Лекция 15. Устойчивость кодов к ошибкам. Коды, обнаруживающие ошибки, и коды, исправляющие ошибки, их свойства. Мощность кода, исправляющего ошибки. Линейные коды.

Лектор — Селезнева Светлана Николаевна
selezn@cs.msu.ru

факультет ВМК МГУ имени М.В. Ломоносова

Лекции на сайте https://mk.cs.msu.ru

Ошибки

Пусть φ — кодирование из A в B, $\alpha \in A^*$ и $\beta = \varphi(\alpha) \in B^*$.

Предположим, что слово β хранится в памяти компьютера или передается по каналу связи.

При этом в нем могут произойти ошибки, и слово β перейдет в слово β', возможно, в другом алфавите, т. е. $\beta' \in (B \cup B')^*$.

Можно ли по слову β' установить, что оно с ошибками?

А можно ли по слову β' восстановить слово β?

Устойчивость кода к ошибкам

Можно ли построить кодирование φ, устойчивое к ошибкам?

Повседневный опыт подсказывает, что в некоторых случаях можно.

Например, *если мы читаем текст, в котором в каких-то словах опечатки, то мы можем восстановить правильные слова* (*если опечаток не очень много*).

Например, *если мы разговариваем по телефону, но связь с помехами, то иногда мы также можем понять, что говорит собеседник*.

Устойчивость кода к ошибкам

Попытаемся построить кодирование, устойчивое к ошибкам.

Будем рассматривать исходный и кодирующий алфавиты из двух букв, т. е. пусть $A = B = \{0, 1\}$.

Ошибки

Какие ошибки в кодах сообщений можно рассматривать?

Например, ошибки зашумления, т. е. когда в слове β буквы 0 и 1 могут заменяться на какие-то буквы, не принадлежащие алфавиту B (все такие неправильные буквы можно обозначить 2, т. е. в этом случае $B' = \{2\}$).

Например, ошибки замещения, т. е. когда в слове β буква 0 может заменяться на букву 1, а буква 1 — на букву 0 (в этом случае $B' = \emptyset$).

Ошибки замещения

Будем рассматривать ошибки замещения, т. е. если 0 может заменяться на 1, а 1 — на 0.

При этом ограничим число ошибок, которые могут происходить в кодах сообщений.

Итак, пусть задано число t, $t \geqslant 1$. Можно ли построить разделимый код, устойчивый к t ошибкам замещения?

Устойчивость к ошибкам

Что означает устойчивость кода к ошибкам?

Определим коды, обнаруживающие t ошибок и коды, исправляющие t ошибок.

Устойчивость к ошибкам

Пусть $A = B = \{0, 1\}$, φ — кодирование из A в B и S — множество сообщений, $S \subseteq A^*$.

Рассмотрим код C_φ, т. е. множество кодов всех сообщений,

$$C_\varphi = \{\varphi(\alpha) \mid \alpha \in S\}.$$

Код, обнаруживающий t ошибок

Код $C_\varphi \subseteq B^*$ назовем **обнаруживающим t ошибок**, если для любого слова $\beta \in C_\varphi$ выполняется следующее условие:

если в слове β произойдет не более t ошибок замещения и при этом оно перейдет в слово β', то по неправильному слову β' можно установить, что ошибки были.

Код, исправляющий t ошибок

Код $C_\varphi \subseteq B^*$ назовем **исправляющим t ошибок**, если для любого слова $\beta \in C_\varphi$ выполняется следующее условие:

если в слове β произойдет не более t ошибок замещения и при этом оно перейдет в слово β', то по неправильному слову β' можно:

1) установить, что ошибки были;

2) в случае, когда ошибки были, восстановить правильное слово β.

Идея: дублирование

Несложно придумать коды, обнаруживающие или исправляющие t ошибок, основанные на дублировании букв.

Идея: дублирование

Пример. Код, обнаруживающий t ошибок замещения.

Рассмотрим алфавитное кодирование $\varphi_1 : A^* \to B^*$, где

$$\begin{aligned} \varphi_1(0) &= \underbrace{00\ldots0}_{t+1}, \\ \varphi_1(1) &= \underbrace{11\ldots1}_{t+1}. \end{aligned}$$

Например, если $t = 1$, то $C_{\varphi_1} = \{00, 11\}$.

