframe2, изначально содержащего 127 ошибок и декодированного за 17 итераций показан на рисунке 19. В случае инверсии этого Subframe2 синдром не сойдется к нулю (рисунок 20).

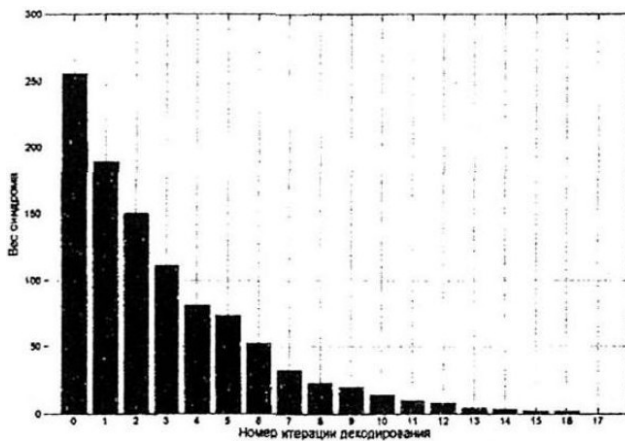


Рисунок 19-Пример сходимости синдрома при декодировании Subframe2

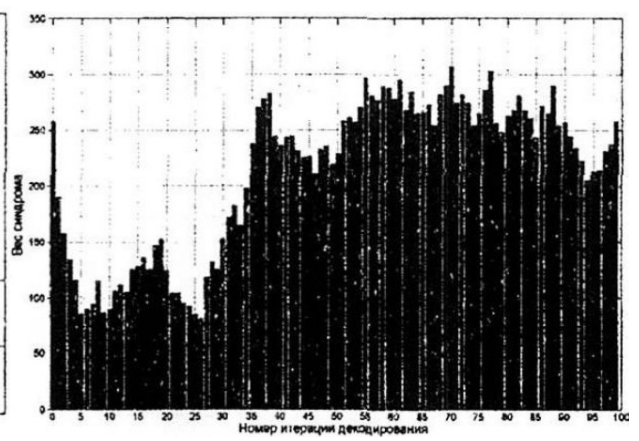


Рисунок 20 - Пример сходимости при декодировании Subframe2 с инверсией

В первом приближении можно сформулировать критерий, позволяющий отличить динамику как на рисунке 19 от динамики на рисунке 20: если значение «веса» синдрома после проведения текущей итерации больше, чем после проведения предыдущей - выносится решение о том, что дальнейшее декодирование не имеет смысла и процесс останавливается.

Сформулированный критерий требует уточнения. Дело в том, что проверка на четность не может обнаружить ошибки четной кратности. Это означает, что если проверка содержит сразу две ошибки (или любое четное число), она будет сходиться (элемент синдрома 0). После проведения одной итерации декодирования одна из этих ошибок может быть исправлена. Число ошибок станет нечетным, что будет зафиксировано проверкой на четность (элемент синдрома 1). Таким образом, число ошибок в результате проведения итерации декодирования уменьшится, а число несошедшихся уравнений на четность («вес» синдрома) увеличится.

Описанное явление иллюстрирует пример на рисунке 21. «Вес» синдрома при декодировании сошелся к нулю, что эквивалентно выполнению всех проверок на четность, однако по ходу декодирования трижды был нарушен сформулированный выше критерий остановки декодирования. Выходом из данной ситуации является разрешение превышения значения «веса» синдрома после текущей итерации над значением после предыдущей итерации, но только фиксированное число раз N . Как только число превышений «веса» синдрома по сравнению с предыдущим значением превышает N - выносится решение об инверсии битового потока. Чем меньше выставленное значение N тем быстрее будет приниматься решение об инверсии, однако тем вероятнее потерять кадры, характеризующиеся

большим, как на рисунке 21, значением N.

