Коммуникационные вычисления

Ф. М. Аблаев, А. Ф. Хайруллин, М. Ф. Аблаев

Аннотация

В методическом пособии излагаются основы теории коммуникационных вычислений. В основу материала пособия положен курс лекций “Коммуникационные вычисления”, читаемый студентам Института Вычислительной математики и информационных технологий Казанского федерального университета.

Оглавление

Глава 1

Детерминированные коммуникационные протоколы

1.1 Коммуникационные протоколы, сложность коммуникационных вычислений

Под коммуникационными вычислениями понимаются распределенные вычисления. Алгоритмы, определяющие коммуникационные вычисления называются протоколами. Коммуникационные вычисления булевых функций определяют следующим образом. Пусть f булева функция следующего вида $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$. Даны два вычислителя с неограниченными вычислительными возможностями (как правило один из них называют A, т.е. Алиса , а другой B, т.е. Боб), каждый из них получает на вход последовательность x и y соответственно, длиной n бит каждая. Ни один из них не знает входного набора, который получил другой вычислитель, и они совместно хотят вычислить функцию $f(x,y)$. Для вычисления значений $f(x,y)$ A и B обмениваются сообщениями(двоичными последовательностями) согласно протоколу(алгоритму) Ф. Под сложностью протокола Ф на наборах x и y понимают количество бит, передаваемых вычислителями. В теории коммуникационных вычислений исследуются различные варианты коммуникационных вычислений.Теория и техника коммуникационных вычислений имеет обширную область применений. В частности она используется при доказательстве нижних оценок времени и памяти реализации вычислений на машинах Тьюринга, в исследовании сложности вычислений для ряда других моделей вычислений.

Структуры данных, такие как частично упорядоченные полные бинарные деревья, массивы сортировки, списки и т. д. – основные объекты в алгоритмической конструкции. Были исследованы многие разновидности схемы,описанной выше, такие как: вероятностные протоколы, недетерминированные, выборочные и т.д. Более того, нижняя оценка коммуникационной сложности используется в СБИС.

Дадим формальные определения протокола вычисления булевой функции и коммуникационной сложности протокола булевой функции. В данном пособии мы будем рассматривать коммуникациионные протоколы, у которых ответ может выдавать только вычислитель B. То есть количество раундов в вычислениях нечетно.

Определение 1.1 *t-раундовым коммуникационным протоколом Φ для булевой функции $f(X,Y)$ называется алгоритм:*

1.Вычислитель A получает на вход $\sigma \in \{0,1\}^n$,вычислитель B получает $\gamma \in \{0,1\}^n$

2. Вычислитель A начинает вычисления: по σ определяет сообщение

$$m^1(\sigma) = m^1 = \{m^1_1, m^1_2, ...m^1_{t_1}\} \in \{0,1\}^{t_1}, t_1 = t_1(\sigma)$$

и передает его B.

3. Вычислитель B получает m^1. По m^1 и γ вычислитель B формирует сообщение

$$m^2 = {m^2}_1 ... {m^2}_{t_2} \in \{0,1\}^{t_2}$$

и отправляет вычислителю A сообщение m^2.

4. Вычислитель A получает m^2. Если вычислитель A по σ, m^1, m^2 может вычислить значение функции, то он выдаёт значение функции $f(\sigma,\gamma)$. В противном случае он формирует m^3 и отправляет его вычислителю B и т.д.

Определение 1.2 *Коммуникационным сообщением протокола Φ на входном наборе σ, γ называется двоичная последовательность:*

$$m = m_1, m_2, ...m_t = m_\Phi(\sigma,\gamma)$$

Определение 1.3 *Сложностью коммуникационного протокола на наборе (σ,γ) – называется количество бит, которыми обмениваются вычислители в протоколе Φ (количество бит в коммуникационном сообщении):*

$$C_\Phi(\delta,\gamma) = \left|m^1\right| + \left|m^2\right| + ... + \left|m^t\right| = |m_\Phi(\delta,\gamma)|$$

Определение 1.4 *Сложностью коммуникационного протокола* Φ *называется величина:*

$$C(\Phi) = \max_{\delta,\gamma} C_\Phi(\delta,\gamma)$$

где max берется по всем наборам $\sigma,\gamma \in \{0,1\}^n$

Определение 1.5 *Коммуникационная сложность* $C(f)$ *для функции* f *определяется как минимум по всем протоколам* Φ*, вычисляющим* f

$$C(f) = \min_{\Phi} C(\Phi)$$

Рассмотрим примеры протоколов вычислений двух известных функций

1.1.1 Примеры

EQ(x,y) Рассмотрим функцию $EQ(x,y)$ и построим протокол Φ для неё. Функция EQ равна 1, если $x = y$, и 0, в противном случае. Построим протокол Φ: вычислитель A получает на вход σ – последовательность из n бит, B — γ. Вычислитель A передаёт вычислителю B всё сообщение σ, вычислитель B, зная σ,γ производит их побитное сравнение и выдаёт ответ. Сложность такого протокола $C(\Phi) = n$, поэтому $C(EQ) \leq n$.

