Санкт-Петербургский государственный политехнический университет
Кафедра компьютерных систем и программных технологий

Надежность систем и устройств

Моисеев Михаил Юрьевич

Лекция №7

Характеристики помехоустойчивых кодов
Коды с произвольным кодовым расстоянием

2012

Повторение предыдущего материала

- Как изменится размерность кода Хэмминга если
  - добавить в порождающую матрицу одну строку
  - добавить в порождающую матрицу одну строку и один столбец
  - удалить из порождающей матрицы одну строку
  - удалить из порождающей матрицы один столбец

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Характеристики помехоустойчивых кодов

- Помехоустойчивый **(n,k)**-код, обнаруживающий или исправляющий **t** ошибок
- Скорость кода

$$R = \frac{k}{n}$$

- Корректирующие способности кода
  - исходная вероятность ошибки на символ p
  - вероятность ошибки на символ после обнаружения или исправления ошибок p'

Корректирующие способности кода

- Комбинации ошибок в кодовом векторе

$$P_{вект} = (1-p)^n + C_n^1 \cdot p \cdot (1-p)^{n-1} + C_n^2 \cdot p^2 \cdot (1-p)^{n-2} + \ldots$$

$$P_{вект} = \sum_{i=0}^{n} C_n^i \cdot p^i \cdot (1-p)^{n-i}$$

- Вероятность неправильного исправления/обнаружения

$$p' = \sum_{i=t+1}^{n} C_n^i \cdot p^i \cdot (1-p)^{n-i} \qquad p' = 1 - \sum_{i=0}^{t} C_n^i \cdot p^i \cdot (1-p)^{n-i}$$

Задача 3

- Вероятность ошибки в КС равна **0.01**. Рассчитать скорость кода и вероятность ошибки для **(7,4)**-кода Хэмминга, выполняющего исправление ошибок

$$p' = 1 - \sum_{i=0}^{1} C_n^i \cdot p^i \cdot (1-p)^{n-i} = 1 - C_7^0 \cdot (1-p)^7 - C_7^1 \cdot p \cdot (1-p)^6 =$$

$$= 1 - (1-0.01)^7 - 7 \cdot 0.01 \cdot (1-0.01)^6 = 1 - 0.9321 - 0.0659 = 0.002$$

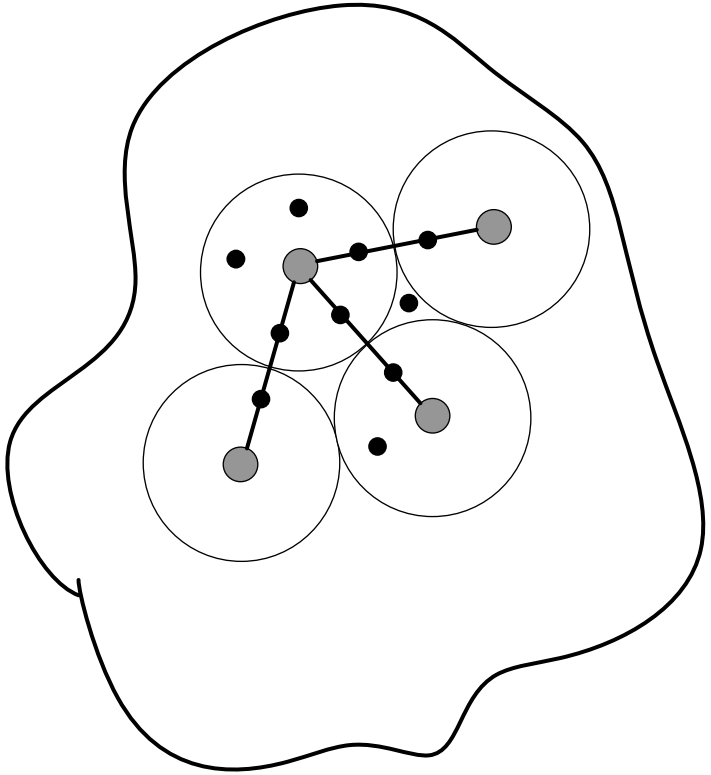
Совершенные и квазисовершенные коды

- Совершенный код – сферы одинакового радиуса не пересекаясь покрывают все пространство
- Коды Хэмминга $\mathbf{t_{испр}=1}$
- Код Голея **(32,12)**-код с $\mathbf{t_{испр}=3}$

$$C_{23}^{0}+C_{23}^{1}+C_{23}^{2}+C_{23}^{3}=2^{11}$$

$$[C_{23}^{0}+C_{23}^{1}+C_{23}^{2}+C_{23}^{3}]\cdot 2^{12}=2^{23}$$

- Эффективность кода растет с увеличением длины кода



Квазисовершенные коды

- Квазисовершенный **(n,k)**-код – код с наибольшим возможным кодовым расстоянием для данных **n** и **k**
- Для некоторых кодов доказано, что они являются квазисовершеными
- В большинстве случаев используют просто «хорошие» коды – сравнение кодов по скорости и вероятности ошибки на символ