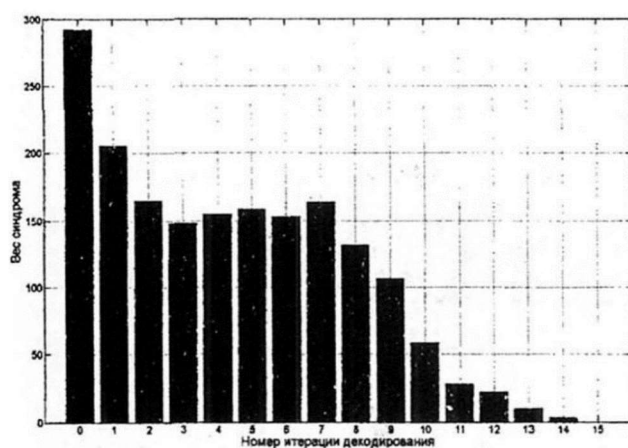


Рисунок 21 - Пример сходимости синдрома при декодировании Subframe2

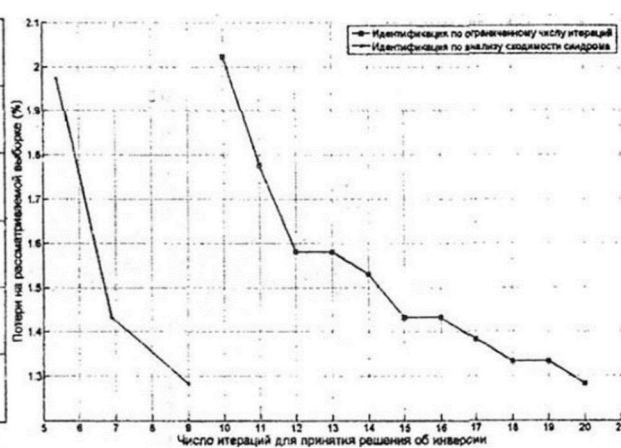


Рисунок 22 - Зависимость числа итераций для идентификации от потерь

На рисунке 22 показаны зависимости числа итераций, необходимых для идентификации инверсии, от потерь на рассматриваемой выборке. В случае использования способа идентификации по ограниченному числу итераций потери складываются из потерь кадров, находящихся на «стыках» при смене полярности и потерь кадров, которые требуют большее число итераций на декодирование, чем выставленное ограничение. В случае использования способа идентификации по анализу сходимости синдрома потери складываются из потерь кадров, находящихся на «стыках» при смене полярности и потерь кадров, которые имеют «нестандартную» сходимости синдрома, превышающую значение N.

Без идентификации инверсии на обрабатываемой выборке было потеряно -50% принятых кадров. Применение предложенных способов идентификации позволило сократить потери до -1-2%, в зависимости от параметров алгоритма идентификации. Без идентификации инверсии на обрабатываемой выборке было потеряно -50% принятых кадров. Применение предложенных способов идентификации позволило сократить потери до -1-2%, в зависимости от параметров алгоритма идентификации.

При равных потерях на выборке способ идентификации по анализу сходимости синдрома в среднем быстрее определяет факт инверсии, чем способ по ограниченному числу итераций. С минимизацией потерь в обоих случаях растет время (число проделываемых итераций) на идентификацию инверсии. В случае идентификации с ограничением числа итераций декодер вынужден проделывать все отведенные ему на декодирование итерации. В

случае идентификации за счет анализа сходимости синдрома декодер имеет критерий, позволяющий принять решение об инверсии на промежуточном этапе декодирования.

Применяя способ идентификации инверсии посредством анализа сходимости синдрома при $N=3$, на обрабатываемой выборке было декодировано 2000 из 2026 кадров (-99%). Еще 26 кадров, как было отмечено выше, было потеряно на «стыках» при смене полярности «мягких» решений на входе декодера. Правильность декодирования подтверждается расшифровкой декодированной информации и расчетом контрольной суммы для каждого кадра. Фрагмент статистики по обработке данной выборки представлен на рисунке 23. Собранная статистика по числу итераций декодирования Subframe2 соответствует полученной на имитационной модели статистике (рисунок 12).

