

Санкт-Петербургский Государственный Университет

Математико-механический факультет

Кафедра системного программирования

Помехоустойчивое свёрточное кодирование

Курсовая работа студента 345 группы

Коноплева Юрия Михайловича

Научный руководитель

..... В.В. Татищев

Санкт-Петербург

2011

Содержание

1. Введение	3
2. Свёрточный кодер.....	4-5
3. Последовательный декодер.....	6
4. Декодер Витерби.....	6-7
5. Заключение.....	8
6. Список литературы.....	9

Введение

В настоящее время происходит интенсивное развитие цифровых систем связи, таких как космическая, спутниковая, мобильная связь и др. Все эти системы используют для передачи беспроводные каналы, в которых на сигнал действуют помехи различной физической природы. Это приводит к тому, что принятые данные с большой вероятностью содержат ошибки, что для многих приложений недопустимо. В результате возникает задача обеспечения надежной передачи цифровой информации по каналам с помехами.

На сегодняшний день известно множество кодов и методов их декодирования, различающихся ЭВК, сложностью реализации и рядом других параметров. Далее описываются методы коррекции ошибок, в том числе рекомендованные стандартами передачи данных, обсуждаются их характеристики и области применения. Все представленные в обзоре характеристики методов помехоустойчивого кодирования даны для канала с аддитивным белым гауссовским шумом (AWGN), двоичной фазовой модуляции (BPSK) и демодулятора, способного формировать мягкие решения.

Многообразие существующих кодов делится на два класса: блочные коды и непрерывные коды. В блочных кодах передаваемая информационная последовательность разбивается на отдельные блоки с добавлением к каждому блоку определенного числа проверочных символов. Кодовые комбинации кодируются и декодируются независимо друг от друга. В непрерывных кодах, называемых также цепными, рекуррентными, конволюционными или сверточными, передаваемая информационная последовательность не разделяется на блоки, а проверочные символы размещаются в определенном порядке между информационными. Процессы кодирования и декодирования также осуществляются в непрерывном режиме.

Свёрточный кодер.

Для начала опишем свёрточный кодер.

Кодер представляет собой автомат с конечным числом состояний. Кодер обладает памятью, т. е. последовательность, выдаваемая на выходе зависит не только от входящего символа, но и от символов, находящихся в памяти кодера.

Характеристики кодера:

1. скорость кода $r=k_0/n_0$ где k_0 - количество входных символов, а n_0 — количество символов на выходе кодера. (Дальше будут рассматриваться кодеры скорости $1/n$, как использующиеся наиболее часто)
2. память кодера — общая длина регистров сдвига, используемая кодером.
3. кодовое ограничение — число символов на входе кодера, влияющие на n символов на выходе кодера

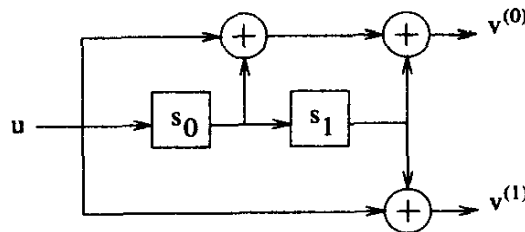


Рис. 1 Пример свёрточного кодера скорости 1/2, памяти 2

Кодер полностью определяется генераторами кода: g_0, \dots, g_{n-1} . Их можно определить как n импульсных откликов кодера, по одному на каждый выход $v^{(j)}$, $j = 0, 1, \dots, n$ на входящую последовательность $u = (10000\dots)$. Так же генераторы кода называют порождающими последовательностями.

Генераторы кода можно представить в полиномиальной форме

$$g_j(D) = g_j[0] + g_j[1]D + g_j[2]D^2 + \dots + g_j[m]D^m$$

Таким образом свёрточный код имеет представление с помощью многочленов.

Обычно, говоря о свёрточном коде, генераторы записывают в виде (g_0, \dots, g_{n-1}) в восьмиричной системе. Например, стандартный код NASA – код памяти 6 и скорости 1/2 с генераторами (171,133).

Генераторы кода однозначно определяют выходящую последовательность символов. Выходная последовательность может быть записана как

$$v^{(j)}[i] = \sum_{\ell=0}^m u[i-\ell]g_j[\ell]$$

В матричной форме это можно записать так

$$v = uG$$

где G – порождающая матрица свёрточного кода, v – последовательность на выходе,

и — входящая последовательность. Для кода памяти m и скорости $1/2$ матрица G выглядит следующим образом

$$G = \begin{pmatrix} g_0[0]g_1[0] & \dots & g_0[m]g_1[m] & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & g_0[0]g_1[0] & \dots & g_0[m]g_1[m] & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & g_0[0]g_1[0] & \dots & g_0[m]g_1[m] \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \ddots & \ddots \end{pmatrix}$$

Свёрточный кодер — конечный автомат с памятью. Поэтому его можно задать диаграммой состояний. Диаграмма состояний представляет собой ориентированный граф, вершинам которого соответствуют возможные состояния кодера, а рёбра указывают возможные переходы между состояниями.

