

Аннотация

В популярной форме описано современное состояние теории и техники помехоустойчивого кодирования. Рассмотрен алгоритм многопорогового декодирования. Дается сопоставление методов повышения достоверности цифровых данных в каналах с большим уровнем шума.

Для широкого круга читателей, может быть полезна студентам и аспирантам институтов связи и инженерам, занимающимся вопросами повышения достоверности передачи данных по каналам спутниковой, космической и иных видов связи с большим уровнем шума.

1. Кругом помехи и шумы

Вы включили приемник, чтобы скоротать непогоду наедине с музыкой. Но вместе с мелодией вы слышите треск, свист и завывания эфира. Они мешают звукам симфонии дойти до вас во всем своем великолепии. Это помехи, шум разной природы. Но музыку вы слышите, мелодию улавливаете, потому что музыка содержательна, прогнозируема и узнаваема, хотя отдельные ноты вы можете за помехами и не расслышать. Таким образом, радиопередача дошла о вас.

А что делать, если при наличии больших помех нужно с высокой достоверностью принять именно ту информацию, которую хотел послать вам отправитель, да еще и в цифровом виде? А ошибиться можно в среднем только в одном бите из многих тысяч или даже миллионов. При этом переспрашивать сообщения нельзя или можно, но крайне редко. Этими задачами занимается теория помехоустойчивого кодирования, в обязанности которой, среди прочих сложных проблем, входит разработка методов декодирования (обработки) корректирующих кодов, обеспечивающих по возможности быстрое и точное решение о переданном сообщении по искаженному шумами бледному подобию оригинала, достигшему получателя.

Устройства, реализующие в приемнике эту важную миссию, называются декодерами. Над тем, как их сделать как можно проще, трудятся большие коллективы специалистов по всему миру. История этих разработок полна совершенно необычных историй. О части из них мы расскажем тоже.

В конце нашего путешествия по огромному полю поистине спортивных достижений теории и техники кодирования мы предложим вам несколько простых программных образцов многопорогового декодера и других наиболее эффективных алгоритмов декодирования. Но сначала предлагаем внимательно прочесть эту книгу, которую мы старались сделать интересной и полезной. После ознакомления с основными теоретическими и прикладными вопросами, которые рассмотрены далее, ваше

понимание возможностей программных версий многопорогового декодера (МПД) будет более глубоким.

Хотя базовые идеи алгоритмов типа МПД оказываются на удивление простыми, возможно, что некоторые разделы книги для их более полного понимания потребуются перечитать еще один–два раза. Разумеется, мы надеемся, что интерес некоторых наших читателей к очень простым по реализации и необычным по своей очень высокой корректирующей способности методам позволит расширить сферу применения этих алгоритмов в системах связи.

2. Что делает теория кодирования?

Сорокалетний путь теории кодирования изобилует удивительными и неожиданными событиями. Впрочем, и само-то ее рождение вместе с первыми успехами было довольно нетрадиционно. Сначала К. Шеннон указал, что для повышения качества передачи, ее точности не нужно увеличивать энергию передачи, т.е. мощность системы связи. Достаточно только хорошо выбрать сигналы и затем правильно или с допустимой точностью воспринять их и обработать. Инженеры весьма удивились и захотели узнать все о правильном выборе сигналов. Но оказалось, что теория сразу вышла на так называемую теорему существования, сказав, что может быть, а чего не бывает ни при каких условиях. О том же, как выбрать систему сигналов, т.е. наилучшую последовательность битов при передаче по двоичному каналу, например, с независимыми ошибками, сначала было не очень ясно. Кстати, именно этот канал мы и будем далее рассматривать. Поскольку все передаваемые двоичные символы (биты) искажаются в этой модели канала одинаково независимо с вероятностью p_0 , будем называть его двоичным симметричным каналом (ДСК) без памяти. Он обладает многими достоинствами. Во-первых, простота модели позволяет широко использовать ДСК в различных аналитических и численных оценках. Во-вторых, особенно важно, что эта модель канала хорошо соответствует существующим спутниковым, космическим и ряду других реальных и обычно весьма дорогих каналов связи. Это

помогает весьма точно определять потенциальные возможности таких каналов и характеристики конкретных реализаций систем цифровой связи.

Коды, обеспечивающие коррекцию ошибок, характеризуются избыточностью, которая вводится в передаваемое сообщение. Если для передачи 100 битов информации в сообщение добавляется такое же количество проверочных символов, то можно говорить о 100%-ной избыточности выбранного кода. Однако чаще используется понятие кодовой скорости R , равной отношению числа информационных символов кода k к его полной длине $n=k+r$, где r – число избыточных символов.

Пусть далее для только что введенного канала типа ДСК задана вероятность p_0 искажения произвольно выбранного символа сообщения $p_0 < 0,5$. Какова допустимая избыточность кода, его кодовая скорость, чтобы передача с высокой достоверностью была в принципе возможна?

Фундаментальное понятие пропускной способности канала C , введенное Шенноном, как оказалось, очень просто соотносится со скоростью R : всегда должно быть $R < C$! Если это условие выполняется, то канал не перегружается информацией и может доставить получателю не слишком испорченное ошибками сообщение.

Если сообщение достаточно длинное, то доля искаженных символов в нем как раз и будет близка к p_0 . Для ДСК p_0 однозначно определяет C :

$$C = 1 - H(p_0),$$

где $H(x) = -x \log_2 x - (1-x) \log_2 (1-x)$ – двоичная энтропия.

На рис. 1 представлена зависимость $C = C(p_0)$, из которой видно, что с ростом p_0 C быстро убывает. Например, для $p_0 = 0,11$ $C = 0,5$, что и определяет возможности кодов с $R = 1/2$: они могут работать только в каналах типа ДСК с $p_0 < 0,11$. Но если это условие выполняется, то возможна передача с последующим восстановлением истинного двоичного сообщения со сколь угодно большой достоверностью. Правда, эта радость, гарантированная теорией, не без горького привкуса: чтобы реально обеспечить очень малую вероятность ошибки после сколь угодно сложной обработки, нужно иметь достаточно длинный кодовый блок, про

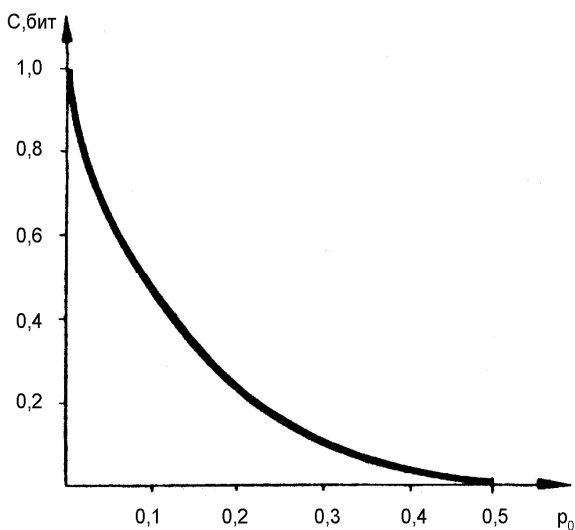


Рис. 1. Зависимость пропускной способности ДСК от вероятности ошибки канала.

метод выбора которого теория, снабдившая инженера сначала лишь теоремой существования, не очень спешила подсказать что-либо конструктивное и полезное. Одновременно с этим и метод обработки такого блока, как догадывались специалисты, не может быть слишком простым.

Кстати, а для каких длин кодов достижимы заданные достоверности? И, как во многих других науках, сразу приходится вместо использования точных выражений переходить к оценкам, хотя во многих случаях они оказываются довольно простыми и удобными, а также, что уж совсем кстати, достаточно точными для многих приложений.

Давайте попробуем получить такие оценки, лучше которых нельзя сделать ничего.

3. Если самый-самый хороший

Для определения соотношений между длиной кодов и достоверностью декодирования введем важнейшее понятие кодового расстояния d . Если выбрать двоичный вектор (последовательность) длины n , то всего возможно 2^n таких векторов, причем для каждого из них есть другой вектор, отличающийся от него только в одном символе.

Конечно, число таких «соседей» у любой последовательности равно n . Количество позиций, в которых два двоичных вектора равной длины отличаются, будем называть расстоянием Хемминга между ними. А для некоторого множества таких векторов назовем расстоянием d минимум по всем попарным расстояниям по Хеммингу между векторами этого множества. Это и есть кодовое расстояние, если множество – код. Значит, для полного набора двоичных векторов длины n в приведенном примере расстояние равно 1.

Хорошо известны последовательности, расстояние в которых равно 2. Для их выбора подсчитаем в последовательностях длины $(n-1)$ количество единичек. Если их будет четное число, то n -й символ в каждом таком векторе выберем равным 0, а если нечетное, то последний символ пусть будет единицей. Тогда получим код контроля по четности.

Так что же такое код? Множество допустимых сообщений! Если векторы с нечетным весом недопустимы, то вы можете проверить расстояние между словами кода проверки на четность. При нашем методе отбора слов для этого кода (и любом другом) расстояние между ними всегда будет равно двум или больше. Это его кодовое расстояние. И такой код уже может обнаруживать одну ошибку, так как при единственном искажении любого из символов вектора длины n сумма по модулю 2 становится равной 1, что и обнаруживает ошибку. Но не исправляет ее!

Для исправления хотя бы одного искажения расстояние должно быть равным 3. Таковы коды Хемминга, широко используемые при защите машинной памяти. Но число дополнительных избыточных символов в коде будет при этом уже суще-

ственно больше за счет дополнительных контрольных разрядов. В коде Хемминга длины n , число избыточных символов равно $r = \log_2 n$. Эти коды можно назвать совершенными. Они составляют весьма ничтожную долю от общего числа всех кодов, и обладают замечательным свойством, которое, вообще говоря, абсолютно не типично для произвольно взятого кода: любая точка в n -мерном пространстве (а каждому вектору длины n , конечно, можно сопоставить точку этого пространства) находится на расстоянии не более $d/2$ хотя бы относительно одного кодового слова.

Иначе говоря, если d нечетно, то любая точка попадает в n -мерный шар радиусом $(d-1)/2$ около единственного кодового слова, и в канале типа ДСК наилучшие (т.е. оптимальные!) решения о переданном сообщении должны приниматься именно по этому правилу: считается, что передано ближайшее к принятому вектору кодовое слово. Вы легко можете проверить, что, например, для кода Хемминга с $n=15$ и $r=n-k=4$ проверочными символами при 2^{11} кодовых словах и n точках, удаленных от каждого из них на 1, общая сумма всех возможных сообщений длины 15 будет равной как раз 2^{15} , что и иллюстрирует приведенные рассуждения.

При желании исправлять все ошибки веса t_0 кодовое расстояние должно быть хотя бы $d=2t_0+1$. Конечно, лучше выбирать более высокие значения d при заданной избыточности, т.е. величине R . Но тогда и обработка вектора, в котором есть ошибки, будет усложняться. Но самое главное, хотя и вполне ожидаемое, обстоятельство состоит в том, что при заданном R существуют коды лишь с ограниченными значениями d .

Совершенные коды можно также называть сферически плотно упакованными, так как они буквально все многомерное пространство без остатка делят на «сферы влияния». Эта их особенность и позволила находить полезные оценки для потенциальной помехоустойчивости кодов через оценки для совершенных кодов, которых очень мало, а среди длинных фактически и нет.

Эти чрезвычайно важные для теории кодирования оценки можно получить на основе очень простых рассуждений. Пусть

есть код длины n с некоторым значением кодовой скорости $R=k/n$. Тогда для 2^k кодовых слов в сферу каждого из них попадает 2^{n-k} точек пространства. А поскольку сфера радиуса m должна содержать C_n^m точек, то все 2^r точек будут перечислены, если окажется, что t_0 такое максимальное число, что сумма всех C_n^m , $0 < m < t_0$, меньше чем 2^r . Это значит, что все 2^r выделенные для каждого кодового вектора последовательности уже использованы для формирования сфер радиуса t_0 и, значит, нет способа дальнейшего увеличения расстояния между 2^k кодовыми словами. Величина t_0 и определяла бы корректирующую способность совершенного кода. Для него $d=2t_0+1$, и найти лучшие коды уже нельзя. Тогда вероятность ошибки декодирования при его использовании определяется вероятностями появления ошибок веса более $d/2$ через обычное биномиальное распределение. И хотя совершенных кодов почти нет, оценки, полученные из приведенных соображений, оказываются вполне приемлемыми для расчета помехоустойчивости в больших шумах.

Весьма интересно, что описанный метод получения оценки для t_0 фактически является способом вывода границы Хемминга для кодового расстояния, но более реалистично смотрится граница Варшавова-Гилберта: $H(d/n)=1-R$, функцию энтропии для которой мы уже определили выше. При больших n она гарантирует наличие только кодов с кодовым расстоянием фактически вдвое меньшим, чем граница Хемминга для сферической упаковки. Она же определяет и вероятностные корректирующие возможности реально существующих кодов при малом шуме.

Однако мы хотим успешно работать при значительных уровнях шума в канале. Именно поэтому сейчас мы и воспользуемся оценками вероятностей ошибки декодирования через найденные выше оценки для t_0 .

Для кодов с $R=1/2$ зависимости вероятности ошибки декодирования $P_B(e)$ для блока длины n в ДСК от вероятности p_0 представлены для разных n на рис. 2 сплошными линиями, а оценки вероятностей ошибки на бит $P_b(e)$ – пунктирными. Как видим, если выбирать длинные коды, то можно достичь хороших показателей помехоустойчивости при $p_0 < 0,11$. Так в чем же пробле-

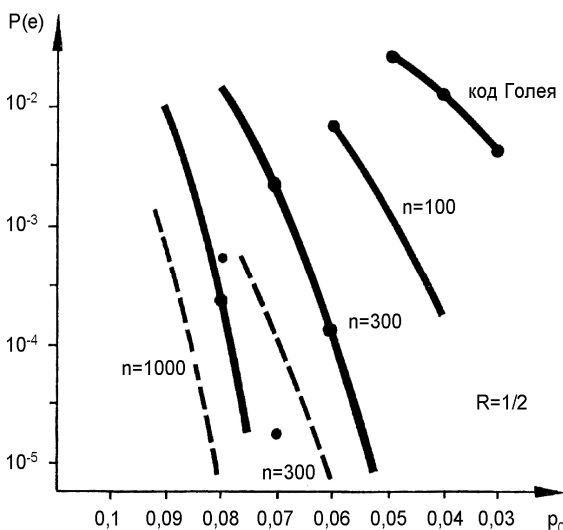


Рис. 2. Нижние оценки вероятностей ошибки блочного кода с $R=1/2$ в ДСК в зависимости от уровня шума канала

ма? Да в том, что хороших методов декодирования для кодов с большим n просто нет. Нужно перебирать 2^r или 2^k вариантов решений декодера, чтобы выяснить, какое кодовое слово наиболее правдоподобно для данного вектора шума, и если код выбран случайно, то при количестве вариантов всего 2^{150} процедура декодирования на самом деле может затянуться весьма надолго. Но из графиков мы видим, что и при $n \sim 1000$ характеристики кодов еще очень далеки от желаемых. Разница с границей C для $R=1/2$, равной $p_0=0,11$ оказывается все еще очень значительной. Число допустимых кодовых комбинаций в этом коде составляет уже 2^{500} , что равно приблизительно $3 \cdot 10^{150}$. Это число, кажется, превышает даже количество атомов во Вселенной. Таковы возможности оптимальных декодеров (ОД) с полным перебором. Слабо, да еще с такими сложностями!

А что же можно сделать проще? Вот эта задача и является основной проблемой теории кодирования: построение по воз-

возможности более простых методов декодирования, которые были бы не намного хуже переборных ОД. Этим в течение многих уже десятилетий занимаются специалисты, работающие в области теории и прикладных методов помехоустойчивого кодирования.

4. Так много способов!

Гораздо проще можно декодировать другими, непереборными способами. Одними из первых в теории кодирования появились коды Боуза-Чоудхури-Хоквингема (БЧХ) и методы их декодирования. Разработанные для них достаточно простые версии алгоритмов декодирования не могут исправить те сообщения, в которых число искаженных символов превышает $d/2$, тогда как другим методам это под силу. Но сложность декодирования этих кодов как число необходимых операций была по порядку величины даже существенно меньше, чем n^2 (сравните с ОД и сложностью 2^k), что в свое время было очень серьезным достижением.

Во многих случаях можно считать, что число ошибок в информационных символах неправильно декодированного блока близко к $dR/2$, что приводит к вероятности ошибки на бит $P_b(e) \sim dRP_B(e)/(2n)$. На рис. 3 представлены характеристики эффективности кодов БЧХ. Сплошными линиями указаны вероятности $P_b(e)$ для этих кодов. А примеры зависимостей $P_b(e)$ от p_0 для этих кодов также показаны пунктирными линиями.

Переход к $P_b(e)$ очень полезен, так как далее будут рассмотрены не только блочные, но и сверточные коды, которые чаще сравнивают именно по этому параметру. Хотя при анализе и использовании блочных и сверточных кодов сразу видны некоторые различия, покажем на примере, что между ними есть и значительная общность. На рис. 4,а приведен вариант кодера двоичного блочного кода, а рис. 4,б показывает его преобразование в сверточный кодер.

В блочном коде 13 информационных битов записываются, например, в регистр, а далее начинается собственно процесс кодирования. Замыкается ключ К1, и регистр сдвига кодера стано-

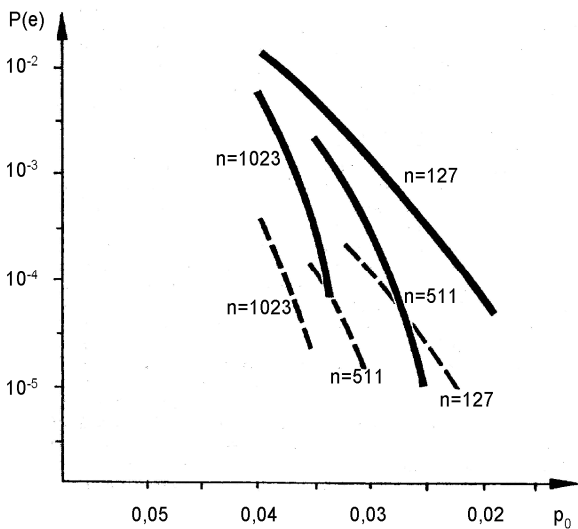


Рис. 3. Вероятности ошибки декодирования блока кода БЧХ в ДСК при $R=1/2$ (сплошные линии) и вероятности ошибки на бит (пунктирные линии)

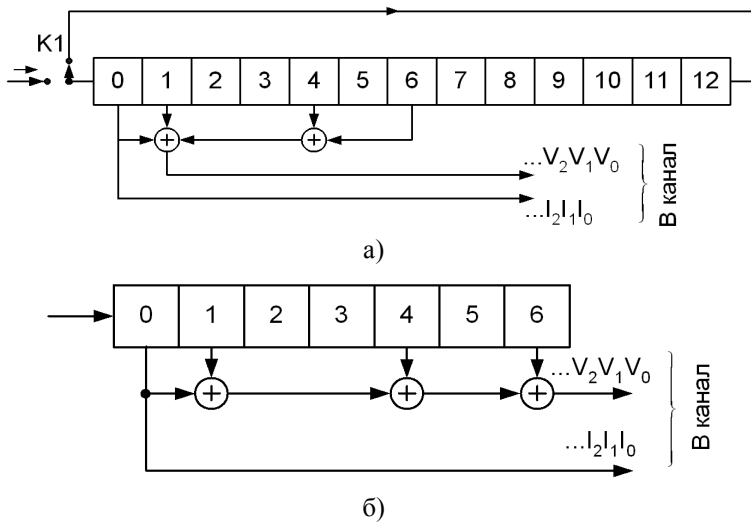


Рис. 4. Кодеры блокового (а) и сверточного (б) кодов с $R=1/2$

вится циклическим. Связи ячеек с многовходовым полусумматором (сумматором по модулю 2) указывают конкретную процедуру формирования проверочных разрядов и, следовательно, сам код.

После замыкания ключа с выхода полусумматора первый проверочный символ поступает в канал, например, с первым же информационным символом. Потом при циклическом сдвиге информационных символов в регистре последовательно получаются все 13 проверочных символов. Таким образом, мы рассмотрели процедуру кодирования (n, k, d) -кодом (26, 13, 5).

Данный код относится к линейным. Это значит, что сумма двух любых кодовых слов a и b также будет кодовым словом $c=a+b$, полагая, что сложение векторов (кодовых слов) производится покомпонентно. Линейность обеспечивает очень малую сложность технической реализации кодера: для многих классов таких кодов для генерации 2^k кодовых слов нужен лишь регистр сдвига длины k битов и один или несколько полусумматоров.

Важнейшим свойством линейности является то, что нулевая последовательность всегда будет кодовым словом. Данное очень полезное обстоятельство значительно упрощает многие рассуждения при изучении линейных кодов. Поэтому одним из соображений (пока еще не доказанным) в пользу того, что в рассмотренном коде $d=5$, оказывается тот факт, что вес (число единиц) w любого кодового слова с одной информационной единицей в рассматриваемом примере кода равен $w=5$. Это очевидно, так как в процессе генерации кода помещенная первоначально в любое место информационного регистра единица (при нулях на всех прочих позициях) обязательно проходит за 13 тактов сдвига через все четыре ячейки, с которых подаются сигналы на полусумматор, что приводит к генерации еще четырех единичек в проверочных символах кода. К оценкам кодовых расстояний мы еще будем возвращаться не раз. А пока перейдем к сверточным кодам.

На рис. 4,б представлен кодер сверточного кода также с $R=1/2$ и $d=5$. В кодер, первоначально содержащий только нули, поступает произвольно длинная информационная последовательность по одному биту за каждый такт. Самые правые сим-

волы регистра при сдвиге просто теряются. Поскольку общая длина кода, с учетом избыточности, удваивается, то она становится равной $n_A=14$ и называется длиной кодового ограничения. Показанный пример иллюстрирует связь между блоковыми и сверточными кодами.

Не вдаваясь пока в детали описания алгоритмов, рассмотрим вероятности ошибки декодирования весьма эффективным алгоритмом Витерби (АВ) для широко используемого в технике связи сверточного кода с длиной кодирующего регистра $K=7$ при $R=1/2$ в рассматриваемом нами канале типа ДСК.

Кодовое расстояние указанного кода равно 10 и количество кодовых слов такого веса, а также близких ему, т.е. 12 и более, составляет несколько десятков, что и определяет его помехоустойчивость при малых вероятностях ошибки в канале. Поскольку АВ относится к оптимальным алгоритмам, обеспечивающим минимальную вероятность ошибки решения для используемых кодов, то фактически мы говорим о потенциальной помехоустойчивости этого кода, которая может быть реализована при наилучших способах обработки, в частности, на основе АВ. Однако сложность реализации АВ имеет порядок 2^{K-1} , что и ограничивает длины тех кодов, для которых он может быть создан.

График зависимости вероятности $P_b(e)$ от p_0 представлен для АВ на рис. 5. Существуют также последовательные алгоритмы Фано и стекового типа (ПА), теоретические возможности которых характеризуются тем, что они могут обычно работать только при $R < R_1$, где R_1 – вычислительная скорость канала, соответствующая $R_1=1/2$ при $p_0=0,045$. Ясно, что это существенно хуже, чем $p_0=0,11$ для $C=1/2$. Возможности весьма сложного ПА также можно увидеть из рис. 5, который построен в предположении, что память декодера составляет порядка 10^5 битов для оперативных вычислений. У этих алгоритмов много специфических свойств и недостатков определенного вида, которые несколько ограничивают их возможности.

К ним относятся, например, частые стирания декодируемых блоков данных, которые невозможно декодировать из-за большого числа ошибок, оказавшихся в них, и превышении вследст-

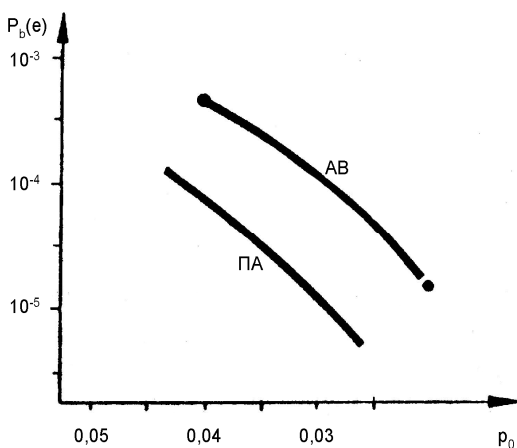


Рис. 5. Характеристики алгоритма Витерби и последовательных процедур в ДСК

вие этого времени, отпускаемого на декодирование. Кроме того, декодер выдает внешнему потребителю поток декодированных символов в весьма неравномерном темпе, что также связано с существенно неодинаковыми по объему вычислениями при вынесении решений о различных символах сообщения.

