Украинская государственная академия
железнодорожного транспорта

На правах рукописи

Приходько Сергей Иванович

УДК 621.391

МЕТОДЫ СИНТЕЗА, КОДИРОВАНИЯ И ДЕКОДИРОВАНИЯ СВЕРТОЧНЫХ КОДОВЫХ КОНСТРУКЦИЙ

Специальность 05. 12. 02 – Телекоммуникационные системы и сети

Диссертация на соискание ученой степени
доктора технических наук

Научный консультант
Сорока Леонид Степанович
доктор технических наук, профессор

Харьков – 2009

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Актуальность темы. В условиях рыночных отношений одной из важных задач является повышение эффективности функционирования областей экономики за счет необходимой координации работы всех управленческих подразделений. Решение этой задачи осуществляется путем применения автоматизированных систем управления разного уровня и назначения [1-4].

Структурными элементами современных автоматизированных систем управления являются подсистемы передачи информации, осуществляющие обмен информацией между источниками и потребителями информации по каналам связи [5, 6]. Одним из основных требований к подсистеме передачи является обеспечение заданной достоверности передаваемой информации, которая непосредственно влияет на эффективность процесса управления [7-11].

В связи с ростом требований к эффективности процессов управления, постоянным увеличением объема и скорости передачи информации существенно возрастают требования и к достоверности передаваемой информации.

Согласно Концепции развития связи Украины [12] современный уровень разработки и производства технических средств связи невозможно обеспечить без проведения опережающих исследований. При этом одними из основных направлений опережающих исследований являются развитие математического сопровождения, анализа и синтеза новых, структурно сложных систем и сетей связи; разработка новых технологий и принципов построения систем связи, прежде всего в сфере обработки и передачи информации, а также их составных частей. Таким образом, исследования, направленные на разработку средств повышения достоверности передаваемой информации являются актуальными (перспективными).

Основными и наиболее эффективными средствами повышения достоверности передаваемой информации являются методы помехоустойчивого кодирования [13-25, 38-42]. В теории помехоустойчивого кодирования можно выделить несколько основных направлений развития.

Первое направление, наиболее полно раскрытое в работах [13-18], базируется на блоковых кодах и, преимущественно, алгебраических методах представления процессов синтеза, кодирования и декодирования. Наибольшее распространение среди блоковых кодов получил обширный класс кодов – циклические коды. Наряду с высокими конструктивными свойствами циклических кодов это направление позволяет строить простые и вычислительно эффективные алгоритмы кодирования и декодирования.

Второе направление развития базируется на непрерывных кодах, подклассом которых являются сверточные коды. Отличительной особенностью сверточных кодов является возможность их простого описания деревом или регулярной решетчатой диаграммой, что позволяет реализовать вероятностное декодирование (алгоритмы последовательного декодирования,

алгоритм Витерби, алгоритм максимума апостериорной вероятности). Кодер сверточного кода представляет собой линейный регистр сдвига, сложность которого из-за регулярной решетчатой диаграммы не зависит от длины кода (но зависит от числа состояний решетчатой диаграммы), что является значительным преимуществом. Наиболее полно основы теории сверточных кодов изложены в работах [13-15, 19-25], а также в работах автора [26-35].

Согласно теореме Шеннона [36, 37], наибольшей эффективностью обладают длинные коды. Циклические коды при большой длине кодового слова не позволяют существенно повысить энергетическую эффективность [36, 37], что объясняется, прежде всего, их неудовлетворительными асимптотическими свойствами. Поэтому с этой точки зрения сверточные коды являются более предпочтительными, так как эффективность кодирования сверточными кодами не ухудшается с ростом длины кодового слова [22]. Однако конструктивные кодовые характеристики сверточных кодов (кодовое расстояние) зависят от числа состояний решетчатой диаграммы (которая экспоненциально зависит от количества элементов памяти кодера сверточного кода – линейного регистра сдвига), что приводит к росту сложности декодирования сверточных кодов с высокими кодовыми характеристиками, из-за необходимости анализа всех состояний кодовой решетки в процессе декодирования [19, 20]. Кроме того, в настоящее время отсутствуют вычислительно эффективные методы синтеза сверточных кодов с заданными конструктивными кодовыми характеристиками (как правило, для поиска сверточных кодов используются переборные методы).

В качестве третьего направления можно выделить методы каскадного кодирования [38-40], появление которых связано с попытками синтеза длинных кодов с высокими кодовыми характеристиками на основе достаточно простых составляющих кодов (которые могут быть как блоковыми, так и сверточными), декодирование которых осуществляется отдельными декодерами. Преимущество каскадных кодов состоит в упрощении алгоритмов декодирования и одновременным повышением общей эффективности кодирования. Каскадные коды позволяют обеспечить высокую достоверность в условиях большого уровня шума при умеренной сложности декодирования. Дальнейшее совершенствование методов каскадного кодирования привело к разработке турбокодов [41-72] – параллельных каскадных рекурсивных сверточных кодов.

Реализация турбокодирования информации блоками большой длины не представляет собой значительных трудностей из-за использования составляющих сверточных кодов, поскольку сложность сверточного кодирования не зависит от длины кодируемой информационной последовательности. В результате, турбокоды могут обеспечить эффективность кодирования, близкую к теоретически предельному значению , определенного теоремой Шеннона [36, 37]. Недостатком существующих турбокодов является уменьшение эффективности кодирования при высоком энергетическом отношении сигнал/шум, что связано с малым минимальным расстоянием составляющих турбокод сверточных кодов.

Видимым путем устранения недостатков турбокодов является использование в качестве составляющих турбокод кодов рекурсивных сверточных кодов с высокими конструктивными характеристиками, что приведет к повышению минимального расстояния турбокода и позволит выбирать скорость турбокодирования без ограничений. Однако препятствием на пути применения составляющих турбокод сверточных кодов с указанными свойствами является высокая сложность декодирования сверточных кодов с высокими конструктивными характеристиками. Кроме того, как показано в работах [14, 18], существующие методы синтеза сверточных кодов не позволяют эффективно строить рекурсивные сверточные коды с высокими конструктивными свойствами (большим кодовым расстоянием), что сдерживает разработку и внедрение перспективных систем турбокодирования.

Из вышесказанного можно сделать вывод, что развитая в настоящее время алгебраическая теория блочного кодирования не может быть непосредственно применена к сверточным кодам по причине значительного различия в их свойствах по сравнению с блочными кодами. Несмотря на это в работах [19, 20, 73] показано, что существует возможность представления сверточного кода в виде блочного кода полубесконечной длины и его последующим алгебраическим описанием. Это направление теории помехоустойчивого кодирования получило развитие в работах автора при написании кандидатской диссертации [74-76]. Однако положительные результаты в рассматриваемых работах получены только для ограниченного диапазона низких скоростей кодирования, значения которых не удовлетворяют современным требованиям, предъявляемым к параметрам помехоустойчивых кодов (как правило, на практике требуются более высокие скорости кодирования). Кроме того, в этих работах не рассматривается возможность применения алгебраической теории для реализации декодирования сверточных кодов.

Таким образом, возникает научная проблема (противоречивая ситуация), в которой существующие положения теории помехоустойчивого кодирования не позволяют вычислительно реализуемо решать задачи синтеза , кодирования и декодирования сверточных кодов с высокими конструктивными кодовыми характеристиками и с произвольными параметрами. Разрешение научной проблемы (противоречивой ситуации) возможно путем разработки на основе единого теоретического подхода методов синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций с требуемыми свойствами и характеристиками.

Актуальность темы диссертационных исследований определяется необходимостью обеспечения заданной достоверности передаваемой информации путем применения вычислительно реализуемых процедур синтеза, кодирования и декодирования алгебраически заданных сверточных кодов (кодовых конструкций) с высокими конструктивными кодовыми характеристиками.

Связь работы с научными программами, планами, темами. Исследования в диссертационной работе проводились в соответствии со следующими нормативными актами.

1. Концепция Национальной программы информатизации, одобренная Законом Украины «Про Концепцію Національної програми інформатизації» от 4 февраля 1998 г. N 75/98-ВР.

2. Концепция развития связи Украины до 2010 года, утвержденная постановлением Кабинета Министров Украины «Про Концепцію розвитку зв' язку України до 2010 року» от 9 декабря 1999 г. №2238.

3. Государственная научно-техническая программа «Створення перспективних телекомунікаційних систем і технологій».

4. Концепция создания Государственной интегрированной информационной системы обеспечения управления подвижными объектами, утвержденная постановлением Кабинета Министров Украины от 17 июля 2003 г. №410-р.

Цель и задачи исследований. Целью диссертационной работы является разработка концептуального подхода на основе новых методов синтеза, кодирования и декодирования сверточных кодовых конструкций с использованием математического аппарата алгебраической теории помехоустойчивого кодирования для повышения достоверности передаваемой информации.

Для достижения поставленной цели необходимо решить следующие научные задачи.

1. Разработать и исследовать методы синтеза алгебраически заданных сверточных кодовых конструкций для повышения достоверности передаваемой информации:

– разработать (с использованием математического аппарата алгебраической теории помехоустойчивого кодирования) методы синтеза алгебраически заданных сверточных кодов, теоретически обосновать аналитические выражения по оценке кодовых соотношений синтезируемых кодов;

– разработать методы и алгоритмы кодирования алгебраически заданными сверточными кодами, исследовать конструктивные свойства синтезированных сверточных кодовых конструкций.

2. Разработать и исследовать вычислительно эффективные (вычислительно реализуемые) методы и алгоритмы декодирования алгебраически заданных сверточных кодов:

– разработать алгебраический метод декодирования синтезированных сверточных кодов;

– разработать комбинированный метод декодирования алгебраически заданных сверточных кодов, объединяющий в себе процедуры переборного поиска по кодовой решетке и алгебраические процедуры локализации и исправления ошибок;

– разработать (вычислительные) алгоритмы декодирования алгебраически заданных сверточных кодов и предложения по программной и

аппаратной реализации.

3. Разработать параллельные каскадные сверточные кодовые конструкции на основе алгебраически заданных рекурсивных сверточных кодов и вычислительно реализуемых алгоритмов их декодирования:

– аналитически формализовать и разработать методы синтеза турбокодов с использованием алгебраически заданных сверточных кодов;

– разработать и исследовать алгоритмы итеративного декодирования параллельных каскадных кодовых конструкций с алгебраически заданными сверточными кодами;

– разработать и исследовать алгоритмы мягкого декодирования составляющих турбокод алгебраически заданных сверточных кодов.

4. Разработать практические рекомендации по использованию алгебраических сверточных кодов в телекоммуникационных системах и сетях:

– разработать (с использованием методов математической статистики и проверки гипотез) имитационную модель системы передачи информации, методику оценки и исследовать достоверность передаваемой информации в телекоммуникационных системах и сетях с использованием алгебраически заданных сверточных кодов и турбокодов на их основе;

– обосновать практические рекомендации по использованию алгебраических сверточных кодов в телекоммуникационных системах и сетях.

Объект исследования. Процесс повышения достоверности передаваемой информации на основе применения алгебраически заданных сверточных кодовых конструкций.

Предмет исследования. Методы и алгоритмы синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций.

Методы исследования. Разработка и исследование алгебраических методов и процедур синтеза, кодирования и декодирования сверточных кодовых конструкций проведены с использованием методов алгебраической теории помехоустойчивого кодирования, теории полей Галуа и теории чисел. Оценка достоверности передаваемой информации проведена с использованием методов статистической теории связи, теории вероятности и математической статистики. Разработка рекомендаций по реализации кодеров алгебраически заданных сверточных кодов проведена с использованием методов теории цифровых автоматов.

Научная новизна полученных результатов. Новым научным результатом диссертации является развитие теории помехоустойчивого кодирования в части синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций (с произвольными кодовыми характеристиками и свойствами).

В рамках главного нового научного результата получено ряд частных научных результатов.

1. Получил дальнейшее развитие единый концептуальный подход алгебраического представления сверточных кодов в виде недвоичных

блоковых циклических кодов (полубесконечной длины), отличающийся от известного (теоретическим обобщением на случай полубесконечной длины кодового слова циклического кода и) использованием порождающих многочленов недвоичных циклических кодов, ограниченных на произвольное подполе, что позволяет рассматривать с единых теоретических позиций процессы синтеза, кодирования и декодирования сверточных кодов с произвольными свойствами и кодовыми характеристиками и теоретически обосновать аналитические выражения по оценке кодовых соотношений синтезируемых сверточных кодовых конструкций, аналитически связать их параметры и выразить через кодовые характеристики соответствующих циклических кодов.

2. Получили дальнейшее развитие вычислительно эффективные (вычислительно реализуемые) алгебраические методы синтеза (алгебраически заданных) сверточных кодов, отличающиеся от известных использованием ограничения недвоичного циклического кода на произвольное подполе, что позволяет синтезировать (алгебраически заданные) сверточные коды с произвольными свойствами и кодовыми характеристиками.

3. Получили дальнейшее развитие методы кодирования алгебраически заданными сверточными кодами, отличающиеся от известных теоретически обоснованными процедурами алгебраического построения рекурсивных и нерекурсивных сверточных кодов через обобщение циклических кодов на случай бесконечной длины, что позволяет аналитически формализовать процесс помехоустойчивого кодирования синтезируемыми сверточными кодами с высокими (конструктивными) кодовыми характеристиками.

4. Впервые разработаны алгебраический и комбинированный методы декодирования алгебраически заданных сверточных кодов, которые отличаются от известных методов процедурами алгебраической локализации и ускоренными процедурами (алгоритмами) последовательного поиска, что позволяет реализовать вычислительно эффективное (вычислительно реализуемое) декодирование непрерывных кодовых конструкций с большой длиной кодового ограничения (с большим кодовым расстоянием) для повышения достоверности передаваемой информации.

5. Получили дальнейшее развитие методы синтеза параллельных каскадных сверточных конструкций (методы турбокодирования), отличающиеся от известных использованием алгебраически заданных рекурсивных сверточных кодов, что позволяет аналитически связать параметры турбокодов с параметрами алгебраически заданных рекурсивных сверточных кодов и синтезировать параллельные каскадные сверточные конструкции с заданными (конструктивными кодовыми) характеристиками.

6. Получил дальнейшее развитие метод итеративного декодирования турбокодов с алгебраически заданными рекурсивными сверточными кодами, который отличается от известного обобщенным представлением бесконечного кодового слова сверточного кода через бесконечную сумму последовательных наборов из кодовых слов циклического кода, что

позволяет за счет сведения декодирования сверточного кода к декодированию последовательности кодовых слов циклического кода декодировать турбокоды на основе алгебраически заданных сверточных кодов с большим числом элементов памяти (с высокими кодовыми характеристиками, высоким кодовым расстоянием).

Практическое значение полученных результатов исследований состоит в следующем.

1. Разработаны вычислительно реализуемые алгоритмы синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций с требуемыми (кодовыми) характеристиками.

2. Разработана методика (эмпирической) оценки достоверности передаваемой информации, которая позволяет для (заданных параметров математической модели) дискретно-непрерывного канала с заданной погрешностью оценить вероятность ошибочного приема бита информации и соответствующий энергетический выигрыш от кодирования.

3. Разработана имитационная модель системы передачи информации с использованием алгебраически заданных сверточных кодовых конструкций, с помощью которой установлено, что

– синтезированные сверточные кодовые конструкции, полученные с помощью разработанных вычислительно реализуемых алгоритмов, не уступают по энергетическим характеристикам известным в настоящее время кодам; их практическое использование позволяет обеспечить повышение достоверности передаваемой информации в каналах со случайно возникающими ошибками за счет отсутствия ограничений при выборе требуемых параметров синтезируемых сверточных кодовых конструкций;

– разработанные вычислительно реализуемые алгоритмы декодирования сверточных кодовых конструкций с высокими конструктивными кодовыми характеристиками имеют параметры близкие к теоретически предельным значениям.

4. Разработаны практические рекомендации по использованию турбокодов с синтезированными алгебраически заданными сверточными кодами. Для обеспечения вероятности ошибки на бит при значении энергетического отношения сигнал/шум 1,5 – 2 дБ, предлагается использовать турбокоды с количеством элементов памяти 2 – 4. Для обеспечения вероятности ошибки на бит предлагается использовать турбокоды с количеством элементов памяти 6 – 8. Скорость кодирования не рекомендуется выбирать менее чем 1/3.

5. Полученные результаты использованы в научно-исследовательских работах «Мрія», «Алгоритм» (Харьковский университет Воздушных Сил, акт реализации от 12.04.2005), на производстве при разработке специального математического и программного обеспечения программно-аппаратного макета помехоустойчивого кодера (декодера) в ЦККБ «Протон» (акт реализации от 26.05.2008) и в учебном процессе Украинской государственной академии железнодорожного транспорта (акт реализации от

15.04.2008).

Таким образом, полученные в ходе исследований научные и практические результаты в совокупности решают важную научную проблему путем разработки на основе нового концептуального подхода методов синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций с требуемыми свойствами и характеристиками, имеющую большое значение как для развития отдельного направления теории помехоустойчивого кодирования, так и для решения прикладных вопросов, связанных с обеспечением заданной достоверности передаваемой информации в телекоммуникационных системах и сетях.

Личный вклад автора. Все результаты, изложенные в диссертационной работе, получены автором самостоятельно. В работах, выполненных в соавторстве и опубликованных в изданиях, которые вошли в перечень ВАК Украины, автору принадлежат:

– в [77] предложен принцип приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам;

– в [78] предлагается дальнейшее развитие способа приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам;

– в [79] разработан алгоритм приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам;

– в [80] разработан метод приведения сверточных кодов к кодам Рида-Соломона;

– в [81] предложен подход приведения ортогонализируемых сверточных кодов к квазиортогональным;

– в [82] предложен подход для приведения ортогональных сверточных кодов к квазиортогональным сверточным кодам;

– в [83] предложен метод приведения ортогональных сверточных кодов к квазиортогональным сверточным кодам;

– в [84] исследованы возможности представления сверточных кодов при помощи циклических кодов;

– в [85] разработан принцип последовательного декодирования обобщенно заданных сверточных кодов;

– в [91] предложено использовать порождающие многочлены недвоичных циклических кодов, ограниченных на произвольное подполе, для алгебраического построения несистематических сверточных кодов;

– в [92] разработан алгебраический метод сверточного кодирования ;

– в [93] разработан алгебраический метод построения систематических сверточных кодов;

– в [94] разработан алгебраический метод построения рекурсивных сверточных кодов;

– в [95] разработаны методы синтеза параллельных каскадных сверточных конструкций на основе алгебраически заданных рекурсивных сверточных кодов;

– в [96] разработан метод декодирования алгебраически заданных сверточных кодов;

– в [97] разработаны процедуры алгебраической локализации и ускоренные процедуры последовательного поиска для комбинированного метода декодирования алгебраически заданных сверточных кодов;

– в [100] разработан метод итеративного декодирования турбокодов на основе алгебраически заданных рекурсивных сверточных кодов.

Апробация результатов диссертации. Основные результаты диссертации докладывались и были одобрены на следующих научно-технических конференциях:

–Международная научно-техническая конференция «Современные методы кодирования в электронных системах» (Сумы, 2002);

–Международная научно-техническая конференция «Современные методы кодирования в электронных системах» (Сумы, 2004);

–Первая научно-техническая конференция Харьковского университета Воздушных Сил (Харьков, 2005);

–Международная научно-техническая конференция «Интегрированные компьютерные технологии в машиностроении» (Харьков, 2007);

–Международная научно-техническая конференция «Стратегии ИТ-технологий в образовании, экономике и экологии» (Харьков, 2007);

–Седьмая международная научно-техническая конференция «Проблемы информатики и моделирования» (Харьков, 2007);

– Четвертая научная конференция Харьковского университета Воздушных Сил (Харьков, 2008);

– Первая Всеукраинская научно-практическая конференция (Львов, 2008);

– 22 международная научно-практическая конференция «Перспективные компьютерные, управляющие и телекоммуникационные системы для железнодорожного транспорта Украины» (Алушта, 2009).

Публикации. Основные положения диссертационной работы изложены в 24 научных статьях [77-100], 3 патентах [101-103], 9 тезисах выступлений [104-112], 2 отчетах о НИР [113-114].

Структура и объем диссертации. Диссертационная работа состоит из введения, шести разделов, выводов и приложения. **В первом разделе** проведен анализ общих положений теории помехоустойчивого кодирования, представлена модель системы передачи информации через совокупность формальных операторов, аналитически описывающих процедуры преобразования информации, проведены анализ и сравнительные исследования известных методов помехоустойчивого кодирования. Обобщены и теоретически обоснованы пути повышения достоверности передаваемой информации на основе использования помехоустойчивых кодов. На основе полученных результатов обосновывается выбор направления исследований, вводятся критерии и показатели эффективности, математически формализуется постановка научной проблемы. **Во втором и**

третьем разделах на основе единого общетеоретического подхода с использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов разрабатываются методы и алгоритмы синтеза алгебраически заданных нерекурсивных и рекурсивных сверточных кодов. **В четвертом разделе** разрабатываются методы декодирования алгебраически заданных сверточных кодов, основанные на использовании бесконечной серии синдромов кодовых слов циклического кода. Предлагается способ формирования бесконечной серии синдромов алгебраически заданного сверточного кода. Разрабатывается подход комбинированного декодирования алгебраически заданных сверточных кодов, состоящий в совмещении алгебраических процедур и процедур последовательного поиска по кодовой решетке. **В пятом разделе** исследуются методы построения параллельных каскадных кодовых конструкций и процедуры их декодирования. Предлагаются схемы турбокодирования с использованием рекурсивных сверточных кодов, заданных через порождающий и/или проверочный многочлены недвоичного циклического кода. Разрабатываются алгоритмы построения турбокодов с требуемыми параметрами. **В шестом разделе** исследуются модели каналов связи, разработана методика оценки достоверности передаваемой информации, которая позволяет для заданных параметров математической модели канала связи с заданной погрешностью оценить вероятность ошибочного приема бита информации и соответствующий энергетический выигрыш от кодирования. Разработана имитационная модель системы передачи информации с использованием алгебраически заданных сверточных кодовых конструкций, которая позволяет оценить эффективность кодирования синтезированными сверточными кодовыми конструкциями. На основе полученных результатов проведенных исследований разработаны практические рекомендации по использованию синтезированных алгебраически заданных кодовых конструкций для повышения достоверности передаваемой информации. **В выводах по работе** сообщаются основные результаты проведенных исследований.

В заключение автор выражает искреннюю благодарность научному консультанту доктору технических наук профессору Сорока Л.С. за оказанную помощь и поддержку при проведении исследований.

РАЗДЕЛ 1
МЕТОДЫ ПОВЫШЕНИЯ ДОСТОВЕРНОСТИ ПЕРЕДАВАЕМОЙ ИНФОРМАЦИИ. ВЫБОР НАПРАВЛЕНИЯ ИССЛЕДОВАНИЙ

В данном разделе проведен анализ общих положений теории помехоустойчивого кодирования, представлена модель подсистемы передачи информации через совокупность формальных операторов, аналитически описывающих процедуры преобразования информации, проведены анализ и сравнительные исследования известных методов помехоустойчивого кодирования. Обобщены и теоретически обоснованы пути повышения помехоустойчивости подсистемы передачи информации на основе использования избыточных кодов в каналах со случайно возникающими ошибками.

На основе полученных результатов обосновывается выбор направления исследований, вводятся критерии и показатели эффективности, математически формализуется постановка научной проблемы, имеющей важное значение, как для развития отдельного направления теории помехоустойчивого кодирования, так и для решения прикладных вопросов, связанных с обеспечением показателей помехоустойчивости и достоверности передаваемой информации в телекоммуникационных системах и сетях.

1.1. Модель системы передачи информации

Рассмотрим основные положения и принципы построения систем передачи информации телекоммуникационных систем, проанализируем основные направления их дальнейшего развития.

В соответствии с классическими определениями [115, 116] под системой передачи информации понимают цифровую систему связи, т.е. совокупность технических средств обработки, передачи и приема информации на приемной и передающей стороне, а так же среду передачи (канал связи). Схема системы передачи информации представлена на рис. 1.1.

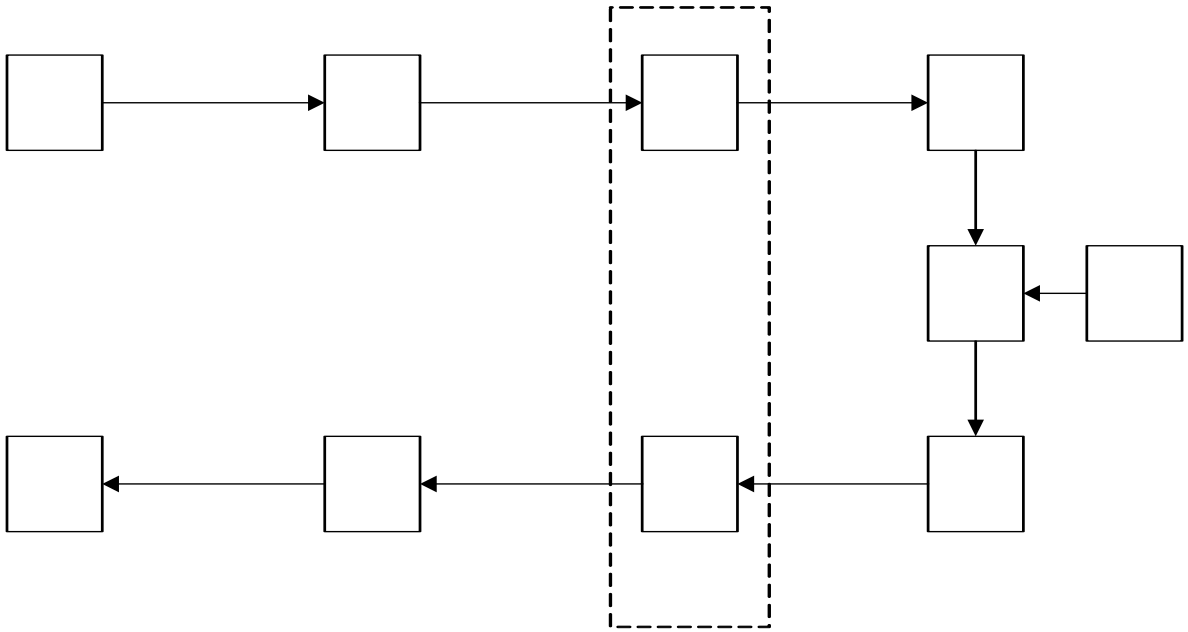


Рис. 1.1. Схема системы передачи информации

Система передачи информации состоит из следующих элементов.

1. Источник информации.
2. Подсистема кодирования источника информации.
3. Подсистема канального (помехоустойчивого) кодирования.
4. Передатчик, который преобразует по некоторому правилу информационные сообщения в сигналы, соответствующие характеристикам данного канала.
5. Канал – среда, которая используется для передачи сигнала от источника к приемнику.
6. Источник шума.
7. Приемник, выполняющий операцию, обратную по отношению к операции, производимой передатчиком.
8. Подсистема декодирования, выполняющая операции, обратные канальному кодированию (декодер помехоустойчивого кода).
9. Подсистема декодирования получателя информации.
10. Получатель информации – это объект, для которого предназначено информационное сообщение.

Математическая модель системы передачи информации задается совокупностью операторов обработки, передачи и приема сообщений:

$\{W_I\}$ – оператор формирования информационных сообщений, который описывает работу источника информационных сообщений;

$\{W_M\}$ – оператор преобразования информационных сообщений в информационные блоки данных (оператор кодирования источника);

$\{W_C\}$ – оператор преобразования блоков данных в кодовые слова (оператор помехоустойчивого кодирования);

$\{W_S\}$ – оператор преобразования кодовых слов в последовательность сигналов (оператор формирования сигналов);

$\{W_Z\}$ – оператор взаимодействия передаваемых сигналов с помехами в канале связи;

$\{W^{-1}_Z\}$ – оператор взаимодействия полученных сигналов с пространством сообщений на входе узла получателя информации;

$\{W^{-1}_S\}$ – оператор преобразования последовательности сигналов в кодовое слово (оператор обработки сигналов);

$\{W^{-1}_C\}$ – оператор преобразования кодовых слов в блоки данных (оператор декодирования помехоустойчивого кода);

$\{W^{-1}_M\}$ – оператор преобразования информационных блоков данных в информационные сообщения (оператор декодирования получателя информации);

$\{W^{-1}_I\}$ – оператор обработки полученных информационных сообщений.

Источник информации (1) порождает поток информационных сообщений из множества $\{M_1, M_2, \ldots, M_m\}$. Процедуру формирования сообщений зафиксируем в виде формального оператора $\{W_I\}$. Каждое сообщение M_i представляется конкретной реализацией некоторого случайного процесса, описывающего работу источника сообщений. Каждому сообщению $M_i \in \{M_1, M_2, \ldots, M_m\}$ соответствует вероятность $P(M_i)$. Распределение вероятностей случайного процесса задается совокупным распределением вероятностей случайных величин, т.е. множеством априорных вероятностей

, (1.1)

причем

.

Каждое сообщение $M_i \in \{M_1, M_2, \ldots, M_m\}$ несет информацию, численно равную мере неопределенности (энтропии) конкретной реализации случайного процесса, описывающего работу источника сообщений, т.е. запишем:

,

где основание логарифма задает единицу измерения количества информации. Для простоты положим основание равное двум, что соответствует двоичному (битовому) исчислению количества информации.

Таким образом, источник информации представляется как случайный процесс, конкретная реализация которого представляется в виде некоторого сообщения $M_i \in \{M_1, M_2, \ldots, M_m\}$. Если в единицу времени источник формирует одно сообщение из множества M, тогда мера информации порождаемая источником за ту же единицу времени, задается функцией вида:

, (1.2)

т.е. энтропией множества вероятностей .

Максимальное значение энтропии источника достигается при равновероятном появлении сообщений из множества $M_i \in \{M_1, M_2, ..., M_m\}$. Тогда

и имеем:

.

Отношение энтропии источника к максимальному значению, которого могла бы достичь энтропия при тех же символах, называют относительной энтропией источника. Это величина максимального сжатия, которого можно достичь при том же алфавите символов. Единица минус относительная энтропия есть избыточность δ:

. (1.3)

Подсистема кодирования источника информации (2) представляет сообщения $M_i \in \{M_1, M_2, ..., M_m\}$, порожденные источником (1), в удобном для дальнейшей обработке виде и служит, прежде всего, для сжатия передаваемых данных, т.е. для устранения избыточности δ. Другими словами , подсистема кодирования источника информации (2) реализует отображение множества сообщений $\{M_1, M_2, ..., M_m\}$ в множество информационных блоков данных $\{I_1, I_2, ...,I_k\}$ так, чтобы все вероятности $P(I_i)$ из множества вероятностей

(1.4)

были, по возможности, равны. В этом (теоретическом) случае энтропия

(1.5)

максимальна и, очевидно, равна максимальной энтропии источника, т.е.

,

что влечет за собой равенство $k = m$.

В общем случае

,

.

Подсистема канального (помехоустойчивого) кодирования (3) реализует отображение информационных блоков данных $\{I_1, I_2, ...,I_k\}$ в множество кодовых слов $\{C_1, C_2, ..., C_n\}$. Целью помехоустойчивого кодирования является внесение по определенному алгоритму в передаваемые данные избыточности. На приемной стороне, анализируя принятые кодовые слова с возможной ошибкой и их соответствие внесенной избыточности, подсистема канального (помехоустойчивого) декодирования (8) уменьшает действие возникших при передаче сообщений

ошибок.

Передатчик (4) преобразует по некоторому правилу информационные сообщения в сигналы, соответствующие характеристикам данного канала. Другими словами, передатчик (4) реализует отображение множества кодовых слов $\{C_1, C_2, \ldots, C_n\}$ в множество сигналов $\{S_1, S_2, \ldots, S_s\}$.

В процессе передачи сигнала по каналу связи (5) на него воздействует шум (6). Подсистема приема (7) принимает смесь сигнала

и помехи и выполняет преобразования, обратные подсистеме передачи. После выполнения канального декодирования и декодирования источника принятое сообщение отправляется получателю информации. Подсистема обратного преобразования (8), (9), (10) выполняет функции, реализующие обратные отображения: множества сигналов

в множество кодовых слов , множества

кодовых слов в множество информационных блоков

, множество информационных блоков в

множество сообщений . Каждое отображение задается соответствующими энтропийными и вероятностными характеристиками.

Анализ рассмотренной модели подсистемы передачи информации показывает, что ее формальное описание задается совокупностью операторов формирования, обработки и передачи сообщений, а так же оператора воздействия шума на передаваемое сообщение в канале связи. При оценке параметров подсистемы передачи информации необходимо произвести выбор модели канала связи, которая будет определять возможные последовательности ошибок, возникающих в канале связи.

1.2. Критерии и показатели эффективности системы передачи информации. Постановка научной проблемы

Построение эффективной системы передачи информации сопряжено с оптимизацией целого спектра взаимосвязанных и противоречивых требований: максимизация скорости передачи информации и минимизация вероятности появления битовой ошибки; минимизация потребляемой мощности и/или отношения энергии сигнала к спектральной плотности мощности шума и минимизация ширины полосы пропускания; снижение сложности практической реализации и максимизация числа абонентов сети связи [117-119]. Кроме того, существуют математически обоснованные теоретические ограничения на предельную скорость передачи данных (теорема Шеннона-Хартли) и на минимально необходимую полосу частот (теорема Найквиста), обуславливающие естественные сдерживающие факторы и неизбежно приводящие к сложным компромиссам в системных требованиях [36, 37].

Рассмотрим основные показатели и критерии эффективности системы передачи информации с точки зрения их взаимосвязи на системном уровне, обоснуем пути построения высокоскоростных помехоустойчивых систем передачи информации на основе эффективных компромиссных системных решений, позволяющих достичь оптимальных показателей на уровне сигнально-кодовых конструкций.

Пусть на вход непрерывного канала в течение некоторого временного интервала подается непрерывная функция , где аргумент функции пробегает значения от 0 до и выполняется условие конечности интеграла

.

Последнее выражение определяет энергию сообщения для сигнала

.

При средняя мощность сообщения определяется как предел (при условии его существования)

.

где - энергия сообщения на интервале длины .

Спектр сигнала определяется преобразованием Фурье

.

Ограниченный по полосе сигнал, спектр которого сосредоточен в интервале частот около несущей частоты , имеет вид

при (или, по крайней мере, при , где - сколь угодно малая величина) [115, 116].

Выходной сигнал ограниченного по полосе канала с аддитивным белым гауссовым шумом (АБГШ) с входным сигналом определяется сверткой

,

где - ограниченная по полосе функция (импульсный отклик канала), - реализация гауссова случайного процесса.

Для информационных сообщений конечной длины, состоящих из битов и имеющих энергию сообщения , энергия , приходящаяся на один бит (энергия бита), определяется соотношением . Для информационных сообщений бесконечной длины, передаваемых с постоянной скоростью , величина определяется соотношением

.

Определим среднюю мощность поступающего в приемник шума и соответствующую ему спектральную плотность мощности шума (удельную к полосе частот мощность): .

В любой реализуемой системе связи, выполняющей неидеальную фильтрацию, наблюдается межсимвольная интерференция. В фундаментальных работах по теории связи показано, что теоретически минимальная полоса пропускания (ширина полосы частот по Найквисту), требуемая для немодулированной передачи символов/секунду без межсимвольной интерференции, составляет Гц [115], что является основным ограничением экономного использования полосы частот.

На практике, минимальная ширина полосы частот по Найквисту вследствие неидеальной фильтрации увеличивается на 10-40 % , т.е. реальная пропускная способность цифровых систем связи снижается с идеальных 2 символов/с/Гц до 1,8 – 1,4 символов/с/Гц [115].

Из набора символов система модуляции или кодирования присваивает каждому символу -битовое значение, .

Следовательно, скорость передачи данных (скорость передачи битов) должна быть в раз больше скорости передачи символов , т.е.:

,

или

.

Другими словами, применение больших ансамблей сигналов позволяет существенно повысить скорость передачи данных при фиксированных требованиях к полосе частот.

Одной из важнейших характеристик системы передачи информации является достоверность передаваемой информации. Под достоверностью понимают свойство системы, характеризующее ее способность обеспечивать точное воспроизведение передаваемых сообщений в пунктах приема [7-11].

Основным показателем достоверности является вероятность правильного приема битов . Чаще используют обратный показатель - показатель потери достоверности как вероятность ошибочного приема битов . Очевидно, что величины и зависят только от отношения или и применяемых методов модуляции и кодирования.

Проведенный анализ показал, что вероятность ошибки на кодовую комбинацию в телекоммуникационных сетях общего пользования должна находиться в пределах в зависимости от реализуемого класса обслуживания [120, 121], а в телекоммуникационных сетях специального назначения вероятность ошибки на кодовую комбинацию не

должна превышать 10^{-7} - 10^{-8}. Однако в реальных каналах связи указанная вероятность не превышает $P_{ош} > 10^{-2} – 10^{-4}$ (вероятность ошибки на кодовую комбинацию на выходе демодулятора).

С достоверностью передаваемой информации тесно связана другая характеристика системы передачи информации – помехоустойчивость. Под помехоустойчивостью понимают способность системы передачи обеспечивать передачу информации в условиях воздействия помех [117, 118]. Количественной мерой помехоустойчивости является минимально необходимое для обеспечения требуемой достоверности соотношение энергии сигнала к спектральной плотности мощности шума. Т.е. этот показатель позволяет при фиксированном уровне достоверности (при фиксированных $P_{ош}$ и $P_{пп}$) оценить (сравнить между собой) энергетическую эффективность различных подсистем передачи информации.

В фундаментальных работах по математической теории связи показано , что пропускная способность канала связи с АБГШ является функцией средней мощности принятого сигнала , средней мощности шума и ширины полосы пропускания :

,

где основание логарифма задает единицу измерения пропускной способности (при основании равном двум пропускная способность измеряется в битах/секунду).

Теорема Шеннона устанавливает связь между пропускной способностью канала связи (предельной скоростью передачи), полосой частот и отношением мощности сигнала и мощности гауссова шума . Теоретически (при использовании соответствующей схемы модуляции и кодирования) информационные сообщения можно передавать со сколь угодно малой вероятностью ошибки и скоростью передачи . Таким образом, величины устанавливают , и устанавливают предел скорости передачи информации , а не вероятности ошибочного приема символов сообщения [117, 118].

Введем следующие обозначения: - нормированная пропускная способность как функция отношения мощности сигнала к мощности гауссова шума в канале связи; - нормированная полоса пропускания как функция отношения мощности сигнала к мощности гауссова шума в канале связи. На рис. 1.2 и рис. 1.3 приведены зависимости и .

Принимая во внимание тот факт, что мощность шума пропорциональна полосе пропускания: , получим

.

Если скорость передачи информации равна пропускной способности канала ( ), тогда учитывая, что время передачи одного бита обратно скорости , запишем:

.

Рис. 1.2. Зависимость нормированной пропускной способности от отношения мощности сигнала к мощности гауссова шума

Рис. 1.3. Зависимость нормированной полосы пропускания от отношения мощности сигнала к мощности гауссова шума

Откуда после подстановки получим:

.

Преобразовав последнее выражение, получим (для двоичного логарифма):

,

откуда имеем предельную зависимость:

.

На рис. 1.4 и рис. 1.5 приведены зависимости нормированной пропускной способности и нормированной полосы пропускания от отношения энергии сигнала, приходящейся на один передаваемый бит данных, к спектральной плотности мощности гауссова шума .

Анализ зависимостей, приведенных на рис. 1.4 и рис. 1.5 показывает, что существует нижнее предельное значение , при котором невозможна безошибочная передача информации.

Рис. 1.4. Зависимость нормированной пропускной способности от отношения энергии бита к спектральной плотности мощности гауссова шума

Рис. 1.5. Зависимость нормированной полосы пропускания от отношения энергии бита к спектральной плотности мощности гауссова шума

Используя соотношение

после подстановки

из уравнения

получим

,

откуда в пределе имеем

,

или, в децибелах,

дБ.

Наименьшее значение отношения , ниже которого передача информации без ошибок невозможна, называют пределом Шеннона. В тоже время, предел Шеннона есть минимально возможное соотношение энергии бита к спектральной плотности мощности АБГШ, которое можно достичь при использовании помехоустойчивого кодирования для обеспечения сколь угодно малой вероятности ошибки. Нахождение оптимальных параметров системы передачи информации на практике сводится к поиску наилучшего компромисса среди ограничений и противоречивых требований к скорости, полосе частот и соотношению энергии бита к спектральной плотности мощности АБГШ.

На рис. 1.6 приведена зависимость минимально необходимого соотношения энергии бита к спектральной плотности мощности АБГШ, требуемого для обеспечения заданной пропускной способности при фиксированной полосе частот, т.е. зависимость . Анализ приведенной зависимости показывает, что повышение требований к пропускной способности повышает минимально необходимое соотношение энергии бита к спектральной плотности мощности АБГШ. Расширение полосы частот напротив снижает это минимально необходимое соотношение. Практическое построение системы передачи информации сводится к выбору оптимальной схемы модуляции и кодирования при накладываемой системе ограничений на отдельные показатели, что позволяет максимально приблизиться к предельной зависимости (поверхности),

приведенной на рис. 1.6.

Рис. 1.6. Зависимость минимально необходимого соотношения энергии бита к спектральной плотности мощности АБГШ, требуемого для обеспечения заданной пропускной способности при фиксированной полосе частот,

.

Введем два дополнительных показателя, характеризующих удельную сложность реализации применяемых процедур кодирования и модуляции:

- минимальное число операций, которые необходимо выполнить для реализации цифровой обработки информационных сообщений, приходящихся на один передаваемый бит данных , операций/бит (асимптотическая временная сложность реализации);

- минимальное число элементов памяти, требуемое для реализации цифровой обработки информационных сообщений, приходящихся на один передаваемый бит данных , элементов/бит (асимптотическая емкостная сложность реализации).

Построение эффективных помехоустойчивых высокоскоростных систем передачи информации на уровне сигнально-кодовых конструкций математически формализуем в виде целевой функции:

. (1.1)

Анализ выражения (1.1) показывает, что построение оптимальных сигнально-кодовых конструкций сопряжено с решением многокритериальной оптимизационной задачи с учетом рассмотренных выше естественных теоретических ограничений на предельную скорость передачи информации и минимально необходимую полосу пропускания. Решение указанной задачи существующими методами невозможно в виду ее чрезвычайно высокой сложности.

Рассмотрим постановку сформулированной выше задачи в условиях принятых в рамках проведения исследования допущений и ограничений. Предположим, что для передачи информации используются каналы связи с фиксированной полосой пропускания , а скорость передачи определяется из предельного теоретического соотношения о минимальной полосе пропускания (ширине полосы частот по Найквисту), т.е.

символов/с

или

битов/с,

где - мощность алфавита символов сообщения, определяемая системой модуляции и кодирования.

Зафиксируем величину (показатель потери достоверности), как величину, задаваемую соответствующими нормативными документами, устанавливающими требования к качеству цифровой связи. Вероятность битовой ошибки является базовым показателем качества связи, максимальное значение которого задает максимально допустимую потерю достоверности для конечного потребителя услуг связи.

Минимально необходимое соотношение энергии бита к спектральной плотности мощности шума , требуемое для обеспечения заданной вероятности , задает величину помехоустойчивости подсистемы передачи информации. Таким образом, задачу построения эффективных помехоустойчивых высокоскоростных систем передачи с учетом принятых в рамках проведения исследований допущений и ограничений запишем в виде:

, (1.2)

т.е. для фиксированной полосы частот и теоретически и предельной (при идеальной фильтрации по Найквисту) скорости передачи требуется минимизировать соотношение энергии бита к спектральной плотности мощности шума , которое необходимо для обеспечения заданного уровня максимальной потери достоверности , где - требуемое (максимально допустимое) значение показателя потери достоверности. Кроме того, в соответствии с постановкой задачи вида (1.2), требуется минимизировать удельную сложность реализации цифровой обработки информационных сообщений с использованием применяемых сигнально-кодовых конструкций.

Задачу построения эффективной помехоустойчивой высокоскоростной подсистемы передачи информации на уровне сигнально-кодовых конструкций будем решать посредством минимизации с учетом принятых допущений и ограничений
и выбора из множества полученных решений (конфигураций) оптимального по критерию минимизации удельной сложности реализации цифровой обработки сообщений ( ) решения.

Таким образом, современная система передачи информации должна обладать заданной помехоустойчивостью и обеспечивать существенное повышение достоверности передаваемой информации.

1.3. Методы повышения достоверности передаваемой информации

Проанализируем известные методы повышения достоверности передаваемой информации и перспективные пути в их развитии.

Методы повышения достоверности передаваемой информации весьма многочисленны и разнообразны, однако все их можно разделить на три группы [122].

К первой группе относятся меры эксплуатационного и профилактического характера, направленные на улучшение качества показателей каналов связи. Эти меры призваны сократить число и уменьшить интенсивность действия источников помех и искажений, вызывающих ошибки при передаче, что достигается рядом технических и организационных мероприятий: улучшением стабильности работы основных узлов системы передачи, схем резервирования; выявлением и своевременной заменой неисправного оборудования и т.п..

Ко второй группе относятся мероприятия, направленные на уменьшение вероятности ошибочного приема элементарного символа за счет повышения отношения энергии сигнала к спектральной плотности мощности шума, а так же путем применения более совершенных методов модуляции, приема, обработки сигналов, а так же рационального выбора мощности, длительности или спектра сигнала сигналов [116, 123-128].

Третья группа методов повышения достоверности передаваемой информации основана на использовании помехоустойчивых кодов, с помощью которых обнаруживаются и исправляются ошибки в принятых информационных сообщениях. Их можно реализовать как в системах без обратной связи, так и в системах с обратной связью, в которых имеются два направления передачи [13-25, 38-42].

Одним из наиболее перспективных и эффективных методов повышения достоверности передаваемой информации в настоящее время является использование специальных процедур, основанных на применении помехоустойчивых кодов. Целью помехоустойчивого кодирования является повышение достоверности передаваемой информации путем обнаружения и исправления ошибок.

Рассмотрим общую постановку задачи помехоустойчивого кодирования. От источника информации поступает последовательность элементарных сообщений . Кодирование заключается в том, что последовательность символов источника заменяется последовательностью двоичных кодовых символов. Такое преобразование является взаимно однозначным, что и позволяет осуществить декодирование, т.е. восстановить сообщение по принятой кодовой комбинации.

Реальный канал может внести в передаваемое сообщение некоторое ко личество ошибок. Для обнаружения этих ошибок приемнику необходимо просто заметить, что они были внесены, а для прямого исправления ошибок (т.е. исправление ошибок без требования повторения передачи) существует дополнительное требование – нужно определить их расположение в принятом сообщении.

Код должен быть определен таким образом, чтобы для большинства возникающих ошибок сообщение можно было восстановить по принятому кодовому слову. Распознавание возможно до тех пор, пока кодовые слова отличаются друг от друга. Задачу формирования различающихся кодовых слов можно решить путем добавления к сообщению избыточной информации.

При синтезе кода, исправляющего ошибки, избыточность, которая добавляется в форме дополнительных символов (в двоичном коде – это дво ичные символы, или биты), следует использовать с осторожностью, поскольку мощность передающего устройства будет расходоваться на передачу избыточных символов, приводя к уменьшению мощности, приходящейся на передачу информационных символов, понижая тем самым величину .

Различие между кодовыми словами можно представить в виде расстояния между ними. Коррекцию ошибок следует выполнять путем просмотра расстояний от полученного кодового слова до всех возможных допустимых кодовых слов, и выбора самого близкого кодового слова. Если это сделано, то ошибки можно всегда исправить, если они разрушают кодовые слова на расстоянии, которое меньше, чем половина расстояния между двумя самыми близкими кодовыми словами, или их можно обнаружить, если они разрушают кодовые слова на расстоянии, которое меньше, чем минимальное расстояние между двумя кодовыми словами.

В общем случае для обнаружения ошибок минимальное кодовое расстояние равно

.

Для кодов, только исправляющих ошибки, минимальное кодовое расстояние равно

где – число исправляемых ошибок.

Для определения кодового расстояния между двумя комбинациями двоичного кода достаточно просуммировать эти комбинации по модулю 2 и подсчитать число единиц в полученной комбинации.

Коррекцию и обнаружение ошибок можно выполнять одновременно. Если минимальное расстояние между любой парой кодовых слов обозначить как (рис. 1.7), то пока исправлять ошибки можно в радиусе , а обнаруживать – в радиусе от каждого кодового слова, т.к. если получен ное сообщение находится вне изображенных кругов, то можно лишь узнать, что имеется ошибка, хотя ее нельзя "исправить", потому что она не находится в круге радиуса вокруг любого кодового слова.

Рис. 1.7. Расстояние между кодовыми словами

Изменяя размер кругов, можно поменять местами возможности исправления и обнаружения ошибок. Заметим, что для фиксированного увеличение уменьшает , и появляется еще одна проблема – возможность неправильного "исправления" полученного сообщения. Если сообщение было так сильно разрушено, что переместилось более чем на и попало в круг радиуса другого кодового слова, то оно будет декодировано неправильно.

Математическим аналогом расстояния является метрика. Самый простой и наиболее общей метрикой для бинарных сигналов является расстояние Хемминга. Расстояние Хемминга между двумя битовыми потоками определяется выражением , где и – это так называемые веса этих битовых потоков, которые определяются числом их ненулевых компонентов.

Существование расстояния между кодовыми словами не обязательно означает, что сообщение можно легко извлечь из полученного кодового слова, даже если случается меньше ошибок, чем код может теоретически исправить . В наихудшем случае нужно будет сравнивать принятый кодовый блок со всеми возможными кодовыми словами, чтобы определить, к какому из них он ближе всего. А при большой длине кодового блока этот процесс может быть продолжительным во времени. Поэтому функция кодирования должна быть выбрана так, чтобы существовал простой метод декодирования.

1.4. Классификация методов помехоустойчивого кодирования

Развитие теории помехоустойчивого кодирования началось с публикаций известного ученого Клода Шеннона [36, 37], в которых показано, что если требуемая от системы связи скорость передачи информации R меньше пропускной способности C, то, используя коды, исправляющие ошибки, можно построить такую систему связи, что вероятность ошибки на выходе будет сколь угодно мала. Практически это означает, что построение слишком хороших каналов (на сигнальном уровне) экономически менее выгодно, чем использование помехоустойчивого кодирования. Последнее утверждение явилось мощным толчком для развития теории кодов, исправляющих ошибки [13-25, 38-42].

Все помехоустойчивые коды можно разделить на два класса: непрерывные и блочные. В непрерывных кодах процесс кодирования и декодирования носит непрерывный характер. В блочных кодах каждому сообщению соответствует кодовая комбинация (блок) из n символов. Блоки кодируются и декодируются отдельно друг от друга.

Равномерные коды – это коды, все кодовые комбинации которых содержат постоянное количество разрядов. Неравномерные коды содержат кодовые комбинации с различным числом разрядов. Ввиду того что неравномерные избыточные коды не нашли применения на практике из-за сложности их технической реализации, их рассматривают очень редко.

Помехоустойчивые коды, в которых определенные разряды кодовых комбинаций отводятся для информационных и проверочных символов, называются разделимыми. Разделимые блочные коды обозначаются обычно (n, k) – кодами, где n – количество разрядов кодовой комбинации, k – число разрядов, отводимых для информационных символов. Неразделимые коды не имеют четкого разделения кодовой комбинации на информационные и проверочные символы. К ним относятся коды с постоянным весом и коды Плоткина.

Разделимые блочные коды, в свою очередь, делятся на несистематические и систематические. Самый большой класс разделимых блочных кодов составляют систематические коды, у которых проверочные символы определяются в результате проведения линейных операций над определенными информационными символами. Для двоичных кодов эти операции сводятся к выбору каждого проверочного символа таким образом, чтобы его сумма по модулю два с определенными информационными символами была равной нулю. К систематическим кодам относятся коды с проверкой на четность, коды с повторением, корреляционный, инверсный, коды Хэмминга, Голея, Рида–Маллера, Макдональда, Варшамова, с малой плотностью проверок на четность, итеративный код. Также к систематическим кодам относится часть циклических кодов. Кроме всех свойств систематического кода, циклические коды имеют следующее свойство: если некоторая кодовая комбинация принадлежит коду, то получающаяся путем циклической перестановки символов новая комбинация также принадлежит данному коду. К наиболее известным циклическим кодам относятся простейшие коды, коды Хэмминга, Боуза-Чоудхури-Хоквингема, мажоритарные, коды Файра, Абрамсона, Миласа-Абрамсона, Рида-Соломона, компаундные коды. Классификация рассмотренных кодов приведена на рис. 1.8.

Использование помехоустойчивых кодов для повышения достоверности передаваемой информации требует учета различных факторов: распределение ошибок в канале связи; допустимую вероятность ошибок кодовой последовательности; обеспечение заданной скорости передачи информации; сложность алгоритмов кодирующих и декодирующих устройств. Одним из основных факторов, влияющих на окончательный выбор кода, является характер распределения ошибок в канале связи.

В табл. 1.1. приведены некоторые коды и их корректирующие способности. Из таблицы видно, что большинство избыточных кодов исправляет только независимые ошибки. Лишь небольшая группа кодов позволяет исправлять пакеты ошибок.

По своей структуре все помехоустойчивые коды делятся на линейные и нелинейные. На рис. 1.8. схематично изображена классификация наиболее известных методов помехоустойчивого кодирования. Их развитие шло по двум основным направлениям. Первое направление в развитии теории кодов, контролирующих ошибки, базируется на алгебраических методах и, преимущественно, оперирует блоковыми кодами. Наибольшее

распространение среди блоковых кодов нашли коды с проверкой на четность, с повторением символов, равновесные коды, коды Хемминга, коды Рида-Малера и, наиболее обширный класс кодов – циклические коды. К последнему классу принадлежат коды Боуза-Чоудхури-Хоквингема (БЧХ) и коды Рида-Соломона. Наиболее полно основы алгебраической теории кодов изложены в [13-18].

Второе направление носило вероятностный характер и привело к появлению неблоковых (непрерывных) кодов бесконечной длины. К непрерывным кодам относятся древовидные коды. Линейные постоянные во времени древовидные коды являются сверточными кодами. Отличительной особенностью сверточных кодов является возможность их описания деревом или решетчатой диаграммой. Кодер сверточного кода представляет собой линейный регистр сдвига, сложность которого не зависит от длины кода, что является значительным преимуществом.

Таблица 1.1

Исправляющая способность некоторых помехоустойчивых кодов

Код	Ошибки			
	Независимые		Групповые	
	Обнаружение	Исправление	Обнаружение	Исправление
С проверкой на четность	+			
С повторением символов	+			
С постоянным весом (равновесные)	+			
Коды Рида-Малера	+	+		
Коды Хемминга	+	+		
Код Голея	+	+		
Коды Файра			+	+
БЧХ	+	+	+	
Коды Гоппы	+	+	+	
Рида-Соломона	+	+	+	+
Алгеброгеометрические	+	+	+	+
Сверточные	+	+	+	+

По критерию минимизации вероятности ошибки при фиксированном соотношении энергии сигнала к спектральной плотности шума сверточные коды являются наиболее эффективными по сравнению с блочными кодами. В тоже время большинство известных методов синтеза сверточных кодов основаны на сложных процедурах переборного поиска и при больших параметрах синтезируемых кодов вычислительно не реализуемы.

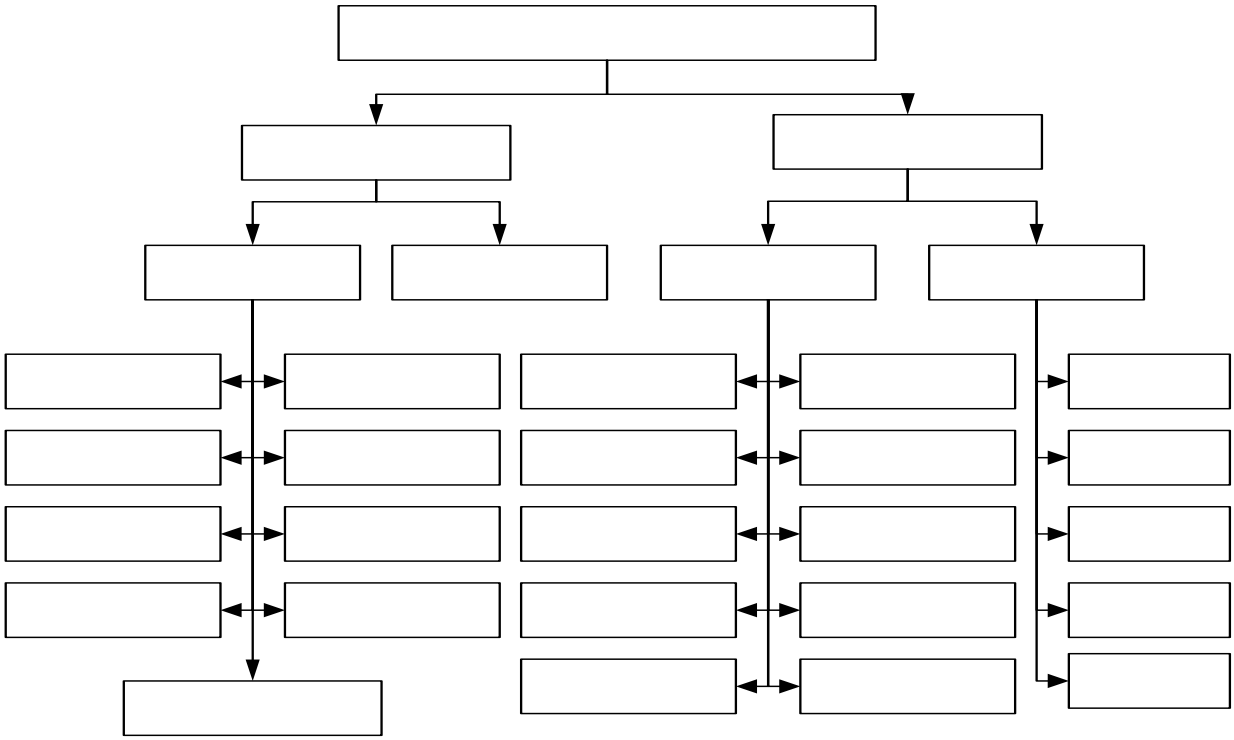


Рис. 1.8. Классификация помехоустойчивых кодов

При высоких требованиях к достоверности передаваемой информации реализация кодирующих и декодирующих устройств является затруднительной из-за их высокой сложности. Появление методов каскадного кодирования связано с необходимостью значительного упрощения алгоритмов декодирования и одновременным повышением их эффективности. Отличительной особенностью каскадных схем кодирования является кодирование (декодирование) информации несколькими составляющими кодерами (декодерами). Каскадные коды позволяют обеспечить высокую достоверность в условиях большого уровня шума при умеренной сложности декодирования. Дальнейшее совершенствование методов каскадного кодирования привело к разработке турбокодов [38-40].

1.5. Методы синтеза непрерывных кодов

Одним из перспективных направлений в развитии теории помехоустойчивого кодирования является разработка методов синтеза непрерывных (древовидных) кодов [13-15, 19-25]. Суть этих методов состоит в представлении информационного потока данных блоками (кадрами) длины k^0 и сопоставлении с каждым из них блока кодовых символов. При этом каждый полученных кадр кодовых символов формируется с учетом предыдущих r кадров информационных символов.

На рис. 1.9. представлена обобщенная структурная схема непрерывного кодера, построенного в виде цепи регистров сдвига, соединенных логическими связями. Работа кодера, представленного на рис. 1.9., состоит в следующем.

Информационная последовательность вводится в кодер, начиная с нулевого момента времени и до бесконечности. Поток входящих информационных символов разбивается на кадры по k^0 символов каждый. Кадр может в частности состоять из одного символа, в кодере хранится r кадров. В течение каждого момента времени в регистр сдвига вводится новый кадр информационных символов. Кодер по введенному кадру и r хранящихся в нем кадрам вычисляет один кадр кодового слова, имеющий длину n^0 символов. Этот кадр кодового слова выводится из кодера, как только следующий кадр информационных символов поступает в него. Следовательно, каждым k^0 информационным символам соответствуют n^0 кодовых символов.

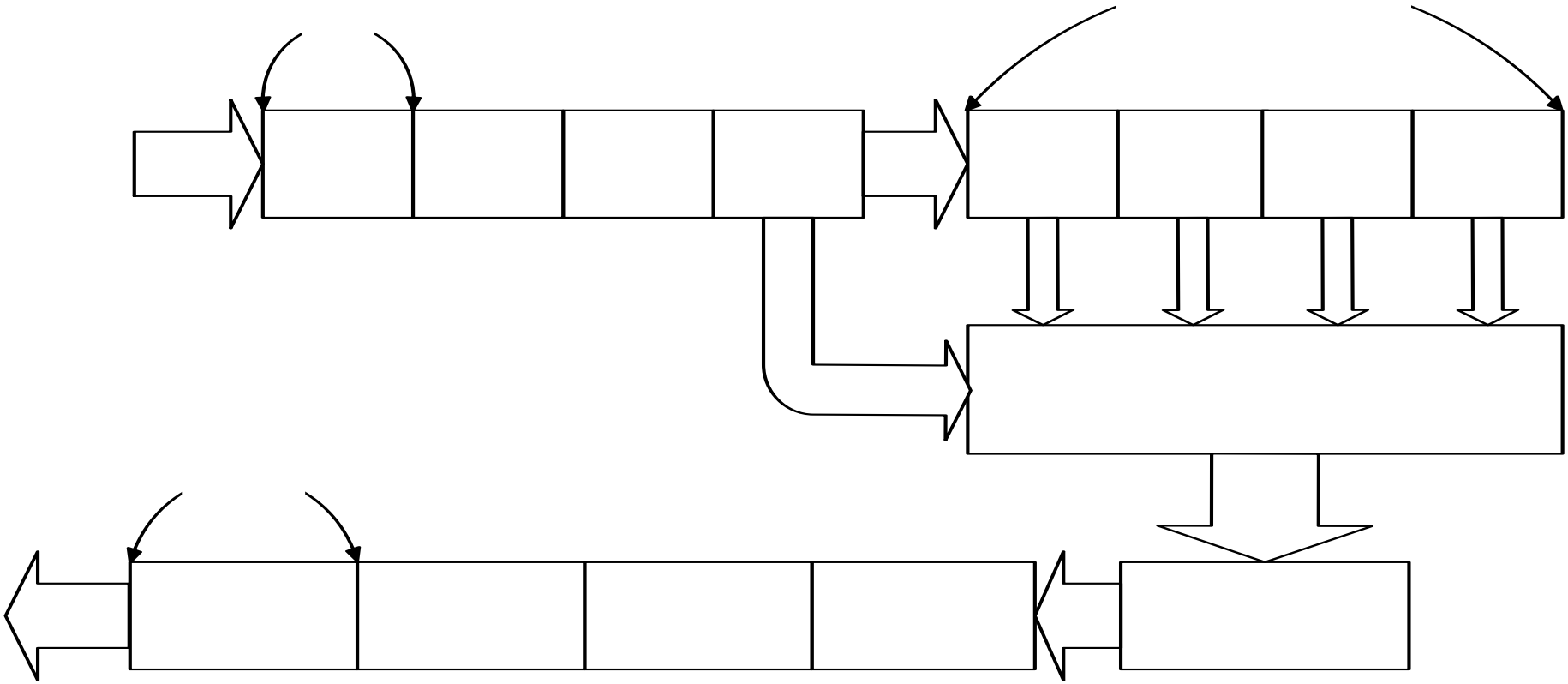


Рис. 1.9. Непрерывный кодер в виде регистра сдвига

Бесконечное множество всех бесконечно длинных кодовых слов, получаемых при поступлении в кодер всех возможных входных последовательностей, называется древовидным (непрерывным) (n^0, k^0) – кодом. Скорость R этого кода определяется как $R = k^0 / n^0$.

Важной характеристикой непрерывного кода является также величина $\nu = r \cdot k^0$, называемая длиной кодового ограничения. Минимальным кодовым расстоянием непрерывного кода d называется минимальное расстояние для любых различных кодовых слов, соответствующих $r + 1$ различным информационным кадрам с ненулевым начальным кадром. Если непрерывный код линеен, то минимальное расстояние равно минимальному весу из всех ненулевых кодовых слов, соответствующих произвольной входной последовательности с ненулевым начальным кадром. Набор минимальных весов d_l, $l = 1, 2, 3, \ldots$ произвольных кодовых слов, соответствующих l различным информационным кадрам с ненулевым начальным кадром называется дистанционным профилем непрерывного кода . Свободным расстоянием непрерывного кода называется $d_\infty = max(d_l)$. Очевидно, что $d = d_{r+1} \le d_{r+2} \le \ldots \le d_\infty$. При декодировании непрерывных кодов синдромными и пороговыми методами пользуются величиной d [14]. При декодировании методом Витерби пользуются величиной d_∞ [14, 22].

При построении непрерывных кодов используют также другие параметры кода [14]. Так, величина

$$k = (r + 1) \cdot k^0$$

непосредственно связана с длиной кодового ограничения и называется информационной длиной слова непрерывного кода. Соответствующая ей мера кодовых последовательностей называется длиной кодового блока:

$$n = (r + 1) \cdot n^0 = k \cdot n^0 / k^0.$$

Кодовая длина блока – это длина кодового слова, на которой сохраняется влияние одного кадра информационных символов. В большинстве известных практических примерах значения k^0 и n^0 выбираются равными небольшим целым числам, как правило, $k^0 = 1$. Это означает, что выбор скорости ограничен: $R = 1 / m$, $m \in \{1, 2, \ldots\}$ – в большинстве известных примеров невозможно построить непрерывный код со скоростью, достаточно близкой к единице, как это делается для большинства блоковых кодов (циклические коды БЧХ, Рида-Соломона и др.).

На практике нашли применение несколько классов непрерывных кодов . Их общая классификация представлена на рис. 1.10.

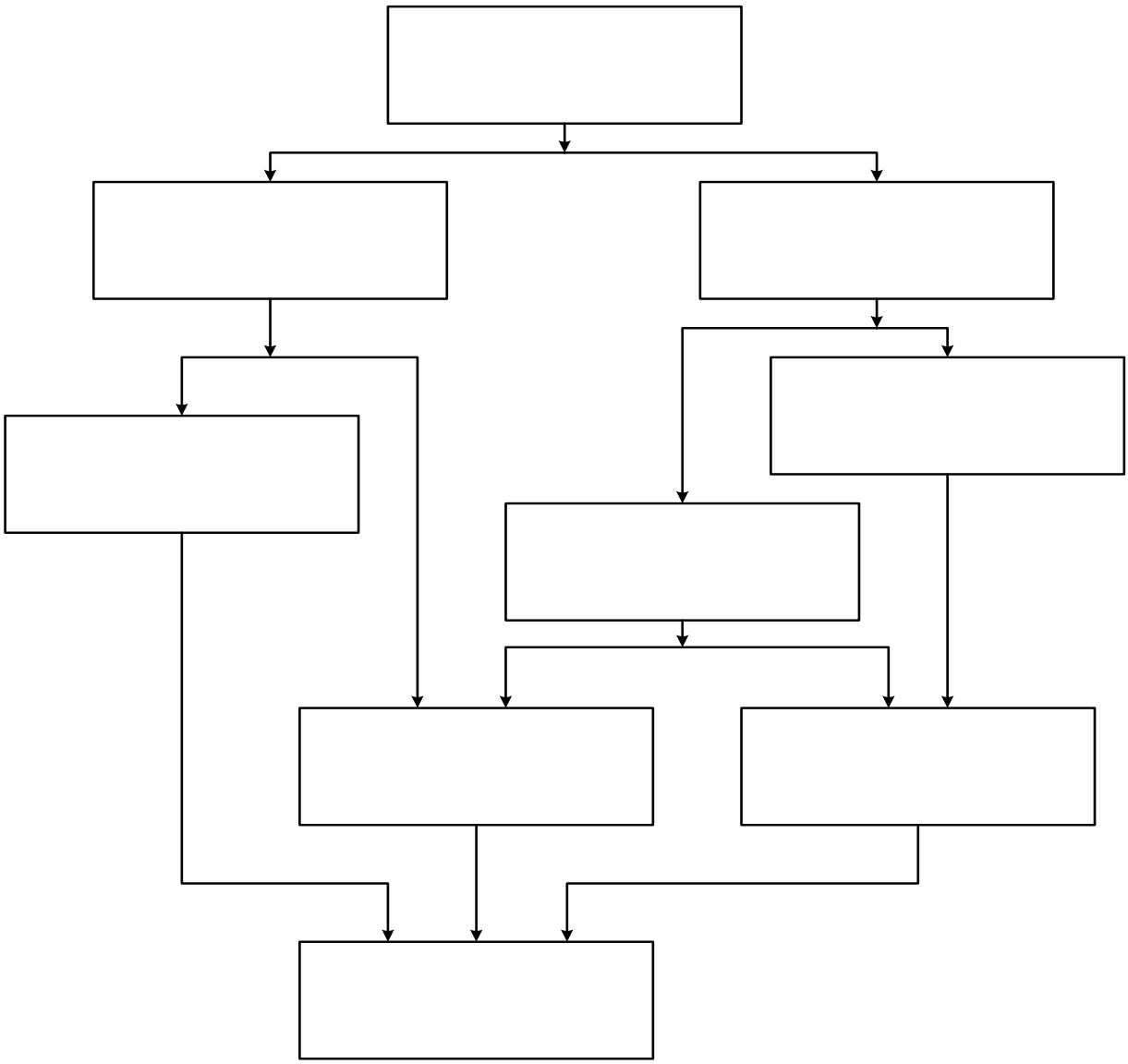


Рис. 1.10. Классификация непрерывных кодов

Частные случаи древовидных кодов получаются различными комбинациями следующих свойств.

1. *Конечность длины кодового ограничения.* Практические непрерывные коды всегда имеют конечную длину кодового ограничения. Древовидный (n^0, k^0) код с конечной длиной кодового ограничения ν, длиной слова $\nu + k^0$ называется так же решетчатым кодом.

2. *Постоянство во времени.* Если две различные входные последовательности совпадают во всем, но с временным сдвигом на целое число кадров, то соответствующие им кодовые последовательности также совпадают во всем, но с временным сдвигом на то же самое число кадров.

3. *Линейность*. Кодовая последовательность любой линейной комбинации двух информационных последовательностей совпадает с такой же линейной комбинацией кодовых последовательностей этих двух информационных последовательностей. Иначе говоря, если i_1 и i_2 являются двумя информационными последовательностями с кодовыми словами $c(i_1)$ и $c(i_2)$, то $a \cdot i_1 + b \cdot i_2$ соответствует кодовая последовательность $c \cdot (a \cdot i_1 + b \cdot$

$i_2) = a \cdot c \cdot (i_1) + b \cdot c \cdot (i_2)$.

4. *Систематичность*. Каждый кадр информационных символов составляет первые k^0 символов первого из тех кадров кодовой последовательности, на которые влияет данный кадр информационной последовательностей.

Наиболее важным классом непрерывных кодов являются сверточные коды, обладающие свойством линейности и постоянства во времени. Опыт практического применения сверточных кодов показывает, что рассмотренные методы позволяют обеспечить эффективный контроль ошибок в каналах с независимыми и группирующимися ошибками.

1.6. Эффективность сверточных кодов в каналах с группирующимися ошибками

Проведем анализ влияния группирующихся ошибок на процесс передачи информации с помехоустойчивым кодированием, для чего представим канал связи как канал с белым гауссовым шумом (АБГШ), в котором время от времени возникают большие шумовые или интерференционные всплески, связанные например, с замираниями или преднамеренными помехами [20]. Предположим, что используется двоичная фазовая манипуляции (ФМ). Близкая к оптимальной стратегия получения информации о правдоподобии символов при наличии пакетов ошибок, вызванных интерференцией, состоит в вычеркивании затронутых интерференцией символов и объявлении их стертыми [20].

Далее будем рассматривать именно такую стратегию, предполагая, что длина пакета больше продолжительности одного символа и что вычеркиваются только полные символы (целиком).

Рассмотрим процесс, приводящий к возникновению шумовых пакетов. В отсутствие пакетов канал моделируется простым каналом с АБГШ и характеризуется некоторым значением . Предполагается, что при появлении пакета шум окажется столь большим, что демодулятор его легко обнаруживает и вычеркивает соответствующие символы. В результате возникает пакет стертых символов. Будем считать, что длина пакета фиксирована либо известна максимальная длина.

Будем рассматривать пакеты стираний появляющиеся периодически и случайно. Пусть пакеты стираний появляются периодически. В этом случае пакет, состоящий из стертых символов, появляется периодически через каждые символов (параметр будем называть периодом, - длиной пакета и - интенсивностью). При случайных пакетах стираний пакеты состоят из фиксированного числа символов со средней интенсивностью и с экспоненциально распределенными интервалами времени между соседними пакетами.

Рассмотрим вначале частный случай, соответствующий случайным стираниям, когда каждый символ стирается с вероятностью и все стирания

статистически ствующие
сверточному к дшение
характеристик ,
необходимое для поддержания значения) для кодов со случайными стираниями показано на рис. 1.12, из анализа которого следует, что с уменьшением скорости кода существенно улучшаются характеристики.

Рис. 1.11. Кривые вероятности ошибки сверточного кода с , при случайных стираниях

Аналогичных результатов можно достичь, сохранив постоянной скорость кода и увеличив длину кодового ограничения (см. кривые для на рис. 1.12). Ухудшение характеристик объясняется уменьшением веса кодовых слов из-за стираний символов (декодер не может использовать кодовое расстояние, накопленное на стертых позициях, для различения между собой кодовых слов), а также уменьшением энергии принятых сигналов. Так, если доля стертых симво
сигналов составляет лишь -ю част
принята в отсутствие стираний. Поэтому
уменьшается на .

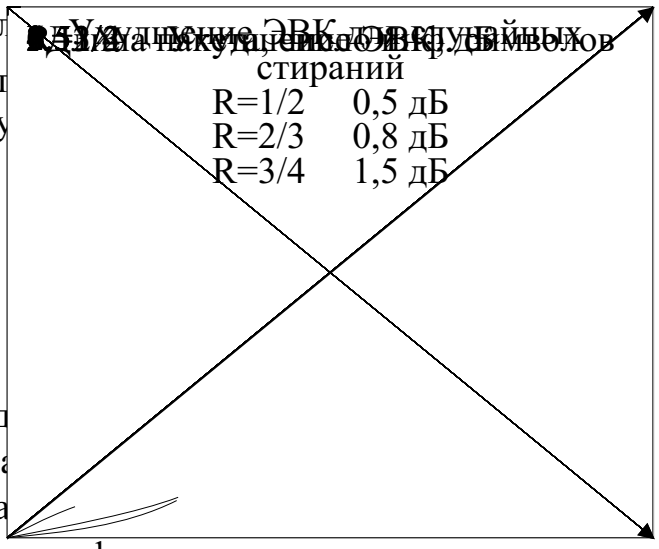


Рис. 1.12. Ухудшение ЭВК (

Предположим, что входная послед
влиянию периодического процесса стира
периодом . Зависимость ухудшения ха
различными значениями , от числа информационных символов в пакете, которое определялось как , представлена на рис. 1.13.

Наилучшие характеристики соответствуют случаю . Для пакетов стираний, длина которых не превышает 5, характеристики ухудшаются сравнительно медленно и они сравнимы с характеристиками для случайных стираний при той же интенсивности. Таким образом, ухудшение сравнительно невелико при длинах пакета стираний, не превышающих длины кодового ограничения.

Рис. 1.13. Ухудшение ЭВК для сверточного кода с при (
)

Это подтверждают и результаты анализа рис. 1.14, на котором представлена зависимость ухудшения характеристик при от длины

пакета при для кодов с и различными . Кроме того, из анализа рис. 1.13 и рис. 1.14 следует, что увеличивать длину кодового ограничения гораздо выгоднее, чем уменьшать скорость кода (при той же информационной скорости). Такое поведение отлично от результатов для случайных стираний, представленных на рис. 1.12.

Проведенные исследования в [20] показали, что короткий пакет стираний не является особенно опасным, если только в непосредственной близости за ним не следует другой пакет стираний. Однако при случайных пакетах ошибок вероятность появления двух близких пакетов весьма высока. В [20] показано, что ухудшение характеристик при случайных пакетах ошибок незначительно только при их малой интенсивности или длине.

Рис. 1.14. Ухудшение ЭВК для сверточных кодов с при ( )

Таким образом, сверточные коды позволяют помимо независимых ошибок исправлять пакеты ошибок. При этом длина исправляемого пакета ошибок определяется кодовым ограничением сверточного кода. С приемлемой для практики точностью можно принять длину исправляемого пакета ошибок , которая в реальных условиях может отличаться от указанного значения из-за статистики расположения пакетов ошибок относительно друг друга. Однако для получения заданной достоверности передачи информации в каналах с группирующимися ошибками необходимы сверточные коды с большим кодовым ограничением, построение которых переборными методами затруднено по причине их низкой производительности (экспоненциальный рост вычислительной сложности в зависимости от значения кодового ограничения). Поэтому необходимы алгебраические методы построения сверточных кодов с большим кодовым ограничением.

Одним из достоинств сверточных кодов является возможность построения на их основе длинных каскадных кодов, включая турбокоды. Отличительной особенностью турбокодера является наличие перемежителя между составляющими сверточными кодерами. Турбокоды, в отличие от других каскадных кодов, позволяют получить на практике устройства кодирования и декодирования приемлемой сложности, реализующие выигрыш от кодирования близкий к максимально возможному. Их практическое использование, при условии выбора соответствующих сверточных кодов, позволит исправлять сложные комбинации коррелированных ошибок.

Проведем исследование потенциальных возможностей турбокодов и сравним их со сверточными кодами, получившими наибольшее распространение в телекоммуникационных системах, работающих при низком энергетическом отношении сигнал/шум.

1.7. Методы параллельного каскадирования сверточных кодов

При высоких требованиях к достоверности передаваемой информации и низком энергетическом отношении сигнал/шум, например, в спутниковых каналах связи, реализация кодирующих и декодирующих устройств широко распространенных помехоустойчивых кодов является затруднительной из-за высокой сложности декодирования, под которой будем понимать количество операций декодирования, приходящееся на один информационный символ.

В этом случае целесообразно применение каскадных кодов, декодирующие устройства которых могут быть реализованы на практике. Наиболее эффективными каскадными кодами являются турбокоды. В отличие от известных последовательных каскадных кодов [38-40], турбокоды позволяют для их декодирования применять процедуру итеративного декодирования. При этом оказывается возможной передача информации при энергетическом отношении сигнал/шум близким к предельно возможному значению, определяемому теоремой Шеннона [36, 37], поскольку характеристики итеративного декодера турбокодов близки к декодеру максимального правдоподобия [7-11].

Проведем исследование возможности повышения достоверности передачи информации в телекоммуникационных системах на основе применения турбокодов.

Турбокод представляет собой параллельный каскадный код, образованный двумя или более составляющими кодами [41-42]. Схема турбокодера с двумя составляющими кодерами со скоростью кодирования представлена на рис. 1.15.