Линейные коды с произвольным кодовым расстоянием

$$H_{(n-k)\times n} = \left[\begin{array}{c|c} P^T & \begin{matrix} 1 & & \\ & 1 & \\ & & 1 \\ & & & \ddots \\ & & & & 1 \end{matrix} \end{array}\right] \begin{matrix} \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \\ n\text{-}k \end{matrix}$$

1 k k+1 n

$$H = [P^T \vdots I]$$

$$\begin{cases} 1.\ W(1, i_1) \geq d - 1 \\ 2.\ W(2, i_2) \geq d - 2 \\ \ldots\ldots \\ j.\ W(j, i_j) \geq d - j \\ \ldots\ldots \\ d-1.\ W(2t, i_{d-1}) \geq 1 \end{cases}$$

- где $W(j, i_j)$ – вес j - кратной суммы чисел подматрицы P^T, i_j – порядковый номер этой суммы

Линейные коды с кодовым расстоянием 5

$$H_{(n-k)\times n} = \left[\begin{array}{c|c} P^T & \begin{matrix} 1 & & \\ & 1 & \\ & & 1 \\ & & & \ddots \\ & & & & 1 \end{matrix} \end{array}\right] \begin{matrix} \\ \\ \\ \\ \\ n\text{-}k \end{matrix}$$

1 k k+1 n

$$H = [P^T \vdots I]$$

$$d = 3 \quad \begin{cases} 1.\ W(1, i_1) \geq 2 \\ 2.\ W(2, i_2) \geq 1 \end{cases}$$

$$d = 5 \quad \begin{cases} 1.\ W(1, i_1) \geq 4 \\ 2.\ W(2, i_2) \geq 3 \\ 3.\ W(3, i_3) \geq 2 \\ 4.\ W(4, i_4) \geq 1 \end{cases}$$

- Коды Хэмминга – частный случай рассматриваемых линейных кодов
- Вектора, удовлетворяющие неравенствам найдены для разных **d**

Синтез линейных кодов по таблицам

- По заданному **d** выбираем таблицу
- Выбираем из таблицы первые **k** векторов и определяем **n**
- Строим матрицу $H = [P^T \vdots I]$, столбцами подматрицы P^T являются выбранные вектора
- По матрице *H* строим матрицу *G*, уравнения проверки и контроля

Пример синтеза линейного кода

- Построить линейный код, если k=4 и $t_{испр}$=2
  - определяем кодовое расстояние: $d = 2t+1 = 5$
  - выбираем 4 вектора из таблицы: 17,63,125,152
  - строим матрицу H

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}
x_1	0	0	1	1	1	0	0	0	0	0	0
x_2	0	1	0	1	0	1	0	0	0	0	0
x_3	0	1	1	0	0	0	1	0	0	0	0
x_4	1	0	0	1	0	0	0	1	0	0	0
x_5	1	0	1	0	0	0	0	0	1	0	0
x_6	1	1	0	1	0	0	0	0	0	1	0
x_7	1	1	1	0	0	0	0	0	0	0	1

Задача 4

- Построить линейный код с k=5 и $t_{испр}$=2, построить порождающую и проверочную матрицы, уравнения проверки и контроля

Декодирование линейных кодов

- Декодирование основано на вычислении синдрома $x = c' \times H^T$
  - обнаружение ошибок – ненулевой синдром
  - исправление ошибок – поиск наименьшей комбинации ошибок, которая приводит к данному синдрому
- Алгоритмы исправления ошибок
  - по методу сопоставлений
  - по таблицам синдромов
  - весовое синдромное декодирование