5. Самый незатейливый

Если для описания декодера кода БЧХ следует изучить некоторые разделы алгебры конечных полей, то наипростейший из известных пороговый декодер (ПД) имеет и самое наглядное описание своей работы. Этот декодер при аппаратной реализации представлен на рис. 6. Мы будем пока говорить о блоковом декодере, но переход к сверточному будет столь же простым, как и при описании кодеров. На полусумматоре А результаты работы кодера из пунктирной рамки (а это проверочные символы) складываются с другими проверочными символами, принятыми из канала связи. Конечно, их сумма по mod 2 всегда даст 0, если в канале не было внесено в сообщение никаких ис-

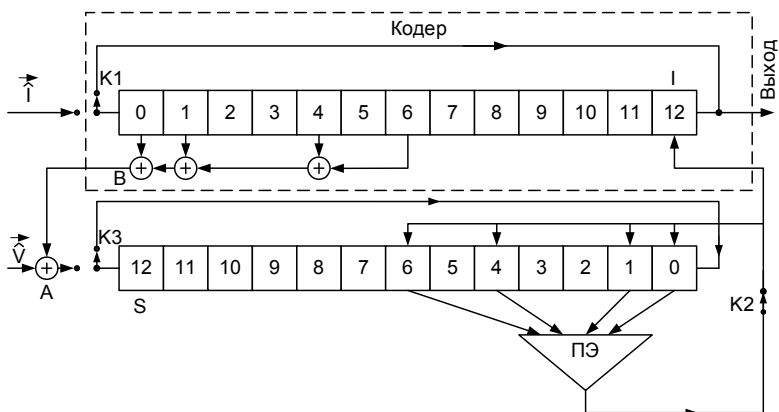


Рис. 6. Схема порогового декодера блочного (26, 13, 5)-кода

кажений. Это обстоятельство хорошо иллюстрирует главное свойство вектора синдрома: его вид полностью определяется вектором ошибки, который исказил при передаче принятое сообщение, и совершенно не связан с информационной частью кодового вектора.

Но если из канала в декодер попали ошибки, то в регистр синдрома S , с которым соединен полусумматор A , начнут поступать единицы. Причем, единственная ошибка в проверочных символах приведет к появлению одной единицы в регистре S декодера на месте, соответствующем ошибочному проверочному символу. Будем считать, что при вычислении синдрома пороговый элемент (ПЭ) декодера отключен ключом $K2$ и не мешает выполнению этой процедуры. При выполнении главного второго шага декодирования этот ключ замкнут и ПЭ вырабатывает на своем выходе 1, если число единиц на входе больше двух, а иначе на его выходе будет 0. Ясно, что единственная ошибка в проверочном символе кода приведет к тому, что на любом из 13 сдвигов синдромного (и одновременно, конечно, информационного) регистра сумма единиц на пороге не может быть более одной и пороговый элемент не срабатывает, так как на его выходе сигнал остается равным 0.

Теперь разберем важнейший случай исправления единственной ошибки, происшедшей в информационном символе кода. Пусть при заполнении информационного регистра I декодера ошибка канала попала в ячейку 9. Воспользуемся свойством линейности кода, которое мы уже обсудили, и будем считать, что посылалось нулевое кодовое слово. Это облегчает описание декодирования, так как ошибки и единички будут означать при дальнейших рассуждениях одно и то же. А так как вид синдромного регистра (подумайте еще раз, почему это следует из линейности) зависит только от принятого вектора ошибок, дальнейший анализ будет сохранять требуемую общность.

Поскольку процедура кодирования уже подробно разбиралась ранее, внимательный читатель может проверить, что в результате вычисления синдрома, совпадающего просто с кодированием, так как на второй вход полусумматора при этом приходят одни нули, получается заполнение синдромного регистра S , соответствующее первой строке табл. 1, где указаны результаты и последующих сдвигов синдрома с учетом решения ПЭ.

Если ПЭ принимает решение «1», то через цепь обратной связи при сдвиге инвертируются именно все те ячейки регистра S , с которых поступали сигналы на входе ПЭ. Это обстоятельство и нужно учесть при анализе 10-й и 11-й строк таблицы. Ясно, что только когда ошибка в регистре I станет в крайней правой позиции, сумма на ПЭ превысит 2, и поэтому ошибка исправится.

Но очень важно здесь и то, что при попытке декодирования предыдущих символов сумма на ПЭ не превышает 1 ни разу. Более того, если бы мы разомкнули ключ 2, не исправив при этом ошибки, то сумма на ПЭ и относительно последующих символов также не превысила бы 1. А в этом случае оказывается возможным исправление еще одного любого другого информационного символа в принятом блоке, поскольку каждая ошибка канала в данном коде искажает лишь одну проверку.

Значит, сумма на ПЭ не будет более двух на правильных символах и окажется не менее трех на ошибочных, тех, которые нужно корректировать. Это гарантированное исправление $t_0=2$ ошибок и доказывает, наконец, что для рассматриваемых нами в качестве примера кодов $d=2t_0+1=5$.

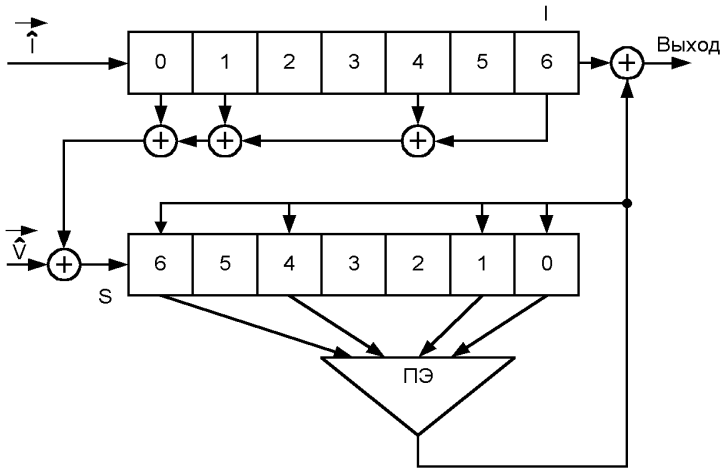


Рис. 7. Декодер сверточного кода с $R=1/2$ и $d=5$

представлена при $R=1/2$ для $d=11$, $n=182$ и $d=15$, $n=366$ кривыми на рис. 8.

Заметим, что результаты весьма скромные. Сопоставление ПД с декодерами кода БЧХ на рис. 3 показывает, что они приблизительно одинаковы, поскольку обеспечивают вероятности $P_b(e) \approx 10^{-5}$ при $p_0=0,01-0,02$. Мелкие детали здесь неважны. Для нашего обсуждения пока что существенно только то, что эти вероятности p_0 , при которых ПД эффективен, много меньше, чем $p_0=0,11$, когда $C=1/2$, и даже меньше, чем $p_0=0,045$, когда $R_1=1/2$. Таким образом, теоретические границы значительно лучше, чем те значения p_0 , при которых могут результативно работать ПД. Но хорошо, что все же сложность, т.е. число операций в ПД, около d в пересчете на каждый информационный символ. Это, конечно, уже не экспонента.

Давайте все же более внимательно посмотрим, почему так слабы результаты у обычного порогового декодера? Может, мы что-то не доглядели? Ведь при элементарно простом алгоритме ПД имеет еще очень важное свойство – возможность исправлять

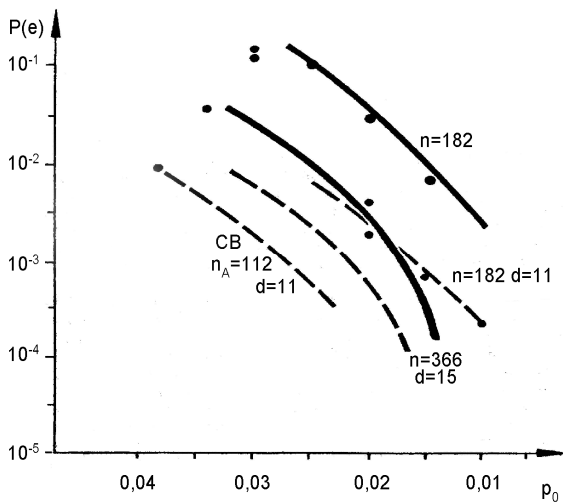


Рис. 8. Характеристики ПД блоковых и сверточных кодов

некоторые (не все!) ошибки большего веса, чем $d/2$. Для большинства других методов это невозможно, а вот ПД делает это, но, действительно, как-то незаметно. Перебирая разные сочетания ошибок даже для простого декодера кода, представленного на рис. 6, можно легко найти такие конфигурации из трех ошибок, которые на самом деле будут исправлены с помощью ПД. (Проверьте: ошибки в информационных позициях 0, 9 и 10 исправляются!) Алгебраические декодеры обычно не могут исправлять такие ошибки, а здесь можно.

И вот здесь возникает этот важный в науке вопрос: почему? А вдруг именно здесь надо хорошо копнуть, и потом найдется повод воскликнуть: «Эврика!». Кстати, рассмотрите, пожалуйста, сочетания ошибок веса 4, конечно, помещая их в синдромный регистр. Кое-что ПД исправит и здесь! А это почему? Может что-то совсем не так просто в теории кодирования?

Так почему же ПД исправляет некоторые ошибки веса более t_0 , гарантируемого минимальным кодовым расстоянием d используемого кода?

6. Так что же можно сделать?

Как мы видим, кроме алгоритма Витерби, относящегося к переборным, очень сложным методам, все другие оказываются очень малоэффективными. И, тем не менее, здесь следует отметить, что среди перечисленных выше методов, оказывается, пороговые алгоритмы обладают несколькими важнейшими свойствами, полной совокупностью которых не обладает ни один другой алгоритм.

Перечислим важнейшие с нашей точки зрения свойства пороговых декодеров (ПД) снова, поскольку мы уже обсудили их выше:

- а) минимальная сложность реализации декодирования;
- б) способность корректировать гораздо большее число ошибок, чем половина кодового расстояния. На это обратил внимание сам Дж. Мессе еще в самой первой своей книге «Пороговое декодирование», вышедшей в 1963 г., где привел яркие примеры такой «избыточной» корректирующей способности;
- в) хорошие возможности повторного декодирования, по крайней мере, при малом уровне шума;
- г) наибольшая степень готовности, приспособленности к реализации совершенно новых принципов декодирования.

Рассмотрим возможности мажоритарных декодеров по дополнительной корректирующей способности. Они определяются несколькими механизмами. Самый простой из них очень нагляден.

Дело в том, что пороговый декодер выполняет очень упрощенные вычисления: суммирует проверки и смотрит результаты голосования. А как получаются эти суммы проверок, он не анализирует. Это было бы уж слишком сложно для него. Так что механизм «суперкорректирующих» способностей ПД (по меньшей мере, один его аспект) очень прост. Пусть для некоторого декодируемого правильно принятого символа i_j в декодере на рис. 6 искажены $t_0=2$ проверки и, значит, этот символ можно корректировать правильно. Возьмем еще одну любую из оставшихся двух проверок и поместим в нее 2 ошибки. Так как в каждой проверке суммируемые символы складываются при вы-

числении синдрома по модулю 2, то новая проверка будет по-прежнему правильной и i_j будет пороговым элементом определен тоже правильно. Это один из механизмов, которые действуют в рамках алгоритма ПД: «сумма четного числа ошибок в проверке \rightarrow правильная проверка».

Но на самом деле, конечно, не все так просто. В приведенном выше примере общее число ошибок в коде равно t_0+2 . Так как мы допустили, что декодируемый символ был принят правильно, то после перехода к декодированию какого-то другого символа в рассматриваемом блоке кода останется этот же набор ошибок веса t_0+2 , и следующие шаги декодирования все же могут привести к ошибочному решению о некоторых информационных символах кода. Так что данный механизм действительно имеет место, хотя он не слишком прост, если внимательно его анализировать.

Рассмотрим еще один полезный аспект, связанный с корректирующей способностью мажоритарных схем: увеличить значение порога в ПД. Иногда это может помочь. Давайте ограничимся качественным комментарием. Если шум в канале с независимыми ошибками (а мы рассматриваем только его) достаточно велик, то увеличение порога может немного уменьшить вероятность ошибки декодера. В самом деле, если мы поднимем величину порога в пороговом элементе (ПЭ) рассматриваемого нами ПД, то в проверках будет необходимо найти уже более половины неправильных, чтобы ПЭ изменил значение правильно принятого информационного символа. Значит, декодер будет вносить меньше собственных ошибок в декодируемый поток. Отметим только, что предыдущее объяснение дополнительной корректирующей способности ПД было комбинаторного свойства, а последнее – чисто вероятностное.

Заметим сразу, что при малом шуме снижение порога бесполезно. Так вот при анализе работы ПД и тех новых декодеров, которые будут рассмотрены далее, нужно рассматривать многие процессы и с комбинаторной точки зрения, и с вероятностной. Например, конечная эффективность кодирования часто выражается как вероятность ошибки при некотором отношении сигнал/шум. А оценки сложности алгоритма чаще используют ал-

горитмические свойства декодера и, частично, иногда, также комбинаторику.

Итак, какие еще могут быть способы объяснения особых возможностей ПД по исправлению ошибок большого веса?

Здесь следует отметить, что наиболее заметным образом это свойство проявляется только для случая применения очень длинных кодов. Дело в том, что для самоортогональных кодов (СОК) проверки относительно разных информационных символов образуются из разных ошибок канала. И вот если это так, то большинство ошибок, которые искажают проверки относительно одного символа, отсутствуют в проверках относительно других. Значит, если до некоторой степени увеличивать число ошибок канала в таких кодах, то они будут поражать проверки относительно небольшой доли декодируемых символов. В очень длинном коде можно разбросать очень много ошибок так, что их общее число будет существенно больше, чем t_0 , а относительно каждого информационного символа в блоке число правильных проверок все равно будет больше и, значит, блок, скорее всего, будет исправлен пороговым декодером правильно.

Приведем простой численный пример, объясняющий эту комбинаторную механику. В коде, который декодируется декодером на рис. 6, как мы уже обсуждали, в проверках присутствуют ошибки во всех информационных символах блока. Поэтому в случае, если в блоке число ошибок канала было более t_0 , то вполне возможно, что и первый и второй информационные символы будут декодированы неправильно, хотя, конечно, как мы уже разобрали, не всегда. Заметим, что в блоковом коде при $d=5$ есть $J=d-1=4$ ошибки в каждой проверке, и одна из этих ошибок всегда проверочная. Тогда в четырех проверках всего есть $3 \cdot 4=12$ информационных ошибок и еще одна – это сам декодируемый символ. Т.е. полное минимальное число информационных ошибок в проверках полностью совпало с длиной информационного регистра кода. В общем случае довольно сложно, а иногда и просто невозможно так подобрать параметры СОК, чтобы все информационные ошибки входили в проверки. Но для не очень большой длины кодов можно найти

многие такие СОК, что эти различия будут невелики. А вот если взять очень длинный код, то картина будет совсем другой. Возьмем код длины 1000, в котором $R=1/2$ и $d=11$. Тогда в 10 проверках кода будет по 9 информационных ошибок, а с учетом декодируемого символа общее число информационных ошибок в проверочных соотношениях будет равно 91. Допустим теперь, что относительно некоторого информационного символа число ошибок в проверках равно 5. Это значит, что они как-то будут распределены между указанными 91 информационными символами и 10 проверочными символами, которые также обязательно находятся в проверках. Но в этом случае некоторое число ошибок в остальных $500-91=409$ информационных и 490 проверочных символах тоже могут присутствовать. Они, как мы обсудили, не попадают в проверки относительно первого символа кода. Значит, декодируемый символ гарантированно будет исправлен. А остальные символы кода также могут иметь значительные вероятности быть исправленными, если дополнительных ошибок за пределами проверок для первого символа немного. Данный механизм представляется ведущим в общей способности ПД корректировать «сверх меры». Его и имеет смысл изучать, чтобы потом использовать, если это окажется возможным.

А теперь первое задание для желающих. Оно состоит из двух пунктов. Сначала надо построить самоортогональный код с $d=11$ и $R=1/2$. Это делается чрезвычайно просто. Нужно найти такие 10 положительных целых чисел, чтобы все возможные их попарные разности были различны. Длина кодового блока в этом случае должна быть не менее чем на 2 больше максимального из этих 10 чисел. Найденные числа и будут коэффициентами порождающего полинома для СОК. Если вы хотите избежать этого задания, но выполнить второе из них, то можете взять порождающие полиномы из любой известной книги по кодированию. Там есть таблицы для многих СОК. А второе задание просто состоит в том, чтобы средствами компьютерного моделирования посмотреть, насколько велика на самом деле «избыточная мощь» ПД. Поскольку этот декодер чрезвычайно прост, то можно написать программу его работы буквально

из 15–20 команд, в которые может уложиться даже какой-нибудь несложный генератор ошибок канала. И если посмотреть его корректирующие способности просто в обычном ДСК, то легко найти какие-то удобные оценки для рассматриваемых свойств ПД. У вас 3–7 команд потребуется на создание ошибок, которые надо разместить в декодере. А затем по 3–5 операторов надо написать для вычисления синдрома, подсчета сумм на пороге и исправления проверок декодируемых символов, если сумма на пороге превысит 5. Эта же микропрограммка поможет вам понять и дальнейшие возможные превращения и улучшения в мажоритарных алгоритмах, которые мы будем далее обсуждать.

7. Наши первые успехи

Вот что следует из результатов одного варианта моделирования работы ПД на компьютере. Для блока длины 1000, $d=11$ и с $R=1/2$ моделировался случайный поток ошибок, т.е. именно ДСК. Вероятность ошибки была равна $p_0=0,035$. Самоортогональный код был очень быстро построен на бумаге. Оказалось, что это действительно можно сделать просто за 5–7 минут. Проверить себя в плане правильности таких построений можно, например, по книге Касами (с соавторами) «Теория кодирования». Можно ли считать, что любой СОК с этими параметрами даст приблизительно одинаковые оценки тех параметров, которые мы хотим оценить? Да. В целом это будет верно.

Если вы тоже выполнили оба задания, предложенных вам в разделе 6, то у вас сразу возникнут многие вопросы. Причина их совершенно очевидна: очень уж необычные выводы могут быть сделаны даже при анализе таких простеньких экспериментальных результатов. Все коды с $R=1/2$ и $d=11$ действительно примерно близки по своим возможностям дополнительной коррекции ошибок. Много интересных деталей в плане группирования ошибок мы сейчас обсуждать не будем. А пока надо просто качественно ощутить наличие важнейшей способности ПД. Именно такие успешные особенно показательные примеры помогут нам сориентироваться в наиболее правильных направ-

лениях ваших возможных исследований. И что же конкретно получилось у нас?

Для того чтобы можно было быстро оценить и обработать эксперимент вручную, ограничимся декодированием 20 блоков при $p_0=0,035$. Из них ровно половина, т.е. 10 были декодированы верно. А среднее число ошибок в этих 10-ти блоках оказалось равным 32. И это при $d=11$ и $t_0=5$! Получается, что дополнительная корректирующая способность более чем в 6 раз превышает гарантированную! Действительно, это неспроста! Эти коды (и алгоритм ПД!) заслуживают очень пристального внимания.

Очень важно иметь в виду, что есть много ошибок веса t_0+1 , когда на этих кодах ошибется даже оптимальный переборный алгоритм. В этом и состоит смысл кодового расстояния: превышение границы t_0 обязательно приведет, хотя бы иногда, к тому, что вектор ошибки переведет переданное кодовое слово в вектор, который находится на расстоянии не более t_0 уже от другого кодового слова. А это уже гарантия ошибки. Другое дело, что вероятность этих событий много меньше, чем средняя вероятность ошибки ПД.

Кстати, нелишне посмотреть, что же было с теми блоками, которые ПД декодировал ошибочно. Почти во всех блоках конечное число ошибок немного, но уменьшалось по сравнению с исходным. Были блоки с совсем небольшим числом оставшихся ошибок. И был блок с одной оставшейся ошибкой. Так и хочется спросить, а повторное декодирование помогло бы улучшить характеристики ПД? Совершенно необходимый вопрос. И ответ на него весьма актуален. Но придется немножко потерпеть.

Поскольку мы сейчас не анализировали тонкие свойства того самоортогонального кода, который мы и кто-то из вас, как мы надеемся, построил, скажем пока только, что повторная попытка декодирования имеет смысл. А вот если рассматривать возможности декодирования на основе мажоритарных методов при $p_0 \approx 0,12$ и выше, что требуется сейчас на всемирном конкурсе алгоритмов при $R \approx 1/2$ в гауссовских каналах, то мы быстро увидим, что радость, которую можно испытать, глядя на эксперименты при малом уровне шума, весьма преждевременна. Где-

нибудь в 1975 г. это было бы чрезвычайно интересно. Так оно и было в те годы. А вот теперь следует ориентироваться на современные требования к системам связи. Так что описываемые нами сейчас эксперименты суть просто краткий обязательный экскурс в историю, который показывает очень непростые пути развития действительно передовых для сегодняшнего дня методов кодирования.

У нас есть результаты еще одного, немного другого эксперимента. Он похож на первый, но выполнен при более высоком уровне входного шума. А у нас при $p_0 \approx 0,045$ из 40 декодированных блоков правильными оказались 6 со средней плотностью ошибок канала в них, равной 0,039. И в этом случае тоже были блоки с небольшим оставшимся числом ошибок. Например, с одной ошибкой – 4 блока. Так что они тоже хорошие кандидаты для успешной коррекции после повторной попытки декодирования.

Но чтобы продемонстрировать более наглядно удивительные и неустаревавшие до сих пор свойства мажоритарных методов декодирования, рассмотрим эксперимент немного другого рода.

Мы собрали очень большой экспериментальный материал, который также сразу был обработан компьютером. Параметры кодового блока, разумеется, те же, что и в предыдущих экспериментах, которые мы уже обсудили, а вот число декодированных кодовых блоков – более 10 000. Алгоритм ПД очень прост. Так что времени на весь эксперимент наш совсем не быстрый компьютер потратил менее пяти минут. Кроме того, идея повторной коррекции ошибок тоже очень проста и слишком сильно заманчива, чтобы ее не замечать. Поэтому мы и показываем одновременно и двукратное мажоритарное декодирование всех кодовых блоков.

Итоговые результаты представлены на рис. 9 графически. По оси абсцисс отложено общее число ошибок канала в кодовом блоке перед его декодированием, а по оси ординат – доля правильно декодированных блоков. Нижняя кривая – для первой попытки коррекции. А верхняя – для второй коррекции этого же блока. Как видим из графиков, даже при общем числе ошибок

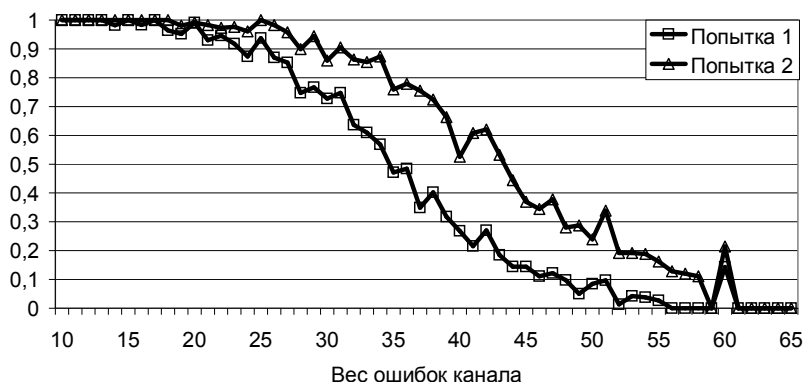


Рис. 9. Характеристики мажоритарного декодирования для блочного СОК с $d=11$, $R=1/2$ и $n=1000$

канала порядка 40 доля правильно обработанных блоков после первой попытки более четверти, а после второй уже превышает половину. Значит, повторное декодирование действительно полезно! Однако вот при числе ошибок в блоке порядка 50 и более алгоритм почти уже совсем не работает. Но ведь и при 40 ошибках ПД достаточно часто исправляет в 8 раз большее число ошибок, чем это гарантировано минимальным кодовым расстоянием $d=11$! А вот третья попытка декодирования практически ничего уже не дает. Существенного улучшения характеристик больше не происходит. Мы не отвлекаем вас от сути обсуждаемого процесса и поэтому не приводим графики для эффективности третьей попытки декодирования.

Все, связанное с методами повторной коррекции, полезное в дальнейших исследованиях, мы уже обсудили. Этот эффект очень важен, а его масштабы весьма значительны. Это можно утверждать просто потому, что число исправляемых ошибок в блоке многократно превышает гарантированный уровень.

Кстати, у Дж. Месси приведен не менее впечатляющий пример ПД с ортогонализируемым кодом, где корректирующая способность кода превышает в длинном коде на много десят-

ков единиц, если рассматривать их абсолютное количество. Напомним, что код ортогонализуем, если на порог поступают линейные комбинации символов синдрома (проверки) такие, что относительно декодируемого символа сохраняется свойство ортогональности проверок.

8. Взгляд иной на эту тему

Давайте теперь попробуем удивиться тому, что существует постановка задачи декодирования, которая очень естественна, очень понятна, но никогда не была озвучена. Конечно, наилучший оптимальный декодер – это такой, который всегда умеет выбрать наиболее близкое к принятому сообщению кодовое слово. К сожалению, как мы уже запомнили, число этих кодовых слов экспоненциально быстро растет с увеличением длины кода.

И тем не менее...

Обратим внимание на то, что нам ведь надо найти решение, совпадающее с оптимальным (что, конечно, очень трудно) или мало отличающееся от него, но при обязательно простом способе реализации самой процедуры поиска этого решения.

Так вот именно поэтому рассмотрим эту постановку задачи хорошего декодирования в несколько другой формулировке. Обозначим через X принятое из канала искаженное шумами сообщение, а произвольное кодовое слово из множества всех возможных слов выбранного кода – через A . Тогда вес их разности может быть представлен в виде $w = |B| = |X - \tilde{A}|$. Ясно, что согласно этому обозначению для двоичных сообщений мы должны найти поразрядную разность всех компонентов векторов X и \tilde{A} , а потом подсчитать вес этой разности, т.е. просто число единичек в результирующем векторе разности B . А цель оптимального декодирования – найти такое (может быть, единственное!) кодовое слово \tilde{A} , для которого $\min w = \tilde{w}_m = |\tilde{B}| = |X - \tilde{A}|$ среди всех допустимых кодовых слов данного кода.