Пусть – двоичная входная информационная последовательность, которая поступает одновременно на вход первого кодера, на перемежитель и на выход турбокодера. На выходе кодера 1 формируется последовательность проверочных символов . Перемеженная информационная последовательность поступает на вход другого кодера. На выходе кодера 2 формируется последовательность проверочных символов .

Общая скорость рассматриваемого турбокода – . Для повышения общей скорости кодирования до проверочные последовательности поступают на схему выкалывания, которая производит поочередное удаление проверочных символов.

В качестве составляющих кодеров наиболее часто используют систематические рекурсивные сверточные кодеры (РСК), что связано с относительной простотой реализации мягкого декодирования сверточных кодов [46] и особенностями весового распределения кодовых слов [47, 48].

Рис. 1.15. Схема турбокодера

Так как турбокод и составляющие его коды являются линейными, то вместо рассмотрения дистанционного спектра можно рассматривать весовой спектр кодов [16]. При этом вес ненулевого кодового слова будет представлять собой расстояние Хэмминга от нулевого слова. Количество ошибок, которые может исправить код, определяется минимальным расстоянием Хэмминга – наименьшим числом позиций, на которые отличаются любые два кодовых слова.

Основные свойства рекурсивных сверточных кодов [48-50].

– Входные полубесконечные последовательности единичного веса , где порождают пути в решетчатой диаграмме, которые будут расходиться от нулевого пути, но никогда не сойдутся с ним, поскольку многочлен не делится без остатка на . Такие пути и соответствующие им проверочные (или кодовые) последовательности будем называть путями (последовательностями) бесконечного веса.

– Для любого существуют полубесконечные входные последовательности веса 2 вида , , (, если примитивный многочлен степени), порождающие пути в решетчатой диаграмме, которые расходятся от нулевого пути и сходятся с ним в некоторый момент времени (если делится на). Такие пути и соответствующие им проверочные (или кодовые) последовательности будем называть путями (последовательностями) конечного веса.

Таким образом, если некоторая информационная последовательность на выходе кодера 1 порождает проверочную последовательность конечного веса, то перемеженная версия этой информационной последовательности, подаваемая на вход кодера 2, с высокой вероятностью приведет к генерации проверочной последовательности бесконечного веса из-за указанных выше свойств РСК. Если какая-либо комбинация ошибок не может быть исправлена одним РСК, то это с высокой вероятностью будет сделано с помощью проверочной последовательности другого РСК и наоборот. При использовании нерекурсивных сверточных кодов перемеженная последовательность веса 2 всегда будет являться последовательностью конечного веса, поэтому выигрыш от кодирования будет значительно меньшим [47, 48].

Верхняя граница вероятности ошибки на бит для турбокодов (декодирование по максимуму правдоподобия) определяется выражением [50, 129]

, (1.3)

где – количество кодовых слов с общим весом ;

– средний информационный вес кодового слова с общим весом ;

.

Аппроксимируем (1.3) первым слагаемым, учитывая только кодовые слова с минимальным расстоянием

, (1.4)

где – количество кодовых слов веса ;

– средний вес информационных символов в кодовом слове веса .

Обычно выбор кода осуществляется путем максимизации . Однако, для турбокодов акцент делается на минимизации коэффициента в (1.3) и (1.4) за счет выбора структуры перемежителя [50, 129].

Перемежитель должен гарантировать, чтобы информационное слово, порождающее кодовое слово с конечным весом проверочной последовательности, после перемежения порождало на выходе другого составляющего сверточного кодера кодовое слово с бесконечным весом проверочной последовательности. Так, для кодовых слов конечного веса

.

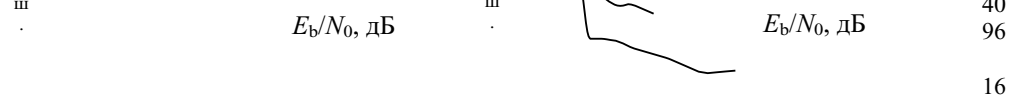
Рассмотрим для примера турбокод с двумя составляющими рекурсивными сверточными кодами с порождающими многочленами , и с (выкалывание). В [130] найдено, что , , . Вероятность ошибки на бит для данного кода:

. (1.5)

Рассмотрим сверточный код с , и порождающими многочленами . Для указанного сверточного кода . Верхняя граница вероятности ошибки [131]:

. (1.6)

Из анализа выражений (1.5) и (1.6) следует, что аргумент Q функции сверточного кода больше, чем турбокода, что приводит к большей крутизне кривой вероятности ошибки сверточного кода по сравнению с турбокодом. Однако коэффициент перед Q функцией намного больше для сверточного кода, чем для турбокода, поэтому кривая вероятности ошибки турбокода будет лежать ниже кривой вероятности ошибки сверточного кода.



Анализируя кривые вероятности ошибки сверточного кода и турбокода, представленные на рис. 1.16, можно сделать вывод, что при малом значении турбокоды имеют преимущества перед сверточными кодами.

SHAPE * MERGEFORMAT

Рис. 1.16. Кривые вероятности ошибки турбокода, сверточного кода и результаты моделирования:

1 – результаты моделирования сверточного кода;
2 – кривая вероятности ошибки сверточного кода;
3 – результаты моделирования турбокода;
4 – кривая вероятности ошибки турбокода.

С увеличением обе асимптоты приближаются друг к другу из-за их различной крутизны и при отношении дБ пересекаются. Поэтому при большом значении сверточные коды имеют преимущества перед турбокодами, что связано с меньшим значением турбокодов по сравнению со сверточными кодами.

Из (1.4) также следует, что обратно пропорциональна . Поэтому кривая вероятности ошибки турбокода будет понижаться с увеличением (см. рис. 1.16).

Таким образом, принципиальное отличие турбокодов от известных каскадных кодов заключается в применении составляющих рекурсивных сверточных кодов совместно с процедурой перемежения, что позволяет обеспечить малое количество кодовых слов минимального веса путем выбора структуры перемежителя.

Эффективность кодирования возрастает с увеличением длины информационной последовательности. Реализация турбокодирования информации блоками большой длины не представляет собой значительных трудностей из-за использования составляющих сверточных кодов, поскольку сложность сверточного кодирования не зависит от длины кодируемой информационной последовательности. В результате, турбокоды могут обеспечить высокую эффективность кодирования при низком энергетическом отношении сигнал/шум.

Недостатком турбокодов является уменьшение эффективности кодирования при высоком энергетическом отношении сигнал/шум, что связано с малым минимальным расстоянием турбокодов. Кроме того, применение процедуры выкалывания для увеличения относительной скорости кодирования также приводит к уменьшению минимального расстояния турбокода.

Таким образом, видимым путем устранения недостатков турбокодов является использование в качестве составляющих турбокод кодов рекурсивных сверточных кодов с большим значением и , что

приведет к повышению минимального расстояния турбокода и позволит выбирать скорость турбокодирования в широких пределах без применения выкалывания. Практическая реализация алгоритма итеративного декодирования турбокодов с большим значением затруднена из-за высокой сложности декодирования, поскольку алгоритмы мягкого декодирования используют только решетчатое представление сверточного кода [19, 20].

Подавляющее большинство хороших и наиболее употребимых сверточных кодов получено переборным методом. Основным его недостатком является быстрый рост вычислительных затрат. Так, для перебора всех двоичных сверточных кодов с кодовым ограничением 100 бит необходимо выполнить перебор $2^{100} \approx 10^{30}$ различных вариантов и выбрать лучший из них, что является практически неразрешимой задачей. На практике переборными методами реализован поиск хороших двоичных сверточных кодов до $\nu \leq 14$. Очевидно, что с практической точки зрения, переборный метод построения сверточных кодов малоэффективен по причине своей низкой производительности. Актуальным направлением является разработка и исследование алгебраических методов построения сверточных кодов.

Известный алгебраический метод построения сверточных кодов состоит в представлении сверточного кода через порождающий многочлен недвоичного циклического кода, что позволяет алгебраически задавать его параметры для скорости $R = 1 / n$. Этот подход позволяет использовать мощный математический аппарат циклического кодирования в целях алгебраического построения сверточных кодов. Однако ограничения по скорости кодирования ($R = 1 / n$) препятствуют его широкому практическому использованию.

Таким образом, необходима разработка алгебраических методов синтеза сверточных кодов, свободных от указанных недостатков и позволяющих алгебраически задавать коды с $R = k^0 / n$, методов синтеза схем турбокодирования на основе алгебраически заданных сверточных кодов, а также разработка методов мягкого декодирования, учитывающих алгебраическую (а не решетчатую) структуру сверточных кодов, что приведет к уменьшению сложности декодирования турбокодов с большим значением кодового ограничения .

1.8. Постановка задач на исследование

Проведенный анализ показал, что развитая в настоящее время алгебраическая теория блочного кодирования не может быть непосредственно применена к сверточным кодам по причине значительного различия в их свойствах по сравнению с блочными кодами. Однако существует возможность представления сверточного кода в виде блочного кода полубесконечной длины и его последующим алгебраическим описанием . Положительные результаты в этом направлении теории помехоустойчивого кодирования получены только для ограниченного диапазона низких

скоростей кодирования, значения которых не удовлетворяют современным требованиям, предъявляемым к параметрам помехоустойчивых кодов (как правило, на практике требуются более высокие скорости кодирования). Кроме того, отсутствуют исследования возможности применения алгебраической теории для реализации декодирования сверточных кодов. Таким образом, возникает научная проблема (противоречивая ситуация), в которой существующие положения теории помехоустойчивого кодирования не позволяют эффективно (вычислительно реализуемо) решать задачи синтеза и декодирования сверточных кодов с большим кодовым расстоянием (с высокими конструктивными кодовыми характеристиками, с произвольными параметрами). Разрешение научной проблемы (противоречивой ситуации) возможно путем решения следующих научных задач.

1. Разработать и исследовать методы синтеза алгебраически заданных сверточных кодовых конструкций кодов для повышения достоверности передаваемой информации:

– разработать (с использованием математического аппарата алгебраической теории кодов) методы синтеза алгебраически заданных сверточных кодов, теоретически обосновать аналитические выражения по оценке кодовых соотношений синтезируемых кодов;

– разработать методы и алгоритмы кодирования алгебраически заданными сверточными кодами, исследовать конструктивные свойства синтезированных сверточных кодовых конструкций.

2. Разработать и исследовать вычислительно эффективные (вычислительно реализуемые) методы и алгоритмы декодирования алгебраически заданных сверточных кодов:

– разработать (с использованием методов алгебраической теории кодов , элементов корреляционного и спектрального анализа) алгебраический метод декодирования синтезируемых (алгебраически заданных) сверточных кодов;

– разработать комбинированный метод декодирования алгебраически заданных сверточных кодов, объединяющий в себе процедуры переборного поиска по кодовой решетке и алгебраические процедуры локализации и исправления ошибок;

– разработать (вычислительные) алгоритмы декодирования алгебраически заданных сверточных кодов и предложения по программной и аппаратной реализации.

3. Разработать параллельные каскадные сверточные кодовые конструкции на основе алгебраически заданных рекурсивных сверточных кодов и (вычислительно эффективных) (вычислительно реализуемых) алгоритмов их декодирования:

– аналитически формализовать и разработать методы синтеза турбокодов с использованием алгебраически заданных сверточных кодов;

– разработать и исследовать алгоритмы итеративного декодирования параллельных каскадных кодовых конструкций с алгебраически заданными

сверточными кодами;

– разработать и исследовать алгоритмы мягкого декодирования составляющих турбокод алгебраически заданных сверточных кодов.

4. Разработать практические рекомендации по использованию алгебраических сверточных кодов в телекоммуникационных системах и сетях:

– разработать (с использованием методов математической статистики и проверки гипотез) методику оценки и исследовать достоверность передаваемой информации в телекоммуникационных системах и сетях с использованием алгебраически заданных сверточных кодов и турбокодов на их основе;

– обосновать практические рекомендации по использованию алгебраических сверточных кодов в телекоммуникационных системах и сетях.

Сформулированные задачи в совокупности решают важную научную проблему, имеющую большое значение как для развития отдельного направления теории помехоустойчивого кодирования, так и для решения прикладных вопросов, связанных с обеспечением заданной достоверности передаваемой информации в телекоммуникационных системах и сетях.

Выводы

1. Проведенный анализ модели подсистемы передачи информации показал, что ее формальное аналитическое описание задается совокупностью операторов формирования, обработки и передачи информации, а так же оператором воздействия шума на передаваемую информацию в канале связи. При оценке параметров подсистемы передачи информации необходимо использовать математические модели каналов связи, которые описывают характер ошибок и шумов в канале связи.

2. Одной из важнейших характеристик подсистемы передачи информации является достоверность, под которой понимают свойство подсистемы, характеризующее ее способность обеспечивать точное воспроизведение передаваемой информации в пункте приема. С достоверностью тесно связана другая характеристика подсистемы передачи информации – помехоустойчивость, под которой понимают способность подсистемы передачи обеспечивать передачу информации в условиях взаимодействия помех.

3. Проведенные исследования позволили обобщить критерии и показатели эффективности подсистемы передачи информации и вскрыть научно-практическое противоречие между постоянно возрастающими требованиями к достоверности и помехоустойчивости и возможностями, применяемых на практике методов и технических средств передачи и обработки информации.

4. Проведенное исследование показало, что вероятность ошибки на кодовую комбинацию в телекоммуникационных сетях общего пользования должна находиться в пределах в зависимости от реализуемого класса обслуживания, а в телекоммуникационных сетях специального назначения вероятность ошибки на кодовую комбинацию не должна превышать 10^{-7} - 10^{-8}. Однако в реальных каналах связи указанная вероятность не превышает $P_{ош} > 10^{-2} – 10^{-4}$ (вероятность ошибки на кодовую комбинацию на выходе демодулятора), что не удовлетворяет современным требованиям.

5. Из проведенного анализа следует, что методы помехоустойчивого кодирования являются основным и наиболее эффективным механизмом повышения помехоустойчивости подсистем передачи информации, а с верточные коды являются наиболее эффективными по критерию минимизации вероятности ошибки при фиксированном соотношении энергии сигнала к спектральной плотности шума. Наряду с простотой аналитического описания алгоритмов кодирования и декодирования, возможностью эффективной программной и аппаратной реализации данный класс непрерывных кодов позволяет получить наибольший энергетический выигрыш от кодирования.

6. Проведенные исследования показали, что сверточные коды позволяют помимо независимых ошибок исправлять пакеты ошибок. При этом длина исправляемого пакета ошибок определяется кодовым ограничением сверточного кода. С приемлемой для практики точностью можно принять длину исправляемого пакета ошибок равной половине длины кодового ограничения, которая в реальных условиях может отличаться от указанного значения из-за статистики расположения пакетов ошибок относительно друг друга. В тоже время для получения высокой помехоустойчивости в каналах с группирующимися ошибками необходимы сверточные коды с большим кодовым ограничением, построение которых переборными методами затруднено по причине их низкой производительности (экспоненциальный рост вычислительной сложности в зависимости от значения кодового ограничения).

7. Проведенные исследования показали, что наибольшую эффективность сверточные коды обеспечивают в параллельных каскадных схемах (турбокоды). Принципиальное отличие турбокодов от известных каскадных кодов заключается в применении составляющих рекурсивных сверточных кодов совместно с процедурой перемежения, что позволяет обеспечить малое количество кодовых слов минимального расстояния путем выбора структуры перемежителя.

8. Эффективность турбокодирования возрастает с увеличением длины информационной последовательности. Реализация турбокодирования информации блоками большой длины не представляет значительных трудностей из-за использования составляющих сверточных кодов, поскольку сложность сверточного кодирования не зависит от длины кодируемой

информационной последовательности. В результате, турбокоды могут обеспечить высокую эффективность кодирования при низком энергетическом отношении сигнал/шум.

9. Недостатком турбокодов является уменьшение эффективности кодирования при высоком энергетическом отношении сигнал/шум, что связано с малым минимальным расстоянием турбокодов. Кроме того, применение процедуры выкалывания для увеличения относительной скорости кодирования также приводит к уменьшению минимального расстояния турбокода. Таким образом, видимым путем устранения недостатков турбокодов является использование в качестве составляющих турбокод кодов рекурсивных сверточных кодов с большим значением кодового ограничения и , что приведет к повышению минимального расстояния турбокода и позволит выбирать скорость турбокодирования в широких пределах без применения выкалывания. Практическая реализация алгоритма итеративного декодирования турбокодов с большим значением кодового ограничения затруднена из-за высокой сложности декодирования, поскольку алгоритмы мягкого декодирования используют только решетчатое представление сверточного кода.

10. Большинство известных методов синтеза сверточных кодов основаны на сложных процедурах переборного поиска и для синтеза сверточных кодов с высокими конструктивными характеристиками (высоким кодовым расстоянием) не применимы по причине больших вычислительных затрат. В этой связи достижение высоких показателей помехоустойчивости сверточного кодирования, в том числе и в параллельных каскадных схемах, сдерживается отсутствием алгебраических методов синтеза кодовых конструкций с требуемыми для практики свойствами.

11. В результате проведенных исследований выявлена научная проблема (противоречивая ситуация), в которой существующие положения теории помехоустойчивого кодирования не позволяют эффективно (вычислительно реализуемо) решать задачи синтеза и декодирования сверточных кодов с высокими конструктивными кодовыми характеристиками.

12. Поставлены задачи на исследование, которые в совокупности решают важную научную проблему, имеющую большое значение как для развития отдельного направления теории помехоустойчивого кодирования, так и для решения прикладных вопросов, связанных с обеспечением заданной достоверности передаваемой информации в телекоммуникационных системах и сетях.

РАЗДЕЛ 2
АЛГЕБРАИЧЕСКИЕ МЕТОДЫ СИНТЕЗА НЕРЕКУРСИВНЫХ СВЕРТОЧНЫХ КОДОВ

Перспективным направлением в развитии теории помехоустойчивого кодирования являются алгебраические методы синтеза сверточных кодов, сочетающие в себе развитый математический аппарат теории конечных полей, алгебраические процедуры построения и декодирования сверточных кодов с высокой энергетической эффективностью. В данном разделе на основе единого общетеоретического подхода с использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов разрабатываются алгебраические методы синтеза нерекурсивных сверточных кодов, исследуются процедуры построения нерекурсивных сверточных кодов , разрабатываются алгоритмы для их реализации.

2.1. Алгебраические методы синтеза несистематических нерекурсивных сверточных кодов

Алгебраический подход к построению сверточных кодов состоит в представлении порождающих многочленов сверточного кода через порождающий многочлен недвоичного циклического кода и сведении задачи синтеза непрерывного линейного кода к обобщению работы кодера недвоичного линейного блокового кода на случай полубесконечной длины кодового слова. Этот подход позволяет использовать развитый математический аппарат полиномиального описания избыточных кодов и в ряде случаев позволяет синтезировать сверточные коды с улучшенными свойствами с относительной скоростью кодирования $R = 1 / m$, $m = 2, 3, \ldots$.

Рассмотрим несистематический нерекурсивный сверточный (n, k) – код над $GF(q)$ с параметрами: $k^0 = 1$, $n^0 = m \cdot k^0 = m$, $k = r + 1$, $n = (r + 1) \cdot n^0 = k \cdot m$ и скоростью $R = 1 / m$, построенный с помощью несистематического сверточного кодера (рис. 2.1).

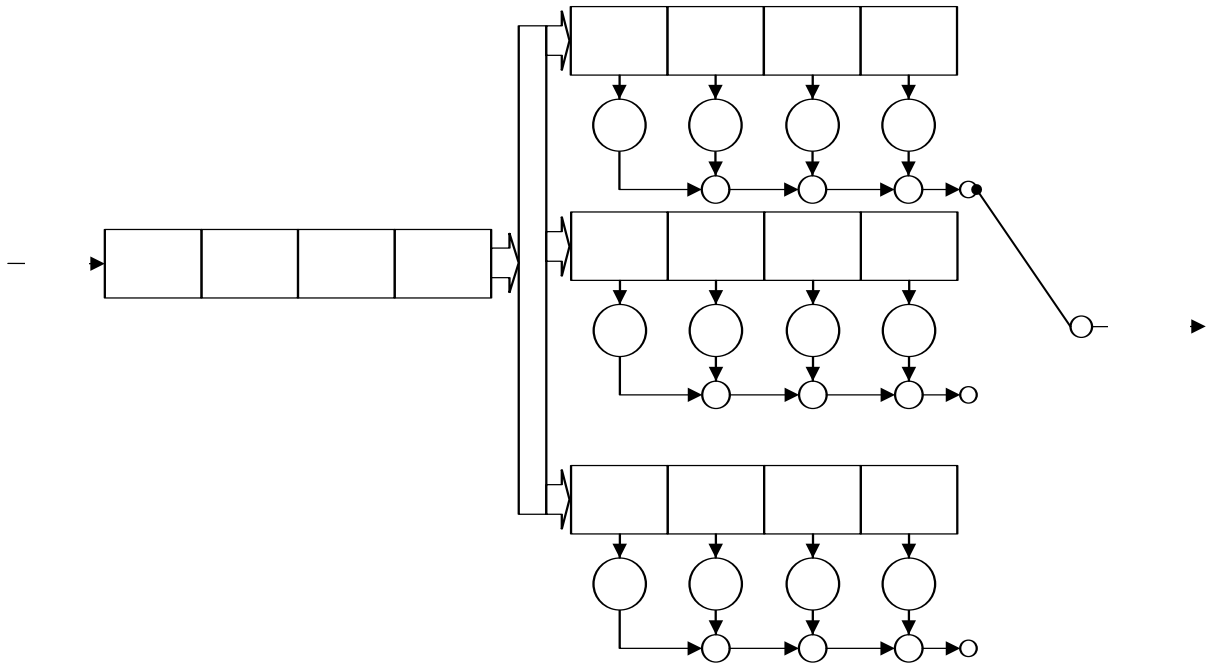


Рис. 2.1. Схема несистематического сверточного кодера с $R = 1/m$

Для описания процесса кодирования информации несистематическим сверточным кодером использован подход, состоящий в формальном определении сверточного кода через недвоичный циклический код над $GF(q^m)$, где степень расширения поля m соответствует числу q – ичных многочленов сверточного кода.

Пусть многочлен

(2.1)

является информационной последовательностью, подлежащей кодированию (в общем случае многочлен $I(x)$ может быть бесконечной длины), а многочлены

;

; (2.2)

…

,

будут порождающими многочленами рассматриваемого сверточного кода. Коэффициенты при переменной в выражениях (2.1) и (2.2) являются элементами $GF(q)$. Если один из многочленов в выражении (2.2) имеет меньший показатель степени, то добавим в этот многочлен необходимое (до большего) количество нулевых коэффициентов при старших степенях формальной переменной x.

Процесс кодирования информации рассматриваемым сверточным кодером опишем следующим образом. Информационная последовательность $I(x)$ вида (2.1) поступает в кодер сверточного кода, где происходит ее умножение на многочлены $P_1(x) \ldots P_m(x)$ вида (2.2) и получение последовательностей $F_1(x) \ldots F_m(x)$ соответственно:

;

; (2.3)

…

,

где $s_{i,j}$ – коэффициент в многочлене $F_i(x)$ при x^j в результате перемножения многочленов $I(x)$ и $P_i(x)$.

Кодовое слово $C(x)$ формируется путем последовательного считывания символов при одинаковых степенях многочленов $F_1(x)$ ... $F_m(x)$, т.е.:

(2.4)

Если на вход схемы несистематического сверточного кодера подать информационный вектор вида $\{0, 0, ..., 1\}$, то информационный многочлен запишется как $I(x)=1$, а кодовое слово запишется в виде

(2.5)

Последнее выражение однозначно определяет несистематическое правило сверточного кодирования.

Рассмотрим конечное поле $GF(q^m)$, построенное по кольцу многочленов, с коэффициентами над $GF(q)$. В выражении (2.5) каждому набору

$$\{p_{1,i}, p_{2,i}, \ldots, p_{m,i}\}$$

сопоставим элемент поля $\beta_i \in GF(q^m)$, такой, что

$$\beta_i = p_{1,i} + p_{2,i}x^2 + \ldots + p_{m,i}x^m .$$

Выражение (2.5) запишем в виде

(2.6)

Если выражение (2.6) суть порождающий многочлен недвоичного (N, K, D) циклического кода над $GF(q^m)$, то справедлива следующая теорема.

Теорема 2.1. Несистематический сверточный код над $GF(q)$ (рис. 2.1) с $R = 1/m$ однозначно задается многочленом $P(x)$ над $GF(q^m)$ вида (2.6). Если многочлен (2.6) задает недвоичный (N, K, D) циклический код над $GF(q^m)$, то он однозначно определяет (n, k) несистематический сверточный код над $GF(q)$ с кодовым ограничением $\nu = r \cdot k^0 = r$ и параметрами

Доказательство. Действительно, недвоичный *(N, K, D)* циклический код над $GF(q^m)$, порожденный многочленом *P(x),* степени *r* однозначно определяет набор регистров сдвига, соединенных связями (рис. 2.1) и задает рекуррентное правило кодирования, т.е. однозначного соответствия входной (информационной) последовательности в кодовую (выходную) последовательность

$$C(x) = I(x) \cdot P(x).$$

Параметры несистематического кода соответствуют рассмотренному выше примеру (рис. 2.1), т.е.: $k^0 = 1$, $n^0 = m$. Циклический код над $GF(q^m)$ задает длину кодирующего регистра и, соответственно, число хранящихся в кодере информационных кадров. Длина кодового ограничения ν, конструктивные параметры *n* и *k*, скорость *R* сверточного кодирования определяются выражениями

$$\nu = r \cdot k^0 = r,\ k = r + 1,\ n = (r + 1) \cdot n^0 = k \cdot m,\ R = 1 / m.$$

Если на вход кодирующего устройства подать информационный блок данных длиной *K q* – ичных символов, то считанная с выхода кодовая последовательность длиной *N* q^m – ичных символов суть кодовое слово циклического *(N, K, D)* кода над $GF(q^m)$. Следовательно, два любых кодовых блока, соответствующих двум произвольным входным последовательностям длиной *K q* – ичных символов, будут отличаться в *D* q^m – ичных символов. Последовательное считывание символов при одинаковых степенях многочленов $F_1(x) \ldots F_m(x)$ – суть отображение элементов поля $GF(q^m)$ в элементы образующего поля *GF(q)*, которое не уменьшает кодовое расстояние между произвольными *q* – ичными кодовыми словами длины $N \cdot m$.

По условию теоремы длина информационного кадра $k^0 = 1$, следовательно, для кодовых слов, соответствующих *K* различным информационным кадрам, $d_K \geq D$. По определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_{r+1} \leq d_{r+2} \leq \ldots \leq d_\infty$. Если выполняется условие $K \leq r$, то, очевидно, $d_\infty \geq d \geq d_K$. Если $K > r$, то выполняется лишь равенство $d_\infty \geq d_K$.

Рассмотренное обобщение несистематического сверточного *(n, k)* кода и теорема 2.1 позволяют алгебраически задавать параметры сверточного кода для произвольной длины кодового ограничения. С использованием такого подхода в работах [77-90] подробно рассмотрены алгоритмы формирования порождающих многочленов сверточного кода и алгебраические алгоритмы

построения на их основе двоичных сверточных кодов с заранее заданными конструктивными свойствами. Отметим, что в результате выполнения этих алгоритмов удается упростить процедуру построения сверточных кодов с предварительной оценкой их параметров. Уточнение кодового расстояния (условие $d_\infty \geq D$) позволяет, как правило, улучшить кодовые характеристики.

Рассмотренный алгебраический метод позволяет строить сверточные коды для скорости $R = 1 / m$, где m – степень расширения базового поля, над которым задается порождающий многочлен циклического кода. Это обстоятельство сужает область практического использования рассмотренного метода. Кроме того, наибольший энергетический выигрыш от кодирования большинство линейных кодов позволяет при скорости $R \approx 1 / 2 – 2 / 3$.

Ниже предлагается алгебраический метод синтеза сверточных кодов, являющийся теоретическим обобщением на случай $R = k^0 / m$ и позволяющий снять указанные ограничения по синтезу нерекурсивных кодов сверточных кодов.

В основе рассмотренного выше алгебраического метода построения сверточных кодов лежит ограничение недвоичного циклического кода над $GF(q^m)$ на подполе $GF(q)$. Подобное представление позволяет алгебраически определить несистематический (n, k) сверточный код с $R = 1 / m$.

Для снятия ограничения по скорости кодирования предлагается алгебраический метод построения сверточных кодов, в основе которого лежит ограничение недвоичного циклического кода над $GF(q^m)$ на произвольное подмножество $H \subseteq GF(q^m)$, $|H| \geq |GF(q)|$.

Если $|H| = |GF(q)|$, получим, как частный случай, изложенный выше метод.

Рассмотрим несистематический сверточный (n, k) – код над $GF(q)$ со скоростью $R = k^0 / m$ (см. рис. 2.2).

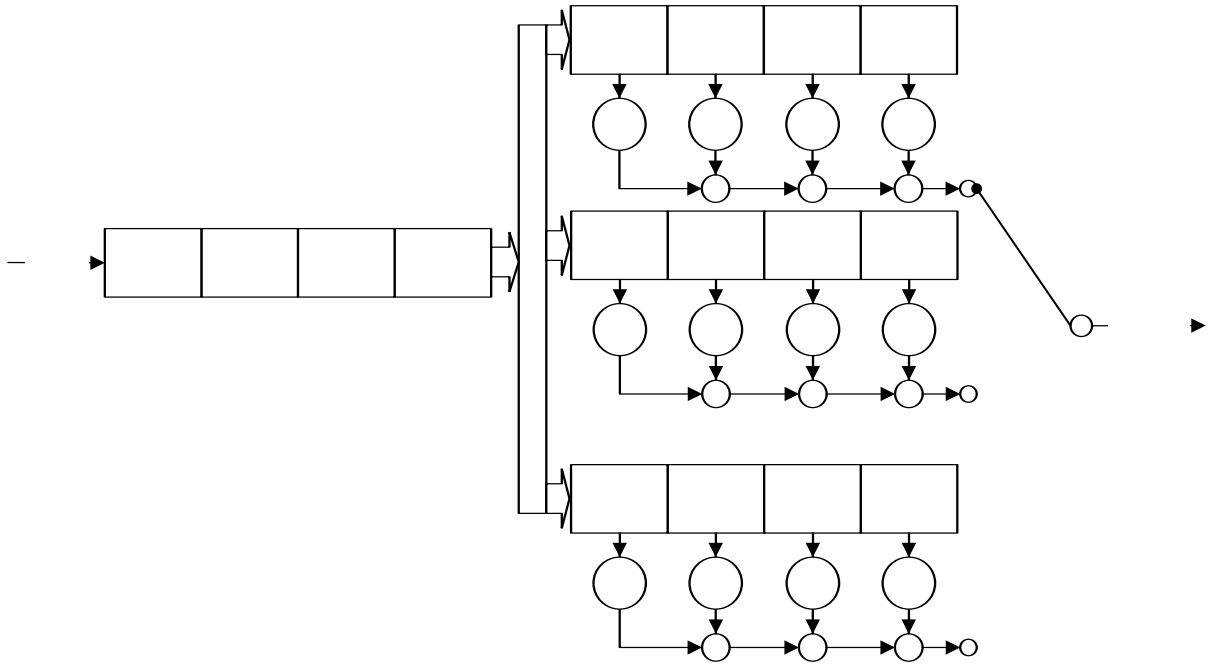


Рис. 2.2. Схема несистематического сверточного кодера при $R = k^0/m$

Разобьем входную информационную последовательность на информационные кадры по $k^0 \geq 1$ символов, каждый символ которых принадлежит $GF(q)$. В общем случае информационная последовательность может быть бесконечной длины, т.е. состоять из бесконечного числа информационных кадров по k^0 символов.

Сопоставим каждому информационному кадру из k^0 символов один символ из множества $H \subseteq GF(q^m)$, $|H| \geq |GF(q)|$.

Тогда информационный многочлен представим в виде

, (2.7)

где $I_j \in H, j = 0,..., r-1$, $\log_q |H| = k^0$, $m \geq k^0$.

Пусть, как и прежде, многочлены $P_1(x), P_2(x), ..., P_m(x)$ – порождающие многочлены представленного на рис. 2.2 несистематического сверточного кода.

Процесс кодирования информации – информационная последовательность $I(x)$ вида (2.7) поступает в кодер (рис. 2.2), где происходит ее умножение на многочлены $P_1(x) \ldots P_m(x)$ вида (2.2). Получим последовательности $F_1(x) \ldots F_m(x)$:

;

; (2.8)

…

,

где $S_{i,j}$ – коэффициент в многочлене $F_i(x)$ при x^j в результате перемножения многочлена $I(x)$ вида (2.7) и многочленов $P_i(x)$ вида (2.2).

Кодовое слово $C(x)$ формируется путем последовательного считывания символов при одинаковых степенях многочленов $F_1(x) \ldots F_m(x)$, т.е.:

(2.9)

Если на вход сверточного кода подать информационный вектор вида $\{0, 0, ...,1\}$, то информационный многочлен запишется как $I(x)=1$, а кодовое слово (2.9) запишется в виде порождающего многочлена циклического кода, т.е. $C(x) = P(x)$. Таким образом, порождающий многочлен циклического кода однозначно определяет несистематическое правило сверточного кодирования . Справедлива следующая теорема.

Теорема 2.2. Если зафиксировать конечное множество H элементов поля $GF(q^m)$, причем $\log_q |H| = k^0$, $m \geq k^0$, то произвольный многочлен степени r с коэффициентами над $GF(q^m)$ полностью определяет несистематический сверточный (n, k) код над $GF(q)$ с информационным кадром длины k^0, кодовым ограничением $\nu = r \cdot k^0$ и параметрами:

Доказательство. Кодирование, по определению, это процесс однозначного сопоставления (соответствия) информационной и кодовой последовательностей. Пусть задан произвольный многочлен $P(x)$ над $GF(q^m)$ степени r вида (2.6) и входная последовательность над $GF(q)$.

Представим информационную последовательность в виде многочлена (2.7) с коэффициентами над H. Т.е. коэффициенты многочлена $I(x)$ в выражении (2.7) являются многочленами над $GF(q)$ степени $m - 1$:

, (2.10)

где $z_i \in GF(q)$, причем $m - k^0$ коэффициентов z_i равны нулю. Положим, для определенности, $z_i = 0$ для $i = k^0, \ldots, m - 1$. Первые k^0 элементов z_i в выражении (2.10) образуют информационный кадр k^0 символов над $GF(q)$. Определенное таким образом отображение символов $GF(q)$ в символы $GF(q^m)$ является однозначным соответствием.

Недвоичный (N, K, D) циклический код над $GF(q^m)$, порожденный многочленом $P(x)$ степени r, однозначно определяет набор регистров сдвига соединенных связями (рис. 2.2) и задает рекуррентное правило кодирования, т.е. однозначного соответствия входной (информационной) последовательности в кодовую (выходную) последовательность: $C(x) = I(x) \cdot P(x)$. Параметры несистематического кода соответствуют рассмотренному выше примеру (рис. 2.2), т.е. каждому информационному кадру длиной k^0 символов над $GF(q)$ (или, что эквивалентно, каждому символу из множества H) ставится в соответствие кадр кодовых символов длиной n^0.

Степень r порождающего многочлена $P(x)$ циклического (N, K, D) кода над $GF(q^m)$ задает длину кодирующего регистра и, соответственно, число хранящихся в кодере информационных кадров. Следовательно, длина кодового ограничения ν, конструктивные параметры n и k и скорость R сверточного кодирования определяются, соответственно, следующими выражениями:

$$\nu = r \cdot k^0,\ k = (r + 1) \cdot k^0,\ n = k \cdot n^0 / k^0,\ R = k^0 / m,\ m \geq k^0.$$

Лемма 2.1. Если существует такое целое w, что $m = w \cdot k^0$, то порождающий многочлен степени r (N, K, D) циклического кода над $GF(q^m)$ полностью определяет несистематический сверточный (n_*, k_*, d_*) код над

с кодовым ограничением $\nu_* = r \cdot k_*^0 = r$ и параметрами

Доказательство. Согласно теореме 2.2 произвольный многочлен степени r с коэффициентами над $GF(q^m)$ полностью определяет несистематический сверточный (n, k, d) код над $GF(q)$ с кодовым ограничением ν, причем $n^0 = m$, $\nu = r \cdot k^0$, $k = (r + 1) \cdot k^0$, $n = k \cdot n^0 / k^0$, $R = k^0 / m$, $m \geq k^0$, $k^0 = \log_q |H|$, $H \subseteq GF(q^m)$. Если каждый информационный кадр длиной k^0 q – ичных символов представить одним – ичным символом, то получим несистематический сверточный $(n*, k*, d*)$ код над , где изоморфно множеству H. На вход такого кодера подступает K информационных кадров по одному – ичному символу, следовательно, $k*^0 = 1$. С выхода кодера снимается кодовая последовательность длиной N q^m – ичных символов. Если при этом выполняется равенство $m = w \cdot k^0$ для произвольного целого w, то $n*^0 = w$. Тогда, очевидно, выполняются равенства: $\nu = r$; $k = r + 1$; $n = k \cdot w$ $R = 1 / w$, а по теореме 2.2: $C(x) = I(x) \cdot P(x)$, что соответствует обобщению теоремы 2.1 на случай несистематических сверточных кодов над .

Лемма 2.2. Если $|H|=|GF(q)|$ получим, как частный случай теоремы 2.2, алгебраически заданный сверточный код для $R = 1 / m$, что соответствует результату теоремы 2.1.

Доказательство. Действительно, если $|H|=|GF(q)|$, то по теореме 2.2 получим $k^0 = 1$. Следовательно, процесс сверточного кодирования соответствует ограничению поля $GF(q^m)$ на подполе $GF(q)$ и $k^0 = 1$, $n^0 = m$, $\nu = r \cdot k^0 = r$, $k = r + 1$, $n = k \cdot n^0 / k^0$, $R = 1 / m$, $C(x) = I(x) \cdot P(x)$, что соответствует результату теоремы 2.1.

Лемма 2.3. Если $q = 2^{m*}$, то получим, как частный случай теоремы 2.2, алгебраически заданный двоичный сверточный код с $R = k^0 / u$, причем $u = m \cdot m*$.

Доказательство. По теореме 2.2 имеем несистематический сверточный (n, k, d) код над $GF(q)$ с кодовым ограничением $\nu = r \cdot k^0$ и параметрами: $n = k \cdot n^0 / k^0$; $k = (r + 1) \cdot k^0$; $R = k^0 / m$. Если на вход такого кодера подать информационный кадр из $m* \cdot k^0$ двоичных символов (что эквивалентно подаче кадра из k^0 q – ичных символов), а снятый с выхода кадр кодового слова – q^m–ичный символ преобразовать в $m \cdot m*$ бит, получим однозначное отображение – двоичное правило кодирования. Подставив эти параметры в результат теорем 2.2–2.3, получим: длина двоичного информационного кадра

$k_2^0 = m^* \cdot k^0$; $n_2^0 = u$; $\nu_2 = r \cdot m^* \cdot k^0$; $k_2 = (r + 1) \cdot m^* \cdot k^0$; $n_2 = k \cdot n^0 / (m^* \cdot k^0)$; $R = m^* \cdot k^0 / u$; $u \geq m^* \cdot k^0$; $k^0 = \log_q |H|$; $H \subseteq GF(q^m)$.

Лемма 2.4. Если $|H| = |GF(q^m)|$, получим отображение информационной последовательности самое в себя, а процесс кодирования будет безизбыточным.

Доказательство. Действительно, если $|H| = |GF(q^m)|$ то $k^0 = m$ и процесс сверточного кодирования соответствует ограничению поля $GF(q^m)$ на поле $GF(q^m)$, т.е. процесс кодирования соответствует отображению элементов поля в элементы этого же поля. Подставив значение $k^0 = m$ в результат теоремы 2.2, получим: $n^0 = m$; $\nu = r \cdot m$; $k = (r + 1) \cdot m$; $n = k \cdot n^0 / k^0 = k$; $R = k / k = 1$. Следовательно, кодирование безизбыточное, а при условии бесконечности многочлена $I(x)$ информационная последовательность отображается самое в себя.

Таким образом, результат теоремы 2.2 и лемм 2.1–2.4 позволяет обобщить построение несистематических сверточных кодов произвольной скорости. Для определения минимального расстояния сверточного кода сформулируем и докажем следующую теорему.

Теорема 2.3. Порождающий многочлен степени r (N, K, D) циклического кода над $GF(q^m)$ полностью определяет несистематический сверточный (n, k, d) код над $GF(q)$ с кодовым ограничением $\nu = r \cdot k^0$ и параметрами

Доказательство. Согласно теореме 2.2 произвольный многочлен степени r с коэффициентами над $GF(q^m)$ полностью определяет несистематический сверточный (n, k, d) код над $GF(q)$ (см. рис. 2.2) с кодовым ограничением ν, причем $n^0 = m$; $\nu = r \cdot k^0$; $k = (r + 1) \cdot k^0$; $n = k \cdot n^0 / k^0$; $R = k^0 / m$; $m \geq k^0$; $k^0 = \log_q |H|$; $H \subseteq GF(q^m)$.

Если на вход устройства (рис. 2.2) подать K информационных кадров по k^0 q–ичных символов (что эквивалентно подаче K кадров по одному q^m – ичному символу), то снятая с выхода кодовая последовательность длиной N q^m – ичных символов суть кодовое слово циклического (N, K, D) кода над $GF(q^m)$. Следовательно, два любых кодовых блока, соответствующих двум произвольным входным последовательностям длиной K – ичных символов, будут отличаться, по крайней мере, в D q^m–ичных символов. Последовательное считывание символов при одинаковых степенях многочленов $F_1(x) \ldots F_m(x)$ суть отображение элементов поля $GF(q^m)$, в элементы образующего поля $GF(q)$ которое не уменьшает кодовое расстояние между произвольными q–ичными кодовыми словами длины $N \cdot m$

. По условию теоремы длина информационного кадра равна k^0, следовательно, $d_K \geq D$. По определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_{r+1} \leq d_{r+2} \leq \ldots \leq d_\infty$. Если выполняется условие $K \leq r$, то, очевидно, $d_\infty \geq d \geq d_K$. Если $K > r$, то выполняется лишь неравенство $d_\infty \geq d_K$, что и завершает доказательство.

Следует отметить, что оценка $d_\infty \geq D$ в теоремах 2.1, 2.3 не точна. Основным недостатком рассмотренного подхода построения сверточных кодов является низкая конструктивная величина свободного минимального расстояния. Ниже предлагается подход по предсказанию (прогнозированию) свободного кодового расстояния несистематических сверточных кодов, заданных с помощью порождающего многочлена циклического кода.

Предложение. Предсказанное (прогнозируемое) свободное минимальное расстояние $d_П$ несистематического сверточного *(n, k, d)* – кода над *GF(q)*, алгебраически заданного порождающим многочленом *(N, K, D)* циклического кода над $GF(q^m)$, определяется выражением

. (2.11)

Вывод выражения (2.11) основан на подсчете ненулевых *q*–ичных символов в выходной кодовой последовательности несистематического сверточного *(n, k, d)* кода, алгебраически заданного с помощью порождающего многочлена *(N, K, D)* циклического кода над $GF(q^m)$.

По теоремам 2.1 – 2.3 несистематический сверточный код эквивалентен ограничению недвоичного циклического кода над $GF(q^m)$ на подполе *GF(q)*, т .е. отображению символов кодовых слов циклического кода над $GF(q^m)$ в символы сверточного кода над *GF(q)*. Мощность множества прообразов равна q^m, а без нулевого символа поля $GF(q^m)$ мощность множества ненулевых прообразов равна $q^m - 1$. Каждому символу над $GF(q^m)$ соответствует m *q* – ичных символов, т.е. мощность множества образов равна $m \cdot q^m$. Всего ненулевых *q*–ичных символов в множестве образов равно $m \cdot (q^m - q^{m-1})$. Таким образом, при алгебраически заданном сверточном кодировании множество из $q^m - 1$ ненулевых символов над $GF(q^m)$ отображается в множество из $m \cdot (q^m - q^{m-1})$ ненулевых *q*–ичных символов. Следовательно, среднее число ненулевых *q*–ичных символов на выходе несистематического сверточного кода будет определяться как

,

где *D* – минимальное кодовое расстояние *(N, K, D)* циклического кода над $GF(q^m)$.

Следствие. Для несистематического сверточного (n_*, k_*, d_*) код над

с параметрами: $k_*^0 = 1$; $n_*^0 = m \cdot k_*^0 = m$; $\nu_* = r \cdot k_*^0 = r$; $k_* = r_* + 1$; $n_* = (r + 1) \cdot n_*^0 = k_* \cdot m_*$; $R = 1 / w$; $d_\infty \geq D$; $C(x) = I(x) \cdot P(x)$ для такого целого *w*, что $m = w \cdot k^0$ выражение (2.11) запишется в виде

. (2.12)

Действительно, по лемме 2.1 порождающий многочлен степени $r\ (N, K, D)$ циклического кода над $GF(q^m)$ полностью определяет несистематический сверточный (n_*, k_*, d_*) код над с параметрами: $k_*^0 = 1$; $n_*^0 = m \cdot k_*^0 = m$; $\nu_* = r \cdot k_*^0 = r$; $k_* = r_* + 1$; $n_* = (r + 1) \cdot n_*^0 = k_* \cdot m_*$; $R = 1 / w$; $d_\infty \geq D$; $C(x) = I(x) \cdot P(x)$ для такого целого w, что $m = w \cdot k^0$. По сути, такой код является ограничением (N, K, D) циклического кода над $GF(q^m)$ на поле $GF(q^w)$, т.е. отображением символов кодовых слов циклического кода над $GF(q^m)$ в символы сверточного кода над $GF(q^w)$. Проведя аналогичные рассуждения, получим искомое выражение (2.12).

По аналогии с рассуждениями для вывода выражений (2.11) и (2.11) для леммы 2.4 прогнозируемое свободное минимальное расстояние $d_П$ будет определяться выражением (2.11) после подстановки значения m* вместо m.

Рассмотрим пример расчета прогнозируемого свободного кодового расстояния для синтезированных алгебраических сверточных кодов.

Пример 1. Зафиксируем конечное поле $GF(2^2)$, построенное по кольцу многочленов по модулю $g(z) = z^2 + z + 1$. Поле $GF(2^2)$ состоит из четырех элементов: $\alpha^{-\infty} = 0$; $\alpha^0 = 1$; $\alpha^1 = z$; $\alpha^2 = z+1$.

Зафиксируем конечное поле $GF(4^3)$, построенное по кольцу многочленов по модулю $g(x) = x^3 + x^2 + x + 3$. Коэффициенты многочлена $g(x)$ – суть элементы поля $GF(2^2)$. В табл. 2.1 представлены элементы поля $GF(4^3)$ по классам сопряженных элементов, порядки элементов поля и степени минимальных многочленов.

Таблица 2.1

Структура конечного поля $GF(4^3)$

α^i	α^{4i}	α^{16i}	$deg(\alpha^i)$	$deg(f_i)$
1	2	3	4	5
α^0 (1 0 0)				1
α^1 (0 1 0)	α^4 (3 2 0)	α^{16} (2 3 0)	63	3
α^2 (0 0 1)	α^8 (2 0 3)	α^{32} (3 0 2)	63	3
α^3 (3 1 1)	α^{12} (2 2 3)	α^{48} (2 3 2)	21	3
α^5 (0 3 2)	α^{20} (1 1 1)	α^{17} (0 2 3)	63	3
α^6 (1 2 1)	α^{24} (2 3 3)	α^{33} (1 1 2)	21	3
α^7 (3 0 3)	α^{28} (2 0 2)	α^{49} (1 0 1)	9	3
α^9 (2 1 3)	α^{36} (0 2 2)	α^{18} (2 3 1)	7	3
α^{10} (2 1 2)	α^{40} (2 2 1)	α^{34} (1 3 3)	63	3
α^{11} (1 0 3)	α^{44} (0 0 2)	α^{50} (3 0 1)	63	3

Продолжение табл. 2.1

1	2	3	4	5

α^{13} *(2 1 1)*	α^{52} *(3 2 3)*	α^{19} *(3 3 2)*	63	3
α^{14} *(3 3 0)*	α^{56} *(1 1 0)*	α^{35} *(2 2 0)*	9	3
α^{15} *(0 3 3)*	α^{60} *(3 1 2)*	α^{51} *(3 2 1)*	21	3
α^{21} *(3 0 0)*			3	1
α^{22} *(0 3 0)*	α^{25} *(2 1 0)*	α^{37} *(1 2 0)*	63	3
α^{23} *(0 0 3)*	α^{29} *(1 0 2)*	α^{53} *(2 0 1)*	63	3
α^{26} *(0 2 1)*	α^{41} *(3 3 3)*	α^{38} *(0 1 2)*	63	3
α^{27} (3 1 3)	α^{45} (1 2 2)	α^{54} (3 3 1)	7	3
α^{30} (1 3 2)	α^{57} (0 1 1)	α^{39} (1 2 3)	21	3
α^{31} (1 3 1)	α^{61} (1 1 3)	α^{55} (3 2 2)	63	3
α^{42} (2 0 0)			3	1
α^{43} (0 2 0)	α^{46} (1 3 0)	α^{58} (3 1 0)	63	3
α^{47} (0 1 3)	α^{62} (2 2 2)	α^{59} (0 3 1)	63	3

Зафиксируем порождающий многочлен *P(x)* примитивного циклического *(N, K, D)* кода с коэффициентами над $GF(4^3)$. Пусть $N = 4095$, $D = 7$. Для выбора многочлена *P(x)* рассмотрим конечное поле $GF(64^2)$, построенное по кольцу многочленов по модулю $G(x) = x^2 + x + 3$. Коэффициенты многочлена *G(x)* – суть элементы поля $GF(4^3)$. Зададим циклический *(N, K, D)* код порождающим многочленом вида:

$$P(x) = HOK(f_1, f_2, f_3, f_4, f_5, f_6) = (x + \alpha^{1}) \cdot (x + \alpha^{64}) \cdot (x + \alpha^{2}) \cdot (x + \alpha^{128}) \cdot (x + \alpha^{3}) \cdot (x + \alpha^{192}) \cdot (x + \alpha^{4}) \cdot (x + \alpha^{256}) \cdot (x + \alpha^{5}) \cdot (x + \alpha^{320}) \cdot (x + \alpha^{6}) \cdot (x + \alpha^{384}) = x^{12} + x^{11} + x^{10} + x^{9} + x^{8} + x^{7} + x^{6} + x^{5} + x^{4} + x^{3} + x^{2} + x^{1} + 1.$$

Очевидно, многочлен *P(x)* задает код БЧХ над $GF(4^3)$ и по теореме БЧХ–кодов имеем: $N = 4095$, $K = 4082$, $D = 7$. Степень многочлена *P(x)* равна $r = 12$.

Воспользовавшись результатом теоремы 2.1 и леммы 2, получим несистематический сверточный *(n, k, d)* код над $GF(2^2)$ с параметрами: $m = 3$; $k^0 = 1$; $n^0 = m = 3$; $\nu = r \cdot k^0 = 12$; $k = r + 1 = 13$; $n = (r + 1) \cdot n^0 = k \cdot m = 39$;

$R = 1 / 3$; $C(x) = I(x) \cdot P(x)$. Работа такого кодера состоит в следующем. Поток входящих информационных символов разбивается на кадры по одному двухбитному символу. В течение каждого момента времени в регистр сдвига вводится новый информационный символ, вплоть до $r = 12$ символов, которые заполняют весь кодирующий регистр. Кодер, по введенному символу и $r = 12$ хранящимся в нем символам вычисляет три двухбитных символа кодового слова (один символ кодового слова БЧХ кода над $GF(4^3)$). Эти три двухбитных кодовых символа выводятся из кодера, как только следующий информационный символ поступает в него. Следовательно, каждому информационному символу над $GF(2^2)$ соответствуют три кодовых символа над $GF(2^2)$.

Воспользуемся рассмотренным сверточным кодом над $GF(2^2)$ для построения на его основе двоичного кода. Воспользовавшись леммой 3, получим обобщение результата теоремы 2.1: $k_2^0 = 2$; $n_2^0 = 6$; $\nu_2 = 24$; $k_2 = 26$;

$n_2 = 78$; $R = 1 / 3$; $d_\infty \geq 7$; $C(x)=I(x)\cdot P(x)$. Работа такого кодера состоит в следующем. Разобьем поток входных информационных символов на подблоки по *2* бита и представим его как элемент поля $GF(2^2)$. В течение каждого момента времени в регистр сдвига вводится новый информационный символ (суть два входных бита), вплоть до $r = 12$ символов , которые заполняют весь кодирующий регистр. Кодер вычисляет три двухбитных символа кодового слова, преобразует их в шестибитовый кадр кодового слова. Следовательно, каждому информационному кадру по 2 бита соответствует кадр кодовых символов из шести бит.

Рассмотрим множество H элементов поля $GF(4^3)$ вида

,

где $z_i \in GF(q)$, причем $z_2 = 0$. Элементы z_1 и z_2 образуют информационный кадр из $k^0 = 2$ символов над $GF(q)$. Определенное таким образом отображение символов $GF(q)$ в символы $GF(q^m)$ является однозначным соответствием. В соответствии с табл. 2.1 множество H содержит следующие элементы: $\{\alpha^{-\infty}, \alpha^1, \alpha^2, \alpha^5, \alpha^{15}, \alpha^{17}, \alpha^{22}, \alpha^{23}, \alpha^{26}, \alpha^{36}, \alpha^{38}, \alpha^{43}, \alpha^{44}, \alpha^{47}, \alpha^{57}, \alpha^{59}\}$, $log_q| H| = 2$. Воспользовавшись результатом теоремы 2.2, получим несистематический сверточный (n, k, d) код над $GF(2^2)$ с параметрами: $m = 3$; $k^0 = log_q| H| = 2$; $m \geq k^0$; $n^0 = 3$; $\nu = r \cdot k^0 = 24$; $k = (r+1)\cdot k^0 = 26$; $n = k \cdot n^0 / k^0 = 39$; $R = k^0 / m = 2 / 3$; $C(x) = I(x) \cdot P(x)$. Работа такого кодера состоит в следующем. Поток входящих информационных символов разбивается на кадры по два двухбитных символа и однозначно отождествляется одному из элементов группы H. По условию $H \subseteq GF(q^m)$, следовательно, имеет смысл выражение $C(x)=I(x)\cdot P(x)$. В течение каждого момента времени в регистр сдвига вводится новый информационный символ, принадлежащий группе $H \subseteq GF(q^m)$ и являющийся образом двух двухбитных входных символов. Поступающие в кодирующий регистр символы заполняют его вплоть до $r = 12$ символов. Кодер, по введенному символу (суть два двухбитных символа) и $r = 12$ хранящимся в нем символам вычисляет три двухбитных символа кодового слова (один символ кодового слова БЧХ кода над $GF(4^3)$). Эти три двухбитных кодовых символа выводятся из кодера, как только следующий информационный символ поступает в него. Следовательно, каждому информационному символу из группы H – двум двухбитным символам над $GF(2^2)$ соответствуют три кодовых символа над $GF(2^2)$.

Последняя конструкция позволяет также определить двоичный несистематический сверточный код. Воспользуемся результатом леммы 2. Разобьем поток входных информационных символов на подблоки по 4 бита. Мощность алфавита подблоков равна $2^4 = 16$, мощность группы H равна 21. Следовательно, всегда можно выбрать однозначное сопоставление потока входных 16–ичных символов набору символов группы H. По условию $H \subseteq GF(q^m)$, следовательно, имеет смысл выражение $C(x) = I(x) \cdot P(x)$. Воспользовавшись леммой 4, получим: $k_2^0 = 4$; $n_2^0 = 6$; $\nu_2 = 48$; $k_2 = 52$; $n_2 = 78$; $R = 2 / 3$.

Таким образом, порождающий многочлен *P(x) (N, K, D)* кода БЧХ над $GF(4^3)$ с параметрами $N = 4095$, $K = 4082$, $D = 7$ в зависимости от способа обработки входных символов однозначно определяет следующие несистематические сверточные коды.

1.Недвоичный сверточный код над $GF(2^2)$ с параметрами: $k^0 = 1$; $n^0 = 3$; $\nu = 12$; $k = 13$; $n = 39$; $R = 1 / 3$;

2.Двоичный сверточный код с параметрами: $k_2^0 = 2$; $n_2^0 = 6$; $\nu_2 = 24$; $k_2 = 26$; $n_2 = 78$; $R = 1 / 3$;

3.Недвоичный сверточный код над $GF(2^2)$ с параметрами: $k^0 = 2$; $n^0 = 3$; $\nu = 24$; $k = 26$; $n = 39$; $R = 2 / 3$;

4.Двоичный сверточный код с параметрами: $k_2^0 = 4$; $n_2^0 = 6$; $\nu_2 = 48$; $k_2 = 52$; $n_2 = 78$; $R = 2 / 3$.

Пример 2. Воспользуемся рассмотренным выше примером построения несистематических сверточных кодов. Согласно теореме 2.3 выполняется условие $K > r$ и справедливо неравенство $d_\infty \geq 7$. Таким образом, имеем четыре несистематических сверточных кода:

- *(*39, 13*)* сверточный код над $GF(2^2)$ с $\nu = 12$; $R = 1 / 3$, $d_\infty \geq 7$;
- двоичный *(*78, 26*)* сверточный код с $\nu = 24$; $R = 1 / 3$, $d_\infty \geq 7$;
- *(*39, 26*)* сверточный код над $GF(2^2)$ с $\nu = 24$; $R = 2 / 3$, $d_\infty \geq 7$;
- двоичный *(*78, 52*)* сверточный код с $\nu = 48$; $R = 2 / 3$, $d_\infty \geq 7$.

Отметим, что для поиска переборным методом сверточных кодов с $\nu = 48$ необходимо перебрать кодирующих устройств. Для определения кодового расстояния необходимо протестировать $2^{52} = 4503599627370496$ кодовых слов в каждом устройстве, что является практически неразрешимой задачей. Рассмотренные примеры наглядно демонстрируют конструктивность предложенного алгебраического метода построения несистематических сверточных кодов.

Пример 3. Воспользуемся рассмотренным выше примером построения несистематических сверточных кодов, алгебраически заданных с помощью порождающего многочлена *P(x) (N, K, D)* кода БЧХ над $GF(4^3)$ с параметрами $N = 4095$, $K = 4082$, $D = 7$. Рассчитаем прогнозируемое свободное кодовое расстояние $d_П$ несистематического сверточного *(n, k, d)* кода над $GF(2^2)$ с параметрами: $k^0 = 1$; $n^0 = 3$; $\nu = 12$; $k = 13$; $n = 39$; $R = 1 / 3$; $d_\infty \geq 7$. Подставив в выражение (2.9) параметры кода, получим

.

Аналогично для двоичного *(78, 26)* сверточного кода с $\nu = 24$, $R = 1 / 3$, $d_\infty \geq 7$ (случай 2):

.

Для *(39, 26)* сверточного кода над $GF(2^2)$ с $\nu = 24$, $R = 2 / 3$, $d_\infty \geq 7$ (случай 3):

.

Для двоичного *(78, 52)* сверточного кода с $\nu = 48$, $R = 2 / 3$, $d_\infty \geq 7$ (случай 4):

.

Рассмотрим еще один пример построения несистематического сверточного кода через порождающий многочлен кода Рида–Соломона (РС). Зафиксируем, как и в предыдущем примере, конечные поля $GF(2^2)$ и $GF(4^3)$. Зададим *(N, K, D)* код РС над $GF(4^3)$ через проверочный многочлен $P(x)$ вида

$$P(x) = (x - \alpha^{i}) \cdot (x - \alpha^{2i}) \cdot \ldots \cdot (x - \alpha^{2ti}),$$

где t – число ошибок, которые должен исправлять *(N, K, D)* код РС, $N = q^m - 1$; $r = degP(x)$; $K = N - r$; $D = 2t + 1$; $\alpha^{i} \in GF(q^m)$.

Пусть $D = 7$, $i = 1$. Тогда $P(x) = (x - \alpha^{1}) \cdot (x - \alpha^{2}) \cdot (x - \alpha^{3}) \cdot \cdot (x - \alpha^{4}) \cdot (x - \alpha^{5}) \cdot (x - \alpha^{6}) = x^6 + x^5 + x^4 + x^3 + x^2 + x + a$, $N = 63$, $r = 6$, $K = 57$.

По аналогии с рассмотренным выше примером определим несистематические сверточные коды с следующими параметрами.

1.Недвоичный сверточный код над $GF(2^2)$ с параметрами: $k^0 = 1$; $n^0 = 3$; $\nu = 6$; $k = 7$; $n = 21$; $R = 1 / 3$; $d_\infty \geq 7$; ;

2.Двоичный сверточный код с параметрами: $k_2^0 = 2$; $n_2^0 = 6$; $\nu_2 = 12$; $k_2 = 14$; $n_2 = 42$; $R = 1 / 3$; $d_\infty \geq 7$; ;

3.Недвоичный сверточный код над $GF(2^2)$ с параметрами: $k^0 = 2$; $n^0 = 3$; $\nu =12$; $k = 14$; $n = 21$; $R = 2 / 3$; $d_\infty \geq 7$; ;

4.Двоичный сверточный код *(n, k, d)* код над $GF(2)$ с параметрами: $k_2^0 = 4$; $n_2^0 = 6$; $\nu_2 = 24$; $k_2 = 28$; $n_2 = 42$; $R = 2 / 3$; $d_\infty \geq 7$; .

В результате проведенного уточнения свободного минимального расстояния рассмотренных кодов получены следующие значения: , , , , что позволяет отнести их к лучшим известным сверточным кодам.

Рассмотренный пример синтеза несистематических сверточных кодов через порождающий многочлен *(N, K, D)* циклического кода над $GF(q^m)$ удовлетворяет условию $d_\infty \geq d_П$, где d_∞ – истинное свободное минимальное расстояние, полученное уточнением кодового расстояния. Отметим также, что при тестировании (уточнении) кодового расстояния несистематических сверточных кодов автором не было получено ни одного случая с $d_\infty < d_П$. Это говорит о конструктивности предложенного способа предсказания свободного минимального расстояния несистематических сверточных кодов.

В следующем подразделе рассмотрим алгоритмы построения несистематических сверточных кодов с заданными конструктивными характеристиками.

2.2. Алгоритмы построения несистематических нерекурсивных сверточных кодов

Практическое использование результатов доказанных теорем 2.1 – 2.3 позволяет алгебраически задавать несистематический сверточный код порождающим многочленом циклического кода. Конструктивные характеристики сверточного *(n, k, d)* кода над *GF(q)* связаны с параметрами образующего циклического *(N, K, D)* кода над $GF(q^m)$ с порождающим многочленом степени r:

$$k^0 = log_q |H|;$$
$$n^0 = m;$$
$$\nu = r \cdot k^0;$$
$$k = (r + 1) \cdot k^0;$$
$$n = k \cdot n^0 / k^0;$$
$$d_\infty \geq D;$$
$$R = k^0 / m,\ m \geq k^0;$$

где $H \subseteq GF(q^m)$.

Алгоритм построения сверточного *(n, k, d)* кода над *GF(q)* определим в виде последовательности следующих шагов.

ШАГ 1. Выбор конструктивных параметров сверточного *(n, k, d)* кода над *GF(q)*.

ШАГ 2. Расчет параметров образующего поля $GF(q^m)$. Выбор циклического кода, расчет его конструктивных *(N, K, D)* параметров над $GF(q^m)$.

ШАГ 3. Выбор порождающего многочлена циклического *(N, K, D)* кода $GF(q^m)$. Расчет прогнозируемого свободного расстояния сверточного кода.

ШАГ 4. Определение порождающих многочленов несистематического сверточного *(n, k, d)* кода, построение схемы кодера.

ШАГ 5. Уточнение минимального кодового расстояния и свободного кодового расстояния несистематического сверточного *(n, k, d)* кода (при необходимости).

Рассмотрим выполнение предложенного алгоритма более подробно.

После ввода конструктивных параметров сверточного кода над *GF(q)* – параметров ν, n^0, k^0 и q на втором шаге выполняется расчет параметров образующего поля $GF(q^m)$, осуществляется выбор циклического кода и расчет его конструктивных *(N, K, D)* параметров над $GF(q^m)$. Для этого выражения связывающие параметры сверточного и циклических кодов перепишем в виде

$$R = k^0 / n^0;$$
$$m = n^0;$$
$$r = \nu / k^0;$$
$$D \leq d_\infty.$$

После расчета параметров образующего поля $GF(q^m)$ необходимо выбрать циклический код, порождающий многочлен которого будет задавать сверточный код.

Рассмотрим случай, когда в качестве циклического кода выбран примитивный код БЧХ. Для расчета его конструктивных (N, K, D) параметров зафиксируем двучлен $(x^M - 1)$ так, что конструктивная длина примитивного кода БЧХ равна

$$N = (q^m)^M - 1.$$

Далее, определив степень r порождающего многочлена примитивного кода БЧХ, рассмотрим поле разложения двучлена $(x^M - 1)$ на минимальные многочлены элементов поля $GF((q^m)^M)$ над $GF(q^m)$. Порождающий многочлен примитивного кода БЧХ задается в виде

$$P(x) = НОК(f_1, f_2, \ldots, f_{2t}),$$

где t – число ошибок, которые должен исправлять циклический (N, K, D) код, $N = (q^m)^M - 1$, $r = degP(x)$, $K = N - r$, $D = 2t + 1$, f_i – минимальные многочлены над $GF(q^m)$ элементов $\alpha^i \in GF((q^m)^M)$. После расчета $d_П$ третий шаг алгоритма для примитивных кодов БЧХ завершен.

Рассмотрим случай, когда в качестве циклического кода выбран непримитивный код БЧХ. По определению, длина непримитивного кода БЧХ равна одному из сомножителей в разложении числа $(q^m)^M - 1$ (если, конечно, число $(q^m)^M - 1$ не является простым), т.е.

$$N = ((q^m)^M - 1)/g$$

для произвольного целого g, делящего нацело число $(q^m)^M - 1$. Очевидно, что должно выполняться также условие $r < N$.

Порождающий многочлен непримитивного кода БЧХ задается в виде

$$P(x) = НОК(\varphi_1, \varphi_2, \ldots, \varphi_{2t}),$$

где t – число ошибок, которые должен исправлять циклический (N, K, D) код, $N = ((q^m)^M - 1)/g$; $r = degP(x)$; $K = N - r$; $D = 2t + 1$, φ_i – минимальные многочлены над $GF(q^m)$ элементов $\beta^i \in GF((q^m)^M)$ такие, что их порядок равен N, т.е. $\beta^i = \alpha^{jg}$, $j = 1, 2, \ldots, M/2$. После расчета $d_П$ третий шаг алгоритма для непримитивных кодов БЧХ завершен.

Рассмотрим случай, когда в качестве циклического кода выбран код РС . По определению, порождающий многочлен кода РС задается в виде

$$P(x) = (x - \alpha^i) \cdot (x - \alpha^{2i}) \cdot \ldots \cdot (x - \alpha^{2ti}),$$

где: t – число ошибок, которые должен исправлять (N, K, D) код РС, $N = q^m - 1$; $r = degP(x)$; $K = N - r$; $D = 2t + 1$; $\alpha^i \in GF(q^m)$. После вычисления (N, K, D) параметров кода РС, выбора порождающего многочлена и расчета $d_П$ третий шаг алгоритма для рассмотренного случая завершен.

На четвертом шаге предложенного алгоритма построения несистематических сверточных кодов производится определение порождающих многочленов сверточного кода над $GF(q)$, строится схема кодера. Если порождающий многочлен $P(x)$ циклического (N, K, D) кода над $GF(q^m)$

$$, \alpha_i \in GF(q^m)$$

записать в виде

то многочлены

;

;

…

будут являться порождающими многочленами искомого несистематического сверточного кода. Схема алгоритма формирования порождающих многочленов несистематического сверточного кода представлена на рис. 2.3.

Подставим в общую схему несистематического сверточного кодера параметры полученных многочленов $P_1(x)$ … $P_m(x)$. Коэффициенты многочленов $P_1(x)$ … $P_m(x)$ однозначно определяют кодирующие регистры с обратными связями, т.е. однозначно задают схему кодера искомого сверточного *(n, k, d)* кода.

На пятом шаге разработанного алгоритма построения сверточных кодов путем тестирования производится уточнение кодового расстояния (при необходимости).

На рис. 2.4 представлена общая схема алгоритма алгебраического построения несистематического сверточного кода с использованием разработанного метода.

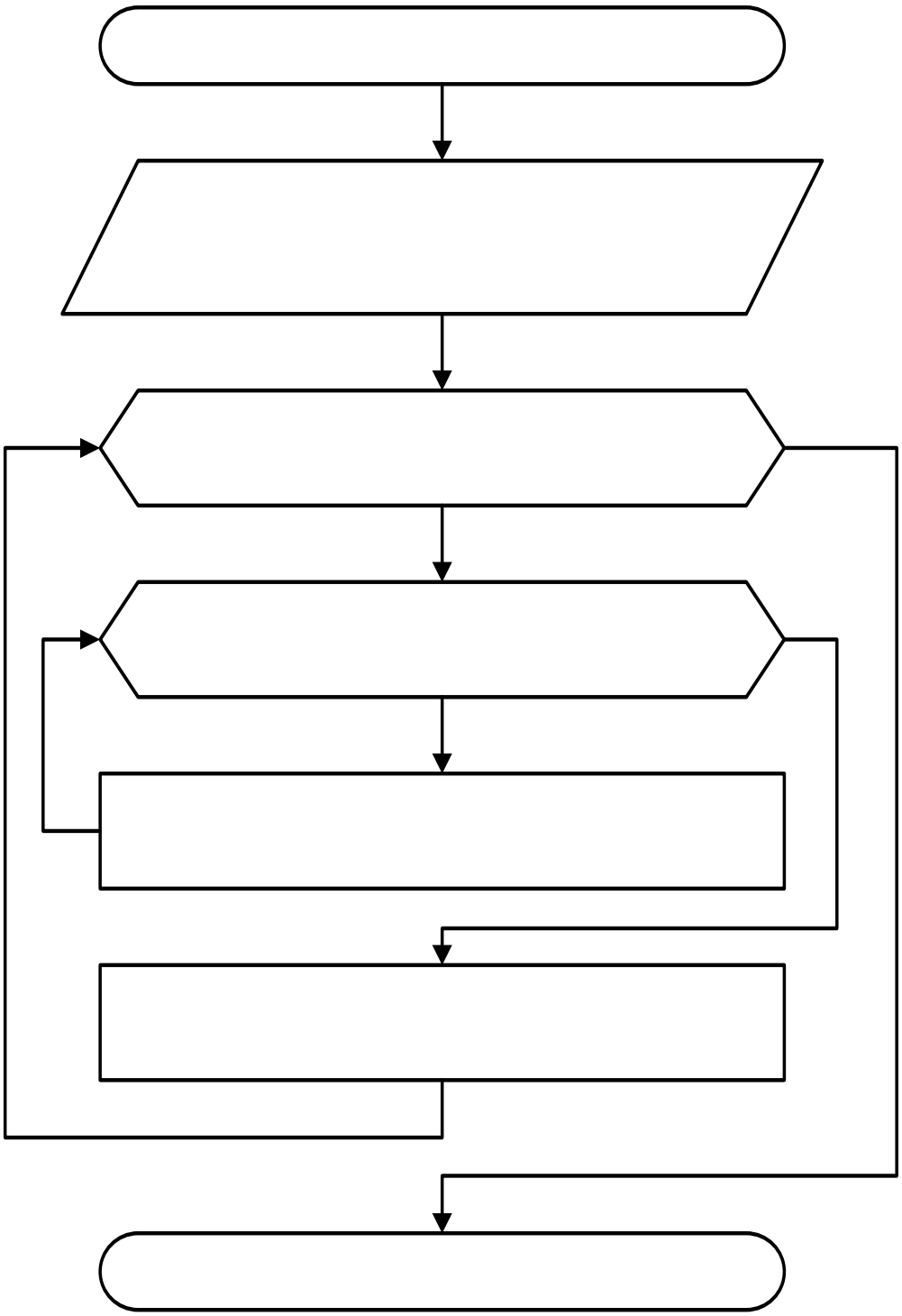


Рис. 2.3. Схема алгоритма формирования порождающих многочленов несистематического сверточного кода

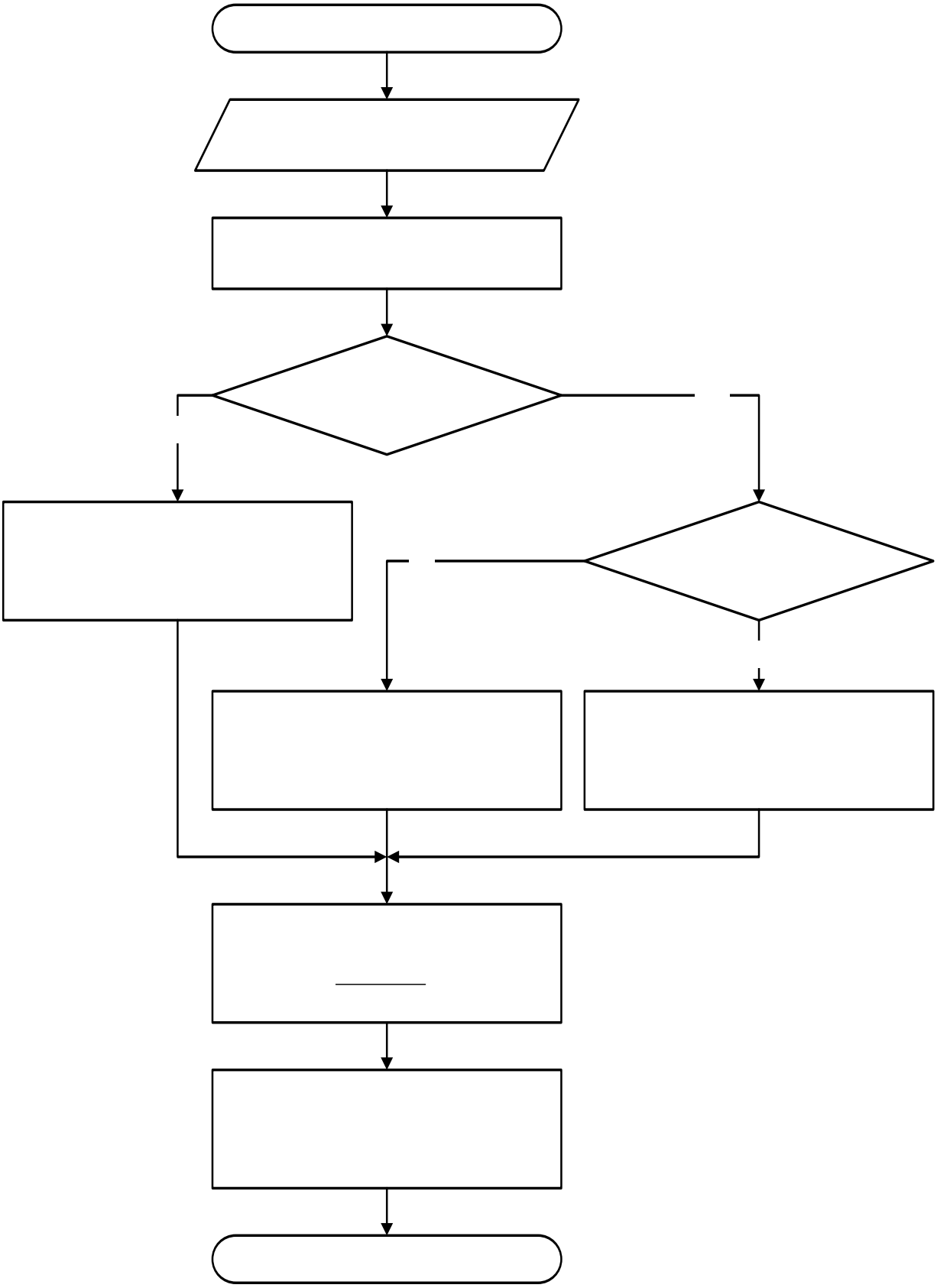


Рис. 2.4. Схема алгоритма алгебраического построения несистематического сверточного кода

2.3. Свойства синтезированных несистематических нерекурсивных сверточных кодов

Проведем исследования свойств несистематических сверточных кодов, заданных порождающими многочленами циклических кодов.

Разработанный метод построения несистематических сверточных кодов позволяет алгебраически определить сверточный *(n, k, d)* код через порождающий многочлен циклического кода. Проведем исследования свойств сверточных кодов в зависимости от выбранного циклического кода.

Зафиксируем циклический *(N, K, D)* код над $GF(q^m)$ через его порождающий многочлен *P(x)* степени *r*. Параметры сверточного *(n, k, d)* кода над *GF(q)* запишутся в виде: $n^0 = m$; $k^0 = log_q |H|$; $H \subseteq GF(q^m)$; $\nu = r \cdot k^0$; $k = (r + 1) \cdot k^0$; $n = k \cdot n^0 / k^0$; $d_\infty \geq D$; $R = k^0 / m$, $m \geq k^0$. Лучшим будет такой сверточный *(n, k, d)* код, который при меньших конструктивных характеристиках *n* и *k* обеспечит большие значения *d* и/или d_∞.

Проанализируем потенциальные возможности примитивных и непримитивных кодов БЧХ, кодов РС для построения хороших сверточных кодов.

Определяющим в выражениях по расчету конструктивных параметров *n* и *k* сверточного кода является показатель *r* – длина регистра сдвига в схеме сверточного кодирования. Если несистематический сверточный *(n, k, d)* код задан через порождающий многочлен *P(x)* циклического *(N, K, D)* кода над $GF(q^m)$, то показатель *r* соответствует степени порождающего многочлена $r = degP(x)$, а задача построения хорошего сверточного кода сводится к выбору такого циклического кода, который при минимальном *r* обеспечивал максимальное значение *D*. В алгебраической теории блоковых кодов известно следующее ограничение:

$$D \leq N - K + 1 = r + 1, \qquad (2.13)$$

которое носит название границы Синлгтона линейного *(N, K, D)* кода. Применительно к алгебраическому построению сверточного кода выражение (2.11) дает верхнюю оценку *d* и/или d_∞ для фиксированного показателя *r*.

Предложенный алгоритм (рис. 2.4) алгебраического построения несистематического сверточного *(n, k, d)* кода оперирует с кодами БЧХ и РС. Для примитивного кода БЧХ показатель $r = degP(x)$ определяется как сумма степеней минимальных многочленов элементов $\alpha^i \in GF((q^m)^M)$, образующих непрерывную цепочку длиной не менее *t*, $D = 2t + 1$. Для непримитивных кодов БЧХ показатель $r = degP(x)$ определяется как сумма степеней минимальных многочленов элементов $\beta^i \in GF((q^m)^M)$ таких, что их порядок равен *N*, т.е. $\beta^i = \alpha^{jg}$; $j = 1, 2, \ldots, M/2$; $N = ((q^m)^M - 1)/g$. В обоих случаях для кодов БЧХ показатель $r = degP(x)$ не удовлетворяет границе (2.13), следовательно, не обеспечивает высокое значение *d* и/или d_∞ алгебраически заданного сверточного *(n, k, d)* кода.