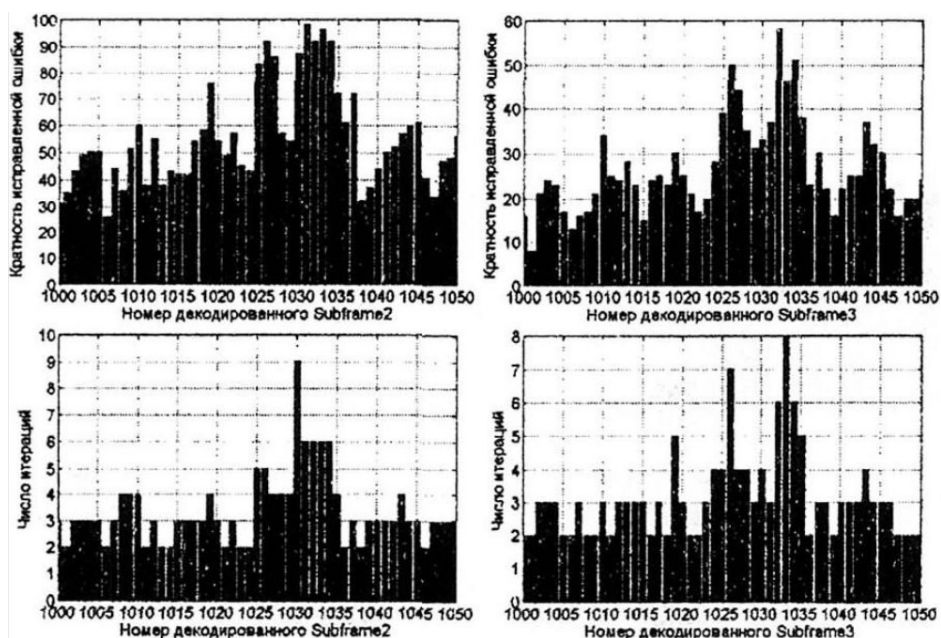


Рисунок 23 - Статистика по декодированию Subframe2 и Subframe3

В пятом разделе приводится краткая классификация турбо кодов и описание выбранного для сравнения турбо кодека. Анализируются результаты турбо декодирования, полученные на имитационной модели. Оценивается вычислительная сложность одной итерации турбо декодирования и проводится сравнение турбо кода и рассматриваемого кода с малой плотностью проверок на четность по выбранным критериям.

На рисунке 24 демонстрируются полученные на имитационной модели зависимости BER от битового отношения сигнал/шум для блочного турбо кодека (32,26) x (32,26) при различном значении длины списка гипотез T.

Увеличение длины списка гипотез приводит к повышению помехоустойчивости.

Помимо этого, на рисунке 24 приводится сравнение помехоустойчивости LDPC кода, декодируемого по алгоритму минимума суммы «Min-sum normalized», и различных реализаций блочного турбо кода, в том числе реализации фирмы АНА. Реализованный LDPC код демонстрирует лучшую исправляющую способность, чем блочный турбо код, опережая реализацию турбо кода фирмы АНА на ~ 0.8 дБ по уровню вероятности битовой ошибки 10^{-5} .

