Лекция 12

Ilya Yaroshevskiy

17 ноября

Содержание

1 Декодирование БЧХ

- Дана последовательность $S_0, \dots, S_{\delta-2}$
- Как восстановить регист сдвига минмальной длины, порождающий эту последовательность по ее начальной части?

$$\Lambda_1 S_{j+t-1} + \dots + \Lambda_t S_j d - S_{j+1}$$

- Фильтр $(L, \Lambda^{(n)}(x) = 1 + \sum_{i=1}^{L} \Lambda_i^{(n)} x^i)$ порождает последовательность S_0^{n-1}, если

$$S_k = -\sum_{i=1}^{L} \Lambda_i^{(n)} S_{k-i}, L \le k \le n-1$$

  Параметры L и $\Lambda^{(n)}(x)$ назваются длиной фильтра (РСЛОС) и многочленом связей
- В общем случае $\deg \Lambda^{(n)}(x) \le L$

Доделать Картинка

1.1 Минимальный РСЛОС

Лемма 1. *Пусть фильтры $(L_{n-1}, \Lambda^{(n-1)}(x))$ и $(L_n, \Lambda^{(n-1)}(x))$ порождает последовательности S_0^{n-2} и S_0^{n-1} соответственно, причем $(L_{n-1}, \Lambda^{(n-1)}(x))$ не прождает $S_0^{(n-1)}$, и величины L_{n-1} и L_n являются наименьшими возможными. Тогда $L_n \ge \max(L_{n-1}, n - L_{n-1})$*

Доказательство. Фильтр, порождающий S_0^{n-1}, обязан порождать и S_0^{n-2}, поэтому $L_n \ge L_{n-1}$. Покажем, что если фильтр $(L_{n-1}, \Lambda^{(n-1)}(x))$ порождает S_0^{n-2}, но не порождает S_0^{n-1}, то $L_n \ge n - L_{n-1}$. Предположим, что это не так. Тогда

$$S_{n-1} \ne -\sum_{i=1}^{L_{n-1}} \Lambda_i^{(n-1)} S_{n-1-i} = \sum_{i=1}^{L_{n-1}} \Lambda_i^{(n-1)} \sum_{k=1}^{L_n} \Lambda_k^{(n)} S_{n-1-k-i}$$

Полседний переход возможен в силу того, что для всех i выполняется $L_n \leq n - L_{n-1} - 1 \leq n - i - 1 \leq n - 2$, т.е. величины S_{n-1-i} могут быть порождены с помощью $(L_n, \Lambda^{(n)}(x))$. Меняя порядок суммирования, получим

$$S_{n-1} \neq \sum_{k=1}^{L_n} \Lambda_k^{(n)} \sum_{i=1}^{L_{n-1}} \Lambda_i^{(n-2)} S_{n-1-k-i} = -\sum_{k=1}^{L_n} \Lambda_k^{(n)} S_{n-1-k} = S_{n-1}$$

Из полученного противоречия вытекает , что $L_n \geq n - L_{n-1}$ □

Теорема 1.1. Предположим, что РСЛОС $(L_i, \Lambda^{(i)}(x))$ порождает $S_0^i, 0 \leq i \leq r-1$, причем величины L_i являются нименьшими возможными. Тогда РСЛОС с многочлено связей

$$\Lambda^{(r)}(x) = \begin{cases} \Lambda^{(r-1)}(x) & \text{,если } \Delta_r^{(r)} = 0 \\ \Lambda^{(r-1)}(x) - \frac{\Delta_r^{(r)}}{\Delta_m^{(m)}} x^{r-m} \Lambda^{(m-1)}(x) & \text{,если } \Delta_r^{(r)} = 0 \end{cases} \tag{1}$$

порождает S_0^{r-1} и имеет нименьшую длину. Здесь $\Delta_r^{(v)} = \sum_{j=0}^{L_{r-1}} \Lambda_j^{(v)} S_{r-1-j}$ – невязка, m – наибольшее число, меньшее r, такое, что $L_{m-1} < L_m$, и $\Delta_0^{(0)} = 1$

- $\Delta_r^{(v)}$ равна разности истинного значения S_{r-1} и значения, вычисленно с помощью $(L_v, \Lambda^{(v)}(x))$
- Наименьшая возможная длина РСЛОС $L_i = \max(L_{i-1}, i - L_{i-1})$