Важным классом циклических кодов являются коды РС. Коды РС – такие коды БЧХ, у которых мультипликативный порядок алфавита символов кодового слова делится на длину кода. Параметры *(N, K, D)* кода РС над $GF(q^m)$ связаны следующим соотношением:

$$N = q^m - 1, N - K = D - 1.$$

Следовательно, коды РС являются оптимальными в смысле границы Синлгтона и являются кодами с максимально достижимым кодовым расстоянием. Практически это означает, что при фиксированных N и K не существует кода, у которого минимальное расстояние D больше, чем у кода РС. Таким образом, несистематический сверточный (n, k, d) код, заданный порождающим многочленом кода РС, будет обладать лучшими конструктивными характеристиками по сравнению с остальными циклическими кодами.

Проведем оценку конструктивных параметров сверточных (n, k, d) кодов, алгебраически заданных порождающим многочленом кода РС.

Зафиксируем конечное поле $GF(2^2)$ и рассмотрим коды РС с параметрами: $N = 2^2 - 1 = 3$, $3 - K = D - 1$. В табл. 2.2 представлены параметры кодов РС над $GF(2^2)$, конструктивные параметры сверточных (n, k, d) кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное и истинное значение свободного кодового расстояния. Случаи для кодов $(n, 1, n)$ соответствуют тривиальному коду с повтором символов.

Таблица 2.2

Конструктивные характеристики двоичных сверточных кодов, заданных через порождающий многочлен кода РС над $GF(2^2)$

(N, K, D)	(n, k, d)	v	R	$d_П$	d_∞
(3, 1, 3)	(6, 3, 3)	2	1 / 2	4	5
(3, 2, 2)	(4, 2, 3)	1	1 / 2	2,7	3

Зафиксируем конечное поле $GF(2^3)$ и рассмотрим коды РС с параметрами $N = 2^3 - 1 = 7$; $7 - K = D - 1$. В табл. 2.3 представлены параметры кодов РС над $GF(2^3)$, конструктивные параметры сверточных (n, k, d) кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное значение свободного кодового расстояния и истинное кодовое расстояние полученных сверточных кодов.

Таблица 2.3

Конструктивные характеристики двоичных сверточных кодов, заданных через порождающий многочлен кода РС над $GF(2^3)$

(N, K, D)	(n, k, d)	v	R	$d_П$	d_∞
1	2	3	4	5	6
(7, 1, 7)	(21, 7, 7)	6	1 / 3	12	15
	(21, 14, 7)	12	2 / 3	12	13
	(12, 8, 4)	6	2 / 3	6,9	7

Продолжение таблицы 2.3

(7, 2, 6)	(18, 6, 6)	5	1 / 3	10,3	12
	(18, 12, 6)	10	2 / 3	10,3	11
(7, 3, 5)	(15, 5, 5)	4	1 / 3	8,6	9
	(15, 10, 5)	8	2 / 3	8,6	8

(7, 4, 4)	(12, 4, 4)	3	1 / 3	6,9	8
(7, 5, 3)	(9, 3, 3)	2	1 / 3	5,1	6
	(9, 6, 3)	4	2 / 3	5,1	5
(7, 6, 2)	(6, 2, 2)	1	1 / 3	3,4	4
	(6, 4, 2)	2	2 / 3	3,4	3

Зафиксируем конечное поле $GF(2^4)$ и рассмотрим коды РС с параметрами $N = 2^4 - 1 = 15$; $15 - K = D - 1$. В табл. 2.4 представлены параметры кодов РС над $GF(2^4)$, конструктивные параметры сверточных (n, k, d) кодов, $d \geq D$, алгебраически заданных порождающим многочленом кода РС, предсказанное значение свободного кодового расстояния и истинное кодовое расстояние полученных сверточных кодов.

Таблица 2.4

Конструктивные характеристики двоичных сверточных кодов, заданных через порождающий многочлен кода РС над $GF(2^4)$

(N, K, D)	*(n, k, d)*	*v*	*R*	$d_П$	d_∞
1	2	3	4	5	6
(15, 1, 15)	(60, 15, 15)	14	1 / 4	32	35
	(60, 30, 15)	28	1 / 2	32	33
	(60, 45, 15)	42	3 / 4	32	32
(15, 2, 14)	(56, 14, 14)	13	1 / 4	29,9	30
	(56, 28, 14)	26	1 / 2	29,9	31
	(56, 42, 14)	39	3 / 4	29,9	30

Продолжение табл. 2.4

1	2	3	4	5	6		
		(52, 13, 13)	12	1 / 4	27,8	29	
			(52, 26, 13)	24	1 / 2	27,8	28
				(52, 39, 13)	36	3 / 4	27,8	27
				(48, 12, 12)	11	1 / 4	25,6	28
				(48, 24, 12)	22	1 / 2	25,6	27
				(48, 36,	33	3 / 4	25,6	26

	12)				
(15, 5, 11)	(44, 11, 11)	10	1 / 4	23,5	26
	(44, 22, 11)	20	1 / 2	23,5	25
	(44, 33, 11)	30	3 / 4	23,5	24
(15, 6, 10)	(40, 10, 10)	9	1 / 4	21,3	23
	(40, 20, 10)	18	1 / 2	21,3	22
	(40, 30, 10)	27	3 / 4	21,3	21
(15, 7, 9)	(36, 9 , 9)	8	1 / 4	19,2	22
	(36, 18, 9)	16	1 / 2	19,2	21
	(36, 27, 9)	24	3 / 4	19,2	21
(15, 8, 8)	(32, 8 , 8)	7	1 / 4	17,1	19
	(32, 16, 8)	14	1 / 2	17,1	18
	(32, 24, 8)	21	3 / 4	17,1	18
(15, 9, 7)	(28, 7 , 7)	6	1 / 4	14,9	17
	(28, 14, 7)	12	1 / 2	14,9	16
	(28, 21, 7)	18	3 / 4	14,9	15
(15, 10, 6)	(24, 6 , 6)	5	1 / 4	12,8	14
	(24, 12, 6)	10	1 / 2	12,8	13
	(24, 18, 6)	15	3 / 4	12,8	13

Продолжение табл. 2.4

	2	3	4	5	6	
1		(20, 5, 5)	4	1 / 4	10,7	12
		(20, 10, 5)	8	1 / 2	10,7	12
		(20, 15, 5)	12	3 / 4	10,7	10
(15, 12, 4)		(16, 4, 4)	3	1 / 4	8,5	11
		(16, 8, 4)	6	1 / 2	8,5	10
		(16, 12, 4)	9	3 / 4	8,5	9
(15, 13, 3)		(12, 3, 3)	2	1 / 4	6,4	8
		(12, 6, 3)	4	1 / 2	6,4	7
		(12, 9, 3)	6	3 / 4	6,4	7
(15, 14, 2)		(8, 2 , 2)	1	1 / 4	4,3	6
		(8, 4 , 2)	2	1 / 2	4,3	5
		(8, 6 , 2)	3	3 / 4	4,3	5

Анализ полученных результатов показывает, что предложенные алгебраические методы синтеза сверточных кодов позволяют без вычислительно сложных процедур переборного поиска синтезировать сверточные коды с высокими конструктивными свойствами.

Практическое применение разработанных методов синтеза сверточных кодов позволяет решить важную решить важную научную задачу поиска эффективных сверточных кодов с большой длиной кодового ограничения. Актуальным направлением дальнейших исследований является разработка методов и алгоритмов декодирования алгебраически заданных сверточных кодов, позволяющих реализовать потенциальные возможности синтезируемых кодовых конструкций для повышения достоверности передаваемой информации.

Выводы

1. На основе единого общетеоретического подхода, с использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов получили дальнейшее развитие алгебраические методы и вычислительно эффективные алгоритмы синтеза нерекурсивных сверточных кодов.

2. Разработанные вычислительно эффективные (вычислительно реализуемые) алгебраические методы синтеза (алгебраически заданных) нерекурсивных сверточных кодов, отличаются от известных использованием ограничения недвоичного циклического кода на произвольное подполе, что позволяет синтезировать (алгебраически заданные) нерекурсивные сверточные коды с произвольными свойствами и кодовыми характеристиками.

3. Теоретически обоснованы процедуры алгебраического построения нерекурсивных сверточных кодов через обобщение циклических кодов на случай бесконечной длины. Доказанные теоремы позволяют аналитически связать параметры несистематических циклических кодов с конструктивными параметрами соответствующих сверточных кодов. Предложенные в работе алгебраические процедуры построения нерекурсивных сверточных кодов позволяют за конечное число шагов однозначно определить правило сверточного кодирования и построить схему нерекурсивного сверточного кодера с заданными конструктивными характеристиками.

4. Получили дальнейшее развитие методы кодирования алгебраически заданными нерекурсивными сверточными кодами, отличающиеся от известных теоретически обоснованными процедурами алгебраического построения некурсивных сверточных кодов через обобщение циклических кодов на случай бесконечной длины, что позволяет аналитически формализовать процесс помехоустойчивого кодирования синтезируемыми нерекурсивными сверточными кодами с высокими (конструктивными) кодовыми характеристиками.

5. Проведенные исследования свойств синтезированных нерекурсивных сверточных кодов, алгебраически заданных порождающими многочленами недвоичных циклических кодов, показали, что полученные коды близки по своим характеристикам к оптимальным кодам.

6. Проведенные исследования свойств алгебраически заданных нерекурсивных сверточных кодов показали, что для построения эффективных кодов в несистематическом виде следует использовать низкоскоростные циклические коды. Это позволяет получить практически весь спектр скоростей сверточного кодирования и высокие конструктивные параметры даже при небольшом кодовом ограничении.

7. Применение разработанных методов синтеза нерекурсивных сверточных кодов позволяет за счет использования алгебраических процедур и полиномиальных методов описания циклических кодов решить важную научную задачу поиска эффективных нерекурсивных сверточных кодов с большой длиной кодового ограничения.

РАЗДЕЛ 3
АЛГЕБРАИЧЕСКИЕ МЕТОДЫ СИНТЕЗА РЕКУРСИВНЫХ СВЕРТОЧНЫХ КОДОВ

В качестве составляющих турбокод кодов необходимо использовать рекурсивные сверточные коды, что связано с особенностями весового распределения кодовых слов рекурсивных сверточных кодов. В данном разделе на основе единого общетеоретического подхода с использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов разрабатываются алгебраические методы синтеза рекурсивных сверточных кодов, исследуются процедуры построения рекурсивных сверточных кодов, разрабатываются алгоритмы для их реализации. Применение разработанных методов синтеза рекурсивных сверточных кодов позволяет за счет использования алгебраических процедур и полиномиальных методов описания циклических кодов решить важную научную задачу поиска эффективных рекурсивных сверточных кодов с большой длиной кодового ограничения.

3.1. Алгебраические методы синтеза несистематических рекурсивных сверточных кодов

При построении алгебраических несистематических рекурсивных сверточных кодов воспользуемся известными свойствами циклических кодов .

Каждый линейный *(n, k, d)* код над *GF(q)* является подпространством $GF^k(q)$ пространства $GF^n(q)$. Циклический код является частным случаем подпространства, так как обладает дополнительным свойством цикличности. Каждый вектор из $GF^n(q)$ представим многочленом от формальной переменной *x* степени не выше *n –1*. Компоненты вектора отождествим . коэффициентам этого многочлена. Множество многочленов обладает структурой векторного пространства, идентичной структуре пространства $GF^n(q)$, а также структурой кольца многочленов $GF(q)[x]/(x^n - 1)$.

В кольце многочленов их умножение определяется

,

а циклический сдвиг записывается в виде выражения

.

Если кодовые слова *(n, k, d)* кода над *GF(q)* задаются в виде многочленов, то код является подмножеством кольца $GF(q)[x]/(x^n - 1)$. Код является циклическим, если вместе с кодовым словом *c(x)* он содержит также многочлен $x \cdot c(x)$.

Любой циклический код можно задать через порождающий многочлен *g(x)* [13-18]. Пусть *g(x)* единственный приведенный ненулевой многочлен наименьшей степени $r = n - k$ однозначно задает *(n, k, d)* циклический код

над *GF(q)* и обозначается порождающим многочленом

,

где $\beta^i \in GF(q^m)$.

В то же время циклический код можно однозначно задать другим многочленом – мультипликативно обратным многочлену *g(x)*. При этом единственный многочлен *h(x)* (проверочный многочлен), мультипликативно обратный приведенному ненулевому многочлену *g(x)*, однозначно задает *(n, k, d)* циклический код над *GF(q)* и обозначается проверочным многочленом, при этом, если

, (3.1)

то

, (3.2)

где $\beta^i, \beta^j \in GF(q^m), j \neq i$.

Равенство (3.2) справедливо, поскольку многочлен *g(x)* делит многочлен $x^n - 1$, который, в свою очередь, делит многочлен $x^m - 1$, так что *g(x)* делит также . Допустим α – примитивный элемент поля $GF(q^m)$, пусть $q^m - 1 = n \cdot b$, и пусть $\beta = \alpha^b$. Тогда все корни многочлена $x^n - 1$, как и корни многочлена *g(x)*, исчерпываются степенями элемента β. Простые делители многочлена $x^n - 1$ имеют своими корнями только такие элементы. Следовательно, в кольце многочленов $GF(q)[x]/(x^n - 1)$ существует некоторый многочлен *h(x),* являющийся сомножителем *g(x)* в разложении двучлена $x^n - 1$, т.е. существует многочлен *h(x)* – делитель $x^n - 1$, и корни многочлена *h(x)* так же исчерпываются степенями элемента β. Это означает, что произвольный циклический код можно однозначно задать либо порождающим многочленом *g(x),* либо мультипликативно обратным ему в кольце $GF(q)[x]/(x^n - 1)$ многочленом *h(x)* , причем, если

,

то

,

где $\beta^i, \beta^j \in GF(q^m), j \neq i$.

Из вышесказанного следует, что $deg\ h(x) = n - deg\ g(x) = n - r = k$.

Воспользуемся понятием проверочного многочлена для построения правила кодирования несистематическим циклическим кодом. Кодовое слово несистематического циклического кода можно представить в виде

$$C(x) = I(x) \cdot g(x).$$

Выразим порождающий многочлен *g(x)* через проверочный многочлен *h(x)* и двучлен $x^n - 1$:

$$g(x) = (x^n - 1)/h(x) = 1/h(x)$$

с операцией деления в кольце многочленов $GF(q)[x]/(x^n - 1)$.

После подстановки получим

$$C(x) = I(x)/h(x). \qquad (3.3)$$

Для реализации процедуры деления на многочлен воспользуемся цифровым фильтром с бесконечным импульсным откликом (рекурсивным фильтром). На рис. 3.1 приведена структурная схема цифрового рекурсивного фильтра.

Если на вход цифрового рекурсивного фильтра подать последовательность символов $\{i_k, \ldots, i_1, , i_0\}$, то считанная с выхода последовательность $\{\ldots, c_1, , c_0\}$ удовлетворяет свойству рекурсии:

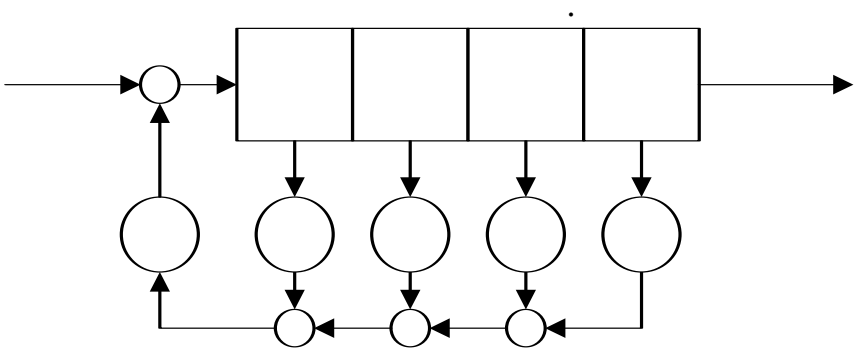


Рис. 3.1. Структурная схема цифрового рекурсивного фильтра

Зададим проверочный многочлен в виде

$$h(x) = h_0 + h_1x + h_2x^2 + \ldots + h_kx^k. \qquad (3.4)$$

Тогда структурную схему несистематического кодера, реализующего правило кодирования (3.3), представим в виде схемы на рис. 3.2.

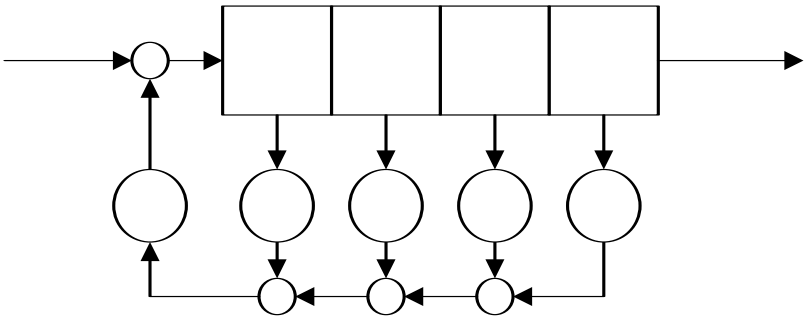


Рис. 3.2. Структурная схема несистематического кодера циклического кода, построенного с использованием проверочного многочлена

Для построения алгебраических рекурсивных несистематических сверточных кодов воспользуемся выражением (3.3). Рассмотрим процедуру сверточного кодирования с $R = 1/m$. Для построения рекурсивного кодера используем рекурсивный фильтр (рис. 3.2). Если на вход устройства подать информационный многочлен (3.2), в общем случае бесконечной длины, то выходную последовательность с символами из $GF(q^m)$ отобразим в последовательность символов из $GF(q)$. Справедлива следующая теорема.

Теорема 3.1. Несистематический сверточный код над $GF(q)$ с $R = 1/m$ однозначно задается многочленом $h(x)$ над $GF(q^m)$ вида $h(x) = h_0 + h_1x + h_2x^2 + ... + h_kx^k$. Если многочлен $h(x)$ – проверочный многочлен недвоичного (N, K, D) циклического кода над $GF(q^m)$, то он однозначно определяет (n, k) несистематический рекурсивный сверточный код над $GF(q)$ с правилом кодирования $C(x)=I(x) /h(x)$ с длиной кодового ограничения

$$\nu = K$$

и конструктивными параметрами

(3.5)

Доказательство. Циклический (N, K, D) код над $GF(q^m)$ с проверочным многочленом $h(x)$ степени K однозначно определяет набор регистров сдвига, соединенных связями (рис. 3.2), и задает рекуррентное правило кодирования

$$C(x) = I(x)/h(x).$$

Если на вход устройства подать последовательность символов из $GF(q)$, то считанная с выхода кодовая последовательность длиной N q^m – ичных символов является кодовым словом циклического (N, K, D)–кода над $GF(q^m)$, а рекурсивный сверточный код является обобщением исходного цикличного кода на непрерывный случай. Параметры сверточного кода связаны соотношениями

$$\nu = K \cdot k^0 = r;\ k^0 = 1;\ n^0 = m;\ k = K + 1;\ n = (K + 1) \cdot n^0 = k \cdot m;\ R = 1 / m$$

и два любых кодовых слова будут отличаться, по крайней мере, в D q^m – ичных символов. Отображение элементов поля $GF(q^m)$ в элементы поля $GF(q)$ не уменьшает кодовое расстояние между произвольными q – ичными кодовыми словами, следовательно, $d_K \geq D$. По определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_{K+1} \leq d_{K+2} \leq ... \leq d_\infty$, откуда $d_\infty \geq d_K$, что и завершает доказательство.

Для рассмотрения процедуры сверточного кодирования с $R = k^0 / m$ сформулируем и докажем следующую теорему.

Теорема 3.2. Если зафиксировать конечное множество H элементов поля $GF(q^m)$, причем $\log_q |H| = k^0$, $m \geq k^0$, то проверочный многочлен циклического (N, K, D) кода над $GF(q^m)$ полностью определяет несистематический рекурсивный сверточный (n, k, d) код над $GF(q)$ с информационным кадром длины k^0, длиной кодового ограничения

$$\nu = K \cdot k^0$$

и параметрами

(3.6)

Доказательство. Представим информационную последовательность в виде многочлена с коэффициентами над H, т.е. коэффициенты многочлена $I(x)$ представим в виде многочленов над $GF(q)$ степени $m - 1$:

,

где $z_i \in GF(q)$, причем $m - k^0$ коэффициентов z_i заданы произвольно.

Положим, для определенности, $z_i = 0$ для $i = k^0, \ldots, m - 1$. Первые k^0 элементов z_i образуют информационный кадр k^0 символов над $GF(q)$. Определенное таким образом отображение символов $GF(q)$ в символы $GF(q^m)$ является однозначным соответствием. Недвоичный (N, K, D) циклический код над $GF(q^m)$ с проверочным многочленом $h(x)$ однозначно задает рекуррентное правило кодирования

$$C(x) = I(x)/h(x).$$

При кодировании каждому информационному кадру длиной k^0 символов над $GF(q)$ (или, что эквивалентно, каждому символу из множества H) ставится в соответствие кадр кодовых символов длиной n^0.

Степень K проверочного многочлена $h(x)$ циклического (N, K, D) кода над $GF(q^m)$ задает длину кодирующего регистра и, соответственно, число хранящихся в кодере информационных кадров. Следовательно, длина кодового ограничения ν, конструктивные параметры n и k и скорость R сверточного кодирования определяются, соответственно, следующими выражениями:

$$\nu = K \cdot k^0;\ k = (K + 1) \cdot k^0;\ n = k \cdot n^0 / k^0;\ R = k^0 / m,\ m \geq k^0.$$

Если на вход устройства (рис. 3.2) подать K информационных кадров по k^0 q–ичных символов (что эквивалентно подаче K кадров по одному –ичному символу), то снятая с выхода кодовая последовательность длиной N q^m–ичных символов является кодовым словом циклического (N, K, D) кода над $GF(q^m)$. Следовательно, два любых кодовых блока, соответствующих двум произвольным входным последовательностям длиной K –ичных символов, будут отличаться, по крайней мере, в D q^m–ичных символов.

Отображение элементов поля $GF(q^m)$ в элементы поля $GF(q)$ не уменьшает кодовое расстояние между произвольными q–ичными кодовыми словами, следовательно, $d_K \geq D$. По определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_{r+1} \leq d_{r+2} \leq \ldots \leq d_\infty$, откуда $d_\infty \geq d_K$, что и завершает доказательство.

Теоремы 3.1. – 3.2 определяют механизм построения алгебраических рекурсивных несистематических сверточных кодов. Их параметры алгебраически связаны с параметрами недвоичных циклических кодов, что позволяет конструктивно строить рекурсивные сверточные коды с требуемыми свойствами. Общая схема сверточного кодера приведена на рис. 3.4 с дополнительно включенными входными и выходными буферами для отображения символов из $GF(q^m)$ в $GF(q)$ и обратно. Такой кодер реализует обработку символов из $GF(q^m)$.

Для построения схемы рекурсивного сверточного кодера с обработкой символов из $GF(q)$ рассмотрим несистематическое кодирование через умножение информационного многочлена на порождающие многочлены $P_1(x), P_2(x), \ldots, P_m(x)$.

Предположим, что некоторый многочлен $P_i(x)$ является делителем двучлена $x^n - 1$. Тогда многочлен $P_i(x)$ порождает циклический (n, k) код над $GF(q)$. Построим проверочный многочлен $h_i(x)$, который так же однозначно задает циклический (n, k) код над $GF(q)$. Используя цифровой рекурсивный фильтр (3.1), получим схему несистематического рекурсивного сверточного кодера с обработкой символов из $GF(q)$(рис. 3.3).

Рассмотрим пример алгебраического синтеза несистематического рекурсивного сверточного кода в конечном поле $GF(2^3)$, построенное по кольцу многочленов с операциями по модулю многочлена $x^3 + x + 1$. Рассмотренный пример демонстрирует конструктивность предложенного подхода алгебраического построения рекурсивных сверточных кодов.

Пример 3.1. Синтез несистематического рекурсивного сверточного кода. Зафиксируем конечное поле $GF(2^3)$, построенное по кольцу многочленов с операциями по модулю многочлена $x^3 + x + 1$. Элементы поля приведены в табл. 3.1.

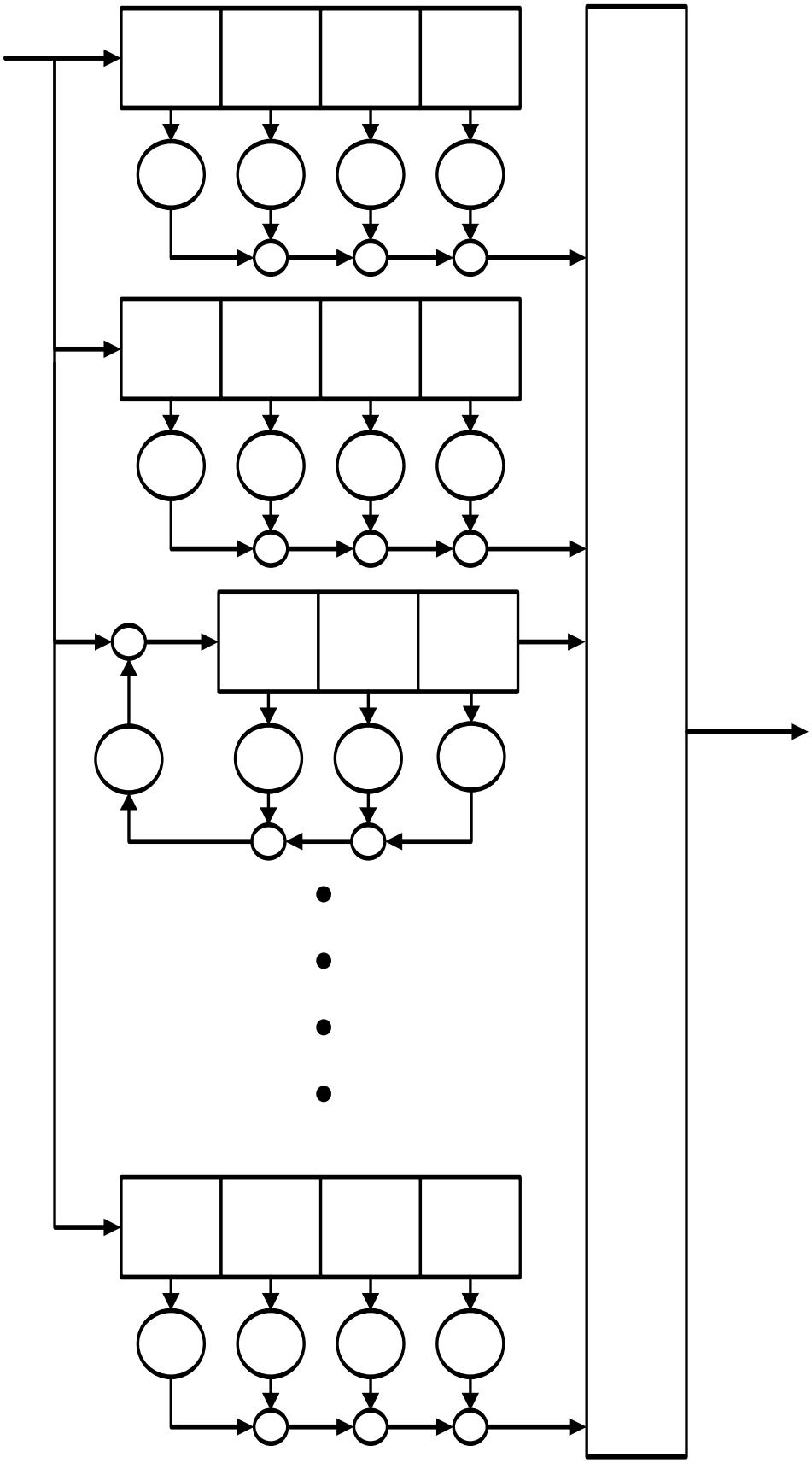


Рис. 3.3. Схема несистематического кодера алгебраического рекурсивного сверточного кода с обработкой элементов из *GF(q)*

Зафиксируем РС код *(7, 3, 5)* над $GF(2^3)$ с порождающим многочленом

$g(x) = (x + \alpha^0)\cdot(x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3) =$

$$=x^4 + \alpha^2 \cdot x^3 + \alpha^5 \cdot x^2 + \alpha^5 \cdot x + \alpha^6.$$

Мультипликативно обратный многочлену $g(x) = x^4 + \alpha^2 \cdot x^3 + \alpha^5 \cdot x^2 + \alpha^5 \cdot x + \alpha^6$ в кольце $GF(q)[x]/(x^n - 1)$ является многочлен $h(x) = x^3 + \alpha^2 \cdot x^2 + x + \alpha^2$. Воспользовавшись результатами теорем 3.1 – 3.3, можем получить рекурсивные сверточные коды в несистематическом виде со следующими параметрами:

1.Двоичный сверточный *(n, k, d)* код с параметрами: $k^0 = 1$; $n^0 = 3$; $\nu = 3$; $k = 4$; $n = 12$; $R = 1 / 3$; $d_\infty \geq 5$; $d_П = 8{,}57$.

2.Двоичный сверточный код *(n, k, d)* код с параметрами: $k^0 = 2$; $n^0 = 3$;$\nu = 6$; $k = 8$; $n = 12$; $R = 2 / 3$; $d_\infty \geq 5$; $d_П = 8{,}57$.

Таблица 3.1

Элементы конечного поля $GF(2^3)$

0.	0	0	0	*0*	$\alpha^{-\infty}$
0.	0	0	1	*1*	α^0
0.	0	1	0	x	α^1
0.	1	0	0	x^2	α^2
0.	0	1	1	$x + 1$	α^3
0.	1	1	0	$x^2 + x$	α^4
0.	1	1	1	$x^2 + x + 1$	α^5
0.	1	0	1	$x^2 + 1$	α^6

Построим кодер с обработкой элементов из $GF(2^3)$, т.е. пакетами по 3 бита (как на рис. 3.2). На рис. 3.4 представлена схема рекурсивного кодера с обработкой символов из $GF(2^3)$.

Для построения кодера с обработкой двоичных символов рассмотрим порождающие многочлены алгебраического сверточного кода, заданного через порождающий многочлен РС кода:

$$g_1(x) = x^3 + x^2 + x + 1;$$
$$g_2(x) = x^2 + x;\ g_3(x) = x^4 + x^2 + x + 1.$$

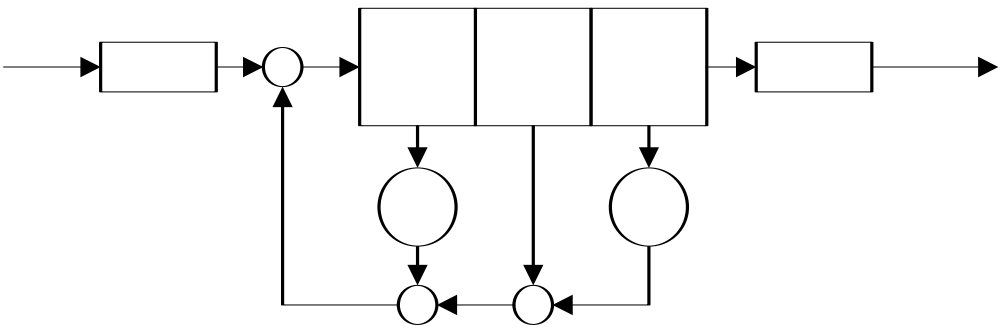


Рис. 3.4. Схема рекурсивного кодера алгебраического сверточного кода (12, 4)/(12, 8) с обработкой символов из $GF(2^3)$

Многочлен $g_3(x)$ является делителем двучлена $(x^n - 1)$, его мультипликативно обратным элементом в кольце $GF(q)[x]/(x^n - 1)$ является

многочлен $h_3(x) = x^3 + x + 1$. Схема соответствующего рекурсивного кодера приведена на рис. 3.5.

Следует отметить, что современные методы построения турбокодов оперируют систематическими рекурсивными сверточными кодами. Это позволяет формировать кодовое слово с использованием одной и той же информационной части (аналогичной для двух сверточных кодеров) и двух различных проверочных частей (для каждого сверточного кодера). Этот прием позволяет существенно повысить информационную скорость передачи при сохранении фиксированного показателя помехоустойчивости, что, соответственно, ведет к росту энергетической эффективности турбокодирования. Таким образом, проблема дальнейшей разработки и исследование алгебраических методов построения систематических сверточных кодов, их эффективная реализация на цифровых рекурсивных фильтрах с бесконечным импульсным откликом представляется весьма актуальной.

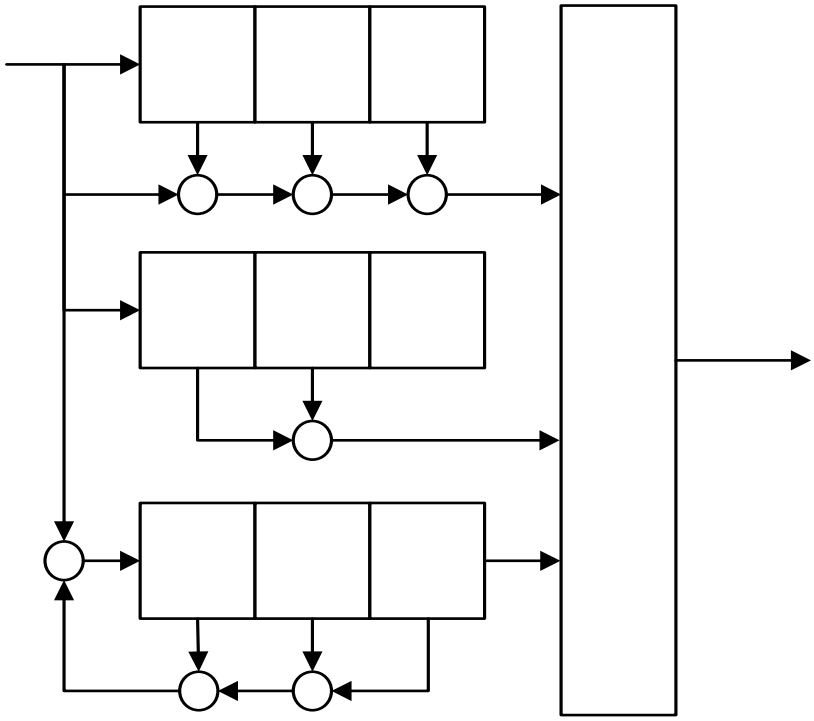


Рис. 3.5. Схема рекурсивного кодера алгебраического сверточного кода (12, 4)/(12, 8) с обработкой двоичных символов

3.2. Алгебраические методы синтеза систематических рекурсивных сверточных кодов

Для решения задачи алгебраического построения рекурсивных систематических сверточных кодов воспользуемся систематическими циклическими кодами. Рассмотрим блок данных длины n символов, поместим информационные символы в старшие разряды и подберем

проверочные символы так, чтобы получить разрешенное кодовое слово, т.е. кодовое слово, принадлежащее циклическому *(n, k, d)* коду. Кодовое слово систематического циклического кода можно записать как

$$C(x) = x^{n-k}I(x) + T(x), \qquad (3.7)$$

где

$$T(x) = -R_{g(x)}[x^{n-k}I(x)],$$

так что

$$R_{g(x)}[C(x)] = 0.$$

Для реализации процедуры систематического кодирования циклического кода воспользуемся цифровым фильтром с бесконечным импульсным откликом (рекурсивным фильтром), который реализует цепь деления на многочлен. Пусть коэффициенты порождающего многочлена $g(x) = g_0 + g_1x + g_2x^2 + \ldots + g_rx^r$ равны весовым множителям в отводах регистра сдвига рекурсивного фильтра. Тогда схема кодера, реализующего систематическое кодирование циклических кодов по выражению (3.7) может быть представлена в виде, отображенном на рис. 3.6.

Действительно, если на вход устройства деления поступает произвольная последовательность, представленная в виде многочлена $I(x) = I_0 + I_1x + I_2x^2 + \ldots + I_{n-1}x^{n-1}$, то рекуррентные равенства процедуры деления многочленов запишем в виде

,

где $Q^{(i)}(x)$ и $R^{(i)}(x)$ – соответственно частное и остаток на i–ом шаге рекурсии с начальными значениями

$Q^{(0)}(x) = 0$ и $R^{(0)}(x) = I(x)$, ,

и после k шагов итерации получаются частное $Q^{(k)}(x)$ и остаток $R^{(k)}(x)$.

Перепишем последнее выражение в виде

,

так что

и

.

Тогда деление многочлена $I(x)$ на многочлен $g(x)$ описывают процессы, происходящие в изображенном на рис. 3.6 регистре сдвига.

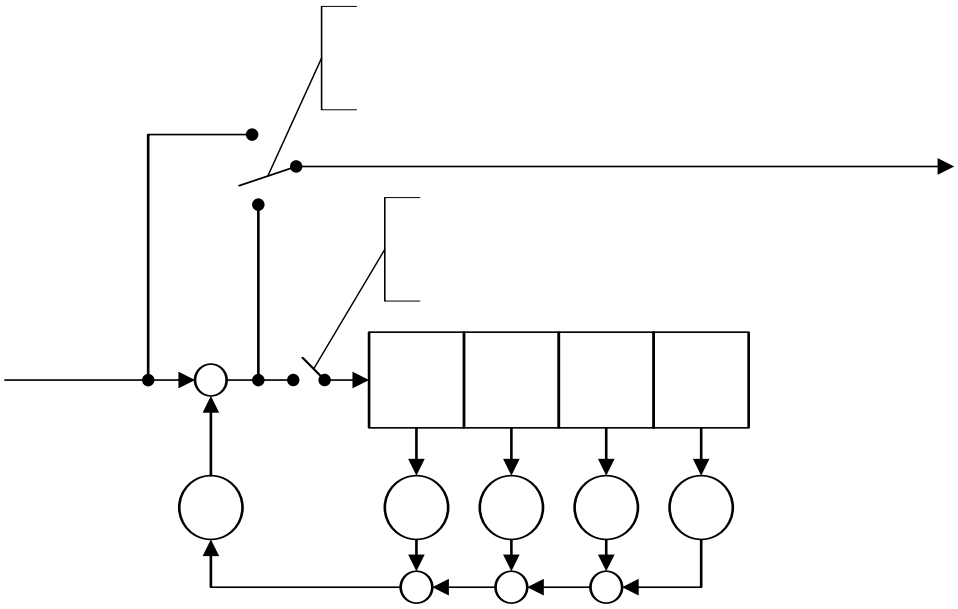


Рис. 3.6. Схема систематического кодера циклических кодов

Обобщим систематическое кодирование циклических кодов на случай бесконечной длины. Используем полученную схему для построения кодера алгебраического систематического рекурсивного сверточного кода. Схема такого кодера представлена на рис. 3.7. Устройство работает следующим образом.

Предположим, что используется циклический *(N, K, D)* код, заданный порождающим многочленом *g(x)*, *deg g(x)= r = N – K*. На вход устройства подадим непрерывный поток символов из *GF(q)*. В первом буфере входные символы накапливаются в блок из *K* символов поля *GF(q)* и подаются на вход мультиплексора и на вход рекурсивного фильтра. Через *K* тактов цифровой фильтр начинает формировать проверочные символы, которые поступают на вход второго буфера. После *N – K* тактов будут сформированы все проверочные символы и поданы на вход мультиплексора. Обработка данных осуществляется посимвольно элементами из *GF(q)*. Скорость кода *R = K /N,* его параметры определяет следующая теорема.

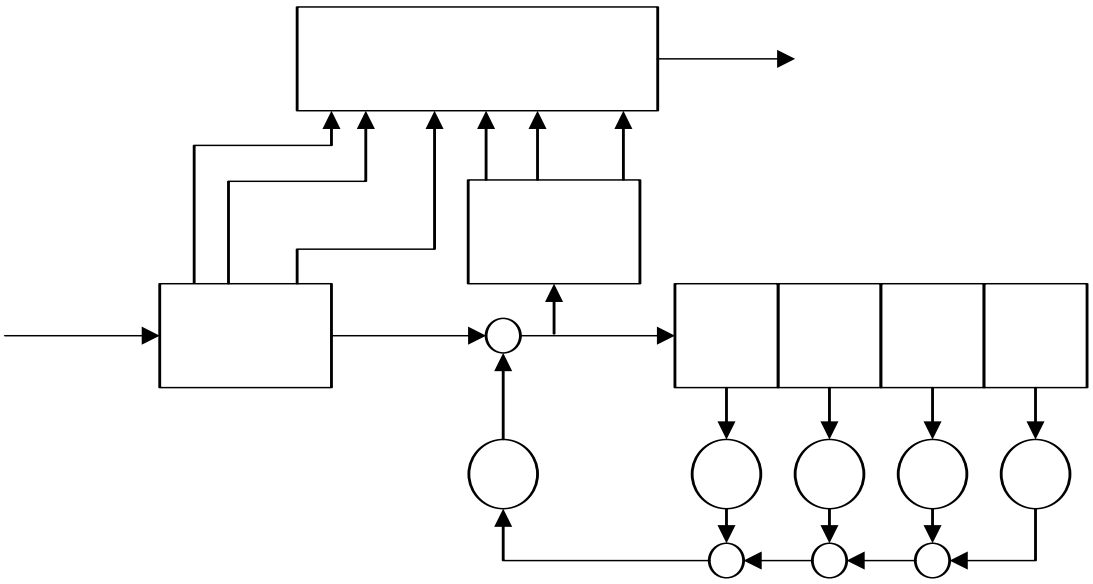


Рис. 3.7. Схема систематического кодера
алгебраического сверточного кода

Теорема 3.3. Порождающий многочлен *g(x)* циклического *(N, K, D)* кода над *GF(q)* полностью определяет рекурсивный систематический сверточный *(n, k, d)* код над *GF(q)* с кодовым ограничением

$$\nu = (N-K)\cdot K$$

и параметрами

(3.8)

Доказательство. Как известно, кодер систематического циклического *(N, K, D)* кода однозначно определяет рекурсивный фильтр с коэффициентами многочлена *g(x)* в виде весовых множителей в отводах регистра. Степень $r = N - K$ многочлена *g(x)* задает длину регистра и, соответственно, число хранящихся в регистре символов. Следовательно, $\nu = (N - K)\cdot k^0$. Если на вход устройства (см. рис. 3.7) подавать непрерывный поток символов из *GF(q)*, то на выходе мультиплексора получится кодовое слово непрерывного кода. Если длина входного (информационного) кадра равна $k^0 = K$, а длина выходного кадра (кадра кодовых символов) равна $n^0 = K + N - K = N$, то полученная на выходе мультиплексора последовательность является кодовым словом систематического циклического кода. Скорость кода определяется выражением $R = K/N$. Для кодовых слов, соответствующих различным информационным кадрам, выполняется условие $d_1 \geq D$. По

определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_1 \le d_2 \le \ldots \le d_\infty$, откуда имеем $d_\infty \ge d_1$.

Применение недвоичных циклических кодов позволяет дополнительно варьировать степенью расширения поля $GF(q^m)$ при изменении параметров сверточного кода над $GF(q)$. Устройство, приведенное на рис. 3.7, в этом случае будет работать следующим образом. На вход устройства подадим непрерывный поток символов из $GF(q)$. В первом буфере входные символы накапливаются и преобразуются в K символов из $H \subseteq GF(q^m)$. На вход мультиплексора с первого буфера поступают $K \cdot \lfloor \log_q H \rfloor$ символов из $GF(q)$. На вход рекурсивного фильтра с первого буфера поступают K символов, каждый из которых отождествлен символу из $GF(q^m)$. Через K тактов цифровой фильтр начинает формировать проверочные символы из $GF(q^m)$, которые поступают на вход второго буфера. После $N - K$ тактов будут сформированы все проверочные символы и поданы на вход мультиплексора в виде $(N - K) \cdot m$ символов из $GF(q)$. Таким образом, обработка входных данных в кодере, представленном на рис. 3.7, осуществляется пакетами по m символов из $GF(q)$ или, что эквивалентно, по одному элементу из $GF(q^m)$. Скорость полученного непрерывного кода над $GF(q)$ составит $K \cdot \lfloor \log_q H \rfloor / ((N - K) \cdot m + K \cdot \lfloor \log_q H \rfloor)$ и может изменяться в пределах

$$K / ((N - K) \cdot m + K) \le R \le K / N, \qquad (3.9)$$

в зависимости от мощности множества H. Параметры кода определяются следующей теоремой.

Теорема 3.4. Если зафиксировать $GF(q^m)$ и конечное множество H элементов поля $GF(q^m)$, причем $\log_q \lfloor H \rfloor = k^0$, $m \ge k^0$, то порождающий многочлен $g(x)$ циклического (N, K, D) кода над $GF(q^m)$ полностью определяет рекурсивный систематический сверточный (n, k, d) код над $GF(q)$ с кодовым ограничением

$$\nu = (N - K) \cdot K \cdot \lfloor \log_q H \rfloor$$

и параметрами

(3.10)

Доказательство. Если на вход кодера (рис. 3.7) подать непрерывный поток символов из $GF(q)$, разбить их на блоки по $k^0 = K \cdot \log_q \lfloor H \rfloor$ символов и сопоставить набору символов из $GF(q^m)$, а кодирование в рекурсивном фильтре осуществлять элементами из $GF(q^m)$, то на выходе мультиплексора получим кодовое слово непрерывного кода.

Если при этом весовые множители будут задаваться коэффициентами многочлена $g(x)$ циклического (N, K, D) кода над $GF(q^m)$, то произвольное слово длины N символов из $GF(q^m)$ на выходе цифрового фильтра будет кодовым словом циклического (N, K, D) кода.

Длина входного (информационного) кадра равна $k^0 = K \cdot \log_q |H|$ и зависит от мощности множества H. Длина выходного кадра (кадра кодовых символов) равна

$$n^0=((N–K)\cdot m+K\cdot|\log_q H|)$$

и также зависит от мощности множества H. Скорость кода определяется, соответственно, выражением

$$R = K\cdot\log_q |H| /((N–K)\cdot m+K\cdot|\log_q H|),$$

а

$$k=(N–K+1)\cdot K\cdot|\log_q H|;\ n=(N–K+1)\cdot((N–K)\cdot m+K\cdot|\log_q H|).$$

Для кодовых слов, соответствующих различным информационным кадрам, выполняется условие $d_1 \geq D$. По определению дистанционного профиля непрерывных кодов выполняется равенство $d = d_1 \leq d_2 \leq \ldots \leq d_\infty$, откуда имеем $d_\infty \geq d_1$.

Следствие 1. Если $H = GF(q^m)$, то $\log_q |H| = m$ и, очевидно,

$$k^0=K\cdot m;\ n^0=((N–K)\cdot m+K\cdot m) = N\cdot m,\ k=(N–K+1)\cdot K\cdot m;$$
$$n=(N–K+1)\cdot((N–K)\cdot m+K\cdot m) = (N–K+1)\cdot N\cdot m;$$
$$R=K\cdot m/((N–)\cdot m+K\cdot m) = K\cdot m/N\cdot m = K/N,\ d_\infty \geq D,$$

что соответствует обобщению результата теоремы 3.6 на недвоичные коды. Скорость кода соответствует верхней границе в выражении (3.9).

Следствие 2. Если $H = GF(q)$, то $\log_q |H|=1$ и, очевидно, $k^0=K$; $n^0=((N–K)\cdot m+K)$; $k=(N–K+1)\cdot K$; $n=(N–K+1)\cdot((N–K)\cdot m+K)$; $R=K/((N–K)\cdot m+ K)$, $d_\infty \geq D$, что соответствует нижней границе скорости в выражении (3.9).

Рассмотрим пример алгебраического синтеза систематического рекурсивного сверточного кода.

Пример 3.2. Синтез систематического рекурсивного сверточного (вариант 1). Рассмотрим конечное поле $GF(2^3)$, построенное по кольцу многочленов $\{0 = \alpha^{-\infty}, 1 = \alpha^0, x = \alpha^1, x^2 = \alpha^2, x + 1 = \alpha^3, x^2 + x = \alpha^4, x^2 + x + 1 = \alpha^5, x^2 + 1 = \alpha^6\}$ по модулю $G(x) = x^3 + x + 1$. Рида Соломона (7, 5, 3) с порождающим многочленом $g(x) = (x + \alpha^0)\cdot(x + \alpha^1) = x^2 + \alpha^3 x + \alpha^1$.

Воспользуемся результатом теоремы 3.4. Получим систематический сверточный (n, k, d) код над $GF(2^3)$ с кодовым ограничением $\nu = 10$ и параметрами $k^0 = 5$; $n^0 = 7$; $k=10$; $n = 14$; $R = 5/7$; $d_\infty \geq 3$. На рис. 3.8 представлен вариант схемы такого кодера в систематическом виде.

Построим теперь двоичный алгебраический рекурсивный сверточный код с использованием результатов теоремы 3.5. Зафиксируем множество $H \subseteq GF(2^3)$ так, что $\log_2 |H| = k^0 = 2$, например, $H = \{0, 1, x^2 + x, x^2 + x + 1\}$.

Каждый элемент множества H сопоставим информационному кадру из $k^0 = 2$ бит, например, следующим образом (см. табл. 3.2).

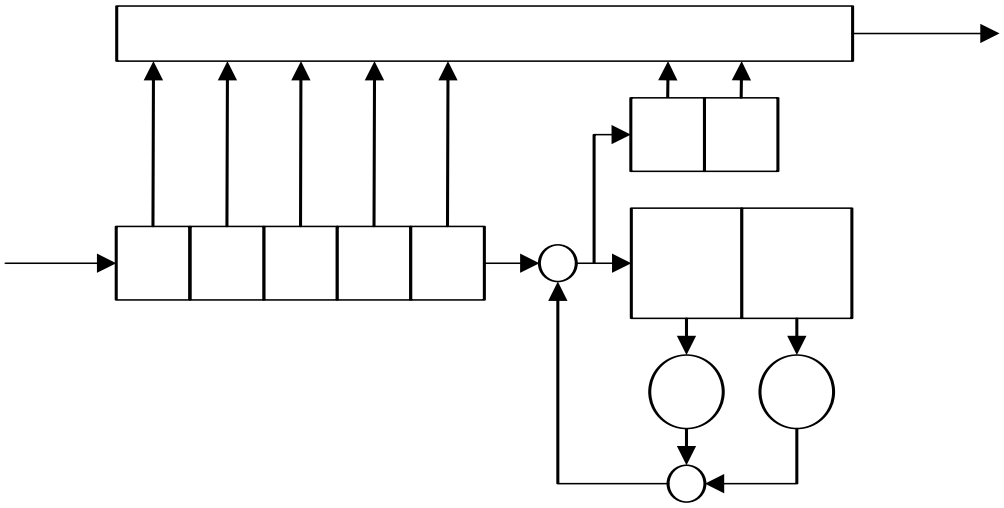


Рис. 3.8. Схема кодера алгебраического систематического рекурсивного сверточного (14,10) кода над $GF(2^3)$

Таблица 3.2

Сопоставление информационных кадров элементам множества H

Инф. кадр	↔	Элемент множества H
00	↔	0(0, 0, 0)
01	↔	α^0(0, 0, 1)
10	↔	α^4(1, 1, 0)
11	↔	α^5(1, 1, 1)

Отображение информационных кадров в символы множества H состоит только в дописывании слева (копирование) первого бита информационной последовательности. Воспользовавшись результатом теоремы 3.7, получим двоичный сверточный *(n, k, d)* код с кодовым ограничением $\nu=20$ и параметрами $k^0=10$; $n^0=16$; $k=30$; $n=48$; $R=5/8$; $d_\infty \geq 3$. Возможный вариант схемы такого кодера в систематическом виде представлен на рис. 3.9.

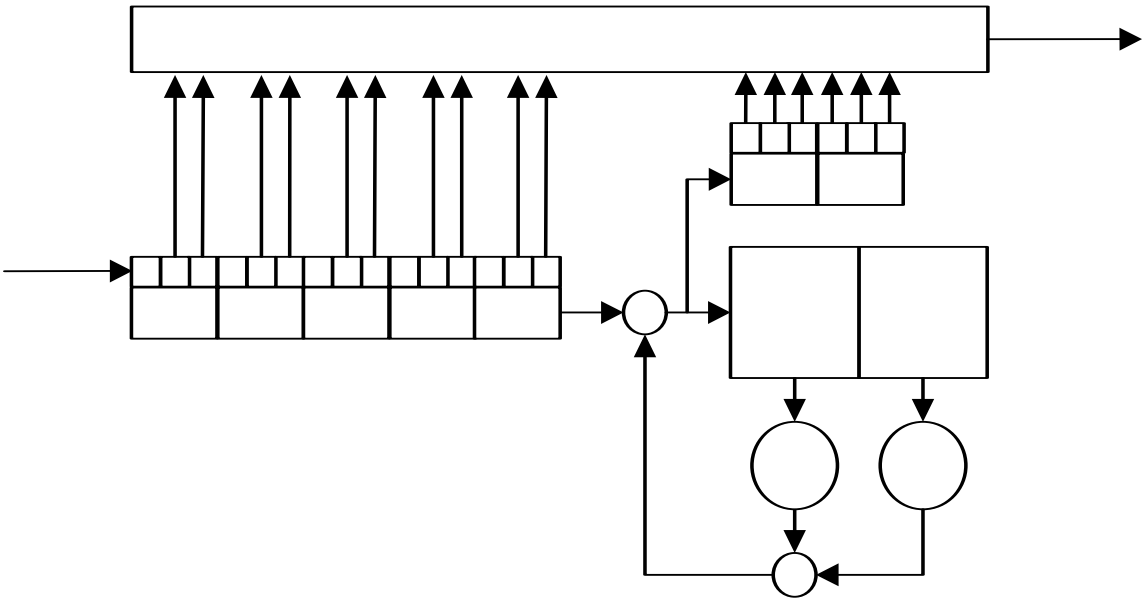


Рис. 3.9. Схема кодера алгебраического систематического двоичного сверточного *(48, 30)* кода

На практике в существующих схемах систематических рекурсивных сверточных кодов входные и выходные буферы заменяют мультиплексированием и добавляют дополнительное устройство, выполняющее процедуру кодирования проверочной части нерекурсивным несистематическим кодом. Тогда общая схема кодера может быть представлена в виде (см. рис. 3.10).

Возможен другой способ алгебраического построения систематических рекурсивных сверточных кодов. Этот способ несколько проще, но не всегда реализуем. Он состоит в возможном представлении одного из многочленов несистематического нерекурсивного алгебраического сверточного кода в виде единичного многочлена и трансформации кодера к требуемой форме.

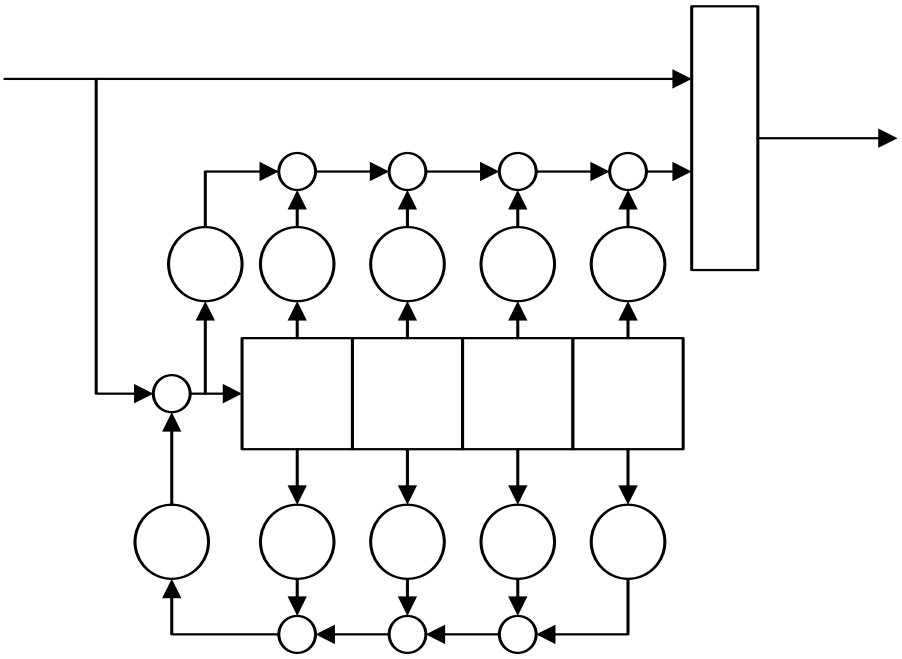


Рис. 3.10. Общая схема рекурсивного систематического кодера

Действительно, если зафиксировать конечное поле $GF(q^m)$, некоторое множество $H \subseteq GF(q^m)$, циклический (N, K, D) код над $GF(q^m)$ и образованные таким способом порождающие многочлены несистематического нерекурсивного сверточного кода $g_1(x), g_2(x), \ldots, g_m(x)$, то для построения систематического сверточного кода оказывается необходимым и достаточным выполнение условия следующей леммы.

Лемма 3.1. Для построения систематического сверточного кода с $R = 1/m$ необходимо и достаточно выполнение равенства единице любого

многочлена из выражения .

Доказательство. Пусть информационная последовательность описывается многочленом вида

и является информационной последовательностью, подлежащей кодированию. Кодовое слово $C(x)$ сверточного кода формируется путем последовательного считывания символов при одинаковых степенях многочленов

$$F_1(x) = I(x)P_1(x),\ F_2(x) = I(x)P_2(x), \ldots, F_m(x) = I(x)P_m(x),$$

где $I(x)$ – информационный многочлен.

Для простоты предположим, что единице равен первый многочлен в (3. 1). Тогда $F_1(x) = I(x),$ а кодовое слово запишется в виде

,

где $s_{i,j}$ – коэффициенты в многочлене $F_i(x)$ при x^j, образующиеся в результате перемножения многочленов $I(x)$ и $P_i(x)$.

Очевидно, что в первых r кодовых кадрах из $n^0 = m$ символов в явном виде содержится по одному информационному символу $i_0, i_1, \ldots, i_{r-1}$, что соответствует систематическому виду кодирования с $R = 1/m$.

Практический интерес представляет систематическое кодирование с $R = k^0/m$. Поэтому обобщим лемму 3.1 для сверточных кодов с $R = k^0/m$.

Лемма 3.2. Для построения систематического сверточного кода с $R = k^0/m$ необходимо и достаточно выполнение равенства единице любых k^0

многочленов из выражения .

Доказательство аналогично лемме 3.1. Действительно, приравняв единице любые k^0 порождающих многочленов и записав соответствующее кодовое слово, получим, что в каждом из первых r кодовых кадров из $n^0 = m$ символов в явном виде содержатся информационные символы $i_0, i_1, \ldots, i_{k0}$, что соответствует систематическому виду кодирования с $R = k^0/m$.

Схема построенного таким образом алгебраического систематического рекурсивного сверточного кода с обработкой элементов из $GF(q)$ в общем виде представлена на рис. 3.11.

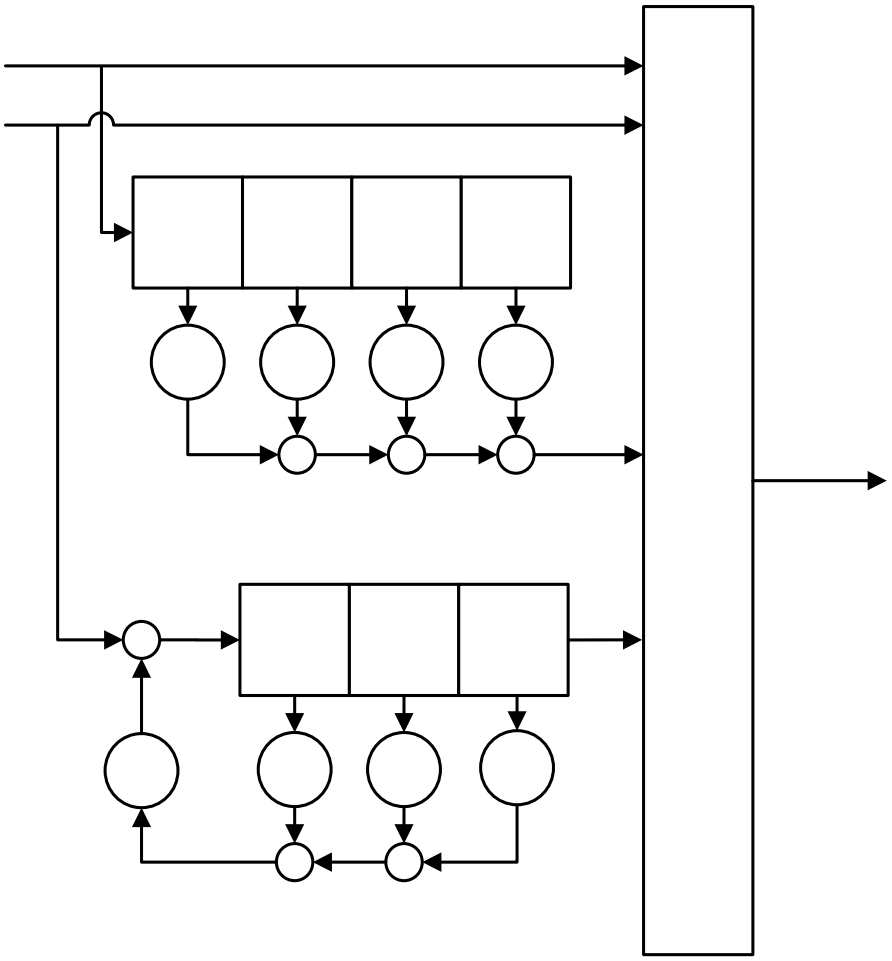
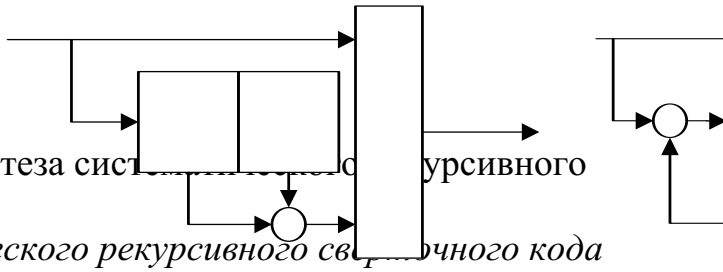


Рис. 3.11. Схема систематического кодера алгебраического рекурсивного сверточного кода с обработкой элементов из $GF(q)$



Рассмотрим еще один пример синтеза систематического рекурсивного сверточного кода.

Пример 3.3. Синтез систематического рекурсивного сверточного кода (вариант 2). Используем лемму 3.2 для построения систематического кодера алгебраического рекурсивного сверточного кода. Зафиксируем конечное поле $GF(2^2)$, построенное по кольцу многочленов

$$\{0 = \alpha^{-\infty}, 1 = \alpha^{0}, x = \alpha^{1}, x + 1 = \alpha^{2}\}$$

с операциями, по модулю

$$G(x) = x^2 + x + 1.$$

Зафиксируем (3, 2, 2) код РС с порождающим многочленом $g(x) = x + \alpha^2$. Тогда соответствующие порождающие многочлены несистематического нерекурсивного сверточного кода окажутся равными

$$g_1(x) = x + 1;$$

$$g_2(x) = 1.$$

Воспользовавшись результатом леммы 3.2 построим систематический нерекурсивный сверточный кодер, схема устройства представлена на рис. 3. 12, *а*.

Многочлен $g_1(x) = x + 1$ является делителем многочлена $x^3 + 1$. Мультипликативно обратным ему в кольце многочленов с операциями по модулю $x^3 + 1$ является многочлен $h_1(x) = x^2 + x + 1$. Тогда систематический рекурсивный сверточный код, построенный по порождающему многочлену (3, 2, 2) кода РС, имеет параметры

$$\nu = 2; k^0 = 1; n^0 = 2; k=3; n = 6; R = 1 / 2; d_\infty \geq 2.$$

Соответствующая схема кодера представлена на рис. 3.12, *б*.

а) *б)*

Рис. 3.12. Схемы сверточных кодеров:
нерекурсивного (*а*); рекурсивного (*б*)

Предложенный подход алгебраического построения систематических рекурсивных сверточных кодов значительно проще, чем метод, предложенный теоремами 3.1 – 3.4. Однако он предъявляет более жесткое требование – выполнение условий лемм 3.1. – 3.2, т.е. необходимо равенство единице любых k^0 многочленов из выражения (3.1).

3.3. Алгоритмы построения рекурсивных сверточных кодов и исследование свойств синтезированных кодовых конструкций

Разработанный метод построения рекурсивных сверточных кодов позволяет выразить конструктивные параметры алгебраически заданных сверточных кодов через соответствующие параметры циклических кодов (см. теоремы 3.3–3.7). Для практического использования полученных результатов разработаны алгоритмы алгебраического построения как систематических, так и несистематических рекурсивных сверточных кодов.

Для алгебраического построения рекурсивного сверточного кода с конструктивными *(n, k, d)* параметрами необходимо и достаточно задать порождающий и/или проверочный многочлен циклического *(N, K, D)* кода. При этом конструктивные параметры сверточного *(n, k, d)* кода будут аналитически связаны с параметрами циклического *(N, K, D)* кода и могут быть заданы выражениями (3.5–3.10). Алгоритм построения рекурсивных сверточных кодов в общем виде представим в виде последовательности шагов.

ШАГ 1. Ввод параметров рекурсивного сверточного *(n, k, d)* кода и мощности алфавита кодовых символов q.

ШАГ 2. Выбор варианта построения сверточного кода над *GF(q)*:

- несистематического рекурсивного сверточного *(n, k, d)* кода с $R = 1/m$ (см. теорему 3.5);
- несистематического рекурсивного сверточного *(n, k, d)* кода с $R = k^0/m$ (см. теорему 3.6);
- систематического рекурсивного сверточного *(n, k, d)* кода с $R = K/N$ (см. теорему 3.4);
- систематического рекурсивного сверточного *(n, k, d)* кода с $R = (K \cdot \log_q H)/((N-K) \cdot m + K \cdot \log_q H)$ (см. теорему 3.8).

ШАГ 3. Расчет параметров циклического *(N, K, D)* кода над $GF(q^m)$.

ШАГ 4. Выбор и формирование порождающего и/или проверочного многочлена циклического *(N, K, D)* кода над $GF(q^m)$.

ШАГ 5. Выбор способа обработки кодовых символов. Формирование порождающих многочленов рекурсивного сверточного *(n, k, d)* кода над *GF(q)*, построение схемы кодера рекурсивного сверточного *(n, k, d)* кода над *GF(q)*.

Разработанный алгоритм позволяет конструктивным способом за конечное число шагов построить рекурсивный сверточный код с требуемыми параметрами. Схема алгоритма представлена на рис. 3.13. После ввода параметров сверточного кода (шаг 1) и выбора варианта его построения (шаг 2) на третьем шаге алгоритма производится расчет параметров циклического *(N, K, D)* кода над $GF(q^m)$. Рассмотрим алгоритм подробно.

В случае, когда на втором шаге алгоритма выбран первый вариант построения сверточного кода, воспользуемся результатами теоремы 3.7. Зафиксируем конечное поле *GF(q)* и параметры несистематического рекурсивного сверточного *(n, k, d)* кода с $R = 1/m$ над *GF(q)*. По теореме 3.5

такой код однозначно задается проверочным многочленом *h(x)* циклического *(N, K, D)* кода над $GF(q^m)$.

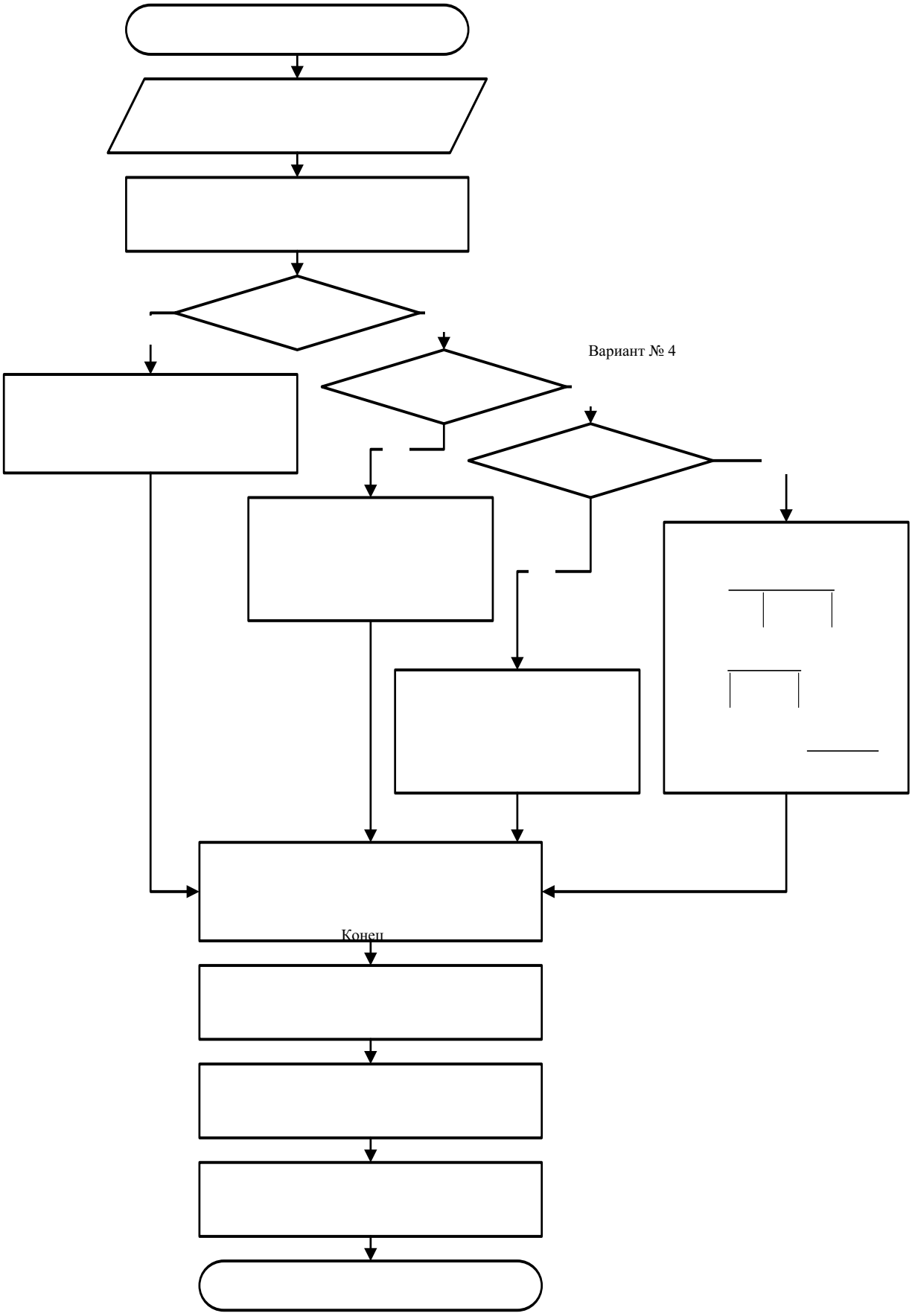


Рис. 3.13. Схема алгоритма построения рекурсивных сверточных кодов

Воспользовавшись выражением (3.5), выразим параметры циклического *(N, K, D)* кода над $GF(q^m)$ через фиксированные параметры несистематического рекурсивного сверточного *(n, k, d)* кода над *GF(q)*. Получим:

(3.11)

Если при этом требуется построить сверточный код с длиной кодового ограничения *v*, то необходимо использовать циклический *(N, K, D)* код над $GF(q^m)$ с $K = v$. На этом третий шаг алгоритма для первого варианта построения рекурсивного сверточного кода завершен.

Когда на втором шаге алгоритма выбран второй вариант построения сверточного кода, воспользуемся теоремой 3.6. Зафиксируем конечное поле *GF(q)* и параметры несистематического рекурсивного сверточного *(n, k, d)* кода с $R = k^0/m$ над *GF(q)*. По теореме 3.6 такой код однозначно задается проверочным многочленом *h(x)* циклического *(N, K, D)* кода над $GF(q^m)$. Используя выражение (3.8), выразим параметры циклического *(N, K, D)* кода над $GF(q^m)$ через фиксированные параметры несистематического рекурсивного сверточного *(n, k, d)* кода над *GF(q)*.

Получим

(3.12)

Если при этом требуется построить сверточный код с длиной кодового ограничения *v*, то необходимо использовать циклический *(N, K, D)* код над $GF(q^m)$ с $K = v/k^0$. На этом третий шаг алгоритма для второго варианта построения рекурсивного сверточного кода завершен.

Предположим, что на втором шаге алгоритма выбран третий вариант построения сверточного кода. Воспользуемся результатами теоремы 3.7. Зафиксируем конечное поле *GF(q)* и параметры систематического рекурсивного сверточного *(n, k, d)* кода над *GF(q)*. По теореме 3.7 такой код с $R = K/N$ однозначно задается порождающим многочленом *g(x)* циклического *(N, K, D)* кода над $GF(q^m)$. Воспользуемся выражением (3.8), выразим параметры циклического *(N, K, D)* кода над $GF(q^m)$ через фиксированные параметры систематического рекурсивного сверточного *(n, k, d)* кода над *GF(q)*. Получим

(3.13)

Если при этом требуется построить сверточный код с длиной кодового ограничения v, то необходимо использовать циклический (N, K, D) код над $GF(q^m)$ с $K = (n^0 – k^0)\cdot k^0$. На этом третий шаг алгоритма для третьего варианта построения рекурсивного сверточного кода завершен.

Предположим, что на втором шаге алгоритма выбран четвертый вариант построения сверточного кода. Воспользуемся результатами теоремы 3.8. Зафиксируем конечное поле $GF(q)$ и параметры систематического рекурсивного сверточного (n, k, d) кода над $GF(q)$. По теореме 3.8 такой код с

однозначно задается порождающим многочленом $g(x)$ циклического (N, K, D) кода над $GF(q^m)$. Воспользуемся выражением (3.8), выразим параметры циклического (N, K, D) кода над $GF(q^m)$ через фиксированные параметры систематического рекурсивного сверточного (n, k, d) кода над $GF(q)$. Получим:

(3.14)

На этом третий шаг алгоритма для четвертого варианта построения рекурсивного сверточного кода завершен.

Рассмотрим особенности выполнения четвертого шага разработанного алгоритма. На этом шаге алгоритма производится выбор схемы кодирования циклического кода (через порождающий или проверочный многочлен), что определяет так же схему кодирования рекурсивного сверточного кода.

Предположим, что в качестве циклического кода выбран примитивный код БЧХ, его длина равна $N = (q^m)^M – 1$. Рассмотрим поле разложения двучлена $(x^M – 1)$ на минимальные многочлены элементов поля $GF((q^m)^M)$ над $GF(q^m)$. Порождающий многочлен примитивного кода БЧХ задается в виде

$$g(x) = НОК(f_1, f_2, \ldots, f_{2t}),$$

где $D = 2t + 1$, f_i – минимальный многочлен над $GF(q^m)$ элементов $\alpha^i \in GF((q^m)^M)$.

Проверочный многочлен $h(x)$ определим как сомножитель $g(x)$ в разложении двучлена $x^N - 1$:

$$h(x) = (x^N - 1)/g(x).$$

Последнее выражение эквивалентно следующему:

$$h(x) = НОК(\varphi_1, \varphi_2, \ldots),$$

где φ_j – минимальный многочлен над $GF(q^m)$ элементов $\alpha^j \in GF((q^m)^M)$, причем $\alpha^i \neq \alpha^j$.

Рассмотрим случай, когда в качестве циклического кода выбран непримитивный код БЧХ. По определению, длина непримитивного кода БЧХ равна одному из сомножителей в разложении числа $(q^m)^M - 1$ (если, конечно, число $(q^m)^M - 1$ не является простым), т.е. $N = ((q^m)^M - 1)/g$ для произвольного целого g, делящего нацело число $(q^m)^M - 1$. Очевидно, что должно выполняться также условие $r < N$.

Порождающий многочлен непримитивного кода БЧХ задается в виде

$$g(x) = НОК(f_1, f_2, \ldots, f_{2t}),$$

где $D = 2t + 1$, f_i – минимальные многочлены над $GF(q^m)$ элементов $\beta^i \in GF((q^m)^M)$ такие, что их порядок равен N, т.е. $\beta^i = \alpha^{jg}$, $j = 1, 2, \ldots, M/2$.

Проверочный многочлен определяется аналогично случаю, рассмотренному выше.

Рассмотрим случай, когда в качестве циклического кода выбран код РС . По определению, порождающий многочлен кода РС задается в виде

$$g(x) = (x - \alpha^i) \cdot (x - \alpha^{2i}) \cdot \ldots \cdot (x - \alpha^{2ti}),$$

где $D = 2t + 1$; $\alpha^i \in GF(q^m)$.

Аналогично рассмотренному выше случаю формируется проверочный многочлен $h(x)$.

На шестом шаге алгоритма выбирается способ обработки кодовых символов: по одному элементу из $GF(q)$ или пакетами по m элементов из $GF(q)$ – по одному символу из $GF(q^m)$. В соответствии с выбранным способом обработки формируются порождающие многочлены рекурсивного сверточного кода.

Таким образом, разработанный алгоритм позволяет алгебраически строить систематические и несистематические рекурсивные сверточные коды.

Проведем оценку параметров алгебраически заданных рекурсивных сверточных кодов. Для чего зафиксируем конечное поле $GF(2^2)$ и рассмотрим коды РС с параметрами $N = 2^2 - 1 = 3$; $3 - K = D - 1$.

Воспользуемся результатами теорем 3.2. – 3.3. Исследуем конструктивные кодовые параметры алгебраически заданных несистематических рекурсивных сверточных кодов. В табл. 3.3 представлены параметры кодов РС над $GF(2^2)$, конструктивные параметры несистематических рекурсивных сверточных (n, k, d) кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное и истинное значение свободного кодового расстояния. Случаи для кодов $(n, 1, n)$ соответствуют тривиальному коду с повтором символов.

Таблица 3.3

Конструктивные характеристики двоичных несистематических рекурсивных сверточных кодов, заданных через порождающий многочлен кода РС над *GF* (2^2)

(N, K, D)	*(n, k, d)*	*v*	*R*	$d_П$	d_∞
(3, 1, 3)	(4, 2, 3)	1	1 / 2	2	3
(3, 2, 2)	(6, 3, 3)	2	1 / 2	3	4

Воспользуемся результатами теорем 3.4. – 3.5. Исследуем конструктивные кодовые параметры алгебраически заданных систематических рекурсивных сверточных кодов.

В табл. 3.4 представлены параметры кодов РС над $GF(2^2)$, конструктивные параметры систематических рекурсивных сверточных *(n, k, d)* кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное и истинное значение свободного кодового расстояния.