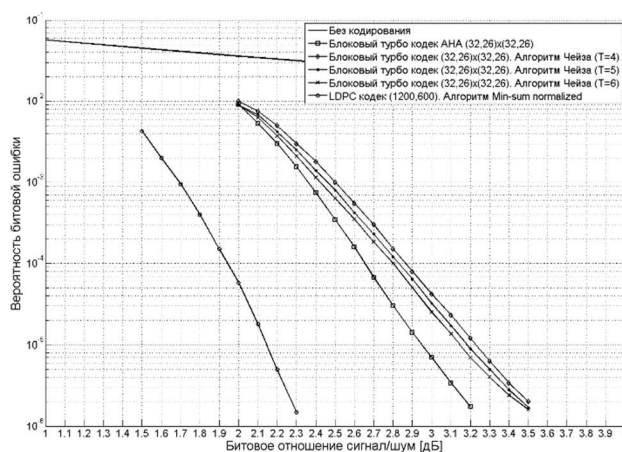


Рисунок 24 - Сравнение BER турбо кода и LDPC кода

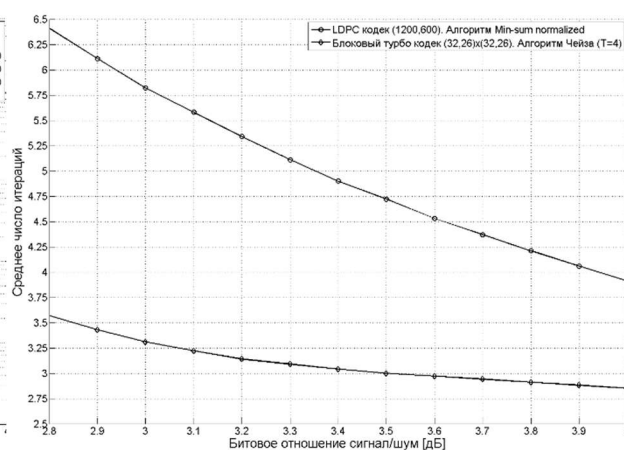


Рисунок 25 - Сравнение среднего числа итераций при декодировании турбо кода и LDPC кода

Моделирование показывает, что в среднем при декодировании кодов с малой плотностью проверок на четность используется больше итераций, чем при декодировании турбо кодов. Зависимости среднего числа итераций от битового отношения сигнал/шум для сравниваемых кодов приведены на рисунке 25.

Сравнение сложности итерации декодирования турбо кода и LDPC кода приводится в таблице 3. Моделирование на ПК показало, что проведение итерации LDPC декодирования требует в 3 раза меньше времени, чем проведение итерации блочного турбо декодирования.

Анализ соотношений между сложностью сравниваемых кодов на итерацию декодирования и их средним используемым числом итераций показывает, что в конкретном случае выгоднее проделывать больше итераций декодирования LDPC с меньшей сложностью, чем осуществлять меньшее число итераций турбо декодирования, но с большей сложностью.

Таблица 3 - Сравнение сложности декодирования

Операция	Количество на итерацию декодирования	
	LDPC («Min-sum normalized»)	Блочный турбо код (Алгоритм Чейза (T=4))
Сложение	37024	72704
Умножение	38706	215040
Сравнение	64159	149825
Взятие модуля	33888	2048
Сложение по модулю 2	4218	201344

ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ ПО РАБОТЕ

Проведенный в данной работе анализ существующих алгоритмов декодирования LDPC показал их тесную взаимосвязь друг с другом в рамках отдельных семейств, а также принципиальные отличия семейств друг от друга.

Полученные в данной работе аналитические соотношения позволяют оценивать сложность итерации декодирования LDPC для различных алгоритмов.

Выявленные характерные особенности в работе алгоритмов декодирования в части расчета поправок позволили разработать модификации с целью повышения вычислительной эффективности декодирования.

Разработанная имитационная модель позволила проводить исследование вероятностных характеристик алгоритмов декодирования и осуществлять отладку программных реализаций декодера для последующего внедрения.

Разработанный способ идентификации инверсии битового потока на входе декодера позволяет определять инверсию за счет внутренних ресурсов LDPC декодера.

Основные результаты можно сформулировать следующим образом:

1. Разработана методика выбора алгоритма декодирования LDPC кодов.
2. Предложена методика представления разреженной проверочной матрицы кода с малой плотностью проверок на четность.
3. Разработаны модификации алгоритмов декодирования, позволяющие без потери в качестве декодирования уменьшить число операций, выполняемых декодером на одну итерацию декодирования.
4. Предложен способ идентификации инверсии битового потока на

входе LDPC декодера.