Над каждым из рёбер надписывают последовательность символов, порождаемая кодером при данном переходе из состояния в состояние.

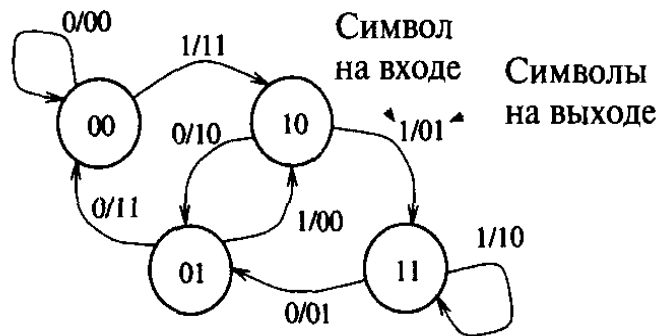


Рис. 2 Диаграмма состояний свёрточного кодера на рис. 1

Динамическая структура свёрточного кода позволяет построить решетчатую диаграмму. Фактически решетчатая диаграмма — это диаграмма состояний, развёрнутая во времени. Строится она следующим образом: в соответствии с диаграммой состояний кодера на каждом временном интервале соединяются рёбрами состояния на j и $j+1$ -ом тактах.

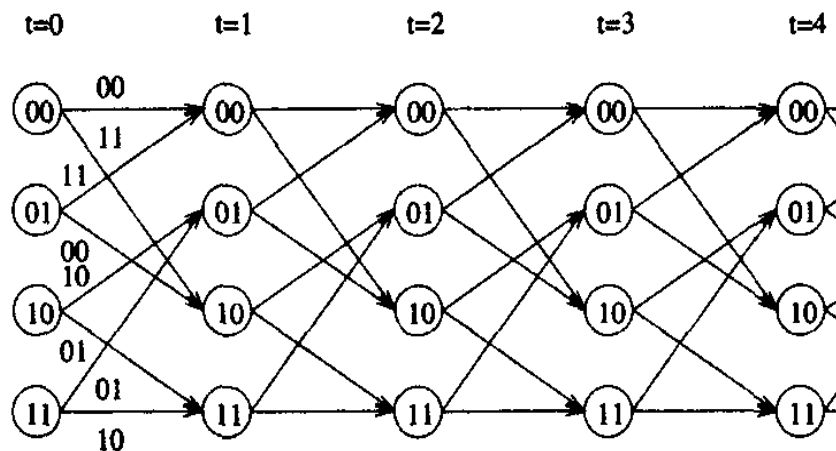


Рис. 3 Решетчатая диаграмма для кодера на Рис. 1. Первые 4 такта.

ПОСЛЕДОВАТЕЛЬНЫЙ ДЕКОДЕР

Для декодирования сверточных кодов с большей конструктивной длиной (например, сверточные коды стандарта ГИА-1008) могут применяться последовательные алгоритмы декодирования. Невысокая эффективность последовательных декодеров определяется тем, что с точки зрения теории такие алгоритмы работоспособны только в области меньшей, чем вычислительная скорость канала. Это примерно на 3 дБ выше границы, соответствующей его пропускной способности.

ДЕКОДЕР ВИТЕРБИ

Классическим методом коррекции ошибок, по праву считавшимся лучшим в течение нескольких десятилетий, является декодер Витерби, применяемый для декодирования сверточных кодов. Данный алгоритм является оптимальным и легко реализуемым для коротких сверточных кодов. В связи с этим сверточные коды, декодируемые с помощью алгоритма Витерби, применяются в большинстве стандартов систем передачи данных, например, в беспроводных сетях стандартов IEEE 802.11, IEEE 802.16, дальней космической связи CCSDS, спутниковой связи ГИА-1008 и др.

Использование длинных (с конструктивной длиной более 9) и потенциально более эффективных кодов в декодере Витерби представляется нецелесообразным из-за экспоненциального роста сложности его реализации от длины кода. Поэтому долгое время усилия многих специалистов были направлены на разработку алгоритмов декодирования, которые способны эффективно декодировать длинные коды при небольшой сложности реализации.

Далее более подробно будет рассмотрен декодер Витерби, так как в разрабатываемой системе используются коды с небольшой конструктивной длиной. Декодер, основанный на алгоритме Витерби в данном случае является наиболее оптимальным.