Казалось бы, это та же самая формулировка оптимального решения. Да, конечно, это так и есть, но – с уточнением понятия близости, т.е. с уточнением метрических свойств двоичных со-

общений. И вот именно теперь для нас будет естественным, если мы обратимся к классическим методам вычислительной математики: поиск оптимального решения – это есть еще и классическая задача поиска глобального экстремума функции от очень большого числа переменных, которыми являются, конечно, n позиций, по которым вычисляются вектора в выражении для $\min w$! Иначе говоря, декодирование – это проблема оптимизации функционала от очень большого числа переменных. Конечно, рассматривая обычный ДСК, мы ищем экстремум функции w , веса Хемминга вектора B , определенного в дискретном пространстве. Эта постановка задачи легко обобщается и на квантованные решения «мягкого» модема и, конечно, на непрерывные решения модема о принятом векторе X при передаче данных по гауссовскому каналу. Правда, при этом проблема практически не упрощается, а даже несколько усложняется по сравнению с поиском оптимального (или почти оптимального) решения в простейшем ДСК.

Итак, что же такое новый подход к декодированию? Просто поиск глобального экстремума в дискретном пространстве?... А именно такой формулировки раньше не было? И, главное, что от этого изменяется?

На самом деле очень много, точнее, практически все! Просто раньше были алгоритмы с полным перебором, например, алгоритм Витерби, или чрезвычайно оригинальные методы решений уравнений в дискретных пространствах, которые или давали какое-то решение, или нет. Ничего такого, что можно было трактовать как постепенное движение в пространстве параметров с оценкой значения функционала, вообще не существовало. Значит, и потребности в развитии методов декодирования в стиле поиска экстремума сначала не было. Впервые такой подход был опубликован в 1974 г. в нашей стране, а через 20 лет, в 1993 г. после открытия турбо кодов это направление стало действительно широко исследоваться и на Западе.

Ладно, хорошо. Но как связать то, что мы сейчас обсудили, с тем, что мы все наше внимание уделяем мажоритарным методам?

Да дело просто в том, что первая попытка принять решение о принятых сообщениях не сразу, а в два этапа у нас только что уже состоялась. Так что мы на самом деле успешно применили новый многошаговый, итеративный подход к декодированию. Повторное декодирование, этот «эмпиризм на грани шарлатанства» долго, чуть ли не 10 лет после открытия самой возможности эффективного декодирования с использованием итеративных методов в 1972 г., не рассматривалось как равноправный (не говоря уж о том, чтобы – «как лучший») метод декодирования в каналах с большим уровнем шума. Когда-то это было очень тяжелым испытанием для математиков, плодотворно работавших в области алгебраической теории кодирования. Их чрезвычайно отрицательное отношение к идее повторных однородных вычислений, итераций в процедурах коррекции ошибок сильно мешало конструктивному развитию методов глобальной оптимизации в сфере построения алгоритмов декодирования. А вот прошлый раз мы очень наглядно убедились, что это действительно можно делать. Другой вопрос, что глубоко понять идущие при этом процессы очень и очень непросто. На это действительно можно потратить годы, что и потребовалось на самом деле. Но теперь в этом можно разобраться гораздо быстрее и затем двигаться вперед. Наши последователи пойдут гораздо дальше нас.

Давайте в связи с этим обратим внимание на одно важное обстоятельство, которое, наряду с возможностью постановки проблемы декодирования как глобальной оптимизации специального функционала, не было в принципе замечено буквально миллионами специалистов в процессе многих десятилетий интенсивного развития теории кодирования.

Для этого рассмотрим представленную ниже на рис. 10 схему очень простого кодирования сверточным кодом с $R=1/2$ и $d=3$. Мы уже понимаем, что этот код гарантированно исправляет $t_0=1$ ошибку в любой кодовой последовательности длины $n_A=4$. Как это будет иметь часто место и в дальнейшем, все основные соотношения, которые мы получим сегодня, будут справедливы и для блочных кодов, а также для двоичных каналов с независимыми ошибками.

Мы уже хорошо освоили начала теории кодирования, так что для вас уже будут вполне очевидны следующие утверждения.

Во-первых, для рассматриваемых нами только линейных кодов большинство рассуждений для произвольного кодового слова можно заменять анализом поведения декодера при передаче нулевого кодового слова. А почему это так? Для объяснения можно использовать и рис. 10.

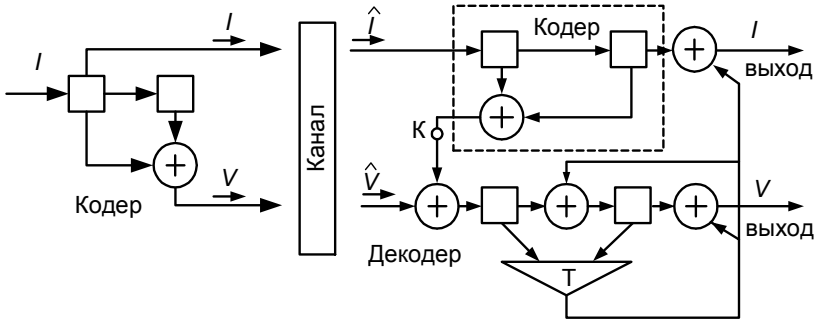


Рис. 10. Главное свойство вектора синдрома

При наличии свойства линейности сумма кодовых слов всегда есть снова кодовое слово. А при умножении любого кодового слова на проверочную матрицу H выбранного кода результат всегда равен нулю, точнее нулевому вектору. Отсюда следует, что поскольку пороговый элемент принимает решение по вектору синдрома принятого сообщения, а код линеен, то синдром для заданного кода есть функция только вектора ошибок. Мы уже немного обсуждали этот факт. Но тогда и поведение порогового декодера (и любого другого декодера для линейного кода) всегда будет определяться только воздействующим на него шумом канала.

А теперь давайте внимательнее посмотрим на правую часть схемы кодирования на этом рис. 10. На нем в декодере выделен обязательно присутствующий в нем кодер, именно такая же схема, как и та, которая формирует код на передающей части представленной модели системы связи. Именно его наличие

и обеспечивает возможность вычисления вектора синдрома. В самом деле, заметим, что если в канале не было ошибок, то проверочные символы V , приходящие из канала на первый вход полусумматора (сумматора по mod 2), точно совпадают с символами, которые формируются в точке К декодера такой же копией кодера, выделенной пунктирной рамкой. Но раз на входы полусумматора, формирующего вектор синдрома, все время поступают только одинаковые символы, то на его выходе будут только нули. Значит, мы еще раз на конкретной схеме увидели, что синдром кодового слова (т.е. когда ошибок канала нет) действительно равен нулю.

Но это пока только разминочные соображения. Вы уже чувствуете приближение к осознанию одного из самых ключевых моментов нового метода? Сейчас мы именно к нему и подошли. Обратим для этого внимание на то, что в точке К декодера всегда формируются проверочные символы того кодового слова, информационные символы которого поступили на вход выделенной пунктиром копии кодера в декодере. Значит, *синдром является разностью по проверочным символам* между принятым вектором и этим кодовым словом. И более того, на вход этой копии кодера можно подать вообще любую информационную последовательность, а не только тот искаженный в канале информационный вектор, который попал в декодер. Так что синдром всегда будет разностью по проверочным символам между принятым вектором и этим совершенно произвольно выбранным кодовым словом. Это очень важный момент рассуждений для понимания главных идей метода, который мы будем обсуждать далее.

Сделанное замечание было связано с тем, что мы хотим объяснить, т.е. понять вместе принципы действия итеративной схемы, которая будет работать с разными векторами синдрома. Простенький пример такой мажоритарной схемы с двумя итерациями мы уже обсуждали выше и увидели, что это имеет смысл. Хотим подчеркнуть, что и на первой, и на второй итерации в этом примере при каждом изменении декодируемых символов через обратную связь всегда изменяется и синдром принятого вектора. И поскольку мы четко видели на представленных ре-

зультатах моделирования, что вторая итерация даже в простейшем ПД весьма полезна, то изменение синдрома в процессе декодирования на самом деле очень важно. Так что сделанное замечание о произвольности выбора кодового вектора как раз и служит более гибкому и широкому пониманию роли вектора синдрома при декодировании.

Кстати, заметим, что ПД может работать и без изменения вектора синдрома через обратную связь (ОС). Такие декодеры называются дефинитными. В них немного меньше уровень размножения ошибок декодирования в том смысле, что группирование ошибок на выходе такой схемы несколько снижается. Ведь в этом случае оказывается, что пакеты ошибок веса $J=d-1$ при неправильных решениях ПЭ в регистр синдрома не проходят. Но это снижение пакетирования невелико. А вот повышение средней вероятности ошибки декодирования в такой схеме без ОС очень значительно. Оно может составлять и десятки раз.

А почему это происходит? Да просто все ошибки канала в информационных символах, которые раньше убирались через ОС с ПЭ, теперь остаются в регистре синдрома. Значит, их число удваивается! Это и ведет к росту вероятности ошибки дефинитного декодера.

Заметим, что если вероятность ошибки в канале p_0 небольшая и декодер во много раз снижает плотность ошибок на своем выходе по сравнению с тем, какова вероятность ошибок в ДСК, то роль ОС очень положительна. Ведь в этом случае в регистре синдрома практически нет ошибок декодера. А если ОС убрать, то все ошибки канала в синдроме остаются. Значит, качество проверок падает в этом случае из-за того, что все проверки имеют максимально возможные размерности (а это, как известно, – число входящих в проверочные уравнения ошибок). Это и есть другая формулировка причины роста средней вероятности ошибки в дефинитном декодере. Но наша-то цель всегда – уменьшить именно вероятность ошибки. Значит, убирать ОС в обычных схемах кодирования, ориентированных на минимизацию средней вероятности ошибки, не следует. Так что и этот

аспект обсуждаемого вопроса о значении и глубинном, почти философском смысле вектора синдрома очень важен.

Возвращаясь непосредственно к обсуждаемому рисунку, отметим, что основную идею итеративного декодирования как задачи поиска экстремума мы с вами уже фактически нашли. Ведь ОД ищет минимум полной разности между принятым вектором и всеми кодовыми словами. А синдром – давайте, повторим эту принципиальную мысль – является разностью между принятым вектором и некоторым кодовым словом только по проверочным символам. Значит, осталось только научиться вычислять разность этих векторов еще и по информационным символам. Тогда будет получена полная разность между этими векторами, что и позволит измерять уже полное расстояние между ними. Правда, после решения этого вопроса еще потребуется найти эффективную процедуру, которая бы была действительно итеративной и обеспечивала целенаправленное движение к решению ОД, как это и делается всегда при многошаговых процедурах оптимизации функционала.

Ну, а какие можно предложить идеи по подсчету разности векторов в информационных символах? Поскольку нужна просто разность, а не какая-либо сложно вычисляемая функция, то надо каждый раз просто прямо вычислять разность между этими векторами: принятым из канала и некоторой возможной гипотезой-решением. Правда, так вот сразу сначала непонятно, как этим потом воспользоваться. Ведь мы же все равно не хотим просто перебирать все 2^k возможных кодовых векторов-решений.

Для того чтобы аккуратно рассмотреть возможности организации эффективной итеративной процедуры, предлагаем воспользоваться тем, с чем мы только что согласились, и что уже почти выучили: синдром есть разность векторов по проверочным символам.

Итак, возьмем некоторый произвольный информационный вектор A_1 , который, как мы договорились, может быть и вектором, принятым из канала, как показано на рис. 10. Для него можно посчитать вес его полной разности с принятым вектором $B_1=A_1-X$, и пусть он будет $w_1=|B_1|$. И давайте теперь учтем

(решающий момент!), что этот вес будет определяться с учетом разности векторов A_1 и X по информационным символам, которую мы учтем в новом разностном векторе D , который и сможет обеспечить запоминание полной разности между этими векторами.

Ну, а сейчас самый важный шаг рассуждений. Возьмем теперь такой новый вектор A_2 , который отличается от A_1 только в некотором единственном информационном символе i_j . Вычислим для этого вектора вес w_2 его разности с принятым вектором: $w_2 = |B_2| = |A_2 - X|$. Как соотносятся веса w_1 и w_2 этих векторов? Поскольку анализируемый нами код линейен, то эти вектора отличаются между собой в тех и только в тех J проверочных символах и обязательно в соответствующем символе d_j вектора D , которые все связаны с выбранным нами символом i_j . Тогда разница в значениях w_1 и w_2 определяется именно значениями в тех проверочных символах кодовых векторов A_1 и A_2 , в которых есть i_j , а также символами d_j . Вычислив далее синдромы S_1 и S_2 и разностные векторы D_1 и D_2 приходим к заключению, что если посмотреть на эти различные для обоих векторов A_1 и A_2 символы, то все они попарно разные и их общее число есть именно $J+1$. Если теперь поставить только на эти различающиеся между собой символы синдрома (а ведь это обычные проверки мажоритарно декодируемого кода!) и символ d_j , пороговые элементы, то получим 2 мажоритарных декодера и сумма этих проверок для обоих вариантов векторов A_i будет равна на двух ПЭ точно $J+1$. Отсюда получаем, что тот вектор, например A_1 , для которого сумма на пороговом элементе окажется меньше, чем у второго, A_2 , будет соответствовать и меньшему весу $w_1 < (J+1)/2$, причем сумма проверок на втором ПЭ будет соответственно не менее $(J+1)/2$. Значит, если мы имеем 2 одинаковых декодера с теми небольшими модификациями, которые мы только что обсудили, то выбрав значение порога в ПЭ обоих декодеров, равным $(J+1)/2$, получим, что ПЭ в первом декодере ничего не изменит, а второй ПЭ, который соответствует другому значению i_j , инвертирует этот символ i_j так, что его решение опять же совпадет с решением первого декодера.

9. Новый супердекодер?

Рассмотренный выше очень простой пример дает искомое решение той сложнейшей задачи, которую мы поставили перед собой. Фактически мы должны взять классический мажоритарный декодер и добавить к нему новый разностный регистр D , в котором надо отмечать изменения, которые будут делаться самим же декодером в информационных символах в процессе декодирования. Естественно, что сначала вектор D полностью нулевой. Это просто следствие того, что у нас при получении из канала искаженного шумами вектора X сначала нет совершенно никаких особых гипотез об истинном векторе, который был послан передатчиком. Новый момент по сравнению с классическим пороговым декодированием состоит только в том, что на пороговый элемент поступает еще одна проверка с регистра D , а решения ПЭ всегда дополнительно изменяют еще и соответствующую ячейку в векторе D , которая относится к декодируемому символу i_j . Это решение порогового элемента учитывается далее на всех следующих итерациях декодирования каждого символа, так как с регистра D декодера на ПЭ теперь поступает еще одна, $(J+1)$ -я проверка.

А как это связано с задачей поиска глобального экстремума? Да совсем просто! Если сумма проверок на пороговом элементе нашего нового мажоритарного алгоритма не превышает половины от их общего числа $d=J+1$, то это значит, что наш алгоритм не нашел более правдоподобного решения, чем то, которое содержится в информационном регистре декодера. А если сумма на пороге превысила половину от общего числа проверок, то декодируемый символ i_j инвертируется и новое решение декодера с инвертированным символом i_j более близко к принятому вектору X и, следовательно, более правдоподобно, чем его предыдущее решение с неисправленным символом i_j .

Но тогда отсюда следует, что, прогоняя все символы много раз через пороговые элементы, получаем, что такой метод позволяет все время приближаться к решению ОД. В этом и есть суть новой мощной процедуры декодирования. Она называется многопороговым декодированием (МПД) и, судя по текущим

публикациям, относится к наиболее простым и одновременно самым эффективным алгоритмам коррекции ошибок в каналах с большим уровнем шума.

Схема МПД в сверточном варианте реализации представлена на рис. 11.

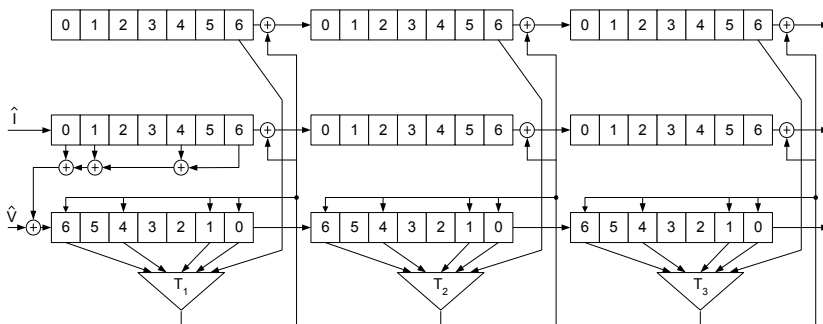


Рис. 11. Многопороговый декодер сверточного кода с $I=3$ итерациями

Такая схема сверточного декодера может быть отнесена к одним из наиболее быстрых декодирующих устройств. Как следует из вида МПД, эта схема имеет быстродействие, которое определяется двумя факторами: скоростью продвижения данных по регистрам сдвига, а также быстродействием пороговых элементов, которые должны суммировать проверки, сравнивать их с пороговым значением, а затем выдавать сигнал на изменение декодируемых символов. Что же здесь наиболее существенно? Да то, что самыми быстродействующими элементами в МПД являются, конечно, регистры сдвига. Разница в быстродействии с пороговыми элементами может быть весьма значительной и достигать одного и более десятичных порядка, т.е. 10–30 раз. Но к сложности реализации МПД мы еще вернемся. Мы обсудим эту проблему и задачу минимизации объема вычислений немного попозже. Там мы также увидим некоторые приятные и полезные для алгоритма МПД свойства и эффекты, несколько не-

обычные для специалистов в области проектирования процедур декодирования.

На рис. 12 представлена аналогичная схема МПД декодера для блочного кода. Как и полагается при декодировании блочных кодов, МПД последовательно, а не параллельно, как сверточный его аналог, выполняет обычные для порогового алгоритма функции: прием символов кода, вычисление синдрома, суммирование проверок и коррекцию символов. Естественно, что исходное состояние нового разностного регистра перед началом собственно коррекции ошибок в блочном МПД также равно 0. А затем, как и было описано выше, при декодировании происходит заполнение этого регистра отметками порогового элемента об измененных на тех или иных итерациях символах кода. Разумеется, в точном соответствии с описанным алгоритмом МПД на ПЭ также подаются все эти метки на всех итерациях декодирования и в сверточном, и в блочном вариантах реализации процедуры.

Может показаться, что произошедшие изменения в алгоритме не столь значительны. Ну, в самом деле, разве добавление к пороговому элементу одной проверки, например, 7-й или

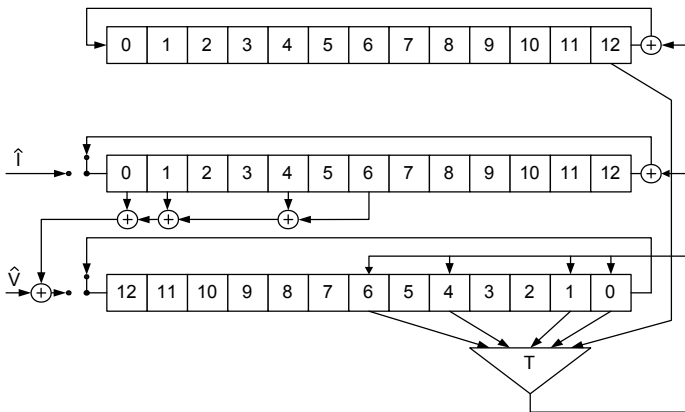


Рис. 12. МПД для блочного кода с $R=1/2$ и $d=5$

15-й, действительно что-то меняет принципиально? Ведь это, в общем-то, немного.

В это трудно поверить? Да, трудно. Именно это и продемонстрировали те, кто отрицал вообще что-то принципиально важное и новое в этом методе. На самом деле с формально-теоретической точки зрения мы получаем совершенно другой алгоритм, который теперь ищет строго более правдоподобные кодовые вектора по сравнению с теми решениями, которые находятся в декодере. А раньше ПД учитывал только часть тех векторов, которые надо было сравнивать, чтобы выбрать более правильное решение с меньшим числом ошибок. Вопрос о мере правдоподобия решений обычного ПД и их соотношение с решениями ОД также никогда и в принципе не ставился.

Еще раз подчеркнем, что каждое новое решение МПД соответствует новому решению декодера A' , более близкому к принятому вектору X по сравнению с предыдущей его версией A_0 до изменения декодируемого символа. Но это и значит, что мы нашли очень простой способ поиска глобального экстремума в процедуре декодирования: взять как бы обычный ПД, но учитывать в новом разностном регистре D решения порогового элемента, которые в свою очередь будут учитываться и на следующих этапах декодирования при подсчете сумм на пороге. И каждый раз при изменениях символов гарантируется переход к более правдоподобному решению, поскольку суммарный вес синдрома S и разностного вектора D только строго уменьшается. Много лет назад для нас это было очень приятно и еще более неожиданно. Теперь все это автоматически подракумеваается во всех дальнейших модификациях и обобщениях этого алгоритма. Надеюсь, что вам все это будет проще воспринять, чем нашим коллегам 20 лет назад, чтобы затем успешно двигаться дальше.

Предлагаемый метод назван алгоритмом многопорогового декодирования (МПД), поскольку при декодировании каждого символа он проходит много раз мимо одного или нескольких пороговых элементов, если это блочный код, или мимо цепочки из многих пороговых элементов, если это сверточный код.

Подчеркнем также, что даже сейчас, скорее всего, во всем мире не существует даже весьма и весьма сложных алгоритмов, которые обладали бы безусловным свойством приближения к оптимальному решению при каждом изменении декодируемых символов. А вот исключительно простой МПД обладает именно этим свойством без каких-либо специальных дополнительных предположений о его свойствах, причем он им обладает совершенно естественно, потому что изначально был создан исходя именно из этой цели: решения задачи глобальной оптимизации. Тем не менее, открытие такого удивительного в очень многих отношениях метода является действительно подарком природы и, в каком-то смысле, судьбы. Поэтому максимальное использование возможностей этого нового алгоритма, несомненно, является достойной и очень интересной задачей для исследователей в области прикладных вопросов теории кодирования.

Возможно, что с последним замечанием об уникальности свойства сходимости решений МПД к решению ОД кому-то будет очень трудно согласиться, потому что оно будет восприниматься как несколько неточное. В качестве «опровергающего примера» такого рода авторам однажды пришлось услышать, что, мол, как раз алгоритм Витерби и является таким методом, который обеспечивает сходимость к решению ОД. Но это несколько неточное утверждение. На самом деле АВ является просто оптимальным методом для любого кода, осуществляющим полный перебор решений и выделяющим из всех возможных своих решений именно наиболее правдоподобное. Просто АВ осуществляет свой поиск на специальных сверточных кодовых структурах, что позволяет реализовать процедуру оптимального декодирования при существенно меньших вычислительных затратах, чем это было бы абсолютно необходимо при использовании обычных блоковых кодов. Но сам факт полного перебора в АВ совершенно безусловен. А это и значит, что он не осуществляет никаких процедур итеративного типа, которые бы постепенно выводили декодер на все более правдоподобные решения. Так что абсолютная уникальность МПД в этом смысле совершенно очевидна.

Укажем в связи с этим и на то, что совершенно замечательные по своим возможностям широко известные турбо коды так же, как и МПД, относятся к мощным итеративным процедурам с достаточно высокой эффективностью и вполне конструктивным алгоритмом. Однако для этих кодов можно говорить о сходимости к решению ОД только в предположении об очень высоком качестве используемого в этих кодах перемежителя. Очень важным при сопоставлении методов декодирования является также и то, что часть операций, выполняемых в турбо декодерах, оказывается, безусловно, слишком сложной. И абсолютное число операций в турбо декодерах также очень велико. Но эти важнейшие аспекты сопоставления алгоритмов мы будем обсуждать в специальном разделе о сложности декодеров.

А пока вернемся непосредственно к МПД.

Вот, например, почему МПД изменяет в самоортогональных кодах по одному информационному символу? Если мы ставим и решаем задачу многократного итеративного улучшения решения, то почему переходы от одних решений к более правдоподобным другим должны происходить при изменениях в этих решениях только по одному символу?

Ну, тут полезно сделать сначала общее замечание. Как видим, по сравнению с обычным ПД, схема МПД декодирования изменилась крайне незначительно, но стала обеспечивать стремление решений МПД к решению ОД. Однако обычно решение ОД – это результат полного перебора, т.е. просмотра экспоненциально большого числа возможных кодовых слов. Но МПД делает это с линейной от длины кода сложностью. (Предлагаем все же сложностью декодеров заняться как-нибудь в другой раз. Это непростая и очень интересная тема.)

А теперь – ответ на поставленный вопрос. Есть только одна серьезная причина для изменения символов по одному: ближайшие кодовые слова в СОК находятся на расстоянии $d=J+1$ друг от друга. И в этих соседних кодовых словах информационные части отличаются всегда только в одном символе. Очевидно, что под воздействием шума наиболее часто оптимальным

декодером будет происходить замена переданного кодового слова именно на наиболее близкое к нему решение.