Зафиксируем конечное поле $GF(2^3)$ и рассмотрим коды РС с параметрами

$$N = 2^3 - 1 = 7;\ 7 - K = D - 1.$$

В табл. 3.5 представлены параметры кодов РС над $GF(2^3)$, конструктивные параметры несистематических рекурсивных сверточных *(n, k, d)* кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное и истинное значение свободного кодового расстояния.

Таблица 3.4

Конструктивные характеристики двоичных систематических рекурсивных сверточных кодов, заданных через порождающий многочлен кода РС над *GF* (2^2)

(N, K, D)	*(n, k, d)*	*v*	*R*	$d_П$	d_∞
(3, 1, 3)	(15, 3, 3)	2	1 / 5	6	7
	(18, 6, 3)	4	1 / 3	5	6
(3, 2, 2)	(8, 4, 2)	2	1 / 2	4	5
	(12, 8, 2)	4	2 / 3	3	4

Таблица 3.5

Конструктивные характеристики двоичных несистематических рекурсивных сверточных кодов, заданных через порождающий многочлен кода РС над *GF* (2^3)

(N, K, D)	*(n, k, d)*	*v*	*R*	$d_П$	d_∞
1	2	3	4	5	6
(7, 1, 7)	(6, 2, 7)	1	1 / 3	7	8
	(6, 4, 7)	2	2 / 3	6	7

(7, 2, 6)	(9, 3, 6)	2	1 / 3	5	6
	(9, 6, 6)	4	2 / 3	5	6
(7, 3, 5)	(12, 4, 5)	3	1 / 3	5	6
	(12, 8, 5)	6	2 / 3	4	5
(7, 4, 4)	(15, 5, 4)	4	1 / 3	4	6
	(15, 10, 4)	8	2 / 3	3	4
(7, 5, 3)	(18, 6, 3)	5	1 / 3	3	3
	(18, 12, 3)	10	2 / 3	2	3

Продолжение табл. 3.5

1	2	3	4	5	6
(7, 6, 2)	(21, 7, 2)	6	1 / 3	3	4
	(21, 14, 2)	12	2 / 3	2	3

В табл. 3.6 представлены параметры кодов РС над $GF(2^3)$, конструктивные параметры систематических рекурсивных сверточных (n, k, d) кодов, алгебраически заданных порождающим многочленом кода РС, предсказанное и истинное значение свободного кодового расстояния.

Таблица 3.6

Конструктивные характеристики двоичных рекурсивных сверточных кодов, заданных через порождающий многочлен кода РС над $GF(2^3)$

(N, K, D)	(n, k, d)	v	R	$d_П$	d_∞
1	2	3	4	5	6
(7, 1, 7)	(133, 7, 7)	6	1 / 19	20	22
	(140, 14, 7)	12	1 / 10	19	20
	(147, 21, 7)	18	1 / 7	17	18
(7, 2, 6)	(102, 12, 6)	10	2 / 17	15	16
	(114, 24, 6)	20	4 / 19	14	15
	(126, 36, 6)	30	2 / 7	13	14
(7, 3, 5)	(75, 15, 5)	12	1 / 5	12	13
	(90, 30, 5)	24	1 / 3	11	12
	(105, 45, 5)	36	3 / 7	10	11
(7, 4, 4)	(52, 16, 4)	12	4 / 13	9	10
	(68, 32, 4)	24	8 / 17	8	9
	(84, 48, 4)	36	4 / 7	7	8

Продолжение табл. 3.6

1	2	3	4	5	6
(7, 5, 3)	(33, 15, 3)	10	5 / 11	6	5
	(48, 30, 3)	20	6 / 8	4	3

	(63, 45, 3)	30	5 / 7	3	3
(7, 6, 2)	(18, 12, 2)	6	2 / 3	3	4
	(30, 24, 2)	12	4 / 5	4	5
	(42, 36, 2)	18	6 / 7	3	4

Полученные параметры кодов свидетельствуют о том, что для построения хороших несистематических рекурсивных сверточных кодов следует использовать низкоскоростные циклические коды. При этом удается получить практически весь спектр скоростей кодирования сверточных кодов и даже при небольшом кодовом ограничении хорошие кодовые параметры. А для построения хороших систематических рекурсивных сверточных кодов в следует использовать высокоскоростные циклические коды. При этом также удается получить практически весь спектр значений скоростей кодирования сверточных кодов и хорошие кодовые параметры даже при небольшом кодовом ограничении.

Практическое применение разработанных методов синтеза алгебраически заданных рекурсивных сверточных кодов позволяет решить важную решить важную научную задачу поиска эффективных сверточных кодов с большой длиной кодового ограничения. Актуальным направлением дальнейших исследований является разработка методов и алгоритмов декодирования алгебраически заданных рекурсивных сверточных кодов, позволяющих реализовать потенциальные возможности синтезируемых кодовых конструкций.

Выводы

1. В качестве составляющих турбокод кодов необходимо использовать рекурсивные сверточные коды, что связано с особенностями весового распределения кодовых слов рекурсивных сверточных кодов. Поэтому на основе единого общетеоретического подхода, с использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов получили дальнейшее развитие алгебраические методы и вычислительно эффективные алгоритмы синтеза рекурсивных сверточных кодов.

2. Разработанные вычислительно эффективные (вычислительно реализуемые) алгебраические методы синтеза (алгебраически заданных) рекурсивных сверточных кодов, отличаются от известных использованием ограничения недвоичного циклического кода на произвольное подполе, что позволяет синтезировать (алгебраически заданные) рекурсивные сверточные коды с произвольными свойствами и кодовыми характеристиками.

3. Теоретически обоснованы процедуры алгебраического построения рекурсивных сверточных кодов через обобщение циклических кодов на случай бесконечной длины. Доказанные теоремы позволяют аналитически

связать параметры несистематических циклических кодов с конструктивными параметрами соответствующих сверточных кодов. Предложенные в работе алгебраические процедуры построения рекурсивных сверточных кодов позволяют за конечное число шагов однозначно определить правило сверточного кодирования и построить схему рекурсивного сверточного кодера с заданными конструктивными характеристиками.

4. Получили дальнейшее развитие методы кодирования алгебраически заданными рекурсивными сверточными кодами, отличающиеся от известных теоретически обоснованными процедурами алгебраического построения рекурсивных сверточных кодов через обобщение циклических кодов на случай бесконечной длины, что позволяет аналитически формализовать процесс помехоустойчивого кодирования синтезируемыми сверточными кодами с высокими (конструктивными) кодовыми характеристиками.

5. Проведенные исследования свойств синтезированных рекурсивных сверточных кодов, алгебраически заданных порождающими многочленами недвоичных циклических кодов, показали, что полученные коды близки по своим характеристикам к оптимальным кодам.

6. Проведенные исследования свойств алгебраически заданных рекурсивных сверточных кодов показали, что для построения эффективных несистематических кодов следует использовать низкоскоростные циклические коды, а для построения эффективных систематических кодов необходимо применять высокоскоростные циклические коды. Это позволяет получить практически весь спектр значений скоростей сверточного кодирования и высокие конструктивные параметры даже при небольшом кодовом ограничении.

7. Применение разработанных методов синтеза рекурсивных сверточных кодов позволяет за счет использования алгебраических процедур и полиномиальных методов описания циклических кодов решить важную научную задачу поиска эффективных рекурсивных сверточных кодов с большой длиной кодового ограничения, которые в дальнейшем будут использоваться для построения турбокодов.

РАЗДЕЛ 4
МЕТОДЫ И АЛГОРИТМЫ ДЕКОДИРОВАНИЯ АЛГЕБРАИЧЕСКИ ЗАДАННЫХ СВЕРТОЧНЫХ КОДОВ

Разработанные методы и алгоритмы алгебраического построения сверточных кодов позволяют синтезировать сверточные кодовые конструкции, близкие по своим свойствам к оптимальным сверточным кодам . Предложенный концептуальный подход позволяет решить важную научную задачу поиска эффективных сверточных кодов с большой длиной кодового ограничения. В тоже время, задача повышения достоверности передаваемой информации на основе синтезированных сверточных кодовых конструкций связана с разработкой вычислительно эффективных методов и алгоритмов декодирования алгебраически заданных сверточных кодов.

В данном разделе разрабатываются методы декодирования алгебраически заданных сверточных кодов, основанные на использовании бесконечной серии синдромов кодовых слов циклического кода. Предлагается способ формирования бесконечной серии синдромов алгебраически заданного сверточного кода. Разрабатывается подход комбинированного декодирования алгебраически заданных сверточных кодов, состоящий в совмещении алгебраических процедур и процедур последовательного поиска по кодовой решетке. Установлено, что применение предложенных процедур позволяет локализовать ошибки в кодовом слове алгебраически заданного сверточного кода и ускорить последовательный поиск по кодовой решетке при комбинированном методе декодирования.

4.1. Методы декодирования сверточных кодов и их вычислительная эффективность

В соответствии с общими положениями теории помехоустойчивого кодирования развитие методов декодирования сверточных кодов происходило в трех направлениях [13-25, 38-42]: методы и алгоритмы порогового декодирования; методы и алгоритмы декодирования по максимуму правдоподобия; методы и алгоритмы последовательного декодирования (см. рис. 4.1). Рассмотрим указанные методы, исследуем их возможности по реализации процедуры декодирования алгебраически заданных сверточных кодов.

Методы порогового декодирования аналогичны мажоритарным декодерам блоковых кодов. Их достоинством является простота алгоритмов и практической реализации. Число операций, необходимое для декодирования одного информационного символа не превосходит некоторой

постоянной величины.

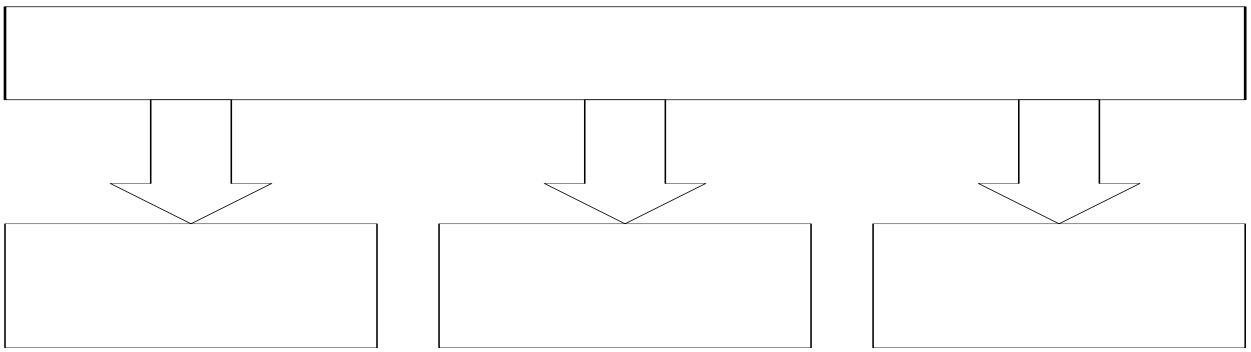


Рис. 4.1. Классификация методов декодирования сверточных кодов

Метод декодирования по максимуму правдоподобия теоретически более эффективен, с точки зрения реализации исправляющей способности кода. Однако сложность алгоритмов и практических устройств, необходимых для его реализации, растет экспоненциально с ростом длины кода [19, 20].

В основе методов последовательного декодирования лежит вероятностный подход, при котором число операция, необходимых для декодирования одного символа, является случайной величиной.

Рассмотрим различные подходы к декодированию сверточных кодов, исследуем их особенности и возможность применения к декодированию алгебраически заданных непрерывных кодов.

4.1.1. Методы порогового декодирования

Пороговое декодирование сверточных кодов [23] основано на тех же принципах, что и мажоритарное декодирование блоковых кодов.

Этот подход является одним из наиболее простых и удобных в реализации методов декодирования. Практическая реализация таких декодеров привлекательна высоким быстродействием и низкой сложностью.

Суть мажоритарного декодирования состоит в мажоритарном оценивании веса ошибки, произошедшей в l-ом символе кодового слова. Подобная оценка проводится над ортогональными относительно l-ого кодового символа проверочными уравнениями. Ортогональное относительно l-ой координаты подмножество проверочных уравнений состоит из всех уравнений, в каждое из которых входит l-я компонента, а остальные компоненты входят не более чем в одно уравнение. Если структура кода такова, что множество проверочных уравнений ортогонально относительно нескольких координат, то мажоритарное решение применимо для локализации ошибки в подмножестве этих компонент. Затем, для нахождения ошибки опять используется мажоритарная логика. Подобный многошаговый мажоритарный декодер позволяет декодировать большее число ошибок, но, тем не менее, не всегда позволяет декодировать все ошибки по минимальному расстоянию. Максимальное число ошибок, которые правит одношаговый мажоритарный декодер, определяется

следующей теоремой.

Теорема 4.1. Число ошибок, которые правит одношаговый мажоритарный декодер для произвольного кода над *GF(q)* не превосходит *(n - 1)/(2d⊥ - 2)*, где *d⊥* - минимальное кодовое расстояние дуального кода.

Доказательство теоремы основано на том факте, что проверочные уравнения мажоритарного декодирования алгебраического кода соответствуют кодовым словам дуального кода. Число ортогональных относительно любой из координат проверок, не превышает *(n - 1)/(2t)*. Известно, что если для каждой координаты мажоритарного декодера имеется *l* проверок, ортогональных относительно ее, то декодер правит *l/2* ошибок. Следовательно, число ошибок, которые правит одношаговый мажоритарный декодер, не превосходит *(n - 1)/(2d⊥ - 2)*.

Несколько большее число ошибок позволяет править многошаговый декодер. Справедлива теорема.

Теорема 4.2. Число ошибок, которые правит многошаговый мажоритарный декодер для произвольного кода над *GF(q)* не превосходит *n/d⊥ - 0,5*. (Доказательство аналогично предыдущему).

При пороговом декодировании сверточных кодов используется аналогичный принцип мажоритарного принятия решения. При этом ошибки в бесконечном кодовом слове последовательно исправляются сначала в первом принятом блоке длины n^0 кодовых символов, затем во втором и т.д. Воздействие найденных ошибок может быть снято и соответствующий синдром откорректирован с помощью цепи обратной связи. Если при этом ошибка декодирована не правильно, синдром будет откорректирован не верно, что приведет к последующему неверному декодированию и будет наблюдаться эффект распространения ошибок. Если схема декодера не предусматривает коррекцию символов синдрома, такое декодирование называют дефинитным. При этом корректирующие способности кода ухудшаются, однако благодаря отсутствию цепи обратной связи возникающий эффект распространения ошибок ограничен конечной глубиной.

Корректирующие способности порогового декодирования определяются исправляющей способностью на одном блоке кодовых символов. Следовательно, исследование пороговых методов декодирования можно ограничить изучением структуры синдромов и проверочной матрицы кода на декодирование первого блока кодовых символов.

Среди кодов, допускающих пороговое декодирование, самыми простыми являются самоортогональные коды. Это коды, допускающие полную ортогонализацию, т.е. коды, которые на блоке из n^0 символов могут быть мажоритарно декодированы. Метод построения таких кодов через использование совершенных разностных множеств изложен в [18], скорость кодирования таких кодов равна $R = (n^0 - 1)/n^0$ и $R = 1/n^0$. К сожалению, иных конструктивных способов построения самоортогональных кодов не известно.

Другим примером сверточных кодов, допускающих пороговое декодирование, являются ортогонализируемые коды – это коды, которые

допускают построение для каждого кодового символа составных проверок из линейной комбинации символов синдромов. По сравнению с самоортогональными эти коды могут иметь меньшее кодовое ограничение и, в этом смысле, имеют более высокие конструктивные параметры. Однако ортогонализируемые коды строятся методом перебора и могут иметь бесконечную глубину распространения ошибок.

4.1.2. Декодирование по максимуму правдоподобия

В работах [19, 20 , 22] показано, что верхние границы вероятности ошибки при декодировании сверточных кодов по максимуму правдоподобия лучше, чем у блоковых кодов той же длины. При этом используется алгоритм декодирования Витерби, который, как показано в [22], реализует декодирование по максимуму правдоподобия.

Декодер Витерби итеративно обрабатывает кадр за кадром и, двигаясь по решетке, пытается повторить путь кодера. Решеткой называется граф, узлы которого находятся в прямоугольной координатной сетке, полубесконечной справа, число узлов в каждом столбце конечно. Конфигурация ребер, соединяющих узлы каждого столбца с узлами столбца справа, одинакова для всех столбцов. Типичная решетка, для двоичного кодового алфавита приведена на рис. 4.2.

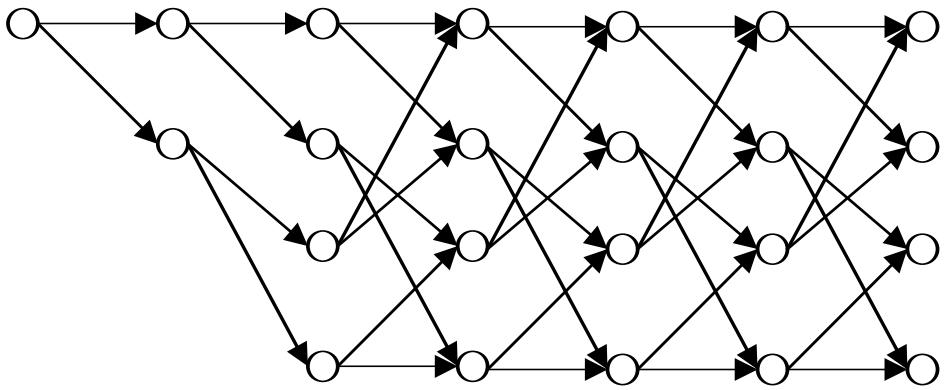


Рис. 4.2. Решетка сверточного кода

Узлы в каждом столбце представляют q^{v} состояний, в которых может находиться регистр сдвига. Каждый следующий столбец представляет собой набор состояний в следующий момент времени. Поступление на вход нового кадра приводит к изменению состояния регистра, соответствующего ребру, которое ведет к следующему узлу.

В любой момент времени декодер не знает, в каком узле находится кодер и поэтому не пытается декодировать этот узел. Вместо этого декодер по принятой последовательности определяет наиболее правдоподобный путь к каждому узлу и определяет расстояние между каждым таким путем и принятой последовательностью. Это расстояние называют мерой расходимости пути. Если все пути во множестве наиболее правдоподобных

начинаются одинаково, то декодер, как правило, знает начало пути, пройденного кодером.

В следующем кадре декодер определяет наиболее правдоподобный путь к каждому из новых узлов этого кадра. Наиболее правдоподобный путь находится прибавлением приращения меры расходимости на продолжениях старых путей к мере расходимости путей, ведущих в старый узел. В каждый новый узел ведет q^v путей, и путь с наименьшей мерой расходимости является наиболее правдоподобным путем. Этот процесс повторяется для каждого из новых узлов. В конце итерации декодер знает наиболее правдоподобный путь к каждому из узлов в новом кадре. Если все выжившие пути проходят через один и тот же узел в первом временном кадре, то вне зависимости от того в каком узле кодер находится в другом временном кадре , становится известным наиболее правдоподобный первый временной кадр. Иначе говоря, принимается решение о первом временном кадре.

Для построения декодера Витерби необходимо выбрать ширину окна декодирования, которая обычно в несколько раз превосходит длину кодового блока. На рис. 4.3. представлено окно декодирования Витерби с изображенной частью кодовой решетки и выжившими путями.

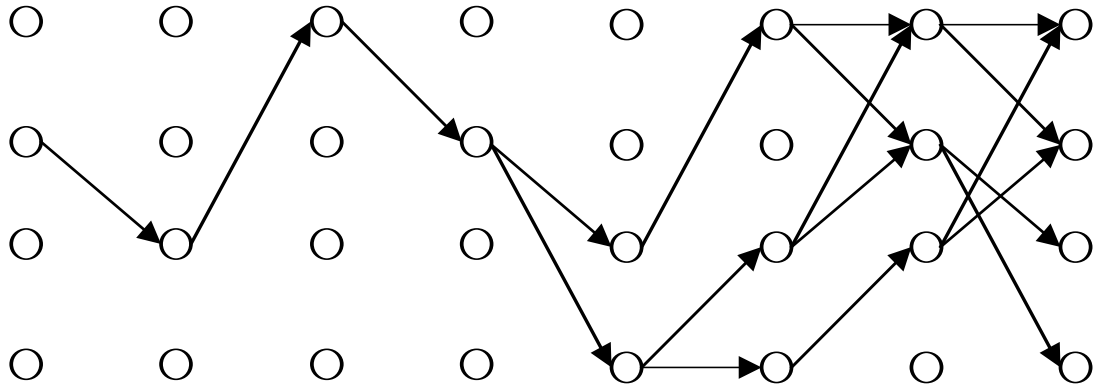


Рис. 4.3. Окно декодирования Витерби

По мере продвижения декодера к последующим кадрам из его памяти выводятся ранние кадры. Если в самом старом кадре существует лишь один узел, через который проходит путь, то декодирование является полным. В случае, когда узлов несколько, декодер неполный. Избежать этого можно путем увеличения ширины окна декодирования. К сожалению, сложность декодера Витерби быстро растет. Действительно, для сверточного кода с длиной кодового ограничения v необходимо хранить в q^v путей, что для больших v становиться совершенно непригодно.

4.1.3. Методы последовательного декодирования

Для того чтобы ослабить влияние больших v на сложность декодирования сверточных кодов в [18, 24, 27] исследованы последовательные процедуры поиска по дереву. Общая стратегия

последовательного декодирования состоит в игнорировании маловероятных путей по решетке. В отличие от оптимальной процедуры Витерби последовательный декодер просматривает только один узел и по определенному правилу принимает решение о продвижении вперед на одно ребро графа. Если принято неправильное решение, то декодер возвращается и начинает последовательный поиск ребер заново. Наиболее популярный алгоритм последовательного декодирования является декодер Фано [18].

Исходными данными для алгоритма Фано является вероятность P_0 появления ошибочного символа в канале (или, по крайней мере, верхняя граница для P_0). Если декодер следует по правильному пути, то вероятное число ошибок в первых l кадрах $\approx P_0 n^0 l$. Выберем параметр P^* так, что $P_0 < P^* < 1/2$ и введем следующий показатель:

$$t(l) = P^* n^0 l - d(l),$$

где $d(l)$ – расстояние Хемминга между принятым словом и текущим путем по решетке.

Для правильного пути $d(l) \approx P_0 n^0 l$, следовательно, $t(l)$ возрастает. Пока $t(l)$ возрастает, декодер следует по правильному пути. Если $t(l)$ начинает уменьшаться, то декодер принимает решение об ошибочности выбранного пути, возвращается назад и начинает последовательно двигаться по решетке далее. Иногда последовательный декодер выполняет так много вычислений, что величина входного буфера становится недостаточной. Это явление называется переполнением буфера и является существенным ограничением для применения алгоритма Фано. Вероятность переполнения буфера с ростом его размера уменьшается очень медленно. Наиболее надежным способом управления переполнением буфера является периодическая подача в кодер заранее известной последовательности (например, нулей) с длиной, равной длине кодового ограничения. Если буфер переполнился, то декодер считает декодирование неудавшимся, ждет соответствующего момента и снова начинает декодирование. Все данные, поступившие между моментом переполнения буфера и следующим включением, теряются. Такой подход несколько уменьшает скорость кода и ставит при конструировании декодеров задачу синхронизации по времени.

Таким образом, существующие методы декодирования сверточных кодов позволяют эффективно, в некоторых случаях, решать задачу исправления ошибок в бесконечном кодовом слове непрерывного кода. Однако им присущи соответствующие недостатки:

- для применения порогового декодирования сверточный код должен обладать дополнительной алгебраической структурой (самоортогональность или ортогонализуемость);
- сложность реализации декодера по максимуму правдоподобия растет экспоненциально от длины кодового ограничения v, что совершенно непригодно для больших v;
- переполнение буфера является существенным ограничением для использования последовательных декодеров, алгоритма Фано в том числе.

В тоже время, как показано ниже, алгебраические сверточные коды, заданные через порождающий многочлен циклического кода, обладают дополнительными алгебраическими свойствами, что существенно упрощает их декодирование.

4.2. Алгебраические методы декодирования алгебраически заданных сверточных кодов

Сверточный код по определению состоит из бесконечного числа бесконечно длинных кодовых слов. Он линеен, следовательно, может быть задан бесконечной порождающей матрицей [13, 19-21].

Предположим, что несистематический нерекурсивный сверточный код над $GF(q)$ задан порождающими многочленами вида

,

,

…

,

где коэффициенты при x являются элементами $GF(q)$, $n^0 = m$.

Тогда соответствующая полубесконечная порождающая матрица запишется в виде [13, 19-21]:

, (4.1)

где G_i – матрица – строка, состоящая из коэффициентов порождающих многочленов сверточного кода при x^i, т.е.

$$G_i = (p_{1,i}, p_{2,i}, \ldots, p_{m,i}). \quad (4.2)$$

Символом 0 в (4.1) обозначена матрица – строка, состоящая из n^0 нулевых символов из $GF(q)$.

В случае систематического сверточного кода

$$G_0 = (1, p_{2,0}, \ldots, p_{m,0}) = (1, P_0)$$

и

$$G_i = (0, p_{2,i}, \ldots, p_{m,i}) = (0, P_i).$$

Тогда матрица (4.1) перепишется в виде

.

Соответствующую полубесконечную проверочную матрицу запишем в виде [13, 19-21]:

.

Воспользуемся введенным выше алгебраическим описанием нерекурсивных сверочных кодов. Сопоставим каждую подматрицу G_i элементу поля $\beta_i \in GF(q^m)$, так, например, что

$$\beta_i = p_{1,i} + p_{2,i}x + \ldots + p_{m,i}x^m.$$

Тогда (4.2) перепишем в виде

$$G_i = (p_{1,i}, p_{2,i}, \ldots, p_{m,i}) = \beta_i,$$

а полубесконечную матрицу (4.1) представим в виде соответствующей матрицы с элементами из $GF(q^m)$:

. (4.3)

В полиномиально-матричном представлении последнее выражение перепишем в виде матрицы многочленов $G(x)$:

,

где по теореме 3.1 многочлен

суть порождающий многочлен недвоичного *(N, K, D)* циклического кода над $GF(q^m)$, который однозначно задает *(n, k)* несистематический сверточный код над $GF(q)$ с параметрами: $k^0 = 1$, $n^0 = m$, $\nu = r \cdot k^0 = r$, $k = r + 1$, $n = (r + 1) \cdot n^0 = k \cdot m$, $R = 1 / m$, $d_\infty \geq D$, $C(x) = I(x) \cdot P(x)$.

Тогда подматрица

(4.4)

суть порождающая матрица *(N, K, D)* циклического кода над $GF(q^m)$. В полиномиально-матричном представлении запишем в виде

.

Введенное обобщение *(N, K, D)* циклического кода на непрерывный случай позволяет использовать свойства колец многочленов при описании соответствующих сверточных кодов.

Пусть $I(x) = i_0 + i_1x + i_2x^2 + \ldots$ - информационный многочлен, возможно бесконечной длины, с коэффициентами из $GF(q)$. Предположим что $I(x)$ поступает на вход несистематического кодера нерекурсивного сверточного *(n, k, d)* кода алгебраически заданного через порождающий многочлен $P(x)$ циклического *(N, K, D)* кода над $GF(q^m)$. Тогда кодовое слово сверточного кода суть обобщение на непрерывный случай кодового слова циклического кода ограниченного на подполе $GF(q)$. Кодовая последовательность на выходе сверточного кодера будет задаваться выражением:

$$C(x) = I(x) \cdot P(x) = C_0 + C_1x + C_2x^2 + \ldots,$$

где C_i – элементы поля $GF(q^m)$, отображаемые в наборы по m символов из подполя $GF(q)$.

В матричной форме последнее выражение примет вид:

$$C = I \cdot G, \qquad (4.5)$$

где $C = (C_0, C_1, C_2, \ldots)$, $I = (i_0, i_1, i_2, \ldots)$ – кодовый и информационный векторы, составленные из коэффициентов соответствующих многочленов.

Рассмотрим правило формирования коэффициентов кодового многочлена (4.5), аналитически свяжем значение каждого кодового символа с информационными символами, поступающими на вход кодера.

Разобьем информационный вектор I на блоки по K символов из $GF(q)$:
$I = (i_0, i_1, i_2, \ldots, i_{K-1}) \cup (i_K, i_{K+1}, i_{K+2}, \ldots, i_{2K-1}) \cup (i_{2K}, i_{2K+1}, i_{2K+2}, \ldots, i_{3K-1}) \cup \ldots$
Обозначим каждый блок из K символов через I_i:

$$I = I_0 \cup I_1 \cup I_2 \cup \ldots.$$

В полиномиальном виде последнее выражение эквивалентно следующему:

$$I(x) = I_0(x) + x^K I_1(x) + x^{2K} I_2(x) + \ldots, \qquad (4.6)$$

где

$$I_i(x) = i_{i \cdot K} + i_{i \cdot K+1} x + i_{i \cdot K+2} x^2 + \ldots i_{(i+1) \cdot K-1} x^{K-1}.$$

Подставим (4.6) в (4.5), получим:

или в матричном виде

, (4.7)

где - единичная матрица с добавленными слева $i \cdot K$ нулевыми столбцами.

Проанализируем полученное выражение (4.7). Каждое слагаемое содержит произведение порождающего многочлена циклического (N, K, D) кода на информационный многочлен $I_i(x)$ степени $deg\, I_i(x) \le K - 1$. Однако, произведение $I_i(x) \cdot P(x)$ – суть кодовое слово циклического (N, K, D) кода, которое соответствует информационному вектору

$$I_i = (i_{i \cdot K}, i_{i \cdot K+1}, i_{i \cdot K+2}, \ldots, i_{(i+1) \cdot K-1}),$$

т.е.

$$I_i(x) \cdot P(x) = c_i(x), \qquad (4.8)$$

где

$$c_i(x) = c_{i,0} + c_{i,1} x + c_{i,2} x^2 + \ldots c_{i,N-1} x^{N-1}.$$

Подставим (4.8) в (4.7), получим:

или, в матричном виде

. (4.9)

Таким образом, как следует из выражения (4.9), бесконечное кодовое слово нерекурсивного сверточного кода алгебраически заданного через порождающий многочлен циклического кода состоит из бесконечной суммы кодовых слов циклического кода, умноженных на соответствующий оператор задержки $x^{i \cdot K}$. Представим, для наглядности, структуру бесконечного кодового слова алгебраического сверточного кода на рис. 4.4.

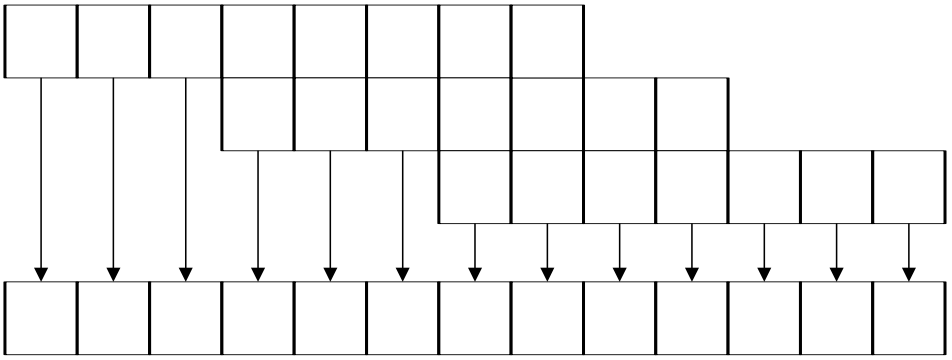


Рис. 4.4. Структура бесконечного кодового слова алгебраического нерекурсивного сверточного кода

Как видно на рис. 4.4 бесконечное кодовое слово сверточного кода формируется наложением бесконечного числа кодовых слов циклического кода и суммированием соответствующих элементов $c_{i,j}$.

Предположим теперь, что при передаче бесконечной кодовой последовательности вектор $C = (C_0, C_1, \ldots)$ исказился, т.е. на приемной стороне получено искаженное кодовое слово

$$C^*(x) = C(x) + E(x), \qquad (4.10)$$

где $E(x) = e_0 + e_1 x + e_2 x^2 + \ldots$ – бесконечный вектор ошибок.

По аналогии с информационным вектором разобьем вектор ошибок E =

$=(e_0, e_1, e_2, \ldots)$, составленный из коэффициентов многочлена ошибок $E(x)$, на блоки по K символов из $GF(q)$:

$$E = (e_0, e_1, e_2, \ldots, e_{K-1}) \cup (e_K, e_{K+1}, e_{K+2}, \ldots, e_{2K-1}) \cup \ldots.$$

Обозначим каждый блок из K символов через E_i:

$$E = E_0 \cup E_1 \cup E_2 \cup \ldots.$$

В полиномиальном виде последнее выражение эквивалентно следующему:

$$E(x) = E_0(x) + x^K E_1(x) + x^{2K} E_2(x) + \ldots, \qquad (4.11)$$

где

$$E_i(x) = e_{i \cdot K} + e_{i \cdot K+1} x + e_{i \cdot K+2} x^2 + \ldots e_{(i+1) \cdot K-1} x^{K-1}.$$

Подставим (4.11) в (4.10), получим:

С учетом (4.9) последнее выражение перепишем в виде:

или матричной форме

, (4.12)

где - вектор ошибок E_i длины K символов с добавленными справа $(N - K)$ нулями.

Проанализируем полученное выражение. Каждое слагаемое содержит сумму кодового слова $c_i(x)$ циклического (N, K, D) кода и многочлена ошибки $E_i(x)$. Размерность вектора E_i составляет K символов, т.е. сумма $c_i(x)+ E_i(x)$ – суть кодовое слово циклического (N, K, D) кода, искаженное вектором ошибки E_i. Следовательно, запишем:

$$c_i^*(x) = c_i(x)+ E_i(x). \quad (4.13)$$

Тогда, с учетом (4.13), выражение (4.12) перепишем в виде:

,

или, соответственно,

, (4.14)

Таким образом, как следует из выражения (4.14), бесконечное кодовое слово алгебраического нерекурсивного сверточного кода искаженное бесконечным вектором ошибок состоит из бесконечной суммы кодовых слов циклического кода, искаженных вектором ошибок конечной размерности, умноженных на соответствующий оператор задержки $x^{i \cdot K}$.

Представим, для наглядности, структуру искаженного ошибками бесконечного кодового слова алгебраического сверточного кода на рис. 4.5.

Как видно на рис. 4.5 искаженное ошибками бесконечное кодовое слово сверточного кода формируется наложением бесконечного числа искаженных кодовых слов циклического кода и суммированием соответствующих элементов $c^*_{i,j}$.

Введем синдромный многочлен алгебраического сверточного кода:

$$S(x) = s_0 + s_1x + s_2x^2 + \ldots, \quad (4.15)$$

как бесконечную сумму синдромных многочленов циклического кода, умноженных на соответствующий оператор задержки $x^{i \cdot K}$, т.е. как бесконечную сумму остатков от деления кодовых слов циклического кода на порождающий многочлен $P(x)$:

. (4.16)

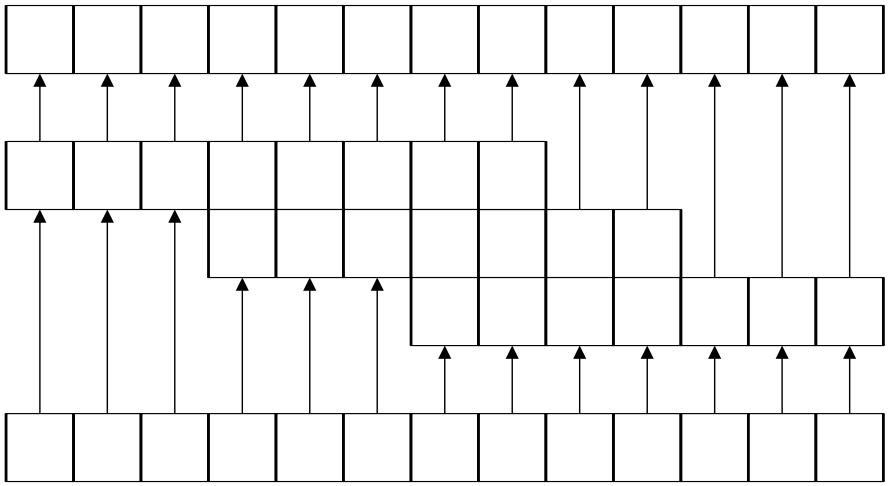


Рис. 4.5. Структура искаженного ошибками бесконечного кодового слова алгебраического нерекурсивного сверточного кода

В кольце многочленов $GF(q^m)[x]/(x^N - 1)$ существует единственный приведенный ненулевой многочлен $h(x)$

наименьшей степени K, который обозначается проверочным многочленом и также однозначно задает (N, K, D) циклический код над $GF(q^m)$ [16, 17]. Соответствующая проверочная матрица (N, K, D) циклического кода может быть записана в виде

,

или в полиномиально-матричном обозначении:

,

где нумерация коэффициентов многочлена идет в обратном $G(x)$ порядке.

Воспользуемся мультипликативно обратным многочлену $P(x)$ в кольце $GF(q^m)[x]/(x^N - 1)$ элементом - проверочным многочленом $h(x)$ циклического (N, K, D) кода, перепишем последнее выражение в виде:

,

что в матричном виде эквивалентно следующему

.

С учетом (4.12) последнее выражение перепишется в виде

Перепишем через проверочный многочлен

,

что в матричной форме примет следующий вид

. (4.17)

Таким образом, как следует из выражения (4.17), бесконечный синдром принятого с ошибками кодового слова алгебраического нерекурсивного сверточного кода состоит из бесконечной суммы синдромов принятых кодовых слов циклического кода, умноженных на соответствующий оператор задержки $x^{i\cdot(N-K)}$. Следовательно, запишем

,

или

, (4.18)

где $S_i(x) = s_{i\cdot K} + s_{i\cdot K+1}x + s_{i\cdot K+2}x^2 + \ldots s_{(i+1)\cdot K-1}x^{K-1}$ – синдромный многочлен циклического *(N, K, D)* кода, $S_i = (s_{i\cdot K}, s_{i\cdot K+1}, s_{i\cdot K+2}, \ldots s_{(i+1)\cdot K-1})$ - соответствующий синдромный вектор.

Значение синдромного многочлена (вектора) зависит только от значения ошибок и не зависит от выбранного кодового слова. Представим, для наглядности, структуру и правило формирования синдромного многочлена на рис. 4.6.

Как видно на рис. 4.6. бесконечный синдром формируется бесконечным суммированием соответствующих синдромов циклического кода $S_i(x)$. Причем синдромы $S_i(x)$ суммируются без наложений, т.е. каждый блок из $(N - K)$ синдромных символов зависит исключительно от блока из K ошибочных символов. Этот факт позволяет реализовать алгебраическое правило декодирования алгебраически заданного сверточного кода.

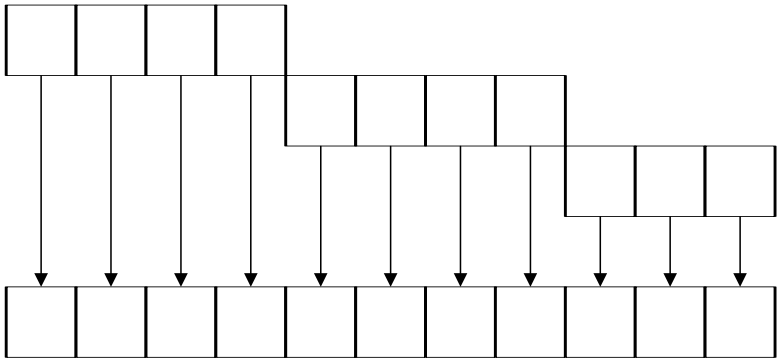


Рис. 4.6. Структура бесконечного синдромного многочлена алгебраического нерекурсивного сверточного кода

Действительно, декодирование бесконечного кодового слова сверточного кода распадается на бесконечную последовательность декодирований кодовых слов циклического *(N, K, D)* кода. Причем каждый синдромный вектор S_i соответствует ошибке, произошедшей на блоке из K символов. В случае неправильного декодирования ошибка распространяется только в пределах блока данных из K символов. Следовательно, независимость блоков синдромных символов позволяет избежать распространения ошибок, которое присуще некоторым известным способам декодирования сверточных кодов [25].

Таким образом, в результате проведенных рассуждений удалось свести декодирование бесконечного кодового слова к бесконечной серии декодирований циклического блокового кода. Рассмотрим теперь декодирование одного блока символов бесконечного сверточного кода, а затем обобщим его на случай бесконечных серий.

Проанализируем выражение (4.17). Оно содержит бесконечную сумму произведений непересекающихся ненулевых векторов ошибок на проверочную матрицу циклического *(N, K, D)* кода, заданного через порождающий многочлен $P(x)$. Очевидно, что матрица может быть записана в виде (4.4), но для декодирования циклических кодов используется другая ее форма, которая отражает структуру колец многочленов и, непосредственно, свойства самого многочлена $P(x)$.

Обозначим через X_l – l – ый корень порождающего многочлена $P(x)$, причем $X_l = \alpha^{J_l}$ для некоторого J_l, $X_l \in GF(q^m)$. Если $X_0, X_1, \ldots, X_{r-1}$ – все корни многочлена $P(x)$, т.е. $P(x) = (x + X_0)\cdot(x + X_1) \cdot \ldots \cdot (x + X_{r-1})$, то справедливо равенство

$$c(X_i) = c_0 + c_1 X_i + c_2 X_i^2 + \ldots + c_{N-1} X_i^{N-1} = 0,$$

где $c(x)$ кодовый многочлен циклического *(N, K, D)* кода.

Перепишем последнее выражение в виде матричного произведения:

$$c(X_i) = (c_0, c_1, c_2, \ldots, c_{N-1}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1})^T = 0,$$

где c кодовое слово циклического *(N, K, D)* кода как набор коэффициентов многочлена $c(x)$.

Обобщим последнее равенство для всех корней $P(x)$, получим:

.

Полученное выражение соответствует условию взаимной ортогональности произвольного кодового слова $c = (c_0, c_1, c_2, \ldots, c_{N-1})$ и матрицы в правой части произведения. Следовательно, положим

. (4.19)

Предположим теперь, что кодовое слово c исказилось при передаче. Пусть число ошибок на блоке из N символов не превышает исправляющей способности $t = (D - 1)/2$ циклического (N, K, D) кода. Обозначим $e(x)$ – многочлен ошибок, так, что

$$e(x) = e_0 + e_1 x + e_2 x^2 + \ldots + e_{N-1} x^{N-1}$$

с $\leq t$ ненулевыми коэффициентами.

Пусть $c^*(x) = c^*_0 + c^*_1 x + c^*_2 x^2 + \ldots + c^*_{N-1} x^{N-1}$ – кодовое слово с ошибками, т.е.

$c^*(x) = c(x) + e(x) = (c_0 + e_0) + (c_1 + e_1)x + (c_2 + e_2)x^2 + \ldots + (c_{N-1} + e_{N-1})x^{N-1}$.

Значение вектора синдромов вычислим из выражения

,

что эквивалентно следующей системе уравнений:

,

, (4.20)

…

.

Задача декодирования блока данных из N символов состоит в нахождении всех e_i, $i = 0,\ldots, N - 1$ по известным элементам синдромной последовательности $(s_0, s_1, \ldots, s_{r-1})$.

Система уравнений (4.20) нелинейная, прямых методов ее решения не известно. Для нахождения вектора ошибок $(e_0, e_1, e_2, \ldots, e_{N-1})$ воспользуемся искусственным приемом. Введем многочлен локаторов ошибок $\Lambda(x)$, корнями которого являются ненулевые элементы вектора ошибок, т.е.

, (4.21)

где j – индекс ненулевых элементов вектора ошибок, X_j - локатор ошибки, произошедшей в j-ом символе кодового слова.

Раскроем скобки в выражении (4.21), получим

$$\Lambda(x) = x^w + \lambda_{w-1}x^{w-1} + \ldots + \lambda_1 x + \lambda_0, \quad (4.22)$$

где степень w многочлена $\Lambda(x)$ задает число произошедших ошибок на блоке из N символов, $w \le t$, т.е. число ненулевых элементов вектора ошибок $(e_0, e_1, e_2, \ldots, e_{N-1})$.

Набор $(\lambda_0, \lambda_1, \ldots, \lambda_{w-1})$ коэффициентов многочлена (4.22) однозначно задает его корни, которые, соответственно, однозначно указывают (локализуют) расположение произошедших ошибок. Умножим многочлен (4. 22) на $e_i X^i$ и вычислим его значение в X_j, получим:

,

где $X_j \in GF(q^m)$, т.е. $X_j = \alpha^{J_j}$ для некоторого J_j.
Следовательно, $X_j^{a+b} = \alpha^{a+b+J_j} = X^b_{j+a}$, т.е. справедливо выражение

Последнее равенство выполняется для любого j и при каждом i. Просуммируем по всем $i = 0\ldots N - 1$, получим

.

Изменив порядок суммирования, вынесем коэффициенты многочлена локаторов ошибок за знак суммирования, получим:

.

Значение каждого слагаемого в последнем выражении соответствует произведению коэффициентов многочлена локаторов ошибок на соответствующие синдромы в выражении (4.20), так что запишем

$$s_{j+w} + \lambda_{w-1} \cdot s_{j+w-1} + \ldots + \lambda_1 \cdot s_{j+1} + \lambda_0 \cdot s_j = 0. \quad (4.23)$$

Перепишем выражение (4.23) для каждого $j = 0\ldots w$, получим систему линейных уравнений:

$$s_w + \lambda_{w-1} \cdot s_{w-1} + \ldots + \lambda_1 \cdot s_1 + \lambda_0 \cdot s_0 = 0,$$

$$s_{w+1} + \lambda_{w-1} \cdot s_w + \ldots + \lambda_1 \cdot s_2 + \lambda_0 \cdot s_1 = 0, \quad (4.24)$$

…

$$s_{2 \cdot w} + \lambda_{w-1} \cdot s_{2 \cdot w-1} + \ldots + \lambda_1 \cdot s_{w+1} + \lambda_0 \cdot s_w = 0.$$

Система из w линейных уравнений (4.24) с w неизвестными разрешима, сложность ее решения растет полиномиально от числа неизвестных [14, 132]. Так, например, для решения системы (4.24) методом Гаусса необходимо выполнить n^2 сложений и умножений над элементами $GF(q^m)$.

Решение системы уравнений (4.24) дает значения коэффициентов многочлена локаторов ошибок (4.22). Корнями многочлена (4.22) являются локаторы - элементы $GF(q^m)$, которые однозначно указывают расположение ошибок. Следовательно, для локализации ошибок необходимо найти корни уравнения (4.22).

Наиболее простая процедура поиска корней многочлена локаторов ошибок состоит в подстановке всех элементов поля $GF(q^m)$ и выборе тех элементов, которые обращают в нуль многочлен (4.22). В литературе такой прием получил название процедуры Ченя [14, 132].

После локализации ошибок - нахождении локаторов ошибок X_j, необходимо вычислить значения ошибок в j-ом символе, т.е. вычислить вектор ошибок $(e_0, e_1, e_2, \ldots, e_{N-1})$ и восстановить кодовое слово: $c = c^* - e$.

Для нахождения значений ошибок воспользуемся выражением (4.20). Подставим значения найденных локаторов X_j и неизвестные значения e_j в систему уравнений. Остальные e_i при $i \neq j$ равны нулю. Следовательно, система уравнений (4.20) запишется в виде:

,

, (4.25)

…

,

где J – множество индексов ненулевых элементов вектора ошибок, т.е. набор номеров локаторов ошибок, причем $|J| = w \leq t$.

Система (4.25) из r линейных уравнений содержит $|J| = w \leq t$ неизвестных значений ошибок e_i, причем $t < r$. Следовательно, система (4.25) разрешима, ее решение дает неизвестные ненулевые значения ошибок вектора $(e_0, e_1, e_2, \ldots, e_{N-1})$. Для восстановления кодового слова длины N кодовых символов достаточно снять действие найденного вектора ошибок:

$$c = c^* - e.$$

Для реализации предложенного подхода алгебраического декодирования бесконечных кодовых слов алгебраического сверточного кода , заданного через порождающий многочлен циклического кода, необходимо и достаточно вычислить бесконечную сумму синдромов соответствующих кодовых слов циклического кода, т.е. вычислить все значения $S_i = (s_{i \cdot K}, s_{i \cdot K+1}, s_{i \cdot K+2}, \ldots s_{(i+1) \cdot K-1})$ в выражении (4.18). Для вычисления синдромной

последовательности в алгебраической теории блоковых кодов используют умножение кодового слова на проверочную матрицу и/или, что эквивалентно , формируют синдромный многочлен $S_i(x)$ через соответствующие операции в кольце многочленов $GF(q)[x]/(x^n - 1)$. Эквивалентной операцией для непрерывных кодов будет произведение кодового слова на полубесконечную проверочную матрицу сверточного кода, заданную через корни порождающего многочлена. Конструктивных способов построения полубесконечной проверочной матрицы несистематического сверточного кода с сохранением таких алгебраических свойств не известно. Следовательно, для реализации предложенного выше подхода алгебраического декодирования сверточных кодов необходимо теоретически обосновать и ввести соответствующие процедуры формирования бесконечной серии синдромных последовательностей $S_i = (s_{i \cdot K}, s_{i \cdot K+1}, s_{i \cdot K+2}, \ldots s_{(i+1) \cdot K-1})$.

4.3. Формирование бесконечной серии конечных синдромов для алгебраического декодирования сверточных кодов

Рассмотрим структуру бесконечного кодового слова алгебраического сверточного кода, схематично представленную на рис. 4.4. Если алгебраический сверточный код задан через порождающий многочлен *(N, K, D)* циклического кода то бесконечное кодовое слово формируется суммированием бесконечного числа кодовых слов циклического кода, сдвинутых на *K* символов вправо. Следовательно, *i*-ый блок из *N* кодовых символов бесконечного кодового слова состоит из суммы *i*-ого кодового слова циклического кода и соответствующих частей *i+j*-ых кодовых слов. Схематично структура произвольного блока из *N* символов бесконечного кодового слова алгебраического нерекурсивного сверточного кода представлена на рис. 4.7.

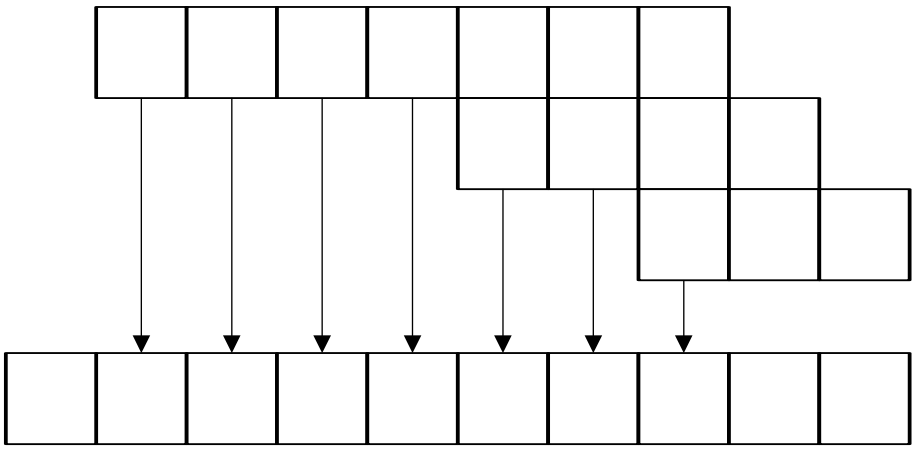


Рис. 4.7. Структура произвольного блока из *N* символов бесконечного кодового слова алгебраического нерекурсивного сверточного кода

Формально, запишем:

, (4.26)

где - блок блока из *N* символов бесконечного кодового слова алгебраического сверточного кода; - *i*-ое кодовое слово циклического кода; - соответствующие подблоки *i+j*-ых кодовых слов циклического кода.

Предположим, что на рассматриваемом блоке из *N* символов произошло не более $t = (D - 1)/2$ ошибок. Сформируем из блока кодовый многочлен и вычислим остаток от деления его на порождающий многочлен циклического кода. Последняя операция эквивалентна умножению на проверочный многочлен и/или произведению блока на проверочную матрицу циклического кода. Получим синдромный вектор *S**:

(4.27)

где $e_i(x)$ – многочлен ошибок, коэффициентами которого являются элементы вектора ошибок длины *N* символов, т.е. вектор ошибки на *i*-ом кодовом слове циклического кода; $Z(x)$ – сумма многочленов, коэффициентами которых являются элементы векторов в выражении (4.26), $j \neq 0$.

Очевидно, что первое слагаемое в выражении (4.27) суть остаток от деления кодового слова циклического кода на соответствующий порождающий многочлен. Следовательно, .

Второе слагаемое в выражении (4.27) соответствует остатку от деления многочлена ошибок кодового слова циклического кода на его порождающий многочлен. Следовательно, во введенных ранее обозначениях.

Третье слагаемое соответствует остатку от деления суммы многочленов из выражения (4.26) на порождающий многочлен циклического кода. Отметим, что все многочлены, коэффициентами которых являются элементы векторов в выражении (4.26), $j \neq 0$ имеют степень $\leq N - K$, а степень порождающего многочлена $\deg g(x) = N - K+1$.

Следовательно, запишем . Тогда выражение (4.27) перепишем в следующем виде

$$S^*(x) = S_i(x) + Z(x). \quad (4.28)$$

Очевидно, что при $Z(x) = 0$ выполняется равенство $S^*(x) = S_i(x)$. Практически это означает, что при выполнении равенства нулю суммы

векторов в выражении (4.26), $j \neq 0$ значения синдромов S^* для блока из N кодовых символов совпадут с соответствующими синдромами $S_i(x)$ i-ых кодовых слов циклического кода. Таким образом, для формирования бесконечной серии конечных синдромов для алгебраического декодирования сверточных кодов достаточно выполнения условия $Z(x) = 0$.

Для выполнения сформулированного условия рассмотрим правило формирования многочлена $Z(x)$. Как показано выше, многочлен $Z(x)$ формируется суммированием многочленов, коэффициентами которых являются элементы кодовых i-ых слов, сдвинутых на $j \cdot K$ символов вправо. Предположим, что бесконечное кодовое слово нерекурсивного сверточного кода алгебраически заданного через порождающий многочлен циклического кода (см. рис. 4.4.) состоит из бесконечной суммы кодовых слов циклического кода, умноженных оператор задержки $x^{i \cdot N}$. Тогда многочлен $Z(x)$ будет равен сумме многочленов, коэффициентами которых являются элементы кодовых i-ых слов, сдвинутых на $j \cdot N$ символов вправо. Но по определению вектор S^* - синдромная последовательность, соответствующая блоку из N кодовых символов бесконечного кодового слова. Практически это означает, что третье слагаемое в выражении (4.27) равно нулю, т.е. $Z(x) = 0$, соответственно. Подставив в (4.28), получим

$$S^*(x) = S_i(x).$$

Последнее выражение позволяет формировать бесконечную серию конечных синдромов бесконечного кодового слова сверточного кода. Практически это позволяет реализовать разработанный выше алгебраический алгоритм декодирования сверточных кодов.

Таким образом, в результате проведенных исследований, разработан способ формирования бесконечной серии конечных синдромов для алгебраического декодирования сверточных кодов. При этом следует отметить некоторое ухудшение конструктивных свойств сверточного кода. Для реализации предлагаемого способа при формировании кодового слова алгебраического сверточного кода оператор задержки $x^{i \cdot K}$ следует заменить на $x^{i \cdot N}$. Практически это означает, что после подачи на вход кодера K информационных символов необходимо подать, дополнительно, $(N - K)$ нулевых символов. В этом случае на приемной стороне выполнится условие $Z(x) = 0$ и, соответственно, равенство $S^*(x) = S_i(x)$. В терминах кодирования подача на вход кодера $(N - K)$ нулевых символов соответствует снижению скорости кодирования в $(N - (N - K))/N = K/N$ раз, т.е. снижение скорости пропорционально скорости циклического (N, K, D) кода. Последнее обстоятельство снижает помехоустойчивость алгебраических сверточных кодов (с алгебраическим способом декодирования). Однако при соответствующем выборе параметров циклического (N, K, D) кода это ухудшение можно минимизировать. Действительно, если при построении сверточного кода использовать циклические (N, K, D) коды с $R = K/N \rightarrow 1$, то для реализации алгебраического декодирования необходимо внести ничтожно малую долю нулевых символов и, таким образом, снижение

помехоустойчивости будет минимальным.

4.4. Комбинированный метод декодирования алгебраически заданных сверточных кодов

Рассмотрим последовательное декодирование (например, алгоритмом Фано) сверточного кода, алгебраически заданного через циклический *(N, K, D)* код. Предположим, что для устранения эффекта переполнения буфера в алгоритме Фано используется искусственное внесение в передаваемые данные набора из $v = k^0 r$ нулей (см. подраздел 3.1). Предположим так же, что эта процедура выполняется с периодичностью $M \cdot K$ информационных символов, где M – произвольное целое, отличное от нуля.

Как показано выше бесконечное кодовое слово сверточного кода распадается на бесконечную сумму кодовых слов циклического кода. Структуру кодового слова с учетом периодичного внесения v нулей представим на рис. 4.8.

Как видно из рис. 4.8 каждые $(M-1)K + N$ кодовых символов суть сумма соответствующих $M \cdot K$ кодовых слов циклического *(N, K, D)* кода, т.е

, (4.29)

где - j-ое кодовое слово циклического кода, ,

- единичная матрица с добавленными слева $i \cdot K$ нулевыми столбцами.

Обозначим, как и ранее, через X_l – l – ый корень порождающего многочлена $P(x)$, причем $X_l = \alpha^{J_l}$ для некоторого J_l, $X_l \in GF(q^m)$.

Если $X_0, X_1, \ldots, X_{r-1}$ – все корни многочлена $P(x)$, т.е. $P(x) = (x + X_0) \cdot (x + X_1) \cdot \ldots \cdot (x + X_{r-1})$, то справедливо равенство

$$c(X_i) = c_{j,0} + c_{j,1} X_i + c_{j,2} X_i^2 + \ldots + c_{j,N-1} X_i^{N-1} = 0,$$

где $c(x)$ кодовый многочлен циклического *(N, K, D)* кода.

Обобщив полученное выражение до суммы (4.35), получим:

$$C_0 + C_1 X_i + C_2 X_i^2 + \ldots + C_{MK+N-1} X_i^{MK+N-1} =$$
$$= (c_{0,0} + c_{0,1} X_i + c_{0,2} X_i^2 + \ldots + c_{0,N-1} X_i^{N-1}) +$$
$$+ X_i^K (c_{1,0} + c_{1,1} X_i + c_{1,2} X_i^2 + \ldots + c_{1,N-1} X_i^{N-1}) + \ldots +$$
$$+ X_i^{MK} (c_{M-1,0} + c_{M-1,1} X_i + c_{M-1,2} X_i^2 + \ldots + c_{M-1,N-1} X_i^{N-1}) = 0.$$

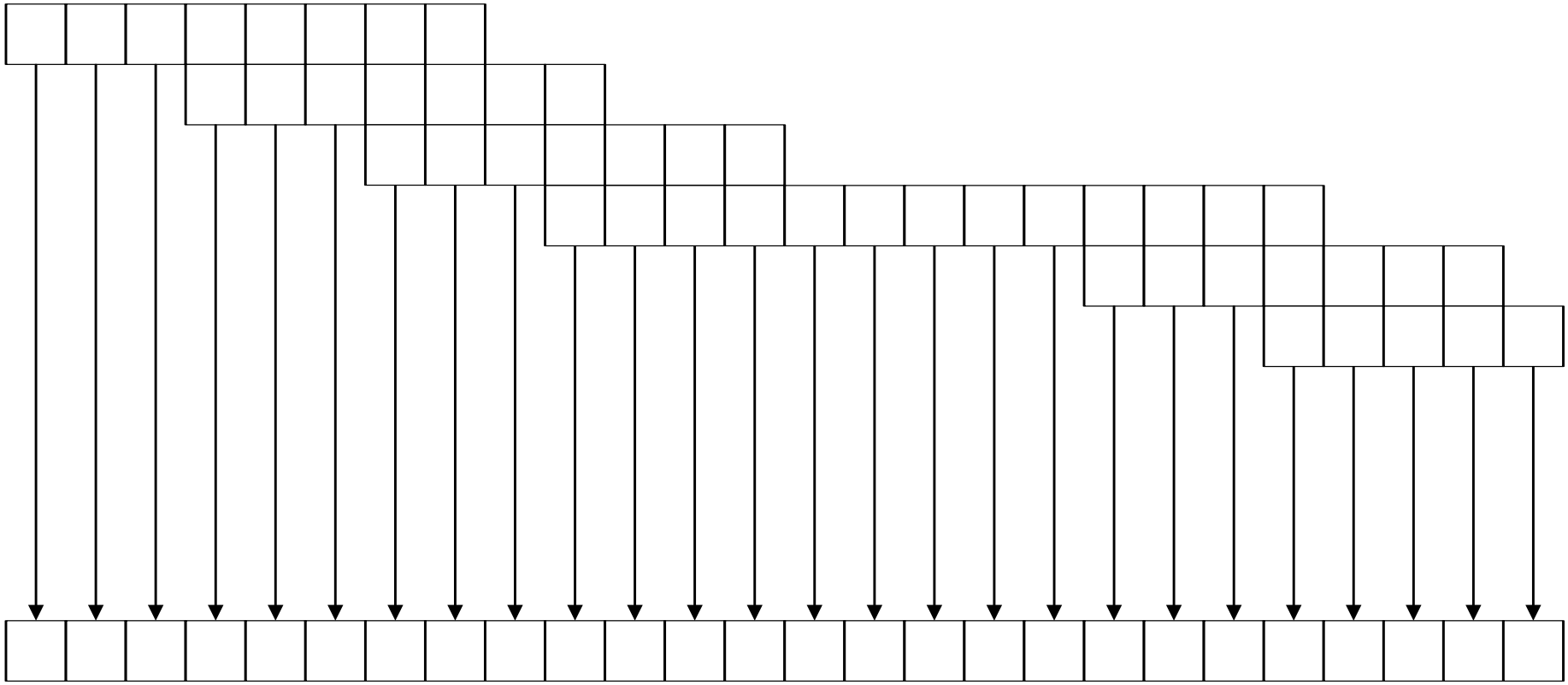


Рис. 4.8. Структура кодового слова алгебраического сверточного кода с периодично вносимыми *v* нулями (период - *M*·*K* информационных символов)

Раскроем скобки, получим:

$$C_0 + C_1 X_i + C_2 X_i^2 + \ldots + C_{MK+N-1} X_i^{MK+N-1} =$$
$$= (c_{0,0} + c_{0,1} X_i + c_{0,2} X_i^2 + \ldots + c_{0,N-1} X_i^{N-1}) +$$
$$+ (c_{1,0} X_i^K + c_{1,1} X_i^{K+1} + c_{1,2} X_i^{K+2} + \ldots + c_{1,N-1} X_i^{K+N-1}) + \ldots +$$
$$+ (c_{M-1,0} X_i^{MK} + c_{M-1,1} X_i^{MK+1} + c_{M-1,2} X_i^{MK+2} + \ldots + c_{M-1,N-1} X_i^{MK+N-1}) = 0.$$

Если $X_i \in GF(Q)$, $Q = q^m$, $X_i \neq 0$, справедливо равенство

$$X_i^{Q+a} = X_i^{a+1}.$$

Если при этом выполняется равенство $N = q^m - 1$, то запишем:

$$C_0 + C_1 X_i + C_2 X_i^2 + \ldots + C_{MK+N-1} X_i^{MK+N-1} =$$
$$= C_0 + C_1 X_i + C_2 X_i^2 + \ldots + C_{N-1} X_i^{N-1} + \ldots + C_{MK+N-1} X_i^{MK-1} =$$
$$= (c_{0,0} + c_{0,1} X_i + c_{0,2} X_i^2 + \ldots + c_{0,N-1} X_i^{N-1}) +$$
$$+ (c_{1,0} X_i^K + c_{1,1} X_i^{K+1} + c_{1,2} X_i^{K+2} + \ldots + c_{1,N-K-1} X_i^0 + \ldots + c_{1,N-1} X_i^{K-1}) + \ldots +$$
$$+ (c_{M-1,0} X_i^{MK^*} + c_{M-1,1} X_i^{(MK+1)^*} + c_{M-1,2} X_i^{(MK+2)^*} + \ldots + c_{M-1,N-1} X_i^{(MK-1)^*}) = 0,$$

где $(\xi)^* = (\xi) mod(q^m - 1)$.

Перепишем последнее выражение в виде матричного произведения:

$$C(X_i) = (C_0, C_1, C_2, \ldots, C_{MK+N-1}) \cdot (1, X_i, X_i^2, \ldots, X_i^{MK+N-1})^T =$$
$$= (C_0, C_1, C_2, \ldots, C_{N-1}, \ldots, C_{MK+N-1}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1}, \ldots, X_i^{MK-1})^T =$$
$$= (c_{0,0}, c_{0,1}, c_{0,2}, \ldots, c_{0,N-1}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1})^T +$$
$$+ (c_{1,0}, c_{1,1}, c_{1,2}, \ldots, c_{1,N-K-1}, \ldots, c_{1,N-1}) \cdot (X_i^K, X_i^{K+1}, X_i^{K+2}, \ldots, X_i^0, \ldots, X_i^{K-1})^T + \ldots +$$
$$+ (c_{M-1,0}, c_{M-1,1}, c_{M-1,2}, \ldots, c_{M-1,N-1}) \cdot (X_i^{MK^*}, X_i^{(MK+1)^*}, X_i^{(MK+2)^*}, \ldots, X_i^{(MK-1)^*})^T = 0.$$

Следует отметить, что в каждом произведении последнего выражения есть сомножитель $(1, X_i, X_i^2, \ldots, X_i^{N-1})^T$, умноженный последовательно на X_i^j, или, что эквивалентно, циклически сдвинутый на K символов вправо. Следовательно, справедливо равенство:

$$C(X_i) = (c_{0,0}, c_{0,1}, c_{0,2}, \ldots, c_{0,N-1}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1})^T +$$
$$+ (c_{1,N-K-1}, c_{1,N-K}, \ldots, c_{1,N-1}, c_{1,0}, c_{1,1}, c_{1,2}, \ldots, c_{1,N-K-2}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1})^T + \ldots +$$
$$+ (c_{M-1,-MK^*}, c_{M-1,(-MK-1)^*}, c_{M-1,(-MK-2)^*}, \ldots, c_{M-1,(-MK+1)^*}) \cdot (1, X_i, X_i^2, \ldots, X_i^{N-1})^T = 0.$$

Обобщим последнее равенство для всех корней $P(x)$, получим:

Приведем подобные, получим:

(4.30)

Последнее выражение соответствует условию взаимной ортогональности вектора $(c_{0,0} + c_{1,N-K-1} + ... + c_{M-1,-MK*}, c_{0,1} + c_{1,N-K} + ... + c_{M-1,(-MK-1)*}, ..., c_{0,N-1} + c_{1,N-K-2} + ... + c_{M-1,(-MK+1)*})$ и матрицы в правой части произведения.

Как показано выше, матрица в правой части произведения суть проверочная матрица *(N, K, D)* циклического кода.

Предположим теперь, что кодовое слово исказилось при передаче. Пусть число ошибок на блоке из $(M-1)K + N$ кодовых символов не превышает исправляющей способности $t = (D - 1)/2$ циклического *(N, K, D)* кода. Тогда кодовое слово с ошибками *C** на длине $(M-1)K + N$ кодовых

символов запишется в виде:

.

Подставим в (4.35), получим:

где - вектор ошибок E_i длины K символов с добавленными справа $(N - K)$ нулями и соответствующий j-му кодовому слову ,

$$E_i(x) = e_{i \cdot K} + e_{i \cdot K+1} x + e_{i \cdot K+2} x^2 + \ldots e_{(i+1) \cdot K-1} x^{K-1}$$

(см. выражение (4.11)).

Вычислим значения кодового многочлена с ошибками во всех корнях порождающего многочлена $P(x)$. После подстановки получим:

Но, как следует из (4.30) первое слагаемое в правой части последнего выражения равно нулю, следовательно:

,

т.е. синдром зависит только от вектора ошибок и не зависит от кодового слова .

Перепишем полученное выражение в виде:

(4.31)

Выражение (4.31) по сути, соответствует ограничению синдромного вектора (4.17) на длину кодового слова *(M-1)K + N* кодовых символов и, очевидно, может быть получено другим способом (другой последовательностью соответствующих преобразований).

Таким образом, в результате проведенных исследований получено обобщенное представление бесконечного кодового слова через бесконечную сумму последовательных наборов из *M* кодовых слов *(N, K, D)* циклического кода. Это дает мощный механизм комбинированного декодирования алгебраически заданных сверточных кодов. Действительно, синдромная последовательность в выражении (4.31) зависит от вектора ошибок, компоненты которого сгруппированы в периодическую структуру. Практически это означает, что соответствующий синдром задает рекуррентное правило формирования систем линейных уравнений (4.24) - (4. 25), решения которых локализуют ошибку с точности до периодичной структуры (коэффициенты вектора ошибок сгруппированы с периодом *K*). При соответствующем выборе параметров циклического *(N, K, D)* кода можно задать период группирования ошибок, превосходящий средний размер пакета ошибок. Это позволит с помощью предложенных алгебраических процедур свести задачу декодирования коррелированных ошибок к декодированию ошибок, близких к независимому распределению. Декодирование независимых ошибок (нахождение значения ошибок, локализованых с точностью до периодичной структуры) может быть возложено на другой метод декодирования сверточных кодов, например на последовательный алгоритм. Действительно, если для устранения эффекта переполнения буфера алгоритма последовательного декодирования (например, алгоритма Фано) используется искусственное внесение в передаваемые данные набора из *v* нулей с периодичностью *M·K* информационных символов, то работа этого алгоритма внутри блока из *M·K* символов может быть существенно упрощена. Поскольку алгебраический алгоритм локализовал (с точностью до периода) ошибку, то последовательный поиск по всей кодовой решетке можно заменить поиском только в тех узлах, которые входят в соответствующую локализацию. Алгоритм такого комбинированного декодирования алгебраически заданного сверточного кода представим в виде последовательности следующих шагов.

Шаг 1. Прием $(M-1)K + N$ кодовых символов из $GF(q^m)$ (или, что эквивалентно, $(M \cdot K + N)m$ кодовых символов из $GF(q)$).

Шаг 2. Вычисление синдрома по выражению (4.31).

Шаг 3. Решение систем линейных уравнений (4.24) - (4.25).

Шаг 4. Локализация ошибок с точностью до периода K символов.

Шаг 5. Последовательный поиск по кодовой решетке в узлах, соответствующих компонентам сгруппированной ошибки.

Шаг 6. Исправление сгруппированной ошибки.

Шаг 7. Прием следующих $(M-1)K + N$ кодовых символов из $GF(q^m)$ (или, что эквивалентно, $(M \cdot K + N)m$ кодовых символов из $GF(q)$). Переход к шагу 2.

Поиск по кодовой решетке (шаг 5) может выполняться и сразу, после приема первого кодового символа (как при последовательном декодировании). Это может быть оправдано при малом числе ошибок. Если число ошибок велико, то сложность последовательно декодирования быстро возрастает и, очевидно, следует ожидать предварительной локализации ошибок (шаг 4).

На рис. 4.9 приведена схема алгоритма комбинированного декодирования алгебраически заданного сверточного кода.

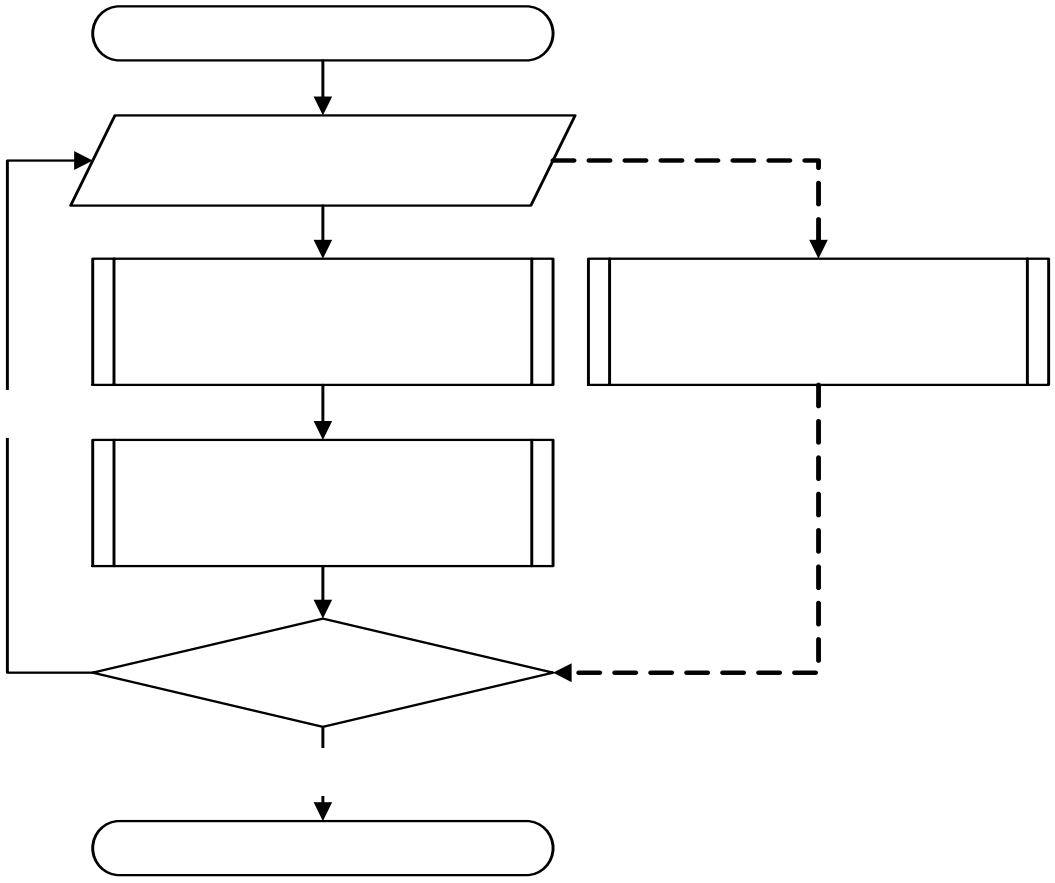


Рис. 4.9. Схема алгоритма комбинированного декодирования алгебраически заданного сверточного кода

Рассмотрим пример декодирования алгебраически заданного сверточного кода. Зафиксируем конечное поле $GF(2^3)$ и порождающий многочлен (7, 3, 5) кода РС, например $g(x) = (x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3)\cdot(x + \alpha^4)$, где α - примитивный элемент поля $GF(2^3)$. Тогда, по теоремам 3.1-3.3 имеем алгебраически заданные сверточные коды с параметрами:

1. $k^0 = 1, n^0 = 3, \nu = 4, k = 5, n = 15, R = 1 / 3, d_\infty \geq 5;$
2. $k^0 = 2, n^0 = 3, \nu = 8, k = 10, n = 15, R = 2 / 3, d_\infty \geq 5.$

В этом случае для реализации алгебраического алгоритма декодирования на передающей стороне после подачи на вход кодера трех информационных символов следует подать четыре нуля. В результате, скорость сверточного кода будет равна, соответственно, $R = 1 / 7$ и $R = 2 / 7$, что, очевидно, очень сильно ухудшает параметры сверточного кода что, соответственно, существенно снизит его помехоустойчивость.

Рассмотрим теперь (63, 55, 5) код БЧХ над $GF(2^3)$ с порождающим многочленом, например, вида $g(x) = (x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3)\cdot(x + \alpha^4)\cdot(x + \alpha^8)\cdot(x + \alpha^{16})\cdot(x + \alpha^{24})\cdot(x + \alpha^{32})$, где α - примитивный элемент поля $GF((2^3)^2)$. Тогда, по теоремам 1-3 (см. раздел 3) имеем алгебраически заданные сверточные коды с параметрами:

1. $k^0 = 1, n^0 = 3, \nu = 8, k = 9, n = 27, R = 1 / 3, d_\infty \geq 5;$
2. $k^0 = 2, n^0 = 3, \nu = 8, k = 18, n = 27, R = 2 / 3, d_\infty \geq 5.$

Теперь, для реализации алгебраического алгоритма декодирования сверточного кода со сходными параметрами на передающей стороне после подачи на вход кодера 55 информационных символов следует подать 8 нулей . В результате, скорость сверточного кода будет равна, соответственно, $R = 55 / 189 \approx 1/3$ и $R = 110 / 189 \approx 2/3$, что, очевидно, практически оставляет без изменения параметры сверточного кода и, соответственно, не приводит к значительному ухудшению его помехоустойчивости.

Другой подход в сохранении высоких конструктивных показателей может состоять в использовании циклических кодов, заданных над большим полем. Приведем, например, следующее сравнение. Зафиксируем конечное поле $GF(2^6)$ и порождающий многочлен (63, 59, 5) кода РС, например $g(x) = (x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3)\cdot(x + \alpha^4)$, где α - примитивный элемент поля $GF(2^6)$. Тогда, по теоремам 3.1-3.3 имеем алгебраически заданные сверточные коды с параметрами:

1. $k^0 = 1, n^0 = 6, \nu = 4, k = 5, n = 30, R = 1 / 6, d_\infty \geq 5;$
2. $k^0 = 2, n^0 = 6, \nu = 4, k = 10, n = 30, R = 1 / 3, d_\infty \geq 5.$
3. $k^0 = 3, n^0 = 6, \nu = 4, k = 15, n = 30, R = 1 / 2, d_\infty \geq 5;$
4. $k^0 = 4, n^0 = 6, \nu = 4, k = 20, n = 30, R = 2 / 3, d_\infty \geq 5;$
5. $k^0 = 5, n^0 = 6, \nu = 4, k = 25, n = 30, R = 5 / 6, d_\infty \geq 5.$

Второй и четвертый коды в приведенном списке по конструктивным параметрам очень близки к рассмотренным выше, однако имеют меньшую длину кодового ограничения. Для реализации алгебраического алгоритма их декодирования на передающей стороне после подачи на вход кодера 59 информационных символов следует подать 4 нуля. В результате, скорость

сверточного кода будет равна, соответственно, $R = 59 / 189 \approx 1/3$ и $R = 118 / 189 \approx 2/3$, что, очевидно, лучший результат, по сравнению с использованием (63, 55, 5) кода БЧХ над $GF(2^3)$.

Теперь зафиксируем $(7, 3, 5)$ код РС над $GF(2^3)$ с порождающим многочленом:

$$P(x) = (x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3)\cdot(x + \alpha^4) = x^4 + x^3 + \alpha^1 x^2 + \alpha^6 x + \alpha^5,$$

где α - примитивный элемент поля $GF(2^3)$.