Доказательство. Будем считать, что при $r = 0 \Lambda^{(0)}(x) = 1$, и покажем, что на каждом шаге правило 1 приводит к РСЛОС наименьшей длины. При $\Delta_r^{(r)} = 0$ в изменении РСЛОС нет необходимости, т.е. $\Lambda^{(r)}(x) = \Lambda^{(r-1)}(x)$. Если старый РСЛОС имел наименьшую длину, то он сохраняет это свойство и для S_0^{r-1}. В противном случае: Длины РСЛОС менялась на шаге $m \implies L_{r-1} = m - L_{m-1}, L_{m-1} < L_{r-1}$ Модифицированный многочлен имеет степень /(° $\Lambda^{(r)}$ (x) ≤ max(° $\Lambda^{(r\ -\ 1)}$ (x), r - m + ° $\Lambda^{(m\ -\ 1)}$(x)) ≤ max($L_{r\ -\ 1}$, r - m + $L_{m\ -\ 1}$) = max($L_{r\ -\ 1}$, r - L_{r-1}) $\implies$ $\Lambda^{(r)}$(x)\) может рассматриваться как многочлен связей для РСЛОС с оптимальной длиной $L_r = \max(L_{r-1}, r - L_{r-1})$. Невязка, соответствующая модифицированному РСЛОС, равна

$$\Delta_r^{(r)} = \sum_{j=0}^{L_{r-1}} \Lambda_j^{(r-1)} S_{r-1-j} - \frac{\Delta_r^{(r-1)}}{\Delta_m^{(m-1)}} \sum_{j=0}^{L_{m-1}} \Lambda_j^{(m-1)} S_{r-1-j-(r-m)} = \Delta_r^{(r-1)} - \frac{\Delta_r^{(r-1)}}{\Delta_m^{(m-1)}} \Delta_m^{(m-1)} = 0$$

Таким образом, $(L_r, \Lambda^{(r)}(x))$ действительно порождает S_{r-1}. $(L_r, \Lambda^{(r)}(x))$ порождает и предшествующие элементы S_{k-1}

$$\Delta_k^{(r)} = \sum_{j=0}^{L_r} \Lambda_j^{(r)} S_{k-1-j} = \sum_{j=0}^{L_{r-1}} \Lambda_j^{(r-1)} S_{k-1-j} - \frac{\Delta_r^{(r-1)}}{\Delta_m^{(m-1)}} \sum_{j=0}^{L_{m-1}} \Lambda_j^{(m-1)} S_{k-1-j-(r-m)} =$$

$$= \Delta_k^{(r-1)} - \frac{\Delta_r^{(r-1)}}{\Delta_m^{(m-1)}} \Delta_{k-r+m}^{(m-1)} = 0, L_{r-1} + 1 \leq L_r + 1 \leq k \leq r - 1$$

Последнее равенство вытекает их того, что РСЛОС $(L_{r-1}, \Lambda^{(r-1)}(x))$ порождает S_0^{r-2}, т.е. $\Delta_k^{(r-1)} = 0, L_{r-1}+1 \leq k \leq r-1$, и $(L_{m-1}, \Lambda^{(m-1)}(x))$ порождает S_0^{m-2}, т.е. $\Delta_k^{(m-1)} = 0, L_{m-1}+1 \leq k \leq m-1$ □

1.2 Алгоритм Берлекэмпа-Месси

Program 1 Алгоритм Берлекэмпа-Месси

$\Lambda(x) \leftarrow 1, r \leftarrow 1, m \leftarrow 0, L \leftarrow 0, B(x) \leftarrow 1$
while $\Delta_r \leftarrow \sum_{j=0}^{L} \Lambda_j S_{r-1-j}$ **do**
  if $\Delta_r \neq 0$ **then**
    $T(x) \leftarrow \Lambda(x) - \Delta_r x^{r-m} B(x)$
    if $2L \leq r - 1$ **then**
      $B(x) \leftarrow \Delta_r^{-1} \Lambda(x)$
      $\Lambda(x) \leftarrow T(x)$
      $L \leftarrow r - L$
      $m \leftarrow r$
    else
      $\Lambda(x) \leftarrow T(x)$
    end if
  end if
  $r \leftarrow r + 1$
end while
return $(L, \Lambda(x))$

- Если $L \neq \deg \Lambda(x)$, число ошибок превышает $(\delta - 1)/2$ (крому случая расширенных кодом БЧХ)
- Сложность $O((\delta - 1)^2)$
- Самый простой этап декодирования кодов БЧХ
- Для двоичных кодов БЧХ в укзом смысле $\Delta_{2i} = 0 \implies$ половину итераций можно пропустить

2 Мягкое декодирование кодов БЧХ

2.1 Метод Чейза-2 мягкого декодирования

- Найдем τ наименее надежных символов принятого векотора $(y_0, \dots, y_{n-1})$. Пусть они расположены в позициях $0, \dots, \tau - 1$
- Пусть $\hat{y}_i \in GF(q)$ – жесткое решение относительно y_i
- Переберем все q^τ векторов $(x_0, \dots, x_{\tau-1}, \hat{y}_\tau, \hat{y}_{\tau+1}, \dots, \hat{y}_{n-1}), x_i \in GF(q)$. Для кжадого такого вектора выполним его жесткое декодирование с исправлением $(\delta - 1)/2$ ошибок
- Из полученных кодовых слов выберем наиболее вероятное для $(y_0, \dots, y_{n-1})$
- Сложность $O(2^\tau \delta^2)$