Но это совсем не единственный способ перехода от одного кодового слова к другому. На самом деле всегда можно посмотреть, в каких множествах проверок отличаются кодовые слова, которые имеют, например, какие-то m различных информационных символов. Тогда берем все m проверок от регистра D и все соответствующие этим символам n проверок из синдрома. Конечно, эти n символов каждый раз надо аккуратно выделить и посчитать. Затем настраиваем пороговые элементы на половину от общего числа проверок $(n+m)$ и изменяем каждым пороговым элементом декодируемые символы по несколько информационных битов сразу. Проблема в этом совсем нетривиальном случае состоит только в том, на какие группы символов надо настраиваться и нельзя ли минимизировать число таких групп символов в тех случаях, когда эта обобщенная процедура вообще имеет хоть какой-то смысл. Такая чисто технологическая задача для многих, хотя и не для всех полезных случаев тоже решена. Но проблемы декодирования постоянно усложняются. Так что мы все время решаем эту и другие подобные задачи роста эффективности МПД в очень необычных, дополнительных или специальных условиях коррекции ошибок.

Какие еще могут быть вопросы о принципах работы МПД? Ведь это был самый сложный в идеологическом плане материал. Да, наверное, и в психологическом тоже.

Ну, вот, например, нам неоднократно приходилось отвечать на вопрос о том, почему и на последующих итерациях, которые реализуются на других пороговых элементах, например, в сверточных кодах, МПД продолжает приближаться к решению ОД?

Тут все относительно несложно. Если вам легко согласиться с тем, что на первой итерации декодирования происходит переход к новому более правдоподобному решению, то это значит, что вы уже поняли многократно обсуждавшийся выше факт: и до, и после изменения декодируемого символа векторы синдрома S и разностный D являются разностью принятого и двух сравниваемых векторов, отличающихся в одном декодируемом символе i_j . Пороговый элемент на любой позиции сравнивает

эту пару решений и выбирает более близкое к вектору X свое новое промежуточное (или окончательное!) решение. Пусть вам, уважаемый читатель, действительно понятно из нашего обсуждения, почему первый ПЭ уменьшает расстояние между векторами A и X при коррекции символов. В самом деле, ведь в новом декодере содержится разность векторов текущих гипотез и вектора X . Поэтому когда после декодирования, например, символа i_0 сначала не меняются несколько следующих символов, а затем происходит коррекция по тому же алгоритму, скажем, символа i_7 , можно утверждать, что МПД снова улучшил свое решение, т.е. он нашел еще более правдоподобное кодовое слово.

Ну, а теперь обратим внимание: выбор следующего символа, который надо декодировать, на самом деле абсолютно произвольный. Мы можем после первого символа попытаться декодировать 17-й. А затем 4-й, а далее 145-й, если такой есть. Так что наличие итераций – условность, согласованный порядок выбора того символа, который мы хотим декодировать на следующем шаге. Здесь важно только то, что сохранение правильных разностей между принятым вектором и текущей гипотезой-решением декодера после каждой коррекции символов заложено в самом алгоритме. Конечно, нужно пытаться корректировать все символы одинаковое число раз. Если один символ мы пытались декодировать 2 раза, а какой-то другой – 60, то мы можем быть почти уверенными, что до оптимального решения мы не доберемся. Вот понятие итерации и позволяет проверять одинаковое число раз все символы на необходимость их коррекции. Это просто очень удобная в технологическом аспекте условность. Вот и все.

Так что пытаться исправлять символы можно в любом порядке: в каждый момент в МПД содержится расстояние между своей текущей гипотезой о решении и принятым вектором. А пороговый элемент можно, конечно же, очень просто настраивать на различные группы одновременно корректируемых символов.

Надеемся также, что для большинства читателей МПД – просто еще один метод, не ломающий ваших прежних, слишком устоявшихся взглядов на проблему эффективного декодирования.

ния. Значит, вы воспримете его значительно легче и быстрее. Именно это в свою очередь, возможно, позволит как раз вам продвинуться в ближайшее же время далеко вперед в деле улучшения характеристик и одновременного дальнейшего упрощения алгоритмов декодирования, может быть, и алгоритма МПД. Хотя, как может показаться некоторым, куда уж проще и лучше...

В общем, дерзайте!

Справка. Строгое доказательство свойств синдрома и разностного регистра, а также сходимости к оптимальному решению всей последовательности улучшающихся решений МПД можно найти на сайте ИКИ РАН www.mtdbest.iki.rssi.ru, где представлены очень многие материалы, относящиеся к алгоритмам МПД. Вы, возможно, очень удивитесь, как компактно и просто там изложены долго обсуждавшиеся нами вопросы.

Есть ли еще какие-то проблемы по схемным реализациям МПД?

Да. Например, однажды нам задали и такой вопрос: почему в алгоритме МПД не оговаривалась необходимость использования самоортогональных кодов (СОК)? Ведь все предварительные эксперименты с двумя итерациями коррекции ошибок и просто сами алгоритмы ПД, вроде бы, исследовались при моделировании именно на этих кодах. Почему же при рассмотрении свойств МПД нигде не обсуждалось в явном виде, что в МПД должны быть использованы коды именно этого класса?

Этот, на первый взгляд, простой вопрос, тоже требует небольших уточнений. Если еще раз прочесть предыдущий раздел о том, из каких рассуждений появляется схема МПД, то вопрос о кодах для МПД действительно может показаться справедливым. Там на самом деле не было ничего такого, что требовало бы использования предположения о необходимости применения в МПД только СОК. Это значит, что мы можем прямо сейчас существенно расширить сферу применения принципов МПД. Дело в том, что свойство приближения решения МПД к решению ОД полностью справедливо вообще для всего класса линейных кодов. И все, что доказывается строго для алгоритмов типа МПД, обычно одинаково справедливо и для сверточных,

и для блоковых кодов. Требуется только наличие свойства линейности. А все, что связано с особыми свойствами каких-либо кодов, в том числе относящихся и к классу СОК, – все это должно аккуратно анализироваться совершенно отдельно. Без каких-либо серьезных изменений МПД применяется и для недвоичных кодов в соответствующих каналах симметричного типа. Следует подчеркнуть, что общим критерием для такого анализа методов декодирования и выбора кодов для этих алгоритмов должно быть только качество и скорость сходимости решения МПД к решению ОД.

10. Так он совсем не оптимальный?

«Что это за странная проблема», – спросите вы. – Что значит «качество сходимости»?

Но она не странная, а главная.

Еще не ясно? Ну, давайте посмотрим. На каждом шаге коррекции МПД гарантированно приближается к оптимальному решению. Чего же боле, как сказал поэт. Но...

И все же, что мы получили на самом деле в виде алгоритма МПД? Можно ли утверждать, что МПД действительно обязательно достигнет решения ОД? Похоже, что с абсолютной неизбежностью каждый стремящийся к объективности читатель постепенно (или сразу!) должен прийти к единственному разумному мнению, которое с теми или иными нюансами будет следующим: МПД, ну, никак не может быть оптимальным алгоритмом. Чтобы такая простенькая схема всегда могла привести к глобальному экстремуму очень непростой функционал в весьма специальном дискретном пространстве? Нет. Не похоже...

Но почему же? Да именно потому, что он чрезвычайно прост. А поэтому он даже с чисто философских позиций никак не может быть оптимальным. Иначе как же это могло бы оказаться, что другие методы с таким трудом, выполняя порой исключительно сложные вычисления, очень и очень медленно улучшают свои характеристики по мере развития технологии

и теоретических исследований. А МПД вот так абсолютно легко и просто всегда достигает решения ОД?

Если вас после этих вопросов и откровенных сомнений посетило чувство усилившего дискомфорта, то это просто замечательно! В самом деле, было бы чрезвычайно странно, если бы все другие методы мучительно трудно приближались к пропускной способности канала по Шеннону, а наш алгоритм самого простейшего мажоритарного типа без проблем достигал бы оптимального решения, соответствующего глобальному экстремуму весьма и весьма сложной функции в пространстве параметров очень большой размерности. А коды в этих пространствах имеют, как хорошо известно, еще и особенно замысловатую структуру. Так что в теории и алгоритмах МПД должны быть особенные и очень серьезные принципиальные препятствия, ограничивающие возможности простого непосредственного применения этого метода. Иначе и результативность других методов должна была бы быть гораздо выше при существенно меньших затратах, чем это практически всегда оказывается сейчас. Наверное, каждый может увидеть это из тех публикаций по методам декодирования, которые постоянно можно найти в ведущих журналах и в Интернете. Буквально для всех алгоритмов декодирования повышение эффективности на пару десятых долей децибела – огромный прогресс. Так что просто исходя из принципов единства мира, МПД должен встретить на своем пути продвижения к эффективному декодированию в больших шумах очень большие, может быть, почти столь же значительные трудности в повышении эффективности, как и другие алгоритмы. А если этих трудностей не видно, то именно на объяснение их отсутствия должны быть направлены очень значительные силы исследователей. Это абсолютно обязательно, чтобы не произошло каких-либо незапланированных неожиданностей в момент первых же попыток применения алгоритма МПД на практике, мол, «ах, как же так мы тут ошиблись!». Вот обо всех основных трудностях применения и повышения эффективности МПД и следует поговорить. Эти проблемы как раз и составляют основные направления работ последних десятилетий. Наверное, эти же задачи

придется в той или в иной степени решать и вам, тем, кто будет продолжать наши работы.

Но, может быть, вы тоже можете указать на основные, с вашей точки зрения, явления, которые легко могут «затормозить» быстрое продвижение МПД к пьедесталу оптимальности и заставить его тоже упорно бороться за место под солнцем вместе с остальными методами декодирования?

В самом деле, давайте вспомним, что было сказано о свойствах повторного декодирования на базе обычного ПД: третья итерация, похоже, была малоэффективна. Так, наверное, что-то действительно не дает МПД возможности продолжить процесс успешного улучшения своих решений до конца? На первый взгляд кажется, что если декодер начал улучшать свои решения, то потом, когда, например, после третьей или двадцатой итерации коррекции число оставшихся ошибок стало существенно меньшим, чем их было перед началом процедуры декодирования, то совершенно непонятно, что же именно может остановить этот уже почти завершившийся успешный процесс.

Более внимательный анализ даже простейших рассмотренных выше экспериментов по моделированию работы ПД может помочь понять суть проблемы. Обратимся к графику первой и второй итерации на рис. 9. Анализ этих результатов декодирования показал, что одиночных ошибок после второй итерации там уже совсем не было! И не было даже двух и четырех ошибок. А вот ошибки веса 10 и больше были! Хотя их относительное количество при длине кода равной 1000 битов, невелико, что-то не позволяло ПД совсем «подмести» эти немногие оставшиеся искажения символов в канале! Так что же, наверное, проблема – в самой структуре используемых кодов? Но, может быть, это явление тоже можно проанализировать и тогда появится возможность строить коды, которые не будут «тормозить» работу МПД? Действительно, оказывается, что многие комбинации из небольшого числа ошибок могут не исправляться с помощью и обычного ПД, и нового МПД. Так что мы постепенно приближаемся к тому, чтобы определить новые аспекты исследований МПД.

11. Но какие характеристики – самые лучшие?

А кстати, может быть, то, что мы наблюдали при моделировании – это и есть границы для реальной корректирующей способности используемых кодов? Какие вообще можно указать хорошие границы для корректирующей способности МПД? Оказывается, что как раз для самоортогональных кодов (СОК) можно построить очень простые и удобные оценки. Возьмем произвольный СОК с нечетным (всего лишь для удобства вычислений) кодовым расстоянием d . В нем просто из правила его построения все кодовые слова веса d имеют единственную информационную единичку. Значит, в обычном ДСК при небольшом уровне шума наиболее часто из-за ошибок канала переданное кодовое слово перейдет в одно из таких ближайших слов веса d . И тогда понятно, что это может произойти, если более $d/2$ таких ошибок канала попадут именно в какие-то из d позиций этих кодовых слов с одной информационной единичкой. Нижняя оценка вероятности такого события – просто обычное биномиальное распределение:

$$P_b(e) = \sum_{i=(d+1)/2}^d C_d^i p_0^i (1-p_0)^{(d-i)}.$$

Почему тут сразу пишутся вероятности ошибки в виде вероятности ошибки на бит? Просто потому, что тут рассматривается вероятность перехода под действием шума нулевого кодового слова в конкретное ненулевое кодовое слово с «1» на определенной заданной позиции. Все другие кодовые слова с единичкой в этой же позиции имеют вес, гораздо больший, чем d . Значит, предложенная оценка при малом шуме достаточно точно определяет вероятность ошибки именно в этом информационном бите. Как хорошо известно, в случае применения многих методов в гауссовском канале, например, алгоритма Витерби (АВ), их характеристики улучшаются на 2 дБ. Аналогичными возможностями, конечно, при относительно малом шуме характеризуются и СОК, и МПД. На следующем графике представлены нижние оценки вероятностей ошибки на бит для рассматриваемых нам СОК в обычном ДСК для $d=5, 9$ и 13 с префиксом h ,

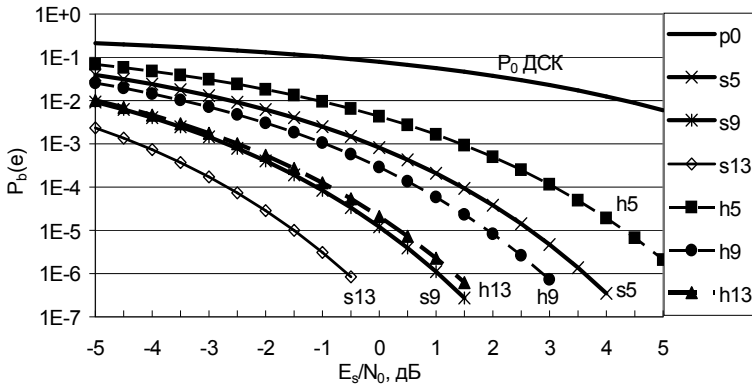


Рис. 13. Эффективность оптимального декодирования СОК

а для гауссовского канала – с префиксом s , в обоих случаях без учета кодовой скорости и для обычной двоичной фазовой манипуляции (ФМ). Это значит, что при любой кодовой скорости R при заданных d представленные оценки окажутся нижними для любых конкретных кодов.

Как следует из их вида, характеристики кодов в ДСК и гауссовском канале действительно различаются на 2 дБ по отношению сигнал/шум на входе ОД. Заметим, что для получения достаточно высоких характеристик на базе СОК, например, для обеспечения вероятности ошибки на бит $P_b(e) \approx 10^{-5}$, нет необходимости создавать декодеры для кодов с очень большими значениями d . По крайней мере, в некоторых случаях значение кодового расстояния $d \approx 15$ или даже менее вполне достаточно.

Но возникает вопрос о том, каковы же реальные возможности МПД? Может быть, характеристики этого алгоритма окажутся такими высокими, что все вопросы будут сразу сняты?

12. Так какой же он, этот МПД?

Рассмотрим, наконец, экспериментальные данные по поводу сравнительных возможностей ПД и МПД. Это будет, несомненно, важное обсуждение, но вот обрести чувство удовлетворения в результате их обсуждения, возможно, и не удастся. На рис. 14 показаны кривые зависимости вероятности ошибки декодирования на блок для ПД, а также для МПД с тремя итерациями декодирования при $R=1/2$ и $d=11$. Как главный итог сопоставления алгоритмов для кодов типа СОК длины, в данном случае $n=182$, состоит в том, что некоторый прогресс при переходе от ПД к МПД наблюдается, но только при весьма малом уровне шума, когда уже и многие другие алгоритмы оказываются весьма эффективными. Более того, анализ результатов моделирования показал, что на самом деле именно сильное группирование ошибок декодирования крайне усложняет работу повторных ступеней МПД, не позволяя ему сразу проявить свои, казалось бы, выдающиеся способности.

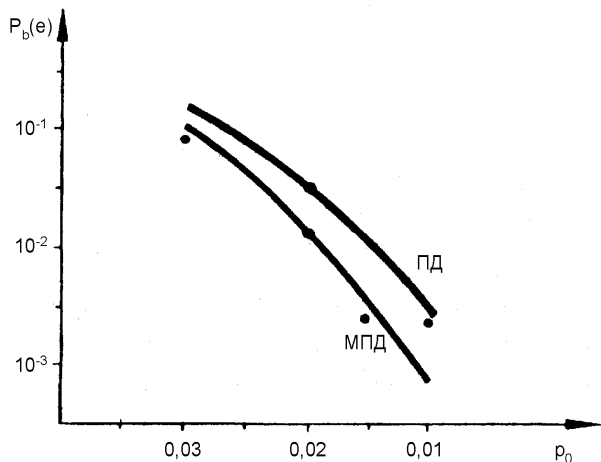


Рис. 14. Характеристики ПД и МПД для блокового СОК с $d=11$ и $n=192$

Но нам туда, в малые шумы, не надо. Это не интересная область и для техники, а тем более для теории. Последней соломинкой, спасающей идею применения МПД, могло бы быть предположение о более успешном декодировании с его помощью при очень большом числе итераций, превышающем, например, 50.

Но такого эксперимента даже не требуется проводить, потому что уже при трех итерациях процесс улучшения решений МПД для кодов на приведенных выше графиках, фактически заканчивается в больших шумах почти для всех типичных принятых кодовых блоков. В отдельных случаях в некоторых кодовых блоках процесс продолжался и на пятой итерации. Но исправлялись еще одна–две ошибки, что уже ничего не решало в принципиальном плане.

13. Как спасти идею?

Мы уже говорили, что в теории кодирования множество неожиданностей обрушилось на исследователей с самого начала. Одна граница сферической упаковки, полученная через почти несуществующие совершенные коды, дающая, тем не менее весьма важные оценки характеристик, стоит очень и очень много. Но это лишь простой пример. А что если наши «самые лучшие коды» (мы имеем в виду СОК) встряхнуть и посмотреть, насколько же они на самом деле хороши.

Тут, правда, нужно сказать, что, например, на последнем рис. 14 были представлены характеристики действительно лучших СОК по отношению d/n для скорости $R=1/2$. Поэтому, вообще говоря, не совсем ясно, что же именно можно сделать с самоортогональными кодами (или с какими-нибудь другими), чтобы в рамках несомненно заманчивых, вроде бы, возможностей МПД попытаться все же реализовать потенциальные возможности этих кодов.

Чтобы не задерживаться слишком на очень сложной проблеме, которую мы будем детально обсуждать чуть позже, можем себе сразу сами и покажем, что действительно хорошие и совершенно простые решения есть. Оглядываясь назад в исто-

рию развития МПД, скажем, что некоторые пути решения задачи повышения эффективности МПД появились тоже почти сразу. Другое дело, что открывшиеся проблемы решались слишком медленно. Но это уже и есть нормальная напряженная работа на правильно выбранном направлении.

Наверное, это очень забавно, но одной из первых относительно успешных попыток оправдать хотя бы частично применение МПД, было использование ножниц. Да. И, однако, это очень эффективный простой метод. Представим, что взят некоторый код с $d=15$, который нужно приспособить к алгоритму типа МПД. В этом конкретном коде, взятом из таблиц, которые есть во многих книгах по кодированию, в проверочные уравнения входят все 183 информационных символа кода длины $n=366$ с $R=1/2$. Как мы уже немного говорили, это способствует сильному проявлению зависимости решений МПД о достаточно близких символах. Если теперь представить картинки кодера и декодера для этого гораздо более длинного кода, чем наш простенький образец на рис. 8 для $d=5$, то разница будет состоять только в длине регистров устройств и в количестве связей между ячейками регистров. Пороговый элемент будет иметь 15 входов, и регистр синдрома окажется формирующимся с помощью 14 полусумматоров, создающих для $R=1/2$ все 14 систем проверочных уравнений с 15 слагаемыми в каждом из них.

Для того чтобы СОК с уравнениями, содержащими не более чем по одной возможной ошибке относительно любого декодируемого символа, превратить в СОК с существенно меньшим значением d , как раз и нужны «ножницы», отстригающие часть входов полусумматора в кодере и проверок в декодере.

Если выбрать, например, $d=9$, то из 14 ячеек в кодере (и соответственно в декодере), с которых подаются слагаемые для формирования проверочных символов, остаются только 8. Соответствующие операции проводятся, конечно, и с пороговым элементом, который должен уже иметь только 9 входов. Что при этом изменилось? Ясно, что декодер по-прежнему соответствует коду класса СОК, так как всего лишь уменьшилось число проверочных уравнений, а в тех, которые остались, число слагаемых

также сократилось, и в них есть только часть из прежних ошибок, номера у которых, естественно, остались различными.

Но главным моментом в этом преобразовании оказывается то, что общее число символов в проверках уменьшилось так, что у соседних символов наборы стали иметь существенно меньше общих информационных ошибок. Этот подход уже немного обсуждался в начале нашего изложения идей относительно группирования ошибок. В рассматриваемом примере при выборе обычного кода с $d=15$ оказывается, что доля общих ошибок в проверках относительно двух произвольных символов для кода длины $n=366$ равна 100%, а при построении кода с $d=9$ из кода с $d=15$ указанным «хирургическим» путем можно получить, что доля пересекающихся символов в различных проверках не превышает 0,3. Разница более чем существенная. А зачем все это? Только затем, чтобы получить меньшую степень группирования ошибок в ПД. Только в этом случае у нас появляется шанс на реальное внедрение метода МПД. Кроме того, более точными оказываются и нижние оценки помехоустойчивости, которые уже не будут противоречить границам сферической упаковки. Последнее замечание относится к тому, что нижние оценки $P_b(e)$ на рис. 13 при повышении уровня шума и больших значениях R перестают быть хорошими, т.е. мало отличающимися от истинных. Эти оценки, если вспомнить способ их получения, не зависели от длины кодов и от кодовых скоростей R . А мы уже видели, что это важнейшие параметры кодовой эффективности.

Но вернемся к нашему «усеченному» коду и его характеристикам. На рис. 15 представлены экспериментальные графики вероятности ошибки декодирования блокового кода с $d=9$ и $R=1/2$ для $n=114$, а также для $n=366$.

Во втором случае при $n=366$ код был действительно получен из СОК с $d=15$ путем отсекаания части проверок описанным выше способом, ножницами. Как мы видим, разница с простым ПД в этом случае применения пяти итераций в МПД уже есть. У традиционного ПД характеристики по выходной вероятности ошибки оказываются более чем на порядок слабее. Но очень важно и то, что более длинный код при равных $d=9$ по меньшей

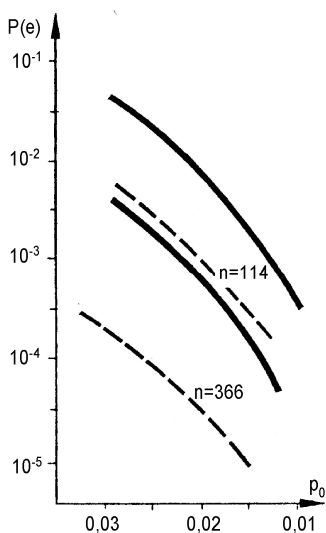


Рис. 15. Характеристики МПД декодера для короткого и длинного СОК с $R=1/2$ и $d=9$

мере еще на порядок более эффективен, чем короткий. Пунктирная кривая на этом рисунке дана для средней вероятности ошибки МПД на бит, а сплошная, как и раньше, на блок.

Важно, что по критерию d/n коды, «удобные» для МПД, с $n=366$, оказываются совсем другими, гораздо более слабыми, чем наиболее «упакованные», например, с $n=114$. Происшедшее смещение понятия «лучшего» кода не должно нас особенно беспокоить, так как обычно нет или очень мало таких алгоритмов декодирования, которые обеспечивали бы улучшение характеристик декодирования просто за счет каких-то модификаций самого алгоритма. На самом деле, конечно, под каждый алгоритм всегда приходится подбирать наиболее соответствующие ему коды. И только потом уже эти алгоритмы сравниваются между собой по более общим критериям, например, по достоверности, быстродействию, числу операций, однородности и т.д. Так и с кодами для МПД. Поставив перед собой чрезвычайно трудную задачу существенного повышения эффективности де-

кодирования самыми простыми методами, нужно по меньшей мере не упускать возможности подобрать под алгоритм, который еще только начал демонстрировать первые успехи, «свои», т.е. наиболее удобные именно для него коды. Тогда позже, когда придет более полное понимание всех тонкостей его работы, можно будет точнее подобрать и другие требуемые параметры кода и алгоритма.

14. Так что же нас ограничивает? Размножение ошибок!

Попробуем все же до конца разобраться в том, почему МПД группирует ошибки. Ведь эта проблема так или иначе уже упоминалась нами в процессе обсуждения экспериментальных данных по декодированию.

Почему это вообще происходит? Попадание значительной части ошибок из проверок, относящихся к одному символу, в проверки другого мы уже упоминали и раньше, в процессе обсуждения самых первых экспериментальных результатов. То, что снижение доли ошибок из одной группы проверок, попадающей в другую группу, очень важно, было показано выше на графиках рис. 15 для кодов разной длины. Так, может быть, группирование ошибок окажется меньшим или почти полностью отсутствующим, если коды сделать очень длинными? Но очень быстро компьютерные эксперименты вполне однозначно показали, что такой эффект снижения группирования ошибок декодера класса МПД если и есть, то очень и очень ограниченный по конечной результативности.

Увеличение длины кода дает очень мало. Эффект от роста длины почти такой же, как и от перемежения: если на входе декодера пакеты ошибок, то это хорошо, перемежение превращает ошибки как бы в случайные. А вот если мы работаем в ДСК, то пакеты сохраняются, но идут тоже с перемежением. Так что фактически достаточно заметных улучшений в работе МПД при использовании простых методов увеличения длины кодов не наблюдается.