Алгоритм Витерби основан на принципе *максимума правдоподобия*. Декодер Витерби находит кодовую последовательность, ближайшую к принятой.

Рассмотрим алгоритм более подробно.

Обозначим за $S^{(k)}_j$ состояние на решётке (см. описание свёрточного кодера) в момент j . Каждому состоянию сопоставим метрику $M(S^{(k)}_j)$ и путь на решётке $y^{(k)}$, заканчивающийся в данном состоянии.

Основное свойство алгоритма: в момент j *все наилучшие пути имеют общее начало в некоторый момент $j-l$* .

Опишем теперь основные шаги алгоритма.

Введём два обозначения, которые нам в дальнейшем пригодятся.

$v[j] = (v_0[j] \ v_1[j] \ \dots \ v_{n-1}[j])$ – метка ребра кодовой решётки

$r[j] = (r_0[j] \ r_1[j] \ \dots \ r_{m-1}[j])$ – j -ое ребро на входе декодера

В момент $j = 0$ установим метрики состояний равными нулю, а пути, соответствующие состояниям пустыми.

Шаг 1: Вычисление метрик рёбер решётки.

Метрика ребра решётки на j -м шаге вычисляется как Хэммингово расстояние между

меткой ребра и j -м ребром на входе декодера.

Расстояние между последовательностями по Хэммингу — количество различающихся битов в последовательностях.

Шаг 2.: Прибавить, сравнить, выбрать.

Для каждого состояния $S_j^{(k)}$ и соответствующей пары рёбер, входящих в данное состояние из двух возможных предшествующих, вычислить и сравнить следующие суммы (пусть метрика первого ребра это a_1 , а второго — a_2):

$$M(S_{j-1}^{(k)}) + a_1 \text{ и } M(S_{j-1}^{(k)}) + a_2$$

Выбрать ребро с меньшей суммой.

Шаг 3: Обновление памяти путей.

Для каждого $S_j^{(k)}$ запомнить выжившие пути $y_{(k)}$, дополнив каждый из них выжившим ребром.

Шаг 4: Декодирование символов

Если $j > L$, то в качестве оценки кодовой последовательности выдать ребро $y_{(j-L)}^{(t)}$, где t - индекс состояния с наименьшей метрикой.

Декодер Витерби обладает следующим свойством: начать процесс декодирования можно с любого момента времени. Однако, необходима надёжная синхронизация по узлам кодового дерева, т. е. Синхронизация по символам рёбер решётки. Несоответствие может быть обнаружено при непрерывном наблюдении за значением некоторой случайной характеристикой декодера. Обычно это две характеристики: рост метрик путей и оценка вероятностной ошибки в канале.

При непрерывной работе декодера, метрики путей растут пропорционально длине принятой последовательности. Для недопущения переполнения метрик производится их нормализация.

Есть два способа нормализации метрик:

1. Пороговый способ. На каждом шаге декодирования значение минимальной метрики сравнивается с заранее заданным порогом. Если минимальная метрика пути больше порогового значения, то она вычитается из метрик всех путей.

2. Модулярный способ. На каждом шаге все метрики берутся по модулю $M = 2\Delta_{\max}$, где Δ_{\max} - это максимальная разность метрик путей.

Оба эти способа основаны на двух свойствах декодера Витерби:

1. Выбор наилучшего пути зависит только от разностей метрик.
2. Разности метрик путей ограничены.

Заключение

- ⤴ В рамках курсовой я ознакомился с помехоустойчивым свёрточным кодированием и методами декодирования свёрточных кодов.
- ⤴ В дальнейшем планируется более глубокое изучение помехоустойчивых кодов и их реализации применительно к практическим задачам.

Литература

1. Морелос-Сарагоса Р. “Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение ” - Москва: Техносфера 2005г.
2. Никитин Г.И. «Свёрточные коды: Учеб. пособие»/СПбГУАП. СПб., 2001
3. Джураев Р.Х., Джаббаров Ш.Ю., Умирзаков Б.М., Хамраев Э.А «ПОМЕХОУСТОЙЧИВЫЕ КОДЫ В ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМАХ»/ ТУИТ 2008
4. Ю.Б. Зубарев, д.т.н., член-корр. РАН, Г.В. Овечкин, к.т.н., доцент РГРТУ «ПОМЕХОУСТОЙЧИВОЕ КОДИРОВАНИЕ В ЦИФРОВЫХ СИСТЕМАХ ПЕРЕДАЧИ ДАННЫХ»
5. Convolutional code - Википедия http://en.wikipedia.org/wiki/Convolutional_code