Декодирование по методу сопоставлений

- Хранятся все кодовые слова $n \cdot 2^k$
- Для принятого вектора находим синдром , если его вес не больше t, то в информационных разрядах нет ошибок
- Складываем поочередно принятый вектор со всеми кодовыми словами пока вес суммы $w(c' + c_i) \le t$

$$H_{(n-k)\times n} = \left[\begin{array}{c|c} P^T & \begin{matrix} 1 & & \\ & 1 & \\ & & 1 \\ & & & \ddots \\ & & & & 1 \end{matrix} \end{array}\right] \begin{matrix} \\ \\ \\ \\ n\text{-}k \end{matrix}$$

1 k k+1 n

$$H = [P^T \vdots I]$$

Декодирование по таблице синдромов

- Хранится таблица соответствия синдромов и ошибок $n \cdot 2^{n-k}$
- Для принятого вектора находим синдром , если его вес меньше *t*, то в информационных разрядах нет ошибок
- Определяем ошибку по синдрому и меняем значения ошибочных разрядов

X	e
000	000000
001	000001
010	001000
011	000010
100	010100
…	…

Весовое синдромное декодирование

- В памяти хранятся столбцы подматрицы P $(n-k)\cdot k$
- Для принятого вектора находим синдром , если его вес меньше t, то в информационных разрядах нет ошибок
- Сравниваем синдром принятого вектора со всеми комбинациями столбцов подматрицы P пока не найдем комбинацию $S_i^{(j)}$

$$w(X - S_i^{(j)}) \le t - j$$

$$H_{(n-k)\times n} = \left[\begin{array}{c|ccc} P^T & 1 & & \\ & & 1 & \\ & & & 1 \\ & & & \ddots \\ & & & 1 \end{array}\right]$$

1 k k+1 n

n-k

$$H = [P^T \vdots I]$$

Поля Галуа

- $GF(q)$ – поле из *q* элементов, поле коэффициентов (целых чисел), *q* – простое число, определены операции +,-,*, / по модулю *q (q=2)*
- $GF(q^n)$ – расширение поля коэффициентов, поле многочленов степени *(n-1)* с коэффициентами из поля $GF(q)$
- Примитивный элемент β – такой элемент $GF(q^n)$, что все элементы кроме 0 могут быть представлены его степенью
- Примитивный многочлен $p(X)$ – приведенный многочлен, который нельзя разложить как произведение многочленов меньшей степени. Примитивный элемент является корнем примитивного многочлена

Поля Галуа

- Если над $GF(q)$ найден примитивный многочлен степени *n*, можно построить расширение $GF(q^n)$ по этому многочлену, выбрав в качестве примитивного элемента $\beta = 000...0010 \equiv X$
- Операции в $GF(q^n)$выполняется по модулю примитивного многочлена $p(X)$
- Будем рассматривать поля Галуа с двоичными коэффициентами $GF(2^n)$

Пример построения поля Галуа

- Построение $GF(2^4)$ $p(X) = X^4 + X + 1$ $\beta = 000...0010 \equiv X$

$$\beta^0 = 0001 \equiv X^0 \qquad \beta^5 = 0110 \equiv X^2 + X \qquad \beta^{10} = 0111 \equiv X^2 + X + 1$$
$$\beta^1 = 0010 \equiv X^1 \qquad \beta^6 = 1100 \equiv X^3 + X^2 \qquad \beta^{11} = 1110 \equiv X^3 + X^2 + X$$
$$\beta^2 = 0100 \equiv X^2 \qquad \beta^7 = 1011 \equiv X^3 + X + 1 \qquad \beta^{12} = 1111 \equiv X^3 + X^2 + X + 1$$
$$\beta^3 = 1000 \equiv X^3 \qquad \beta^8 = 0101 \equiv X^2 + 1 \qquad \beta^{13} = 1101 \equiv X^3 + X^2 + 1$$
$$\beta^4 = 0011 \equiv X + 1 \qquad \beta^9 = 1010 \equiv X^3 + X \qquad \beta^{14} = 1001 \equiv X^3 + 1$$
$$\beta^{15} = 0010 \equiv X$$

$$\beta^{15} = 0010 = \beta$$

$$\beta^3 + \beta^5 = X^3 + X^2 + X = \beta^{11},\ \beta^8 - \beta^{14} = \left(X^2 + 1\right) - \left(X^3 + 1\right) = X^3 + X^2,$$
$$\beta^6 \cdot \beta^9 = \beta^{15} = \beta, \beta^7 / \beta^{11} = \beta^{-4} = \beta^{-4} \cdot \beta^{15} = \beta^{11}$$