NEQ(x,y) Теперь рассмотрим функцию $NEQ(x,y)$ и построим протокол Φ для неё. Функция NEQ равна 1, если $x \neq y$, и 0, в противном случае. Построим протокол Φ: вычислитель A получает на вход σ – последовательность из n бит, B — γ. Вычислитель A передаёт вычислителю B всё сообщение σ, вычислитель B, зная σ,γ производит их побитовое сравнение и выдаёт ответ. Сложность такого протокола $C(\Phi) = n$, поэтому $C(NEQ) \leq n$.

Parity $Parity(z) = z_1 \oplus z_2 \oplus ... \oplus z_{2n}$. Рассмотрим два коммуникационных протокола для функции $Parity$ 1. Протокол Φ_1 для функции f: пусть вычислитель A получает на вход σ – последовательность из n бит,

B — γ. Вычислитель A передаёт вычислителю B всё сообщение σ, вычислитель B, зная σ, γ производит их побитовое сложение и выдаёт ответ. Сложность такого протокола $C_{\Phi_1}(\sigma, \gamma) = n$, поэтому $C(Parity) \leq n$.

2. Теперь для этой же функции построим ещё один протокол Φ_2, в котором оба вычислителя получают на вход те же последовательности, но вычислитель A сначала выполняет побитовое сложение последовательности σ и передаёт вычислителю B результат этого сложения. Вычислитель B, зная этот результат, выполняет побитовое сложение последовательности γ, к результату этого сложения прибавляет результат, полученный им от вычислителя A и выдаёт ответ. Сложность такого протокола $C_{\Phi_2}(\sigma, \gamma) = 1$. Таким образом, $C(Parity) \leq 1$

Исходя из определения функции $Parity$ получаем, что верхняя оценка $C(Parity) \geq 1$. Отсюда $C(Parity) = 1$

$MOD_m(X, Y)$ Рассмотрим функцию $MOD_m(X, Y)$ и построим протокол Ф. Функция MOD_m равна 1, если X равен Y по модулю m, то есть при делении на m оба числа дают одинаковый остаток. Значение функции равно 0 в противном случае. Построим протокол Ф: вычислитель A получает на вход σ – последовательность из n бит, B — γ. Вычислитель A вычисляет остаток от деления σ на m и передаёт вычислителю B, вычислитель B, зная остаток от деления σ, вычисляет его для γ, сравнивает их и выдаёт ответ. Сложность такого протокола $C(\Phi) = logm$, количество бит необходимых для передачи числа меньшего m . Таким образом сложность функции будет $C(MOD_m) \leq log(m)$

Теорема 1.1 *Для произвольной булевой функции $f(X, Y)$ выполняется $C(f) \leq n$*

Доказательство: Для произвольной булевой функции $f(X, Y)$ построим протокол Ф. Пусть на вход вычислителя A поступает набор σ, B получает γ. A генерирует следующее сообщение $m = \sigma_1, \sigma_2, \ldots, \sigma_n$. Тогда вычислитель B, получив это сообщение будет знать оба входных набора σ и γ и исходя из определения коммуникациионых вычислений может выдать ответ. В связи с произвольностью выбора функции f, получаем что $C(f) \leq C(\Phi) = n$. □

1.2 Методы доказательств нижних оценок коммуникационной сложности

Сейчас мы опишем методы доказательства нижний оценки коммуникационной сложности для функции EQ. Как уже было рассмотрено $C(EQ) \leq n$. Далее мы увидем, что $C(EQ) \geq n$.

1.2.1 Метод полных множеств «fooling set»

Продемонстрируем схему доказательства, основанного на методе полных множеств, на примере функции EQ.

Теорема 1.2 $C(EQ) \geq n$.

Доказательство: Предположим, что существует такой протокол Ф, что его сложность не более, чем $n-1$. Это означает, что вычислители могут обмениваться не более чем $2^0 + 2^1 + ... + 2^{n-1} = 2^n - 1$ сообщениями. Это предположение означает, что существует такой протокол Ф, вычисляющий EQ, что вычислители могут обмениваться не более чем $2^n - 1$ коммуникационными сообщениями, так как $C(f) \leq n-1$. Возьмём множество всех 2^n пар (σ, σ). По принципу Дирихле получаем, что существуют две пары (σ, σ) и (σ', σ'), для которых существует одинаковое коммуникационное сообщение. