Лекция 6

Ilya Yaroshevskiy

6 Октября

Содержание

1 Укорочение **1**

2 Выкалывание **1**

3 Расширение кодов **2**

4 Каскадный код **2**

5 Прямое произведение кодов **3**

6 TODO Лестничные коды **3**

7 Каскадные коды **3**
7.1 Граница Зяблова 4
7.2 Обобщенные каскадные коды 4

8 Турбо коды **4**
8.1 Алгоритм Бала-Коке-Елинека-Равина декодирования светочных кодов 5
8.2 Декодирование с использованием ЛОПП 5
8.3 Построение перемежителей 6

9 Заключение **6**

1 Укорочение

Определение. Укороченный код полсучается путем выбора кодовых слов исходного кода, содержащих нули на заданных позициях, с последующим удалением этих нулей

Пусть дан (n, k, d) код с порождающей матрицей $G = (I|A)$. Удалим из порождающей матрицы m столбцов удиничной подматрицы и соответствующие m строк.

Пример. $(7, 4, 3)$ совершенный код, эквивалентный коду Хэмминга

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \Rightarrow G' = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$(5, 2, 3)$ совершенным не является

2 Выкалывание

Определение. Выколотый код: удалим из всех кодовых слов заданные символы (как правило, проверочных).

Пусть дана проверочная матрица (n, k, d) кода в форме $H = (A| - I)$. Удалим из H m столбцов единичной подматрицы и соответствующие им m строк. Если проявятся линейно зависимые строки, удалим их $\implies$ $(n - m, \leq k, \geq d - m)$ код

Пример. Построение оптимального кода $(10, 3, 5)$.

- Граница Грайсмера: $N(3,5) \geq 5 + N(2,3) \geq 5 + 3 + N(1,2) = 5 + 3 + 2 = 10$
- C_1: Код Хемминга $(7, 4, 3)$ может быть укорочен до $(6, 3, 3)$
- C_2: Код с повторениями $(6, 1, 6)$
- Конструкция Плоткина $(C_1, C_1 + C_2)$: код $(12, 4, 6)$
- Выкалывание одного (последнего) символа: код $(11, 4, 5)$
- Укорочение на один символ: код $(10, 3, 5)$

$$G_{\text{Ham}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \Rightarrow \begin{matrix} G_1= \\ \\ \\ G_2= \end{matrix} \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & & & & & & & & \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & & & & & & & & \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & & & & & & & & \end{pmatrix} \Rightarrow$$

$$\Rightarrow \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & & & & & & & & & \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & & & & & & & & & \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & & & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & & & & & & & & & \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

3 Расширение кодов

Определение. Наиболее распространенный способ – добавление проверки на четность. $(n, k, d) \Rightarrow (n + 1, k, d')$.

Если минимальное расстояние d исходного кода нечетно, то минимальное расстояние расширенного кода $d' = d + 1$.

Пример. $(7, 4, 3)$ код Хемминга

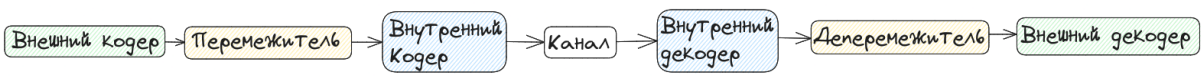
$$G = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, H = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$(8, 4, 4)$ расширенный код Хемминга

$$G' = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, H' = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

4 Каскадный код

Перемежитель таким образов переставляет символы, что последствия ошибочного декодирования одного кода могут быть легко ликвидированы декодером другого кода



5 Прямое произведение кодов

Определение. Пусть даны (n_1, k_1, d_1) (кодирование по строчкам) и (n_2, k_2, d_2) (кодирование по столбцам) коды с порождающими матрицами G', G''. Кодовое слово образуется путем выписывания полученной матрицы по столбцам. $(n_1 n_2, k_1 k_2, d_1 d_2)$ код с порождающей матрицей

$$G' \otimes G'' = \begin{pmatrix} G'_{11}G'' & G'_{12}G'' & \dots & G'_{1n_1}G'' \\ G'_{21}G'' & G'_{22}G'' & \dots & G'_{2n_1}G'' \\ \vdots & \vdots & \ddots & \vdots \\ G'_{k_11}G'' & G'_{k_12}G'' & \dots & G'_{k_1n_1}G'' \end{pmatrix}$$

$$\mathrm{R} = \mathrm{k}_1 k_2 \overline{{}_{n_1 n_2 < \frac{k_1}{n_1} \cdot \frac{k_2}{n_2}}}$$

Код способен исправить многие конфигурации ошибок веса $> \frac{d_1 d_2}{2}$

Замечание. Параллельный алгоритм кодирования

Доделать Картинка

Пример. Код Рао-Редди $(48, 31, 8)$

Используется РЖД в рельсовых цепях сигнализации. Прямое произведение расширенного $(16, 11, 4)$ C_1 кода Хемминга и $(3, 2, 2)$ кода C_2 с проверкой на четность $\implies (48, 22, 8)$.