Задача 5

- Построить поле Галуа $GF(2^3)$ и вычислить в нем результаты следующих выражений

$$\beta^3 + \beta^{15} = ?$$
$$\beta^5 - \beta^{14} = ?$$
$$\beta^{16} \cdot \beta^{49} = ?$$
$$\beta^{37} / \beta^{11} = ?$$

Свойства полей Галуа

- Каждое поле Галуа содержит хотя бы один примитивный элемент
- Над каждым полем Галуа существует хотя бы один примитивный многочлен любой степени

Циклические коды

- Циклический код – линейный код, который инвариантен к циклическому сдвигу кодовых векторов

$$c = (c_1, c_2, \ldots, c_n),$$
$$\hat{c} = (c_n, c_1, \ldots, c_{n-1}),$$

- Циклический **(n,k)**-код может определяться с помощью порождающего многочлена $g(X)$ степени *(n-k)* и состоит из произведений $g(X)$ на все многочлены степени не больше *(k-1)*

Коды Хэмминга как циклические

- Для любого целого *m* существует **(2^m-1, 2^m-m-1)**-код с кодовым расстоянием *d=3*
- Проверочная матрица этого кода задается

$$H = (\beta^{2^m-2}, \beta^{2^m-1} \ldots \beta^1, 1)$$

- Циклический код длины $n=2^m-1$, построенный по порождающему многочлену $g(X) = P(X)$, имеет кодовое расстояние *d=3*
- Например для $GF(2^4)$ можно построить (15,11)-код с порождающим многочленом $g(X) = X^4 + X + 1$

Кодирование для циклических кодов

- Несистематический код

$$c(X) = i(X) \cdot g(X)$$

- Систематический код

$$r(X) = i(X) \cdot X^{n-k} \bmod g(X),$$
$$c(X) = i(X) \cdot X^{n-k} + r(X) = q(X) \cdot g(X)$$

- Код будет систематическим, так как степень остатка меньше (*n-k*) (остаток определяет проверочные разряды)

Задача 6

- Выполнить кодирование $i(X)=1010$ для систематического и несистематического **(7,4)**-кода с $g(X)=X^3+X+1$

Матричное представление циклических кодов

- Порождающую матрицу можно построить, выбрав в качестве строк сдвиги порождающего многочлена $g(X)$
- Проверочную матрицу можно построить, выбрав в качестве строк сдвиги проверочного многочлена $h(X)$

$$h(X) = (X^n + 1) / g(X)$$

Пример построения матриц циклических кодов

$GF(2^3) \quad g(X) = X^3 + X + 1$

$$h(X) = (X^7 + 1)/(X^3 + X + 1) = X^4 + X^2 + X + 1$$

$$G = \begin{matrix} X^3 \cdot g(X) \\ X^2 \cdot g(X) \\ X \cdot g(X) \\ g(X) \end{matrix} = \left| \begin{matrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{matrix} \right|$$

$$H = \begin{matrix} X^2 \cdot h(X) \\ X \cdot h(X) \\ h(X) \end{matrix} = \left| \begin{matrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{matrix} \right|$$

Коды Боуза-Чоудхури-Хоквингема (БЧХ)

- Код, состоящий из всех многочленов с коэффициентами из поля $GF(q)$ для которых элементы $\beta^1, \beta^2, \beta^3 \ldots \beta^{d-1}$ являются корнями, называется БЧХ-кодом
- Код БЧХ обладает кодовым расстоянием ***d***
- Задача синтеза БЧХ-кода – необходимо построить $g(X)$, для которого все элементы являются корнями (тогда все кодовые вектора будут иметь требуемые корни $c(X) = i(X) \cdot g(X)$)
- Построение порождающего многочлена, $g(X) = НОК(f_1, f_2 \ldots f_{d-1})$ где f_j имеет корень β^j

Пример синтеза кода БЧХ

- Построим код БЧХ с *d*=7

$$GF(2^4),\ p(X) = X^4 + X + 1,\ n = 2^4 - 1$$

$$\beta^1, \beta^2, \beta^3, \beta^4, \beta^5, \beta^6 \qquad g(X) = НОК(f_1, f_2, f_3, f_4, f_5, f_6)$$