Несколько более существенное снижение группирования ошибок происходит, если выбираются коды с очень малыми кодовыми скоростями. Но для действительно эффективного применения кодирования при малых скоростях следует выбирать и коды с очень большими значениями минимального кодового расстояния d . А это снова ведет к росту абсолютного числа ошибок в проверочных уравнениях и, значит, к увеличению группирования ошибок на выходе декодера мажоритарного типа уже по другим причинам. И хотя мы уже обсуждали немного причины группирования ошибок, относящиеся и к самим кодам, и к алгоритмам их декодирования, принять решение о дальнейших путях одновременного улучшения характеристик пары «МПД – оптимальный для него код» долгое время не удавалось.

Эта сложнейшая проблема теории и техники помехоустойчивого кодирования в те далекие времена, когда изучению мажоритарных методов уделялось еще какое-то внимание, фактически не была решена так, чтобы полученные аналитические результаты можно было бы успешно применять в прикладной области, т.е. непосредственно в технике кодирования.

В те давние 1970-е годы размножение ошибок анализировали примерно так. Обычно брали регистр синдрома и рассматривали его устойчивость, т.е. способность быстро переходить в нулевое состояние после того, как туда попала первая же ошибка декодера через цепь обратной связи. Но это был чисто комбинаторный эффект, который мало соответствовал реальному поведению простого ПД в канале с достаточно большим уровнем шума.

По формальным комбинаторным критериям длины зоны размножения ошибок самоортогональные коды оказались весьма хорошими по сравнению с некоторыми другими кодами, но из всего этого от исследователей размножения ошибок так и не последовало никаких конкретных рекомендаций по улучшению возможностей мажоритарных процедур.

Еще сложнее были проблемы по оценке группирования ошибок в тех кодах, где перед самой процедурой декодирования необходимо сначала выполнить процедуру формирования орто-

гональных проверок. Формальные комбинаторные результаты для подобных кодов оказывались очень различными, даже в некоторой степени противоречивыми. И они даже не всегда соответствовали реальным результатам моделирования кодов этого типа. Таким образом, комбинаторные методы были более чем условным критерием реального уровня размножения ошибок (РО). Чего-либо, что могло на самом деле указать путь реального прогресса для мажоритарных алгоритмов, не было совсем. Как же можно было в те годы поддержать исследования мажоритарных процедур, помочь декодеру МПД с очень высоким потенциалом эффективности обрести путевку в жизнь?

И решение было найдено! Правда, оно оказалось длиной в 20 лет. Это решение формировалось постепенно и продолжает развиваться в целом достаточно интенсивно, иногда медленно, изредка почти прекращая свое продвижение вперед, попадая в тупиковые ситуации, а иногда – и скачком. Но оно – это решение – есть!

Оказалось же это решение столь же трудным, сколь и плодотворным просто потому, что оно привлекло к решению задачи роста эффективности МПД известные методы из совершенно другой математической сферы, которые позволили решить эту проблему, что называется, «в лоб».

А новый подход, который стал отправной точкой долгого и трудного пути «улучшения породы» МПД, называется методом производящих функций вероятности (ПФВ). Но не простых, а обязательно многомерных! И прямо с самого начала таких совершенно необычных для теории кодирования исследований оказалось, что все имеющиеся возможности средств вычислительной техники того времени, середины 1970-х годов, для оценки реального уровня размножения ошибок были совершенно недостаточны. Казалось бы, чему удивляться. Ведь это было 30 лет назад. Это же столько эпох назад! (Для вычислительной техники это действительно целые эпохи.) Но самое удивительное состоит в том, что и сейчас, в начале нового тысячелетия точная оценка размножения ошибок, т.е. уровня группирования в МПД практически невозможна! Это выяснилось практически в первый же период работы с многомерными ПФВ. Оказалось,

что при точных расчетах основных эффектов сложнейшего явления размножения ошибок (РО) объем вычислений растет как экспонента от произведения числа итераций декодирования на кодовое расстояние и на вес того пакета ошибок, вероятность появления которого надо оценить. Скорее всего, точные методы вычисления даже только основных эффектов РО, таких, как вероятность появления пакетов ошибок МПД алгоритма заданного веса, будут недоступны даже для быстро совершенствуемых сейчас компьютеров еще многие и многие годы. Увы, но вполне возможно, что это не будет возможно для действительно мощных и длинных кодов никогда!

Но главное-то было сделано! Была сформулирована совершенно конкретная задача, которую надо было просчитать. А это почти всегда уже и решение такой задачи, даже если просчитать абсолютно точно все, что надо, фактически в принципе невозможно.

Имея четкие цели и критерии качества декодирования, исследователям стало намного легче заниматься оценками качества и соответствия кодов и алгоритмов МПД друг другу. Фактически абсолютно все успехи применения МПД связаны именно с эффективностью, скоростью и точностью вычисления оценок РО для этого алгоритма. За годы развития метода многомерных ПФВ среди разных классов кодов найдены такие группы наиболее известных блоковых и сверточных кодовых структур, для которых методы приблизительных оценок РО можно усовершенствовать столь значительно, что получение достаточно простых, но вполне содержательных оценок стало абсолютно доступным.

Оказалось, что группирование ошибок в весьма точном понимании этого эффекта как вероятности появления тех или иных конкретных пакетов ошибок на выходе алгоритма вполне понятным образом зависит практически от всего: от настроек порогов в пороговых элементах МПД, от весов каждой из проверок во всех итерациях декодирования, от положения границ областей решений в многопозиционных или даже просто в двоичных мягких модемах, от методов ортогонализации проверок,

способа выделения и настроек специальных порогов, которые ориентированы на особые взаимозависимые группы символов.

Еще более сложные зависимости в рамках оценок РО наблюдаются при формировании разнообразных каскадных схем с использованием МПД в одной или нескольких ступенях декодирования. Как и во всех других системах кодирования, МПД достигает действительно впечатляющих характеристик как по эффективности, так и по сложности реализации именно в каскадных кодовых структурах. В этом случае объемы необходимых оценок РО многократно увеличиваются, потому что при этом нужно согласовать между собой во много раз большее число различных параметров кодов и ступеней декодеров в двух или более каскадах декодирования.

Подчеркнем, что при всех тех огромных трудностях, которые приходилось преодолевать в те годы, да нередко приходится как-то частично избегать или все же решать и сейчас, чрезвычайно важно, что после завершения всех необходимых оценок и создания обычной или каскадной схемы с МПД его сложность остается совершенно такой же, какой она была до проведения процедуры адаптации декодера к коду по всем выбранным при начале проектирования критериям РО.

Однако вскоре быстро выяснилось, что и этой огромной проделанной работы оказалось совершенно недостаточно. В качестве конечного результата исследования РО на совершенно новом уровне необходимо было отдельно научиться строить достаточно быстро и непосредственно именно такие коды, которые были бы в наибольшей степени свободны от эффекта размножения ошибок. Это потребовало создания дополнительных самостоятельных критериев комбинаторного типа для отбора кодов и новых стимулов для создания еще более эффективных модификаций МПД алгоритмов.

В процессе этой работы выяснилось, в частности, что по мере увеличения минимального кодового расстояния кода, который требуется построить, растет и число критериев, которым должны удовлетворять такие коды. Число этих критериев быстро возрастает и они претерпевают довольно сложные изменения, например, по сравнению, с кодовой скоростью $R=1/2$. Все кри-

теории несколько видоизменяются как при уменьшении R и создании очень низкоскоростных кодов, так и наоборот, при росте скорости R .

Последняя постановка задачи о поиске хороших кодов также нашла свое успешное разрешение в виде пакета программ построения кодов в широком диапазоне кодовых скоростей R . В настоящее время достаточно просто строить необходимые для тех или иных приложений коды с низким уровнем РО в диапазоне $R=0,05-0,95$. Такая ширина диапазона возможностей оказывается вполне достаточной для абсолютно всех реалистических приложений теории и техники кодирования. Разумеется, что в особых случаях, которые пока не встречались разработчикам алгоритма МПД на практике, этот диапазон может быть произвольно расширен при самых небольших уточнениях дополнительных параметров кодов вплоть до любых значений R от 0,01 до 0,99. В настоящее время задача построения подобных кодов решается на основе одновременного рассмотрения вероятностных и комбинаторных свойств кодов с постоянным обращением в процессе построения к критериям качества кодов при очень умеренных объемах вычислений. Этот программный комплекс непрерывно развивается и совершенствуется, что позволяет продолжать работы по улучшению характеристик эффективности и сложности разных алгоритмов, в том числе и декодеров типа МПД.

Таким образом, после осознания идей, заложенных в алгоритм МПД, основная работа по усовершенствованию этого несомненно замечательного метода, при всех сложных и неоднозначных обстоятельствах создания декодеров этого класса, в ближайшем будущем будет по-прежнему состоять именно в непрерывных огромных усилиях по созданию кодов и алгоритмов со все более низкими уровнями подверженности эффекту РО. Это, как уже давно стало ясным, такая же долгая и интересная работа с весьма медленным продвижением вперед, как и для всех других методов коррекции ошибок, которые считаются сейчас в той или иной степени перспективными методами для новых систем связи. Единственное, но принципиальное отличие идей МПД от всех прочих методов декодирования со-

стоит только в одном: чрезвычайная простота реализации и отменная эффективность. Одновременно этими обоими чрезвычайно важными свойствами такого рода сейчас не обладает, судя по всему, ни один из алгоритмов декодирования для каналов с большим уровнем шума. Такова реальность соотношения свойств самых мощных алгоритмов, наиболее полезных для современных систем связи.

В чем же состоят технологические итоги уроков изучения РО?

В настоящее время построены процедуры поиска требуемых кодов, которые в одном из своих вариантов представляют собой перевод трехмерного массива коэффициентов порождающих полиномов кодов в некоторое пятимерное пространство, в котором строится специальный вектор-функционал, компоненты которого в некотором приоритетном порядке последовательно оптимизируются. Исходная сложность этой процедуры имеет порядок nd^4 . Но здесь тоже оказалось возможным получить у природы маленькую, но очень полезную призовую скидку: при детальном анализе возможностей такого подхода удалось избавиться от сомножителя n . А зависимость, близкая по порядку величины к d^4 , уже вполне реализуема даже при достаточно больших коэффициентах пропорциональности. Коды с $d < 30$ для $n < 400\ 000$ можно за несколько часов построить под разные требуемые уровни РО на современных персональных компьютерах общего назначения.

А в чем же проявляется неполнота решения проблемы? Что надо исследовать, разрабатывать в области алгоритмов МПД именно сейчас, теперь? Оказывается, что при попытке построить наилучший код для средних кодовых скоростей $R \approx 1/2$ при некоторых разумно выбранных значениях остальных кодовых параметров оказывается, что ожидаемого разделения множеств ошибок в проверках и, следовательно, требуемого минимального уровня РО нельзя добиться даже теоретически. Во многих случаях необходимо переходить к сложным кодовым структурам со специфическими наборами свойств, причем с ростом требуемого качества кода необходимость поиска только кодов со специальными параметрами становится все более очевидной.

А обычные коды большей частью таковы, что вероятности появления пакетов декодирования нескольких конкретных типов почти не зависят даже от уровня шума канала. Так что все очень непросто в королевстве МПД. Например, хороший СОК в смысле максимального отношения d/n при $d=11$ и $R=1/2$ имеет длину $n_d=112$. Но он почти непригоден для реализации на нем процедур типа МПД при большом уровне шума канала. Экспериментальные результаты для таких кодов мы уже обсуждали. А если мы хотим работать еще и на уровне эффективности турбо кодов, т.е. при максимально большом уровне шума, то сразу оказывается, что длина такого кода при том же значении $d=11$ должна быть не менее $n=20\,000$ битов, а в некоторых случаях и еще большей.

При других значениях величин R и d нередко получаются еще более экзотические результаты и параметры требуемых кодов. Разумеется, если посмотреть на сложившуюся ситуацию с более общих позиций, то особого удивления она вызывать не должна. Ведь для работы при кодовых скоростях, близких к пропускной способности канала C согласно границе сферической упаковки и построенных с ее учетом нижних оценок эффективности кодов на рис. 4, коды могут быть только очень и очень длинными. Как раз это и демонстрируют все эффективные методы, стремящиеся к хорошим результатам при $R \leq C$.

Именно поэтому и можно считать, что тематика развития алгоритмов типа МПД требует усилий еще многих и многих исследователей. А то, что при этом будут получены новые, еще более удивительные результаты, всем, кто занимается этими проблемами, абсолютно ясно. Мы думаем, что с этим согласятся и все заинтересованные читатели, которые прочитают эту книгу до конца.

А теперь можно и посмотреть, что же все-таки предлагают специалистам в области связи многопороговые процедуры, такие их версии, которые используют все основные теоретические достижения в области снижения уровня РО в декодерах мажоритарного типа.

15. Что умеют МПД

Рассмотрим характеристики МПД, которые могут быть получены при весьма небольших затратах на реализацию этого алгоритма декодирования. Для начала обсудим только те возможности МПД, которые определяют собственную корректирующую способность этого метода. Иначе говоря, все, что относится к обширному кругу методов комбинирования систем кодирования, включая очень мощные методы каскадирования, мы пока отложим до момента более широкого их обсуждения.

Мы предлагаем для этого посмотреть опубликованные всего несколько лет назад результаты моделирования основных алгоритмов коррекции ошибок некаскадного, будем говорить базового, типа. Они сами могут быть основой для каскадирования различного (параллельного, последовательного и т.д.) типа. Это, конечно же, МПД и АВ. Постараемся привести все основные и наиболее интересные данные из некоторых последних публикаций практически без купюр.

Итак, как известно, использование помехоустойчивого кодирования позволяет обеспечивать заданные достоверности передачи цифровых данных в спутниковых и других каналах связи при существенно более низком уровне сигнала по сравнению с вариантом передачи без кодирования. Для каналов, в которых энергетика передачи ограничена, это создает эффект увеличения мощности передатчика, что и обуславливает техническую ценность использования кодирования. Например, применение алгоритма Витерби (АВ) на кодовой скорости $R=1/2$ и длине кодирующего регистра $K=7$ при использовании «мягкого» демодулятора обеспечивает при двоичной ФМ энергетический выигрыш кодирования (ЭВК), равный при вероятности ошибки на бит $P_b(e)=10^{-5}$ величину порядка 5,2 дБ (график 4 на рис. 16). Вертикаль $C_{\text{ДСК}}$ указывает на границу пропускной способности канала ДСК, равную 0,5, т.е. $C=R$.

Указанная величина ЭВК оказывается достаточно большой, чтобы активно применять коды в реальных системах связи. Но она существенно ниже теоретически достижимой границы, соответствующей для указанных выше параметров ко-

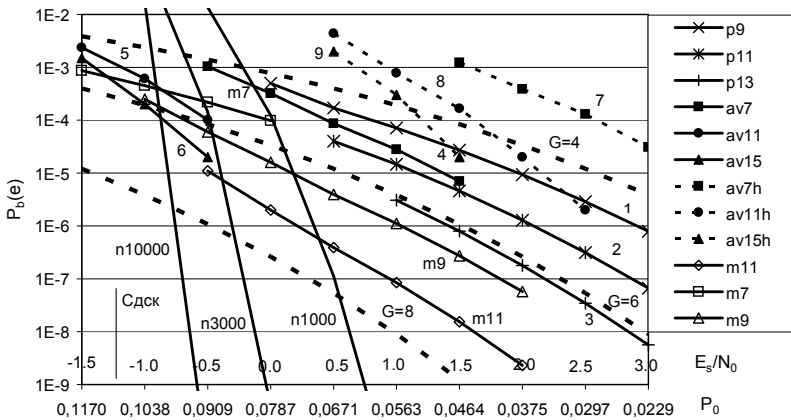


Рис. 16. Эффективность базовых алгоритмов помехоустойчивого кодирования

кодирования и канала величине ≈ 9 дБ. Однако дальнейшее повышение ЭВК для реальных систем кодирования связано с быстрым, а в некоторых случаях даже экспоненциальным с ростом длины кода, усложнением обработки сигнала на приемной стороне. Поэтому для многих высокоскоростных каналов, к которым предъявляются повышенные требования к достоверности приема цифровых данных, достижение больших значений ЭВК оказывается крайне сложной и очень важной целью, которая может быть реализована только на основе помехоустойчивого кодирования.

Вместе с тем задача полного использования энергетики каналов становится все более актуальной как из-за технических и экологических ограничений на мощность излучаемых сигналов, так и из-за конечной емкости выделяемых для тех или иных систем связи полосы частот. Поэтому, хотя рост вычислительных возможностей, закладываемых в аппаратуру связи, позволяет решать многие сложные задачи, недоступные ранее, пробле-

ма повышения ЭВК на длительное время останется актуальнейшей задачей связи для многих типов каналов.

Одним из простейших способов увеличения ЭВК является использование многопороговых декодеров (МПД). Они представляют собой модификацию обычных пороговых декодеров, с помощью которых при правильном выборе весьма специфических кодов длиной в сотни и тысячи битов, отобранных по критериям минимизации известного эффекта размножения ошибок, удается получить декодируемое сообщение, мало отличающееся от результата оптимального декодера, но при линейной сложности алгоритма декодирования вместо экспоненциальной, как, в частности, это имеет место при использовании АВ. В правильно построенном МПД для хорошего в указанном выше смысле кода оказывается возможным многократное уточнение принятого из канала сообщения до тех пор, пока решение МПД не станет близким к решению оптимального, по максимуму правдоподобия, декодера. При этом количество таких попыток последовательного улучшения решения предыдущего шага декодирования может быть от трех до сорока и более.

На рис. 16 приведен график 2 зависимости вероятности ошибки на бит $P_b(e)$ от уровня шума в канале типа ДСК для сверточного МПД и кода с расстоянием $d=11$ при кодовой скорости $R=1/2$. Приведенный результат относится к случаю использования простого «жесткого» двоичного демодулятора, который уменьшает пропускную способность формируемого им канала по сравнению с «мягким» модемом, используемым декодерами АВ, почти на 2 дБ. Тем не менее, МПД оказывается настолько более эффективным, что даже в таких совершенно неэквивалентных условиях сравнения его характеристики несколько превосходят характеристики АВ. Подчеркнем, что это оказалось возможным из-за почти оптимального решения МПД для гораздо более длинного кода, чем код длины $K=7$, который был взят для оценки возможностей АВ. График 1 дан для МПД и кода с $d=9$, а 3 – для кода с $d=13$.

В целях более удобного сопоставления алгоритмов декодирования и по ЭВК пунктиром на рис. 16 указаны также уровни равного ЭВК для кодов с $R=1/2$: $G=4$, $G=6$ и $G=8$ дБ. Они помо-

гают сравнить коды по главному критерию их эффективности – энергетическому.

Для иллюстрации результата работы алгоритмов в одинаковых условиях использования «мягких» модемов был также взят код с $R=1/2$ и соответствующий ему «мягкий» МПД. В связи с тем, что используемый для МПД код имел весьма большую длину и минимальное кодовое расстояние $d=9$, его декодирование практически не отличалось от оптимального даже при отношении сигнал/шум в канале порядка $-1,0$ дБ, как следует на рис. 16. На графике m9 приведены теоретическая оценка и совпадающие с ней результаты моделирования работы этого МПД. Для $d=7$ и $d=11$ аналогичные характеристики представлены графиками m7 и m11.

Исходя из известных оценок и представленных результатов, можно считать, что МПД обеспечивает теоретически предельные характеристики по ЭВК при весьма высоких уровнях шума. Отметим, что если осуществляется переход к более длинным кодам, то при их декодировании по АВ рост эффективности оказывается не столь значительным, как можно было бы ожидать. В частности, код с $K=11$ при использовании АВ также при $R=1/2$ характеризуется графиком 5, а возможности кода с декодированием по АВ для $K=15$ приведены на графике 6, что показывает ограниченный рост эффективности этого алгоритма даже в случае роста объема вычислений АВ по сравнению с $K=7$ на много десятичных порядков.

Заметим, что увеличение эффективности МПД оказывается значительным по сравнению и с обычным пороговым декодированием, и с АВ вследствие такой организации в нем процедур коррекции, при которой при сверточном варианте реализации кодирования задержка декодирования становится весьма значительной, достигая многих десятков тысяч кодовых символов из-за большого роста длины кодов и значительного числа итераций декодирования. Но следует подчеркнуть, что с теоретической точки зрения высокая эффективность обеспечивается только действительно весьма длинными кодами. На рис. 16 представлены для $R=1/2$ в обычном ДСК нижние оценки вероятности ошибки на бит блоковых кодов длины от $n=1000$ до $n=10\,000$

в предположении оптимального декодирования, что заведомо недостижимо. Графики 7, 8 и 9 снова соответствуют кодам с $K=7, 11$ и 15 , декодируемым по АВ, но в ДСК, т.е. при моделировании их работы с «жестким» модемом, когда в равных условиях оцениваются потенциальные возможности кодов, а не алгоритмов.

Сравнение основных конкурирующих алгоритмов для спутниковых и космических каналов связи в разных публикациях показывает, что среди базовых алгоритмов коррекции ошибок предпочтительными являются АВ для коротких кодов и МПД, а в ДСК у МПД конкурентов в принципе быть не может, так как в этом случае предлагается метод декодирования, который часто мало отличается по эффективности от оптимального, но имеет очень простую реализацию. Если же большая задержка принятия решения не является ограничением, МПД оказывается много быстрее, чем АВ, а также эффективнее и в гауссовских каналах.

Полезно подчеркнуть и то, что указанная более 20 лет назад Берлекэмпом ценность дополнительно получаемого с помощью кодирования ЭВК, равная 1 миллиону долларов за каждый децибел выигрыша, в настоящее время многократно возросла из-за дефицита свободного спектра частот и множества технических и экологических проблем ограничительного плана. Это значит, что алгоритмы декодирования и далее будут очень высоко цениться по обеспечиваемому ими совокупности уровня ЭВК, быстродействия и простоте реализации.

Как известно, на основе базовых алгоритмов можно строить и каскадные схемы декодирования, которые еще более повышают эффективность применения кодирования. Они заслуживают отдельного анализа. Но представляется очевидным, что каскадные схемы всегда оказываются тем более эффективными, чем более высокие результаты обеспечивают коды и алгоритмы их декодирования, используемые на каждой ступени соответствующей кодовой конструкции. Отсюда можно сделать достаточно правдоподобное предположение о том, что МПД и в каскадных или итеративных схемах обеспечат более высокие характеристики, чем в случае других подходов к реализации кас-

кадирования. Следует также отметить, что и сама идея каскадирования изначально была связана с использованием весьма длинных кодов. А это как раз и соответствует той области кодовых параметров, в которой выбор длинных кодов позволяет организовать эффективное декодирование на базе алгоритмов типа МПД.

Таким образом, представленные (в цитируемом докладе) данные позволяют указать на новые возможности реализации алгоритмов кодирования типа МПД, мало отличающихся от оптимальных в широком диапазоне параметров каналов спутниковой и космической связи. Применение МПД на практике уже доказало достижимость заявляемых характеристик и широкие возможности их применения для блоковых, сверточных, параллельных, итеративных и каскадных кодов при использовании как двоичных, так и недвоичных, в том числе многопозиционных сигналов.

На этом мы и закончим цитирование одной из последних публикаций по методам МПД. Как следует из представленных графиков, среди базовых алгоритмов декодирования МПД практически не имеет конкурентов. Заметим, что успешное решение проблемы РО позволило при $R=1/2$ применять МПД в ДСК при вероятности ошибки до 0,07, а в гауссовском канале достичь отношения сигнал/шум порядка -1 дБ и даже несколько меньших значений. Это и значит, что все основные проблемы разработки МПД к настоящему времени фактически полностью решены. К этому можно только добавить, что из графиков, представленных на рис. 16, следует что МПД является отличным кандидатом на использование и в каскадных схемах. Хорошо известно из опыта разработок эффективных каскадных схем, что внутренний код в такой конструкции должен снижать уровень ошибок канала на входе до величин порядка 10^{-3} . На рис.16 видно, что МПД выходит на этот уровень при отношении сигнал/шум в гауссовском канале порядка $-1,5$ дБ. Это и гарантирует успешность его применения во многих каскадных конструкциях. Но этой интереснейшей проблемой мы займемся несколько позже.

16. Но еще важнее – экономика

Сейчас мы предлагаем рассмотреть один из весьма острых вопросов при выяснении необходимости применения теоретических достижений к реальной практической сфере деятельности. Вопрос звучит очень просто: а вообще-то ЭВК имеет какое-то экономическое выражение, чтобы его имело смысл применять при сопоставлении систем и, главное, продолжать теоретически и практически (т.е. с разумными затратами) бороться за его увеличение? Оказывается, что да, все это очень важно и более того, ценность кодирования все более возрастает. В самом деле, применение кодирования позволяет многократно повысить достоверность обмена информацией, увеличить скорость ее передачи, существенно уменьшить размеры передающих антенн или повысить дальность, устойчивость и надежность связи или, наконец, как мы уже отмечали, значительно снизить необходимую мощность передатчика. При совместном использовании кодирования с другими методами одновременно достижимы все перечисленные выше цели, а также снижение объема и времени передачи данных, скрытность и защита обмена информацией, сужение необходимой полосы частот для связи и ряд других преимуществ. Именно эти достоинства кодирования и привели к тому, что уже в течение нескольких десятилетий применение все более мощных методов помехоустойчивого кодирования является обязательной процедурой обработки информации во всех системах эффективной связи.

