

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Омский государственный университет им. Ф.М. Достоевского»

Физический факультет
Кафедра экспериментальной физики и радиофизики

Анализ алгоритмов, реализующих декодер LDPC

Выполнил: студент гр. ФРМ-502-О-07 Зимбович В.А.

Научный руководитель: к.т.н., руководитель группы ОАО «ОНИИП»
Привалов Д.Д.

Омск 2015

Актуальность

LDPC коды становятся востребованными в системах передачи информации, требующих максимальной скорости передачи при ограниченной полосе частот. Высокая вычислительная эффективность LDPC кодов заключается в использовании сильно прореженных проверочных матриц, тогда как энергетическая эффективность достигается за счет использования итеративных алгоритмов декодирования. Поэтому данная работа в настоящее время актуальна в связи с бурным развитием технологий передачи данных.

LDPC коды используются в стандарте спутниковой связи DVB-S2.



Цель и задачи

Цель работы: Анализ алгоритмов, реализующих декодер LDPC

Для достижения цели решались следующие задачи:

анализ литературных данных по алгоритмам декодирования LDPC кодов.

Рассмотрен :

- алгоритм распространения доверия
- Алгоритм Ричардсона-Новичкова
- Алгебра ЛОП
- Алгоритм, основанный на аппроксимации логарифмом Якоби
- Проведен анализ вычислительных затрат

1. Алгоритм распространения доверия

- **1 шаг.** Инициализация. Для всех узлов i инициализировать значения L_i , используя выражение:

$$L_i = L(x_i | y_i) = \ln \left(\frac{p(x_i = 0 | y_i)}{p(x_i = 1 | y_i)} \right) \quad (1)$$

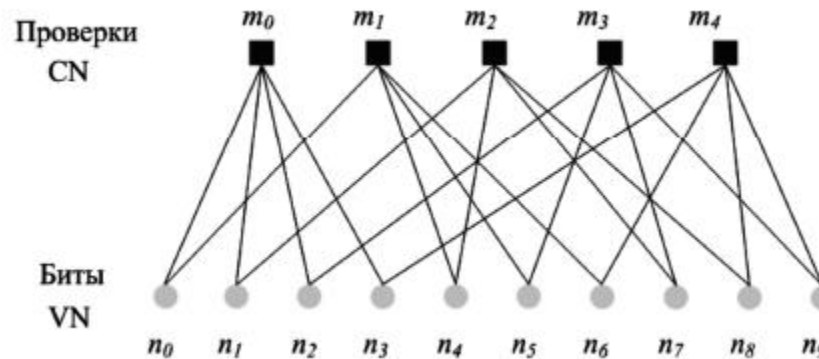
- **2 шаг.** Обновление проверочных узлов. Для всех проверочных узлов CN вычислить по формуле исходящие сообщения

$$L_{j \rightarrow i} = \prod_{i' \in V_{ji}} \tanh \left(\frac{1}{2} L(q_{ij}') \right) L_{i' \rightarrow j} \quad (2)$$

3 шаг. Обновление битовых узлов. Вычислить сообщения, исходящие от информационных узлов VN .

И передать их соответствующим проверочным узлам

$$L_{i \rightarrow j} = L_i + \sum_{j' \in N(j) - \{i\}} L_{i' \rightarrow i} \quad (3)$$



Граф Таннера для линейного блочного кода

4 шаг. Вычисление апостериорных ЛОП . Для всех $j=0,1,\dots,N-1$ вычислить

$$L_i^{total} = L_i + \sum_{j' \in N(i) - \{j\}} L_{j \rightarrow i} \quad (4)$$

5 шаг. Получение жестких решений. Для всех $i=0,1,\dots, N-1$ найти жесткие решения

$$V_i = \left\{ \begin{array}{l} 1, L_i^{total} < 0, \\ 0, L_i^{total} \geq 0 \end{array} \right\} \quad (5)$$