Теперь если $\beta_1' = 00$, то в нем нет ошибки; а если $\beta_2' = 01$, то оно с ошибкой, но восстановить правильное слово β_2 нет возможности.

Идея: дублирование

Пример. Код, исправляющий t ошибок замещения.

Рассмотрим алфавитное кодирование $\varphi_2 : A^* \to B^*$, где

$$\begin{aligned} \varphi_2(0) &= \underbrace{00\ldots0}_{2t+1}, \\ \varphi_2(1) &= \underbrace{11\ldots1}_{2t+1}. \end{aligned}$$

Например, если $t = 1$, то $C_{\varphi_2} = \{000, 111\}$.

Теперь если $\beta_1' = 000$, то в нем нет ошибки.

Если же $\beta_2' = 011$, то оно с ошибкой, и при этом можно восстановить правильное слово $\beta_2 = 111$.

Неэкономность дублирования

Но так построенные коды кратно увеличивают длину кода сообщения по сравнению с длиной исходного сообщения.

А именно, если $\alpha \in A^*$, то

$$\frac{|\varphi_1(\alpha)|}{|\alpha|} = t + 1, \quad \frac{|\varphi_2(\alpha)|}{|\alpha|} = 2t + 1.$$

Можно ли построить более экономные коды, обнаруживающие или исправляющие t ошибок?

Идея: расстояние

Идея дублирования букв состоит в том, что кодовые слова отличаются не менее, чем в $(t+1)$-й букве, а ошибок может быть не более t.

Поэтому если произойдет не более t ошибок замещения, то никакое кодовое слово не может перейти ни в какое другое кодовое слово.

Но можно применить эту же идею к кодам сообщений: пусть они отличаются не менее, чем в $(t+1)$-й букве.

Тогда если произойдет не более t ошибок замещения, то код никакого сообщения не может перейти в код какого-то другого сообщения.

Идея: голосование

Идея восстановления правильного слова при дублировании букв состоит в том, что кодовые слова отличаются не менее, чем в $(2t+1)$-й букве, а ошибок может быть не более t.

Поэтому если произойдет не более t ошибок замещения, то в любом кодовом слове правильных букв останется больше, чем неправильных.

Но можно применить эту же идею к кодам сообщений: пусть они отличаются не менее, чем в $(2t+1)$-й букве.

Тогда если произойдет не более t ошибок замещения, то код ровно одного сообщения окажется ближе к неправильному слову, чем коды всех других сообщений.

Коды из слов одинаковой длины

Итак, $A = B = \{0, 1\}$. Пусть φ — разделимое кодирование из A в B.

Пусть $n \geqslant 1$ и $S \subseteq A^*$ — множество слов, которые кодированием φ преобразуются в слова длины n, т. е.

$$S = \{\alpha \in A^* \mid |\varphi(\alpha)| = n\}.$$

Рассмотрим код C_φ,

$$C_\varphi = \{\varphi(\alpha) \mid \alpha \in S\}.$$

Отметим, что все слова в коде C_φ имеют одну и ту же длину n.

Устойчив ли код C_φ к t ошибкам замещения?

Слова и наборы

Далее иногда будем взаимозаменять понятия слова длины n в алфавите B и набора длины n из B^n.

Поэтому код C_φ можно рассматривать как подмножество множества B^n, т. е. $C_\varphi \subseteq B^n$.

Равномерные коды

Пусть $B = \{0, 1\}$, $n \geqslant 1$ и $C \subseteq B^n$.

Множество C назовем **равномерным кодом**.

Если $\beta \in C$, то β назовем **кодовым словом**.

Шары в множестве B^n

Вспомним некоторые определения.

Расстоянием $\rho(\alpha, \beta)$ между наборами $\alpha, \beta \in B^n$ называют число разрядов, в которых они отличаются.

Шаром радиуса r, $r \geqslant 0$, с центром в точке $\alpha \in B^n$ называется множество:

$$S_r(\alpha) = \{\beta \in B^n \mid \rho(\alpha, \beta) \leqslant r\}.$$

Т. е. шар $S_r(\alpha)$ содержит в точности все такие наборы $\beta \in B^n$, которые от набора α находятся на расстоянии не более r.

Если $S_r(n)$ обозначает число наборов в шаре радиуса r в B^n, то

$$S_r(n) = \sum_{k=0}^{r} C_n^k,$$

где C_n^k — биномиальный коэффициент из n по k.