Замечание. Кодовые слова имеют вид $(c_1, c_2, c_1 + c_2), c_1, c_2 \in C_1$

Дополнение кодовыми словами (с дописанными в конец 32 нулями) кода Рида-Маллера $(16, 5, 8)$ C_3

Замечание. Кодовые слова имеют вид $(c_1{+}c_3, c_2, c_1{+}c_2), c_1, c_2 \in C_1, c_2 \in C_3$. Вес ненулевого кодового слова имеют вид $(c_1 + c_3, c_2, c_1 + c_2)$:

- $c_3 = 0 : \mathrm{wt}((c_1, c_2, c_1 + c_2)) \geq 8$
- $c_3 \neq 0 : \mathrm{wt}((c_1{+}c_3, c_2, c_1{+}c_2)) = \mathrm{wt}(c_1{+}c_3){+}\mathrm{wt}(c_2){+}\mathrm{wt}(c_1{+}c_2) \geq \mathrm{wt}((c_1{+}c_3){+}(c_2){+}(c_1{+}c_2)) = \mathrm{wt}(c_3) \geq 8$

Получен код $(48, 27, 7)$

Пусть C_1 – код с порождающей матрицей

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Код $(48, 21, 8)$ Рао-Редди состоит из кодовых слова вида $(c_1 + c_3 + c_4, c_2 + c_4, c_1 + c_2 + c_3), c_1, c_2 \in C_1, c_3 \in C_3, c_4 \in C_4$

6 TODO Лестничные коды

7 Каскадные коды

Закодируем данные внешним (N, K, D) кодом над $GF(q^k)$

Замечание. Строить хорошие коды над $GF(q^k), k > 1$ проще, чем над $GF(q)$

Пример. код Рида-Соломона с $D = N - K + 1$

Представим символы кодового слова как векторы длины k над $GF(q)$. Закодируем каждый символ (n, k, d) внутренним кодом над $GF(q)$. (Nn, Kk, Dd) код над $GF(q)$. Существуют каскадные коды, достигающие предела Шеннона для воичного симметричного канала

7.1 Граница Зяблова

Выберем внутренний (n, k, d) код на границе Варшамова-Гилберта с $r = \frac{k}{n} \geq 1 - h\left(\frac{d}{n}\right) = 1 - h(\delta)$. Внешний $(N, K, D = N - K + 1)$ код Рида-Соломона с $R = 1 - \frac{D-1}{N} \approx 1 - \Delta$. Сущесвтует код со скоростью $\rho = Rr$ и относительными расстоянием

$$\bar{\delta} = \frac{Dd}{Nn} = \Delta\delta \geq (1 - R)(1 - h^{-1}(r)) = \left(1 - \frac{\rho}{r}\right)(1 - h^{-1}(r))$$

$$\bar{\delta} \geq \max_{0 \leq r \leq h(\bar{\delta})} \left(1 - \frac{\rho}{r}\right)(1 - h^{-1}(r))$$

Далеко не все семейства кодов при длине $\to \infty$ одновременно обеспечивают скорость $\rho > 0$ и относительное минимальное расстояние $\bar{\delta} > 0$

7.2 Обобщенные каскадные коды

Определение. Внешние (N_i, K_i, D_i) коды $\mathcal{A}_i$ над $GF(q^{m_i}), 1 \leq i \leq s$.

Замечание. Вложенные внутренние (n_i, k_i, d_i) коды $\mathcal{B}_i : B_1 \supset \mathcal{B}_2 \supset \cdots \supset \mathcal{B}_s$ над $GF(q)$

- $k_i - k_{i+1} = m_i$
- Код $\mathcal{B}_i$ порождается последними k_i строками $k_1 \times n$ матрицы B

Кодирование:

- Закодируем данные внешними кодами и запишем полученные кодовые слова в $s \times N$ матрицу X
- Заменим элементы i-ой строки X на их векторное представление (столбец длиной m_i). Пусть Y – полученная $k_1 \times N$ матрица
- Умножим каждый столбец Y на B (т.е. закодируем в коде $\mathcal{B}_1$)
- Полученная $n \times N$ матрица может рассматриваться как кодовое слово

Линейный $(Nn, \sum_{i=1}^{s} K_i m_i, \geq \min_{1 \leq i \leq s} d_i D_i)$ код над $GF(q)$. Некоторые ОКК (полярные коды) достигают предела Шеннона.