Пусть задан алгебраический сверточный код с параметрами: $k^0 = 1$, $n^0 = 3$, $\nu = 4$, $k = 5$, $n = 15$, $R = 1 / 3$, $d_\infty \geq 5$. Соответствующие порождающие многочлены сверточного кода равны:

$$P_1(x) = x^4 + x^3 + x + 1,$$
$$P_2(x) = x^2 + 1,$$
$$P_1(x) = x + 1.$$

Схема кодера приведена на рис. 4.10. Соответствующая кодовая решетка приведена на рис. 4.11.

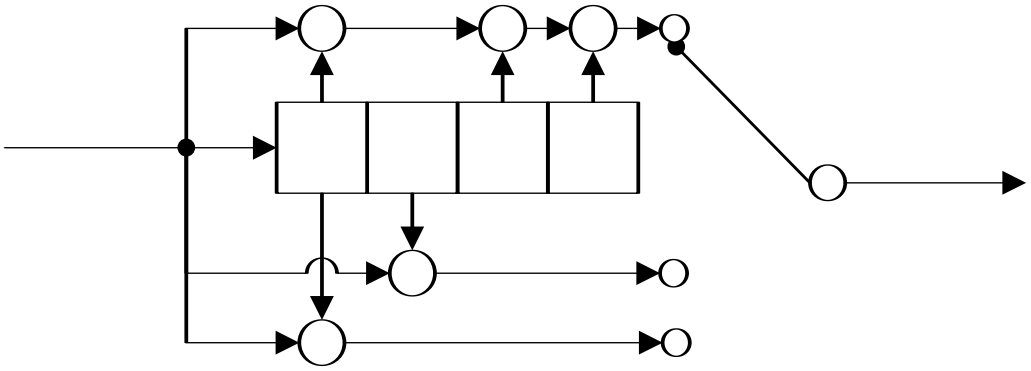


Рис. 4.10. Схема кодера алгебраического сверточного (15, 5) кода

Предположим, что используется комбинированный метод декодирования с $M = 2$. Тогда структуру кодового слова представим в виде рис. 4.12.

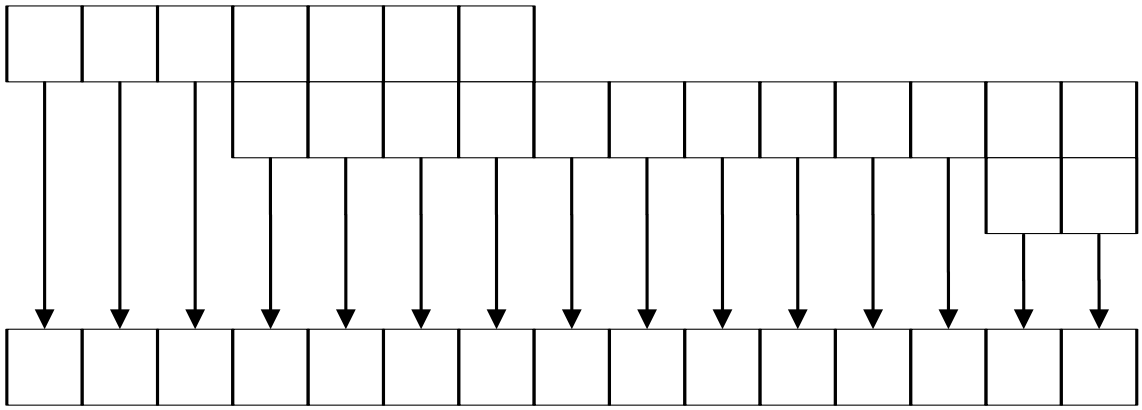


Рис. 4.12. Структура кодового слова сверточного (15, 5) кода при комбинированном алгоритме декодировании ($M = 2$)

Воспользуемся алгоритмом комбинированного декодирования. Рассмотрим блок данных из $(M-1)K + N = 11$ кодовых символов из $GF(q^m)$, т. е. фрагмент бесконечного кодового слова из 11 символов:
$||C||_{10} = (C_0, C_1, ..., C_{10})$.

Пусть на вход кодера подается информационная последовательность $I = (1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 1, ...)$. Тогда с выхода кодера в канал поступает кодовое слово $C = (111, 101, 101, 001, 110, 011, 001, 101, 001, 110, 011, ...)$.

Предположим также, что при передаче по каналу связи на $||C||_{10}$ произошло 2 ошибки, например, $||e||_{10} = (0, 1, ..., 0, 1, 0)$, т.е. ошибки произошли в C_1 и C_9 (см. табл. 4.1).

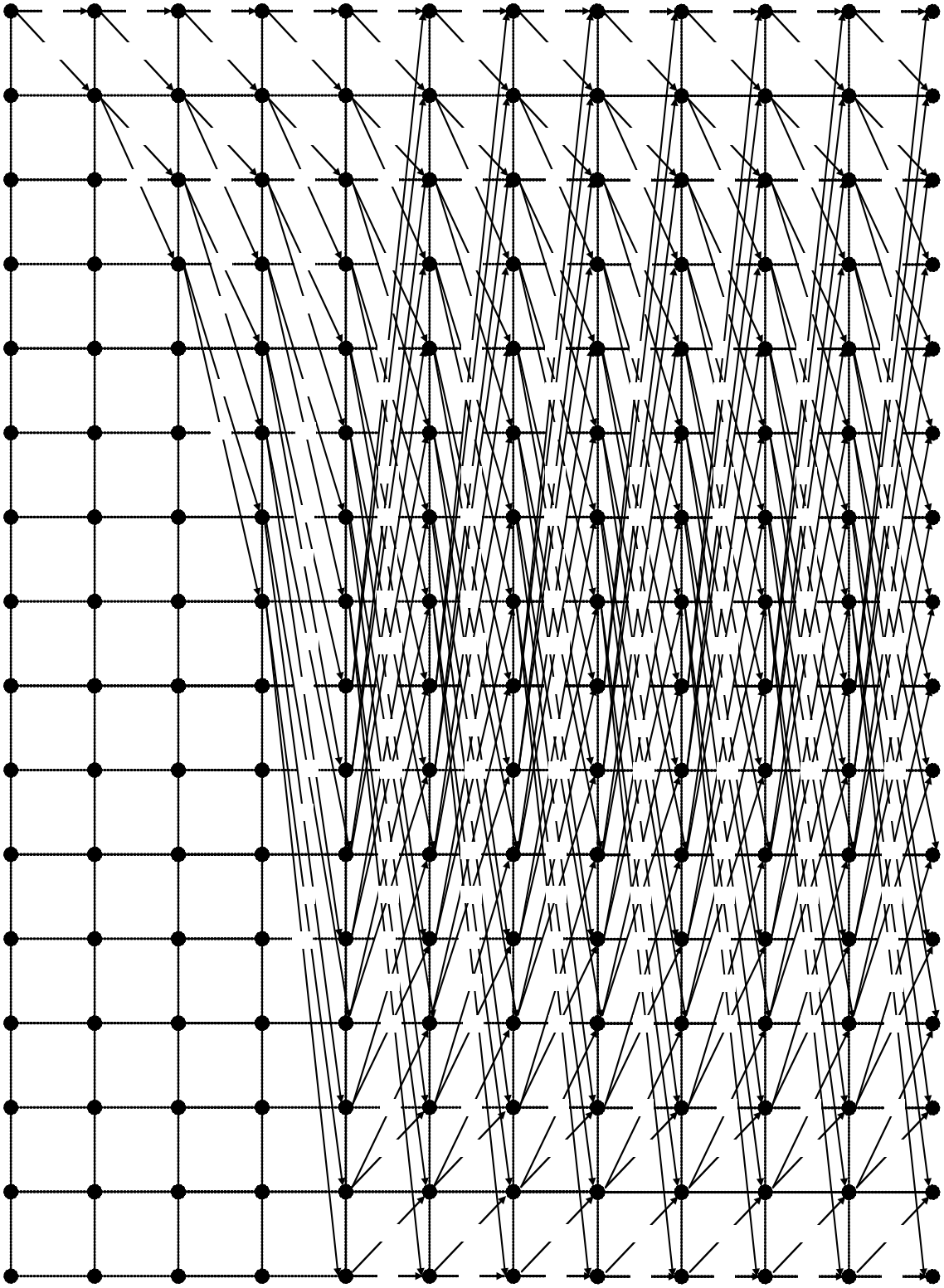


Рис. 4.11. Кодовая решетка алгебраического сверточного (15, 5) кода

Таблица 4.1

Кодовое слово, вектор ошибки, искаженное кодовое слово сверточного (15, 5) кода и фрагменты последовательного декодирования

	0	1	2	3	4	5	6	7	8	9	10	…
C	*111*	*101*	*101*	*001*	*110*	*011*	*001*	*101*	*001*	*110*	*011*	...
e		*100*								*100*		
*C**	*111*	*001*	*101*	*001*	*110*	*011*	*001*	*101*	*001*	*010*	*011*	...
t(l)	*1*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*8*	*9*	
t(l)-		*0*								*7*		

Вычислим синдром:

Подставим вместо X_i корни порождающего многочлена $P(x) = (x + \alpha^1)\cdot(x + \alpha^2)\cdot(x + \alpha^3)\cdot(x + \alpha^4)$. Для всех $\alpha^I \neq 0$ выполняется условие $(\alpha^i)^{Q+a} = (\alpha^i)^{a+1}$, следовательно, запишем:

Найденный синдромный вектор задает систему линейных уравнений:

$$\alpha^4 + \lambda_1 \cdot \alpha^1 + \lambda_0 \cdot \alpha^4 = 0,$$

$$\alpha^2 + \lambda_1 \cdot \alpha^4 + \lambda_0 \cdot \alpha^1 = 0.$$

Решив систему уравнений, получим $\lambda_0 = \alpha^3$, $\lambda_1 = \alpha^4$, что дает возможность сформировать многочлен локаторов ошибок:

$$\Lambda(x) = x^2 + \alpha^4 x + \alpha^3.$$

Воспользуемся процедурой Ченя, найдем корни $\Lambda(x)$. Получим:

$$\Lambda(x) = (x + \alpha^1)\cdot(x + \alpha^2),$$

откуда следует, что ошибки локализуются следующим образом: $(e_1 + e_8)$ и $(e_2 + e_9)$. Найдем значение ошибок. С учетом локализации сформируем систему линейных уравнений:

$$(e_1 + e_8) \cdot \alpha^1 + (e_2 + e_9) \cdot \alpha^3 = \alpha^4,$$

$$(e_1 + e_8) \cdot \alpha^2 + (e_2 + e_9) \cdot \alpha^4 = \alpha^1.$$

Решение системы уравнений дает следующее:

$$(e_1 + e_8) = \alpha^0 = (100),$$

$$(e_2 + e_9) = \alpha^0 = (100).$$

Полученное решение означает следующее – произошло две ошибки:

- одна ошибка *(100)* в символе C_1 или C_8;
- другая ошибка *(100)* произошла в символе C_2 или C_9.

Таким образом, разработанная алгебраическая процедура локализации ошибок позволила установить размещение ошибок с точностью до периода в семь символов. Практически это означает, что при выполнении последовательного поиска (шаг 5 в алгоритме комбинированного декодировании) необходимо осуществить поиск по решетке только в символах *(C_1 или C_8)* и *(C_2 или C_9)*. Строго говоря, на этом этапе может использоваться любой алгоритм декодирования сверточных кодов, который позволяет принять решение о текущем символе. Решение принимается с учетом того, что все символы за исключением *(C_1 или C_8)* и *(C_2 или C_9)* приняты без ошибок.

Рассмотрим теперь работу комбинированного алгоритма декодирования на шаге 5 – последовательный поиск по кодовой решетке в узлах, соответствующих компонентам сгруппированной ошибки. Поскольку известно, что ошибок в символах $C_0, C_3 – C_7, C_{10}$ не произошло, поиск по решетке в этих местах проводить не нужно, что существенно ускоряет работу последовательного декодера. Перейдем сразу к символу C_1.

На рис. 4.13 приведена иллюстрация работы алгоритма Фано при комбинированном декодировании. Предположим, что исходными данными алгоритма Фано являются следующие: $P_0 = 0,3$, $P^* = 1/3$.

Тогда правило принятия решения запишем в виде:

$$t(l) = P^* n_0 l – d(l) = l - d(l),$$

т.е. движение по решетке считается правильным, если величина *t(l)* не уменьшается. В противном случае результат поиска считается не верным и нужно изменить путь.

В табл. 4.1. в строке «*t(l)*» отмечены значения *t(l)* для истинного пути (выделен на рис. 3.13). В строке «*t(l)*-» отмечены значения *t(l)* для ошибочно выбранного пути (тонкие стрелки на рис. 3.13).

Действительно, при работе алгоритма Фано вычисленное в символе C_1 значение *t(l)* для истинного пути равно следующему:

$$t(l) = 2 - d(101, 001) = 2 – 1 = 1,$$

т.е. *t(l)* не уменьшается.

Для другого направления движения (ошибочного) это значение равно:

$$t(l) = 2 - d(010, 001) = 2 – 2 = 0$$

и *t(l)* уменьшается, что свидетельствует об ошибочности этого пути.

Далее последовательный поиск по кодовой решетке осуществляется в символе C_2, что дает аналогичные результаты. В символах C_3-C_7 поиск производить не требуется, поскольку заранее известно, что там ошибок нет. Результаты поиска по кодовой решетке в символах C_8 и C_9 с помощью алгоритма Фано представлены в табл. 4.1 и отражены на рис. 4.13.

Следует отметить, что исправление ошибок в символах C_2 и C_9 можно осуществить несколько быстрее, а весь процесс декодирования еще больше упростить. Действительно, как только величина *d(l)* становиться отличной от нуля, очевидно, что этот кадр с ошибкой. Но значение ошибки уже вычислены алгебраической процедурой: $(e_1 + e_8) = \alpha^0 = (100)$, $(e_2 + e_9) = \alpha^0 = (100)$.

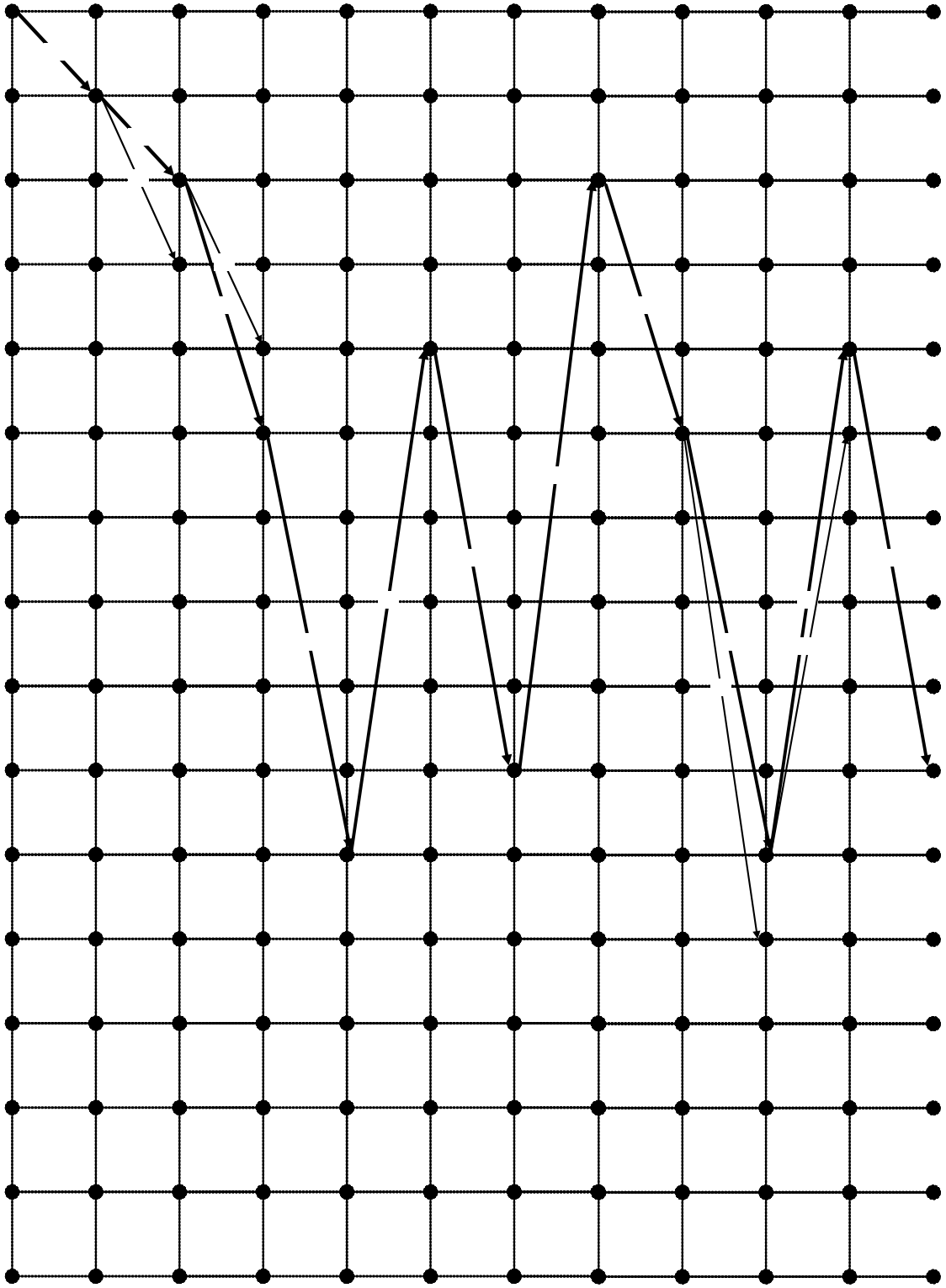


Рис. 4.13. Иллюстрация работы алгоритма Фано
при комбинированном декодировании (шаг 5)

Если последовательный поиск привел к $d(l)>0$ в символе C_1, значит $e_1=\alpha^0 = (100)$, а $e_8 = 0 = (000)$ и можно сразу исправить ошибку в символе C_1 и отказаться от ее поиска в символе C_8. Подобную процедуру можно

выполнить и в символах C_2 и C_9.

Таким образом, последовательный поиск необходим только для уточнения расположения ошибок среди символов, сгруппированных алгебраической процедурой.

Приведенный пример показывает, что разработанная алгебраическая процедура локализации ошибок позволяет установить размещение ошибок с точностью до периода в семь символов. Дальнейшее декодирование основано на выполнении последовательного поиска по кодовой решетке в тех символах, где предположительно произошли ошибки.

Таким образом, в ходе проведенных исследований установлено, что применение комбинированного метода позволяет существенно ускорить процедуру декодирования алгебраически заданных сверточных кодов.

Выводы

1. В результате проведенных исследований получено теоретическое обобщение задачи декодирования линейного блокового кода на случай бесконечной длины кодового слова.

2. Представление бесконечного синдрома кодового слова алгебраически заданного нерекурсивного сверточного кода бесконечной суммой синдромов кодовых слов циклического кода позволяет свести декодирование бесконечного кодового слова сверточного кода к бесконечной серии декодирований кодовых слов циклического блокового кода. Для реализации предложенного подхода необходимо и достаточно вычислить бесконечную сумму синдромов соответствующих кодовых слов циклического кода.

3. Разработанный способ формирования бесконечной серии конечных синдромов позволяет получить взаимно независимые синдромы путем подачи на вход кодера $(N - K)$ нулевых символов, следующих за N информационными символами.

4. Полученное обобщенное представление бесконечного кодового слова алгебраически заданного сверточного кода через бесконечную сумму последовательных наборов из M кодовых слов (N, K, D) циклического кода дает мощный механизм комбинированного декодирования алгебраически заданных сверточных кодов (с использованием предложенных алгебраических процедур и известных алгоритмов, например, последовательного алгоритма Фано). Применение предложенных алгебраических процедур декодирования позволяет локализовать ошибки в кодовом слове сверточного кода с точностью до некоторого, заранее заданного, периода. Это позволяет значительно упростить работу второго алгоритма, например, ускорить последовательный поиск по кодовой решетке.

5. Разработанные впервые алгебраический и комбинированный методы и алгоритмы декодирования алгебраически заданных сверточных кодов, отличаются от известных методов процедурами алгебраической локализации и ускоренными процедурами (алгоритмами) последовательного поиска, что

позволяет реализовать вычислительно эффективное (вычислительно реализуемое) декодирование непрерывных кодовых конструкций с большой длиной кодового ограничения (с большим кодовым расстоянием) и повысить достоверность передаваемой информации.

6. Актуальным направлением прикладного применения полученных научных результатов является разработка параллельных каскадных кодовых конструкций на основе алгебраически заданных рекурсивных сверточных кодов и вычислительно эффективных алгоритмов их декодирования.

РАЗДЕЛ 5
ПАРАЛЛЕЛЬНЫЕ КАСКАДНЫЕ КОНСТРУКЦИИ НА ОСНОВЕ АЛГЕБРАИЧЕСКИ ЗАДАННЫХ РЕКУРСИВНЫХ СВЕРТОЧНЫХ КОДОВ И ВЫЧИСЛИТЕЛЬНО ЭФФЕКТИВНЫЕ АЛГОРИТМЫ ИХ ДЕКОДИРОВАНИЯ

Важной задачей практического применения синтезированных алгебраически заданных сверточных кодов является разработка параллельных каскадных кодовых конструкций (турбокодов), выработка практических рекомендаций по их программной, аппаратной и аппаратно-программной реализации.

В данном разделе исследуются методы построения параллельных каскадных кодовых конструкций и процедуры их декодирования. Предлагаются схемы турбокодирования с использованием рекурсивных сверточных кодов, заданных через порождающий и/или проверочный многочлены недвоичного циклического кода. Разрабатываются алгоритмы построения турбокодов с требуемыми параметрами.

5.1. Методы построения параллельных каскадных кодов и процедур их декодирования

Энергетическая эффективность телекоммуникационных систем зависит , в первую очередь, от энергетической эффективности применяемых методов помехоустойчивого кодирования, формирования и обработки сигнально-кодовых конструкций [116-119]. Под энергетической эффективностью здесь и далее будем понимать минимально допустимое значение отношения энергии сигнала к спектральной плотности мощности шума E/N_o, требуемое для обеспечения заданной достоверности приема сообщения [117-119].

Проведенный анализ показал, что применение турбокодов позволяет получить высокую энергетическую эффективность помехоустойчивого кодирования [41-59]. Так, в работах [41, 42] показано, что использование турбокодов обеспечивает передачу сообщений с вероятностью ошибки $P_{ош} = 10^{-5}$ при величине E/N_o, превышающей лишь на 0,5 дБ минимально необходимую (граничную) величину для заданной скорости передачи информации, что является лучшим на сегодняшний день результатом.

Проведем исследование методов построения турбокодов и процедур их декодирования. Исследуем влияние параметров турбокодов на помехоустойчивость и энергетическую эффективность.

Основным принципом построения турбокодов является параллельное соединение двух рекурсивных систематических сверточных кодеров (Recursive Systematic Convolutional Codes — RSC). Структурная схема турбокодера, в общем виде, представлена на рис. 5.1.

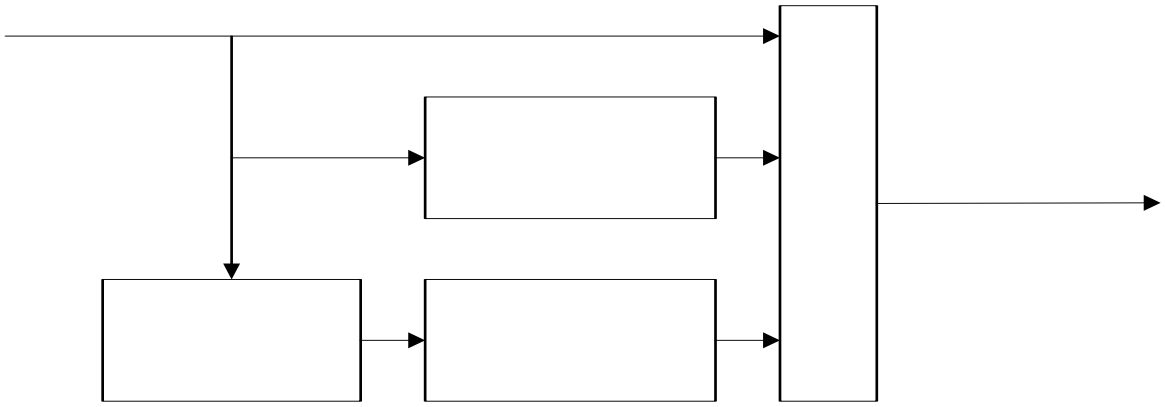


Рис. 5.1. Структурная схема турбокодера

Работа кодера осуществляется следующим образом. На вход подается информационная последовательность $I = \{I_1, I_2, I_3, ...\}$, которая одновременно поступает на вход первого систематического сверточного кодера, на вход перемежителя и на вход мультиплексора. В перемежителе информационная последовательность перемешивается и подается на вход второго систематического сверточного кодера. Оба сверточных кодера формируют по введенной информационной последовательности проверочную часть: P и P^*, соответственно. Сформированные проверочные данные поступают на вход мультиплексора. В мультиплексоре формируется кодовое слово C путем поочередного считывания поступивших на его вход информационных и проверочных символов: $C = \{I_1, P_1, P^*_1, I_2, P_2, P^*_2, ...\}$. Если скорость используемых сверточных кодов $R_{CK} = 1/2$, то скорость турбокода составляет $R_{TK} = 1/3$ (см. рис. 5.1). В общем случае, если $R_{CK} = 1/m$ то скорость турбокода определяется следующей леммой.

Лемма 5.1. Скорость турбокода, построенного на систематических рекурсивных сверточных кодах с $R_{CK} = 1/m$, задается выражением

. (5.1)

Доказательство. Турбокодер, по определению, это параллельное каскадное соединение двух рекурсивных систематических сверточных кодов. Если скорость сверточных кодов $R_{CK} = 1/m$ то на каждый информационный символ оба кодера формируют m кодовых символов. Если сверточный код задан в систематическом виде, то m кодовых символов содержат один информационный и $(m - 1)$ проверочных символов. В мультиплексоре поочередно считываются один информационный символ и с выхода каждого сверточного кодера по $(m - 1)$ проверочных символов. Всего, для каждого информационного символа, поданного на вход турбокодера, на выходе мультиплексора будет получено $1 + (m - 1) + (m - 1) = 2 \cdot m - 1$ символов. Отношение последних задает скорость турбокодирования.

Для турбокодов, построенных на несистематических рекурсивных сверточных кодах, справедлива лемма.

Лемма 5.2. Скорость турбокода, построенного на несистематических рекурсивных сверточных кодах с $R_{СК} = 1/m$, задается выражением

. (5.2)

Доказательство очевидно. Если каждый сверточный кодер задан в несистематическом виде, то на вход мультиплексора подается по m символов , без разделения их на информационные и проверочные. В результате, для каждого информационного символа, поданного на вход турбокодера, будет сформировано $2 \cdot m$ кодовых (выходных) символов.

Таким образом, в рассмотренной схеме (рис. 5.1) формирование кодового слова осуществляется считыванием информационных и проверочных символов с двух параллельно соединенных сверточных кодеров . В канал связи поступает два кодовых слова сверточных кодов, соединенных в одно кодовое слово турбокода. Действительно, зная правило работы перемежителя, всегда можно по известному информационному вектору $I = \{I_1, I_2, I_3, ...\}$ сформировать перемешанную последовательность $I^* = \{I^*_1, I^*_2, I^*_3, ...\}$. Вектор $I = \{I_1, I_2, I_3, ...\}$ в явном виде записан в кодовом слове $C = \{I_1, P_1, P^*_1, I_2, P_2, P^*_2, ...\}$, из которого можно так же выделить проверочные части $P = \{P_1, P_2, P_3, ...\}$ и $P^* = \{P^*_1, P^*_2, P^*_3, ...\}$. Вектор $I = \{I_1, I_2, I_3, ...\}$ и вектор $P = \{P_1, P_2, P_3, ...\}$ в совокупности образуют кодовое слово первого систематического сверточного кода. Вектор $I^* = \{I^*_1, I^*_2, I^*_3, ...\}$ и вектор $P^* = \{P^*_1, P^*_2, P^*_3, ...\}$ в совокупности образуют кодовое слово второго систематического сверточного кода.

В источниках [41-44] рассматриваются так же перфорированные (выколотые) турбокоды. Кодовое слово выколотого кода формируется путем выкалывания некоторых проверочных символов, в результате чего повышается скорость R_{TK} турбокода и несколько снижается его корректирующая способность.

Параллельное соединение двух сверточных кодов и одновременная передача в канал связи их кодовых слов, соединенных в одно слово турбокода, позволяет существенно повысить энергетическую эффективность помехоустойчивого кодирования. Действительно, если энергетическая эффективность каждого сверточного кода в отдельности определяется его исправляющей способностью, т.е. величиной $t = \lfloor (d_\infty - 1)/2 \rfloor$, то энергетическая эффективность турбокода определяется некоторой средней величиной $t_{ср} = \lfloor (d_{ср} - 1)/2 \rfloor$, т.е. выражается через среднее значение $d_{ср}$ расстояний между кодовыми блоками. Причем, в большинстве случаев, $d_{ср} > d_\infty$ и, соответственно, $t_{ср} > t_\infty$. Графически эту особенность турбокодов представим на рис. 5.2.

На рис. 5.2. представлены сферы упаковки турбокода в виде соответствующих сфер упаковки двух рекурсивных сверточных кодов. Символами $C_1, C_2, ...$ обозначены кодовые слова первого сверточного кода. Радиус каждой сферы задает вес ошибки в соответствующем кодовом слове, которую потенциально можно исправить первым сверточным кодом. Минимальный радиус сфер, образованных вокруг всех кодовых слов

$C_1, C_2, \ldots$ задает его исправляющую способность.

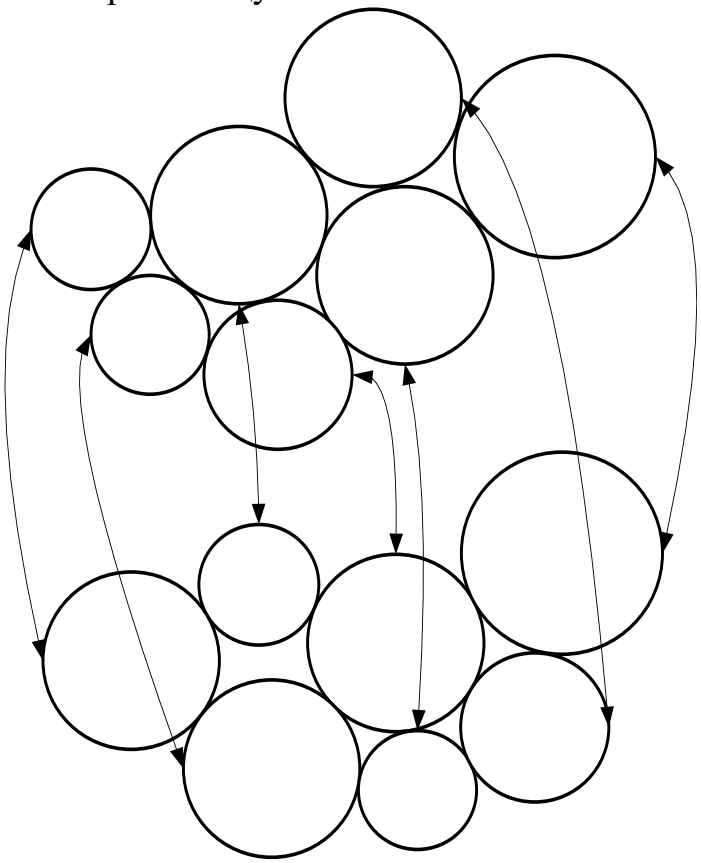


Рис. 5.2. Сферы упаковки турбокода

Символами $C^*_1, C^*_2, \ldots$ обозначены кодовые слова второго сверточного кода. Минимальный радиус сфер, образованных вокруг кодовых слов $C^*_1, C^*_2, \ldots$ задает его исправляющую способность.

Стрелками обозначено объединение кодовых слов двух сверточных кодов в одно кодовое слово турбокода. Среднее значение между радиусами соответствующих кодовых слов задает вес ошибки, которую потенциально можно исправить турбокодом в этом слове. Минимальное значение из всех средних радиусов задает исправляющую способность турбокода. Этот показатель будет зависеть, очевидно, от спектральных распределений сверточных кодов, т.е. от распределения радиусов сфер упаковки кода по различным кодовым словам. В тоже время, вычисление спектра кода является одной из сложнейших задач теории помехоустойчивого кодирования, которая решена на сегодняшний день лишь для узкого класса кодов. Однако, как показано в [41, 42], применение рекурсивного сверточного кодера (кодер с обратной связью), имеющего неограниченную реакцию при воздействии на его вход единичного символа, позволяет получить наиболее благоприятную форму спектрального распределения с точки зрения влияния его на вероятность ошибочного декодирования.

Для эффективной реализации турбокодов необходимо обеспечить параллельное декодирование первого и второго сверточного кода и, таким образом, реализовать их среднюю исправляющую способность. При этом решение об ошибке в каждом символе принятого слова должно приниматься

взвешенно, с учетом соответствующего решения первого и второго декодера. Структурная схема турбодекодера представлена на рис. 5.3. Декодер работает следующим образом.

Поступившее кодовое слово $C = \{I_1, P_1, P^*_1, I_2, P_2, P^*_2, ...\}$ разделяется в демультиплексоре на информационную часть $I = \{I_1, I_2, I_3, ...\}$ и две проверочные части: $P = \{P_1, P_2, P_3, ...\}$ и $P^* = \{P^*_1, P^*_2, P^*_3, ...\}$, соответственно.

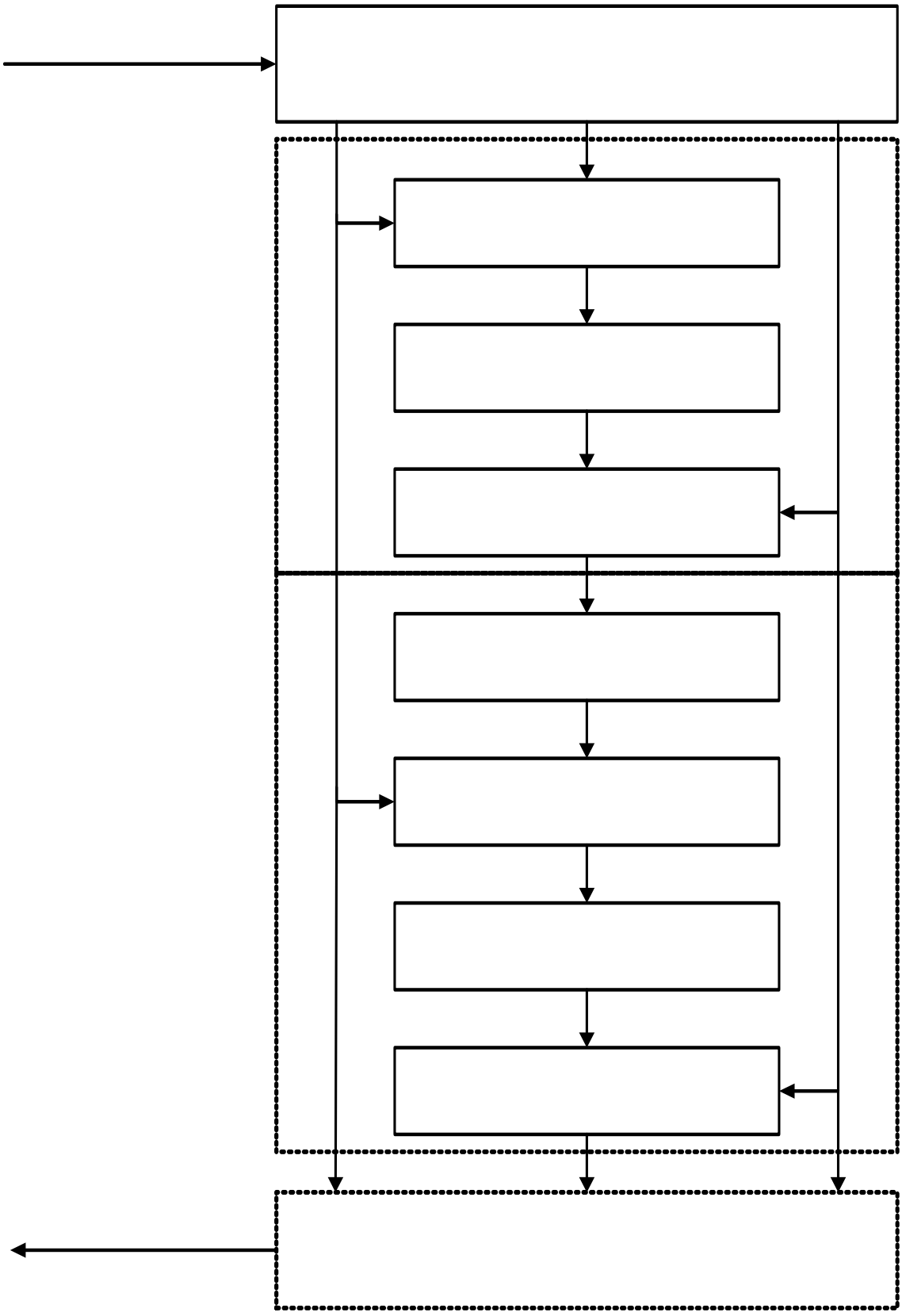


Рис. 5.3. Схема турбодекодера

На первой итерации на вход первого декодера поступают "мягкие" оценки (решения) символов информационной *I* и первой проверочной частей

P кодового слова первого сверточного кода. На выходе первого декодера формируется "мягкая" оценка (решение) информационных символов I_+, которая затем используется в качестве априорной информации для второго декодера.

Второй декодер производит "мягкую" оценку информационных символов с выхода перемежителя (I^*_+) на основе проверочной части *P** кодового слова второго сверточного кода.

На второй и последующих итерациях декодирования эта оценка обновляется и используется как "мягкая" априорная информация о переданном символе для первого декодера. Таким образом, на вход каждого из двух элементарных декодеров первого и второго сверточного кода поступают "мягкие" решения. В результате декодирования так же формируется "мягкое" решение. В зарубежной литературе такая процедура декодирования получила название Soft Input Soft Output (SISO) [41-59]. Окончание декодирования происходит либо после выполнения заданного количества итерационных циклов, либо после того, как величина поправки результата декодирования достигнет установленного порога.

Анализ результатов экспериментальных исследований энергетической эффективности турбокодов показал, что структура перемежителя слабо влияет на его эффективность [63, 64, 67, 69, 71]. В тоже время она пропорционально увеличивается с ростом длины кодового ограничения *v* используемых сверточных кодов [41, 42]. Однако отсутствие регулярных алгебраических алгоритмов построения сверточных кодов с хорошими конструктивными (n, k, d_∞) параметрами и большим *v* сдерживает дальнейшее развитие методов турбокодирования. Актуальной научной задачей является разработка и исследование турбокодов, построенных с использованием алгебраических рекурсивных сверточных кодов.

Воспользуемся алгебраическим методом построения рекурсивных сверточных кодов для определения турбокодов с заранее заданными конструктивными характеристиками.

5.2. Турбокоды на основе алгебраически заданных несистематических рекурсивных сверточных кодов

Для построения турбокодов воспользуемся результатами теорем из раздела 3. Доказанные теоремы дают мощный механизм построения алгебраических несистематических рекурсивных сверточных кодов, их параметры алгебраически связаны с параметрами недвоичных циклических кодов.

Рассмотрим рекурсивный сверточный кодер, построенный в виде недвоичного регистра сдвига с обратными связями. Такой кодер реализует пакетную обработку данных по *m* символов из $GF(q)$ или, что эквивалентно, по одному символу из $GF(q^m)$.

Соответствующая схема турбокодера, построенного на алгебраических рекурсивных сверточных кодах, с обработкой элементов из $GF(q^m)$

представлена на рис. 5.4.

Устройство, схема которого представлена на рис. 5.4., работает следующим образом.

На вход кодера поступает информационная последовательность с символами из *GF(q)*. Во входном буфере символы из *GF(q)* преобразуются в символы из $H \subseteq GF(q^m)$, и, как в теореме 3.2, сопоставляются символам из $GF(q^m)$. Полученные символы из $GF(q^m)$ поступают на вход первого несистематического алгебраического рекурсивного сверточного кодера и, через перемежитель, на вход второго кодера. Выходные символы из $GF(q^m)$ поступают на мультиплексор где формируется кодовое слово турбокода с элементами из $GF(q^m)$. Выходной буфер преобразует символы из $GF(q^m)$ в кодовые символы из *GF(q)*. Справедлива следующая теорема.

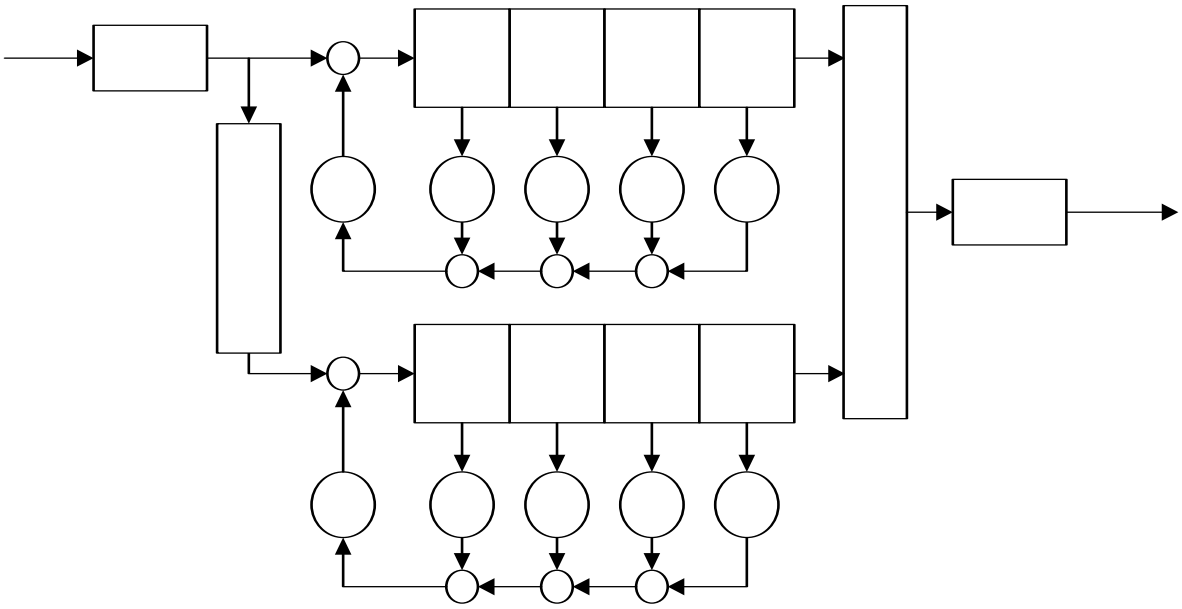


Рис. 5.4. Турбокодер на алгебраических несистематических рекурсивных сверточных кодах с обработкой элементов из $GF(q^m)$

Теорема 5.1. Турбокодер, построенный на алгебраических несистематических рекурсивных сверточных кодах, имеет скорость кодирования:

. (5.3)

Доказательство. Каждый сверточный кодер в схеме турбокодера на рис. 5.4. построен в виде цепи регистров с обратными связями, где отводы в цепи обратной связи задаются коэффициентами проверочного многочлена $h(x)$ недвоичного рекурсивного несистематического циклического (N, K, D) кода над $GF(q^m)$. Для каждых K введенных информационных символов из $GF(q^m)$ кодовое слово на выходе каждого сверточного кодера суть кодовое слово циклического (N, K, D) кода над $GF(q^m)$.

Если на вход каждого кодера подавать непрерывную последовательность, то, по теоремам раздела 3, полученное отображение

суть правило сверточного кодирования с параметрами: $\nu = K \cdot k^0$, $n^0 = m$, $k = (K + 1) \cdot k^0$, $n = (K + 1) \cdot n^0$, $R = k^0/m$, $d_\infty \geq D$. Следовательно, для каждых k^0 входных символов из $GF(q)$ на выходе каждого сверточного кодера будет сформировано $n^0 = m$ символов из $GF(q)$ или, что эквивалентно, по одному символу из $GF(q^m)$. В мультиплексоре эти символы поочередно считываются, а затем в выходном буфере преобразуются в последовательность из $2 \cdot m$ символов из $GF(q)$. Следовательно, для каждых k^0 входных символов из $GF(q)$ турбокодером будет сформировано $2 \cdot m$ символов из $GF(q)$, что и задает скорость турбокода по выражении (5.3).

Следствие. Если $k^0 = 1$, то, по теореме 3.1, имеем рекурсивный несистематический сверточный код с параметрами: $\nu = K$, $n^0 = m$, $k = K + 1$, $n = (K + 1) \cdot n^0$, $R = 1/m$, $d_\infty \geq D$. Соответствующий турбокодер имеет скорость кодирования $R_{TK} = 1/(2 \cdot m)$, что соответствует обобщению результата леммы 5.2.

Другую схему турбокодера построим на алгебраических несистематических сверточных кодах с обработкой символов из $GF(q)$ (рис. 5 .5).

Следует отметить особенности работы устройств, схемы которых приведены на рис. 5.4. и 5.5. Они отличаются способом обработки входных данных. Действительно, кодер, приведенный на рис. 5.5., реализует посимвольную обработку элементов из $GF(q)$. Кодер, приведенный на рис. 5. 4., реализует посимвольную обработку элементов из $GF(q^m)$. Если при этом выполняются условия теорем раздела 3, то последовательности на выходе соответствующих устройств совпадут. Приведем пример.

Пример. Рассмотрим РС код (7, 3, 5) над $GF(2^3)$ и построеные на его основе алгебраические рекурсивные сверточные коды. На рис. 5.6. приведена соответствующая схема турбокодера с обработкой элементов из $GF(2^3)$, а на рис. 5.7. – с обработкой двоичных элементов.

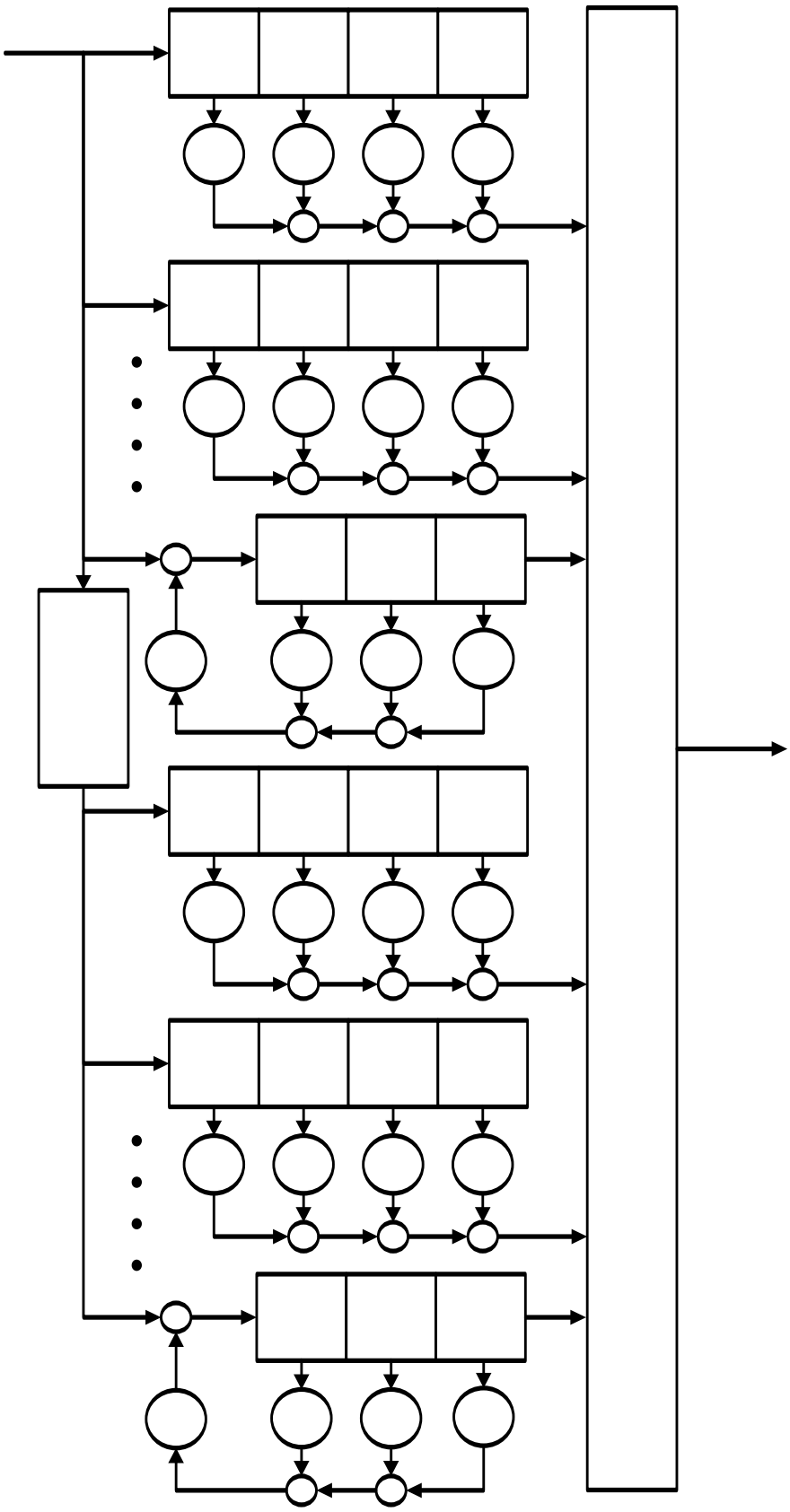


Рис. 5.5. Турбокодер на алгебраических несистематических рекурсивных сверточных кодах с обработкой элементов из *GF(q)*

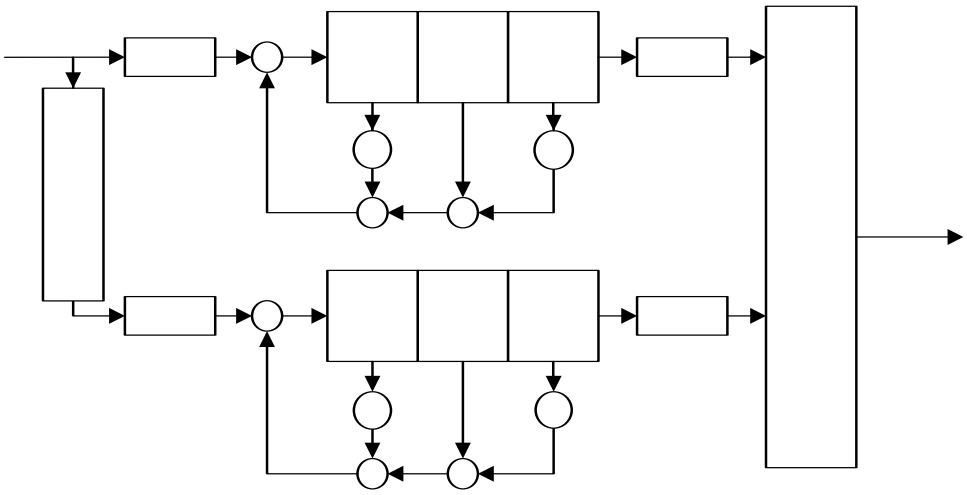


Рис. 5.6. Пример турбокодера на алгебраических рекурсивных сверточных кодах с обработкой символов из $GF(2^3)$

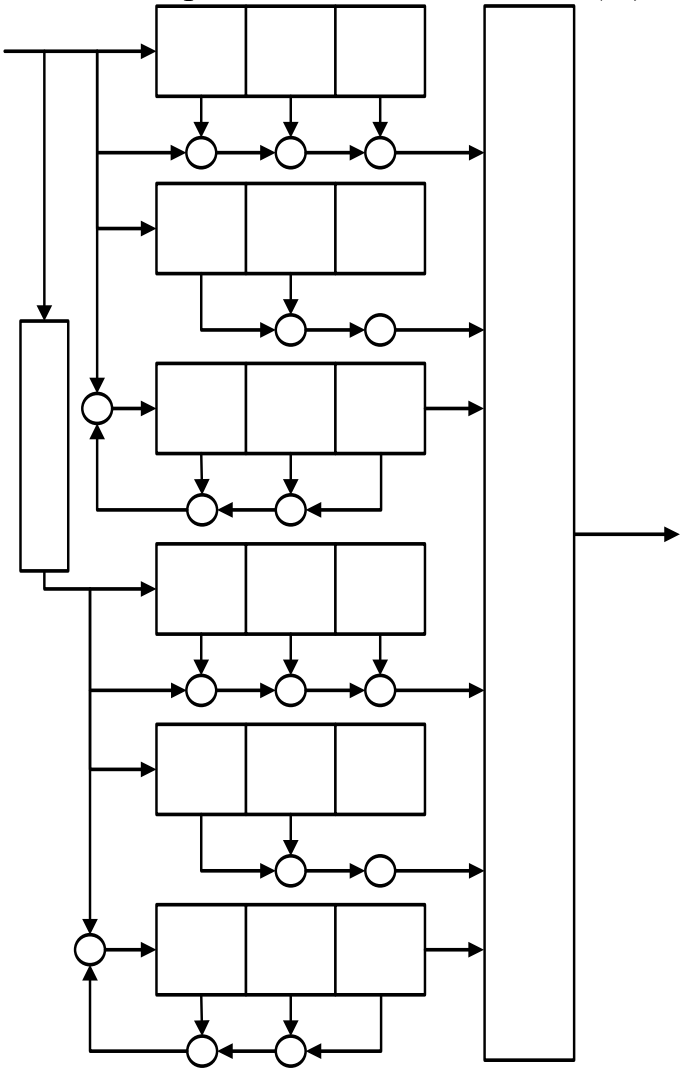


Рис. 5.7. Пример турбокодера на алгебраических рекурсивных сверточных кодах с обработкой двоичных символов

5.3. Турбокоды на основе алгебраически заданных систематических рекурсивных сверточных кодов

Для построения турбокодов на алгебраических систематических рекурсивных сверточных кодах воспользуемся результатами теорем раздела 3. Доказанные теоремы дают мощный механизм построения алгебраических систематических рекурсивных сверточных кодов, их параметры алгебраически связаны с параметрами недвоичных циклических кодов.

Рассмотрим рекурсивный сверточный кодер, построенный в виде недвоичного регистра сдвига с обратными связями (см. раздел 3). Такой кодер реализует пакетную обработку данных по m символов из $GF(q)$ или, что эквивалентно, по одному символу из $GF(q^m)$.

Схема турбокодера, построенного на алгебраических рекурсивных сверточных кодах с обработкой элементов из $GF(q^m)$ представлена на рис. 5.8 . Устройство, схема которого представлена на рис. 5.8, работает следующим образом.

На вход кодера поступает информационная последовательность с символами из $GF(q)$. Во входном буфере символы из $GF(q)$ преобразуются в символы из $H \subseteq GF(q^m)$, и, как в теореме раздела 3, сопоставляются символам из $GF(q^m)$. Полученные символы из $GF(q^m)$ поступают на вход первого алгебраического систематического рекурсивного сверточного кодера и, через перемежитель, на вход второго кодера. Информационные символы из $GF(q)$ и проверочные символы с первого и второго сверточных кодеров из $GF(q^m)$ поступают на мультиплексор, где формируется кодовое слово турбокода с элементами из $GF(q^m)$. Выходной буфер преобразует символы из $GF(q^m)$ в кодовые символы из $GF(q)$.

Для определения параметров построенного таким образом турбокода сформулируем и докажем следующую теорему.

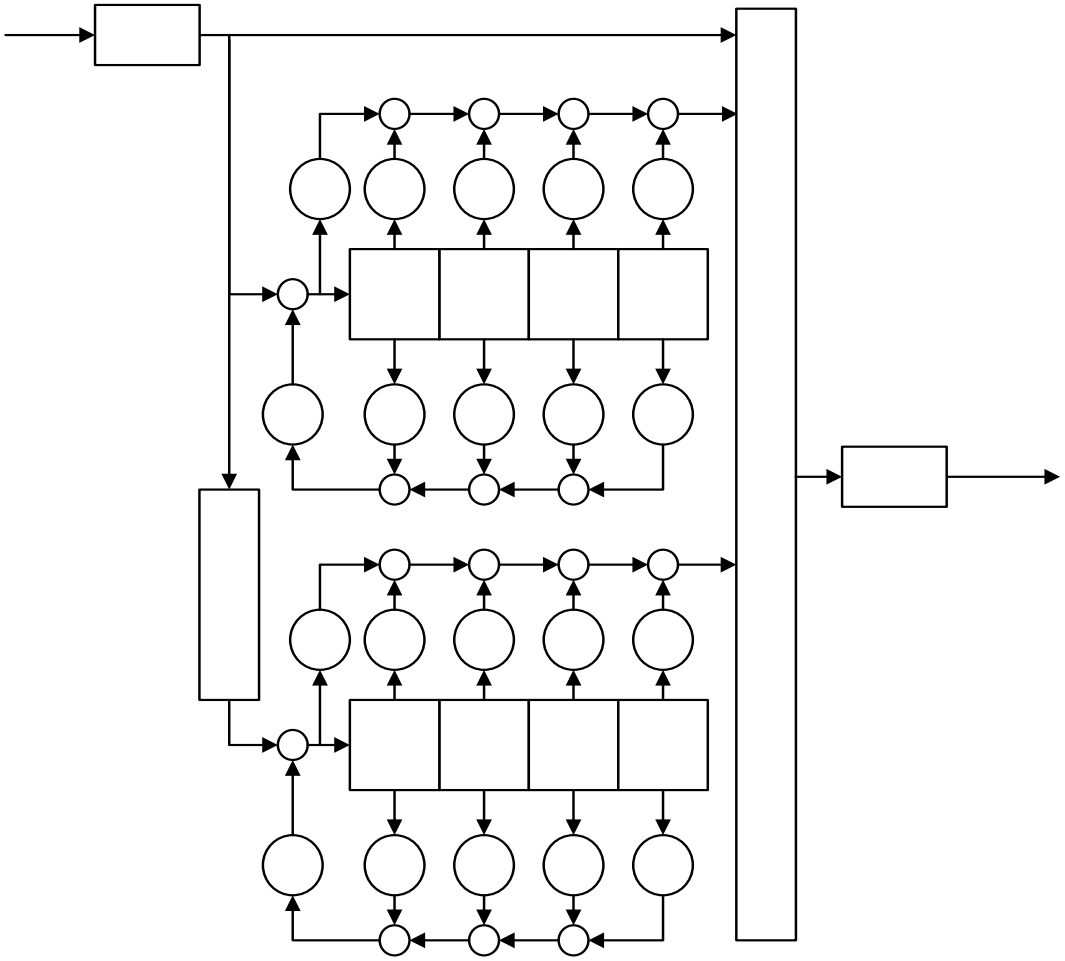


Рис. 5.8. Общая схема турбокодера с использованием алгебраических систематических рекурсивных сверточных кодов с обработкой элементов из $GF(q^m)$

Теорема 5.2. Турбокодер, построенный на алгебраических несистематических рекурсивных сверточных кодах, имеет скорость кодирования:

. (5.4)

Доказательство. Каждый сверточный кодер в схеме турбокодера на рис. 5.8 построен в виде цепи регистров с обратными связями, где отводы в цепи обратной связи задаются коэффициентами порождающего многочлена $g(x)$ недвоичного рекурсивного систематического циклического (N, K, D) кода над $GF(q^m)$. Для каждых $N - K$ введенных информационных символов из $GF(q^m)$ кодовое слово на выходе каждого сверточного кодера суть кодовое слово циклического (N, K, D) кода над $GF(q^m)$. Если на вход каждого кодера подавать непрерывную последовательность, то, по теоремам 3.4-3.5, полученное отображение суть правило сверточного кодирования с параметрами: $\nu = (N-K)\cdot K\cdot \lfloor \log_q H \rfloor$, $k^0 = K\cdot \lfloor \log_q H \rfloor$, $n^0 = ((N-K)\cdot m + K\cdot \lfloor \log_q H \rfloor)$, $k = (N-K+1)\cdot K\cdot \lfloor \log_q H \rfloor$, $n = (N-K+1)\cdot((N-K)\cdot m + K\cdot \lfloor \log_q H \rfloor)$, $R = K\cdot \lfloor \log_q H \rfloor / ((N-K)\cdot m + K\cdot \lfloor \log_q H \rfloor)$, $d_\infty \geq D$. Следовательно, для каждых $k^0 = K\cdot \lfloor \log_q H \rfloor$ входных символов из

GF(q) на выходе каждого сверточного кодера будет сформировано $n^0=((N–K)\cdot m+K\cdot \lfloor \log_q H \rfloor)$ символов из *GF(q)*.

В мультиплексоре поочередно считываются информационные и проверочные символы, а в выходном буфере считанные символы из *GF(q)* отображаются в поле из $GF(q^m)$. Следовательно, для каждых $k^0=K\cdot \lfloor \log_q H \rfloor$ входных символов из *GF(q)* турбокодером будет сформировано $(2\cdot n^0-k^0) = (2\cdot (N–K)\cdot m+K\cdot \lfloor \log_q H \rfloor)$ символов из *GF(q)*, что и задает скорость турбокода: $R_{TK} = k^0/(2\cdot n^0-k^0) = K\cdot \lfloor \log_q H \rfloor/(2\cdot (N–K)\cdot m+K\cdot \lfloor \log_q H \rfloor)$.

Следствие 1. Если $k^0 = K$, $n^0 = N$, то имеем рекурсивный систематический сверточный код с параметрами: $\nu =(N–K)\cdot K$, $k^0=K$, $n^0=N$, $k =(N-K+1)\cdot K$, $n=(N-K+1)\cdot N$, $R=K/N$, $d_\infty \geq D$. Соответствующий турбокодер имеет скорость кодирования $R_{TK} = k^0/(2\cdot n^0-k^0) = K/(2\cdot N–K)$, т.е. скорость турбокода будет определяться исключительно скоростью циклического *(N, K, D)* кода.

Следствие 2. Если $k^0 = K = 1$, $n^0 = N$, то имеем рекурсивный систематический сверточный код с параметрами: $\nu =N–1$, $k^0=1$, $n^0=N$, $k=N$, $n =N^2$, $R=1/N$, $d_\infty \geq D$. Соответствующий турбокодер имеет скорость кодирования $R_{TK} = k^0/(2\cdot n^0-k^0) = 1/(2\cdot N–1)$, что соответствует обобщению результата леммы 5.1.

Другую схему турбокодера на алгебраических систематических сверточных кодах построим с обработкой символов из *GF(q)* (рис. 5.9).

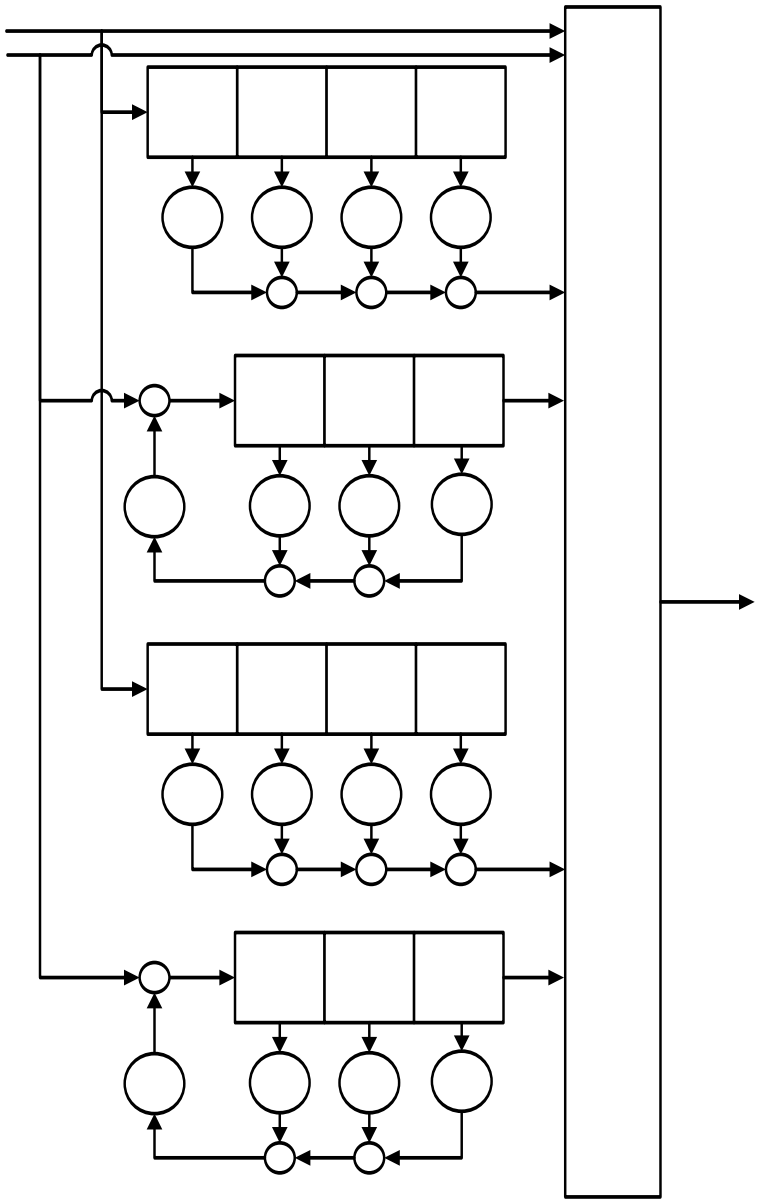


Рис. 5.9. Общая схема турбокодера с использованием алгебраических рекурсивных систематических сверточных кодов с обработкой элементов из *GF(q)*

Приведем пример. Зафиксируем конечное поле $GF(2^2)$, построенное по кольцу многочленов $\{0 = \alpha^{-\infty}, 1 = \alpha^0, x = \alpha^1, x + 1 = \alpha^2\}$ с операциями, по модулю $G(x) = x^2 + x + 1$. Зафиксируем (3, 2, 2) код РС с порождающим многочленом $g(x) = x + \alpha^2$ и алгебраический рекурсивный систематический сверточный код с параметрами $\nu = 2$, $k^0 = 1$, $n^0 = 2$, $k=3$, $n = 6$, $R = 1 / 2$, $d_\infty \geq 2$. Построим турбокодер с обработкой двоичных символов, схема кодера приведена на рис. 5.10.

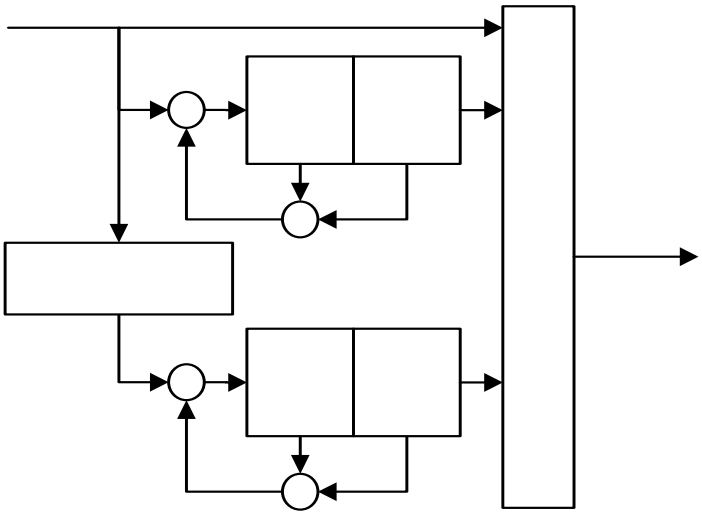


Рис. 5.10. Схема турбокодера с использованием алгебраического рекурсивного систематического сверточного кода с обработкой двоичных элементов

Воспользуемся результатами теоремы 5.2. Скорость алгебраически заданного турбокода определяется выражением (5.4) и в данном случае она

равна , что соответствует также результату леммы 5.1.

Приведенный пример наглядно демонстрирует возможности предложенного подхода турбокодирования с использованием алгебраических рекурсивных сверточных кодов.

Для использования предложенного подхода разработаем практический алгоритм построения турбокодов с требуемыми параметрами.

Предположим, что в параллельной каскадной схеме с требуемой скоростью R_{TK} необходимо использовать сверточные коды с фиксированными k^0 и n^0 параметрами. Тогда для формирования порождающих и/или проверочных многочленов сверточного кода необходимо выбрать соответствующий недвоичный циклический код и вариант его использования. Подробно эти вопросы рассмотрены в разделе 3.

Общая схема предлагаемого алгоритма построения турбокодов на алгебраических рекурсивных сверточных кодах приведена на рис. 5.11. Алгоритм состоит из последовательности следующих шагов.

ШАГ 1. Ввод требуемой скорости турбокода R_{TK}, требуемых k^0 и n^0 параметров соответствующих сверточных кодов, ввод мощности алфавита кодовых символов q.

ШАГ 2. Расчет скорости $R_{ск}$ рекурсивных сверточных кодов

ШАГ 3. Выбор способа обработки кодовых символов.

ШАГ 4. Выбор варианта построения рекурсивных сверточных кодов, расчет параметров соответствующего циклического кода над $GF(q^m)$, формирование порождающих многочленов и построение схемы кодера сверточного кода над $GF(q)$ (по отдельному алгоритму, см. раздел 3).

ШАГ 5. Построение параллельной каскадной схемы с алгебраическими рекурсивными сверточными кодами.

Рассмотрим выполнение каждого шага предложенного алгоритма отдельно.

Рассмотрим параллельные каскадные схемы турбокодирования на алгебраических несистематических рекурсивных сверточных кодах. Предположим, что необходимо построить турбокод со скоростью R_{TK}. Тогда скорость R_{CK} соответствующих сверточных кодов можно получить из выражения (5.3) в тереме 5.1:

(5.5)

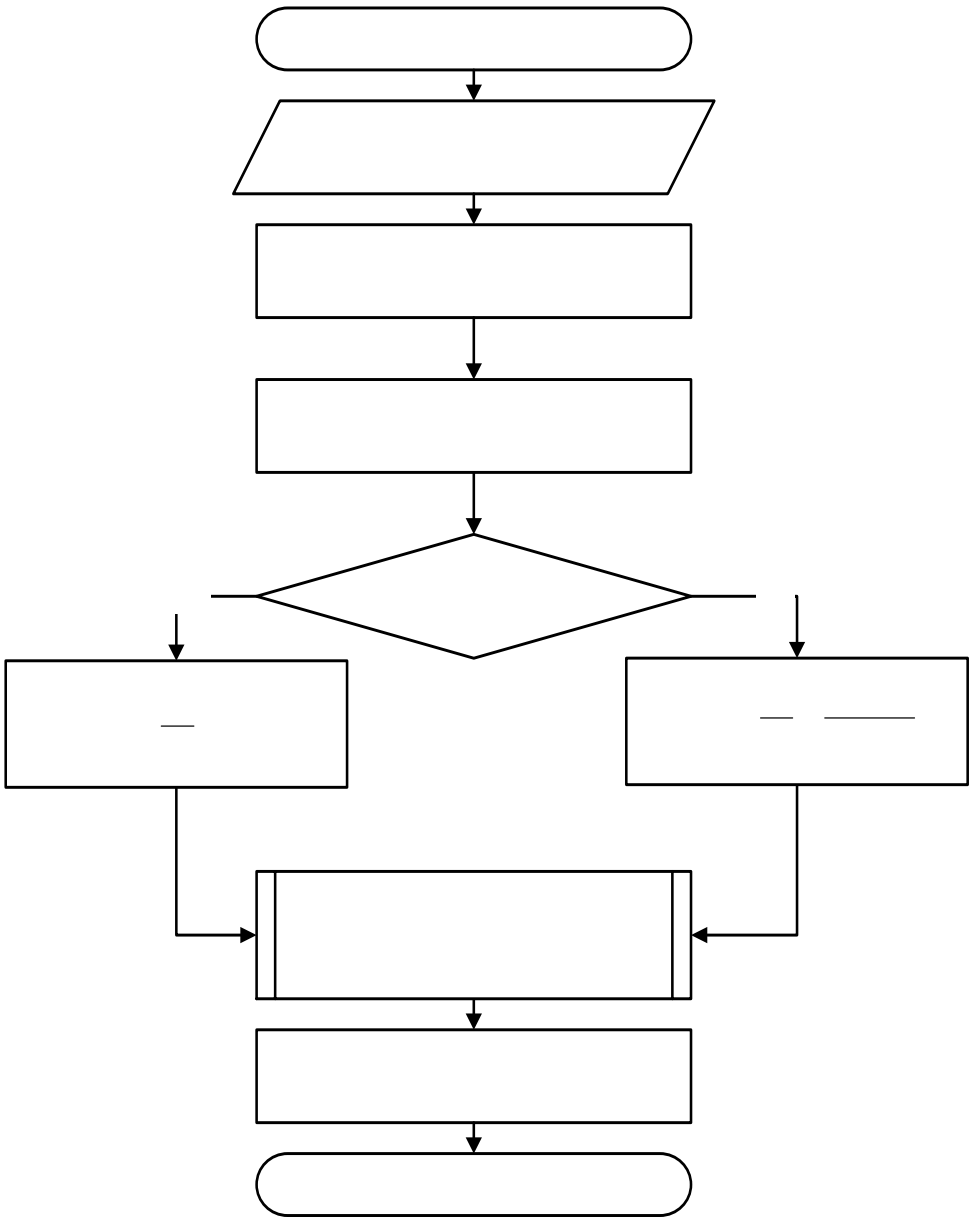


Рис. 5.11. Общая схема алгоритма построения турбокодов с использованием алгебраических рекурсивных сверточных кодов

После расчета скорости $R_{ск}$ сверточных кодов выполняется процедура алгебраического построения рекурсивных сверточных кодов (см. раздел 3).

Расчет кодовых параметров соответствующего циклического кода выполняется по соответствующим аналитическим выражениям.

Рассмотрим параллельные каскадные схемы турбокодирования на алгебраических систематических рекурсивных сверточных кодах. Тогда скорость $R_{ск}$ соответствующих сверточных кодов можно получить из выражения (5.4) в тереме 5.2:

.

Раскроем скобки и приведем подобные, получим:

.

Разделим обе части уравнения на n^0, получим

,

откуда имеем:

. (5.6)

После расчета скорости R_{CK} сверточных кодов выполняется процедура алгебраического построения рекурсивных сверточных кодов. Расчет кодовых параметров соответствующего циклического кода выполняется по соответствующим аналитическим выражениям.

На пятом шаге разработанного алгоритма строится параллельная каскадная схема с полученными алгебраическими рекурсивными сверточными кодами. При этом учитывается вид сверточного кодера (систематический или несистематический), и способ обработки кодовых символов (посимвольно, элементами из $GF(q)$, или посимвольно, элементами из $GF(q^m)$). Особенности построения рекурсивных сверточных кодов определяются условиями теорем раздела 3, а параллельных каскадных схем - теоремами 5.1-5.2.

5.4. Алгоритм итеративного декодирования параллельных каскадных кодов

Проведем исследование алгоритма итеративного декодирования турбокода, состоящего из двух составляющих кодов.

Декодирование турбокода осуществляется итеративным турбодекодером [41-43, 133], схема которого представлена на рис. 5.12.

Рис. 5.12. Схема итеративного турбодекодера с двумя составляющими декодерами систематических сверточных кодов

Обозначим последовательности принятых информационных и проверочных символов первого и второго кодеров как

, , ,

где ;

;

;

, , – случайные величины, имеющие нормальное распределение с нулевым средним значением и дисперсией .

В отличие от декодера каскадного кода, в котором составляющие декодеры принимают независимые решения, составляющие декодеры итеративного турбодекодера обмениваются мягкими решениями с целью уточнения результата декодирования. Мягкие решения первого декодера после перемежения используются в качестве априорной информации для второго декодера. Мягкие решения второго декодера после деперемежения используются как априорная информация для первого декодера. Подобный обмен мягкими решениями между декодерами назовем итерацией.

Окончательное решение турбодекодером принимается после некоторого количества итераций с использованием принятой информационной последовательности и мягких решений первого и второго составляющих декодеров.

Информационный символ может принимать значения 0 и 1 с априорной вероятностью и , причем .

Каждая итерация состоит из двух фаз. В первой фазе итерации декодирование производится первым декодером. Во второй фазе итерации декодирование производится вторым декодером с учетом мягких решений первого декодера.

На первой фазе декодер использует априорные вероятности и принятые последовательности , для вычисления апостериорных вероятностей , :

, (5.7)

где .

Пусть – отношение апостериорных вероятностей информационных символов после фазы итерации (,), которое будем называть апостериорным отношением правдоподобия информационного символа или мягкими решениями декодера.

Выражение для имеет вид

.

Используя выражение (5.7), получим

. (5.8)

Обозначим

.

Будем называть априорным отношением правдоподобия информационного символа или априорной информацией [41-43]. На первой фазе первой итерации априорная информация о передаваемых символах обычно отсутствует, поэтому , а .

Обозначим

.

Будем называть внутренним отношением правдоподобия информационного символа или внутренней информацией [41-43]. В течение всех итераций .

Обозначим

(5.9)

Будем называть внешним отношением правдоподобия информационного символа или внешней информацией [41-43].

Для второй фазы -той итерации выражение (5.8) приведем к виду

.

где – внутренняя информация после перемежения;

– внешняя информация после перемежения.

Для случайного перемежителя большой длины можно считать, что перемеженная версия последовательности – не будет зависеть от , которая используется в выражении (5.9), поэтому внешнюю информацию одного из составляющих декодеров также можно считать независимой от внешней информации другого составляющего декодера. Это позволяет использовать внешнюю информацию одного из декодеров в качестве априорной информации для другого декодера.

Жесткие решения принимаются после некоторого количества итераций с использованием правила

.

В общем случае турбодекодер может содержать и большее количество составляющих декодеров. Из-за чрезмерного увеличения сложности декодирования в настоящее время не используют турбодекодеры с количеством составляющих декодеров больше двух.

Из анализа алгоритма итеративного декодирования турбокодов следует , что для реализации итеративного турбодекодера необходимы алгоритмы мягкого декодирования составляющих кодов, позволяющие оценить апостериорные вероятности информационных символов. Проведем исследование известных алгоритмов мягкого декодирования сверточных кодов.

5.5. Алгоритмы мягкого декодирования составляющих турбокод сверточных кодов

Известны следующие алгоритмы мягкого декодирования сверточных кодов: MAP (maximum a posteriori probability), log-MAP, min-log-MAP и SOVA (soft output Viterbi algorithm). Рассмотрим данные методы мягкого декодирования систематического сверточного кода.

Описание MAP алгоритма для декодирования систематических сверточных кодов представлено в [43, 52, 54].

Пусть информационный символ ассоциируется с переходом кодера из состояния в состояние . Будем считать, что последовательность информационных символов состоит из независимых символов, которые могут принимать значения 0 и 1 с априорной вероятностью и , . Начальное и конечное состояния кодера – нулевые. Последовательность необходима для установки кодера в конечное нулевое состояние.

Определим принятую последовательность

,

где – принятые символы во время ;

, ;

, – независимые случайные величины, имеющие нормальное распределение с нулевым средним значением и дисперсией .

Определим отношение правдоподобия информационного символа следующим образом

,

где – апостериорная вероятность , .