Кодовое расстояние

Пусть $B = \{0, 1\}$, $n \geqslant 1$ и $C \subseteq B^n$ — равномерный код.

Кодовым расстоянием кода C назовем величину

$$d_C = \min_{\beta_1, \beta_2 \in C, \beta_1 \neq \beta_2} \rho(\beta_1, \beta_2).$$

Т. е. кодовое расстояние кода C равно наименьшему расстоянию между различными его кодовыми словами.

Коды, обнаруживающие t ошибок

Теорема 15.1. *Пусть $B = \{0, 1\}$ и $C \subseteq B^n$ — равномерный код, $n \geqslant 1$. Код C обнаруживает t ошибок замещения тогда и только тогда, когда $d_C \geqslant t + 1$.*

Коды, обнаруживающие t ошибок

Доказательство. Пусть $\beta \in C$, в слове β произошли не более t ошибок замещения, и оно перешло в слово $\beta' \in B^n$. Значит, $\beta' \in S_t(\beta)$.

Тогда можно установить, что ошибки были, в том и только в том случае, когда β' не совпадает ни с каким кодовым словом из C, не равным слову β.

Другими словами, когда никакому шару радиуса t с центром в кодовом слове из C не принадлежит никакое другое кодовое слово из C.

Т. е. когда $d_C \geqslant t + 1$.

□

Коды, обнаруживающие t ошибок

Пример. Сколько ошибок замещения может обнаружить код

$$C = \{000,\ 011,\ 101,\ 110\}?$$

Решение. Найдем кодовое расстояние кода C:

$$d_C = \min_{\beta_1,\beta_2 \in C, \beta_1 \neq \beta_2} \rho(\beta_1, \beta_2) = 2.$$

По теореме 15.1 если код C обнаруживает t ошибок, то

$$d_C = 2 \geqslant t + 1.$$

Поэтому $t \leqslant 1$.

Значит, код C может обнаружить не более одной ошибки.

Коды, обнаруживающие t ошибок

Пример. Сколько ошибок замещения может обнаружить код

$$C = \{00000,\ 10011,\ 11100,\ 01111\}?$$

Решение. Найдем кодовое расстояние кода C:

$$d_C = \min_{\beta_1,\beta_2 \in C, \beta_1 \neq \beta_2} \rho(\beta_1, \beta_2) = 3.$$

По теореме 15.1 если код C обнаруживает t ошибок, то

$$d_C = 3 \geqslant t + 1.$$

Поэтому $t \leqslant 2$.

Значит, код C может обнаружить не более двух ошибок.

Коды, исправляющие t ошибок

Теорема 15.2. *Пусть $B = \{0, 1\}$ и $C \subseteq B^n$ — равномерный код, $n \geqslant 1$. Код C исправляет t ошибок замещения тогда и только тогда, когда $d_C \geqslant 2t + 1$.*

Коды, исправляющие t ошибок

Доказательство. Пусть $\beta \in C$, в слове β произошли не более t ошибок замещения, и оно перешло в слово $\beta' \in B^n$. Значит, $\beta' \in S_t(\beta)$.

Тогда можно установить, что ошибки были и, кроме того, их исправить, в том и только в том случае, когда β' не совпадает ни с каким словом из B^n, в которое может перейти некоторое кодовое слово из C, не равное слову β, при условии, что в нем произойдет не более t ошибок замещения.

Другими словами, когда никакие два шара радиуса t с центрами в различных кодовых словах из C не пересекаются.

Т. е. когда $d_C \geqslant 2t + 1$.

□

Коды, исправляющие t ошибок

Пример. Сколько ошибок замещения может исправить код

$$C = \{000,\ 011,\ 101,\ 110\}?$$

Решение. Отметим, что $d_C = 2$.

По теореме 15.2 если код C исправляет t ошибок, то

$$d_C = 2 \geqslant 2t + 1.$$

Поэтому $t \leqslant \lfloor \frac{1}{2} \rfloor = 0$.

Значит, код C не может исправить ни одной ошибки.

Коды, исправляющие t ошибок

Пример. Сколько ошибок замещения может исправить код

$$C = \{00000,\ 10011,\ 11100,\ 01111\}?$$

Решение. Отметим, что $d_C = 3$.

По теореме 15.2 если код C исправляет t ошибок, то

$$d_C = 3 \geqslant 2t + 1.$$

Поэтому $t \leqslant 1$.

Значит, код C может исправить не более одной ошибки.

Мощность кода, исправляющего ошибки

Пусть $M_t(n)$ обозначает наибольшее число кодовых слов в коде C, $C \subseteq B^n$, исправляющем t ошибок замещения.

Теорема 15.3. *При $t \geqslant 1$, $n \geqslant 1$ справедливы следующие неравенства:*

$$\frac{2^n}{S_{2t}(n)} \leqslant M_t(n) \leqslant \frac{2^n}{S_t(n)},$$

$S_r(n)$ обозначает число наборов в шаре радиуса r из B^n.

Мощность кода, исправляющего ошибки

Доказательство. 1. *Верхняя оценка*. Пусть C, $C \subseteq B^n$, — код, исправляющий t ошибок.

Тогда по теореме 15.2 никакие два шара радиуса t с центрами в различных кодовых словах из C не пересекаются.

Поэтому

$$|C| \leqslant \frac{|B^n|}{S_t(n)} = \frac{2^n}{S_t(n)}.$$

Мощность кода, исправляющего ошибки

Доказательство. 2. *Нижняя оценка*. По индукции построим код C, $C \subseteq B^n$, исправляющий t ошибок, в котором не менее $\frac{2^n}{S_{2t}(n)}$ слов.

Базис индукции. Пусть $C_1 = \{\beta_1\}$, где β_1 — произвольное слово из B^n. Заметим, что код C_1 исправляет t ошибок.

Мощность кода, исправляющего ошибки

Доказательство. *Индуктивный переход*. Пусть уже построен код $C_k = \{\beta_1, \ldots, \beta_k\} \subseteq B^n$, исправляющий t ошибок.

Попытаемся к нему так добавить еще одно слово из B^n, чтобы получился код, исправляющий t ошибок.

По теореме 15.2 каждое слово β_i запрещает добавлять все слова из шара $S_{2t}(\beta_i)$, $i = 1, \ldots, k$.

Т. е. каждое слово из C_k запрещает $S_{2t}(n)$ слов из B^n.

Поэтому все слова из C_k запрещают не более $k \cdot S_{2t}(n)$ слов из B^n.

Значит, если $k \cdot S_{2t}(n) < |B^n| = 2^n$, то еще хотя бы одно новое слово можно добавить к коду C_k, чтобы получить код C_{k+1}, исправляющий t ошибок.

Мощность кода, исправляющего ошибки

Доказательство. Пусть построен код $C_m = \{\beta_1, \ldots, \beta_m\} \subseteq B^n$, исправляющий t ошибок, $m \geqslant 1$, для которого выполняется неравенство $m \cdot S_{2t}(n) \geqslant |B^n| = 2^n$.

Тогда положим $C = C_m$, $|C| = m$.

Тогда выполняется условие:

$$m \cdot S_{2t}(n) \geqslant 2^n,$$

а значит,

$$|C| = m \geqslant \frac{2^n}{S_{2t}(n)}.$$

□

Линейные коды

Одним из видов равномерных кодов являются *линейные коды*.

Операции над наборами

Пусть $n \geqslant 1$. Определим для наборов из B^n операции **сложения** и **умножения на число** из B.

Если $\beta, \gamma \in B^n$, то

$$\beta \oplus \gamma = (\beta_1 \oplus \gamma_1, \ldots, \beta_n \oplus \gamma_n) \in B^n.$$

Т. е. сложение наборов выполняется поразрядно.

Если $\beta \in B^n$ и $c \in B$, то

$$c\beta = (c \cdot \beta_1, \ldots, c \cdot \beta_n) \in B^n.$$

Т. е. умножение набора на число из B выполняется поразрядно.

Линейная независимость

Наборы $\beta_1, \ldots, \beta_k \in B^n$ называются **линейно независимыми**, если из равенства

$$c_1\beta_1 \oplus \ldots \oplus c_k\beta_k = (0, \ldots, 0)$$

следует

$$c_1 = \ldots = c_k = 0.$$

В обратном случае наборы $\beta_1, \ldots, \beta_k \in B^n$ называются **линейно зависимыми**.

Линейное пространство

Множество $V \subseteq B^n$ называется **линейным пространством**, если из $\beta, \gamma \in V$ следует $\beta \oplus \gamma \in V$ (считаем, что $V \neq \emptyset$).