Доделать Картинка

- Запишем принятые символы в виде $n \times N$ матрицы
- `for i=1, ... , s`
  - Продекодируем столбцы в коду $\mathcal{B}_i$
  - Продекодируем i-ую строку в коде $\mathcal{A}_i$. Пусть $(c_i, \ldots, c_N)$ – найденное кодовое слово
  - Вычтем из j-ого столбца $c_j B_i^T$

8 Турбо коды

Определение. Одновременное кодирование информационных битов нескольких сверточными кодами позволяет исключить многие конфигурации ошибок, которые могли бы возникнуть при использовании независимых сверточных кодов. Должны использоваться рекурсивные систематические сверточные коды. Многие информационные последовательности маого веса превращаются в кодовые слова большого веса. Длина кодового ограничения должна быть небольшой. Кодам с большим кодовым ограничением присущи длинные пакеты ошибок декодирования. Турбо-код является линейным блоковым кодом. Для повышения скорости кода используется выкалывание

Одновременное кодирование информационных битов нескольких сверточными кодами позволяет исключить многие конфигурации ошибок, которые могли бы возникнуть при использовании независимых сверточных кодов. Должны использоваться рекурсивные систематические сверточные коды. Многие информационные последовательности маого веса превращаются в кодовые слова большого веса. Длина кодового ограничения должна быть небольшой. Кодам с большим кодовым ограничением присущи длинные пакеты ошибок декодирования. Турбо-код является линейным блоковым кодом. Для повышения скорости кода используется выкалывание

Минимальное расстояние Доделать Картинка

Доделать Картинка

Декодеры сверточных кодов входящих в турбо-код, обмениваются информацией, полученной в результате декодирования. Как правило, достаточно $5-10$ итераций. Апостериорные логарифмические отношение правдоподобия информационных символов, вычисленные одним декодером, являются априорными ЛОПП для другого декодера. Аппроксимация декодера максимального правдоподобия. Этот подход применим и для декодирования прямого произведения кодов

8.1 Алгоритм Бала-Коке-Елинека-Равина декодирования светочных кодов

- Прямая рекурсия $\alpha'_i(s) = \frac{\sum_{\tilde{s}\in V_{i-1}} \alpha_{i-1}(\tilde{s})\gamma_i(\tilde{s},s)}{\sum_{s'\in V_i}\sum_{\tilde{s}\in V_{i-1}} \alpha_{i-1}(\tilde{s})\gamma_i(\tilde{s},s')}$, $0 < i \le n$
- Обратная рекурсия $\beta'_i(\tilde{s}) = \frac{\sum_{s\in V_{i+1}} \gamma_{i+1}(\tilde{s},s)\beta'_{i+1}(s)}{\sum_{s\in V_i}\sum_{s'\in V_{i+1}} \alpha'_i(s')\gamma_{i+1}(s',s)}$, $0 \le i < n$
- Вычисление апостериорных ЛОПП $L_i = \ln \frac{\sum_{(s',s)\in S_1} \alpha'_i(s')\gamma_{i+1}(s',s),\beta'_{i+1}(s)}{\sum_{(s',s)\in S_0} \alpha'_i(s')\gamma_{i+1}(s',s)\beta'_{i+1}(s)}$
- Вероятности переходов (для кода со скоростью $1/2$)

$$\gamma_{i+1}(s',s) = p(s_{i+1}=s, y_{2i}, y_{2i+1}|s_i=s') = P\{s_{i+1}=s|s_i=s'\}p(y_{2i},y_{2i+1}|s_i=s', s_{i+1}=s) = \\ = P\{u_i=\delta(s',s)\}p(y_{2i},y_{2i+1}|(c_{2i},c_{2i+1})=\Delta(s',s))$$

  - $P\{u_i=\delta(s',s)\}$ – априорная вероятность того, что информационный символ u_i принимает значение, соответствующее метке перехода $\delta(s',s)\in\{0,1\}$. В турбо-декодере эту вероятность вычисляет второй компонентный декодер
  - $\Delta(s',s)$ – пара символов, формируемая кодером при переходе из состояния s' в s