2.2 Метод Пинди декодирвания с мягким выходом

- Апостериорные ЛОПП

$$L_i = \ln \frac{\sum_{c \in \mathcal{C}: c_i = 0} \prod_{j=0}^{n-1} P(c_j | y_j)}{\sum_{c \in \mathcal{C}: c_i = 1} \prod_{j=0}^{n-1} P(c_j | y_j)} \approx \ln \frac{\max_{c \in \mathcal{C}: c_i = 0} \prod_{j=0}^{n-1} P(c_j | y_j)}{\max_{c \in \mathcal{C}: c_i = 0} \prod_{j=0}^{n-1} P(c_j | y_j)} = \min_{c \in \mathcal{C}: c_i = 1} E(c, y) - \min_{c \in \mathcal{C}: c_i = 0} E(c, y)$$

- Пусть $\mathcal{L}$ – список, полученный декодированием y в коде $\mathcal{C}$ (Чейз-2б Тал-Варди, ...), и $\hat{c}$ – наиболее вероятный его элемент
- $L_i \approx \min_{c \in \mathcal{L}: c_i = 1} E(c, y) - \min_{c \in \mathcal{L}: c_i} = 0 E(c, y)$
- Внешние ЛОПП:

$$\hat{L}_i = \begin{cases} L_i - \ln \frac{P(y_i | c_i = 0)}{P(y_i | c_i = 1)} & \text{, если } \exists c', c'' \in \mathcal{L} : c'_i = 0, c''_i = 1 \\ \beta(1 - 2\hat{c}_i) & \text{, иначе} \end{cases}$$

, где β – экспериментально подбираемый параметр

3 QR-коды (1967)

- Число y называется квадратичным вычетом *quadraticresidue* по модулю n, если существует решени сравнения $x^2 \equiv y \mod n$. В противном случае y – квадратичный невычет
- $(n-x)^2 \equiv x^2 \mod n \implies$ квадратичными вычетами являются $1^2, 2^2, \ldots, ((n-1)/2)^2 \mod n$
- Произведение квадратичных вычетом – квадратичный вычет

Определение. Квадратично-вычетным называется циклический код длины n над полем $GF(p)$ с порождающим многочленом $g_1(x) = \prod_{i \in Q}(x-\alpha^i), (x-1)g_1(x), g_2(x) = \prod_{i \in N}(x-\alpha^i)$ или $(x-1)g_2(x)$, где n – простое число, p – квадратичный вычет по модулю n, $\alpha \in GF(p^m)$ – примитивный корень степени n из 1, Q и N – множество квадратичных вычетов и невчетов по модулю n

Минимальное расстояние d КВ кода удовлетворяет $d^2 \geq n$. Если $n = 4s-1, s \in \mathbb{N}$, то $d^2-d+1 \geq n$

4 QR-коды (1994) (жалкая подделка)

- Quick Response code
- Данные представляются в виде черных и белых точек
- Для защиты от ошибок считывания используются коды Рида-Соломона с различными параметрами
- Возможность исправления ошибок позволяет создавать художественные QR-коды

5 Cyclic Redundancy Check

- CRC – циклический код, используемый для обнаружнеия ошибок
- Контрольная сумма (многочлен проверочных символов $r(x)$) для многочлена данных $a(x)$ вычисляется с помощью формулы систематического кодирования $r(x) \equiv x^{N-K}a(x) \mod g(x), \deg a(x) \leq K-1$
- Число проверочных символов равно $N-K = \deg g(x), N \leq n$

$$g(x) | (x^n - 1)$$

- $K \leq n - \deg g(x)$
- Нельзя допускать $K > n - \deg g(x)$, т.к. это приведем к коду с неизвестным (вероятно, плохим) минимальным расстоянием

6 Выводы

- Циклические коды допускают еще более компактное задание по сравнению с линейными блоковыми кодами
- Конструкция кодов БЧХ позволяет получить коды с заданным минимальным расстоянием
- Коды Рида-Соломона – коды БЧХ, лежащие на границе Синглтона
- Существуют алгоритмы декодирования кодов БЧХ с исправлением $\lfloor (\delta-1)/2 \rfloor$ ошибок со сложностью $O(n\delta + \delta^2)$
- Расширенные примитивные коды БЧХ в узком смысле достигают предела Шеннона для двоичного стирающего канала
- Метод Чейза-Пиндии мягкого декодирования