Правило принятия жестких решений определим как

.

Отношение правдоподобия представим следующим образом [43, 52, 54]:

, (5.10)

где – прямая метрика состояния во время ;

– обратная метрика состояния во время ;

– метрика ветви.

Из проведенного анализа MAP алгоритма следует, что в процессе получения мягких решений необходимо произведение большого числа умножений, что является недостатком этого алгоритма.

Один из путей уменьшения сложности декодирования MAP алгоритма – это осуществление вычислений в логарифмической области [43, 52, 54]. В этом случае все операции произведения и деления заменяются операциями сложения и вычитания, которые реализовать значительно проще. Операция сложения преобразуется в E операнд, который определим следующим образом

, (5.11)

где , , .

Функция быстро убывает и имеет максимальное значение (для). Если значением функции можно пренебречь и установить .

Введем обозначения: , , ,

. Преобразуем выражение с учетом введенных обозначений:

, (5.12)

, (5.13)

, (5.14)

,

где – константа;

;

– априорная информация log-MAP декодера;

.

Выражение (5.12) с учетом введенных обозначений примет вид

,

где – внешняя информация log-MAP декодера.

Таким образом, осуществление вычислений в логарифмической области позволяет заменить операции умножения и деления операциями сложения и вычитания. При этом операция сложения заменяется E операндом, который можно представить с помощью операции нахождения минимального значения и некоторой поправки, которая может быть задана таблично. Характеристики log-MAP декодера такие же, как и у MAP декодера.

Упростим выражения (5.12 – 5.14), устанавливая и заменив E операнд функцией нахождения минимального значения для получения min-log-MAP алгоритма:

,

,

.

Min-log-MAP алгоритм является субоптимальным алгоритмом по сравнению с log-MAP алгоритмом.

Использование составляющих min-log-MAP декодеров в турбодекодере уменьшает сложность декодирования, однако приводит к увеличению вероятности ошибки декодирования и уменьшению энергетического выигрыша от кодирования на 0,3ч0,6 дБ [43, 52, 54].

Алгоритм SOVA представляет собой модификацию алгоритма Витерби [133-135]. Пусть имеется последовательность наблюдений дискретного по времени марковского процесса с конечным числом состояний, наблюдаемого в канале без памяти с АБГШ. Найдем последовательность состояний, , для которой апостериорная вероятность максимальна.

Пусть – метрика пути, ведущего в состояние . Метрика может быть вычислена рекурсивно

. (5.15)

Введем обозначения , . Представим (5.15) с учетом введенных обозначений как

,

где , , – константа, которой можно пренебречь.

Для реализации итеративного турбодекодера необходимо получение мягких решений для каждого информационного символа, а не для всей принятой последовательности [41, 42]. В решетчатой диаграмме двоичного сверточного кода с существуют два пути, ведущие в состояние . Будем называть эти пути сливающимися путями в состоянии .

Путь, имеющий большую метрику, обозначим и назовем выжившим путем, обозначая его метрику . Путь с меньшей метрикой, который обозначим , будет отброшен алгоритмом Витерби, поэтому назовем этот путь отброшенным путем и обозначим его метрику . При этом метрика выжившего пути будет всегда больше, чем метрика отброшенного пути.

Логарифм отношения правдоподобия для выжившего пути [133-135]:

.

где – разностная метрика выжившего и отброшенного пути,

.

С выжившим путем сливается путей с индексами , которые будут отброшены алгоритмом Витерби. Их разностные метрики . Если информационный символ отброшенного пути с индексом равен информационному символу выжившего пути , то ошибки при определении не произойдет, если выбрать отброшенный путь. Таким образом, логарифм отношения правдоподобия для этого случая может быть принят равным бесконечности (считается, что решение абсолютно достоверно). Если же то логарифм отношения правдоподобия равен . Логарифм отношения правдоподобия для информационного символа имеет вид [134-136]

.

Таким образом, мягкое решение SOVA алгоритма представляет собой произведение жесткого решения алгоритма Витерби на наименьшую из разностных метрик. Преимущество турбодекодера с составляющими SOVA декодерами – наименьшая сложность декодирования. Энергетический проигрыш –наибольший, который составляет 0,5ч1,0 дБ по сравнению с MAP или log-MAP составляющими декодерами [51, 56, 134-136]. Для оценки возможности применения рассмотренных алгоритмов в итеративном турбодекодере проведем оценку их сложности.

5.6. Сложность итеративного декодирования параллельных каскадных кодов

Сложность декодирования log-MAP, min-log-MAP и SOVA составляющих декодеров приведена в табл. 5.1, а сложность декодирования турбокодов с двумя составляющими декодерами определяется выражением [133]:

,

где – количество итераций турбодекодера;

, – сложность декодирования составляющих декодеров.

Кривые зависимости сложности декодирования турбокодов от количества элементов памяти составляющих сверточных кодов для турбодекодера с двумя составляющими декодерами (log-MAP, min-log-MAP и SOVA) и 8 итераций представлены на рис. 5.13.

Из рис. 5.13 следует, что с увеличением сложность декодирования значительно возрастает. Поэтому в настоящее время не используют турбокоды с .

Таблица 5.1

Сложность декодирования

Операции	log-MAP	min-log-MAP	SOVA
Сложение			
Умножение			
Нахождение минимального значения			
Обращение к таблице		–	–
Общее количество операций			

Рис. 5.13. Сложность декодирования сверточных кодов и турбокодов, 8 итераций

На рис. 5.14 представлены кривые зависимости сложности декодирования турбодекодера от для сверточных кодов.

Рис. 5.14. Сложность декодирования турбодекодера, 8 итераций, – скорость турбокода, – скорость составляющих сверточных кодов

Из анализа рис. 5.14. следует, что сложность декодирования возрастает не только с увеличением , но также и с увеличением .

Таким образом, видимым путем устранения недостатков турбокодов является использование в качестве составляющих турбокод кодов

алгебраически заданных рекурсивных сверточных кодов, разработанных в разделе 3, с большим значением и , что приведет к повышению минимального расстояния турбокода и позволит выбирать скорость турбокодирования в широких пределах без применения выкалывания. Однако практическая реализация алгоритма итеративного декодирования турбокодов с большим значением затруднена из-за высокой сложности декодирования, поскольку рассмотренные выше алгоритмы мягкого декодирования используют только решетчатое представление сверточного кода.

Поэтому необходима разработка метода (и алгоритма) мягкого декодирования алгебраических рекурсивных сверточных кодов, учитывающего алгебраическую (а не решетчатую) структуру сверточных кодов, что приведет к уменьшению сложности декодирования турбокодов с большим значением .

5.7. Метод мягкого декодирования алгебраически заданных сверточных кодов

Проведенные исследования в подразделах 5.1 – 5.3 показали, что для реализации процедуры итеративного декодирования турбокодов необходимы алгоритмы мягкого декодирования, позволяющие оценить апостериорную вероятность каждого из информационных символов кодового слова, т.е. реализующие посимвольное правило принятия решений с минимизацией средней вероятности ошибки символа. Однако существующие алгоритмы мягкого декодирования сверточных кодов даже при современном уровне развития микроэлектроники неприменимы для использования в итеративном декодере турбокодов с большим значением . Поэтому будем использовать алгебраический подход для декодирования алгебраически заданных сверточных кодов.

В разделе 4 показано, что бесконечное кодовое слово сверточного кода, алгебраически заданного через порождающий многочлен циклического кода, состоит из бесконечной суммы кодовых слов циклического кода, что позволяет свести декодирование сверточного кода к декодированию последовательности кодовых слов циклического кода. Таким образом, мягкое посимвольное декодирование сверточного кода сведем к мягкому посимвольному декодированию последовательных наборов кодовых слов циклического кода. Далее рассмотрим алгоритмы мягкого декодирования блоковых кодов.

Необходимо отметить, что посимвольное декодирование блоковых кодов можно осуществлять и с помощью рассмотренных ранее алгоритмов мягкого декодирования: MAP, log-MAP, min-log-MAP и SOVA, если использовать решетчатое представление блоковых кодов. Однако кодовая решетка блоковых кодов, в общем случае, характеризуется большим числом состояний и нерегулярностью структуры, что значительно увеличивает сложность декодирования.

Наиболее важными среди алгоритмов, в которых производится минимизация средней вероятности ошибки символа являются алгоритм декодирования по апостериорной вероятности (APP алгоритм), предложенный Месси [20], оптимальный алгоритм посимвольного декодирования Хартмана-Рудольфа [20] и алгоритм Велдона [20].

APP алгоритм Месси представляет собой обобщение порогового декодирования, позволяющее принимать мягкие решения. Этот алгоритм применим для ограниченного класса кодов, допускающих ортогонализацию.

Оптимальный алгоритм посимвольного декодирования Хартмана-Рудольфа применим к любому групповому коду. Он включает в себя вычисление решающей функции на каждом кодовом слове дуального кода, т. е. на множестве всех проверочных уравнений. Поэтому сложность алгоритма увеличивается с уменьшением скорости кода.

Алгоритм Гринбергера [20] является упрощением алгоритма Хартмана-Рудольфа. Он включает предварительное упорядочение принятых символов по убыванию их достоверности. Далее алгоритм Хартмана-Рудольфа применяется к наиболее достоверным символам, затем к $к$+1, $к$+2 и т.д. до тех пор, пока решающие функции, соответствующие первым $к$ символам, не изменятся существенным образом. При небольших отношениях сигнал/шум эта процедура требует небольшого числа итераций и поэтому лишь небольшого числа слов дуального кода. Недостатком алгоритма Гринбергера является то, что требуемую форму проверочной матрицы (все элементы над крайней правой диагональю должны быть нулевыми) нельзя рассчитать заранее и что при декодировании нельзя использовать цикличность применяемого кода.

Стандартное применение алгоритма Велдона включает предварительное порождение m различных последовательностей, соответствующих каждому из m бит в демодуляторе, квантованном на уровней. Каждая из этих последовательностей декодируется декодером с жестким решением, и каждый символ оценивается с помощью взвешенной суммы соответствующих символов в каждой декодированной последовательности. Хотя реализация алгоритма Велдона довольно проста, но его характеристики хуже, чем у других методов декодирования.

Таким образом, в дальнейшем для мягкого декодирования цикличеких кодов будем использовать алгоритм Гринбергера, поскольку он не требует использования всех проверочных уравнений, что позволит уменьшить сложность декодирования, по сравнению с оптимальным алгоритмом Хартмана-Рудольфа при сохранении достаточно высоких характеристик.

Поскольку алгоритм Гринбергера является упрощением алгоритма Хартмана-Рудольфа, то вначале рассмотрим алгоритм Хартмана-Рудольфа.

Пусть - -тое кодовое слово дуального кода и -тый символ этого кодового слова. Определим отношение правдоподобия для -того принятого символа как

.

Положим

.

Теперь правило декодирования можно записать следующим образом. Полагаем, что в кодовом слове -тый символ тогда и только тогда, когда

,

где при и в остальных случаях.

В качестве примера рассмотрим циклический (7, 4) код Хэмминга, для которого матрица имеет вид

.

Восемью словами дуального кода являются все линейные комбинации строк матрицы . Они имеют вид (0000000), (0010111), (010110), (1001011), (1011100), (1010101), (1110010). Используя эти слова, легко вычислить решающую функцию для последнего символа :

Таким образом, символ на позиции 6 полагается равным нулю, если , и равным 1, если . Для декодирования остальных символов следует циклически сдвинуть кодовое слово и заново вычислить .

Алгоритм Гринбергера использует предположение, что информативность символов, принятых с малой достоверностью, мала, так что содержащие их проверочные уравнения можно не рассматривать. Алгоритм Гринбергера состоит из следующих шагов.

1. Упорядочим принятые символы в соответствии с их достоверностью, помещая наиболее достоверные символы в начало списка.

2. Упорядочим столбцы проверочной матрицы таким образом, чтобы левее всего находился столбец, соответствующий наиболее достоверному символу, следующим шел столбец, соответствующий второму по достоверности символу, и т.д.

3. Используя элементарные операции на строками проверочной матрицы, добьемся того, чтобы все элементы над крайней правой диагональю стали нулевыми. Первая строка полученной матрицы будет одержать нули в наименее достоверных позициях, вторая строка – в

наименее достоверных позициях и т.д.

4. Используя первую строку модифицированной матрицы , применяем алгоритм Хартмана-Рудольфа для оценки каждого из принятых символов. При этом будем использовать, конечно, всего два кодовых слова дуального кода: нулевое слово и слово, соответствующее первой строке.

5. Рассмотрим теперь первые две строки матрицы . Эти две строки порождают четыре кодовых слова дуального кода, два из которых были порождены ранее, а два других являются новыми и содержат вклады от

-го наиболее достоверного символа. Используем эти два новых символа для уточнения оценок каждого из принятых символов.

6. Продолжим аналогичным образом, добавляя каждый раз по одной новой строке и уточняя каждую из оценок. Остановимся после того, как оценки проявят тенденцию к стабилизации, или после заранее заданного числа итераций.

7. Используя полученное множество оценок, выберем наиболее достоверных символов, образующих информационное множество, и породим кодовое слово, произведя операцию кодирования.

8. Переупорядочим символы и доставим полученное слово потребителю.

Алгоритм итеративного декодирования турбокодов представим в виде последовательности следующих шагов.

Шаг 1. Прием кодового слова турбокода. Выделение последовательности информационных символов и последовательностей проверочных символов составляющих сверточных кодов.

Шаг 2. Учет мягких решений второго составляющего декодера на всех итерациях, кроме первой (на первой итерации вместо последовательности мягких решений используется нулевая последовательность). Преобразование кодового слова первого составляющего сверточного кода в последовательность кодовых слов циклического кода. Мягкое декодирование последовательности кодовых слов циклического кода (алгоритмы Хартмана-Рудольфа, Гринбергера). Преобразование последовательности кодовых слов циклического кода в кодовое слово первого составляющего сверточного кода .

Шаг 3. Перемежение мягких решений первого составляющего декодера и информационной последовательности.

Шаг 4. Учет мягких решений первого составляющего декодера. Преобразование кодового слова второго составляющего сверточного кода в последовательность кодовых слов циклического кода. Мягкое декодирование последовательности кодовых слов циклического кода (алгоритмы Хартмана-Рудольфа, Гринбергера). Преобразование последовательности кодовых слов циклического кода в кодовое слово второго составляющего сверточного кода .

Шаг 5. Деперемежение мягких решений второго составляющего декодера.

Шаг 6. Если текущий номер итерации равен максимальному числу итераций, то осуществляется принятие жестких решений, в противном случае – переход к Шагу 2.

Таким образом, предложенный алгоритм итеративного декодирования турбокодов, на основе обобщенного представления кодовых слов составляющих сверточных кодов, позволяет свести декодирование сверточного кода к декодированию последовательности кодовых слов циклического кода.

Выводы

1. В результате проведенных исследований получили дальнейшее развитие методы синтеза параллельных каскадных сверточных конструкций (методы турбокодирования), отличающиеся от известных использованием алгебраически заданных рекурсивных сверточных кодов, что позволяет аналитически связать параметры турбокодов с параметрами алгебраически заданных рекурсивных сверточных кодов и синтезировать параллельные каскадные сверточные конструкции с заданными (конструктивными кодовыми) характеристиками.

2. Теоретическое обобщение процедур алгебраического построения параллельных каскадных кодов с использованием алгебраически заданных рекурсивных сверточных кодов позволило аналитически связать параметры турбокодов с параметрами алгебраически заданных сверточных кодов.

3. Разработанный алгоритм построения турбокодов на основе алгебраически заданных сверточных кодов позволяет за конечное число шагов сформировать параллельные каскадные схемы с требуемыми характеристиками. Разработанные схемы турбокодеров с использованием алгебраически заданных рекурсивных сверточных кодов позволяют реализовать непрерывную обработку входных данных посимвольно – элементами из $GF(q)$ или элементами из $GF(q^m)$.

4. Проведенное исследование алгоритма итеративного декодирования турбокодов показало, что для реализации итеративного турбодекодера необходимы алгоритмы мягкого декодирования составляющих кодов, позволяющие оценить апостериорные вероятности информационных символов.

5. Из проведенного исследования алгоритмов мягкого декодирования сверточных кодов: MAP, log-MAP, min-log-MAP и SOVA следует, что наибольшей сложностью декодирования обладают MAP и log-MAP алгоритмы. Дальнейшее упрощение log-MAP алгоритма приводит к min-log-MAP алгоритму, который является субоптимальным по сравнению с log-MAP алгоритмом. Энергетический проигрыш турбодекодера с составляющими min-log-MAP декодерами составляет 0,3ч0,6 дБ по сравнению с MAP или log-MAP декодерами. Наименьшую сложность декодирования обеспечивает SOVA алгоритм, являющийся модификацией алгоритма Витерби. Энергетический проигрыш турбодекодера с

составляющими SOVA декодерами наибольший, и составляет 0,5ч1,0 дБ по сравнению с MAP или log-MAP декодерами.

6. Устранить недостатки, характерные для турбокодов с малым кодовым ограничением, можно использованием в качестве составляющих турбокод кодов алгебраически заданных рекурсивных сверточных кодов с большим значением кодового ограничения и , что приведет к повышению минимального расстояния турбокода и позволит выбирать скорость турбокодирования в широких пределах без применения выкалывания. Однако практическая реализация алгоритма итеративного декодирования турбокодов с большим значением кодового ограничения затруднена из-за высокой сложности декодирования, поскольку алгоритмы мягкого декодирования сверточных кодов используют только решетчатое представление сверточного кода.

7. Обобщенное представление бесконечного кодового слова сверточного кода через бесконечную сумму последовательных наборов из кодовых слов циклического кода позволяет свести мягкое декодирование алгебраически заданного сверточного кода к мягкому декодированию последовательности кодовых слов циклического кода.

8. Получил дальнейшее развитие метод итеративного декодирования турбокодов с алгебраически заданными рекурсивными сверточными кодами, который отличается от известного обобщенным представлением бесконечного кодового слова сверточного кода через бесконечную сумму последовательных наборов из кодовых слов циклического кода, что позволяет за счет сведения декодирования сверточного кода к декодированию последовательности кодовых слов циклического кода декодировать турбокоды на основе алгебраически заданных сверточных кодов с большим числом элементов памяти (с высокими кодовыми характеристиками, высоким кодовым расстоянием).

РАЗДЕЛ 6
ДОСТОВЕРНОСТЬ ПЕРЕДАВАЕМОЙ ИНФОРМАЦИИ С ИСПОЛЬЗОВАНИЕМ АЛГЕБРАИЧЕСКИ ЗАДАННЫХ СВЕРТОЧНЫХ КОДОВЫХ КОНСТРУКЦИЙ И ОБОСНОВАНИЕ РЕКОМЕНДАЦИЙ ПО ИХ ПРИМЕНЕНИЮ

В данном разделе исследуются математические модели каналов связи, разработана методика оценки достоверности передаваемой информации, которая позволяет для заданных параметров математической модели канала связи с заданной погрешностью оценить вероятность ошибочного приема бита информации и соответствующий энергетический выигрыш от кодирования. Разработана имитационная модель системы передачи информации с использованием алгебраически заданных сверточных кодовых конструкций, которая позволяет оценить эффективность кодирования синтезированными сверточными кодовыми конструкциями. На основе полученных результатов проведенных исследований разработаны практические рекомендации по использованию синтезированных алгебраически заданных кодовых конструкций для повышения достоверности передаваемой информации.

6.1. Математические модели каналов связи

Рассмотрим систему передачи информации на уровне канального кодирования и модуляции, обоснуем методику количественной оценки достоверности передаваемой информации в рамках введенных в диссертационной работе ограничений и допущений. В соответствии с введенной в разделе 1 моделью системы передачи информации, передающая сторона содержит подсистему канального (помехоустойчивого) кодирования, имеющую дискретный вход и дискретный выход, за которым следует передатчик (модулятор), который преобразует по некоторому правилу информационные сообщения в сигналы, соответствующие характеристикам данного канала. Функция канального кодирования состоит во внесении по определенному правилу в передаваемые данные избыточности, на основании анализа которой на приемной стороне удается контролировать возникшие при передаче по каналу связи ошибки.

На приемной стороне демодулятор выполняет операцию, обратную по отношению к операции, производимой передатчиком, т.е. он обрабатывает сигналы, искаженные шумами при передаче по каналу связи, и преобразует каждый принятый сигнал в скаляр (или вектор), который представляет собой оценку переданных символов цифровых данных. Приемник (демодулятор) принимает жесткое решение о принятых символах сообщения, если мощность множества возможных оценок модулятора совпадает с мощностью алфавита символов цифровых данных. Мягкое решение приемника соответствует дополнительному квантованию выхода демодулятора с целью подачи на вход декодера дополнительной информации о надежности

принятого решения. Декодер помехоустойчивого кода, используя поступившие на его вход оценки кодовых символов и внесенную на передающей стороне избыточность, обнаруживает и/или исправляет возникшие при передаче данных ошибки.

Рассмотрим математические модели каналов связи с точки зрения количественной оценки достоверности передаваемой информации.

Наиболее простой и распространенной является математическая модель двоичного симметричного канала связи без памяти, которая соответствует двоичному входу и двоичному выходу, т.е. дискретные сообщения обрабатывается устройством канального кодирования и побитно поступают на вход модулятора. С выхода демодулятора поступают двоичные оценки принятых сигналов из канала связи, дополнительное квантование не производится [19, 115-119].

Математическая модель двоичного симметричного канала связи без памяти формально описывается следующими элементами (см. рис. 6.1.):

1.Множество (алфавит) входных символов , поступающих с выхода устройства канального кодирования на вход двоичного симметричного канала без памяти;

2.Множество (алфавит) выходных символов , поступающих с выхода двоичного симметричного канала без памяти на устройство канального декодирования;

3.Набор условных вероятностей возможных выходов при условии возможных входов , , ,

.

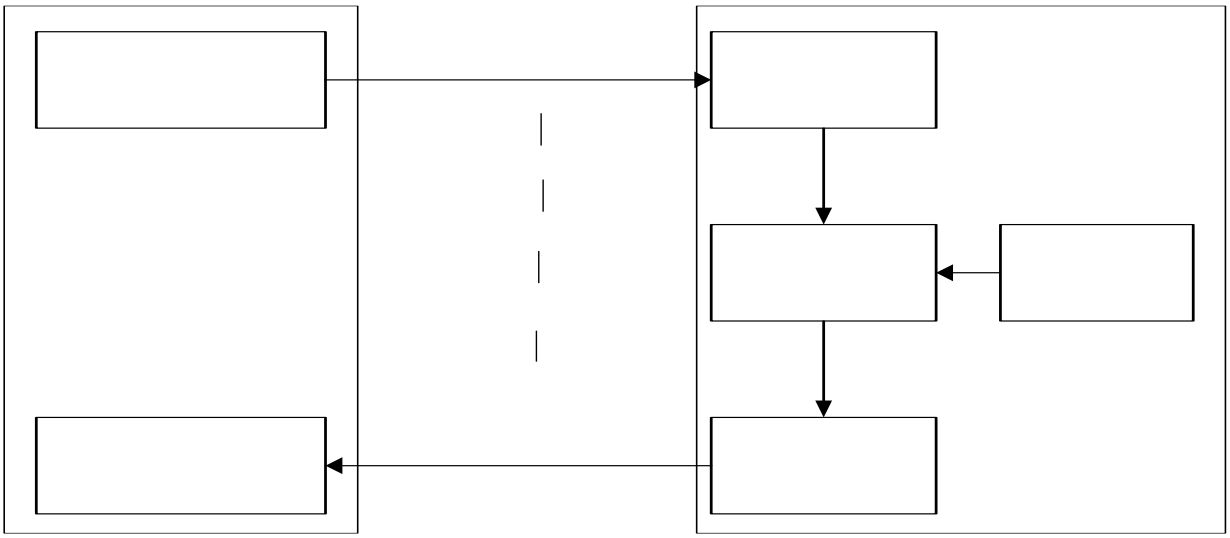


Рис. 6.1. Математическая модель двоичного симметричного канала связи

Если принять условия статистической независимости возникающих в канале связи ошибок (отсутствие памяти канала связи) и симметричности переходных вероятностей в канале связи, тогда выполняются следующие

равенства для условных вероятностей (вероятностей ошибок):

;

,

где - средняя вероятность возникновения ошибки в двоичном кодовом символе.

Взаимная информация о событии когда имеет место событие равна

,

где

.

Средняя взаимная информация, получаемая по выходу двоичного канала о входе равна

Максимизируемая по набору вероятностей входных символов величина задает пропускную способность канала и зависит только от свойств канала связи (от значений переходных вероятностей):

.

Максимум достигается при равновероятном появлении на входе двоичного канала связи символов , т.е. при

. После подстановки получим:

Естественным обобщением математической модели двоичного симметричного канала передачи данных без памяти является расширение мощности алфавита входных и выходных символов на -ичный случай (см. рис. 6.2.).

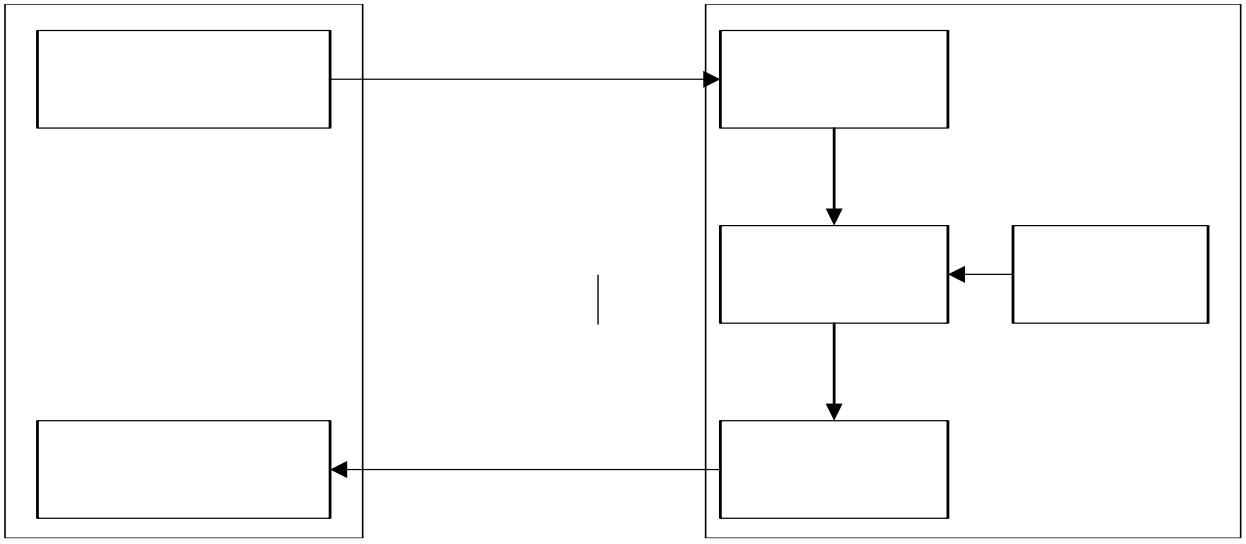


Рис. 6.2. Математическая модель дискретного симметричного канала связи

Предположим, что входом канала являются символы из алфавита , а выходом – символы из алфавита . Тогда множество условных вероятностей

, , …, ,

, , …, ,

…,

, , …, ,

задает множество переходных вероятностей рассматриваемого дискретного канала передачи данных.

Запишем совокупность приведенных условных вероятностей в виде матрицы переходных вероятностей канала передачи данных , где

. Если принять условия статистической независимости возникающих в канале связи ошибок (отсутствие памяти канала связи) и симметричности переходных вероятностей в канале связи (симметричности

матрицы), тогда совокупность рассмотренных элементов описывает математическую модель дискретного симметричного канала передачи данных без памяти.

Предположим, что на вход дискретного канала связи подан символ

, а с выхода принят символ . Взаимная информация о

событии когда имеет место событие равна

.

Средняя взаимная информация, получаемая по выходу дискретного канала о входе равна

.

Величина , максимизируемая по набору вероятностей входных

символов , является пропускной способностью канала и зависит только от свойств канала связи (от значений переходных вероятностей

):

Предположим, что переходные вероятности симметричны и имеют вид:

;

; ,

т.е. ошибки, возникающие в дискретном канале связи равновероятны.

Предположим, что входные символы так же равновероятны, т.е.

.

Тогда после подстановки получим:

При двоичном основании логарифма пропускная способность измеряется в битах/символ. Если символы дискретных сообщений поступают в канал связи каждые секунд, тогда пропускная способность в единицу времени равна бит/секунду.

Величина в свою очередь является функцией от соотношения энергии сигнала к спектральной плотности мощности шума и зависит от выбранной системы модуляции и способа обработки принимаемых сигналов на уровне непрерывного канала связи (см. рис. 6.1 и 6.2).

Условная вероятность ошибочного приема полностью известных равновероятных сигналов при условии оптимального приема в каналах связи с аддитивным белым гауссовым шумом определяется выражением

, (6.1)

где - энергия сигнала, приходящаяся на один передаваемый бит сообщения, N_o - спектральная плотность мощности белого шума; b_s - коэффициент взаимной корреляции между сигналами.

Так, для случая использования двоичного фазоманипулированного (ФМ) сигнала с манипуляцией фазы на 180^0 коэффициент взаимной корреляции $b_s = -1$, т.е. имеем пример ансамбля противоположных друг другу сигналов.

Сделанные допущения приводят к математической модели полунепрерывного канала связи, в которой на вход канала передачи данных подаются символы из дискретного входного алфавита , а с выхода демодулятора поступают не квантованные величины , т.е. вещественные значения , где - множество вещественных чисел (см. рис. 6.3).

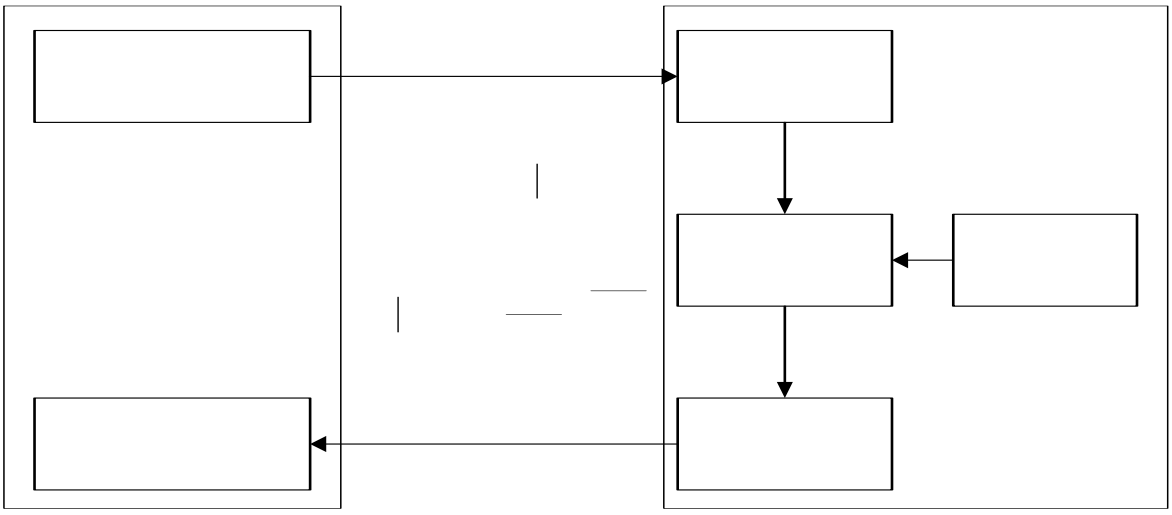


Рис. 6.3. Математическая модель полунепрерывного симметричного канала связи

Распределение условных вероятностей , задает переходные вероятности рассматриваемого полунепрерывного канала

передачи данных. Для модели полунепрерывного канала с аддитивным белым гауссовым шумом справедливо равенство , где - значение гауссовой случайной величины с нулевым математическим ожиданием и дисперсией , т.е. условная вероятность определяется выражением:

.

Средняя взаимная информация, получаемая по выходу дискретного канала о входе равна

.

Максимизация по набору дает пропускную способность полунепрерывного канала:

.

В простейшем случае, при равновероятном двоичном входе:

,

пропускная способность равна

Таким образом, задача оценки достоверности передаваемой информации состоит в расчете вероятности ошибочного приема бита и сопоставлении с соответствующим соотношением энергии сигнала к спектральной плотности мощности шума до и после кодирования.

Для экспериментальной оценки эффективности алгебраически заданных сверточных кодов в ходе исследований была разработана имитационная модель системы передачи информации и методика эмпирической оценки достоверности.

6.2. Имитационная модель системы передачи информации и методика оценки достоверности

Рассмотрим задачу оценки достоверности передаваемой информации в двоичных симметричных каналах без памяти и соответствующих обобщениях на полунепрерывный случай.

Оценка потери достоверности передаваемой информации в двоичных симметричных каналах без памяти есть оценка средней вероятности возникновения ошибки в двоичном кодовом символе. Без применения

канального (помехоустойчивого) кодирования это оценка условной вероятности ошибочного приема сигналов, которая при условии оптимального приема в каналах связи с АБГШ определяется выражением (6. 1).

Применение систем помехоустойчивого кодирования изменяет величину вероятности ошибки и для случайных методов кодирования показатель потери достоверности как оценка вероятности ошибочного приема одного бита сообщения может быть вычислена эмпирическим путем.

В качестве эмпирической оценки вероятности ошибочного приема одного бита сообщения примем частоту ошибок, которую будем оценивать следующим образом

,

где - количество принятых ошибочных бит; - общее количество принятых двоичных символов.

Предположим, что эмпирическая оценка вероятности ошибочного

приема одного бита сообщения производится в результате опытов.

Тогда соответствующая оценка математического ожидания случайной

величины является среднее арифметическое ее наблюдаемых значений (или статистическое среднее) [137, 138]:

где - эмпирическая оценка вероятности ошибочного приема одного бита сообщения в результате -го опыта.

Оценим соответствующую дисперсию случайной величины

Из центральной предельной теоремы теории вероятностей следует, что при больших значениях количества реализаций *N* среднее арифметическое

будет иметь распределение, близкое к нормальному распределению, с математическим ожиданием:

и среднеквадратическим отклонением:

,

где – среднеквадратическое отклонение оцениваемого параметра .

При этом вероятность того, что эмпирическая оценка

отклоняется от своего математического ожидания меньше, чем на (доверительная вероятность), определяется выражением:

,

где - функция Лапласа:

,

а - точность оценки .

Предположим, что находится в доверительном интервале:

,

где – множитель, зависящий от доверительной вероятности

.

Определим множитель из условия

,

т.е.:

,

где – уровень значимости отклонения оценки (вероятность того, что значение оценки вышло за пределы доверительного интервала).

Запишем выражение следующим образом:

,

где – граница доверительного интервала (точность оценки):

.

Зафиксируем требуемую точность оценки , т.е. потребуем, что бы выполнялось неравенство для заранее заданного .

Зафиксируем так же соотношение , соответствующую величину

и оценку потери достоверности без использования кодов.

Увеличивая (до тех пор, пока не выполниться условие) число выполняемых опытов для каждого получим соответствующие эмпирические оценки вероятности ошибочного приема одного бита

сообщения , среднее арифметическое ее наблюдаемых значений ,

дисперсию и величины и . Примем полученную в ходе

моделирования эмпирическую оценку в качестве оценки потери достоверности передаваемых символов сообщения а соответствующую

величину в качестве оценки помехоустойчивости. Разность величин

до и после кодирования при фиксированных вероятностях примем за оценку энергетического выигрыша (проигрыша) от кодирования. Выигрыш (

проигрыш) определяется знаком разности соответствующих величин .

Для эмпирической оценки помехоустойчивости (достоверности) каналов передачи данных разработана имитационная модель системы дискретных сообщений. Ее основные элементы представлены на рис. 6.4.

Разработанная имитационная модель состоит из следующих компонентов:

- источник информационных сообщений, имитируемый датчиком случайных равновероятных величин;
- устройство канального кодирования, имитируемое разработанной программной реализацией кодера алгебраического сверточного кода;
- источник шума, имитируемый датчиком гауссовых случайных

величин с нулевым математическим ожиданием и дисперсией ;

- полунепрерывный канал передачи данных, имитируемый с помощью сумматора значений кодовых символов и отсчетов шума;
- устройство детектирования кодовых символов, имитируемое с помощью программно реализованных правил мягкой и/или жесткой оценки принятых кодовых символов;

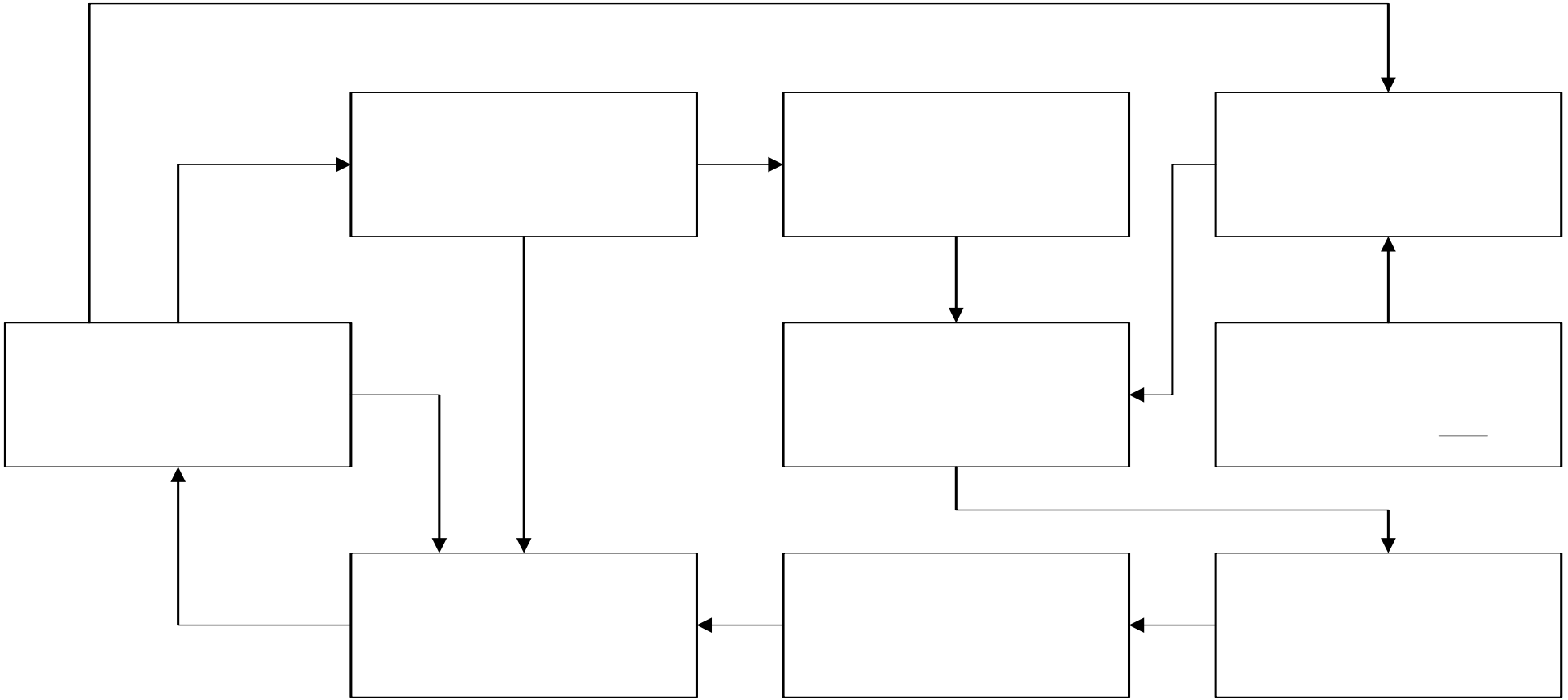


Рис. 6.4. Имитационная модель системы дискретных сообщений с использованием алгебраических сверточных кодов

- устройство канального декодирования, имитируемое разработанной программной реализацией декодера алгебраических сверточных кодов;

- получатель информации, имитируемый программно реализованной процедурой сравнения сформированных информационных и полученных после декодирования символов, а так же программно реализованной процедурой расчета частоты ошибок и соответствующей эмпирической оценки вероятности ошибочного приема бит сообщения;

- устройство управления, осуществляющее согласование работы всех компонентов имитационной модели по введенным параметрам эмпирической оценки вероятности ошибочного приема бит сообщения.

Исходными данными для функционирования имитационной модели являются:

- соотношение энергии сигнала, приходящейся на один бит сообщения,

к спектральной плотности мощности шума , при котором будет производиться эмпирическая оценка вероятности ошибочного приема бит сообщения, а так же рассчитанная дисперсия аддитивного белого гауссова

шума (вводится в источник шума);

- требуемая точность оценки вероятности ошибочного приема бит сообщения (вводится в устройство управления).

Разработанная имитационная модель функционирует следующим образом.

Шаг 1. Вводятся исходные данные , и . Рассчитываем

вероятность ошибочного приема без кодирования. Для моделирования принимаем следующие исходные данные

, , где - относительная скорость кодирования.

Шаг 2. Принимаем .

Шаг 3. Формируем информационных сообщений (с помощью датчика равновероятной случайной величины).

Шаг 4. Формируем кодовых последовательностей (с помощью кодера алгебраического сверточного кода).

Шаг 5. Формируем последовательностей отсчетов шума (с помощью датчика гаусовой случайной величины).

Шаг 6. Формируем последовательностей выходных символов полунепрерывного канала связи (с помощью сумматора значений кодовых символов и отсчетов шума).

Шаг 7. Формируем последовательностей принятых кодовых символов (с помощью правил мягкой и/или жесткой оценки).

Шаг 8. Формируем последовательностей принятых информационных символов (с помощью декодера алгебраического сверточного кода).

Шаг 9. Рассчитываем частоты ошибок для каждого из

опытов, а так же среднее арифметическое и дисперсию

случайной величины .

Шаг 10. Из условия ,

находим и рассчитываем

точность оценки .

Шаг 11. Если принимаем и переходим к шагу 3.

Шаг 12. Если принимаем .

Шаг 13. Рассчитываем соответствующее найденной вероятности соотношение энергии сигнала к спектральной плотности мощности шума, т.е

находим , где .

Шаг 14. Оцениваем энергетический выигрыш от кодирования

, дБ. Если ЭВК имеет положительный знак, применение кодирования приводит к повышению энергетической эффективности, в противном случае система помехоустойчивого кодирования при введенных параметрах дает проигрыш.

Таким образом, разработанная имитационная модель системы передачи информации и предложенная методика эмпирической оценки достоверности позволяет для заданных параметров полунепрерывного канала с заданной точностью оценить вероятность ошибочного приема одного бита сообщения и соответствующий энергетический выигрыш от кодирования.

6.3. Достоверность передаваемой информации с использованием алгебраически заданных сверточных кодовых конструкций

Турбокод, как и составляющие его сверточные коды, является линейным кодом. Линейная природа турбокодов позволяет минимальное расстояние

кода определять путем сравнения каждого кодового слова с кодовым словом, состоящим из всех нулей. Это значительно упрощает анализ кода, так как в этом случае минимальное расстояние кода эквивалентно наименьшему весу кодового слова, который определяется как количество единиц в кодовой последовательности. Так как турбокод состоит из нескольких сверточных кодов, то сначала рассмотрим верхнюю границу вероятности ошибки для сверточных кодов, а затем используем ее для получения верхней границы вероятности ошибки на бит для турбокодов.

Будем считать, что для декодирования сверточного кода используется декодер Витерби, который находит наиболее вероятный путь в решетчатой диаграмме и передаваемое кодовое слово состоит только из нулевых символов. Если декодер Витерби выберет неправильный путь в решетчатой диаграмме, то это приведет к ошибке декодирования. На рис. 6.5 показаны три ошибочных события, заключающиеся в том, что от узла расходятся три пути и сходятся позже с правильным путем (путем из всех нулей). Такие события, приводящие к ошибкам декодирования, будем называть ошибками в узле или узловыми ошибками.

Рис. 6.5. Примеры ошибочных событий, возникающих в решетчатой диаграмме сверточного кода при декодировании декодером Витерби

Вероятность ошибки в узле можно оценить сверху, используя аддитивную границу [19, 20]:

. (6.2)

Предположим, что известны ошибочных путей, находящихся на расстоянии Хэмминга от правильного пути. Тогда выражение (6.2) можно записать следующим образом

, (6.3)

где - попарная вероятность ошибки для кодовых слов, находящихся на расстоянии . Для канала с двоичным входным алфавитом и бесконечным

выходным алфавитом (предполагается принятие мягких решений демодулятором) попарная вероятность ошибки определяется как

.

Однако нас интересует верхняя граница вероятности ошибки на бит. Математическое ожидание числа двоичных ошибок (то есть искажений отдельных двоичных символов), порождаемых неправильными путями, ответвляющимися от правильного пути в узле , может быть ограничено сверху путем взвешивания каждого слагаемого аддитивной границы выражения (6.3) числом двоичных ошибок, которые порождаются соответствующим неправильным путем. Это число двоичных ошибок будет совпадать с числом единичных символов в информационной последовательности, если считать, что нулевой путь является правильным путем.

Для скорости кода каждый узел соответствует одному биту информации, вводимой в кодер. Поэтому вероятность ошибки на бит, определяемая как математическое ожидание числа двоичных ошибок на один декодированный двоичный символ, может быть ограничена следующим образом

, (6.4)

где - количество путей, ответвляющихся от правильного пути в узле решетчатой диаграммы, находящихся на расстоянии от этого пути и имеющих единичных символов в соответствующей информационной последовательности. Значения и могут быть найдены методами, предложенными в [20, 131].

Рассмотрим сверточный код конечной длины. Модифицируя выражение (6.4), переопределим как количество кодовых слов, имеющих общий вес и вес информационных символов . В этом случае вероятность ошибки на бит может быть найдена делением правой части выражения (6.4) на длину последовательности информационных символов

. (6.5)

Так как сверточный код конечной длины является блочным кодом, а исправляющая способность блочных кодов определяется минимальным расстоянием, то в дальнейшем, при рассмотрении кодов конечной длины, будем использовать минимальное расстояние вместо свободного расстояния .

Изменим порядок суммирования в выражении (6.5):

; (6.6)

, (6.7)

где - вес информационных символов всех кодовых слов веса , разделенный на количество информационных символов в кодовом слове :

.

Определим как количество кодовых слов с весом и - как средний информационный вес кодовых слов с весом . Средний информационный вес всех кодовых слов с общим весом определяется произведением :

.

Учитывая (6.7), получим

,

где назовем эффективной кратностью кодового слова весом . Таким образом, можно записать выражение для верхней границы вероятности ошибки на бит:

.

Для выработки практических рекомендаций по применению синтезированных параллельных каскадных кодовых конструкций, построенных на основе алгебраически заданных рекурсивных сверточных кодов, найдем верхнюю границу вероятности ошибки на бит для турбокода.

Известно, что для систематического блочного кода функция распределения веса кодовых слов (ФРВ) определяется следующим образом [48-50]:

,

где - количество кодовых слов с весом Хэмминга , - формальная переменная.

Определим расширенную функцию распределения веса (РФРВ):

,

где - количество кодовых слов, порождаемых входными последовательностями веса и имеющих вес проверочной последовательности , так, что общий вес кодового слова определяется как , , - формальные переменные.

Функция распределения веса кодовых слов ФРВ может быть использована для вычисления вероятности ошибки в кодовом блоке, в то время как расширенная функция распределения веса кодовых слов РФРВ может быть использована для вычисления вероятности ошибки на бит.

Связь между ФРВ и РФРВ определяется как

,

где .

Введем условную функцию распределения веса, которая учитывает все возможные кодовые слова, имеющие вес информационной последовательности равный [48-50]:

.

Нахождение РФРВ турбокода для конкретного типа перемежителя (конкретного правила перестановки информационных символов) представляет собой трудноразрешимую задачу (особенно если перемежитель большого размера). Поэтому с целью упрощения будем использовать абстрактную модель случайного перемежителя, которую далее будем называть равномерным перемежителем [48]. Такая модель перемежителя позволяет усреднить характеристики турбокода по всем возможным перемежителям длины .

Для длины перемежителя существует возможных псевдослучайных перемежителей (возможных перестановок) и, предполагая, что каждая перестановка равновероятна, вероятность выбора какой-либо перестановки определяется как .

Пусть перемежитель осуществляет перестановку символов информационной последовательности веса . Всего существует таких перестановок из возможных. Вероятность любой перестановки последовательности веса определяется как

.

Таким образом, равномерный перемежитель длины производит отображение входного слова веса во все возможные перестановок с вероятностью . Использование модели равномерного перемежителя позволяет считать, что РФРВ составляющих кодов турбокода (и) являются независимыми друг от друга. Поэтому РФРВ турбокода с равномерным перемежителем можно записать следующим образом [48]:

.

.

Определим верхнюю границу вероятности ошибки турбокода с равномерным перемежителем [48]:

;

; (6.8)

.

Из (6.8) получим окончательное выражение для оценки верхней границы вероятности ошибки турбокода:

, (6.9)

где - количество проверочных символов турбокода.

Отметим, что внешнее суммирование в выражении (6.9) производится по всем возможным весам информационных последовательностей, а внутреннее суммирование – по всем возможным весам проверочных последовательностей. Для рекурсивных сверточных кодов минимальный вес информационной последовательности, которая порождает кодовые последовательности конечной длины, равен 2. Любые информационные последовательности веса 1 будут порождать кодовые последовательности турбокода весом и поэтому слагаемым с в выражении (6.9) можно пренебречь.

Для равномерного перемежителя справедливы следующие соотношения: , , где - количество информационных последовательностей веса , которые порождают кодовые последовательности конечного веса одновременно на выходе двух составляющих кодеров [44], приводящих к появлению кодовых слов турбокода с весом . Из этого следует, что слагаемыми верхней границы в

выражении (6.9) с можно пренебречь (при условии, что перемежитель равномерный) и считать, что кодовые слова минимального веса турбокода порождаются информационными последовательностями веса 2. Однако если использовать в качестве составляющих кодов нерекурсивные сверточные коды, то кодовые слова минимального веса могут порождаться и информационными последовательностями веса 1, что будет приводить к увеличению вероятности ошибки такого турбокода, за счет влияния слагаемого в выражении (6.9) с .

Верхняя граница, определяемая выражением (6.9), предполагает знание всех возможных комбинаций ошибочных путей. Для больших размеров перемежителей (более 100) нахождение всех слагаемых выражения (6.9) становится затруднительным. Поэтому для упрощения вычисления верхней границы ограничим количество слагаемых в выражении (6.9), путем использования слагаемых, для которых вес кодовых последовательностей не превышает , где - вес кодовой последовательности, порождаемой информационной последовательностью веса 2. Получаемая таким образом верхняя граница будет менее точна для области, в которой . Однако наибольший интерес представляет область, в которой . В указанной области верхняя граница с ограниченным количеством слагаемых дает очень близкие результаты к верхней границе со всеми возможными слагаемыми.

На рис. 6.6 – 6.15 представлены кривые зависимости верхней границы вероятности ошибки от для различных значений , , полученные при помощи (6.9).

Из анализа рис. 6.6 – 6.15 следует, что при фиксированном значении вероятность ошибки уменьшается с ростом количества элементов памяти и уменьшении скорости кода. Однако при использовании кодов со скоростями менее 1/3 рост эффективности кодирования приостанавливается. Использование составляющих сверточных кодов со скоростями для увеличения общей скорости турбокода более эффективно, чем применение процедуры выкалывания.

Для обеспечения вероятности ошибки на бит при значении энергетического отношения сигнал/шум, близкому к теоретическому пределу (1,5 – 2 дБ), предлагается использовать турбокоды с количеством элементов памяти 2 – 4. Для обеспечения вероятности ошибки на бит предлагается использовать турбокоды с количеством элементов памяти 6 – 8. Скорость кодирования не рекомендуется выбирать менее чем 1/3.

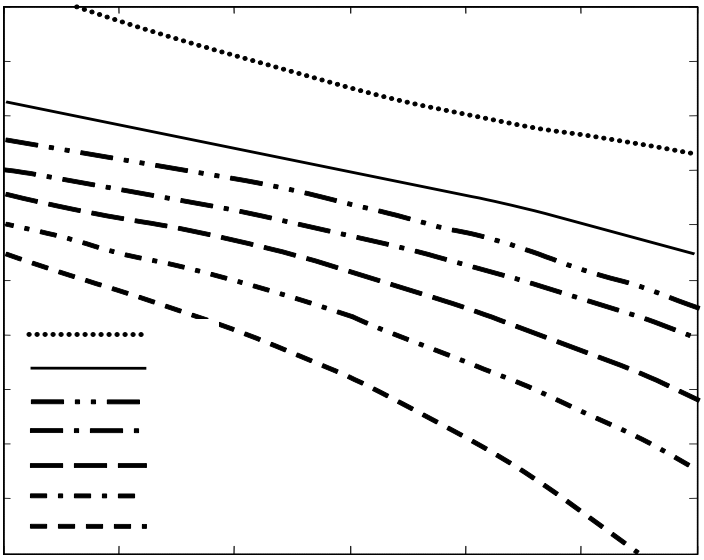


Рис. 6.6. Верхняя граница $P_{ош}$, , ,

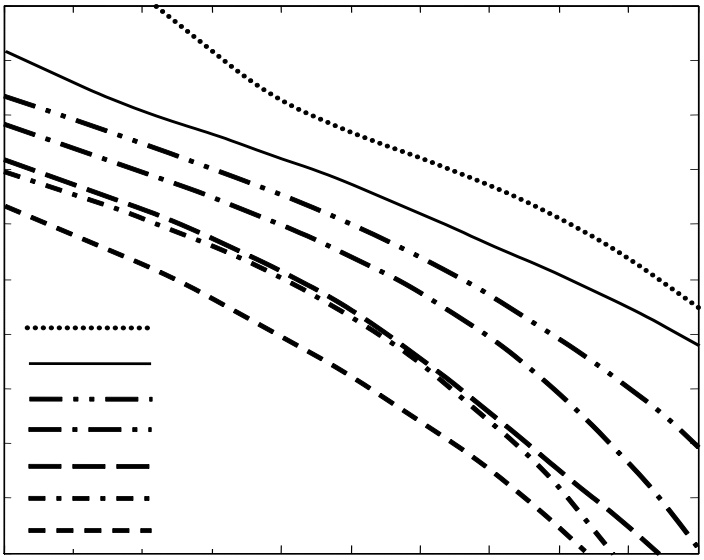


Рис. 6.7. Верхняя граница $P_{ош}$, , , (выкалывание)

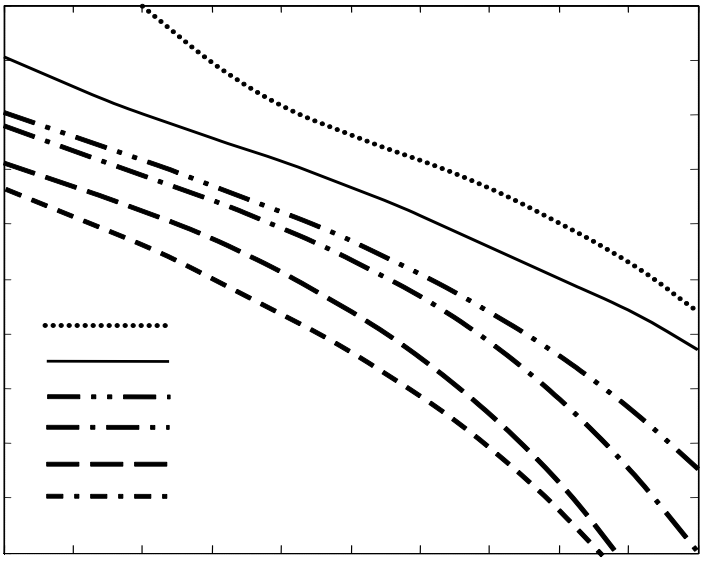


Рис. 6.8. Верхняя граница $P_{ош}$, , ,

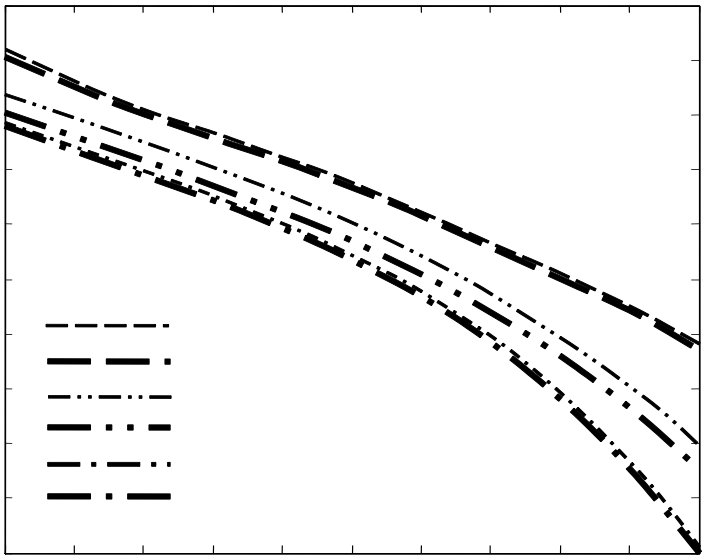


Рис. 6.9. Верхняя граница $P_{ош}$, , ,
и (выколотый)

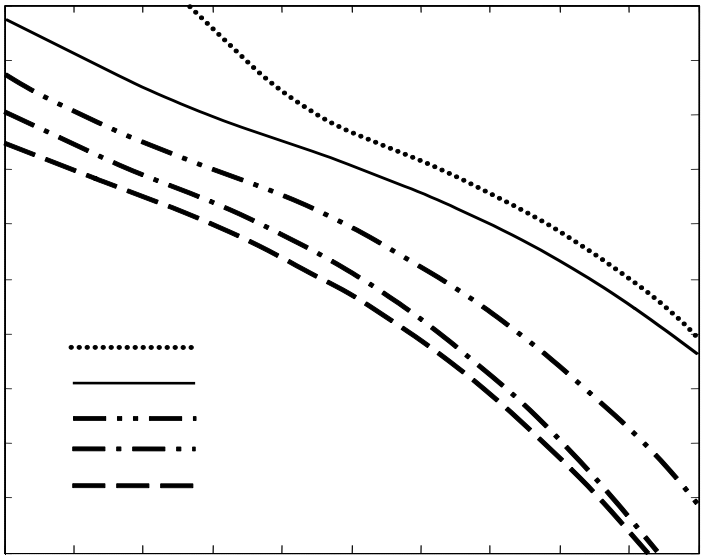


Рис. 6.10. Верхняя граница $P_{ош}$, , ,

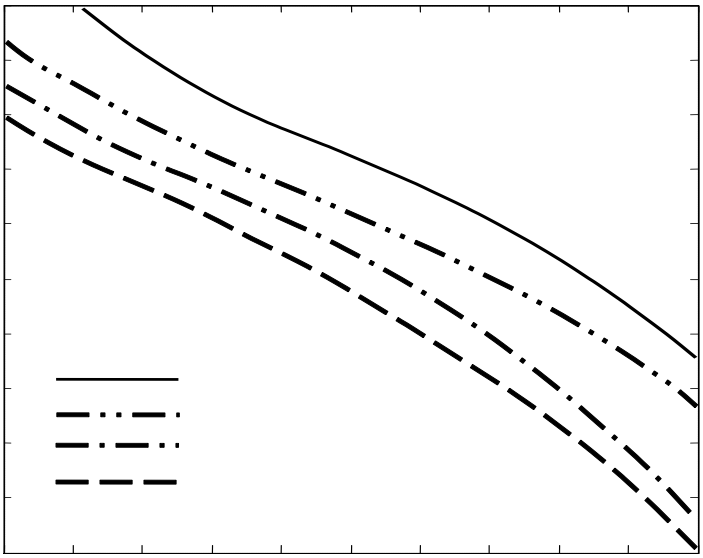


Рис. 6.11. Верхняя граница $P_{ош}$, , ,

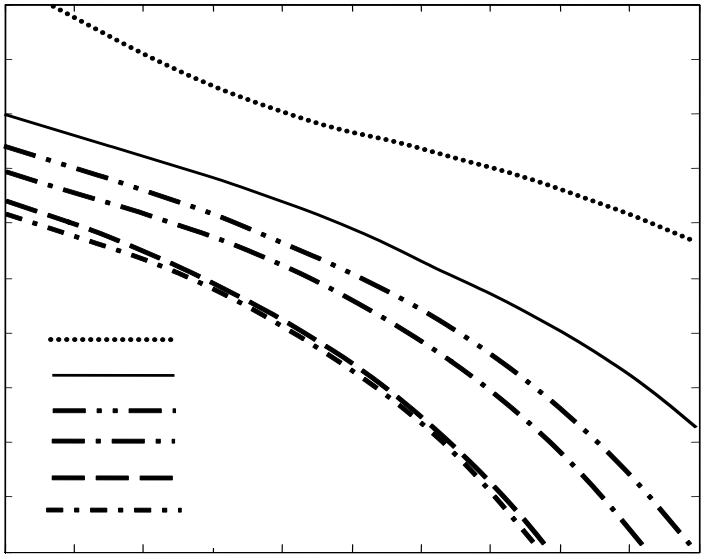


Рис. 6.12. Верхняя граница $P_{\text{ош}}$, , ,

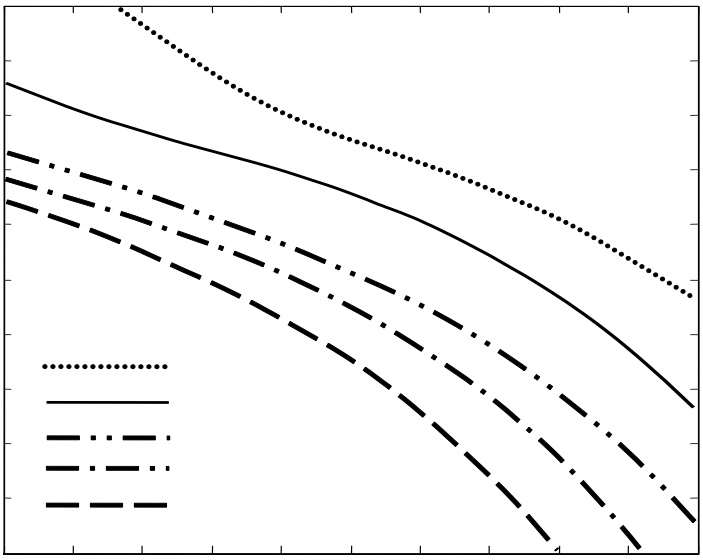


Рис. 6.13. Верхняя граница $P_{\text{ош}}$, , ,

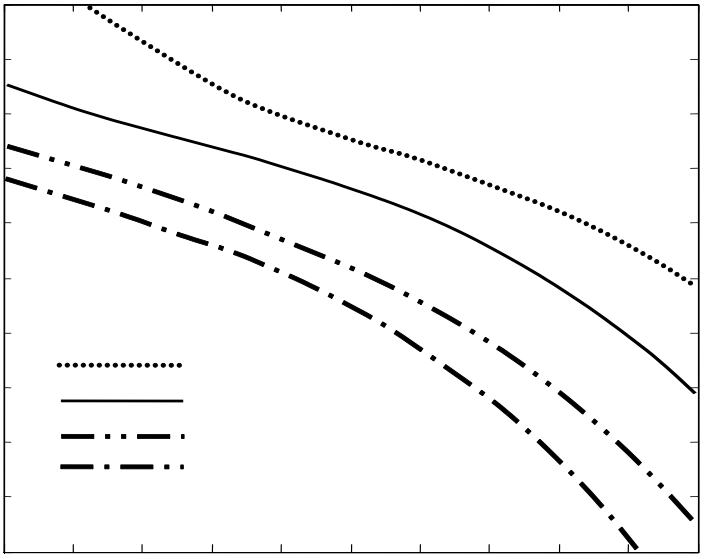


Рис. 6.14. Верхняя граница $P_{ош}$, , ,

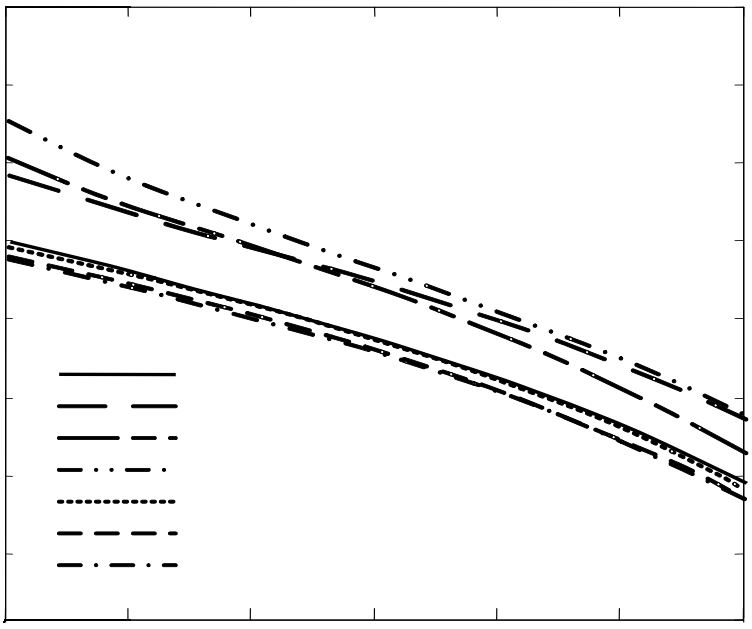


Рис. 6.15. Верхняя граница $P_{ош}$, для различных , ,

В ходе проведенных исследований была разработана имитационная модель системы передачи информации, использующая турбокоды с синтезированными алгебраически заданными сверточными кодами. В приложении А приведен листинг программной реализации имитационной

модели.

С помощью разработанной имитационной модели были экспериментально подтверждены аналитически полученные кривые, приведенные на рис. 6.6 – 6.15 для дБ. Для примера, на рис. 6.16 показаны кривые зависимости вероятности ошибки на бит от отношения энергии сигнала к спектральной плотности мощности шума, полученные аналитически (сплошная линия) и при помощи имитационного моделирования (пунктирная линия). Из анализа рисунка 6.16 следует хорошее совпадение кривых при дБ, что подтверждает достоверность полученных результатов.

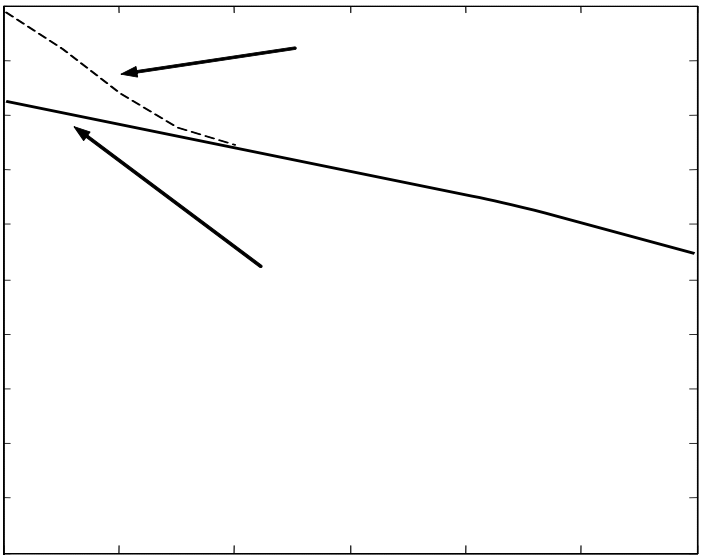


Рис. 6.16. Кривые $P_{\text{ош}}$, полученные при помощи моделирования и путем приближенного вычисления верхней границы

Выводы

1. Проведенные исследования показали, что алгебраически заданные сверточные коды и параллельные каскадные кодовые конструкции на их основе (турбокоды) позволяют обеспечить существенное повышение достоверности передаваемой информации по каналам со случайными ошибками.

2. В результате проведенных исследований математических моделей каналов связи разработана методика оценки эффективности помехоустойчивого кодирования. Следует отметить, что на методику оценки эффективности помехоустойчивого кодирования не накладываются ограничения на вид закона распределения длин пакетов ошибок, что позволяет проводить исследования эффективности помехоустойчивого

кодирования для широкого класса каналов связи.

3. Аналитически получены кривые зависимости вероятности ошибки на бит от энергетического отношения сигнал/шум для различных длин , , . Анализ полученных результатов показал, что при фиксированном значении вероятность ошибки уменьшается с ростом кодового ограничения и уменьшении скорости сверточного кода. Однако при использовании кодов со скоростями менее 1/3 рост эффективности кодирования приостанавливается. Использование составляющих сверточных кодов со скоростями для увеличения общей скорости турбокода более эффективно, чем применение процедуры выкалывания.

3. С помощью разработанной имитационной модели были экспериментально подтверждены аналитически полученные результаты при дБ, что подтверждает достоверность полученных результатов.

4. В ходе проведенных исследований с использованием разработанной имитационной модели системы передачи информации установлено, что турбокоды на основе алгебраически заданных рекурсивных сверточных кодов не уступают по эффективности известным в настоящее время кодам.

5. Исходя из анализа полученных результатов исследований, разработаны практические рекомендации по использованию турбокодов на основе синтезированных алгебраически заданных рекурсивных сверточных кодов. Для обеспечения вероятности ошибки на бит при значении энергетического отношения сигнал/шум 1,5 – 2 дБ, предлагается использовать турбокоды с количеством элементов памяти 2 – 4. Для обеспечения вероятности ошибки на бит предлагается использовать турбокоды с количеством элементов памяти 6 – 8. Скорость кодирования не рекомендуется выбирать менее чем 1/3.

6. Практическое использование полученных результатов позволяет достичь высокой достоверности передаваемой информации при отношении сигнал/шум, близкому к теоретически предельному значению, определяемому теоремой Шеннона, и, удовлетворить наиболее жестким современным требованиям.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Михалєвич В.С. Інформатизація України. Концепція державної політики інформатизації. Основні напрями Національної програми інформатизації України / Михалєвич В.С. // Управляющие системы и машины. – 1994. – №4/5. – С. 7 – 21.
2. Про Концепцію Національної програми інформатизації. Закон України. № 75/98-ВР від 04.02.98.
3. Згуровський М. Інформаційні технології у сучасному суспільстві / Згуровський М., Сергієнко І. // Вісн. НАН України. – 2000. – №12. – С. 9 – 16.
4. Сорока Л.С. Основы построения АСУ / Сорока Л.С., Приходько С.И. – Харьков: ХВВКИУ, 1988. – 132 с.
5. Овчинников В.Н. Организация передачи информации в автоматизированных системах управления / Овчинников В.Н. – М.: Энергия, 1974. – 198 с.
6. Королев А.В. Обработка информации в АСУ / Королев А.В. – МОУ, 1996. – 372 с.
7. Ключко В.И. Защита информации от ошибок в АСУ / Ключко В.И. – МО СССР, 1980. – 256 с.
8. Ключко В.И. Защита информации от ошибок в АСУ / Ключко В.И. – МО СССР, 1980. – 256 с.
9. Николаев Ф.А. Проблемы повышения достоверности в информационных системах / Николаев Ф.А., Фолин В.И., Хохлачев Л.М. – Л.: Энергоатомиздат, 1982. – 138 с.
10. Пивоваров А.Н. Методы обеспечения достоверности информации в АСУ / Пивоваров А.Н. – М.: Радио и связь, 1982. – 325 с.
11. Котов П.А. Повышение достоверности передачи цифровой информации / Котов П.А. – М.: Связь, 1966. – 184 с.
12. Концепция развития связи Украины до 2010 года, утвержденная постановлением Кабинета Министров Украины «Про Концепцію розвитку зв'язку України до 2010 року» от 9 декабря 1999 г. №2238.
13. Блейхут Р. Теория и практика кодов, контролирующих ошибки / Р. Блейхут; пер. с англ. И.И. Грушко, В.М. Блиновский под. ред. К.Ш. Зигангирова. – М.: Мир, 1986. – 576 с.
14. Берлекэмп Э.Р. Алгебраическая теория кодирования / Э. Берлекэмп; пер. с англ. И.И. Грушко под. ред. С.Д. Бермана. – М.: Мир, 1971. – 477 с.
15. Возенкрафт Дж. Теоретические основы техники связи / Дж. Возенкрафт, И. Джекобс; пер. с англ. под. ред. Р.Л. Добрушина. – М.: Мир, 1969. – 640 с.
16. Мак-Вильямс Ф.Дж., Слоэн Н.Дж. Теория кодов, исправляющих ошибки / Ф.Дж. Мак-Вильямс, Н.Дж. Слоэн; пер. с англ. И.И. Грушко, В.А. Зиновьева под. ред. Л.А. Бассалыго – М.: Связь, 1979. – 744 с.

17. Питерсон У. Коды, исправляющие ошибки / У. Питерсон, Э. Уэлдон пер. с англ. под ред. Р.Л. Добрушина, С.И. Самойленко. – М.: Мир, 1976. – 576 с.
18. Теория кодирования / Т. Касами, Н. Токура, Е. Ивадари, Я. Инагаки; пер. с японского А.В. Кузнецова; под ред. Б.С. Цыбакова, С.И. Гельфанда. – М .: Мир, 1978. – 576 с.
19. Витерби А.Д. Принципы цифровой связи и кодирования / А.Д. Витерби, Дж.К. Омура; пер. с англ. под ред. К.Ш. Зигангирова. – М.: Радио и связь, 1982. – 535 с.
20. Кларк Дж. Кодирование с исправлением ошибок в системах цифровой связи / Дж. Кларк, мл., Дж. Кейн; пер. с англ. С.И. Гельфанда под ред. Б.С . Цыбакова. – М.: Радио и связь, 1987. – 392 с.
21. Нейфах А.Э. Свёрточные коды для передачи дискретной информации / Нейфах А.Э. – М.: Наука, 1979. – 320 с.
22. Витерби А.Д. Границы ошибок для свёрточных кодов и асимптотически оптимальный алгоритм декодирования / А.Д. Витерби // Некоторые вопросы теории кодирования: сб. науч. труд. – М.: Мир, 1970. – С.67-72.
23. Месси Дж. Пороговое декодирование / Дж. Месси. – М.: Мир, 1966. – 253 с.
24. Зигангиров К.Ш. Процедуры последовательного кодирования / К.Ш. Зигангиров. – М.: Связь, 1974. – 279 с.
25. Робинсон Дж. П. Размножение ошибок и прямое декодирование свёрточных кодов / Дж. П. Робинсон // Некоторые вопросы теории кодирования: сб. науч. труд. – М.: Мир, 1970. – С.57-62.
26. Приходько С.И., Гусев С.А., Сидоренко Н.Ф. Спектральное представление сверточных кодов / Приходько С.И., Гусев С.А., Сидоренко Н.Ф. // Обработка информации: сб. науч. труд. – Х.: НАНУ, ПАНИ, ХВУ, 1996. – С.99-105.
27. Приходько С.И. Критерий ошибки при алгебраическом последовательном декодировании сверточных кодов / Приходько С.И. // Информатика: сб. науч. труд. – К.: Наукова Думка, 1999. – №7. – С.9-12.
28. Приходько С.И. Принцип декодирования сверточных кодов с контролем верности // Информационные системы: сб. науч. труд. – Х.: НАНУ, ПАНИ, ХВУ, 1998. – №3. – С.32-36.
29. А.с. 1252944 СССР, МКИ H03M 3/12. Пороговый декодер сверточного кода / В.И. Ключко, Г.Е. Березняков, С.И. Приходько, Ю.И. Николаев, И. В. Чистяков (СССР). – № 3836495/24-24; заявл. 02.01.85; опубл. 23.08.86, Бюл. № 31.
30. А.с. 1381720 СССР, МКИ H03M 13/02. Декодирующее устройство / С.В. Кузнецов, Ю.И. Николаев, В.О. Александров, С.И. Приходько, С.Г. Рассомахин, Л.С. Сорока (СССР). – № 4124958/24-24; заявл. 26.09.86; опубл. 15.03.88, Бюл. № 10.
31. А.с. 1522415 СССР, МКИ H03M 13/02. Декодирующее устройство / С.В. Кузнецов, Л.С. Сорока, Ю.И. Николаев, В.О. Александров, С.И. Приходько, С.Г. Рассомахин, А.Ф. Чипига, О.П. Малофей (СССР). – №

4381598/24-24; заявл. 29.02.88; опубл. 15.11.89, Бюл. № 42.
32. А.с. 1580567 СССР, МКИ H03M 13/12. Кодек несистематического сверточного кода / С.И. Приходько, Л.С. Сорока, А.С. Столяров, В.И. Глушков, А.Г. Снисаренко (СССР). – № 4398980/24-24; заявл. 29.03.88; опубл. 23.07.90, Бюл. № 27.
33. А.с. 1662013 СССР, МКИ H03M 13/12. Кодек несистематического сверточного кода / С.И. Приходько, А.Г. Снисаренко, Л.С. Сорока, А.С. Столяров, В.И. Глушков, (СССР). – № 4499301/24; заявл. 20.07.88; опубл. 07.07.91, Бюл. № 25.
34. А.с. 1695516 СССР, МКИ H03M 13/12. Кодер несистематического сверточного кода / А.Г. Снисаренко, Л.С. Сорока, С.И. Приходько, А.С. Столяров, О.А. Снисаренко, (СССР). – № 4789088/24; заявл. 07.02.90; опубл. 23.02.92, Бюл. № 7.
35. А.с. 1714812 СССР, МКИ H03M 13/12. Кодек несистематического сверточного кода / А.Г. Снисаренко, С.И. Приходько, Л.С. Сорока, А.С. Столяров, О.А. Снисаренко, (СССР). – № 4788868/24; заявл. 07.02.90; опубл. 30.11.91, Бюл. № 44.
36. Шеннон К. Работы по теории информации и кибернетике / Шеннон К.. – М.: Изд. иностр. лит, 1963. – 829 с.
37. Шеннон К. Связь при наличии шума / Шеннон К. // Теория информации и ее приложения: сборник переводов. – М.: ФИЗМАТГИЗ, 1959. – С.82 – 112.
38. Форни Д. Каскадные коды / Форни Д. – М.: Мир, 1970. – 207с.
39. Блох Э.Д. Линейные каскадные коды / Блох Э.Д., Зяблов В.В. – М.: Наука , 1982. – 229 с.
40. Блох Э.Д. Обобщенные каскадные коды / Блох Э.Д., Зяблов В.В. – М.: Связь, 1976. – 240 с.
41. Berrou C. Near Shannon limit error correcting coding: Turbo codes / Berrou C ., Glavieux A., Thitiumjshima P. // Int. Conf. on Commun. – Geneva, Switzerland, 1993. – P. 1061 – 1070.
42. Berrou C. Near Optimum Error Correcing Coding and Decoding: Turbo-Codes / Berrou C., Glavieux A. // IEEE Trans. On Comm. – October 1996. – Vol. 44. – №10. – P. 1261 – 1271.
43. Barbulescu A. Iterative decoding of turbo codes and other concatenated codes: PhD Thesis / Barbulescu A. – University of South Australia. – 1996. – 159 p. – Режим доступу: http://www.itr.unisa.edu.au/~steven/thesis/sab.ps.gz.
44. Chuen Ho M.S. Serial and parallel concatenated turbo-codes: PhD Thesis / Chuen Ho M.S. – University of South Australia. – 2002. – 144 p. – Режим доступу: http://www.itr.unisa.edu.au/rd/pubs/thesis/msch.ps.gz.
45. Turbo codes: Principles and applications / Divsalar D., Benedetto S., Pollara F ., Montorsi G. // Lecture Notes. – October 1997. – P. 42 – 51.
46. Optimal decoding of linear codes for minimizing symbol error rate / Bahl L., Cocke J., Jelinek F., Raviv J. // IEEE Trans. Inform. Theory. – 1974. – Vol. IT –20. – P. 284 – 287.