Наиболее очевидна экономическая эффективность кодирования, если рассматривать возможность снижения размеров дорогостоящих антенн. Но и постоянный дефицит энергии в космических и спутниковых системах связи тоже совершенно очевиден. Точный расчет финансового баланса проекта современной системы связи как функции от результата применения кодирования весьма сложен, особенно, когда анализируется комплексное воздействие кодирования и на выбор антенн, и на источники питания передатчика, и на надежность приема или вид систем сигналов, на количество возможных ретрансляторов, ширину полосы частот и другие проблемы.

Но уже более 20 лет назад выдающийся американский специалист в области теории кодирования Э.Р. Берлекэмп указывал в своем обзоре достижений в этой области, что каждый дополнительный децибел энергетики за счет кодирования в системах среднего масштаба 80-х годов давал экономический эффект порядка миллиона долларов. Напомним, что стоимость современных систем связи растет все быстрее и уже совершенно несопоставима с теми затратами, которые были возможны несколько десятилетий назад. И добавим к этому, что сейчас существенно возрастают взаимные помехи приему между различными системами, появляются все больше экологических ограничений на мощность передатчиков, усиливается потребность в резком увеличении скоростей обмена данными между сетями, а также в значительном дальнейшем росте его стоимости. И эти тенденции в условиях глобализации экономики и взрывной информатизации общества будут только усиливаться. Отсюда можно сделать очевидный вывод, что возрастание на несколько порядков экономической (как и экологической, и вообще, системной!) ценности увеличения достижимых уровней ЭВК на базе методов помехоустойчивого кодирования становится чрезвычайно ценным результатом внедрения теоретических достижений в практику создания компьютерных систем связи.

17. Что заказывают связисты

Поскольку ЭВК удобен как главный критерий поиска кодов и алгоритмов их декодирования, то полезно взглянуть на предельно возможные, с точки зрения теории, уровни ЭВК, а также, для ориентировки, на реальные возможности некоторых конкретных методов коррекции ошибок.

Возможности кодирования для обеспечения требуемой эффективности по этому новому критерию представлены на рис. 17. Как следует из рисунка, абсолютная граница Шеннона соответствует условиям $R=0$ и $M \gg 1$. Она и все другие границы получены из естественного условия $R < C$. Разумеется, при получении этих предельных оценок предполагается, что использует-

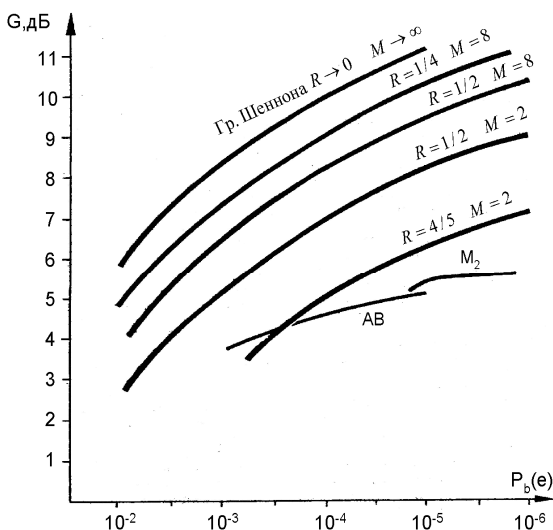


Рис. 17. Предельные значения ЭВК из условия $R < C$ в гауссовском канале для различных кодовых скоростей и уровней квантования сигналов в модеме

ся оптимальное декодирование со сколь угодно высокой сложностью реализации. Тогда заданные средние вероятности $P_b(e)$, скорость R и число уровней квантования модема M определяют в достаточно общем смысле меру эффективности кодирования. При переходе от $M=2$ (случай «жесткого» модема, просто определяющего, какой символ пришел из канала) к «мягкому» модему с $M=8$, который оценивает свои решения, возможно получение дополнительно ЭВК чуть менее 2 дБ. При увеличении R , наоборот, предельные возможности кода даже при наилучшем декодировании заметно снижаются. Ну и, конечно, с ростом требований к достоверности привлекательность использования кодирования становится все более очевидной, так как достижимый ЭВК быстро растет.

Ясно, что если требования к достоверности передачи растут, то достижимая величина ЭВК значительно возрастает.

А это и значит, что ценность кодирования в дальнейшем будет только расти.

Полезно также посмотреть, какие значения ЭВК обеспечивают конкретные коды и алгоритмы. Кривая АВ на рис. 17 показывает возможности алгоритма Витерби для сверточных кодов с длиной кодирующего информационного регистра $K=7$, которые широко используются в настоящее время. АВ обеспечивает ЭВК порядка 5,2 дБ при $P_b(e)=10^{-5}$, что можно признать значимым достижением. Но заметим, что почти 2 дБ от полных 5,2 дБ роста энергетики, как только что было указано, получены за счет «мягкого» модема.

Кривая M_2 на обсуждаемом рис. 17 представляет возможности декодирования по типу МПД. Более того, этот график соответствует работе с «жестким» модемом, т.е. работу в ДСК при $M=2$. Если бы широко используемый АВ тоже работал при $M=2$, то его ЭВК был бы порядка 3,3 дБ. Но можно перейти к МПД процедурам, которые могут работать и с «мягким» модемом. Тогда вместо 5,6 дБ ЭВК при $P_b(e)=10^{-5}$ этот метод даст при весьма высокой достоверности ЭВК более 7,3 дБ. Заметим, что пример МПД процедуры показывает, что можно эффективно декодировать очень длинные коды достаточно простыми методами. В рассматриваемом примере длина кода составляет около 800 битов и может быть взята еще гораздо большей. А сложность декодера, реализующего АВ растет как экспонента от длины кода K , которая в следствие этого вынужденно весьма мала.

Как видно из сопоставления графиков для предельных возможностей и реальных характеристик, до границы, достижение которой допускает теория, еще очень далеко, несколько децибел энергетики при $R=1/2$. При других значениях R ситуация качественно остается той же самой.

18. А если еще лучше?

Можно и лучше. Уже много лет для многих кодовых систем используют каскадный метод кодирования. Он состоит в том, что в передатчике ставятся два или более, например, последова-

тельно соединенных кодера, а в приемнике (конечно, в обратном порядке) соответствующие им декодеры. Такие схемы позволяют намного упростить сложность процедуры декодирования по сравнению с базовыми, теми из которых они состоят, например, с тем же АВ в сочетании с широко применяемыми кодами Рида-Соломона. При этом одновременно значительно растет ЭВК, особенно для очень высоких уровней достоверностей. А это тоже очень важно. Достижимые границы ЭВК при малых вероятностях ошибки декодера гораздо выше, чем для больших вероятностей. Да и требования к достоверности обмена данными современная революция в области информатизация общества повышает очень быстро.

Дело еще и в том, что при решении проблемы построения той или иной системы кодирования и вынесении приговора типа «получится – не получится» после быстрого (но, конечно, квалифицированного) просмотра теоретических характеристик и других основных особенностей алгоритмов коррекции берут обязательно конкретные схемы и определяют, сколь быстро они будут работать, какие обеспечат вероятностные характеристики, насколько задержат момент окончательного вынесения решения декодером и т.д. Но самое главное – это во что обернется аппаратная реализация предлагаемого декодера и как он будет работать с остальной аппаратурой системы связи, в которую он будет включен. А при микропроцессорной реализации декодера главным критерием, по существу, является число его операций, пересчитанное к одному декодируемому символу. Вот здесь-то и бьется мысль конструкторов и программистов над проблемой, как и что упростить или облегчить. А ограничение тем или иным конкретным алгоритмом вовсе не является обязательным. Вернее, это вообще никогда не должно быть ограничением.

Важнейшим из таких эффективных методов значительного повышения эффективности кодирования и является каскадирование. Ему посвящены многие сотни статей, он рассматривается практически во всех монографиях, патентуется в разных своих модификациях и при этом продолжает интенсивно исследоваться. При наличии многих важных особенностей в раз-

личных системах каскадирования все они обладают одним общим свойством. Главная идея состоит в том, что кодирование осуществляется двумя или более (например, последовательно) соединенными кодерами, работающими на передающей стороне, а в приемнике ставят два декодера, включенных, разумеется, в обратном порядке. Оба кодера вносят в передаваемое сообщение некоторую избыточность и, таким образом, кодовая скорость $R_0=R_1R_2$ результирующего кода будет (только в случае последовательного, конечно, каскадирования) еще меньше, чем у составляющих его кодов. А огромное значение каскадирования для теории, и особенно техники декодирования, состоит в том, что вероятностные характеристики кодирования преобразуются столь сильно, что при правильном выборе составляющих кодов, который учитывает при этом и все основные вопросы их согласования между собой, при общем улучшении характеристик, особенно для очень большой требуемой достоверности декодирования, возможно одновременное снижение и затрат на реализацию декодирования. Качественное

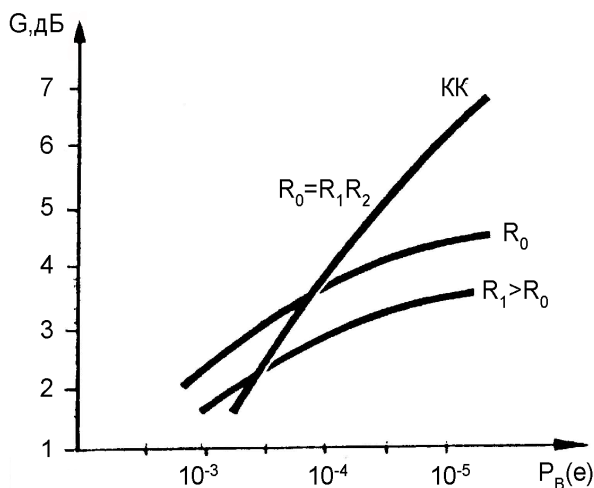


Рис. 18. Принципы каскадирования в каналах гауссовского типа

поведение графиков ЭВК в зависимости от требуемой достоверности показано на рис. 18. Более слабые характеристики внутреннего кода с R_1 входящего в состав каскадной схемы, обусловлены выбором параметров каскадного кода, обеспечивающих одинаковые результирующие кодовые скорости у обычного некаскадного кода, взятого в качестве примера, и у каскадной конструкции. Но внутренний код с более высокой скоростью, конечно, должен несколько ухудшить вероятностные характеристики на выходе, учитывая к тому же несколько меньшую сложность его декодера, чем у некаскадного кода. Результирующие же характеристики декодера каскадного кода (КК) при малой выходной вероятности ошибки всегда много лучше, чем у исходного, причем иногда возможны варианты такого выбора параметров КК, что некаскадные методы декодирования вообще ни при какой сложности реализации не смогут достичь тех же пределов эффективности, что и КК. Такова в общем идея, которая может изменить ситуацию в сравнительном анализе различных алгоритмов декодирования. Может ли?

Невообразимая сложность проблемы учета всех методов при сопоставлении различных каскадных кодов и алгоритмов их декодирования заставляет обратиться только к самым лучшим по критерию ЭВК методам с использованием КК, которые, собственно, и будут сравниваться с МПД. Для большинства базовых методов декодирования, которые применяются в первом и втором каскадах КК, справедливо простое утверждение, которое фактически и решает спор при более полном анализе всех алгоритмов, включающем и каскадные схемы с их использованием: более эффективные базовые алгоритмы для кодов, входящих в КК, действительно позволяют добиваться более высоких значений ЭВК и для всего КК в целом.

Строго говоря, это утверждение весьма серьезно, если правильно. В частности, оно должно приводить к более высокой эффективности и КК на основе МПД. В тех случаях, когда внешний код является двоичным, для получающегося такого класса схем чаще используют термин «итеративный код» (ИК), а не КК, но это различие в терминологии не столь принципиально.

Можно отметить, что для КК имеются более оптимистичные, чем для ИК, теоретические оценки помехоустойчивости. Однако, это не влияет сколько-нибудь заметно на характеристики КК и ИК при конкретных, хотя, порой, и не малых длинах кодов.

19. Как дела у конкурентов?

В настоящей науке, развивающейся обычно вширь по многим направлениям, конкурирующих между собой направлений всегда хватает. Коды никогда не были в этом плане исключением. Достаточно высокие уровни ЭВК демонстрируют различные методы. Большинство из них каскадные и итеративные. Длины используемых ими кодов составляют тысячи и десятки тысяч символов, как того и требует теория.

Наиболее интересными в настоящее время кажутся результаты использования турбо алгоритмов для одноименных кодов. Этой тематикой заняты тысячи исследователей по всему миру. Для примера, при $R=1/2$ и $P_b(e)=10^{-5}$ турбо коды дают рекордные уровни ЭВК, которые лишь на 0,7 дБ хуже, чем это позволяет граница $R=C$. Но ничего не бывает бесплатно. Алгоритмы для турбо кодов должны работать с действительными числами, а иногда и с некоторыми сложными для вычислений функциями, выполняя при этом весьма большое число итераций.

МПД в каскадном варианте обеспечивает на несколько десятых долей децибела меньшие значения ЭВК, чем турбо коды. Это тоже чрезвычайно важный результат. Но МПД работает только с небольшими целыми числами и даже в этом своем тоже рекордном варианте остается таким же простым в реализации, как и самый обычный пороговый декодер, который мы уже рисовали выше. Множество внедрений МПД в различные системы связи за истекшие годы прошло легко и успешно. И те несколько десятых долей децибела по энергетике, которые сейчас МПД немного не добывает по сравнению с турбо кодами, будут достигнуты в самое ближайшее время.

20. А если канал не двоичный?

Рассмотрим обобщение многопорогового декодирования (МПД) для двоичных данных в гауссовских каналах на недвоичные симметричные каналы. Ценность этого метода заключается в том, что мажоритарные алгоритмы имеют всего лишь линейный рост сложности (числа операций декодирования) от длины кода n . Поскольку обычно оптимальные методы характеризуются экспоненциально нарастающей с длиной кода сложностью, применение недвоичных МПД, обозначаемых далее как QМПД, представляется особенно желательным. Еще более существенно, что в случае больших значений основания кода q , $q > 10$, практически невозможно создать эффективные истинно оптимальные декодеры (ОД), поскольку при этом их сложность в большинстве случаев будет иметь вид q^k , где k – длина кодирующего регистра. Это и определяет важность применения QМПД, поскольку, например, возможности декодеров для кодов Рида-Соломона (РС) очень ограничены, а их сложность реализации неоправданно велика.

Пусть задан q -ичный, $q > 2$, симметричный канал с вероятностью ошибки $p_0 > 0$, такой, что при передаче любой исходный символ кода переходит в один из оставшихся $q-1$ символов случайно, независимо и равновероятно. По аналогии с двоичным симметричным каналом без памяти (ДСК) назовем этот канал также q -ичным симметричным каналом (QСК). Для этого канала оптимальным решением при передаче любого символа будет такое, возможно, единственное кодовое слово из q^{nR} возможных, которое отличается от принятого сообщения в минимальном числе символов кода (здесь предполагалось, что n – длина кода, выраженная числом символов кода, R – кодовая скорость, $R < 1$).

Рассмотрим линейный недвоичный код, проверочная матрица которого имеет такой же вид, как и в двоичном случае, т.е. состоит только из нулей и единиц. Пусть эта матрица соответствует самоортогональному систематическому блоковому или сверточному коду, которые мы уже несколько раз рассматривали. В этом случае слова минимального веса d , где d – минималь-

ное расстояние кода, имеют единственный ненулевой символ i_k , со значением q , $q > 0$, в его информационной части. Поскольку проверочные (а значит, и порождающие) матрицы кода содержат только нули и единички, то операции кодера и декодера по формированию проверочных символов кода и вычислению синдрома S принятого сообщения являются только сложениями. Таким образом, для кодирования и декодирования не требуется наличие двоичного поля, а достаточно только определить группу по сложению для целых чисел. Это дополнительно и очень существенно упрощает все процедуры кодирования и реализации последующего декодирования.

Пусть декодер типа QМПД устроен так, что после вычисления обычным образом вектора синдрома S принятого сообщения процедура декодирования состоит просто в том, что для очередного контролируемого пороговым (двоичным!) элементом информационного символа кода i_k происходит подсчет числа и определение значений двух относящихся к нему и наиболее часто встречающихся проверок кода, например, q_1 и q_2 , причем q_1 встречается m_1 раз, q_2 — m_2 раз, $m_1 > m_2$, а остальные значения проверок для декодируемого символа i_k встречаются также не более m_2 раз. Тогда QМПД при каждом изменении символа i_k будет переходить ко все более правдоподобным решениям. Здесь абсолютно справедливы все методы анализа и рассуждений. Которые мы уже применяли для анализа работы МПД в ДСК и гауссовских каналах. Если окажется, что два наиболее часто встречающихся значения проверок таковы, что $m_1 = m_2$, то символ i_k не изменяется и делается попытка декодирования любого другого информационного символа кода. Наиболее существенным обстоятельством, повышающим корректирующие возможности описанного двоичного МПД по сравнению с двоичным, является возможность принимать безошибочные решения при больших значениях q всего при двух правильных проверках относительно i_k из d возможных. Это обычно происходит в том случае, когда все неправильные проверки s_i относительно декодируемого символа i_k имеют различные значения s_i , $q > s_i > 0$. Для больших значений q это условие почти всегда выполняется.

Рассмотрим, как можно вычислить нижнюю оценку вероятности оптимального декодирования для кода, задаваемого описанным выше способом. Во всех подобных случаях это будет выявление наиболее часто встречающихся условий того, что вектор ошибки будет иметь расстояние Хемминга до ближайшего кодового слова меньше, чем его собственный вес. В силу линейности кода этого достаточно для вынесения неправильного решения даже оптимальным переборным алгоритмом. Рассматривая такой вектор ошибки, можно учесть, что нужно анализировать только те символы этого вектора, которые соответствуют позициям проверок относительно очередного декодируемого символа i_k .

Выпишем вероятности таких наиболее частых событий, которые приводят к ошибкам оптимального декодера (ОД).

К искомым векторам ошибки относятся такие, что:

- все проверочные символы и декодируемый символ i_0 ошибочны:

$$P_1(e) = p_0^{J+1};$$

- все проверочные символы ошибочны, но два из них одинаковы, а i_0 принят верно:

$$P_2(e) = \frac{(1-p_0)p_0^J J(J-1)}{2(q-1)} \prod_{i=1}^{J-2} \left(1 - \frac{i}{q-1}\right);$$

- есть один правильно принятый проверочный символ, а остальные ошибочны, как и i_0 :

$$P_3(e) = J(1-p_0)p_0^J;$$

- есть один правильно принятый проверочный символ, а также i_0 , но из всех остальных неправильно принятых символов есть 3 одинаковых значения ошибок:

$$P_4(e) = \frac{(1-p_0)^2 p_0^{J-1} J!}{6(J-4)!(q-1)^2} \prod_{i=1}^{J-4} \left(1 - \frac{i}{q-1}\right);$$

- есть 2 правильных проверочных символа, а все остальные, включая i_0 , неправильны, причем есть 2 ошибочно принятых проверочных символа с одинаковыми значениями, а также некоторое количество других подобных ситуаций, число типов которых невелико:

$$P_5(e) = \frac{(1-p_0)^2 p_0^{J-1} J!}{4(J-4)!(q-1)} \prod_{i=1}^{J-4} \left(1 - \frac{i}{q-1}\right).$$

Перечисленных событий вполне достаточно, чтобы для большинства реальных условий применения кодов получать удовлетворительные по точности вероятностные оценки потенциальной помехоустойчивости кода. А поскольку QМПД на каждом шаге стремится к решению ОД, то можно ожидать, что при некотором достаточно высоком уровне шума он в большинстве случаев достигнет искомого оптимального решения. В этом случае его вероятности ошибки декодирования на символ будут мало отличаться от оценок для ОД, получение которых мы только что рассмотрели.

Особенно удобно в технических системах работать с данными, имеющими байтовую структуру. Отметим, что кроме кодов Рида-Соломона (РС) в настоящее время вообще нет других сколько-нибудь эффективных методов декодирования двоичных символьных данных. Сравним вероятностные характеристики кодов РС с возможностями QМПД. Будем считать, что выбран код РС длины 255 (символ – 8 бит). Подчеркнем, что для QМПД никаких ограничений по длине кода вообще нет, поскольку он работает с числами, образующими группу целых чисел по сложению, выполняя в ней только операции сложения и сравнения в выбранном множестве. Очевидно, что двоичный пороговый элемент, рассмотренный выше при описании операций в QМПД, – простейшее устройство или подпрограмма с числом операций N сложения и сравнения небольших целых чисел $N \approx 10-50$ для всех тех небольших значений минимального кодового расстояния d , $d < 15$, которое следует применять в таком декодере.

На рис. 19 представлены характеристики декодеров для кодов РС длины $n=255$ (обозначены как RS) и QМПД в QСК. По горизонтальной оси отложены вероятности ошибки p_0 в указанном канале, а по оси ординат – средние вероятности ошибки на символ в результате декодирования. Для достижения решения, обычно совпадающего с оптимальным или близкого к решению ОД, QМПД для $q=256$ необходимо 5–20 итераций (повторных

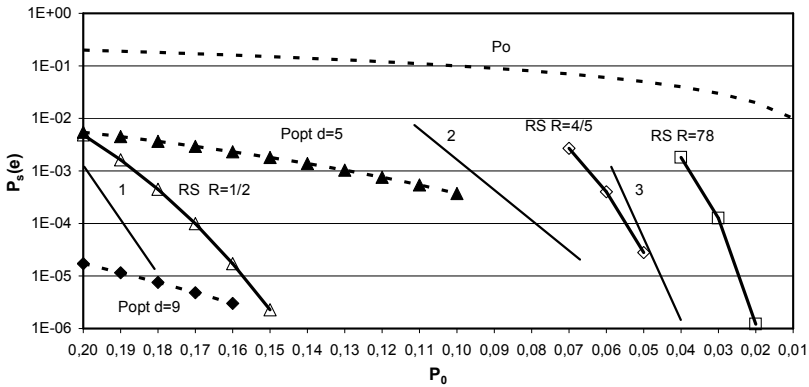


Рис.19. Характеристики недвоичных МПД и декодеров для кодов Рида-Соломона.

попыток) декодирования принятого сообщения. Это полностью соответствует методу МПД для двоичных кодов.

Как следует из вида графиков зависимостей средней вероятности ошибки декодирования на символ $P_s(e)$ от вероятности p_0 канала QСК на входе декодеров для кодовых скоростей $R=1/2$, $R=4/5$ и $R=7/8$, простейший по своему устройству QМПД представлен графиками 1, 2 и 3 соответственно для указанных выше кодовых скоростей и обеспечивает гораздо более высокие характеристики, чем декодеры для кода РС, благодаря несколько большей длине $n=1000$ используемых кодов и хорошей сходимости решений QМПД к решению ОД. Заметим, что в настоящее время неизвестны другие алгоритмы декодирования с приемлемой сложностью реализации, которые могут обеспечить такие же характеристики. При увеличении длин кодов характеристики QМПД могут быть дополнительно существенно улучшены.

Очевидно, что каскадирование нескольких недвоичных МПД также значительно улучшит вероятностные характеристики декодирования без значительного увеличения сложности, т.е. числа операций, осуществляемых декодером, работающим

только с целыми числами и не выполняющего никаких операций умножения или деления. Это является его решающим преимуществом перед алгоритмами для кодов РС при сопоставлении их по сложности реализации.

Представленные результаты позволяют утверждать, что описанные почти 20 лет назад недвоичные МПД обладают действительно высокой эффективностью, недоступной для декодеров кодов РС. При этом сложность их реализации весьма невелика и, как показывает детальный анализ, может быть дополнительно значительно снижена.

21. А если в эфире тесно...

Как мы уже привыкли справедливо считать, энергия у связистов в большом дефиците. Но ведь и все радиодиапазоны упакованы максимально плотно, поделены различными службами и странами. Постепенно наша цивилизация сдвигается в сторону все более высоких радиочастот. Других путей, кроме, естественно, еще большей экономии частот, нет. Еще чуть-чуть, и начнутся гораздо более сложные проблемы, связанные с использованием эфира: кто, сколько и когда. А ведь введение кодирования при сохранении той же информационной скорости, например 10 Мбит/с, требует в $1/R$ раз большей скорости передачи двоичных символов по сравнению с простой пересылкой, когда не применяется помехоустойчивое кодирование.

Для рассматривавшихся выше кодов кодовая скорость $R=1/2$ приведет к расширению спектра сообщения вдвое. А это уже серьезно. Что делать? Если хочется и обеспечить ЭВК, и крайне экономно израсходовать отпущенный спектральный диапазон, можно увеличить R .

Итак, что могут дать технике связи коды с большой скоростью? На рис. 20 показаны возможные значения ЭВК для самого сложного из доступных сегодня АВ при $R=3/4$ с жестким модемом, граница вычислительной скорости $R_1=4/5$ для ДСК, ставящая теоретические пределы для последовательных алгоритмов с $R=0,8$, а также возможности кодов БЧХ различной длины при $R=0,8$. Еще один график относится к МПД также с $R=0,8$, ко-

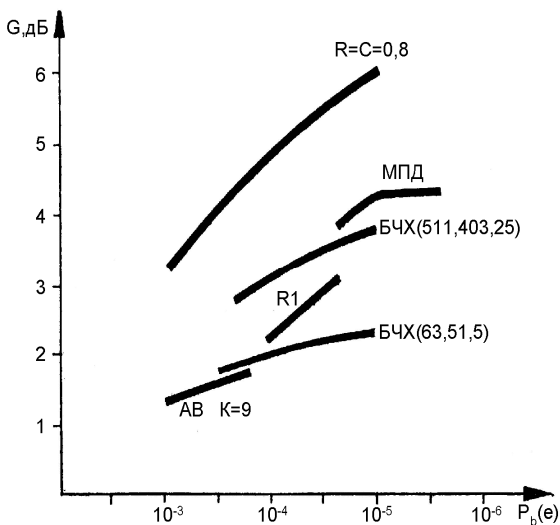


Рис.20. Энергетические характеристики алгоритмов декодирования в ДСК при $R \approx 0,8$

торый около 15 лет назад уже был создан на матричной БИС. Если заметить, что возможности кодов с ростом R убывают, то характеристики АВ при $R=3/4$ всегда можно использовать в качестве верхней оценки его возможностей при $R=0,8$, так как существуют достаточно хорошо разработанные приемы использования АВ с более низкими, чем требуется, скоростями для работы с высокоскоростными кодами.