6 шаг. Проверка условия остановки. Вычислить синдром $\hat{v}H^T$

Реализация функций th и th^{-1} является сложной вычислительной задачей, и Галлагер предложил упрощенный вариант вычисления этого выражения

$$\prod_{i' \in V_{ji}} \tanh\left(\frac{1}{2} L(q_{ij}^i)\right) L_{i' \rightarrow j} = \prod_{i' \in N(j) - \{i\}} \alpha_{i'} \phi\left(\sum_{i' \in N(j) - \{i\}} \phi(\beta_{i'j})\right) \quad (6)$$

Где

$$L_{i \rightarrow j} = \alpha_{i \rightarrow j} \beta_{i \rightarrow j},$$

$$\alpha_{i \rightarrow j} = \text{sign}(L_{i \rightarrow j}),$$

$$\beta_{i \rightarrow j} = |L_{i \rightarrow j}|,$$

$$\phi(x) = -\ln[th(x/2)] = \ln\left(\frac{e^x + 1}{e^x - 1}\right)$$

2. Упрощенные алгоритмы

Основным недостатком АД является его вычислительная сложность, а именно второй шаг алгоритма – обновление проверочных узлов, поскольку требует вычисления специальных функций. Поскольку алгоритм итеративный, ситуация еще более усугубляется, так как все шаги алгоритма выполняются многократно. Естественным направлением исследований стали попытки упрощения этого шага с наименьшими потерями в производительности. На сегодняшний день известно более десятка различных модификаций АД

2.Алгоритм Ричардсона-Новичкова

Алгоритм основан на аппроксимации функции

$$\phi(x) = -\ln[th(x/2)] = \ln\left(\frac{e^x + 1}{e^x - 1}\right) \quad (7)$$

Выражение для расчета сообщений от проверочных узлов имеет вид:

$$|L_{j \rightarrow i}| = \phi^{-1}\left(\sum_{i' \in N(j) - \{i\}} \phi(\beta_{i'j})\right) *$$

Для больших x справедливо $\log(1 \pm e^{-x}) \approx \pm e^{-x}$

получим $\phi(x) = \log\left(\frac{1 + e^{-\beta_{ij}}}{1 - e^{-\beta_{ij}}}\right) = \log(1 + e^{-\beta_{ij}}) - \log(1 - e^{-\beta_{ij}}) \approx 2e^{-\beta_{ij}}$ (8)

При этом обратная функция $\phi^{-1}(x) \approx -\ln\left(\frac{x}{2}\right)$

Дружественность алгоритма аппаратной реализации состоит в том, что путем достаточно простых преобразований в выражении * можно заменить натуральный логарифм и возведение в степень экспоненты, соответственно двоичным логарифмом и возведением в степень 2. Алгоритм показывает хорошую эффективность для различных кодов

3.Алгебра ЛОП

Обновление проверочных узлов можно представить в несколько ином виде, во многих случаях более удобном для реализации и позволяющем получить достаточно простые (с точки зрения реализации) аппроксимации. Но для этого нам потребуется следующий важный результат.

Для двух независимых случайных величин a_1 и a_2 с вероятностями

$p(a_i = x) = p_x^{(i)}$, $x \in \{0,1\}$ и ЛОП $L_i = L(a_i) = \ln(p_0^{(i)} / p_1^{(i)})$ можно показать, что

ЛОП их суммы равно:

$$L(A_2) = \ln\left(\frac{1 + e^{L_1 + L_2}}{e^{L_1} + e^{L_2}}\right) \quad (9)$$

Этот факт может быть непосредственно применен для вычисления исходящего из проверочного узла CN сообщения в случае, если его степень равна двум $d_c - 1 = 2$

4. Алгоритм, основанный на аппроксимации логарифмом Якоби

Данный алгоритм основывается на следующей численной аппроксимации выражения при помощи логарифма Якоби

$$\ln(e^{x_1} + e^{x_2}) = \max(x_1, x_2) + \ln(1 + e^{-|x_1 - x_2|}) \quad (10)$$

Применив логарифм Якоби к выражению дважды получим:

$$\begin{aligned} L_{sum}(L_1, L_2) &= \ln\left(\frac{1 + e^{L_1 + L_2}}{e^{L_1} + e^{L_2}}\right) = \\ &= \ln(1 + e^{L_1 + L_2}) - \ln(e^{L_1} + e^{L_2}) = \\ &= \max(0, L_1 + L_2) + \ln(1 + e^{-|L_1 + L_2|}) - \max(L_1, L_2) - \ln(1 + e^{-|L_1 - L_2|}) \end{aligned} \quad (11)$$