47. Benedetto S. Role convolutional codes in turbo codes / Benedetto S., Montorsi G. // Electronics Letters. – 1994. – Vol. 31, №11. – P. 858-859.
48. Benedetto S. Unveiling turbo codes: some results on parallel concatenated coding schemes / Benedetto S., Montorsi G. // IEEE Trans. Inform. Theory. – 1996. – Vol. IT – 42, №2. – P. 409 – 428.
49. Benedetto S. Design of parallel concatenated codes / Benedetto S., Montorsi G. // IEEE Trans. Comm. – May 1996. – P. 591 – 600.
50. Perez L. A distance spectrum interpretation of turbo codes / Perez L., Seghers J ., Costello D. // IEEE Trans. Inform. Theory. – Nov. 1996. – P. 1698 – 1709.
51. Robertson P. A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain / Robertson P., Villebrun E., Hoeher P. // IEEE Int. Conf. Communications. – Seattle, WA. – June 1995. – P. 1009 – 1013.
52. Pietrobon S.S. Implementation and performance of a Turbo/MAP decoder / Pietrobon S.S. // International Journal of Satellite Communications. – 1997. – Vol. 16. – P. 23 – 46.
53. Divsalar D. Multiple turbo codes / Divsalar D., Pollara F. // Proc. IEEE MILCOM. – San Diego, CA. – 1995. – P. 279 – 285.
54. Pietrobon S.S. A simplification of the modified Bahl decoding algorithm for systematic convolutional codes / Pietrobon S.S., Barbulescu A.S. // Int. Symp. on Inform. Theory and its Applications. – Sydney, Australia. – November 1994 – P. 1073 – 1077.
55. Hagenauer J. Iterative decoding of binary block and convolutional codes / Hagenauer J., Offer E., Papke L. // IEEE Transactions On Information Theory. – 1996. – Vol. 42. – P. 429 – 445.
56. Valenti M.C. Iterative detection and decoding for wireless communications: PhD thesis / Valenti M.C. – Virginia Polytechnic Institute. – 1999. – 217 p. – Режим доступу: http://csee.wvu.edu/~mvalenti/research.html.
57. Dolinar S. Weight distributions for turbo codes using random and nonrandom permutations / Dolinar S., Divsalar D. // TDA progress report 42-122. – Jet propulsion Lab. – Pasadena, CA. – August 1995. – P. 56 – 64. – Режим доступу: http://tmo.jpl.nasa.gov/progress_report.
58. Hagenauer J. Applications of iterated (turbo) decoding / Hagenauer J. // Communications Technical University of Munich, Germany. – Режим доступу: http://www.eei.uni-erlangen.de.
59. Malardel F. Simulation and optimization of the turbo decoding algorithm / Malardel F. // EFREI. – November 1996. – Режим доступу: http://www.itr/levels.unisa.edu.au.
60. Solomon G. A connection between block and convolutional codes / Solomon G ., van Tilborg C.A. // SIAM Journal of Applied Mathematics. – 1979. – № 2. – Vol. 37.
61. Приходько С.И. Модифицированный метод декодирования турбокодов / Приходько С.И., Жученко А.С., Пархоменко Д.А. // Системи обробки інформації. – Х.: ХВУ, 2004. – Вип. 3. – С. 174 – 178.

62. Приходько С.И. Метод оценки дисперсии шума в турбодекодере / Приходько С.И., Жученко А.С., Пархоменко Д.А. // Системи обробки інформації. – Х.: ХВУ, 2004. – Вип. 9(37). – С. 136 – 140.
63. Жученко А.С. Оценка влияния перемежителей на эффективность итеративного декодирования турбокодов / Жученко А.С. // Системи обробки інформації. – Х.: ХВУ, 2004. – Вип. 8(36). – С. 157 – 164.
64. Приходько С.И. Алгебраический метод формирования структуры перемежителя кодека турбокода / Приходько С.И., Жученко А.С. // Інформаційно-керуючі системи на залізничному транспорті. – Харків, 2005.– №3 – С. 44 – 48.
65. Приходько С.И. Один из способов построения случайных кодов большой длины / Приходько С.И., Жученко А.С., Пархоменко Д.А. // Інформаційно -керуючі системи на залізничному транспорті. – Харків, 2004.– №1 – С. 19 – 21.
66. Приходько С.И. Анализ числовых характеристик логарифма отношения правдоподобия МАР декодера / Приходько С.И., Жученко А.С., Пархоменко Д.А. // Радиоэлектроника и информатика. – 2004. – №2(27). – С. 109 – 112.
67. Жученко А.С. Метод формирования перемежителя турбодекодера для перемежения последовательностей различных длин / Жученко А.С. // Збірник наукових праць. – Інститут проблем моделювання в енергетиці ім . Г.Є. Пухова. – Київ: НАН України, 2004. – Вип. 25. – С. 42 – 47.
68. Приходько С.І. Модифікований метод декодування турбокодів / Приходько С.І., Жученко О.С., Пархоменко Д.О. // IV наукова конференція молодих вчених Харківського військового університету: тези доповідей, 14–15 квітня 2004 р. – Х.: ХВУ, 2004. – С. 59.
69. Приходько С.І. Метод формування перемежувача турбокодера для перемеження послідовностей різних довжин / Приходько С.І., Жученко О. С., Пархоменко Д.О. // Перша науково-технічна конференція Харківського університету Повітряних Сил: тези доповідей, 16-17 лютого 2005 р. – Х.: ХУПС, 2005. – С. 212–213.
70. Приходько С.И. Модифицированный метод декодирования турбокодов / Приходько С.И., Жученко А.С., Пархоменко Д.А. // Материалы 8-го международного форума «Радиоэлектроника и молодежь в XXI веке». – Х .: ХНУРЭ, 2004. – Ч. 1. – С. 79.
71. Приходько С.И. Алгебраический метод формирования структуры перемежителя кодека турбокода / Приходько С.И., Жученко А.С. // Перша міжнародна наукова конференція «Теорія та методи обробки сигналів»: тези доповідей. – К.: НАУ, 2005. – С. 78–79.
72. Приходько С.И. Метод итеративного декодирования турбокодов уменьшенной сложности в информационно-телекоммуникационных системах / Приходько С.И., Жученко А.С., Гиневский А.М. // Материалы 18 международной научно-практической конференции „Перспективні системи управління на залізничному, промисловому і міському транспорті”. – Інформаційно-керуючі системи на залізничному

транспорті. – 2005.– №5. – С. 91 – 92.
73. Solomon G. A connection between block and convolutional codes / Solomon G ., van Tilborg C.A. // SIAM Journal of Applied Mathematics, 1979. –№ 2. – Vol.37.
74. Приходько С.И. Повышение помехоустойчивости радиосистем при помощи двоичных кодов БЧХ и сверточных кодов, построенных с помощью кодов РС / Приходько С.И., Ключко В.И. // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1985. – №72. – С.77-81.
75. Приходько С.И. Особенности циклового декодирования сверточных кодов / Приходько С.И., Ключко В.И. // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1985. – №73. – С.52-60.
76. Приходько С.И. Построение сверточных кодов с использованием кодов РС / Приходько С.И., Березняков Г.Е. // Тематический научно-технический сборник. – Х.: ХВВКИУРВ, 1986. – №330. – С.103-107.
77. Приходько С.И. Принцип приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам. Часть I. / Приходько С.И., Столяров А.С. // Специальная техника средств связи. – МО СССР, 1988. – №3. – С.14-16.
78. Приходько С.И. Принцип приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам. Часть II / Приходько С.И., Столяров А.С. // Специальная техника средств связи. – МО СССР, 1988. – №4. – С.25-29.
79. Приходько С.И. Приведение двоичных сверточных кодов к недвоичным суженным циклическим кодам / Приходько С.И., Снисаренко А.Г. // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1989. – №90. – С.80-86.
80. Приходько С.И. Приведение сверточных кодов к кодам РС / Приходько С.И. // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1989. – №91. – С.81-84.
81. Приходько С.И. Приведение ортогонализируемых сверточных кодов к квазиортогональным / Приходько С.И., Березняков Г.Е. // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1990. – №8. – С.76-81.
82. Приходько С.И. Приведение ортогональных сверточных кодов к квазиортогональным сверточным кодам / Приходько С.И., Березняков Г.Е . // Радиотехника. Республиканский межведомственный научно-технический сборник. – Х.: ХИРЭ, 1990. – №83. – С.65-69.
83. Приходько С.И. Принцип приведения ортогональных сверточных кодов к квазиортогональным сверточным кодам / Приходько С.И., Гусев С.А., Сидоренко Н.Ф. // Системы информационного взаимодействия: сб. науч. труд. – Х.: НАНУ, ПАНИ, ХВУ, 1996. – С.83-88.

84. Приходько С.И. Циклические сверточные коды / Приходько С.И., Гусев С .А. // Управление и связь: сб. науч. труд. – Х.: НАНУ, ПАНИ, ХВУ, 1996. – С.98-101.
85. Приходько С.И. Принцип последовательного декодирования обобщенно заданных сверточных кодов / Приходько С.И. // Системы обработки информации: сб. науч. труд. – Х.: НАНУ, ПАНИ, ХВУ, 1998. – С.67-71.
86. Приходько С.И. Алгебраическое представление сверточных кодов / Приходько С.И. // Вестник международного славянского университета. – Х.: НАНУ, 1998. – Вып. 3. – С.72-75.
87. Приходько С.И. Алгебраическое кодирование сверточных кодов / Приходько С.И. // Информатика: сб. науч. труд. – К.: Наукова Думка, 1998. – Вып. 5. – С.72-75.
88. Приходько С.И. Алгоритм построения сверточных кодов / Приходько С.И. // Информационные системы: сб. науч. труд. – Х.: НАНУ, 1998. – Вып.1(9). – С.75-82.
89. Приходько С.И. Построение сверточных кодов / Приходько С.И. // Информационные системы: сб. науч. труд. – Харьков: НАНУ, ПАНИ, ХВУ, 1998. – Вып.1(19). – С. 144-146.
90. Приходько С.И. Алгебраические сверточные коды / Приходько С.И. // Информационно-управляющие системы на железнодорожном транспорте. – Х.: ХарГАЖТ, 1999. – №2(17). – С. 62-63.
91. Приходько С.И. Алгебраическое построение несистематических сверточных кодов / Приходько С.И., Гусев С.А., Кужель И.Е. // Системи обробки інформації. – Х.: ХВУ, 2004. – Вип. 8(36). – С. 170-175.
92. Приходько С.И. Алгебраический метод сверточного кодирования / Приходько С.И., Гусев С.А., Кужель И.Е. // Комп’ютерні системи та інформаційні технології. – Х.: ХАИ, 2005. – №1. – С.35-43.
93. Тимочко А.И. Алгебраический метод построения сверточных кодов в систематическом виде / Тимочко А.И., Приходько С.И., Постольный А.С. // Східно-Европейський журнал передових технологій. – Х.: Технологічний центр, 2005 – № 2/2(14). – С. 118-123.
94. Тимочко А.И. Алгебраический метод построения рекурсивных сверточных кодов для стандартов космической связи / Тимочко А.И., Приходько С.И., Постольный А.С. // Авиационно-космическая техника и технология. – Х.: ХАИ, 2005. – №1(17). – С. 78-86.
95. Тимочко А.И. Алгебраические рекурсивные сверточные коды и схемы турбокодирования / Тимочко А.И., Приходько С.И., Постольный А.С. // Інформаційно-керуючі системи на залізничному транспорті. – Х.: УкрДАЗТ, 2005. – №1-2. – С. 59-65.
96. Алгебраическое декодирование сверточных кодов / Приходько С.И., Гусев С.А., Постольный А.С, Жученко А.С. // Інформаційно-керуючі системи на залізничному транспорті. – Х.: УкрДАЗТ, 2005. – №6. – С. 29-37.

97. Комбинированный метод декодирования алгебраических сверточных кодов / Приходько С.И., Гусев С.А., Постольный А.С., Жученко А.С. // Інформаційно-керуючі системи на залізничному транспорті. – Х.: УкрДАЗТ, 2006. – №2 (58). – С. 8-15.
98. Приходько С.И. Оценка нижней границы свободного кодового расстояния алгебраически заданных сверточных кодов / Приходько С.И. // Системи обробки інформації. – Х.: ХУПС, 2007. – Вип. 5(65). – С. 120 – 124.
99. Приходько С.И. Метод декодирования алгебраических сверточных кодов / Приходько С.И. // Системи обробки інформації. – Х.: ХУПС, 2008. – Вип . 2(69). – С. 93 – 96.
100. Итеративное декодирование турбокодов на основе алгебраических рекурсивных сверточных кодов / Приходько С.И., Северинов А.В., Жученко А.С., Постольный А.С. // Збірник наукових праць. – Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. – Київ: НАН України, 2005. – Вип. 32. – С. 178 – 183.
101. Пат. 14180 Україна, МПК (2006) H03M 13/00. Спосіб опису пристроїв кодування нерекурсивних згорнених кодів / Приходько С.І., Постольний О.С., Гусєв С.А., Жученко О.С., Кужель І.Є.; заявник і власник Харківський університет Повітряних Сил. – № u 2005 08658; заявл. 12.09.2005; опубл. 15.05.2006, Бюл. № 5.
102. Пат. 14179 Україна, МПК (2006) H03M 13/00. Спосіб опису пристроїв кодування рекурсивних згорнених кодів / Приходько С.І., Постольний О.С ., Гусєв С.А., Жученко О.С., Кужель І.Є.; заявник і власник Харківський університет Повітряних Сил. – № u 2005 08657; заявл. 12.09.2005; опубл. 15.05.2006, Бюл. №5.
103. Пат. 14181 Україна, МПК (2006) H03M 13/00. Спосіб опису пристроїв кодування згорнених кодів / Приходько С.І., Гусєв С.А., Жученко О.С., Кужель І.Є.; заявник і власник Харківський університет Повітряних Сил. – № u 2005 08661; заявл. 12.09.2005; опубл. 15.05.2006, Бюл. №5.
104. Приходько С.И. Алгебраический метод сверточного кодирования / Приходько С.И., Гусев С.А. // Современные методы кодирования в электронных системах. Материалы международной НТК 26-27 октября 2004. – Сумы: СМКЭС, 2004. – С.49-50.
105. Приходько С.И. Алгебраические сверточные коды / Приходько С.И., Гусев С.А., Кужель И.Е. // Перша науково-технічна конференція Харківського університету Повітряних Сил: тези доповідей, 16-17 лютого 2005 р. – Х.: ХУПС, 2005. – С. 210 – 211.
106. Приходько С.И. Метод декодирования алгебраических сверточных кодов / Приходько С.И. // Матеріали четвертої наукової конференції Харківського університету Повітряних Сил ім. Івана Кожедуба, 16-17 квітня 2008 р. – Х.: ХУПС, 2008. – С. 149-150.
107. Приходько С.И. Алгебраический метод построения сверточных кодов для повышения помехоустойчивости передачи дискретных сообщений / Приходько С.И. // Перспективи розвитку озброєння і військової техніки в

Збройних Силах України. Збірка тез доповідей Першої Всеукраїнської науково-практичної конференції, 4-5 березня 2008 р. – Львів: ЛІСВ НУ “ЛП”, 2008. – С. 215.
108. Приходько С.И. Исследование свойств алгебраически заданных сверточных кодов / Приходько С.И. // Матеріали міжнародної науково-технічної конференції «Стратегії ІТ – технологій в освіті, економіці та екології». – Х.: ХНУ, 2007. – С. 78-79.
109. Приходько С.И. Исследование корректирующих свойств алгебраических сверточных кодов / Приходько С.И. // Міжнародна науково-технічна конференція «Інтегровані комп’ютерні технології в машинобудуванні «ІКТМ – 2007»: тези доповідей. – Х.: НАКУ «ХАІ», 2007. – С.428-429.
110. Приходько С.И. Оценка свободного кодового расстояния алгебраических сверточных кодов / Приходько С.И. // Проблеми інформатики і моделювання. Матеріали сьомої міжнародної науково-технічної конференції 22 листопада – 1 грудня 2007 р. – Х.: НТУ «ХПІ», 2007. – С. 13-14.
111. Приходько С.И. Алгебраические процедуры декодирования сверточных кодов / Приходько С.И. // Современные методы кодирования в электронных системах. Материалы международной НТК 23-24 апреля 2002 г. – Сумы: СМКЭС, 2002. – С.11–12.
112. Приходько С.И. Особенности алгебраических самоортогональных сверточных кодов в частотной области / Приходько С.И., Волков А.С. // 22 международная научно-практическая конференция «Перспективные компьютерные, управляющие и телекоммуникационные системы для железнодорожного транспорта Украины». – Алушта. – 2009.
113. Розробка методів та програмних засобів підвищення достовірності та своєчасності передачі даних у телекомунікаційній системі АСУ Військ Протиповітряної Оборони Збройних Сил України комплексу засобів автоматизації «Ореанда»: звіт про НДР (проміжний). Шифр “Алгоритм” / [Приходько С.І., Кузнецов О.О., Кужель І.Є. та ін.]. – Х.: ХУПС, 2005. – 381 с. – № держреєстрації 0101U000413.
114. Розробка методів підвищення якості військового зв’язку АСУ ракетних військ та артилерії: звіт про НДР (заключний). Шифр «Мрія» / [Стасєв Ю. В., Приходько С.І., Грабчак В.І. та ін.]. – Харків: ХУПС, 2005. – 133 с. – № держреєстрації 0101U000414.
115. Скляр Б. Цифровая связь. Теоретические основы и практическое применение / Бернард Скляр [пер. с англ. Е.Г. Гроза, В.В. Марченко, А.В. Назаренко, О.М. Ядренко под. ред. А.В. Назаренко]. – М.: Издат. дом «Вильямс», 2003. – 1104 с.
116. Долгов В.И. Основы статистической теории приема дискретных сигналов / Долгов В.И. – Х.: ХВВКИУ РВ, 1989. – 448 с.
117. Зюко А.Г. Помехоустойчивость и эффективность систем связи / Зюко А .Г. – М.: Связь, 1972. – 360 с.

118. Помехоустойчивость и эффективность систем передачи информации / А.Г. Зюко, А.И. Фалько, И.П. Панфилов и др. – М.: Радио и связь, 1985. – 272 с.
119. Юрлов Ф.Ф. Технико-экономическая эффективность сложных радиоэлектронных систем / Юрлов Ф.Ф. – М.: Сов. радио, 1980. – 280 с.
120. ITU-T Y.1540. Internet protocol data communication service - IP packet transfer and availability performance parameters, 2007.
121. ITU-T Y.1541. Network Performance objectives for IP-based services, Amendment 3, 2008.
122. Поставной В.И. Теория передачи сигналов / Поставной В.И. – МО СССР, 1985. – 264 с.
123. Уидроу Б. Адаптивная обработка сигналов / Уидроу Б., Стирз С. – М.: Радио и связь, 1989. – 440 с.
124. Варакин Л.Е. Теория сложных сигналов / Варакин Л.Е. – М.: Сов. Радио, 1970. – 376 с.
125. Варакин Л.Е. Системы связи с шумоподобными сигналами / Варакин Л. Е. – М.: Радио и связь, 1985. – 384 с.
126. Передача информации с обратной связью / Под ред. З.М. Каневского. – М.: Связь, 1976. – 350 с.
127. Каневский З.М. Передача сообщений с информационной обратной связью / Каневский З.М. – М.: Радио и связь, 1969. – 263 с.
128. Мартынов Ю.М. Обработка информации в системах передачи данных / Мартынов Ю.М. – М.: Связь, 1969. – 263 с.
129. Barbulescu A. Terminating the trellis of turbo-codes in the same state / Barbulescu A., Pietrobon S. // Electronics Letters. – January 1995. – Vol. 31. – P. 22 – 23.
130. Seghers S. On the free distance of turbo codes and related product codes: PhD Thesis / Seghers S. – Swiss Federal Institute of Technology. – Zurich, Switzerland, 1995. – 196 p. – Режим доступу: http://citeseer.nj.nec.com.
131. Cederval M. A fast algorithm for computing distance spectrum of convolutional codes / Cederval M., Johannesson R. // IEEE Trans. Inform. Theory. – November 1989. – IT–35. – P. 1146 – 1159.
132. Муттер В.М. Основы помехоустойчивой телепередачи информации / Муттер В.М. – Л.: Энергоатомиздат. – Ленингр. отд-ние, 1990. – 288 с.
133. Жученко А.С. Метод итеративного декодирования турбокодов уменьшенной сложности в телекоммуникационных системах: дис. кандидата техн. наук: 05.12.02 / Жученко Александр Сергеевич. – Х., 2005. – 201 с.
134. Hagenauer J. Viterbi algorithm with soft–decision outputs and its applications / Hagenauer J., Hoeher P. // Proc. IEEE GLOBLECOM. – Dallas, USA. – November 1989. – P. 1680 – 1686.
135. Hagenauer J. Source-controlled channel coding / Hagenauer J. // IEEE Trans . Commun. – September 1995. – Vol. 43. – P. 2449 – 2457.
136. Hagenauer J. Iterative decoding of binary block and convolutional codes / Hagenauer J., Offer E., Papke L. // IEEE Transactions On Information Theory.

– 1996. – Vol. 42. – P. 429 – 445.
137. Вентцель Е.С. Теория вероятностей и ее инженерные приложения / Вентцель Е.С. Овчаров Л.А. – М.: Наука, 1988. – 480 с.
138. Пугачев В.С. Теория вероятностей и математическая статистика / Пугачев В.С. – М.: Наука, 1979. – 496 с.

Приложение А

Описание программного макета алгебраического построения сверточных кодов

Программный макет алгебраического построения сверточных кодов реализован в среде программирования Delphi и предназначен для экспериментального подтверждения полученных научных результатов, для проведения практических и лабораторных работ по теории помехоустойчивого кодирования. Программный макет позволяет алгебраически строить сверточные коды с заранее заданными конструктивными свойствами, имитировать работу кодера и проводить тестирование сверточного кода по кодовому расстоянию.

Общий вид интерфейса разработанного программного макета представлен на рис. А.1. Работа программного макета осуществляется следующим образом. В окно ввода «GF(2^m), m =» в верхней части интерфейса вводится степень расширения двоичного поля Галуа, над которым задается код РС. В окно ввода «t =» вводится исправляющая способность кода РС.

После нажатия в нижней части интерфейса кнопки «Поле» в центральной таблице выводятся все элементы конечного поля Галуа $GF(2^m)$ в виде двоичных коэффициентов соответствующих многочленов.

После нажатия кнопки «Код РС» в поле вывода «g(x) = » выводится порождающий многочлен кода РС с параметрами: $n = 2^m - 1$, $k = n - 2t$, $d = 2t + 1$, где n и t соответствуют введенным выше значениям. Коэффициенты многочлена $g(x)$ соответствуют номерам строк в таблице элементов поля $GF(2^m)$.

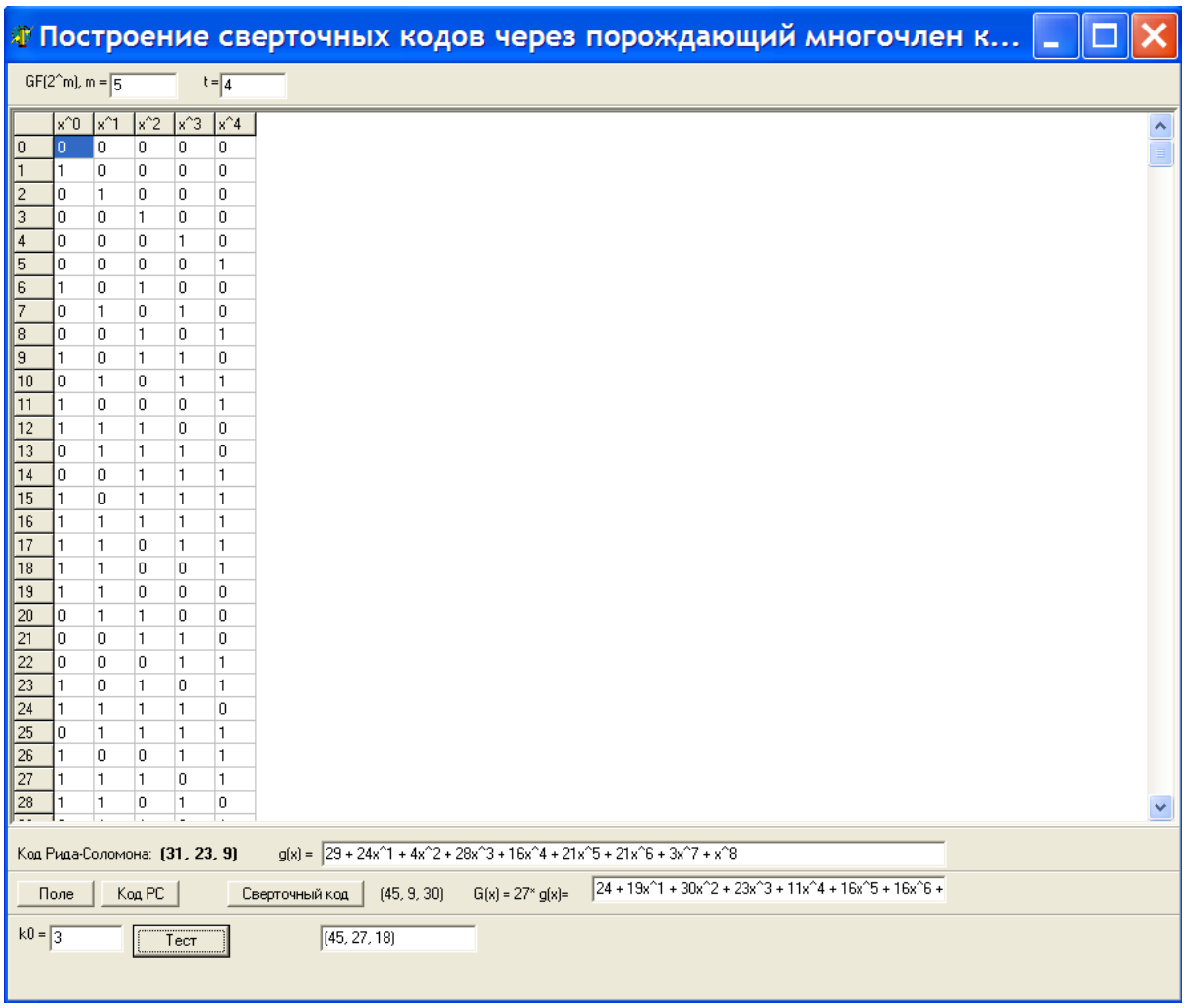


Рис. А.1. Интерфейс программного макета алгебраического построения сверточных кодов

После нажатия кнопки «Сверточный код» в поле вывода «G(x) = b*g(x) =» выводится порождающий многочлен сверточного кода, как произведение порождающего многочлена кода РС на множитель «b». Множитель „b" формируется путем полного перебора по всему множеству элементов поля $GF(2^m)$ с целью формирования такого $G(x) = bg(x)$, который давал бы наибольшее число ненулевых элементов при отображении недвоичных коэффициентов многочлена $G(x)$ в двоичное поле. Одновременно с выводом многочлена $G(x) = bg(x)$ в поле вывода «n, k, d» рядом с кнопкой «Сверточный код» выводятся параметры алгебраически заданного сверточного кода со скоростью кодирования $R = 1/m$. Выводимое кодовое расстояние рассчитывается как число ненулевых элементов при отображении недвоичных коэффициентов многочлена $G(x)$ в двоичное поле. Это соответствует случаю подачи на вход кодера одного единичного символа и, как показывает проведенный анализ, проведенная таким образом приближенная оценка обеспечивает высокую достоверность.

Для алгебраического построения сверточных кодов со скоростью $R = k^0/m$ необходимо в окно ввода «k0 =» ввести длину информационного блока. После нажатия кнопки «Тест» в поле вывода (n, k, d) выводятся

соответствующие параметры алгебраического сверточного кода. Кодовое расстояние трестируется путем подсчета ненулевых элементов при отображении недвоичных коэффициентов многочлена *fG(x)* в двоичное поле, где *f* – пробегает все элементы поля $GF(2^m)$ первые $m - k^0$ коэффициентов которых (в полиномиальной записи) равны нулю. Это соответствует подаче на вход кодера набора ненулевых элементов из множества *H* (см. раздел 2) путем полного перебора элементов этого множества.

На рис. А.1. приведен пример алгебраического построения сверточного кода, заданного порождающим многочленом

$$g(x) = 29 + 24x^1 + 4x^2 + 28x^3 + 16x^4 + 21x^5 + 21x^6 + 3x^7 + x^8$$

кода РС (31, 23, 9) над конечным полем $GF(2^5)$. Кодовое расстояние удовлетворяет условию: $d \geq D = 9$, конструктивные кодовые параметры *(45, 27, 9), v = 24*. Предсказанное минимальное расстояние равно

.

В результате тестирования получено значения $d_m = 18$.

Ниже приводится фрагмент исходного кода разработанной программы.

```
unit Unit1;
interface
uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, ExtCtrls, StdCtrls, Pole, Grids;
type
MyType=array[0..100] of integer;
TForm1 = class(TForm)
Panel1: TPanel;
Panel2: TPanel;
Panel3: TPanel;
PoleGaluaExtend1: TPoleGaluaExtend;
Label1: TLabel;
Edit1: TEdit;
Label2: TLabel;
Edit2: TEdit;
Button1: TButton;
Panel4: TPanel;
Button2: TButton;
Button3: TButton;
ScrollBox1: TScrollBox;
StringGrid1: TStringGrid;
Label4: TLabel;
Label5: TLabel;
Edit3: TEdit;
Label3: TLabel;
Label6: TLabel;
Edit4: TEdit;
Panel5: TPanel;
Label7: TLabel;
Edit5: TEdit;
Button4: TButton;
Label8: TLabel;
Edit6: TEdit;
procedure Button1Click(Sender: TObject);
procedure UmnKorn;
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);//процедура умножения одночленов -
вычисление коэффициентов многочлена
private
{ Private declarations }
public
{ Public declarations }
end;
```

```
var Form1: TForm1;
Mnogochlen:MyType;//параметры многочлена: каждый элемент
//массива соответствует коэффициенту при соответствующей степени, всего
до 100 степени
Mnogochlen_svert:MyType;//параметры многочлена сверточного кода -
произведение "Mnogochlen" на "Mn"
Mnogochlen_Korni:MyType;//корни многочлена - элементы поля  (a^i + 1)
Step_Mnogochlen:integer;//степень многочлена
Dsk:integer;//свободное кодовое расстояние
k0:integer;//длина информационного кадра
Mn:integer;//для этого множителя (порождающего многочлена кода РС)
implementation
{$R *.dfm}
procedure TForm1.UmnKorn;//процедура умножения одночленов - вычисление
коэффициентов многочлена
var i,j:integer; Step:integer;//служ. переменная - текущая степень многочлена
Mnog:MyType;//служебная переменная - параметры многочлена (его
коэффициенты)
begin
for i:=0 to 100 do
Mnogochlen[i]:=0;//начальные обнуления
//первый корень
Mnogochlen[0]:=Mnogochlen_Korni[0];
Mnogochlen[1]:=1;
//со второго корня
for i:=1 to Step_Mnogochlen-1 do//по всем корням многочлена
begin
Mnog:=Mnogochlen;
//умножение многочлена на формальную переменную 'x'
Mnogochlen[0]:=0;
for j:=1 to Step_Mnogochlen do
begin
Mnogochlen[j]:=0;
if Mnog[j-1]<>0 then
Mnogochlen[j]:=Mnog[j-1];
end;
//умножение исходного многочлена на текущий корень и добавление его к
многочлену, умноженному на формальную переменную
for j:=0 to Step_Mnogochlen do
begin
if Mnog[j]<>0 then
Mnogochlen[j]:=PoleGaluaExtend1.SumExt(Mnogochlen[j],PoleGaluaExtend1.
UmnExt(Mnog[j],Mnogochlen_Korni[i]));
end; end; end;
procedure TForm1.Button1Click(Sender: TObject);
```

```
var i,j:integer;
begin
PoleGaluaExtend1.VvodExt(2,StrToInt(Edit1.text));//ввод параметров
расширенного поля
PoleGaluaExtend1.SintPoleExtend;//построение расширенного поля
//вывод элементов расширенного поля
with StringGrid1 do
begin
colcount:=PoleGaluaExtend1.PoleExtend.m+1;//число колонок
rowcount:=PoleGaluaExtend1.PoleExtend.n+1;//число строк
for i:=1 to RowCount do//название строк
cells[0,i]:=IntToStr(i-1);
for i:=1 to ColCount do//название колонок
cells[i,0]:='x^'+IntToStr(i-1);
for i:=1 to ColCount do
for j:=1 to RowCount do//заполнение ячеек таблицы
cells[i,j]:=IntToStr(PoleGaluaExtend1.PoleExtend.ArrExtDec[i-1][j-1]);
end; end;
procedure TForm1.Button2Click(Sender: TObject);
label 1;
var i,j:integer;
MyStr:string;
begin
Step_Mnogochlen:=2*StrToInt(Edit2.text);//ввод степени порождающего
многочлена кода Рида-Соломона
if Step_Mnogochlen>PoleGaluaExtend1.PoleExtend.n-1 then
begin
Application.MessageBox('Число корней больше мощности поля. Уменьшите
степень многочлена.', 'Такой код нельзя построить!', MB_OK);
goto 1;//безусловный переход - конец процедуры
end;
for i:=1 to Step_Mnogochlen do//ввод корней многочлена
Mnogochlen_Korni[i-1]:=i;
UmnKorn;//формирование порождающего многочлена
//вывод порождающего многочлена кода Рида-Соломона
MyStr:=IntToStr(Mnogochlen[0])+' + ';
for i:=1 to Step_Mnogochlen-1 do
begin
if Mnogochlen[i]<>0 then//если не нулевой коэффициент при i-ой степени
begin
if Mnogochlen[i]>1 then//если не нулевой коэффициент при i-ой степени
MyStr:=MyStr+IntToStr(Mnogochlen[i])+'x^'+IntToStr(i)+' + '
else
MyStr:=MyStr+'x^'+IntToStr(i)+' + ';
end; end;
```

```
if Mnogochlen[i]>1 then//если не нулевой коэффициент при i-ой степени
MyStr:=MyStr+IntToStr(Mnogochlen[i])+'x^'+IntToStr(i)
else
MyStr:=MyStr+'x^'+IntToStr(i);
Edit3.Text:=MyStr;
label5.Caption:='('+IntToStr(PoleGaluaExtend1.PoleExtend.n-1)+', '+IntToStr(
PoleGaluaExtend1.PoleExtend.n-1-Step_Mnogochlen)+', '+IntToStr(Step_
Mnogochlen+1)+')';
1:end;
procedure TForm1.Button3Click(Sender: TObject);
var i,j,k,MyVar:integer; Mnog,Mnog1:MyType;//служебная переменная -
параметры многочлена (его коэффициенты)
A:array[1..1024]of integer;//служебная переменная - i-ый єлемент массива
соответствует числу единиц при отображении порождающего многочлена*i в
двоичное поле
MyStr:string;
begin
for i:=1 to 1024 do
A[i]:=0;
Dsk:=0;//начальные обнуления
for i:=1 to PoleGaluaExtend1.PoleExtend.n do
begin
Mnog:=Mnogochlen;
MyVar:=0;
for j:=0 to Step_Mnogochlen do//умножение всех коэффициентов многочлена
на i-ый элемент поля
begin
Mnog[j]:=PoleGaluaExtend1.UmnExt(Mnog[j],i);
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[j]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end; end;
A[i]:=MyVar;
if A[i]>Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=A[i];
Mn:=i;
Mnog1:=Mnog;
end; end;
Mnogochlen_svert:=Mnog1;//выбранный многочлен для сверточного кода
Label6.Caption:='('+IntToStr((Step_Mnogochlen+1)*PoleGaluaExtend1.
PoleExtend.m)+', '+IntToStr(Step_Mnogochlen+1)+', '+IntToStr(Dsk)+')       '+'G(
```

```
x) = '+IntToStr(Mn)+'* g(x)=';
//вывод порождающего многочлена кода Рида-Соломона, умноженного на (
Mn)
MyStr:=IntToStr(Mnog1[0])+' + ';
for i:=1 to Step_Mnogochlen-1 do
begin
if Mnog1[i]<>0 then//если не нулевой коэффициент при i-ой степени
begin
if Mnog1[i]>1 then//если не нулевой коэффициент при i-ой степени
MyStr:=MyStr+IntToStr(Mnog1[i])+'x^'+IntToStr(i)+' + '
else
MyStr:=MyStr+'x^'+IntToStr(i)+' + ';
end; end;
if Mnog1[i]>1 then//если не нулевой коэффициент при i-ой степени
MyStr:=MyStr+IntToStr(Mnog1[i])+'x^'+IntToStr(i)
else
MyStr:=MyStr+'x^'+IntToStr(i);
Edit4.Text:=MyStr;
end;
procedure TForm1.Button4Click(Sender: TObject);
var iiii,jjjj,kkkk,iii,jjj,kkk,i,j,k,ii,jj,kk,l,ll,lll,MyVar,MyVar1:integer;
Mnog,Mnog1,Mnog2,Mnog3,Mnog4:MyType;//служебная переменная -
параметры многочлена (его коэффициенты)
A:array[1..1024]of integer;//служебная переменная - i-ый єлемент массива
соответствует числу единиц при отображении порождающего многочлена*i в
двоичное поле
MyStr:string; MyBoolSL:Boolean;//служебная булева переменная
begin
Dsk:=10000000;//начальные обнуления
k0:=StrToInt(edit5.text);
Mnog:=Mnogochlen_svert;//многочлен сверточного кода
//одна единица
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k
,i]<>0 then
MyBoolSL:=true;
end;
```

```
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
end;
for j:=0 to Step_Mnogochlen do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog1[j]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end;
//две единицы
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
end;
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
```

```
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for j:=1 to Step_Mnogochlen+1 do
begin
Mnog[j]:=PoleGaluaExtend1.SumExt(Mnog1[j],Mnog2[j-1]);
end;
for j:=0 to Step_Mnogochlen+1 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[j]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end; end; end;
//две единицы через одну
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k
,i]<>0 then
```

```
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
end;
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for j:=2 to Step_Mnogochlen+2 do
begin
Mnog[j]:=PoleGaluaExtend1.SumExt(Mnog1[j],Mnog2[j-2]);
end;
for j:=0 to Step_Mnogochlen+2 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[j]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для различных множителей порождающего многочлена (Mn)
begin
```

```
Dsk:=MyVar;
end; end; end; end; end;
//две единицы через две
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k
,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
end;
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for j:=3 to Step_Mnogochlen+3 do
begin
```

```
Mnog[j]:=PoleGaluaExtend1.SumExt(Mnog1[j],Mnog2[j-3]);
end;
for j:=0 to Step_Mnogochlen+3 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[j]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end;
end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end; end; end;
//три единицы
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k
,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
```

```
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for iii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog3:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kkk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kkk,iii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jjj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog3[jjj]:=PoleGaluaExtend1.UmnExt(Mnog3[jjj],iii);
end;
for jjj:=2 to Step_Mnogochlen+2 do
begin
Mnog[jjj]:=PoleGaluaExtend1.SumExt(PoleGaluaExtend1.SumExt(Mnog1[jjj],
Mnog2[jjj-1]),Mnog3[jjj-2]);
end;
for jjj:=0 to Step_Mnogochlen+2 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[jjj]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end; end;
```

```
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end;  end;    end; end; end; end;
//три единицы, третья через одну
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
```

```
for iii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog3:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kkk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kkk,iii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jjj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog3[jjj]:=PoleGaluaExtend1.UmnExt(Mnog3[jjj],iii);
end;
for jjj:=3 to Step_Mnogochlen+3 do
begin
Mnog[jjj]:=PoleGaluaExtend1.SumExt(PoleGaluaExtend1.SumExt(Mnog1[jjj],
Mnog2[jjj-1]),Mnog3[jjj-3]);
end;
for jjj:=0 to Step_Mnogochlen+3 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[jjj]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end;
end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end;  end; end; end; end; end;
//три единицы, вторая через одну, третья еще через одну
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
```

```
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k
,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for iii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog3:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kkk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kkk,iii]<>0 then
MyBoolSL:=true;
```

```
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for jjj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
begin
Mnog3[jjj]:=PoleGaluaExtend1.UmnExt(Mnog3[jjj],iii);
end;
for jjj:=4 to Step_Mnogochlen+4 do
begin
Mnog[jjj]:=PoleGaluaExtend1.SumExt(PoleGaluaExtend1.SumExt(Mnog1[jjj], Mnog2[jjj-2]),Mnog3[jjj-4]);
end;
for jjj:=0 to Step_Mnogochlen+4 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[jjj]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end;  end; end; end; end; end;
//четыре единицы
for i:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog1:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for k:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-k,i]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем порождающий многочлен на этот элемент
begin
for j:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов многочлена на i-ый элемент поля
```

```
begin
Mnog1[j]:=PoleGaluaExtend1.UmnExt(Mnog1[j],i);
for ii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog2:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kk,ii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog2[jj]:=PoleGaluaExtend1.UmnExt(Mnog2[jj],ii);
end;
for iii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
Mnog3:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kkk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kkk,iii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jjj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog3[jjj]:=PoleGaluaExtend1.UmnExt(Mnog3[jjj],iii);
end;
for iiii:=1 to PoleGaluaExtend1.PoleExtend.n-1 do
begin
MyVar:=0;
```

```
Mnog4:=Mnogochlen_svert;//многочлен сверточного кода
MyBoolSL:=false;
for kkkk:=0 to PoleGaluaExtend1.PoleExtend.m-k0-1 do//проверка на нули в
первых k0 коэффициентах многочлена
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[PoleGaluaExtend1.PoleExtend.m-1-
kkkk,iiii]<>0 then
MyBoolSL:=true;
end;
if MyBoolSL=false then  //если в первых k0 символах все нули, то умножаем
порождающий многочлен на этот элемент
begin
for jjjj:=0 to Step_Mnogochlen-1 do//умножение всех коэффициентов
многочлена на i-ый элемент поля
begin
Mnog4[jjjj]:=PoleGaluaExtend1.UmnExt(Mnog4[jjjj],iiii);
end;
for jjjj:=3 to Step_Mnogochlen+3 do
begin
Mnog[jjjj]:=PoleGaluaExtend1.SumExt(PoleGaluaExtend1.SumExt(
PoleGaluaExtend1.SumExt(Mnog1[jjjj],Mnog2[jjjj-1]),Mnog3[jjjj-2]),Mnog4[jjjj-
3]);
end;
for jjjj:=0 to Step_Mnogochlen+2 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog[jjjj]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end; end; end;  end; end; end; end; end; end; end;
Mnog:=Mnogochlen_svert;
Mnog1:=Mnogochlen_svert;
//Проверка на бесконечную серию единиц
MyVar1:=0;
for i:=0 to Step_Mnogochlen do
MyVar1:=PoleGaluaExtend1.SumExt(MyVar1,Mnog[i]);
if MyVar1=0 then//если сумма всех коэффициентов равна нулю, то вычисляем
сумму многочленов
begin
```

```
for i:=1 to Step_Mnogochlen-1 do
begin
for j:=1 to Step_Mnogochlen do
Mnog1[j]:=PoleGaluaExtend1.SumExt(Mnog1[j],Mnog[j-i]);
end; end;
//Посчет единиц
for i:=0 to Step_Mnogochlen-1 do
begin
for k:=0 to PoleGaluaExtend1.PoleExtend.m-1 do
begin
if PoleGaluaExtend1.PoleExtend.ArrExtDec[k][Mnog1[i]]=1 then
MyVar:=MyVar+1;//подсчет ненулевых элементов отображенных в двоичное
поле коэффициентов
end; end;
if MyVar<Dsk then //выбор максимального свободного расстояния (Dsk) для
различных множителей порождающего многочлена (Mn)
begin
Dsk:=MyVar;
end;
Label8.Caption:='('+IntToStr((Step_Mnogochlen+1)*PoleGaluaExtend1.
PoleExtend.m)+', '+IntToStr((Step_Mnogochlen+1)*k0)+', '+IntToStr(Dsk)+')';
Edit6.Text:='('+IntToStr((Step_Mnogochlen+1)*PoleGaluaExtend1.PoleExtend.m
)+', '+IntToStr((Step_Mnogochlen+1)*k0)+', '+IntToStr(Dsk)+')';
end; end.
```

Приложение Б

Акты реализации диссертационных исследований

ЗАТВЕРДЖУЮ
Заступник начальника Харківського університету Повітряних Сил імені Івана Кожедуба з навчальної та наукової роботи
доктор технічних наук, професор
полковник СТАСЄВ Ю.В.
„12" 04 2005 р.

АКТ

реалізації результатів наукових досліджень дисертаційної роботи Приходька Сергія Івановича при проведенні науково-дослідних робіт, що виконувалися у науково-дослідній лабораторії Харківського університету Повітряних Сил імені Івана Кожедуба

Комісія у складі:
Голови – заступника начальника кафедри „Комп'ютерних систем" кандидата технічних наук, доцента, підполковника Сєвєрінова О.В.
та **членів комісії**:
- начальника науково-дослідної лабораторії кафедри „Комп'ютерні системи" кандидата технічних наук, старшого наукового співробітника майора Кузнецова О.О.;
- старшого наукового співробітника науково-дослідної лабораторії кафедри „Комп'ютерні системи" кандидата технічних наук капітана Гинєвського О.М.

ВСТАНОВИЛА: що при розробці спеціального математичного та програмного забезпечення програмно-апаратного макету перешкодостійкого кодеру (декодеру), виконаного в межах науково-дослідних робіт: "Розробка методів та програмних засобів підвищення достовірності та своєчасності передачі даних в телекомунікаційній системі АСУ ВППО ЗС України КЗА "Ореанда", шифр "Алгоритм"; "Розробка методів підвищення якості військового зв'язку автоматизованої системи управління ракетних військ і артилерії", шифр "Мрія" впроваджені такі результати наукових досліджень Приходька Сергія Івановича.

1. Метод ітеративного декодування турбокодів з алгебраїчно заданими рекурсивними згортувальними кодами, що відрізняється від відомого узагальненим поданням нескінченного кодового слова згортувального коду через нескінченну суму послідовних наборів з кодових слів циклічного коду, що дозволяє за рахунок зведення декодування згортувального коду до декодування послідовності кодових слів циклічного коду декодувати турбокоди на основі алгебраїчно заданих згортувальних кодів з високими конструктивними кодовими характеристиками.

2. Обчислювально ефективні (обчислювально реалізуємі) методи с
алгебраїчно заданих згортувальних кодів, що відрізняються від в
використанням обмеження недвійкового циклічного коду на д
підполе, що дозволяє синтезувати алгебраїчно задані згортувальні
довільними властивостями й кодовими характеристиками.

3. Методи синтезу паралельних каскадних згортувальних конст
(методи турбокодування), що відрізняються від відомих використ
алгебраїчно заданих рекурсивних згортувальних кодів, що д
аналітично зв'язати параметри турбокодів з параметрами алгебраїчно з
рекурсивних згортувальних кодів і синтезувати паралельні к
згортувальні конструкції із заданими конструктивними ко
характеристиками.

Голова комісії:
Заступник начальника кафедри „Комп'ютерні системи" кандидат тех
наук, доцент
підполковник Сєвєрінов О.В.

Члени комісії:
Начальник науково-дослідної лабораторії кафедри „Комп'ютерні систе
кандидата технічних наук, старший науковий співробітник
майор Кузнецов О.О.

Старший науковий співробітник науково-дослідної лабораторії к
„Комп'ютерні системи"
кандидат технічних наук
капітан Гинєвський О.М.

Складений акт заслухано та обговорено на засіданні к
„Комп'ютерні системи" (каф. №702) Харківського універ
Повітряних Сил, протокол № 9/2004-2005 від 11.04.2005 р.

ЗАТВЕРДЖУЮ

Директор Центрального казенного конструкторського бюро «ПРОТОН»

В.С. КУЗНІЧЕНКО

"____" _____ 2008 р.

АКТ

реалізації результатів наукових досліджень дисертаційної роботи Приходька Сергія Івановича при проведенні науково-дослідних робіт, що виконувалися у ЦККБ «ПРОТОН»

Комісія, що призначена Наказом директора ЦККБ "Протон" № 133 від 22.05.2008 року у складі:

– **Голови** – заступника директора ЦККБ «ПРОТОН» з наукової роботи кандидата технічних наук, доцента Харченка В.М.

та **членів комісії**:

– начальника науково-дослідного відділу № 3 кандидата технічних наук, доцента Голобородька Ю.М.

– начальника науково-дослідного відділу № 2 Романенка Ю.М.

ВСТАНОВИЛА: що при розробці спеціального математичного та програмного забезпечення програмно-апаратного макету завадостійкого кодеру (декодеру), виконаного в межах ескізно-технічного проекту на дослідно-конструкторську роботу "Жанр-Р", що виконувалась на підставі "Плану НДДКР в інтересах Міністерства Оборони України" впроваджені такі результати наукових досліджень завідувача кафедри "Транспортний зв'язок" кандидата технічних наук, доцента Приходька Сергія Івановича.

1. Використано імітаційну модель підсистеми передачі інформації із застосуванням алгебраїчно заданих згорткових кодових конструкцій, які, за рахунок відсутності обмежень при виборі необхідних параметрів синтезованих згорткових кодових конструкцій, забезпечують можливість підвищення достовірності інформації що передається в каналах із помилками. Алгоритми декодування згорткових кодових конструкцій з високими конструктивними кодовими характеристиками мають параметри, близькі до теоретично граничних значень.

2. Надані рекомендації по використанню турбокодів із синтезованими алгебраїчно заданими згортковими кодами забезпечують імовірності помилки на біт $10^{-5} \div 10^{-6}$ при значенні енергетичного відношення сигнал/шум 1,5 – 2 дБ. Для забезпечення імовірності помилки на біт

$10^{-8} \div 10^{-9}$ швидкість кодування рекомендовано вибрати не менше ніж застосовувати турбокоди з кількістю елементів пам'яті 6 – 8.

Складений акт заслухано, обговорено та схвалено на науково-тех нараді Центрального казенного конструкторського бюро «ПРОТОН»

Голова комісії:
Заступник директора ЦККБ «ПРОТОН» з наукової роботи
кандидат технічних наук, доцент

Харченко В.М.

Члени комісії:
начальник науково-дослідного відділу № 3
кандидат технічних наук, доцент

Голобородько Ю.М.

начальник науково-дослідного відділу № 2

Романенко Ю.М.

ЗАТВЕРДЖУЮ
Перший проректор Української державної академії залізничного транспорту
В.М.Астахов
"5" 04 2008 р.

АКТ
реалізації результатів наукових досліджень дисертаційної роботизавідувача кафедри „Транспортний зв'язок" кандидата технічних наук, доцента, Приходька Сергія Івановича

Комісія у складі голови – професора кафедри "Транспортний зв'язок" Української державної академії залізничного транспорту кандидата технічних наук, професора, Єлізаренка О.В., членів комісії – доцента кафедри "Транспортний зв'язок" кандидата технічних наук, доцента Батаєва О.П., доцента кафедри "Транспортний зв'язок" кандидата технічних наук, доцента Кириченка М.П. склала цей акт про те, що при розробці лекційних, практичних та лабораторних занять з навчальної дисципліни "Системи та мережі передачі дискретних повідомлень" за темою „Завадостійке кодування" (лекція "Завадостійке кодування повідомлень", лекція "Лінійні блочні коди", практичне заняття „Будування кодерів та декодерів лінійних кодів", практичне заняття "Кодувальні та декодувальні пристрої лінійних кодів") у навчальному процесі Української державної академії залізничного транспорту були використані наступні результати наукових досліджень Приходька Сергія Івановича:

1. Методи кодування алгебраїчно заданими згортувальними кодами, що відрізняються від відомих теоретично обґрунтованими процедурами алгебраїчної побудови рекурсивних і перекурсивних згортувальних кодів через узагальнення циклічних кодів на випадок нескінченної довжини, що дозволяє аналітично формалізувати процес завадостійкого кодування синтезованими згортувальними кодами з високими конструктивними кодовими характеристиками.

2. Обчислювально ефективні (обчислювально реалізуємі) методи синтезу алгебраїчно заданих згортувальних кодів, що відрізняються від відомих використанням обмеження недвійкового циклічного коду на довільне підполе, що дозволяє синтезувати алгебраїчно задані згортувальні коди з довільними властивостями й кодовими характеристиками.

Застосування результатів дисертаційних досліджень Приходька Сергія Івновича дозволило підвищити рівень засвоєння навчального матеріалу з дисципліни "Теорія інформації" за рахунок більш поглибленого вивчення

сучасних та перспективних методів завадостійкого кодування інфор які використовуються в телекомунікаційних системах.

Голова комісії

Професор кафедри "Транспортний зв'язок" Української держ академії залізничного транспорту

кандидат технічних наук, професор

О.В. ЄЛІЗАРЕНКО

Члени комісії:

Доцент кафедри "Транспортний зв'язок"

кандидат технічних наук, доцент

О.П. БАТАЄВ

Доцент кафедри "Транспортний зв'язок"

кандидат технічних наук, доцент

М.П. КИРИЧЕНКО

УКРАЇНСЬКА ДЕРЖАВНА АКАДЕМІЯ
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ

Приходько Сергій Іванович

УДК 621.391

МЕТОДИ СИНТЕЗУ, КОДУВАННЯ ТА ДЕКОДУВАННЯ ЗГОРТКОВИХ КОДОВИХ КОНСТРУКЦІЙ

05.12.02 – Телекомунікаційні системи та мережі

Автореферат дисертації на здобуття наукового
ступеня доктора технічних наук

Харків – 2010

Дисертацією є рукопис.
Робота виконана в Українській державній академії залізничного транспорту Міністерства транспорту і зв'язку України

Науковий консультант: доктор технічних наук, професор
Сорока Леонід Степанович, Харківський національний університет імені В.Н. Каразіна, декан факультету комп'ютерних наук.

Офіційні опоненти: доктор технічних наук, професор
Захарченко Миколай Васильович, Одеська национальна академия звязку ім. О.С. Попова, проректор з навчальної роботи;

доктор технічних наук, професор
Лосєв Юрій Іванович, Харківський національний університет імені В.Н. Каразіна, професор кафедри теоретичної та прикладної системотехніки;

доктор технічних наук, доцент
Климаш Михайло Миколайович, Національний університет «Львівська політехніка», професор кафедри телекомунікацій.

Захист відбудеться «_____» _____ 2010 року о ____ годин на засіданні спеціалізованої вченої ради Д 64.820.01 Українській державній академії залізничного транспорту, 61050, м. Харків, вул. Фейєрбаха 7.

З дисертацією можна ознайомитись у бібліотеці Української державній академії залізничного транспорту, 61050, м. Харків, вул. Фейєрбаха 7.

Автореферат розісланий «_____» _____ 2010 р.

Вчений секретар
спеціалізованої вченої ради _____ Г.В. Альошин

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. В умовах ринкових відносин одним з важливих завдань є підвищення ефективності функціонування галузей економіки за рахунок необхідної координації роботи всіх управлінських підрозділів. Рішення цього завдання здійснюється шляхом застосування автоматизованих систем керування різного рівня й призначення.

Структурними елементами сучасних автоматизованих систем керування є підсистеми передачі інформації, що здійснюють обмін інформацією між джерелами й споживачами інформації з каналів зв'язку. Одним з основних вимог до підсистеми передачі є забезпечення заданої достовірності переданої інформації, що безпосередньо впливає на ефективність процесу управління.

У зв'язку з ростом вимог до ефективності процесів управління, постійним збільшенням обсягу й швидкості передачі інформації істотно зростають вимоги й до достовірності переданої інформації.

Відповідно до Концепції розвитку зв'язку України сучасний рівень розробки й виробництва технічних засобів зв'язку неможливо забезпечити без проведення випереджальних досліджень. При цьому одними з основних напрямків випереджальних досліджень є розвиток математичного супроводження, аналізу й синтезу нових, структурно складних систем і мереж зв'язку; розробка нових технологій і принципів побудови систем зв'язку, насамперед у сфері обробки й передачі інформації, а також їхніх складових частин. Таким чином, дослідження, спрямовані на розробку засобів підвищення достовірності переданої інформації є актуальними (перспективними).

Основними й найбільш ефективними засобами підвищення достовірності переданої інформації є методи завадостійкого кодування. У теорії завадостійкого кодування можна виділити кілька основних напрямків розвитку.

Перший напрямок базується на блокових кодах й, переважно, алгебраїчних методах подання процесів синтезу, кодування й декодування. Найбільше поширення серед блокових кодів одержав великий клас кодів - циклічні коди. Поряд з високими конструктивними властивостями циклічних кодів цей напрямок дозволяє будувати прості й обчислювально ефективні алгоритми кодування й декодування.

Другий напрямок розвитку базується на безперервних кодах, підкласом яких є згорткові коди. Відмінною рисою згорткових кодів є можливість їхнього простого опису деревом або регулярною ґратчастою діаграмою, що дозволяє реалізувати імовірнісне декодування (алгоритми послідовного декодування, алгоритм Вітербі, алгоритм максимуму апостеріорної ймовірності). Кодер згорткового коду являє собою лінійний регістр зсуву, складність якого із-за регулярної ґратчастої діаграми не залежить від довжини коду (але залежить від числа станів ґратчастої діаграми), що є значною перевагою.

Відповідно до теореми Шенона, найбільшу ефективність мають довгі коди. Циклічні коди при великій довжині кодового слова не дозволяють істотно підвищити енергетичну ефективність, що пояснюється, насамперед, їх незадовільними асимптотичними властивостями. Тому із цього погляду згорткові коди є більш кращими, тому що при їхньому використанні ефективність кодування не

погіршується з зростанням довжини кодового слова. Однак конструктивні кодові характеристики згорткових кодів (кодова відстань) залежать від числа станів ґратчастої діаграми (яка експоненціально залежить від кількості елементів пам'яті кодера згорткового коду - лінійного регістра зсуву), що призводить до збільшення складності декодування згорткових кодів з високими кодовими характеристиками, через необхідність аналізу всіх станів кодової решітки у процесі декодування. Крім того, у цей час відсутні обчислювально ефективні методи синтезу згорткових кодів із заданими конструктивними кодовими характеристиками (як правило, для пошуку згорткових кодів використовуються переборні методи).

Як третій напрямок можна виділити методи каскадного кодування, поява яких пов'язана зі спробами синтезу довгих кодів з високими кодовими характеристиками на основі досить простих складових кодів (які можуть бути як блоковими, так і згортковими), декодування яких здійснюється окремими декодерами. Перевага каскадних кодів полягає в спрощенні алгоритмів декодування з одночасним підвищенням загальної ефективності кодування. Каскадні коди дозволяють забезпечити високу достовірність в умовах великого рівня шуму при помірній складності декодування. Подальше вдосконалювання методів каскадного кодування призвело до розробки турбокодів - паралельних каскадних рекурсивних згорткових кодів.

Реалізація турбокодування інформації блоками великої довжини не являє собою значних труднощів через використання складових згорткових кодів, оскільки складність згорткового кодування не залежить від довжини інформаційної послідовності, що кодується. У результаті, турбокоди можуть забезпечити ефективність кодування, близьку до теоретично граничного значення, визначеного теоремою Шенона. Недоліком існуючих турбокодів є зменшення ефективності кодування при високому енергетичному відношенні сигнал/шум, що пов'язане з малою мінімальною відстанню складових турбокод згорткових кодів.

Видимим шляхом усунення недоліків турбокодів є використання в якості складових турбокод кодів рекурсивних згорткових кодів з високими конструктивними характеристиками, що призведе до підвищення мінімальної відстані турбокоду й дозволить вибирати швидкість турбокодування без обмежень. Однак перешкодою на шляху застосування складових турбокод згорткових кодів із указаними властивостями є висока складність декодування згорткових кодів з високими конструктивними характеристиками. Крім того, як показано в відомих роботах методи синтезу згорткових кодів не дозволяють ефективно будувати рекурсивні згорткові коди з високими конструктивними властивостями (великою кодовою відстанню), що стримує розробку й впровадження перспективних систем турбокодування.

З вищесказаного можна зробити висновок, що розвинена в цей час алгебраїчна теорія блокового кодування не може бути безпосередньо застосована до згорткових кодів через значну різницю в їхніх властивостях у порівнянні із блоковими кодами. Незважаючи на це відомо, що існує можливість представлення згорткового коду у вигляді блокового коду напівнескінченної довжини і його наступним алгебраїчним описом. Цей напрямок теорії завадостійкого кодування одержав розвиток у роботах автора при написанні кандидатської дисертації. Однак позитивні результати в

розглянутих роботах отримані тільки для обмеженого діапазону низьких швидкостей кодування, значення яких не задовольняють сучасним вимогам до параметрів завадостійких кодів (як правило, на практиці потрібні більш високі швидкості кодування). Крім того, у цих роботах не розглядається можливість застосування алгебраїчної теорії для реалізації декодування згорткових кодів. Таким чином, виникає наукова проблема (суперечлива ситуація), у якій існуючі положення теорії завадостійкого кодування не дозволяють обчислювально реалізуємо вирішувати завдання синтезу, кодування й декодування згорткових кодів з високими конструктивними кодовими характеристиками й з довільними параметрами. Вирішення наукової проблеми (суперечливої ситуації) можливо шляхом розробки на основі єдиного концептуального підходу методів синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій з необхідними властивостями й характеристиками.

Актуальність теми дисертаційних досліджень визначається необхідністю забезпечення заданої достовірності переданої інформації шляхом застосування процедур синтезу, кодування й декодування алгебраїчно заданих згорткових кодів (кодових конструкцій) з високими конструктивними кодовими характеристиками, що можуть бути обчислювально реалізовані.

Зв’язок роботи з науковими програмами, планами, темами. Дослідження в дисертаційній роботі проводилися у відповідності з наступними нормативними актами.

1. Концепція Національної програми інформатизації, схвалена Законом України «Про Концепцію Національної програми інформатизації» від 4 лютого 1998 р. N 75/98-ВР.
2. Концепція розвитку зв'язку України до 2010 року, затверджена постановою Кабінету Міністрів України «Про Концепцію розвитку зв'язку України до 2010 року» від 9 грудня 1999 р. №2238.
3. Державна науково-технічна програма «Створення перспективних телекомунікаційних систем і технологій».
4. Концепція створення Державної інтегрованої інформаційної системи забезпечення управління рухомими об'єктами (зв'язок, навігація, спостереження), схвалена розпорядженням Кабінету Міністрів України від 17 липня 2003 р. N 410-р.

Мета і завдання дослідження. Метою дисертаційної роботи є розробка концептуального підходу на основі нових методів синтезу, кодування й декодування згорткових кодових конструкцій з використанням математичного апарату алгебраїчної теорії завадостійкого кодування для підвищення достовірності переданої інформації.

Для досягнення поставленої мети необхідно вирішити наступні наукові завдання.

1. Розробити й дослідити методи синтезу алгебраїчно заданих згорткових кодових конструкцій для підвищення достовірності переданої інформації:

– розробити (з використанням математичного апарата алгебраїчної теорії завадостійкого кодування) методи синтезу алгебраїчно заданих згорткових кодів, теоретично обґрунтувати аналітичні вирази по оцінці кодових співвідношень синтезованих кодів;

– розробити методи й алгоритми кодування алгебраїчно заданими згортковими кодами, дослідити конструктивні властивості синтезованих згорткових кодових конструкцій.
2. Розробити й дослідити обчислювально ефективні (такі, що можуть бути обчислювально реалізовані) методи й алгоритми декодування алгебраїчно заданих згорткових кодів:
– розробити алгебраїчний метод декодування синтезованих згорткових кодів;
– розробити комбінований метод декодування алгебраїчно заданих згорткових кодів , що поєднує в собі процедури переборного пошуку по кодовій решітці й алгебраїчні процедури локалізації й виправлення помилок;
– розробити алгоритми декодування алгебраїчно заданих згорткових кодів і пропозиції по програмній й апаратній реалізації.
3. Розробити паралельні каскадні згорткові кодові конструкції на основі алгебраїчно заданих рекурсивних згорткових кодів і алгоритмів їхнього декодування, що можуть бути обчислювально реалізовані:
– аналітично формалізувати й розробити методи синтезу турбокодів з використанням алгебраїчно заданих згорткових кодів;
– розробити й дослідити алгоритми ітеративного декодування паралельних каскадних кодових конструкцій з алгебраїчно заданими згортковими кодами;
– розробити й дослідити алгоритми м'якого декодування складових турбокод алгебраїчно заданих згорткових кодів.
4. Розробити практичні рекомендації з використання алгебраїчних згорткових кодів у телекомунікаційних системах і мережах:
– розробити (з використанням методів математичної статистики й перевірки гіпотез) імітаційну модель системи передачі інформації, методику оцінки й дослідити достовірність переданої інформації в телекомунікаційних системах і мережах з використанням алгебраїчно заданих згорткових кодів і турбокодів на їхній основі;
– обґрунтувати практичні рекомендації з використання алгебраїчних згорткових кодів у телекомунікаційних системах і мережах.
Об'єкт дослідження. Процес підвищення достовірності переданої інформації на основі застосування алгебраїчно заданих згорткових кодових конструкцій.
Предмет дослідження. Методи й алгоритми синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій.
Методи дослідження. Розробка й дослідження алгебраїчних методів і процедур синтезу, кодування й декодування згорткових кодових конструкцій проведені з використанням методів алгебраїчної теорії завадостійкого кодування, теорії полів Галуа й теорії чисел. Оцінка достовірності переданої інформації проведена з використанням методів статистичної теорії зв'язку, теорії імовірності й математичної статистики. Розробка рекомендацій з реалізації кодерів алгебраїчно заданих згорткових кодів проведена з використанням методів теорії цифрових автоматів.
Наукова новизна отриманих результатів. Новим науковим результатом дисертації є розвиток теорії завадостійкого кодування в частині синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій (з довільними

кодовими характеристиками й властивостями). У рамках головного нового наукового результату отриманий ряд часткових наукових результатів.

1. Одержав подальший розвиток єдиний концептуальний підхід алгебраїчного представлення згорткових кодів у вигляді недвійкових блокових циклічних кодів (напівнескінченної довжини), що відрізняється від відомого (теоретичним узагальненням на випадок напівнескінченної довжини кодового слова циклічного коду й) використанням породжувальних багаточленів недвійкових циклічних кодів, обмежених на довільне підполе, що дозволяє розглядати з єдиних теоретичних позицій процеси синтезу, кодування й декодування згорткових кодів з довільними властивостями й кодовими характеристиками й теоретично обґрунтувати аналітичні вирази по оцінці кодових співвідношень синтезованих згорткових кодових конструкцій, аналітично зв'язати їхні параметри й виразити через кодові характеристики відповідних циклічних кодів.

2. Одержали подальший розвиток обчислювально ефективні (такі, що можуть бути обчислювально реалізовані) алгебраїчні методи синтезу (алгебраїчно заданих) згорткових кодів, що відрізняються від відомих використанням обмеження недвійкового циклічного коду на довільне підполе, що дозволяє синтезувати (алгебраїчно задані) згорткові коди з довільними властивостями й кодовими характеристиками.

3. Одержали подальший розвиток методи кодування алгебраїчно заданими згортковими кодами, що відрізняються від відомих теоретично обґрунтованими процедурами алгебраїчної побудови рекурсивних і нерекурсивних згорткових кодів через узагальнення циклічних кодів на випадок нескінченної довжини, що дозволяє аналітично формалізувати процес завадостійкого кодування синтезованими згортковими кодами з високими (конструктивними) кодовими характеристиками.

4. Уперше розроблені алгебраїчний і комбінований методи декодування алгебраїчно заданих згорткових кодів, які відрізняються від відомих методів процедурами алгебраїчної локалізації й прискорених процедур (алгоритмами) послідовного пошуку, що дозволяє реалізувати обчислювально ефективне (таке, що може бути обчислювально реалізоване) декодування безперервних кодових конструкцій з великою довжиною кодового обмеження (з більшою кодовою відстанню) для підвищення достовірності переданої інформації.

5. Одержали подальший розвиток методи синтезу паралельних каскадних згорткових конструкцій (методи турбокодування), що відрізняються від відомих використанням алгебраїчно заданих рекурсивних згорткових кодів, що дозволяє аналітично зв'язати параметри турбокодів з параметрами алгебраїчно заданих рекурсивних згорткових кодів і синтезувати паралельні каскадні згорткові конструкції із заданими (конструктивними кодовими) характеристиками.

6. Одержав подальший розвиток метод ітеративного декодування турбокодів з алгебраїчно заданими рекурсивними згортковими кодами, що відрізняється від відомого узагальненим представленням нескінченного кодового слова згорткового коду через нескінченну суму послідовних наборів з кодових слів циклічного коду, що дозволяє за рахунок зведення декодування згорткового коду до декодування послідовності кодових слів циклічного коду декодувати турбокоди на основі алгебраїчно заданих згорткових кодів з великою кількістю елементів пам'яті (з

високими кодовими характеристиками, високою кодовою відстанню).

Практичне значення отриманих результатів досліджень полягає в наступному.

1. Розроблені алгоритми синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій з необхідними (кодовими) характеристиками, такі що можуть бути обчислювально реалізовані.
2. Розроблено методику (емпіричної) оцінки достовірності переданої інформації, що дозволяє для (заданих параметрів математичної моделі) дискретно-безперервного каналу із заданою погрішністю оцінити ймовірність помилкового прийому біта інформації й відповідний енергетичний виграш від кодування.
3. Розроблено імітаційну модель системи передачі інформації з використанням алгебраїчно заданих згорткових кодових конструкцій, за допомогою якої встановлено, що

– синтезовані згорткові кодові конструкції, отримані за допомогою розроблених алгоритмів, що можуть бути обчислювально реалізовані, не поступаються по енергетичних характеристиках відомим у цей час кодам; їхнє практичне використання дозволяє забезпечити підвищення достовірності переданої інформації в каналах з випадково виникаючими помилками за рахунок відсутності обмежень при виборі необхідних параметрів синтезованих згорткових кодових конструкцій;

– розроблені обчислювально реалізуємі алгоритми декодування згорткових кодових конструкцій з високими конструктивними кодовими характеристиками мають параметри близькі до теоретично граничних значень.

4. Розроблено практичні рекомендації з використання турбокодів із синтезованими алгебраїчно заданими згортковими кодами. Для забезпечення ймовірності помилки на біт при значенні енергетичного відношення сигнал/шум 1,5 – 2 дБ, пропонується використовувати турбокоди з кількістю елементів пам'яті 2 – 4. Для забезпечення ймовірності помилки на біт пропонується використовувати турбокоди з кількістю елементів пам'яті 6 – 8. Швидкість кодування не рекомендується вибирати менш ніж 1/3.
5. Отримані результати використані в науково-дослідних роботах «Мрія», «Алгоритм» (Харківський університет Повітряних Сил, акт реалізації від 12.04.2005), на виробництві при розробці спеціального математичного та програмного забезпечення програмно-апаратного макету завадостійкого кодеру (декодеру) у ЦККБ «Протон» (акт реалізації від 26.05.2008) і в навчальному процесі Української державної академії залізничного транспорту (акт реалізації від 15.04.2008).

Таким чином, отримані в ході досліджень наукові й практичні результати в сукупності вирішують важливу наукову проблему шляхом розробки на основі нового концептуального підходу методів синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій з необхідними властивостями й характеристиками, що має велике значення як для розвитку окремого напрямку теорії завадостійкого кодування, так і для рішення прикладних питань, пов'язаних із забезпеченням заданої достовірності переданої інформації в телекомунікаційних системах і мережах.

Особистий внесок здобувача. Всі результати, викладені в дисертаційній роботі, отримані автором самостійно. У роботах, виконаних у співавторстві й опублікованих у виданнях, які ввійшли в перелік ВАК України, автору належать: у роботі [1] запропонований принцип приведення двійкових згорткових кодів до недвійкових звужених циклічних кодів; у роботі [2] пропонується подальший розвиток способу приведення двійкових згорткових кодів до недвійкових звужених циклічних кодів; у роботі [3] розроблений алгоритм приведення двійкових згорткових кодів до недвійкових звужених циклічних кодів; у роботі [4] розроблений метод приведення згорткових кодів до кодів Рида-Соломона; в [5] запропонований підхід приведення ортогоналізуємих згорткових кодів до квазіортогональних; у роботі [6] запропонований підхід для приведення ортогональних згорткових кодів до квазіортогональних згорткових кодів; у роботі [7] запропонований метод приведення ортогональних згорткових кодів до квазіортогональних згорткових кодів; в [8] досліджені можливості представлення згорткових кодів за допомогою циклічних кодів; у роботі [9] розроблений принцип послідовного декодування узагальнено заданих згорткових кодів; у роботі [15] запропоновано використовувати породжувальні багаточлени недвійкових циклічних кодів, обмежених на довільне підполе, для алгебраїчної побудови несистематичних згорткових кодів; у роботі [16] розроблений алгебраїчний метод згорткового кодування; у роботі [17] розроблений алгебраїчний метод побудови систематичних згорткових кодів; у роботі [18] розроблений алгебраїчний метод побудови рекурсивних згорткових кодів; у роботі [19] розроблені методи синтезу паралельних каскадних згорткових конструкцій на основі алгебраїчно заданих рекурсивних згорткових кодів; у роботі [20] розроблений метод декодування алгебраїчно заданих згорткових кодів; у роботі [21] розроблені процедури алгебраїчної локалізації й прискорені процедури послідовного пошуку для комбінованого методу декодування алгебраїчно заданих згорткових кодів; у роботі [24] розроблений метод ітеративного декодування турбокодів на основі алгебраїчно заданих рекурсивних згорткових кодів.

Апробація результатів дисертації. Основні результати дисертації доповідалися й були схвалені на наступних науково-технічних конференціях:

– Міжнародна науково-технічна конференція «Сучасні методи кодування в електронних системах» (Суми, 2002);

– Міжнародна науково-технічна конференція «Сучасні методи кодування в електронних системах» (Суми, 2004);

– Перша науково-технічна конференція Харківського університету Повітряних Сил (Харків, 2005);

– Міжнародна науково-технічна конференція «Інтегровані комп'ютерні технології в машинобудуванні» (Харків, 2007);

– Міжнародна науково-технічна конференція «Стратегії ІТ-технологій в освіті, економіці й екології» (Харків, 2007);

– Сьома міжнародна науково-технічна конференція «Проблеми інформатики й моделювання» (Харків, 2007);

–Четверта наукова конференція Харківського університету Повітряних Сил (Харків , 2008);

–Перша Всеукраїнська науково-практична конференція (Львів, 2008);

–22 міжнародна науково-практична конференція «Перспективні комп'ютерні, керуючі й телекомунікаційні системи для залізничного транспорту України» (Алушта, 2009).

Публікації. Основні положення дисертаційної роботи викладені в 24 наукових статтях, 3 патентах, 9 тезах доповідей, 2 звітах з НДР.

Структура та обсяг дисертації. Дисертація складається зі вступу, шести розділів, висновку та додатків. Повний обсяг дисертації складає 319 сторінки, у тому числі 2 додатка на 34 сторінках, 77 рисунків, 13 таблиць, перелік використаних літературних джерел складається з 138 найменувань на 14 сторінках.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У **вступі** обґрунтовується актуальність теми, формулюються мета і завдання дослідження, вказується наукова новизна, практичне значення та впровадження отриманих результатів.

У першому розділі проведений аналіз загальних положень теорії завадостійкого кодування, представлена модель системи передачі інформації через сукупність формальних операторів, що аналітично описує процедури перетворення інформації, проведені аналіз і порівняльні дослідження відомих методів завадостійкого кодування. Узагальнені й теоретично обґрунтовані шляхи підвищення достовірності переданої інформації на основі використання завадостійких кодів. На основі отриманих результатів обґрунтовується вибір напрямку досліджень, уводяться критерії й показники ефективності, математично формалізується постановка наукової проблеми.

Однієї з найважливіших характеристик системи передачі інформації є достовірність переданої інформації. Основним показником достовірності є ймовірність правильного прийому біта . Частіше використовують зворотний показник - імовірність помилкового прийому біта . Очевидно, що показники достовірності залежать тільки від відношення енергії, що приходиться на один біт до спектральної щільності потужності шуму й застосовуваних методів модуляції й кодування.

З достовірністю переданої інформації тісно зв'язана інша характеристика системи передачі інформації – завадостійкість. Кількісною мірою завадостійкості є мінімально необхідне для забезпечення необхідної достовірності співвідношення енергії сигналу до спектральної щільності потужності шуму. Тобто цей показник дозволяє при фіксованому рівні достовірності ( ) оцінити (порівняти між собою) енергетичну ефективність різних систем передачі інформації.

Побудова ефективної системи передачі інформації зв'язана з оптимізацією цілого спектра взаємозалежних і суперечливих вимог: максимізація швидкості передачі інформації й мінімізація ймовірності появи бітової помилки; мінімізація споживаної потужності й/або відношення енергії сигналу до спектральної щільності потужності шуму й мінімізація ширини смуги пропущення; зниження складності практичної реалізації й максимізація числа абонентів мережі зв'язку.

Уведемо два додаткових показники, що характеризують питому складність реалізації застосовуваних процедур кодування й модуляції:
- мінімальне число операцій, які необхідно виконати для реалізації цифрової обробки інформаційних повідомлень, що приходяться на один переданий біт даних , операцій/біт (асимптотична часова складність реалізації);
- мінімальне число елементів пам'яті, необхідне для реалізації цифрової обробки інформаційних повідомлень, що приходяться на один переданий біт даних , елементів/біт (асимптотична ємнісна складність реалізації).

Побудова ефективних завадостійких високошвидкісних систем передачі інформації на рівні сигнально-кодових конструкцій математично формалізуємо у вигляді цільової функції:

. (1)

Аналіз виразу (1) показує, що побудова оптимальних сигнально-кодових конструкцій зв'язана з рішенням багатокрітеріального оптимізаційного завдання з обліком розглянутих вище природних теоретичних обмежень на граничну швидкість передачі інформації й мінімально необхідну смугу пропущення. Рішення зазначеного завдання існуючими методами неможливо у зв’язку з її надзвичайно високої складністю.