8.2 Декодирование с использованием ЛОПП

- Внешние (extrinsic) ЛОПП $L_i^e = \ln\frac{P\{u_i=0\}}{P\{u_i=1\}} = \frac{P\{u_i=0\}}{1-P\{u_i=0\}}; P\{u_i=0\} = \frac{\exp(L_i^e)}{1+\exp(L_i^e)}$

$$P\{u_i=a\} = \begin{cases} \frac{\exp(L_i^e)}{1+\exp(L_i^e)}\exp(L_i^e/2) & ,a=0 \\ \frac{\exp(L_i^e)}{1+\exp(L_i^e)}\exp(-L_i^e/2) & ,a=1 \end{cases}$$

$$P\{u_i=a\} = A\exp((-1)^a L_i^e/2)$$

- Пусть $S_i = \ln\frac{P\{y_i|c_i=0\}}{P\{y_i|c_i=1\}}$. Вероятности выходных символов

$$p(y_{2i},y_{2i+1}|(c_{2i},c_{2i+1})) = B\exp\left(\frac{(-1)^{c_{2i}}S_{2i} + (-1)^{c_{2i+1}}S_{2i+1}}{2}\right)$$

- Коэффициенты A, B сокращаются во всех выражениях, используемых в алгоритме БКЕР, а потому могут быть отброшены

Получение внешних ЛОПП:
Предположим, что используется систематическое кодирование сверточных кодов и $c_{2i}=u_i$, где u_i – информационные символы. Результатом работы алгоритма БКЕР являются $L_i = \ln\frac{P\{u_i=0|y_0,\dots,y_{2n-1}\}}{P\{u_i=1|y_0,\dots,y_{2n-1}\}}$.
Для турбо-декодирования необходимо вычислить $\tilde{L}_i^c = \ln\frac{P\{u_i=0|Y_{2i}\}}{P\{u_i=1|Y_{2i}\}}$, $Y_{2i} = (y_0,\dots,y_{2i-1},y_{2i+1},\dots,y_{2n-1}$.
Это позволит исключить двойной учет принятых символов (по крайней мере, на первой итерации)

$$P\{u_i=a|Y_{2i},y_{2i}\} = \frac{P\{u_i=a,Y_{2i},y_{2i}\}}{P(Y_{2i},y_{2i})} = \frac{P(Y_{2i},y_{2i}|u_i=a)P\{u_i=a\}}{P(Y_{2i},y_{2i})} = \\ = \frac{P(Y_{2i}|u_i=a)P(y_{2i}|u_i=a)P\{u_i=a\}}{P(Y_{2i},y_{2i})} = \frac{P(Y_{2i}|u_i=a)P(y_{2i}|c_{2i}=a)P\{u_i=a\}}{P(Y_{2i},y_{2i})}$$

$$L_i = \ln\frac{P\{Y_{2i}|u_i=0\}}{P\{Y_{2i}|u_i=1\}} + S_i + L_i^e$$

Различные полуитерации используют различные Y_i, при этом изначально известно, что $P\{u_i = 0\} = P\{u_i = 1\} = 1/2$. Поэтому $\tilde{L}_i^c = L_i - S_i - L_i^e$

Итеративный алгоритм декодирования:

1. Положить $L_{1\to 2}^e(u_i) = 0$
2. Воспользоваться декодером БКЕР для сверточного кодера 1
3. Подвергнуть перемежению полученные ЛОПП $\tilde{L}_i^e$ и использовать их как $L_{1\to 2}^e(u_i$
4. Воспользоваться декодером БКЕР для сверточного кодера 2
5. Подвергнуть деперемежению полученные апостериорные ЛОПП $\tilde{L}_i^e$ и использовать их как $L_{2\to 1}^e(u_i)$
6. Перейти к шагу 2, если не превышено максимальное число итераций $(5 - 10)$

8.3 Построение перемежителей

Требование: близкие позиции во входной последовательности должны отображаться в максимально удаленные позиции в выходной последовательности

$$0 < |i - j| < d \implies |\pi(i) - \pi(j)| \geq S$$

Важны объем памяти, требуемый для реализации перемежителя, его задержка. Псевдослучайный перемежитель: случайная генерация с отбрасыванием перестановок с неудовлетворительными S, d

Пример. Табличный перемежитель

$$(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16) \to \begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{pmatrix} \to$$

$$\to (13,9,5,1,14,10,6,2,15,11,7,3,16,12,8,4)$$

Перестановочный полином: если $Q(x) = \sum_{i_0}^{l} q_i x^i \mod N$ – биекция в $\mathbb{Z}_N$, то перестановка может быть задана как $Q(i) \to Q(i+1$

Пример. $Q(x) = \frac{kx(x+1)}{2} \mod N, k \equiv 1 \mod 2$

9 Заключение

- Длинные коды можно строить из коротких
- Составные коды допускаюи простое декодирование
- Существуют каскадные коды
  - у которых относительное минимальное расстояние положительным при всех скоростях
  - достигающие предела Шеннона для двоичного симметричного канала (1966, Форни)
- Некоторые обобщенные каскадные коды (полярные) достигают предела Шеннона
- Турбо-коды стали первым классом корректирующих кодов, которые смогли на практике приблизится в пределу Шеннона (1993)