- Выбор приведет $f_j(X) = \left(X - \beta^j\right)$ к тому, что коэффициенты кодового многочлена будут из $GF(2^4)$ – нужно чтобы из $GF(2)$
- Для того чтобы коэффициенты были из $GF(2)$ необходимо чтобы для каждого корня β все $\beta, \beta^2, \beta^{2^2} \ldots \beta^{2^{m-1}}$ также были бы корнями кодовых векторов ($GF(2^4)$, т.е. *m* = 4)

$$f_j(X) = \left(X - \beta^j\right) \cdot \left(X - \left(\beta^j\right)^2\right) \cdot \left(X - \left(\beta^j\right)^4\right) \cdot \ldots$$

Пример синтеза кода БЧХ

- Для корня β дополнительные корни $\beta^2, \beta^4, \beta^8, \left(\beta^{16} = \beta^2\right)$

$$f_1(X) = (X - \beta)\cdot\left(X - \beta^2\right)\cdot\left(X - \beta^4\right)\cdot\left(X - \beta^8\right) = X^4 + X + 1 = P(X)$$

- Для корня β^3 дополнительные корни $\beta^3, \beta^6, \beta^{12}, \beta^{24} = \beta^9, \left(\beta^{48} = \beta^3\right)$

$$f_3(X) = \left(X - \beta^3\right)\cdot\left(X - \beta^6\right)\cdot\left(X - \beta^{12}\right)\cdot\left(X - \beta^9\right) =$$
$$= X^4 + X^3 + X^2 + X + 1$$

- Для корня β^5 дополнительный корень β^{10}

$$f_5(X) = \left(X - \beta^5\right)\cdot\left(X - \beta^{10}\right) = X^2 + X + 1$$

Пример синтеза кода БЧХ

- Порождающий многочлен

$$g(X) = НОК\left(f_1(X), f_3(X), f_5(X)\right) = f_1(X) \cdot f_3(X) \cdot f_5(X) =$$
$$= X^{10} + X^8 + X^5 + X^4 + X^2 + X + 1$$

- Степень порождающего многочлена определяет число проверочных разрядов – получили **(15,5)**-код БЧХ с кодовым расстоянием *d* =7

Синтез кодов БЧХ

- Для любых целых **m** и **t** существует двоичный код БЧХ длины $\mathbf{n=2^m-1}$, исправляющий все ошибки кратности **t** и меньше и имеющий не более чем **mt** проверочных символов
- Коды БЧХ строятся с помощью таблиц минимальных функций или таблиц порождающих многочленов

Синтез кодов БЧХ по таблицам

- По заданным **k** и **t** находим минимальное значение **m**
- Из соответствующей таблицы выбираем нужное количество минимальных функций $f_1(X), f_3(X),\ldots, f_{2t-1}(X)$
- Находим порождающий многочлен как произведение минимальных функций

Пример синтеза кода БЧХ по таблицам

- Построить код БЧХ с **k=20** и **t=2**
- Минимальное **m=5**
- Выбираем $f_1(X), f_3(X)$ из таблицы для m=5
  - 45 = 100 101 $f_1(X) = X^5 + X^2 + 1$
  - 75 = 111 101 $f_3(X) = X^5 + X^4 + X^3 + X^2 + 1$
- Определяем порождающий многочлен

$$g(X) = (X^5 + X^2 + 1) \cdot (X^5 + X^4 + X^3 + X^2 + 1) =$$
$$= X^{10} + X^9 + X^8 + X^6 + X^5 + X^3 + 1$$

- Получили **(31,21)**-код БЧХ, его можно сократить до **(30,20)**

Задача 7

- Построить код БЧХ с **k=8** и **t=3**, определить параметры кода

Алгоритмы декодирования кодов БЧХ

- Обнаружение ошибок – по ненулевому остатку от деления на порождающий многочлен

$$c(X) \bmod g(X) = 0$$

- Для исправления ошибок могут применяться алгоритмы декодирования линейных кодов
- Имеются более эффективные алгоритмы (меньшая вычислительная сложность)
  - Алгоритм Питерсона–Горинстейна–Цирлера
  - Алгоритм Берклекэмпа-Месси
  - Алгоритм Форни

Вопросы

- Перечислите основные характеристики помехоустойчивых кодов
- Что такое совершенные и квазисовершенные коды?
- Можно ли построить ЛК с произвольным кодовым расстоянием?
- Чем отличается поле многочленов от поля коэффициентов?
- Что такое циклический код?
- Что такое код БЧХ ?
- Можно ли построить код БЧХ произвольной длины?