Тогда окончательно имеем:

$$\begin{aligned} L_{sum}(L_1, L_2) &= \\ &= \text{sign}(L_1) \text{sign}(L_2) \min(|L_1|, |L_2|) + g(L_1, L_2) = \\ &= \text{sign}(L_1) \text{sign}(L_2) \left[\min(|L_1|, |L_2|) + g(|L_1|, |L_2|) \right] = \\ &= \text{sign}(L_1) \text{sign}(L_2) \min^*(|L_1|, |L_2|), \end{aligned} \tag{12}$$

Где $g(L_1, L_2) = \ln(1 + e^{-|L_1+L_2|}),$

$$\min^*(|L_1|, |L_2|) = \min(|L_1|, |L_2|) + g(|L_1|, |L_2|)$$

Для самой распространенной реализации $g(L)$ – с использованием таблицы, соответствующие значения реализации приведены в табл.1. Было показано, что при использовании таблицы из 8 значений функции, погрешность аппроксимации составляет не более 5 %, что достаточно для приемлемой точности декодирования.

Оценка вычислительных затрат

Название	Синдром	Проверочные узлы	Информационные узлы	Жесткие решения
АРД th	$M(d_r-1)$	$d_r M\{(d_r-1)+(d_r-1)[th]+[arth]+[*]\}$	$Ndc\{1+(dc-1)\}$	N
АРД	.	$dr M\{(d_r-1)[\varphi]+(d_r-1)[+][\varphi]+[*]+ +(d_r-1)\}$ $\varphi = -\ln(\text{th}(x/2))$ $\varphi = \ln((ex+1)/(e^x-1))$.	.

Заключение

эффективность всех алгоритмов, кроме АРН, сопоставима и отличается от АРД менее чем на 0,1 дБ. Для моделирования алгоритма АРН использовались значения поправочных коэффициентов $C1 = 0,1$; $C2 = 0$; его проигрыш АРД составляет порядка 0,1 дБ. Поскольку эффективность рассмотренных алгоритмов очень близка, то при выборе одного из них для реализации декодера низкоплотностного кода, следует руководствоваться, в первую очередь, сложностью реализации каждого из алгоритмов на используемой аппаратной платформе. Стоит отметить, что для других кодов разница в эффективности может быть более существенной, что может оказаться критичным для выбора алгоритма. Анализируя таблицу вычислительных затрат, становится очевидным тот факт, что все приведенные модификации алгоритмов нацелены на снижение сложности расчета сообщений от проверочных узлов, что достаточно логично, поскольку это самый сложный из всех шагов АРД.

Список литературы

- 1. Gallager R.G. Low-density parity-check codes // Cambridge, MA: M.I.T. Press, 1963.
- Ryan W.E., Lin S. Channel codes. Classical and modern // Cambridge, University Press, 2009.
- Richardson T. and Novichkov V., «Node processors for use in parity check decoders», United State Patent 6,938,196 B2, August 30, 2005
- DVB-S2 low density parity check codes with near Shannon limit performance Mustafa Erozu, Feng Wen Sun and Lin-Nan Lee Hughes Network Systems, 11717 Exploration Lane, Germantown, MD 20876, U.S.A.
- Richardson T. and Novichkov V., «Node processors for use in parity check decoders», United State Patent 6,938,196 B2, August 30, 2005
- Лихобабин Е., Дворкович А. Использование квазиоптимальных алгоритмов декодирования LDPC кодов в системе цифрового телевизионного вещания стандарта DVB-T2 // Труды РНТОРЕС имени А.С. Попова. Серия: Цифровая обработка сигналов и ее применения – М. Выпуск XII-1. С. 25-27. 2010.
- Кравченко А.Н. Снижение сложности декодирования низкоплотностного кода. «Цифровая обработка сигналов». 2010, № 2, С.35-41.
- 4. Tanner R.M. A recursive approach to low complexity codes // IEEE Trans. Info. Theory, vol. IT-27. №.5. pp. 533-547. September 1981.

Спасибо за внимание!