Розглянемо постановку сформульованого вище завдання в умовах прийнятих у рамках проведення дослідження допущень й обмежень. Припустимо, що для передачі інформації використовуються канали зв'язку з фіксованою смугою пропущення , а швидкість передачі визначається із граничного теоретичного співвідношення про мінімальну смугу пропущення біт/с, де - потужність алфавіту сигналів, обумовлена системою модуляції й кодування.

Зафіксуємо величину , як величину, задану відповідними нормативними документами, що встановлює вимоги до якості цифрового зв'язку. Мінімально необхідне співвідношення енергії біта до спектральної щільності потужності шуму , необхідне для забезпечення заданої ймовірності , задає величину завадостійкості системи передачі інформації. Таким чином, завдання побудови ефективних завадостійких високошвидкісних систем передачі з урахуванням прийнятих у рамках проведення досліджень допущень й обмежень запишемо у вигляді:

, (2)

тобто для фіксованої смуги частот і теоретично граничної швидкості передачі потрібно мінімізувати співвідношення енергії біта до спектральної щільності потужності шуму , яке необхідно для забезпечення заданого рівня достовірності , де - необхідне (максимально

припустиме) значення показника достовірності. Крім того, відповідно до постановки завдання виду (2), потрібно мінімізувати питому складність реалізації цифрової обробки інформаційних повідомлень із використанням застосовуваних сигнально-кодових конструкцій.

Завдання побудови ефективної завадостійкої високошвидкісної системи передачі інформації на рівні сигнально-кодових конструкцій будемо вирішувати за допомогою мінімізації з урахуванням прийнятих допущень й обмежень і вибору з безлічі отриманих рішень оптимального за критерієм мінімізації питомої складності реалізації цифрової обробки повідомлень ( ) рішення.

На основі проведеного аналізу, відповідно до мети дисертаційної роботи, були сформульовані завдання дослідження.

В другому розділі на основі єдиного концептуального підходу з використанням методів алгебраїчної теорії блокових кодів, теорії кінцевих полів і поліноміальних методів опису завадостійких кодів розробляються алгебраїчні методи синтезу нерекурсивних згорткових кодів, досліджуються процедури побудови нерекурсивних згорткових кодів, розробляються алгоритми для їхньої реалізації.

В основі існуючого алгебраїчного методу побудови згорткових кодів зі швидкістю $R = 1 / m$ лежить обмеження недвійкового циклічного коду над $GF(q^m)$ на підполе $GF(q)$. Для зняття обмеження по швидкості кодування розроблений алгебраїчний метод побудови згорткових кодів, в основі якого лежить обмеження недвійкового циклічного коду над $GF(q^m)$ на довільну підмножину $H \subseteq GF(q^m)$, $|H| \geq |GF(q)|$. Якщо $|H| = |GF(q)|$, то одержимо, як окремий випадок, існуючий метод алгебраїчної побудови згорткових кодів зі швидкістю $R = 1 / m$.

Розглянемо несистематичний згортковий (n, k) – код над $GF(q)$ зі швидкістю $R = k^0 / m$ (див. рис. 1). Розіб'ємо вхідну інформаційну послідовність на інформаційні кадри по $k^0 \geq 1$ символів, кожен символ яких належить $GF(q)$.

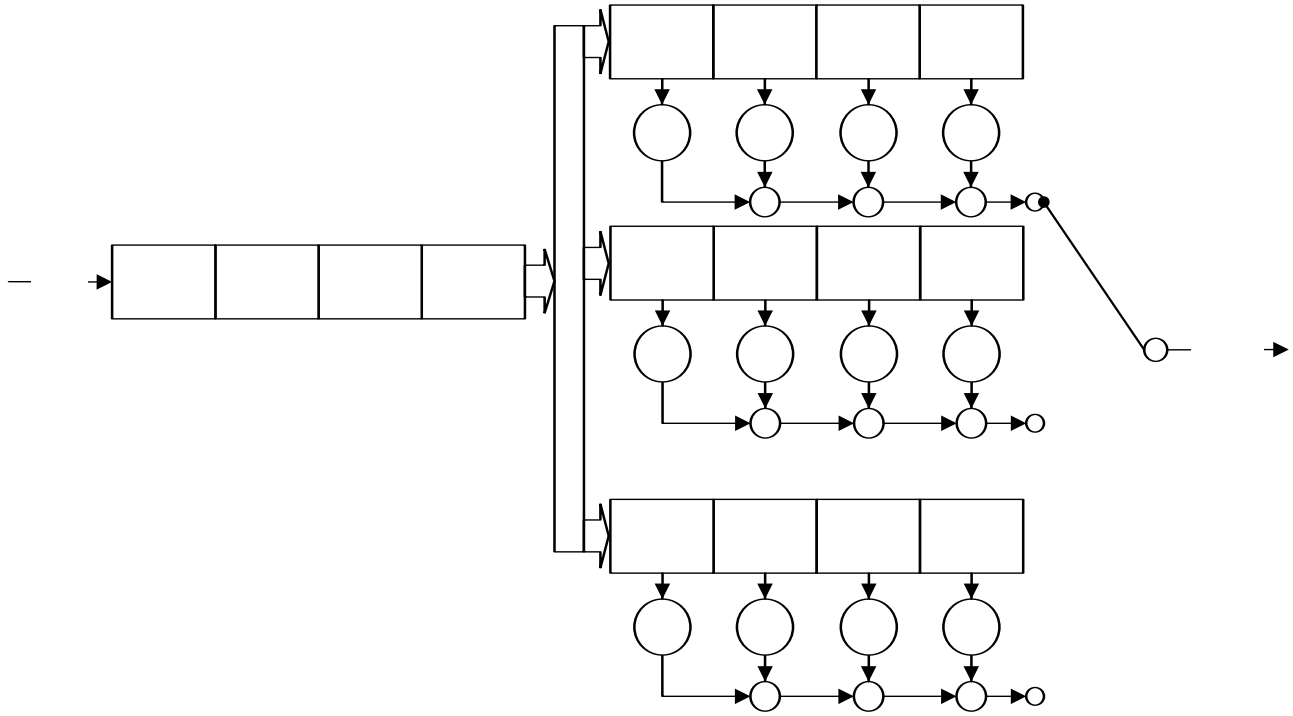


Рис. 1. Схема несистематичного згорткового кодера з $R = k^0/m$

У загальному випадку інформаційна послідовність може бути нескінченної довжини, тобто складатися з нескінченного числа інформаційних кадрів по k^0 символів. Зіставимо кожному інформаційному кадру з k^0 символів один символ з безлічі $H \subseteq GF(q^m)$, $|H| \geq |GF(q)|$. У цьому випадку інформаційний багаточлен можна представити у вигляді

, (3)

де $I_j \in H, j = 0,..., r-1$, $\log_q|H| = k^0$, $m \geq k^0$.

Нехай багаточлени $P_1(x), P_2(x), ..., P_m(x)$ – породжувальні багаточлени представленого на рис. 1 несистематичного згорткового коду. Процес кодування інформації – інформаційна послідовність $I(x)$ виду (3) надходить у кодер (рис. 1), де відбувається її множення на багаточлени $P_1(x) \ldots P_m(x)$. Одержимо послідовності $F_1(x) \ldots F_m(x)$:

;

;

…

,

де $S_{i,j}$ – коефіцієнт багаточлена $F_i(x)$ при x^j у результаті перемножування багаточлена $I(x)$ виду (3) і багаточленів $P_i(x)$.

Кодове слово $C(x)$ формується шляхом послідовного зчитування символів при однакових ступенях багаточленів $F_1(x) \ldots F_m(x)$, тобто:

(4)

Якщо на вхід згорткового кодера подати інформаційний вектор виду *{0, 0, ...,1 }*, то інформаційний багаточлен запишеться як $I(x)=1$, а кодове слово (4) запишеться у вигляді породжувального багаточлена циклічного коду, тобто $C(x) = P(x)$. Таким чином породжувальний багаточлен циклічного коду однозначно визначає несистематичне правило згорткового кодування.

Теорема 1. Породжувальний багаточлен ступеня r *(N, K, D*) циклічного коду над $GF(q^m)$ повністю визначає несистематичний згортковий *(n, k, d*) код над $GF(q)$ з кодовим обмеженням $\nu = r \cdot k^0$ і параметрами

Слід зазначити, що оцінка $d_\infty \geq D$ у теоремі 1 не точна. Тому пропонується прогнозувати вільну мінімальну відстань d_{II} несистематичного згорткового *(n, k, d)* –

коду над *GF(q)*, алгебраїчно заданого породжувальним багаточленом *(N, K, D)* циклічного коду над $GF(q^m)$, як

. (5)

Вивід виразу (5) заснований на підрахунку ненульових *q*–ічних символів у вихідній кодовій послідовності несистематичного згорткового *(n, k, d)* коду, алгебраїчно заданого за допомогою породжувального багаточлена *(N, K, D)* циклічного коду над $GF(q^m)$.

Практичне використання результатів доведених у дисертаційній роботі теорем дозволяє зв'язати конструктивні характеристики згорткового *(n, k, d)* коду над *GF(q)* з параметрами утворюючого циклічного *(N, K, D)* коду над $GF(q^m)$ з породжувальним багаточленом ступеня *r*:

$$k^0 = log\ _q| H|; n^0 = m; \nu = r \cdot k^0; k = (r + 1) \cdot k^0; n = k \cdot n^0 / k^0; d_\infty \geq D;$$
$$R = k^0 / m,\ m \geq k^0,$$

де $H \subseteq GF(q^m)$.

Алгоритм побудови згорткового *(n, k, d)* коду над *GF(q)* визначимо у вигляді послідовності таких кроків.

КРОК 1. Вибір конструктивних параметрів згорткового *(n, k, d)* коду над *GF(q)*.

КРОК 2. Розрахунок параметрів утворюючого поля $GF(q^m)$. Вибір циклічного коду, розрахунок його конструктивних *(N, K, D)* параметрів над $GF(q^m)$.

КРОК 3. Вибір породжувального багаточлена циклічного *(N, K, D*) коду $GF(q^m$). Розрахунок прогнозованої вільної відстані згорткового коду.

КРОК 4. Визначення породжувальних багаточленів несистематичного згорткового *(n, k, d*) коду, побудова схеми кодера.

КРОК 5. Уточнення мінімальної кодової відстані й вільної кодової відстані несистематичного згорткового *(n, k, d)* коду (при необхідності).

Проведені дослідження властивостей синтезованих нерекурсивних згорткових кодів, алгебраїчно заданих породжувальними багаточленами недвійкових циклічних кодів, показали, що отримані коди близькі по своїх характеристиках до оптимальних кодів. Застосування розроблених методів синтезу нерекурсивних згорткових кодів дозволяє за рахунок використання алгебраїчних процедур і поліноміальних методів опису циклічних кодів вирішити важливе наукове завдання пошуку ефективних нерекурсивних згорткових кодів з високими кодовими характеристиками.

У третьому розділі на основі єдиного концептуального підходу одержали подальший розвиток алгебраїчні методи й обчислювально ефективні алгоритми синтезу рекурсивних згорткових кодів, які використовуються в якості складових кодів турбокоду через особливості вагового розподілу кодових слів рекурсивних згорткових кодів.

У дисертаційній роботі доведений зв'язок між параметрами циклічних кодів і рекурсивних згорткових кодів з $R = k^0 / m$.

Теорема 2. Якщо зафіксувати кінцеву безліч *H* елементів поля $GF(q^m)$, причому $log\ _q| H| = k^0$, $m \geq k^0$, то перевірочний багаточлен циклічного *(N, K, D)* коду над $GF(q^m)$ повністю визначає несистематичний рекурсивний згортковий *(n, k, d)* код над *GF(q)*

з інформаційним кадром довжини k^0, довжиною кодового обмеження $\nu = K \cdot k^0$ і параметрами

Теорема 2 визначає механізм побудови алгебраїчних рекурсивних несистематичних згорткових кодів. Їхні параметри алгебраїчно пов'язані з параметрами недвійкових циклічних кодів, що дозволяє конструктивно будувати рекурсивні згорткові коди з необхідними властивостями. Загальна схема згорткового кодера наведена на рис. 2. Такий кодер реалізує обробку символів з $GF(q^m)$.
Для рішення завдання алгебраїчної побудови рекурсивних систематичних згорткових кодів скористаємося систематичними циклічними кодами. Для реалізації процедури систематичного кодування циклічного коду скористаємося цифровим фільтром з нескінченним імпульсним відкликом (рекурсивним фільтром), що реалізує ланцюг ділення на багаточлен.

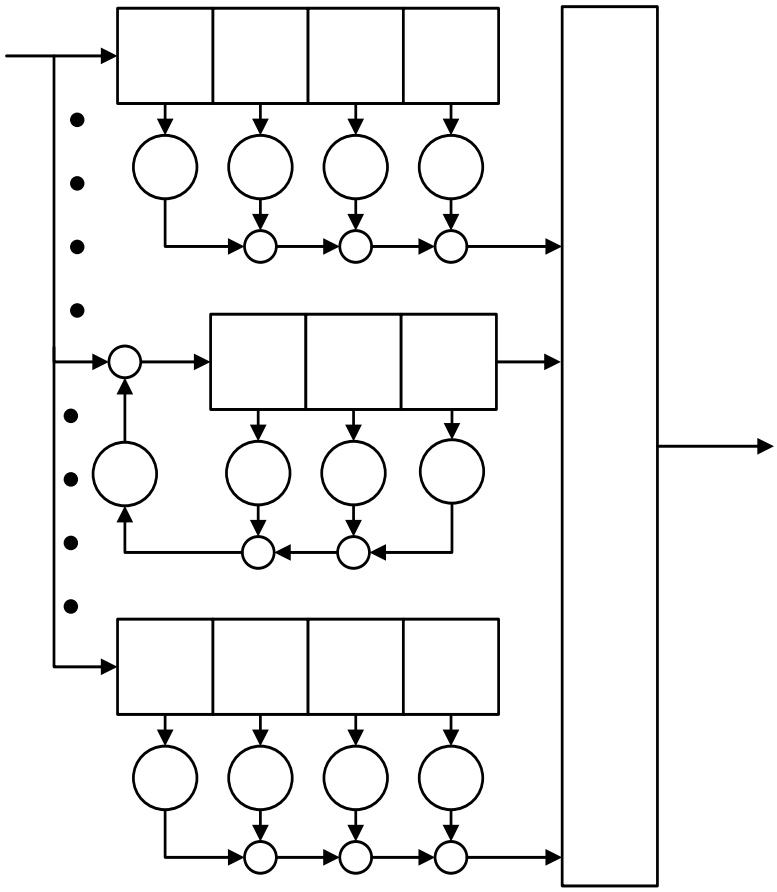


Рис. 2. Схема несистематичного кодера алгебраїчного рекурсивного згорткового коду з обробкою елементів з $GF(q)$

Нехай коефіцієнти породжувального багаточлену $g(x) = g_0 + g_1x + g_2x^2 + \ldots + g_rx^r$ дорівнюють ваговим множникам у відводах регістра зсуву рекурсивного фільтру.

Використаємо таку схему для побудови кодеру алгебраїчного систематичного рекурсивного згорткового коду (див. рис. 3).

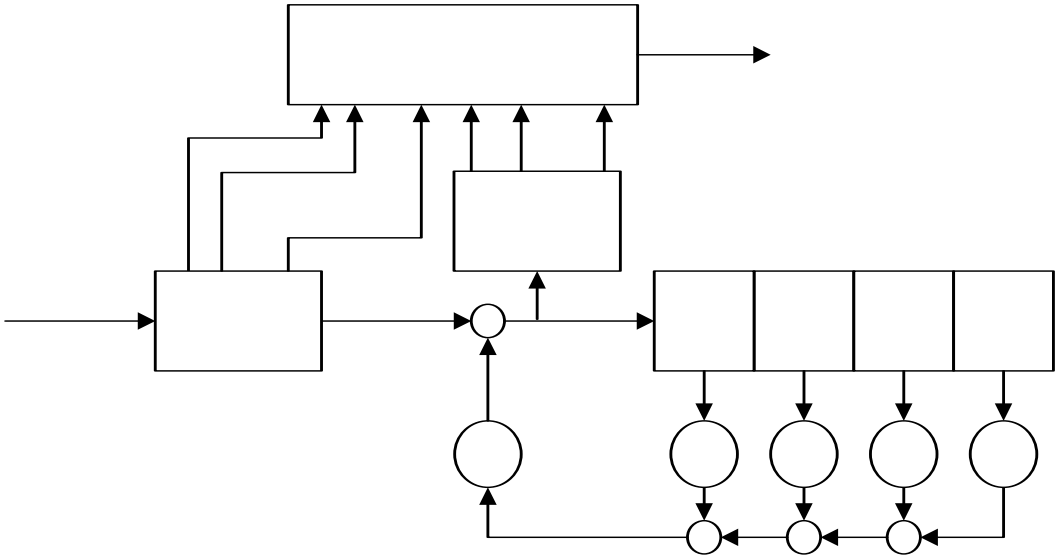


Рис. 3. Схема систематичного кодера алгебраїчного згорткового коду

Теорема 3. Породжувальний багаточлен *g(x)* циклічного *(N, K, D)* коду над *GF (q)* повністю визначає рекурсивний систематичний згортковий *(n, k, d)* код над *GF(q)* з кодовим обмеженням

$$\nu = (N-K)\cdot K$$

і параметрами

Теорема 4. Якщо зафіксувати $GF(q^m)$ і кінцеву безліч H елементів поля $GF(q^m)$, причому $\log_q |H| = k^0$, $m \geq k^0$, то породжувальний багаточлен *g(x)* циклічного *(N, K, D)* коду над $GF(q^m)$ повністю визначає рекурсивний систематичний згортковий *(n, k, d)* код над *GF(q)* з кодовим обмеженням

$$\nu = (N-K)\cdot K\cdot |\log q|$$

і параметрами

Для алгебраїчної побудови рекурсивного згорткового коду з конструктивними *(n, k, d)* параметрами необхідно й досить задати породжувальний й/або перевірочний багаточлен циклічного *(N, K, D)* коду. При цьому конструктивні параметри згорткового *(n, k, d)* коду будуть аналітично пов'язані з параметрами циклічного *(N, K, D)* коду. Алгоритм побудови рекурсивних згорткових кодів у загальному виді представимо у вигляді послідовності таких кроків.

КРОК 1. Уведення параметрів рекурсивного згорткового *(n, k, d)* коду й потужності алфавіту кодових символів q.

КРОК 2. Вибір варіанта побудови згорткового коду над *GF(q)*:

–несистематичного рекурсивного згорткового коду;

–систематичного рекурсивного згорткового коду.

КРОК 3. Розрахунок параметрів циклічного *(N, K, D)* коду над $GF(q^m)$.

КРОК 4. Вибір і формування породжувального й/або перевірочного багаточлена циклічного *(N, K, D)* коду над $GF(q^m)$.

КРОК 5. Вибір способу обробки кодових символів. Формування породжувальних багаточленів рекурсивного згорткового *(n, k, d)* коду над *GF(q)*, побудова схеми кодера рекурсивного згорткового *(n, k, d)* коду над *GF(q)*.

Розроблений алгоритм дозволяє конструктивним способом за кінцеве число кроків побудувати рекурсивний згортковий код з необхідними параметрами.

У четвертому розділі розробляються методи декодування алгебраїчно заданих згорткових кодів, засновані на використанні нескінченної серії синдромів кодових слів циклічного коду. Пропонується спосіб формування нескінченної серії синдромів алгебраїчно заданого згорткового коду. Розробляється підхід комбінованого декодування алгебраїчно заданих згорткових кодів, що складає в сполученні алгебраїчних процедур і процедур послідовного пошуку по кодовій решітці.

Нехай $I(x) = i_0 + i_1x + i_2x^2 + \ldots$ - інформаційний багаточлен, можливо нескінченної довжини, з коефіцієнтами з *GF(q)*. Кодова послідовність на виході несистематичного нерекурсивного згорткового кодера буде задаватися виразом:

$$C(x) = I(x) \cdot P(x) = C_0 + C_1x + C_2x^2 + \ldots,$$

де C_i – елементи поля $GF(q^m)$, відображувані в набори по m символів з підполя *GF(q)*.

Представимо кодове слово *C(x)* у поліноміальному виді

(6)

де - одинична матриця з доданими ліворуч $i\cdot K$ нульовими стовпцями.

Проаналізуємо отриманий вираз (6). Кожен доданок містить добуток породжувального багаточлена циклічного *(N, K, D)* коду на інформаційний багаточлен $I_i(x)$ ступеня $deg\ I_i(x) \le K - 1$. Однак, добуток $I_i(x)\cdot P(x)$ – суть кодове слово циклічного *(N, K, D)* коду, що відповідає інформаційному вектору

$$I_i = (i_{i\cdot K}, i_{i\cdot K+1}, i_{i\cdot K+2}, \ldots, i_{(i+1)\cdot K-1}),$$

тобто

$$I_i(x)\cdot P(x) = c_i(x), \quad (7)$$

де

$$c_i(x) = c_{i,0} + c_{i,1}x + c_{i,2}x^2 + \ldots c_{i,N-1}x^{N-1}\ldots$$

Підставимо (7) в (6), одержимо:

. (8)

З виразу (8) випливає, що нескінченне кодове слово нерекурсивного згорткового коду, алгебраїчно заданого через породжувальний багаточлен циклічного коду, складається з нескінченної суми кодових слів циклічного коду, помножених на відповідний оператор затримки $x^{i\cdot K}$. Представимо, для наочності, структуру нескінченного кодового слова алгебраїчного згорткового коду на рис. 4.

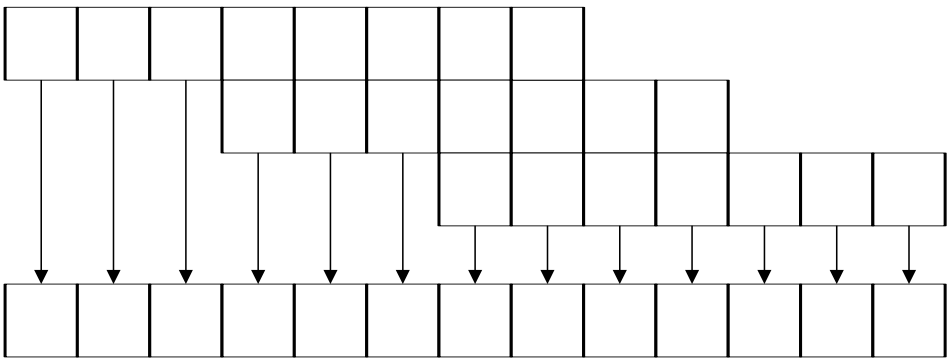


Рис. 4. Структура нескінченного кодового слова алгебраїчного нерекурсивного згорткового коду

Як видно з рис. 4 нескінченне кодове слово згорткового коду формується накладенням нескінченного числа кодових слів циклічного коду й підсумовуванням відповідних елементів $c_{i,j}$.

Припустимо тепер, що при передачі нескінченної кодової послідовності вектор $C = (C_0, C_1, \ldots)$ спотворився, тобто на приймальній стороні отримане перекручене кодове слово

$$C^*(x) = C(x) + E(x),$$

де $E(x) = e_0 + e_1x + e_2x^2 + \ldots$–нескінченний вектор помилок.

За аналогією з інформаційним вектором розіб'ємо вектор помилок $E = (e_0, e_1, e_2, \ldots)$, складений з коефіцієнтів багаточлена помилок $E(x)$, на блоки по K символів з $GF(q)$: $E = (e_0, e_1, e_2, \ldots, e_{-1}) \cup (e, e_{+1}, e_{+2}, \ldots, e_{2K-1}) \cup \ldots\ldots$

З урахуванням (8) останній вираз перепишемо у вигляді:

(9)

де - вектор помилок E_i довжиною K символів з доданими праворуч $(N - K)$ нулями.

Проаналізуємо отриманий вираз. Кожен доданок містить суму кодового слова $c_i(x)$ циклічного (N, K, D) коду й багаточлена помилки $E_i(x)$. Розмірність вектора E_i становить K символів, тобто сума $c_i(x) + E_i(x)$ – суть кодове слово циклічного (N, K, D) коду, перекручене вектором помилки E_i. Отже, запишемо:

$$c_i^*(x) = c_i(x) + E_i(x). \quad (10)$$

Тоді, з урахуванням (10), вираз (9) перепишемо у вигляді:

. (11)

Таким чином, як випливає з виразу (11), нескінченне кодове слово алгебраїчного нерекурсивного згорткового коду, перекручене нескінченним вектором помилок, складається з нескінченної суми кодових слів циклічного коду, перекручених вектором помилок кінцевої розмірності, помножених на відповідний оператор затримки $x^{i \cdot K}$.

Представимо, для наочності, структуру перекрученого помилками нескінченного кодового слова алгебраїчного згорткового коду на рис. 5.

Як видно з рис. 5 перекручене помилками нескінченне кодове слово згорткового коду формується накладенням нескінченного числа перекручених кодових слів циклічного коду й підсумовуванням відповідних елементів $c^*_{i,j}$.

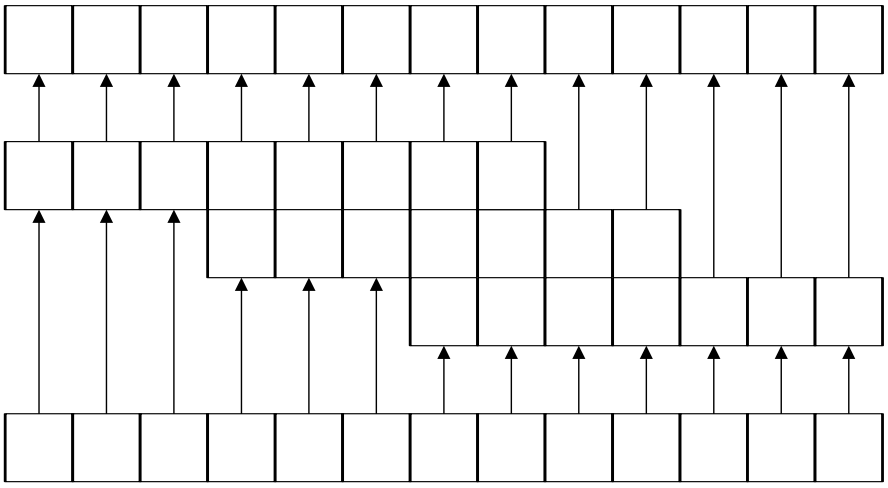


Рис. 5. Структура перекрученого помилками нескінченного кодового слова алгебраїчного нерекурсивного згорткового коду

Уведемо синдромний багаточлен алгебраїчного згорткового коду:

$$S(x) = s_0 + s_1 x + s_2 x^2 + \ldots,$$

як нескінченну суму синдромних багаточленів циклічного коду, помножених на відповідний оператор затримки $x^{i \cdot K}$, тобто як нескінченну суму залишків від розподілу кодових слів циклічного коду на породжувальний багаточлен $P(x)$:

і перепишемо його через перевірочний багаточлен

.

Таким чином, як виходить з отриманого виразу для $S(x)$, нескінченний синдром прийнятого з помилками кодового слова алгебраїчного нерекурсивного згорткового коду складається з нескінченної суми синдромів прийнятих кодових слів циклічного коду, помножених на відповідний оператор затримки $x^{i \cdot (N-K)}$. Отже, запишемо

, (12)

де $S_i(x) = s_{i \cdot K} + s_{i \cdot K+1} x + s_{i \cdot K+2} x^2 + \ldots s_{(i+1) \cdot K-1} x^{K-1}$ – синдромний багаточлен циклічного (N, K, D) коду, $S_i = (s_{i \cdot K}, s_{i \cdot K+1}, s_{i \cdot K+2}, \ldots s_{(i+1) \cdot K-1})$ - відповідний синдромний вектор.

Синдромний багаточлен (вектор) залежить тільки від значення помилок і не залежить від обраного кодового слова. Як видно з рис. 6. нескінченний синдром формується нескінченним підсумовуванням відповідних синдромів циклічного коду $S_i(x)$.

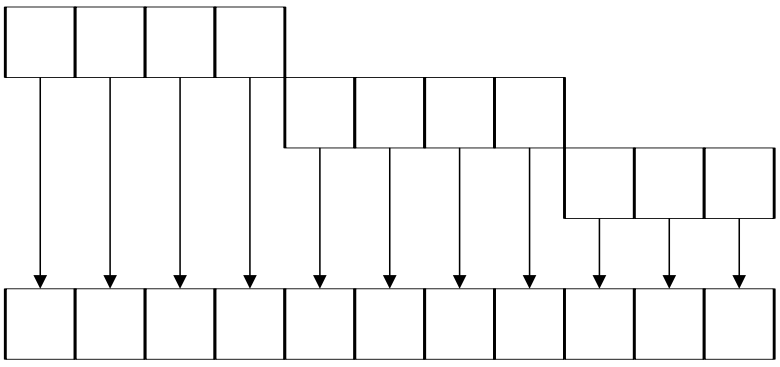


Рис. 6. Структура нескінченного синдромного багаточлена алгебраїчного нерекурсивного згорткового коду

Причому синдроми $S_i(x)$ підсумовуються без накладень, тобто кожен блок з $(N - K)$ синдромних символів залежить винятково від блоку з K помилкових символів. Цей факт дозволяє реалізувати алгебраїчне правило декодування алгебраїчно заданого згорткового коду.

Дійсно, декодування нескінченного кодового слова згорткового коду розпадається на нескінченну послідовність декодувань кодових слів циклічного (N, K, D) коду. Причому кожен синдромний вектор S_i відповідає помилці, що відбулася

на блоці з K символів. У випадку неправильного декодування помилка розповсюджується тільки в межах блоку даних з K символів. Отже, незалежність блоків синдромних символів дозволяє уникнути розповсюдження помилок, що властиво деяким відомим способам декодування згорткових кодів. Таким чином, у результаті проведених міркувань удалося звести декодування нескінченного кодового слова до нескінченної серії декодувань циклічного блокового коду.

Для реалізації запропонованого підходу алгебраїчного декодування нескінченних кодових слів алгебраїчного згорткового коду, заданого через породжувальний багаточлен циклічного коду, необхідно й досить обчислити нескінченну суму синдромів відповідних кодових слів циклічного коду, тобто обчислити всі значення $S_i = (s_{i\cdot K}, s_{i\cdot K+1}, s_{i\cdot K+2}, \ldots s_{(i+1)\cdot K-1})$ у виразі (12). Для обчислення синдромної послідовності в алгебраїчній теорії блокових кодів використовують множення кодового слова на перевірочну матрицю й/або, що еквівалентно, формують синдромний багаточлен $S_i(x)$ через відповідні операції в кільці багаточленів $GF(q)[x]/(x^n - 1)$. Еквівалентною операцією для безперервних кодів буде добуток кодового слова на напівнескінченну перевірочну матрицю згорткового коду, задану через корінь породжувального багаточлена. Конструктивних способів побудови напівнескінченної перевірочної матриці несистематичного згорткового коду зі збереженням таких алгебраїчних властивостей невідомо. Отже, для реалізації запропонованого вище підходу алгебраїчного декодування згорткових кодів необхідно теоретично обґрунтувати й увести відповідні процедури формування нескінченної серії синдромних послідовностей $S_i = (s_{i\cdot K}, s_{i\cdot K+1}, s_{i\cdot K+2}, \ldots s_{(i+1)\cdot K-1})$.

Розглянемо структуру нескінченного кодового слова алгебраїчного згорткового коду на рис. 4. Якщо алгебраїчний згортковий код заданий через породжувальний багаточлен (N, K, D) циклічного коду то нескінченне кодове слово формується підсумовуванням нескінченного числа кодових слів циклічного коду, зрушених на K символів вправо. Отже, i-й блок з N кодових символів нескінченного кодового слова складається із суми i-го кодового слова циклічного коду й відповідних частин $i+j$-их кодових слів. Схематично структура довільного блоку з N символів нескінченного кодового слова алгебраїчного нерекурсивного згорткового коду представлена на рис. 7.

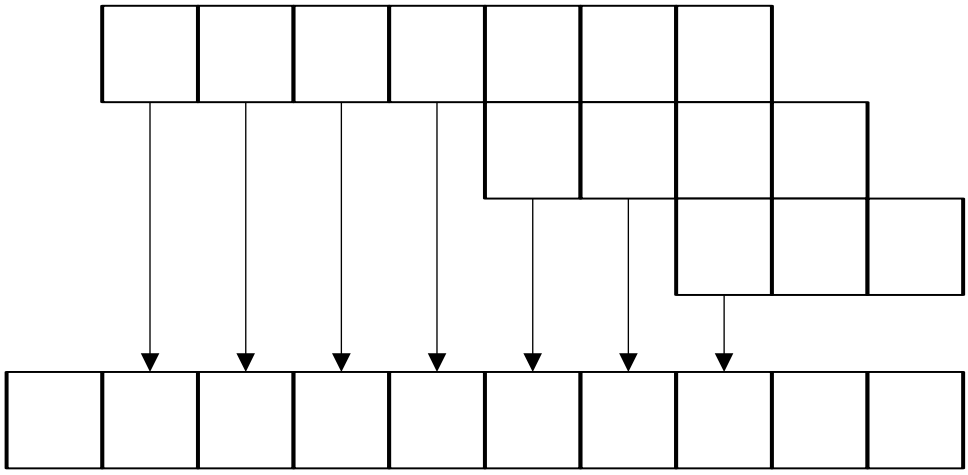


Рис. 7. Структура довільного блоку з *N* символів нескінченного кодового слова алгебраїчного нерекурсивного згорткового коду

Формально, запишемо:

, (13)

де - блок блоку з *N* символів нескінченного кодового слова алгебраїчного

згорткового коду; - *i*-е кодове слово циклічного коду; - відповідні підблоки *i+j*-их кодових слів циклічного коду.

Припустимо, що на розглянутому блоці з *N* символів відбулося не більше *t = (D - 1)/2* помилок. Сформуємо із блоку кодовий багаточлен й обчислимо залишок від ділення його на породжувальний багаточлен циклічного коду. Остання операція еквівалентна множенню на перевірочний багаточлен й/або добутку блоку

на перевірочну матрицю циклічного коду. Одержимо синдромний вектор *S**:

(14)

де $e_i(x)$ – багаточлен помилок, коефіцієнтами якого є елементи вектора помилок довжини *N* символів, тобто вектор помилки на *i*-му кодовому слові циклічного коду;

Z(x) – сума багаточленів, коефіцієнтами яких є елементи векторів у виразі (13), $j \neq 0$.

Очевидно, що перший доданок у виразі (14) суть залишок від ділення кодового слова циклічного коду на відповідний породжувальний багаточлен. Отже,

.

Другий доданок у виразі (14) відповідає залишку від розподілу багаточлена помилок кодового слова циклічного коду на його породжувальний багаточлен. Отже

, в введених раніше позначеннях.

Третій доданок відповідає залишку від ділення суми багаточленів з виразу (13) на породжувальний багаточлен циклічного коду. Відзначимо, що всі багаточлени,

коефіцієнтами яких є елементи векторів у виразі (13), $j \neq 0$ мають ступінь $\leq N - K$, а ступінь породжувального багаточлена $\deg g(x) = N - K+1$. Отже, запишемо

. Тоді вираз (14) можна переписати в наступному виді

$$S^*(x) = S_i(x) + Z(x). \quad (15)$$

Очевидно, що при $Z(x) = 0$ виконується рівність $S^*(x) = S_i(x)$. Практично це

означає, що при виконанні рівності нулю суми векторів у виразі (13), $j \neq 0$ значення синдромів *S** для блоку з *N* кодових символів збіжаться з відповідними синдромами $S_i(x)$ *i*-их кодових слів циклічного коду. Таким чином, для формування нескінченної серії кінцевих синдромів для алгебраїчного декодування

згорткових кодів досить виконання умови $Z(x) = 0$.

Для виконання сформульованої умови розглянемо правило формування багаточлена $Z(x)$. Як показано вище, багаточлен $Z(x)$ формується підсумовуванням багаточленів, коефіцієнтами яких є елементи кодових i-их слів, зсунутих на $j \cdot K$ символів вправо. Припустимо, що нескінченне кодове слово нерекурсивного згорткового коду алгебраїчно заданого через породжувальний багаточлен циклічного коду (див. рис. 4) складається з нескінченної суми кодових слів циклічного коду, помножених оператор затримки $x^{i \cdot N}$. Тоді багаточлен $Z(x)$ буде дорівнювати сумі багаточленів, коефіцієнтами яких є елементи кодових i-их слів, зсунутих на $j \cdot N$ символів вправо. Але по визначенню вектор S^* - це синдромна послідовність, що відповідає блоку з N кодових символів нескінченного кодового слова. Практично це означає, що третій доданок у виразі (14) дорівнює нулю, тобто $Z(x) = 0$, відповідно. Підставивши в (15), одержимо

$$S^*(x) = S_i(x).$$

Останній вираз дозволяє сформувати нескінченну серію кінцевих синдромів нескінченного кодового слова згорткового коду. А це дозволяє реалізувати алгебраїчний алгоритм декодування згорткових кодів.

Таким чином, у результаті проведених досліджень, був розроблений спосіб формування нескінченної серії кінцевих синдромів для алгебраїчного декодування згорткових кодів. При цьому слід зазначити деяке погіршення конструктивних властивостей згорткового коду. Для реалізації пропонованого способу при формуванні кодового слова алгебраїчного згорткового коду оператор затримки $x^{i \cdot K}$ варто замінити на $x^{i \cdot N}$. Практично це означає, що після подачі на вхід кодера K інформаційних символів необхідно подати, додатково, $(N - K)$ нульових символів. У цьому випадку на прийомній стороні виконається умова $Z(x) = 0$ й, відповідно, рівність $S^*(x) = S_i(x)$. У термінах кодування подача на вхід кодера $(N - K)$ нульових символів відповідає зниженню швидкості кодування в $(N - (N - K))/N = K/N$ раз, тобто зниження швидкості пропорційно швидкості циклічного (N, K, D) коду. Остання обставина знижує завадостійкість алгебраїчних згорткових кодів (з алгебраїчним способом декодування). Однак при відповідному виборі параметрів циклічного (N, K, D) коду це погіршення можна мінімізувати. Дійсно, якщо при побудові згорткового коду використовувати циклічні (N, K, D) коди з $R = K/N \rightarrow 1$, то для реалізації алгебраїчного декодування необхідно внести мізерно малу частку нульових символів й, таким чином, зниження завадостійкості буде мінімальним.

Отримане узагальнене подання нескінченного кодового слова алгебраїчно заданого згорткового коду через нескінченну суму послідовних наборів з M кодових слів (N, K, D) циклічного коду дає потужний механізм комбінованого декодування алгебраїчно заданих згорткових кодів (з використанням запропонованих алгебраїчних процедур і відомих алгоритмів декодування, наприклад, послідовного алгоритму Фано). Застосування запропонованих алгебраїчних процедур декодування дозволяє локалізувати помилки в кодовому слові згорткового коду з точністю до деякого, заздалегідь заданого, періоду. Це дозволяє значно спростити роботу другого алгоритму, наприклад, прискорити послідовний пошук по кодовій решітці.

Алгоритм комбінованого декодування алгебраїчно заданого згорткового коду представимо у вигляді послідовності наступних кроків.

Крок 1. Прийом $(M-1)K + N$ кодових символів з $GF(q^m)$ (або, що еквівалентно, $(M \cdot K + N)m$ кодових символів з $GF(q)$).

Крок 2. Обчислення синдрому.

Крок 3. Рішення систем лінійних рівнянь.

Крок 4. Локалізація помилок з точністю до періоду K символів.

Крок 5. Послідовний пошук по кодовій решітці у вузлах, що відповідають компонентам згрупованої помилки.

Крок 6. Виправлення згрупованої помилки.

Крок 7. Прийом наступних $(M-1)K + N$ кодових символів з $GF(q^m)$ (або, що еквівалентно, $(M \cdot K + N)m$ кодових символів з $GF(q)$). Перехід до кроку 2.

Пошук по кодовій решітці (крок 5) може виконуватися й відразу, після прийому першого кодового символу (як при послідовному декодуванні). Це може бути виправдане при малому числі помилок. Якщо число помилок велике, то складність послідовного декодування швидко зростає й, мабуть, варто очікувати попередньої локалізації помилок (крок 4).

У п'ятому розділі досліджуються методи побудови паралельних каскадних кодових конструкцій і процедури їхнього декодування. Пропонуються схеми турбокодування з використанням рекурсивних згорткових кодів, заданих через породжувальний й/або перевірочний багаточлени недвійкового циклічного коду. Розробляються алгоритми побудови турбокодів з необхідними параметрами.

Теорема 5. Турбокодер, побудований на алгебраїчних несистематичних рекурсивних згорткових кодах, має швидкість кодування:

.

Якщо $k^0 = 1$, то маємо рекурсивний несистематичний згортковий код з параметрами: $\nu = K$, $n^0 = m$, $k = K + 1$, $n = (K + 1) \cdot n^0$, $R = 1/m$, $d_\infty \geq D$. Відповідний турбокодер має швидкість кодування $R_{TK} = 1/(2 \cdot m)$. Схема турбокодера, побудованого на алгебраїчних несистематичних рекурсивних згорткових кодах, з обробкою елементів з $GF(q^m)$, представлена на рис. 8.

Зв'язок параметрів турбокоду, побудованого на основі алгебраїчних систематичних рекурсивних згорткових кодів визначається наступною теоремою.

Теорема 6. Турбокодер, побудований на алгебраїчних несистематичних рекурсивних згорткових кодах, має швидкість кодування:

.

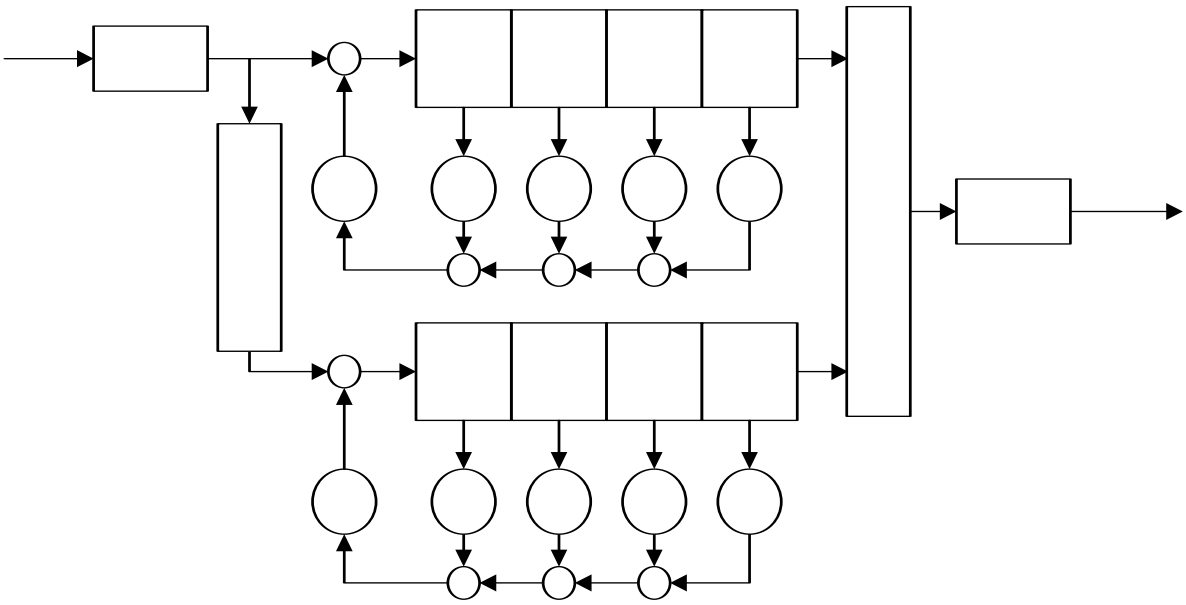


Рис. 8. Турбокодер на алгебраїчних несистематичних рекурсивних згорткових кодах з обробкою елементів з $GF(q^m)$

Якщо $k^0 = K$, $n^0 = N$, то маємо рекурсивний систематичний згортковий код з параметрами: $\nu = (N–K)\cdot K$, $k^0=K$, $n^0=N$, $k=(N-K+1)\cdot K$, $n=(N-K+1)\cdot N$, $R=K/N$, $d_\infty \geq D$. Відповідний турбокодер має швидкість кодування $R_{TK} = k^0/(2\cdot n^0-k^0) = K/(2\cdot N–K)$, тобто швидкість турбокода буде визначатися винятково швидкістю циклічного (N, K, D) коду.

Якщо $k^0 = K = 1$, $n^0 = N$, то маємо рекурсивний систематичний згортковий код з параметрами: $\nu = N–1$, $k^0=1$, $n^0=N$, $k=N$, $n=N^2$, $R=1/N$, $d_\infty \geq D$. Відповідний турбокодер має швидкість кодування $R_{TK} = k^0/(2\cdot n^0-k^0) = 1/(2\cdot N–1)$.

Схема турбокодера, побудованого на алгебраїчних рекурсивних згорткових кодах, з обробкою елементів з $GF(q^m)$ представлена на рис. 9.

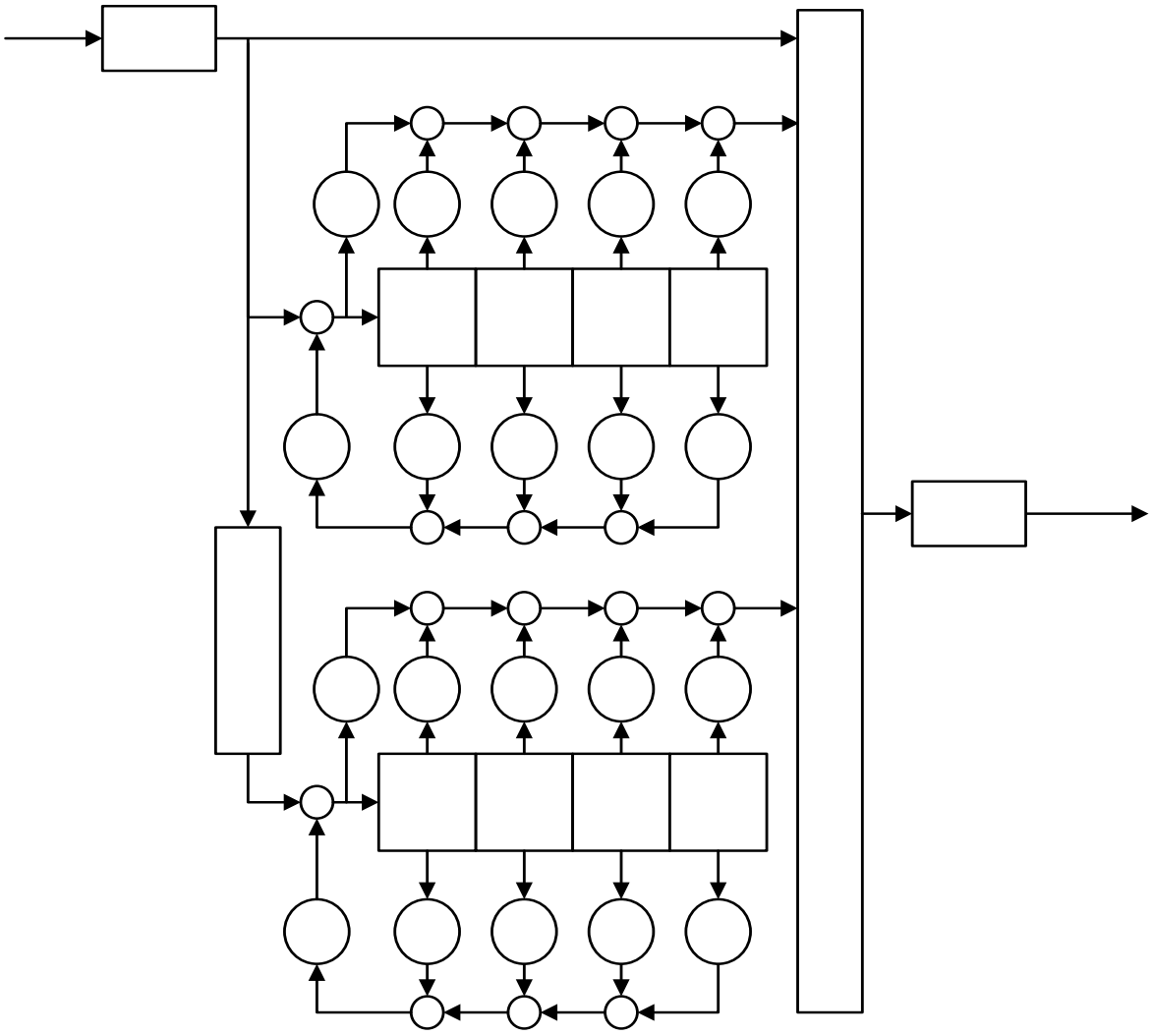


Рис. 9. Загальна схема турбокодера з використанням алгебраїчних систематичних рекурсивних згорткових кодів з обробкою елементів з $GF(q^m)$

Алгоритм побудови турбокодів на алгебраїчних рекурсивних згорткових кодах складається з послідовності наступних кроків.

КРОК 1. Уведення необхідної швидкості турбокода R_{TK}, необхідних k^0 й n^0 параметрів відповідних згорткових кодів, уведення потужності алфавіту кодових символів q.

КРОК 2. Розрахунок швидкості $R_{ск}$ рекурсивних згорткових кодів

КРОК 3. Вибір способу обробки кодових символів.

КРОК 4. Вибір варіанта побудови рекурсивних згорткових кодів, розрахунок параметрів відповідного циклічного коду над $GF(q^m)$, формування породжувальних багаточленів і побудова схеми кодера згорткового коду над $GF(q)$.

КРОК 5. Побудова паралельної каскадної схеми з алгебраїчними рекурсивними згортковими кодами.

Для реалізації процедури ітеративного декодування турбокодів необхідні алгоритми м'якого декодування, що дозволяють оцінити апостеріорну ймовірність кожного з інформаційних символів кодового слова, тобто реалізуючі посимвольне

правило прийняття рішень із мінімізацією середньої ймовірності помилки символу. Однак існуючі алгоритми м'якого декодування згорткових кодів навіть при сучасному рівні розвитку мікроелектроніки незастосовні для використання в ітеративному декодері турбокодів з великим значенням кодового обмеження. Тому будемо використовувати алгебраїчний підхід для декодування алгебраїчно заданих згорткових кодів.

Зведемо декодування згорткового коду до декодування послідовності кодових слів циклічного коду. Таким чином, м'яке посимвольне декодування згорткового коду можна звести до м'якого посимвольного декодування послідовних наборів кодових слів циклічного коду. Алгоритм ітеративного декодування турбокодів представимо у вигляді послідовності наступних кроків.

Крок 1. Прийом кодового слова турбокоду. Виділення послідовності інформаційних символів і послідовностей перевірочних символів складових згорткових кодів.

Крок 2. Урахування м'яких рішень другого складового декодера на всіх ітераціях, крім першої (на першій ітерації замість послідовності м'яких рішень використається нульова послідовність). Перетворення кодового слова першого складового згорткового коду в послідовність кодових слів циклічного коду. М'яке декодування послідовності кодових слів циклічного коду (алгоритми Хартмана-Рудольфа, Грінбергера). Перетворення послідовності кодових слів циклічного коду в кодове слово першого складового згорткового коду.

Крок 3. Перемеження м'яких рішень першого складового декодера й інформаційної послідовності.

Крок 4. Урахування м'яких рішень першого складового декодера. Перетворення кодового слова другого складового згорткового коду в послідовність кодових слів циклічного коду. М'яке декодування послідовності кодових слів циклічного коду (алгоритми Хартмана-Рудольфа, Грінбергера). Перетворення послідовності кодових слів циклічного коду в кодове слово другого складового згорткового коду.

Крок 5. Деперемеження м'яких рішень другого складового декодера.

Крок 6. Якщо поточний номер ітерації дорівнює максимальному числу ітерацій, то здійснюється прийняття жорстких рішень, у противному випадку – перехід до Кроку 2.

Таким чином, запропонований алгоритм ітеративного декодування турбокодів на основі узагальненого подання кодових слів складових згорткових кодів дозволяє звести декодування згорткового коду до декодування послідовності кодових слів циклічного коду.

У шостому розділі досліджуються моделі каналів зв'язку, розроблена методика оцінки достовірності переданої інформації, що дозволяє для заданих параметрів моделі каналу зв'язку із заданою погрішністю оцінити ймовірність помилкового прийому біта інформації й відповідний енергетичний виграш від кодування. Аналітично отримані криві залежності ймовірності помилки на біт від енергетичного відношення сигнал/шум для різних довжин , , . Аналіз отриманих результатів показав, що при фіксованому значенні ймовірність

помилки зменшується з ростом кодового обмеження й зменшенні швидкості згорткового коду. Однак при використанні кодів зі швидкостями менш 1/3 зростання ефективності кодування припиняється. Використання складових згорткових кодів зі швидкостями для збільшення загальної швидкості турбокоду більш ефективно, ніж застосування процедури виколювання.

За допомогою розробленої імітаційної моделі були експериментально підтверджені аналітично отримані результати при дБ, що підтверджує достовірність отриманих результатів. У ході проведених досліджень із використанням розробленої імітаційної моделі системи передачі інформації встановлено, що турбокоди на основі алгебраїчно заданих рекурсивних згорткових кодів не уступають по ефективності відомим у цей час кодам. На основі отриманих результатів проведених досліджень розроблені практичні рекомендації з використання синтезованих алгебраїчно заданих кодових конструкцій для підвищення достовірності переданої інформації.

У **додатках** представлена програмна реалізація алгоритмів побудови згорткових кодових конструкцій.

ВИСНОВКИ

У дисертаційній роботі вирішена важлива наукова проблема, пов'язана з розробкою на основі єдиного концептуального підходу методів синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій з необхідними властивостями й характеристиками, що має велике значення як для розвитку окремого напрямку теорії завадостійкого кодування, так і для вирішення прикладних питань, пов'язаних із забезпеченням заданої достовірності переданої інформації в телекомунікаційних системах і мережах.

1. Проведений аналіз показав, що розвинена в цей час алгебраїчна теорія блокового кодування не може бути безпосередньо застосована до згорткових кодів через значне розходження в їхніх властивостях у порівнянні із блоковими кодами. Незважаючи на це відомо, що існує можливість представлення згорткового коду у вигляді блокового коду напівнескінченної довжини і його наступним алгебраїчним описом. Позитивні результати в цьому напрямку отримані тільки для обмеженого діапазону низьких швидкостей кодування, значення яких не задовольняють сучасним вимогам, пропонованим до параметрів завадостійких кодів. Крім того, раніше не розглядалася можливість застосування алгебраїчної теорії для реалізації декодування алгебраїчно заданих згорткових кодів з довільними параметрами. Таким чином, виникає наукова проблема (суперечлива ситуація), у якій існуючі положення теорії завадостійкого кодування не дозволяють обчислювально реалізуємо вирішувати завдання синтезу й декодування згорткових кодів з високими конструктивними кодовими характеристиками.

2. У ході вирішення виявленої наукової проблеми були отримані наступні наукові й практичні результати.

–Одержав подальший розвиток єдиний концептуальний підхід алгебраїчного представлення згорткових кодів у вигляді недвійкових блокових циклічних кодів (

напівнескінченної довжини), що відрізняється від відомого (теоретичним узагальненням на випадок напівнескінченної довжини кодового слова циклічного коду й) використанням породжувальних багаточленів недвійкових циклічних кодів, обмежених на довільне підполе, що дозволяє розглядати з єдиних теоретичних позицій процеси синтезу, кодування й декодування згорткових кодів з довільними властивостями й кодовими характеристиками й теоретично обґрунтувати аналітичні вирази по оцінці кодових співвідношень синтезованих згорткових кодових конструкцій, аналітично зв'язати їхні параметри й виразити через кодові характеристики відповідних циклічних кодів.

–Одержали подальший розвиток обчислювально ефективні (такі, що можуть бути обчислювально реалізовані) алгебраїчні методи синтезу (алгебраїчно заданих) згорткових кодів, що відрізняються від відомих використанням обмеження недвійкового циклічного коду на довільне підполе, що дозволяє синтезувати (алгебраїчно задані) згорткові коди з довільними властивостями й кодовими характеристиками.

–Одержали подальший розвиток методи кодування алгебраїчно заданими згортковими кодами, що відрізняються від відомих теоретично обґрунтованими процедурами алгебраїчної побудови рекурсивних і нерекурсивних згорткових кодів через узагальнення циклічних кодів на випадок нескінченної довжини, що дозволяє аналітично формалізувати процес завадостійкого кодування синтезованими згортковими кодами з високими (конструктивними) кодовими характеристиками.

–Уперше розроблені алгебраїчний і комбінований методи декодування алгебраїчно заданих згорткових кодів, які відрізняються від відомих методів процедурами алгебраїчної локалізації й прискорених процедур (алгоритмами) послідовного пошуку, що дозволяє реалізувати обчислювально ефективне (таке, що може бути обчислювально реалізоване) декодування безперервних кодових конструкцій з великою довжиною кодового обмеження (з більшою кодовою відстанню) для підвищення достовірності переданої інформації.

–Одержали подальший розвиток методи синтезу паралельних каскадних згорткових конструкцій (методи турбокодування), що відрізняються від відомих використанням алгебраїчно заданих рекурсивних згорткових кодів, що дозволяє аналітично зв'язати параметри турбокодів з параметрами алгебраїчно заданих рекурсивних згорткових кодів і синтезувати паралельні каскадні згорткові конструкції із заданими (конструктивними кодовими) характеристиками.

–Одержав подальший розвиток метод ітеративного декодування турбокодів з алгебраїчно заданими рекурсивними згортковими кодами, що відрізняється від відомого узагальненим представленням нескінченного кодового слова згорткового коду через нескінченну суму послідовних наборів з кодових слів циклічного коду, що дозволяє за рахунок зведення декодування згорткового коду до декодування послідовності кодових слів циклічного коду декодувати турбокоди на основі алгебраїчно заданих згорткових кодів з великою кількістю елементів пам'яті (з високими кодовими характеристиками, високою кодовою відстанню).

–Розроблені алгоритми синтезу, кодування й декодування алгебраїчно заданих згорткових кодових конструкцій з необхідними (кодовими) характеристиками, такі що можуть бути обчислювально реалізовані.

–Розроблено методику (емпіричної) оцінки достовірності переданої інформації , що дозволяє для (заданих параметрів математичної моделі) дискретно-безперервного каналу із заданою погрішністю оцінити ймовірність помилкового прийому біта інформації й відповідний енергетичний виграш від кодування.

–Розроблено імітаційну модель системи передачі інформації з використанням алгебраїчно заданих згорткових кодових конструкцій, за допомогою якої встановлено, що синтезовані згорткові кодові конструкції, отримані за допомогою розроблених алгоритмів, що можуть бути обчислювально реалізовані, не поступаються по енергетичних характеристиках відомим у цей час кодам; їхнє практичне використання дозволяє забезпечити підвищення достовірності переданої інформації в каналах з випадково виникаючими помилками за рахунок відсутності обмежень при виборі необхідних параметрів синтезованих згорткових кодових конструкцій; розроблені обчислювально реалізуємі алгоритми декодування згорткових кодових конструкцій з високими конструктивними кодовими характеристиками мають параметри близькі до теоретично граничних значень.

–Розроблено практичні рекомендації з використання турбокодів із синтезованими алгебраїчно заданими згортковими кодами. Для забезпечення ймовірності помилки на біт при значенні енергетичного відношення сигнал/шум 1,5 – 2 дБ, пропонується використовувати турбокоди з кількістю елементів пам'яті 2 – 4. Для забезпечення ймовірності помилки на біт пропонується використовувати турбокоди з кількістю елементів пам'яті 6 – 8. Швидкість кодування не рекомендується вибирати менш ніж 1/3.

–Отримані результати використані в науково-дослідних роботах «Мрія», «Алгоритм» (Харківський університет Повітряних Сил, акт реалізації від 12.04.2005), на виробництві при розробці спеціального математичного та програмного забезпечення програмно-апаратного макету завадостійкого кодеру (декодеру) у ЦККБ «Протон» (акт реалізації від 26.05.2008) і в навчальному процесі Української державної академії залізничного транспорту (акт реалізації від 15.04.2008).

3. При вирішенні наукових завдань використовувалися наступні методи дослідження. Розробка й дослідження алгебраїчних методів і процедур синтезу, кодування й декодування згорткових кодових конструкцій проведені з використанням методів алгебраїчної теорії кодів, теорії полів Галуа й теорії чисел. Оцінка достовірності переданої інформації проведена з використанням методів статистичної теорії зв'язку, теорії імовірності й математичної статистики. Розробка рекомендацій з реалізації кодерів алгебраїчно заданих згорткових кодів проведена з використанням методів теорії цифрових автоматів.

4. Обґрунтованість отриманих результатів заснована на коректному застосуванні основних положень теорії кодування, теорії ймовірностей, статистичної теорії зв'язку, теорії множин, математичної статистики.

5. Достовірність отриманих результатів підтверджується збіжністю теоретичних результатів і результатів по обробці експериментальних даних, отриманих у ході функціонування розробленої імітаційної моделі.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

139. Приходько С.И., Столяров А.С. Принцип приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам. Часть I. // Специальная техника средств связи. – МО СССР. – №3. – 1988. – С.14-16.

140. Приходько С.И., Столяров А.С. Принцип приведения двоичных сверточных кодов к недвоичным суженным циклическим кодам. Часть II // Специальная техника средств связи. – МО СССР. – №4. – 1988. – С.25-29.

141. Приходько С.И., Снисаренко А.Г. Приведение двоичных сверточных кодов к недвоичным суженным циклическим кодам // Радиотехника. Республиканский межведомственный научно-технический сборник. – Харьков: ХИРЭ. – 1989. – №90. – С.80-86.

142. Приходько С.И. Приведение сверточных кодов к кодам РС // Радиотехника. Республиканский межведомственный научно-технический сборник. – Харьков: ХИРЭ. – 1989. – №91. – С.81-84.

143. Приходько С.И., Березняков Г.Е. Приведение ортогонализируемых сверточных кодов к квазиортогональным // Радиотехника. Республиканский межведомственный научно-технический сборник. – Харьков: ХИРЭ. – 1990. – №8. – С.76-81.

144. Приходько С.И., Березняков Г.Е. Приведение ортогональных сверточных кодов к квазиортогональным сверточным кодам // Радиотехника. Республиканский межведомственный научно-технический сборник. – Харьков: ХИРЭ. – 1990. – №83. – С.65-69.

145. Приходько С.И., Гусев С.А., Сидоренко Н.Ф. Принцип приведения ортогональных сверточных кодов к квазиортогональным сверточным кодам // Системы информационного взаимодействия. Сборник научных трудов. – Харьков: НАНУ, ПАНИ, ХВУ. – 1996. – С.83-88.

146. Приходько С.И., Гусев С.А. Циклические сверточные коды // Управление и связь. Сборник научных трудов. – Харьков: НАНУ, ПАНИ, ХВУ. – 1996. – С.98-101.

147. Приходько С.И. Принцип последовательного декодирования обобщенно заданных сверточных кодов. Системы обработки информации. Сборник научных трудов. – Харьков: НАНУ, ПАНИ, ХВУ. – 1998. –
С.67-71.

148. Приходько С.И. Алгебраическое представление сверточных кодов // Вестник международного славянского университета. Вып.3. Харьков: НАНУ. – 1998 . – С.72-75.

149. Приходько С.И. Алгебраическое кодирование сверточных кодов. Информатика. Сборник научных трудов. Вып.5. Киев: Наукова Думка. – 1998. – С.72-75.

150. Приходько С.И. Алгоритм построения сверточных кодов // Информационные системы. Сборник научных трудов. Вып.1(9). Харьков: НАНУ. – 1998. –
С.82-75.

151. Приходько С.И. Построение сверточных кодов // Сборник научных трудов. Информационные системы. Вып.1(19). – Харьков: НАНУ, ПАНИ, ХВУ. – 1998. – С. 144-146.

152. Приходько С.И. Алгебраические сверточные коды // Информационно-управляющие системы на железнодорожном транспорте. – Харьков: ХарГАЖТ. – №2(17). – 1999. – С. 62-63.

153. Приходько С.И., Гусев С.А., Кужель И.Е. Алгебраическое построение несистематических сверточных кодов // Системи обробки інформації. – Харків: ХВУ . – 2004 – Вип. 8(36). – С. 170-175.

154. Приходько С.И., Гусев С.А., Кужель И.Е. Алгебраический метод сверточного кодирования // Комп'ютерні системи та інформаційні технології. – Х.: ХАИ. – 2005. – №1 – С.35-43.

155. Тимочко А.И., Приходько С.И., Постольный А.С. Алгебраический метод построения сверточных кодов в систематическом виде // Східно-Европейський журнал передових технологій. – Харків: Технологический центр. – 2005 – № 2/2(14). – С. 118-123.

156. Тимочко А.И., Приходько С.И., Постольный А.С. Алгебраический метод построения рекурсивных сверточных кодов для стандартов космической связи // Авиационно-космическая техника и технология. – Харків: ХАИ. – 2005. – №1(17). – С. 78-86.

157. Тимочко А.И., Приходько С.И., Постольный А.С. Алгебраические рекурсивные сверточные коды и схемы турбокодирования // Інформаційно-керуючі системи на залізничному транспорті. – Харків: УкрДАЗТ. – №1-2. – 2005. – С. 59-65.

158. Приходько С.И., Гусев С.А., Постольный А.С, Жученко А.С. Алгебраическое декодирование сверточных кодов // Інформаційно-керуючі системи на залізничному транспорті. – Харків: УкрДАЗТ. – №6. – 2005. – С. 29-37.

159. Приходько С.И., Гусев С.А., Постольный А.С., Жученко А.С. Комбинированный метод декодирования алгебраических сверточных кодов // Інформаційно-керуючі системи на залізничному транспорті. – Харків: УкрДАЗТ. – №2 (58). – 2006. – С. 8-15.

160. Приходько С.И. Оценка нижней границы свободного кодового расстояния алгебраически заданных сверточных кодов // Системи обробки інформації. – Х.: ХУПС. – 2007. – Вип. 5(65). – С. 120 – 124.

161. Приходько С.И. Метод декодирования алгебраических сверточных кодов // Системи обробки інформації. – Х.: ХУПС. – 2008. – Вип. 2(69). – С. 93 – 96.

162. Приходько С.И., Северинов А.В., Жученко А.С., Постольный А.С. Итеративное декодирование турбокодов на основе алгебраических рекурсивных сверточных кодов // Збірник наукових праць. – Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова. – Київ: НАН України, 2005. – Вип. 32. – С. 178 – 183.

163. Спосіб опису пристроїв кодування нерекурсивних згорнених кодів Пат. UA 14180 U, МКІ (2006) H03M 13/00. – № u 2005 08658; Заявл. 12.09.2005; Опубл. 15.05.2006, Бюл. №5, 2006р. – 6с. // Приходько С.І., Постольний О.С., Гусєв С.А., Жученко О.С., Кужель І.Є.

164. Спосіб опису пристроїв кодування рекурсивних згорнених кодів Пат. UA 14179 U, МКІ (2006) H03M 13/00. – № u 2005 08657; Заявл. 12.09.2005; Опубл.

15.05.2006, Бюл. №5, 2006р. – 4с. // Приходько С.І., Постольний О.С., Гусєв С.А., Жученко О.С., Кужель І.Є.

165. Спосіб опису пристроїв кодування згорнених кодів Пат. UA 14181 U, МКІ (2006) H03M 13/00. – № u 2005 08661; Заявл. 12.09.2005; Опубл. 15.05.2006, Бюл. №5, 2006р. – 6с. // Приходько С.І., Гусєв С.А., Жученко О.С., Кужель І.Є.

166. Приходько С.И., Гусев С.А. Алгебраический метод сверточного кодирования // Современные методы кодирования в электронных системах. Материалы международной НТК 26-27 октября 2004. – Сумы: СМКЭС. – 2004. – С. 49-50.

167. Приходько С.И., Гусев С.А., Кужель И.Е. Алгебраические сверточные коды // Перша науково-технічна конференція Харківського університету Повітряних Сил, 16-17 лютого 2005. Тези доповідей. – Х.: ХУПС. – 2005. – С. 210 – 211.

168. Приходько С.І. Метод декодирования алгебраических сверточных кодов. // Четверта наукова конференція Харківського університету Повітряних Сил ім. Івана Кожедуба. 16-17 квітня 2008 р. Матеріали конференції. – Х.: ХУ ПС. – 2008. – С. 149-150.

169. Приходько С.І. Алгебраический метод построения сверточных кодов для повышения помехоустойчивости передачи дискретных сообщений. // Перспективи розвитку озброєння і військової техніки в збройних силах України. Збірка тез доповідей Першої Всеукраїнської науково-практичної конференції 4-5 березня 2008 р. – Львів: ЛІСВ НУ “ЛП” – 2008. – С. 215.

170. Приходько С.І. Исследование свойств алгебраически заданных сверточных кодов .// Управління розвитком “Стратегії ІТ – технологій в освіті, економіці та екології. Матеріали міжнародної науково-технічної конференції. – Х.: ХНУ. – 2007. – С. 78-79.

171. Приходько С.І. Исследование корректирующих свойств алгебраических сверточных кодов. // Міжнародна науково-технічна конференція “Інтегровані комп’ютерні технології в машинобудуванні” ІКТМ – 2007. Тези доповідей. – Х.: НАКУ “ХАІ”. – 2007. – С.428-429.

172. Приходько С.І. Оценка свободного кодового расстояния алгебраических сверточных кодов // Проблеми інформатики і моделювання. Матеріали сьомої міжнародної науково-технічної конференції 22 листопада – 1 грудня. – Х.: НТУ “ХПІ” – 2007. – С. 13-14.

173. Приходько С.И. Алгебраические процедуры декодирования сверточных кодов // Современные методы кодирования в электронных системах. Материалы международной НТК 23-24 апреля 2002. – Сумы: СМКЭС. – 2002. – С.11–12.

174. Приходько С.И., Волков А.С. Особенности алгебраических самоортогональных сверточных кодов в частотной области // 22 международная научно-практическая конференция «Перспективные компьютерные, управляющие и телекоммуникационные системы для железнодорожного транспорта Украины». – Алушта. – 2009.

175. Розробка методів та програмних засобів підвищення достовірності та своєчасності передачі даних у телекомунікаційній системі АСУ Військ Протиповітряної Оборони Збройних Сил України комплексу засобів автоматизації “Ореанда”. Звіт про НДР. Шифр “Алгоритм”. Проміжний. № держреєстрації

0101U000413. / Приходько С.І., Кузнецов О.О., Кужель І.Є. та інші // Х.: ХУПС., 2005. – 381с.

176. Розробка методів підвищення якості військового зв’язку АСУ ракетних військ та артилерії”. Шифр «Мрія». Звіт про НДР. № держреєстрації. 0101U000414. Заключний. / Стасєв Ю.В., Приходько С.І., Грабчак В.І., та інші // Харків: ХУПС. - 2005. - 133с. - Інв.№ 1607/2.

АНОТАЦІЯ

Приходько С.І. Методи синтезу, кодування та декодування згорткових кодових конструкцій. – Рукопис.

Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.12.02 – Телекомунікаційні системи та мережі. – Українська державна академія залізничного транспорту, Харків, 2010.

Дисертаційна робота присвячена розробці методів синтезу, кодування та декодування згорткових кодових конструкцій, орієнтованих на застосування в телекомунікаційних системах, що функціонують в умовах малого енергетичного відношення сигнал/шум. Розроблені методи засновані на єдиному концептуальному підході алгебраїчного представлення згорткових кодів у вигляді недвійкових блокових циклічних кодів, що дозволяє розглядати з єдиних теоретичних позицій процедури синтезу, кодування й декодування згорткових кодів з довільними властивостями й кодовими характеристиками й теоретично обґрунтувати аналітичні вирази по оцінці кодових співвідношень синтезованих згорткових кодових конструкцій, аналітично зв'язати їхні параметри й виразити через кодові характеристики відповідних циклічних кодів.

Ключові слова: згортковий код, циклічний код, згорткові кодові конструкції, синдром, турбокод, турбокодер, турбодекодер, ітеративне декодування, перемежувач.

АННОТАЦИЯ

Приходько С.И. Методы синтеза, кодирования и декодирования сверточных кодовых конструкций. – Рукопись.

Диссертация на соискание ученой степени доктора технических наук по специальности 05.12.02 – Телекоммуникационные системы и сети. – Украинская государственная академия железнодорожного транспорта, Харьков, 2010.

Диссертационная работа посвящена разработке методов синтеза, кодирования и декодирования сверточных кодовых конструкций, ориентированных на применение в телекоммуникационных системах, функционирующих в условиях малого энергетического отношения сигнал/шум.

Основными и наиболее эффективными средствами повышения достоверности передаваемой информации являются методы помехоустойчивого кодирования. В теории помехоустойчивого кодирования можно выделить несколько основных направлений развития.

Первое направление базируется на блоковых кодах и, преимущественно, алгебраических методах представления процессов синтеза, кодирования и декодирования. Наибольшее распространение среди блоковых кодов получил обширный класс кодов – циклические коды. Наряду с высокими конструктивными свойствами циклических кодов это направление позволяет строить простые и вычислительно эффективные алгоритмы кодирования и декодирования.

Второе направление развития базируется на непрерывных кодах, подклассом которых являются сверточные коды. Отличительной особенностью сверточных кодов является возможность их простого описания деревом или регулярной решетчатой диаграммой, что позволяет реализовать вероятностное декодирование (алгоритмы последовательного декодирования, алгоритм Витерби, алгоритм максимума апостериорной вероятности). Кодер сверточного кода представляет собой линейный регистр сдвига, сложность которого из-за регулярной решетчатой диаграммы не зависит от длины кода (но зависит от числа состояний решетчатой диаграммы), что является значительным преимуществом.

В качестве третьего направления можно выделить методы каскадного кодирования, появление которых связано с попытками синтеза длинных кодов с высокими кодовыми характеристиками на основе достаточно простых составляющих кодов (которые могут быть как блоковыми, так и сверточными), декодирование которых осуществляется отдельными декодерами. Преимущество каскадных кодов состоит в упрощении алгоритмов декодирования и одновременным повышением общей эффективности кодирования.

Развитая в настоящее время алгебраическая теория блочного кодирования не может быть непосредственно применена к сверточным кодам по причине значительного различия в их свойствах по сравнению с блочными кодами. Несмотря на это существует возможность представления сверточного кода в виде блочного кода полубесконечной длины и его последующим алгебраическим описанием. Однако положительные результаты получены только для ограниченного диапазона низких скоростей кодирования, значения которых не удовлетворяют современным требованиям, предъявляемым к параметрам помехоустойчивых кодов (как правило, на практике требуются более высокие скорости кодирования).

Таким образом, возникает научная проблема, в которой существующие положения теории помехоустойчивого кодирования не позволяют вычислительно реализуемо решать задачи синтеза, кодирования и декодирования сверточных кодов с высокими конструктивными кодовыми характеристиками и с произвольными параметрами. В диссертационной работе данная научная проблема решается путем разработки на основе единого концептуального подхода методов синтеза, кодирования и декодирования алгебраически заданных сверточных кодовых конструкций с требуемыми свойствами и характеристиками.

С использованием методов алгебраической теории блоковых кодов, теории конечных полей и полиномиальных методов описания помехоустойчивых кодов разработаны методы и алгоритмы синтеза алгебраически заданных нерекурсивных и рекурсивных сверточных кодов. Разработаны методы декодирования алгебраически заданных сверточных кодов, основанные на использовании бесконечной серии синдромов кодовых слов циклического кода. Предлагается способ формирования

бесконечной серии синдромов алгебраически заданного сверточного кода. Разрабатывается подход комбинированного декодирования алгебраически заданных сверточных кодов, состоящий в совмещении алгебраических процедур и процедур последовательного поиска по кодовой решетке. Установлено, что применение предложенных процедур позволяет локализовать ошибки в кодовом слове алгебраически заданного сверточного кода и ускорить последовательный поиск по кодовой решетке при комбинированном методе декодирования. Исследуются методы построения параллельных каскадных кодовых конструкций и процедуры их декодирования. Предлагаются схемы турбокодирования с использованием рекурсивных сверточных кодов, заданных через порождающий и/или проверочный многочлены недвоичного циклического кода. Разработаны алгоритмы построения турбокодов с требуемыми параметрами. Исследуются модели каналов связи, разработывается методика оценки достоверности передаваемой информации, которая позволяет для заданных параметров математической модели канала связи с заданной погрешностью оценить вероятность ошибочного приема бита информации и соответствующий энергетический выигрыш от кодирования. Разрабатывается имитационная модель системы передачи информации с использованием алгебраически заданных сверточных кодовых конструкций, которая позволяет оценить эффективность кодирования синтезированными сверточными кодовыми конструкциями. На основе полученных результатов проведенных исследований разработаны практические рекомендации по использованию синтезированных алгебраически заданных кодовых конструкций для повышения достоверности передаваемой информации.

Ключевые слова: сверточный код, циклический код, сверточные кодовые конструкции, синдром, турбокод, турбокодер, турбодекодер, итеративное декодирование, перемежитель.

ABSTRACT

Prihodko S.I. Methods of synthesis, encodings and decoding of convolutional code constructions. - the Manuscript.

The thesis on competition of a scientific degree of a Dr.Sci.Tech. on a speciality 05. 12.02 - Telecommunication systems and webs. - the Ukrainian state academy of a railway transportation, Kharkov, 2009.

Dissertational operation is devoted development of methods synthesis, encoding and decoding of the convolutional code constructions oriented to application in telecommunication systems, functioning in the conditions of a small power signal to noise ratio. The developed methods are grounded on the uniform conceptual approach of algebraic representation of convolutional codes in the form of not binary block cyclic codes that gives the chance to consider from uniform theoretical positions of procedure of synthesis, encoding and decoding of convolutional codes with casual properties and code performances and theoretically to justify analytical expressions according to code relations of the synthesised convolutional code constructions, analytically to link their parametres and to express by means of code performances of appropriate cyclic codes.

Keywords: a convolutional code, a cyclic code, convolutional code constructions, a syndrome, a turbo-code, a turbo-encoder, a turbo-decoder, iterated decoding, interleaver.

Підписано до друку 17.03.2010 р.
Формат паперу 60х84 1/16 Друк. різограф.
Папір офсетний. Обсяг 1,8 друк. арк. Наклад 100 прим.
Зам. № Безкоштовно.

Видавництво УкрДАЗТ.
Свідоцтво про державну реєстрацію ДК № 2874 від 12.06.2007 р.
61050, м. Харків, вул. Фейєрбаха, 7
Друкарня УкрДАЗТу, 61050, м. Харків, вул. Фейєрбаха, 7

Підписано до друку 17.03.2010 р.
Формат паперу 60х84 1/16 Друк. різограф.
Папір офсетний. Обсяг 1,8 друк. арк. Наклад 100 прим.
Зам. № Безкоштовно.

Видавництво УкрДАЗТ.
Свідоцтво про державну реєстрацію ДК № 2874 від 12.06.2007 р.
61050, м. Харків, вул. Фейєрбаха, 7
Друкарня УкрДАЗТу, 61050, м. Харків, вул. Фейєрбаха, 7