Основной вывод по сопоставлению кодов и алгоритмов их декодирования при довольно высокой скорости $R=0,8$ состоит в сравнении позиций МПД и в этой области, если возможно использование длинных кодов. Интересно, что при $P_b(e) < 10^{-3}$ МПД для ДСК будет лучше мягкого АВ и существенно обгоняет по своим возможностям последовательные алгоритмы, обеспечивая более хорошую энергетику, чем даже труднодостижимые теоретические границы для этих процедур. Что касается кодов БЧХ, то для тех длин, когда они дают приемлемый результат по

энергетике, алгоритмы их декодирования становятся уж слишком сложными, чтобы их имело смысл реализовывать для достаточно высокоскоростных каналов связи.

22. Кодируем источники тоже

Обзор соседних с помехоустойчивым кодированием сфер деятельности неожиданно дал еще одну возможность удивиться тому, насколько наш мир взаимосвязан. Оказалось, что МПД может успешно и совсем без каких-либо изменений своей структуры делать еще одну полезную работу, выполняя при этом, конечно, и свои прямые обязанности.

Дело в том, что с самого начала своего развития теория кодирования позволила понять неравноценность передаваемых сообщений, их различную вероятность и необходимость. А кроме того, передавая информацию по дорогим каналам связи, следовало заботиться не только о ее достоверности, но и о том, чтобы сами сообщения были по возможности короче. Всем этим занимается теория кодирования источника со своими различными ветвями и направлениями. Среди них развиваются и так называемые методы кодирования с заданным критерием качества, когда требуются сжатие, передача и восстановление до прежней длины передаваемой последовательности, но после восстановления в ней допустим некоторый небольшой уровень ошибок. Чрезвычайно похоже на кодирование в канале, не правда ли?

Но какие сообщения можно сжимать? Многие, в том числе и бернуллиевские последовательности, т.е. такие, в которых вероятности появления нулей и единичек очень неодинаковы, причем они появляются независимо.

«Да это же модель ДСК!» – скажете вы. Да, правильно. Вот МПД и пригодился для одновременного сжатия бернуллиевского потока от источника информации и одновременного помехоустойчивого кодирования канала. Для этого берется высокоскоростной код с $R=K/(K+1)$ и предполагается, что передается нулевое кодовое слово, о чем декодер не знает. При этом, конечно, характеристики декодера не меняются. Но тогда до-

пустим далее, что ошибки в информационных символах кода заменяются единичками бернуллиевского потока на входе, т.е. кодер приемника совсем не нужен, а тот кодер, который находится в приемнике при обычной работе декодера, теперь переносится в приемник. И в этот момент становится ясно, что передача информационного потока кода в такой системе не нужна, так как вся полезная информация для декодирования находится в векторе синдрома (это мы уже знаем), а он в такой системе формируется передатчиком с помощью кодера, который перенесен из приемника. При этом закодированная последовательность направляется в канал, где остается, как и ранее, влияние его ошибок на проверочные символы, а затем МПД восстанавливает позиции единичек источника (это то же самое, что и ошибки в информационных символах при обычном режиме). Но информационная-то последовательность при таком подходе к кодированию не передавалась! Значит, произошло сжатие информации в K раз, поскольку в канал был фактически направлен синдром кода, если воспользоваться терминологией теории корректирующих кодов.

Наверное, такого краткого описания дополнительных свойств МПД вполне достаточно. Особенно привлекательно использование МПД в подобных системах сжатия и передачи еще и потому, что традиционные методы, приводящие к редким ошибкам восстановления, оказываются иногда весьма чувствительными именно к ошибкам передачи по каналу, искажая порой большие фрагменты данных после обнаруженной ошибки канала. Иначе говоря, при кодировании источника закодированные данные оказываются «хрупкими». Это достаточно хорошо известная проблема в теории и техники кодирования источника. А МПД, предназначенный именно для восстановления (конечно, с ограниченной точностью!) сообщений как раз в таких условиях, вполне нормально справляется и с этой задачей.

23. Характеристики декодирования

Характеристики МПД как при самостоятельном использовании, так и в сложных кодовых конструкциях сопоставимы с возможностями АВ и различных каскадных схем. На следующем рис. 21 представлена в традиционной форме зависимость эффективности основных алгоритмов декодирования для скорости $R=1/2$ вблизи уровня вероятности ошибки на бит $P_b(e) \approx 10^{-5}$ как функции от требуемых значений отношений средней энергии на бит к плотности мощности шума E_b/N_0 .

Возможности декодеров, реализующих алгоритм Витерби (VA) представлены для различных длин кодирующего регистра $K=7, 11, 15$ и 20 соответствующими графиками ВАК. Как следует из их вида, экспоненциально растущая с увеличением длины K сложность декодирования для этих декодеров не влечет за собой быстрого роста энергетической эффективности. Например, VA с $K=20$ более, чем в $2 \cdot 10^4$ раз сложнее, чем стандартный VA с $K=7$. Но его характеристики еще далеки от приведенной на рисунке вертикали при $E_b/N_0=0,2$ дБ, соответствующей пропускной способности двоичного гауссовского канала с когерентной ФМ, равной $C=1/2$.

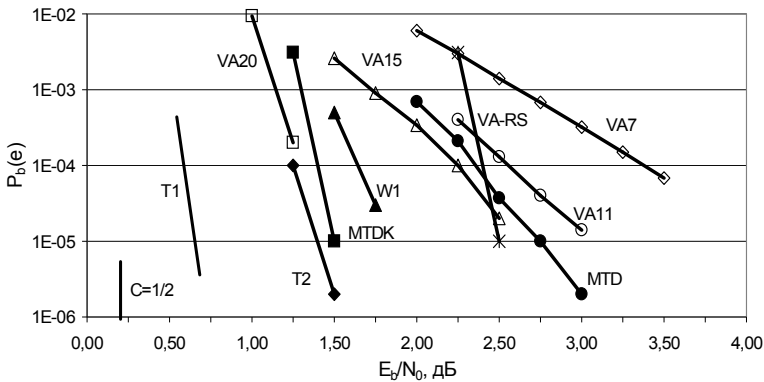


Рис. 21. Эффективность основных алгоритмов декодирования в гауссовском канале

Энергетика каскадной схемы для АВ с кодом Рида-Соломона дается графиком VA-RS. Лучшему турбо коду с $R=1/2$ и длиной кодового блока $n=130\ 000$ битов соответствует график T1, а турбо коду длины $n=3570$ – график T2.

Возможности простейшего МПД в гауссовском канале представлены графиком МТД для сверточного кода с задержкой решения $L\sim 8\ 000$ кодовых символов, а каскадный МПД с кодами контроля по четности (вместо кодов РС для других каскадных схем – один сумматор по mod2!!!): МТДК с $L\sim 100\ 000$. Свойства МПД допускают некоторое дополнительное уменьшение задержек по сравнению с указанными выше.

Как видим, возможности каскадных вариантов реализации МПД оказались действительно очень высокими.

24. Сложность реализации МПД

Главным достоинством МПД является крайне низкая сложность декодирования, состоящая, как и в случае обычного ПД, в суммировании взвешенных проверок, сравнении с порогом и изменении декодируемого символа и проверок, если этот порог превышен. Число итераций декодирования I в этом случае не более 30, а общая сложность декодирования с помощью МПД оценивается для $d<25$ как $N_1\sim(d+2)(I+4)$. Если при тех же условиях возможно ухудшение характеристик МПД примерно на 0,1 дБ по энергетике, что обычно вполне допустимо, то вычисления МПД еще более упрощаются: $N_2\sim 4d+3I$.

Подчеркнем, что формально определяемая сложность как число операций у МПД примерно на 2 порядка (100 раз!) меньше, чем у турбо подобных кодов с сопоставимой энергетической эффективностью. Существенно, что при таких оценках особенная сложность части операций, выполняемых при декодировании турбо кодов, не учитывается. Поэтому реальная разница в сложности реализации МПД и турбо подобных кодов еще более значительна.

Естественно, что усилиями большого числа специалистов, занимающихся этими кодами, многие из турбо подобных алгоритмов за 10 лет их интенсивного исследования были усовер-

шенствованы и упрощены. Но при этом обычно несколько снижается и энергетическая эффективность новых версий декодеров для этих кодов. Текущая сложность декодеров этого класса для хороших энергетических параметров составляет тысячи операций на каждый декодируемый бит данных.

Подчеркнем еще раз, что существенная разница в эффективности МПД и других методов заключается в том, что этот метод при линейной от длины кода сложности декодирования обеспечивает при правильном проектировании практически оптимальное декодирование длинных кодов.

25. Особенности проектирования МПД декодеров

При правильном проектировании аппаратных версий МПД функция суммирования проверок легко может быть реализована аппаратно таким образом, что она будет полностью эквивалентной простому одноклапчатому сумматору. В этом случае окажется, что преимущество МПД перед другими методами будет определяться скоростью движения данных по регистрам сдвига декодера. При скорости сдвига w_0 и кодовой скорости $R=k_0/n_0$, $k_0 \approx 3-20$, быстродействие как максимальная скорость декодирования данных составит $W=w_0k_0$, что составляет относительно других методов уже более трех десятичных порядков по производительности. Это показывает, что МПД по крайней мере в своем сверточном варианте, допускающем полное распараллеливание операций, обладает абсолютным преимуществом перед всеми остальными методами. На каждом такте работы аппаратные МПД устройства декодируют по несколько битов информационного потока как в блоковом, так и в сверточном вариантах своей реализации. Несомненно, что никакие другие алгоритмы декодирования никогда не приблизятся к этому уровню быстродействия.

Отметим, что возможности МПД по выбору своих параметров оказываются весьма широкими. При необходимости максимально быстрой аппаратной реализации нужно создавать сверточный декодер с достаточно большой задержкой L , выраженной в кодовых символах, но увеличить и количество итера-

ций декодирования I . Если реализация программная, то следует выбрать версию алгоритма с минимальным числом операций N и при максимально полном использовании памяти декодера M . Если необходима маленькая задержка, то целесообразно увеличить число операций N и итераций L при максимально возможной скорости работы регистров W . Возможность организации очень широкого обмена между значениями и уровнем основных параметров МПД можно компактно выразить в виде $W \leftrightarrow N \leftrightarrow I \leftrightarrow M \leftrightarrow L$. Это создает максимально разнообразные возможности для разработчиков программного обеспечения и аппаратуры высокоэффективного помехоустойчивого кодирования с практически сколь угодно высокой производительностью.

26. Сравнение с АВ

При сопоставлении МПД с АВ также оказалось, что при задержках принятия решения, всего в 3–5 раз больших, чем требуется для АВ, МПД обеспечивает те же характеристики по энергетике для всех длин кодирующего регистра, не превышающих значения $K=14$. Более того, во многих случаях, например, при допустимости каскадирования хотя бы с простейшими кодами контроля по четности, энергетические характеристики МПД могут быть еще существенно улучшены. Отметим при этом, что число операций в МПД останется очень небольшим, тогда как АВ – переборный (с экспоненциально растущей от длины кода сложностью!) алгоритм.

27. Оптимизация схемы МПД

Переходя к выводам по возможностям алгоритмов МПД, подчеркнем, что все чрезвычайно высокие параметры декодирования этими методами обеспечиваются применением предельно простой, но результативной и очень мощной процедуры оптимизации функционала на базе процедур мажоритарного декодирования, чем достигается практически оптимальное декодирование длинных кодовых конструкций, доступное ранее только

для коротких кодов и алгоритма Витерби. При этом важной оказывается не только простая по своей сути модификация мажоритарного декодирования, но и обязательный выбор весьма специфических кодов, например, в отдельных случаях из класса самоортогональных, которые обеспечивают высокие характеристики при использовании МПД благодаря очень малой их подверженности эффекту размножения ошибок декодирования.

Но не менее существенным моментом для успешного развития методов МПД является так же то обстоятельство, что при создании этих декодеров обязательным и очень эффективным этапом проектирования всегда является оптимизация всех параметров такого декодера: значений порогов, весов проверок, правил вычисления вспомогательных функций и надежности проверок. Число оптимизируемых параметров МПД может иногда достигать нескольких сотен. Поэтому при создании МПД последний этап может потребовать иногда до 99% процессорного времени специального автоматизированного комплекса средств создания этих высокоэффективных декодеров. Принципиально важно, что после завершения всех этапов работ фазы проектирования МПД по-прежнему остается простейшей схемой обработки принятых потоков данных с очень высоким быстродействием. Но подбор наиболее точно соответствующих конкретному кодеку индивидуальных значений его параметров позволяет получить весьма значительный дополнительный прирост энергетического выигрыша без излишних аппаратных и вычислительных затрат. Разумеется, два последних обстоятельства не могли быть учтены в те уже очень давние годы, когда были опубликованы некоторые совершенно справедливые результаты по низкой эффективности попыток простого повторного декодирования на основе обычных мажоритарных схем.

Заключение

Важнейшим достоинством МПД оказывается легкость обмена в очень широких пределах сложности аппаратуры на ее быстродействие, память и задержку принятия решения. Именно возможность простой вариации этих тесно взаимосвязанных

технологических параметров разработки, наряду с очень высокими уровнями ЭВК, делает его универсальным и легко адаптируемым к любым условиям применения в аппаратуре как широкого, так и специального применения. Он просто стыкуется с различными модемами, и легко адаптируем к разным типам сигналов. Более того, оказалось, что его свойства столь уникальны, что позволяют особым образом взаимодействовать с некоторыми сигналами особой структуры, еще более повышающими достижимый с его помощью ЭВК.

Последнее слово в соревновании алгоритмов должна сказать, конечно, технология. Если в каком-то технологическом базисе, например, некоторая процедура окажется 2,6 раза более дешевой при определенной степени тиражирования, то только ее там и надо применять.

Ориентировочно МПД от 30 до 200 раз проще сопоставимых с ним по ЭВК алгоритмов, если от него не требуются самые предельные уровни ЭВК. Но и для других алгоритмов это тоже очень трудная, если вообще достижимая, цель.

Если снизить требования по ЭВК примерно на децибел (что все равно очень трудно для абсолютного большинства других алгоритмов), то МПД по сложности будет просто соответствовать обычному ПД, проще которого вообще ничего создать нельзя. Для создания же других декодеров с такой же эффективностью придется очень напряженно поработать. Так что широкое применение МПД в новых системах связи неизбежно. И чем активнее он будет внедряться, тем быстрее все заботы о достоверности обмена в любых сколь угодно сложных компьютерных сетях уйдут в историю, так как МПД, если это необходимо, всегда сможет работать вблизи теоретических границ по энергетике канала. Несомненно, быстрое действие аппаратных версий МПД является максимально возможным при заданной скорости регистров сдвига декодеров.

Ясно, что платой за субоптимальность нашего декодера является большая работа по оптимизации элементов проектируемых МПД: весов проверок, уровней порогов, других связей и параметров. Как и самих кодов, наконец. Неточные настройки этих параметров, изменение хоть одной из проверок по позиции,

весу или другим параметрам в коде длиной тысячи и более символов может просто разрушить весь процесс сходимости к оптимальному решению. То есть алгоритм МПД, как и все декодеры, успешно работающие в условиях большого шума канала, достаточно хрупок, если можно так сказать, по отношению к процессу проектирования. Но если он построен правильно, то потом будет работать отменно, потому что идея метода естественна и понятна, а схема – элементарна.

Все, что для вас оказалось непонятным или сложным, приглашаем посмотреть на нашем сайте www.mtdbest.iki.rssi.ru, где регулярно обновляются большинство материалов, относящихся к методу МПД. Там же можно получить и переписать на свой компьютер специальный мультфильм, который оказывается хорошей иллюстрацией возможностей алгоритма МПД в классическом канале ДСК. Для правильного понимания возможностей этой демопрограммы прочитайте внимательно сопровождающее этот компьютерный мультик описание работы с ним и комментарии к его характеристикам. Оно прекрасно поможет вам в понимании возможностей многопороговых алгоритмов.

Вы можете также присылать свои пожелания и отзывы о представленных в этой книге материалах по e-mail: zolasd@yandex.ru или g_ovechkin@mail.ru.

Некоторые публикации по теме многопорогового декодирования

1. Самойленко С.И., Давыдов А.А., Золотарев В.В., Третьякова Е.И. Вычислительные сети. Адаптивность, помехоустойчивость, надежность. – М.: Наука, 1981. – 285 с.

2. Брауде-Золотарев Ю.М., Золотарев В.В. Характеристики аппаратуры кодирования на матричных БИС // Труды НИИР, 1986. – № 2.

3. Золотарев В.В. Коды и кодирование. – «Знание». Радиоэлектроника и связь. 1990. – № 9.

4. Золотарев В.В. Использование помехоустойчивого кодирования в технике связи // Электросвязь. – 1990. – № 7. – С. 7–10.

5. Золотарев В.В. Простые методы исправления ошибок в каналах с большим уровнем шума // Радиотехника. – 1991. – № 10.
6. Золотарев В.В. Устройство для декодирования линейных кодов. – Патент на изобретение № 2035123. – 1995.
7. Золотарев В.В., Петровский И.Б. Об эффективности кодирования в спутниковых каналах // 54-я Научная сессия, посвященная Дню радио. Тезисы докладов. – М., 1999. – С. 256–258.
8. Золотарев В.В.. Эффективность лучших методов кодирования для спутниковых каналов // Труды 56-й Научной сессии РНТОРЭС им. А.С. Попова. – М., 2001. – Том 2. – С. 379–362.
9. Золотарев В.В., Овечкин Г.В. Сравнение турбоподобных кодов и многопорогового алгоритма декодирования // 10-я междунар. науч.-техн. конф. «Проблемы передачи и обработки информации в сетях и системах телекоммуникаций». – Рязань, 2001. – С. 193–195.
10. Золотарев В.В. Реальный энергетический выигрыш кодирования для спутниковых каналов // 4-я междунар. конф. «Спутниковая связь–2000». – М., 2000. – Т. 2. – С. 20–25.
11. Золотарев В.В., Овечкин Г.В. Модификация многопорогового алгоритма декодирования // 3-я междунар. науч.-техн. конф. «Космонавтика. Радиоэлектроника. Геоинформатика». – Рязань, 2000. – С. 146–148.
12. Золотарев В.В., Овечкин Г.В. Эффективные алгоритмы помехоустойчивого кодирования для цифровых систем связи // Электросвязь. – 2003. – № 9. – С. 12–15.
13. Золотарев В.В. Характеристики каскадирования многопороговых декодеров для спутниковых каналов связи // 5-я междунар. конф. «Цифровая обработка сигнала и ее применение». – 2003. – С. 353–356.
14. Золотарев В.В., Овечкин Г.В. Использование многопорогового декодера в каскадных схемах // Вестник РГРТА. – 2003. – Вып. 11. – С. 112–115.
15. Zolotarev V.V. The Multithreshold Decoder Performance in Gaussian Channels // In Proc.: 7-th International Symposium on Communication Theory and applications (13–18 July 2003, St. Martin's College, Ambleside, 7-th ISCTA'03, UK). – P. 18–22.

См. также презентацию «Обзор современных методов помехоустойчивого кодирования» на сайте Института космических исследований РАН www.iki.rssi.ru в разделе семинаров по механике и информатике от 27.11.2002 г.

Более полное представление об МПД можно получить на страничке ИКИ РАН www.mtdbest.iki.rssi.ru, полностью посвященной многопороговым декодерам. Там же находятся новые объявления, предложения и другая полезная информация.

Для контакта:

Золотарев В.В.

ИКИ РАН: тел. (095) 333-23-56;

НИИРадио: (095) 261-03-27, 261-54-44;

дом. тел.: (095) 573-51-32;

моб.: 8-916-518-86-28;

e-mail: zolotasd@yandex.ru, (или zolotarev@iki.rssi.ru как резервный).

Овечкин Г.В.

РГРТА: тел. (0912) 72-00-56;

моб.: 8-910-644-51-46;

e-mail: g_ovechkin@mail.ru или fvt@rgrta.ryazan.ru.

ЭФФЕКТИВНЫЕ АЛГОРИТМЫ ПОМЕХОУСТОЙЧИВОГО КОДИРОВАНИЯ ДЛЯ ЦИФРОВЫХ СИСТЕМ СВЯЗИ

В.В. Золотарев, Г.В. Овечкин

Передача информации по каналам связи существенно усложняется из-за помех и искажений в канале. Эффективным средством повышения достоверности передаваемой информации является помехоустойчивое кодирование. В статье производится сравнение основных алгоритмов декодирования помехоустойчивых кодов, сделанное по материалам научно-технических конференций и публикациям по теории кодирования.

Критерии эффективности кодирования. За последние 12 лет, прошедшие с момента публикации обзорной статьи [1], теория помехоустойчивого кодирования продвинулась далеко вперед. Появились мощные коды и методы их декодирования, обеспечивающие работу систем связи вблизи пропускной способности канала. Наиболее общим показателем для качественной оценки методов декодирования является энергетический выигрыш кодирования (ЭВК), показывающий снижение энергии, необходимой для передачи одного бита данных (при некоторой выбранной средней вероятности ошибки на бит $P_b(e)$) в случае использования тех или иных алгоритмов кодирования и декодирования, по сравнению со случаем, когда кодирования нет.

Зарубежные специалисты более 20 лет назад оценивали каждый 1 дБ ЭВК в миллионы долларов [2]. Сейчас ценность ЭВК возросла еще больше, поскольку появилась возможность уменьшать размеры очень дорогих антенн или повышать дальность связи, увеличивать скорость передачи или снижать необходимую мощность передатчика, улучшать другие важные свойства современных систем связи. Заметим, что их стоимость растет все быстрее и уже совершенно несопоставима с теми затратами, которые были необходимы несколько десятилетий назад. Кроме того, существенно возрастают взаимные помехи

приему между различными системами, появляется все больше экологических ограничений на мощность передатчиков, растет потребность в резком увеличении скоростей обмена данными между сетями, а также в значительном росте их достоверности.

И в дальнейшем, в условиях глобализации экономики и взрывной информатизации общества эти тенденции будут только усиливаться. Отсюда можно сделать очевидный вывод, что возрастание на несколько порядков (!) экономической (как и экологической, и вообще, системной) ценности ЭВК на базе методов помехоустойчивого кодирования становится чрезвычайно актуальным и требует внедрения теоретических достижений в области помехоустойчивого кодирования при создании компьютерных систем связи.

Характеристики декодеров. На рис. 22 представлена зависимость вероятности ошибки на бит $P_b(e)$ на выходе декодера от уровня шума E_b/N_0 в двоичном симметричном канале без памяти (ДСК). Ошибки возникают при работе жесткого модема с двоичной фазовой модуляцией в канале с аддитивным белым гауссовским шумом, когда в декодере не используются оценки достоверности принятых символов. Такая модель канала достаточно точно описывает реальные спутниковые и некоторые другие типы каналов.

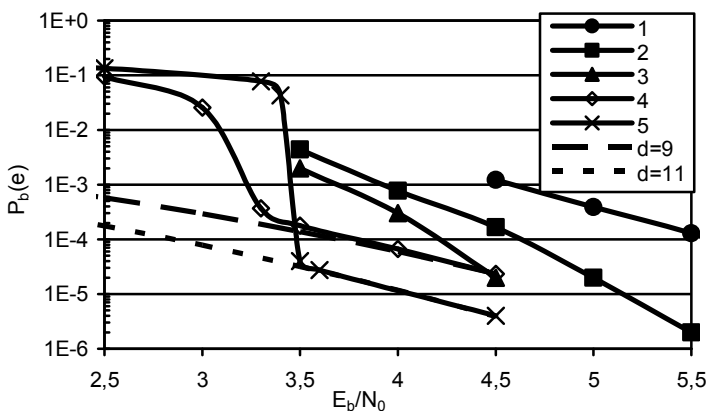


Рис. 22

Кривые 1, 2, 3 соответствуют широко используемому алгоритму Витерби (AB) [3, 4] для кодовой скорости $R=1/2$ и длины кодирующего регистра $k=7, 11$ и 15 . Данный метод является оптимальным, однако его сложность растет экспоненциально с ростом k и поэтому на практике декодеры с $k>9$ не используются. В результате возникает необходимость поиска более простых (с точки зрения практической реализации) декодеров, обеспечивающих в то же время близкое к оптимальному декодирование.

Многopогpовые алгоритмы. Один из простейших алгоритмов коррекции ошибок – пороговый декодер (ПД) Мессис [5], принимающий решение о значении каждого декодируемого символа на основе обычного «голосования» проверок. Схема порогового декодера сверточного самоортогонального кода (СОК) с кодовым расстоянием $d=5$, кодовой скоростью $R=1/2$ и длиной кодового ограничения $n_d=14$ представлена на рис. 23. Он состоит из двух двоичных регистров, нескольких полусумматоров (сумматоров по mod 2) и порогового элемента T , который просто суммирует двоичные проверки декодируемого символа и сравнивает их с порогом, изменяя этот символ и все относящиеся к нему проверки, если сумма проверок превысила некоторое пороговое значение. Но данный метод имеет невысокую корректирующую способность, из-за чего редко применяется в системах передачи, данных.