Отметим, что если $V \subseteq B^n$ — линейное пространство, то для любого $k \geqslant 1$ для любых наборов $\beta_1, \ldots, \beta_k \in V$ и для любых $c_1, \ldots, c_k \in B$ верно

$$c_1\beta_1 \oplus \ldots \oplus c_k\beta_k \in V.$$

Базис линейного пространства

Если $V \subseteq B^n$ — линейное пространство, то наибольшее множество линейно независимых наборов из V называется его **базисом**.

Известно, что любой базис V содержит одно и то же число наборов, называемое **размерностью** пространства V.

Если $\beta_1, \ldots, \beta_k$ — базис V, то $|V| = 2^k$ и для любого набора $\beta \in V$ найдется однозначное представление:

$$\beta = c_1\beta_1 \oplus \ldots \oplus c_k\beta_k,$$

где $c_1, \ldots, c_k \in B$.

Линейный код

Пусть $n \geqslant 1$ и $C \subseteq B^n$ — равномерный код.

Код C называется **линейным**, если множество C является линейным пространством.

Кодовое расстояние линейного кода

Напомним, что для набора $\beta \in B^n$ его **весом** $|\beta|$ называется число разрядов, равных единице.

Теорема 15.4. *Если $C \subseteq B^n$ — линейный код, $n \geqslant 1$, то для его кодового расстояния d_C верно равенство:*

$$d_C = \min_{\beta \in C, \beta \neq (0,\ldots,0)} |\beta|.$$

Кодовое расстояние линейного кода

Доказательство. 1. Сначала установим, что в C найдется набор, вес которого совпадает с d_C.

По определению

$$d_C = \min_{\beta_1,\beta_2 \in C, \beta_1 \neq \beta_2} \rho(\beta_1, \beta_2).$$

Пусть кодовое расстояние достигается на паре наборов $\gamma_1, \gamma_2 \in C$, т. е. $\gamma_1 \neq \gamma_2$ и

$$d_C = \rho(\gamma_1, \gamma_2).$$

Кодовое расстояние линейного кода

Доказательство. Но C — линейный код, поэтому набор $\gamma = \gamma_1 \oplus \gamma_2$ также принадлежит коду C, т. е. $\gamma \in C$.

Кроме того, вес $|\gamma|$ набора γ равен числу разрядов, в которых наборы γ_1 и γ_2 различаются.

Значит,

$$|\gamma| = \rho(\gamma_1, \gamma_2) = d_C.$$

Кодовое расстояние линейного кода

Доказательство. 2. Теперь покажем от обратного, что в C не найдутся ненулевые наборы, вес которых меньше d_C.

Предположим, что для некоторого набора $\gamma' \in C$ верно

$$|\gamma'| < d_C.$$

Но C — линейный код, поэтому нулевой набор $(0, \ldots, 0) \in B^n$ также принадлежит коду C, т. е. $(0, \ldots, 0) \in C$.

Получаем противоречие:

$$d_C \leqslant \rho(\gamma', (0, \ldots, 0)) = |\gamma'| < d_C.$$

Значит, ненулевой набор с весом, меньшим d_C, в линейном коде C не найдется.

□

Порождающая матрица линейного кода

Пусть $C \subseteq B^n$ — линейный код и $\beta_1, \ldots, \beta_k$ — какой-то базис линейного пространства C.

Порождающей матрицей кода C назовем матрицу H_C размера $k \times n$ из нулей и единиц, строками которой являются наборы $\beta_1, \ldots, \beta_k$.

Линейный код, порожденный матрицей

Пусть H — матрица размера $k \times n$ из нулей и единиц со строками $\beta_1, \ldots, \beta_k \in B^n$.

Кодом, порожденным матрицей H, назовем линейный код $C(H)$, где

$$C(H) = \{c_1\beta_1 \oplus \ldots \oplus c_k\beta_k \in B^n \mid c_1, \ldots, c_k \in B\}.$$

Задачи для самостоятельного решения

1. Установите, сколько ошибок может обнаружить код, исправляющий t ошибок.

Литература к лекции

1. Алексеев В. Б. Лекции по дискретной математике. М.: Инфра-М, 2012. С. 56–58.

2. Гаврилов Г. П., Сапоженко А. А. Задачи и упражнения по дискретной математике. М.: Физматлит, 2004. Гл. VII 3.18–3.20, 3.25, 4.7(1).