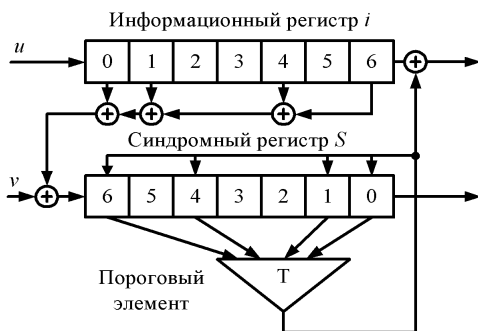


Рис. 23

Значительно лучшими характеристиками обладают многопороговые декодеры (МПД) [6], являющиеся модификацией представленного выше ПД. МПД в процессе контроля принятых кодовых последовательностей многократно исправляют информационные символы пришедшего из канала сообщения. Эти декодеры обладают важнейшим строго доказанным свойством – сходимостью к решению оптимального декодера (ОД), сохраняя линейную от длины кода сложность реализации.

Схема МПД сверточного СОК с $d=5$, $n_A=14$ и двумя итерациями декодирования представлена на рис. 24. При большем числе итераций, обычно необходимом для работы в области больших шумов канала, последующие итерации полностью аналогичны второй. Как видно из представленной схемы, каждая итерация МПД отличается от обычного ПД только наличием «разностного» регистра, в котором отмечаются измененные пороговым элементом (ПЭ) информационные символы. Существенно, что решения ПЭ из разностного регистра затем используются другим ПЭ на следующей итерации декодирования. На каждой итерации МПД при декодировании информационного символа i_k на ПЭ – единственном активном элементе декодера

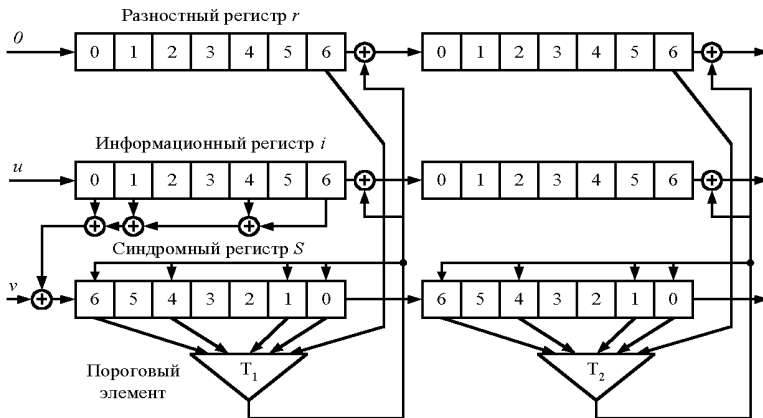


Рис. 24

(при использовании жесткого модема в ДСК) выполняются следующие операции.

1. Вычисляется сумма проверок (каждая из которых для случая ДСК равна 0 или 1), т.е. функция

$$L_k = \sum_{m=1}^J S_{g_m} + r_k,$$

где $J=d-1$ – количество проверок (ненулевых элементов g_m порождающего полинома кода G); r_k – символ разностного регистра, относящийся к декодируемому символу i_k также равный 0 или 1; S_m – m -й элемент синдромного регистра, входящий в множество проверок относительно декодируемого символа i_k .

2. Если $L_k > T$, где $T \geq (d-1)/2$ – значение порога порогового элемента, то информационный символ i_k , все связанные с ним проверки $\{S_{g_m}\}_{m=1, J}$ и символ r_k инвертируются.

3. Переход к декодированию следующего символа (п. 1).

Заметим, что при применении мягкого модема, когда в гауссовском канале оценивается надежность приема каждого двоичного символа, в МПД выполняются те же операции, но проверки суммируются уже с весами, определяющими надежность оценок принятых из канала битов. Пороги T для декодируемых символов будут переменными, а не фиксированными, как в случае ДСК.

Анализ характеристик МПД. Кривая 4 на рис. 22 соответствует работе МПД сверточного СОК с $R=1/2$, минимальным кодовым расстоянием $d=9$, длиной кодового ограничения $n_A=4168$ бит и $I=15$ итерациях декодирования [6]. Возможности МПД с $R=1/2$, $d=11$ и $I=20$ представлены на рис. 22 кривой 5. Пунктиром на рисунке показаны вероятности ошибочного декодирования кода с $d=9$ и $d=11$ оптимальным декодером. Как следует из графиков, МПД (для достаточно длинных кодов) оказывается лучше даже заведомо нереализуемого оптимального АВ с $k=15$ и более. Отметим, что с ростом длины кодового ограничения n_A и увеличением числа итераций характеристики МПД быстрее сходятся к характеристикам оптимального декодера.

На рис. 25 показаны характеристики тех же алгоритмов декодирования при использовании мягкого модема, оценивающего

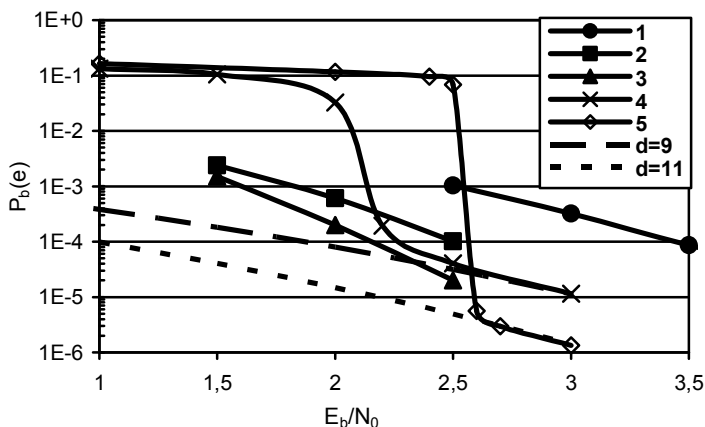


Рис. 25

достоверность принимаемых двоичных символов с квантованием выхода на $Q=16$ уровней для вышеперечисленных кодов. Из сопоставления рис. 22 и 25 видно, что переход к мягкому модему при использовании АВ позволяет получить дополнительный ЭВК порядка 2 дБ, а при использовании МПД – 1,4...1,7 дБ. Но и в этом случае МПД, сохраняя простоту обычного ПД, существенно превосходит АВ по эффективности. Основные соотношения между этими алгоритмами имеют место и при других кодовых скоростях.

Приведенные выше данные показывают, что доступные для реализации оптимальные АВ уступают МПД, в котором могут легко декодироваться длинные коды, в широком диапазоне кодовых скоростей при использовании как жесткого, так и мягкого модемов. При этом МПД выполняет только простейшие операции сложения и сравнения небольших целых чисел, что обуславливает его крайнюю простоту при всех вариантах программной или аппаратной реализации.

Существенное улучшение характеристик декодирования обеспечивают каскадные методы кодирования [7] при меньшей сложности реализации по сравнению с некаскадными. При этом

исходные данные кодируются двумя и более кодами, которые затем определенным образом декодируются соответствующими этим кодам методами.

Эффективность турбокодов. Рассмотрим открытые в 1993 г. *турбокоды* [8–10] – коды с параллельным каскадированием. Они создаются с помощью двух и более систематических кодеров, связанных между собой перемежителями. Название «турбо» отражает свойства используемого при декодировании итеративного алгоритма: информация с выхода одной итерации декодирования является входной для следующей итерации. В декодерах составляющих кодов здесь обычно применяют оптимальный алгоритм декодирования по максимуму апостериорной вероятности (MAP алгоритм) или его упрощенный вариант – max-log-MAP (MLM).

На рис. 26 показаны характеристики турбокодов: кривая 1 – $R=1/2$, $K=65536$, MAP; кривая 2 – $R=1/2$, $K=1000$, MAP; кривая 3 – $R=1/2$, $K=1000$, MLM. Здесь K – длина информационного блока. При получении данных графиков выполнялось до 20 итераций декодирования. Коды показывают очень хорошие харак-

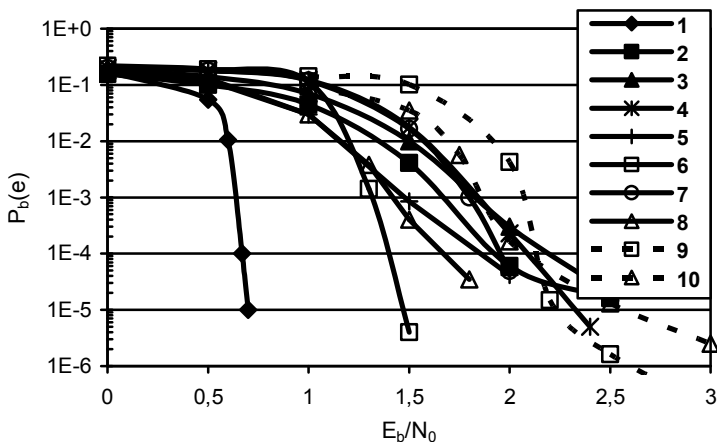


Рис. 26

теристики при большом шуме в канале связи. С их помощью достигается $P_b(e)=10^{-5}$ при $E_b/N_0=0,7$ дБ, что всего лишь на 0,5 дБ выше теоретически достижимой границы. Однако при снижении уровня шума наблюдается эффект «насыщения» вероятности ошибки, возникающий из-за небольшого кодового расстояния приведенных кодов.

Сложность декодеров турбо кодов очень велика. Например, турбо код, основанный на MAP декодере, выполняет порядка $12N$ FLOP/IB/Iter, а на MLM – $30N$ AEO/IB/Iter. Здесь N – число возможных состояний кодера, FLOP/IB/Iter – количество операций с вещественными числами (включая умножение и сложение), приходящихся один на информационный бит на каждой итерации, а AEO/IB/Iter – количество сложений, вычитаний и сравнений целых чисел.

Другим видом каскадных кодов являются *последовательные каскадные коды* [11], для декодирования которых также применяется итеративная схема. Их возможности также показаны на рис. 26: кривая 4 – $R=1/2$, $K=1000$, MLM; кривая 5 – $R=1/2$, $K=1000$, MAP. Коды обладают большим кодовым расстоянием, и поэтому показывают лучшие характеристики при среднем и низком уровне шума.

Плетеные коды [12–15] также являются последовательными каскадными кодами. В кодере данного кода ряд внешних декодеров и внутренний как бы переплетаются. На рис. 26 представлены возможности плетеных кодов: кривая 6 – $R=1/2$, $K=10000$, MLM; кривая 7 – $R=1/2$, $K=1000$, MLM; кривая 8 – $R=1/2$, $K=1000$, MAP. Коды оказываются хуже турбо при большом уровне шума и лучше – при среднем и низком.

Из рис. 26 видно, что в случае работы при умеренных шумах МПД, характеризующийся высоким быстродействием и простотой (МПД с $d=9$ выполняет порядка 12 AEO/IB/Iter), свойственной всем мажоритарным алгоритмам, ничем не уступает рассмотренным каскадным схемам. Кроме того, при таком шуме ошибки на выходе МПД в основном одиночные [6], что позволяет легко применять его в очень простых каскадных кодах даже без использования дополнительных перемежителей. Кривые 9 и 10 на рис. 26 отражают возможности каскадных

схем на основе МПД и кода с контролем четности (ККЧ): 9 – МПД с $R=1/2$, $d=9$, $n_A=4168$ и ККЧ с $n=50$; 10 – МПД с $R=1/2$, $d=7$, $n_A=3052$ и ККЧ с $n=50$. Использование данной каскадной схемы позволяет уменьшить вероятность ошибки декодирования в 10–100 раз по сравнению с простым МПД.

Заметим, что число операций в МПД (и без того очень большое) можно еще сократить – в некоторых случаях примерно в 4 раза без потери эффективности декодирования, что значительно увеличивает скорость работы алгоритма.

Сравнение алгоритмов декодирования. Завершим сопоставление алгоритмов рассмотрением обеспечиваемого ими ЭВК. На рис. 27 отражена энергетическая эффективность алгоритмов декодирования при работе с жестким модемом: кривая 1 – АВ с $k=7$, $R=1/2$; кривая 2 – МПД с $R=1/2$, $d=9$; кривая 3 – МПД с $R=1/2$, $d=11$; кривая 4 – каскадный: ККЧ с $n=50$ и МПД с $R=1/2$, $d=9$. Рис. 28 показывает ЭВК алгоритмов декодирования при работе с мягким модемом: кривая 1 – АВ с $k=7$, $R=1/2$; кривая 2 – турбо с $R=1/2$, $K=65536$, MAP; кривая 3 – турбо с $R=1/2$, $K=1000$, MLM; кривая 4 – МПД с $R=1/2$, $d=9$. Каскадные схемы на основе МПД с $d=9$ и $d=7$ представлены здесь кривыми 5 и 6. Как видно, МПД и каскадные схемы на его основе при малых $P_b(e)$ незначительно уступают турбокодам.

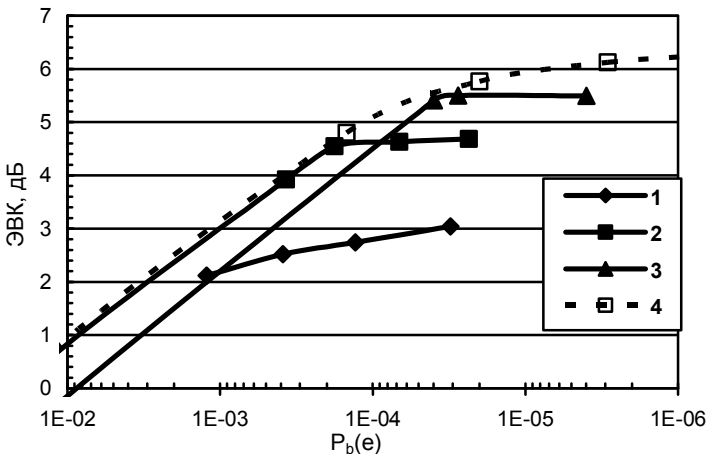


Рис. 27

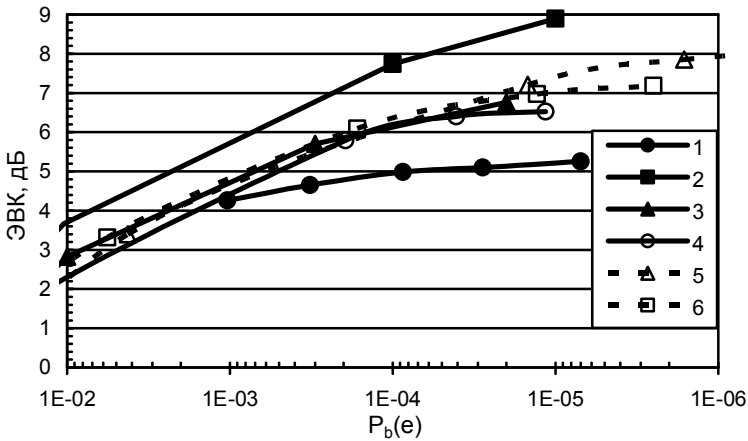


Рис. 28

Приведенные результаты относились к двоичным кодам и системам с двоичной фазовой модуляцией. Но на основе важнейшего свойства сходимости решения МПД к оптимальному при всех изменениях декодируемых символов [6] реализованы недвоичные МПД, обеспечивающие характеристики, близкие или несколько более высокие, чем у кодов Рида–Соломона. Аналогично, высокие характеристики МПД, обеспечиваемые ими в каналах со стираниями и в системах с многопозиционной модуляцией делают их универсальным средством простого достижения высокого уровня помехозащищенности сообщений в системах широкого спектра применения. При этом сохраняется такая же сложность реализации, как и в простейшем пороговом декодере.

З а к л ю ч е н и е . В настоящее время возможности технологии позволяют даже для высокоскоростных каналов создавать аппаратно-программные версии декодеров, которые будут успевать выполнять достаточно большое число операций с каждым декодируемым символом. Это дает возможность строить все более сложные кодовые конструкции.

Однако алгоритмы, нерационально использующие вычислительные ресурсы, все же значительно проигрывают гораздо более простым, но решающим задачу декодирования более эффективно.

Несомненно, что проблемы сложности реализации кодирования сохранятся в обозримом будущем, а в связи с ростом скоростей обмена информацией требования более простой реализации декодеров будут все актуальнее. Самыми дешевыми при всех вариантах реализации окажутся те алгоритмы, которые выполняют только очень простые, однородные и быстрые операции. Наиболее полно этим требованием удовлетворяют МПД. А соответствие его возможностей характеристикам самых сложных алгоритмов делает МПД еще более привлекательным.

Конечно, при необходимости достижения уровней энергетического выигрыша, сравнимого с лучшими турбокодами, размеры памяти МПД и число выполняемых им операций сложения заметно возрастают. Но даже в этом случае сложность их реализации останется, видимо, существенно меньшей. Однако необходимость реализации декодеров для аппаратуры связи с энергетикой, всего на несколько десятых децибел, т.е. всего на 2–4% меньшей предельно возможных значений, требует более глубокой технологической проработки и обоснования. Видимо, эти вопросы станут действительно актуальными не ранее, чем через несколько лет.

Таким образом, в истекший после публикации [1] период были созданы условия для достижения уровней энергетики, весьма близких к теоретическим пределам для каналов космической и спутниковой связи. Высокие характеристики могут обеспечить несколько методов. Однако их сложность при сопоставимых уровнях ЭВК показывает, что алгоритмы на базе МПД близки по числу операций к простейшему декодеру порогового типа. Поэтому в большинстве случаев МПД являются наиболее предпочтительным алгоритмом декодирования в высокоскоростных системах связи.

Список литературы

1. Золотарев В.В. Использование помехоустойчивого кодирования в технике связи // Электросвязь. – 1990. – № 7. – С. 7–10.
2. Берлекэмп Э.Р. Техника кодирования с исправлением ошибок // ТИИЭР. – 1980. – Т. 68, № 5. – С. 24–58.
3. Кларк Дж., Кейн Дж. Кодирование с исправлением ошибок в системах цифровой связи: Пер. с англ. под ред. Б.С. Цыбакова. – М.: Радио и связь, 1987. – 392 с.
4. Золотарев В.В. Реальный энергетический выигрыш кодирования для спутниковых каналов // 4-я Междунар. конф. «Спутниковая связь – ICSC-2000». – М.: МЦНТИ, 2000. – Т. 2. – С. 20–25.
5. Месси Дж. Пороговое декодирование: Пер. с англ. под ред. Ю.Л. Сагаловича. – М.: Мир, 1966. – 208 с.
6. Самойленко С.И., Давыдов А.А., Золотарев В.В., Третьякова Е.И. Вычислительные сети. – М.: Наука, 1981.
7. Форни Д. Каскадные коды: Пер. с англ. под ред. С.И. Самойленко. – М.: Мир, 1970. – 208 с.
8. Berrou C., Glavieux A., Thitimajshima P. Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes // Proceeding of ICC'93, Geneva, Switzerland, May 1993. – P. 1064–1070.
9. Barbulescu S.A. Iterative decoding of turbo codes and other concatenated codes. Ph.D. dissertation. – Feb. 1996.
10. Andrews K., Berner J., Chen V. at all. Turbo-decoder implementation for the deep space network // IPN Progress Report 42–148. – Feb. 15, 2002.
11. Benedetto S., Montorsi G., Divsalar D., and Pollara F. Serial concatenation of interleaved codes: Performance analysis, design and iterative decoding // JPL TDA Progress Report. – August 1996. – V. 42–126,
12. Höst S., Johannesson R., Zyablov V. A first encounter with binary woven convolutional codes // In Proc. International Symposium on Communication Theory and Applications, Lake District, UK. – July 1997. – P. 13–18.

13. Freudenberger J. Untersuchung von woven-codes. Ph.D. dissertation. – Jan. 1999.
14. Зяблов В.В., Йоханнессон Р., Скопинцев О.Д., Хест С. Асимптотические дистанционные свойства двоичных плетеных сверточных кодов // Проблемы передачи информации. – 1999. – Т. 35, вып. 4. – С. 29–46.
15. Freudenberger J, Shavgulidze S., Zyablov V., Bossert M. Woven codes with outer warp: variations, design and distance properties // Journal on Selected Areas in Communications issue on The Turbo Principle: From Theory to Practice, 2001.

Содержание

1.	Кругом помехи и шумы	2
2.	Что делает теория кодирования?	3
3.	Если самый-самый хороший	6
4.	Так много способов!	10
5.	Самый незатейливый	14
6.	Так что же можно сделать?	20
7.	Наши первые успехи	24
8.	Взгляд иной на эту тему	28
9.	Новый супердекодер?	36
10.	Так он совсем не оптимальный?	45
11.	Но какие характеристики – самые лучшие?	48
12.	Так какой же он, этот МПД?	50
13.	Как спасти идею?	51
14.	Так что же нас ограничивает? Размножение ошибок!	55
15.	Что умеют МПД	63
16.	Но еще важнее – экономика	69
17.	Что заказывают связисты	70
18.	А если еще лучше?	72
19.	Как дела у конкурентов?	76
20.	А если канал не двоичный?	77
21.	А если в эфире тесно.	82
22.	Кодируем источники тоже	84
23.	Характеристики декодирования	86
24.	Сложность реализации МПД	87
25.	Особенности проектирования МПД декодеров	88
26.	Сравнение с АВ	89
27.	Оптимизация схемы МПД	89
	Заключение	90
	Приложение. Статья из журнала «Электросвязь», 2003, № 9	95

Многопороговые декодеры – современное решение проблемы помехоустойчивого кодирования

ФГУП НИИР предлагает эффективные декодеры пятого поколения на базе самых последних теоретических и экспериментальных исследований метода многопорогового декодирования (МПД) блоковых и сверточных кодов. Впервые на новейших программных и аппаратных средствах достигнуты энергетические выигрыши кодирования 7 дБ и более при линейной, т.е. минимально возможной сложности реализации самого декодера и практически неограниченном быстродействии.

При сопоставимой эффективности кодирования программные версии МПД проще других методов в каналах с большим уровнем шума на 2 порядка.

Реальный выигрыш по скорости декодирования при аппаратной реализации еще более значителен. Для некоторых параметров систем кодирования МПД может быть почти в 1000 раз быстрее, чем аналогичные по эффективности турбо декодеры.

НИИР предлагает программные декодеры на базе МПД для компьютеров общего назначения со скоростями более 2 Мбит/с при отношении сигнал/шум канала менее $-0,25$ дБ.

МПД обеспечивает дополнительный энергетический выигрыш в 3–5 дБ и более по сравнению со стандартным алгоритмом Витерби.

Принимаем заказы на высокоскоростные аппаратные версии МПД.

Дополнительную постоянно обновляемую информацию об МПД алгоритмах можно также найти на специализированном сайте www.mtdbest.iki.rssi.ru.

Приглашаем специалистов в области связи на цикл лекций о методах МПД «Современные методы помехоустойчивого кодирования с характеристиками оптимального декодирования» ведущего специалиста НИИР д.т.н. В.В. Золотарева

Предлагаем организовать учебу специалистов по методам помехоустойчивого кодирования на вашем предприятии.

Для контакта:

НИИРрадио: (095) 261-03-27, (095) 261-54-44;

e-mail: zolotasd@yandex.ru, моб.: 8-916-518-86-28 (Золотарев В.В.) или

e-mail: g_ovechkin@mail.ru, моб.: 8-910-644-51-46 (Овечкин Г.В.)

Multithreshold decoders – are up-to-date solution for a noiseproof coding problem

FGUE NIIRadio suggests fifth generation of effective multithreshold decoders (MTD) at the latest theoretical and experimental research basis for block and convolutional codes. For the first time a code gain 7 dB and more is reached with linear complexity on the most modern program and hardware means. MTD has minimally possible hardware implementation of the decoder itself and practically unlimited throughput.

At comparable coding efficiency soft MTD versions are simpler than other methods in channels with the big noise level on 2 decimal orders (~100 times).

The real MTD advantage on decoding rate at hardware realization is even more significant. For some parameters of coding systems MTD can be more fast almost in 1000 times, than similar by efficiency turbo decoders.

NIIRadio suggests the software decoding systems on the MTD basis for general purpose computers with speeds more than 2 Mbit/s at a signal-to-noise ratio in a channel less than $-0,25$ dB.

MTD provides additional code gain 3–5 dB and even more with respect to standard Viterbi algorithm.

We are ready for hardware versions of MTD orders.

We are ready to international cooperation.

The full additional permanently upgraded information on MTD algorithms can be also found on our specialized site **www.mtdbest.iki.rssi.ru**.

We invite the specialists in the field of communication to a lecture cycle about MTD methods «Modern methods of noiseproof coding with performance of total searching algorithms», which one will be delivered by the leading NIIRadio specialist in the noiseproof coding Doctor of Sciences V.V. Zolotarev.

We suggest to organize study of experts on methods of noiseproof coding at your enterprise.

For contact:

In FGUE NIIRadio, Russia, Moscow:

tel.: +7 095 261 03 27 and +7 095 261 54 44;

e-mail: zolotas@yandex.ru, mob.: +7 916 518 86 28, V.V. Zolotarev or

e-mail: g_ovechkin@mail.ru, mob.: +7 910 644 51 46, G.V. Ovechkin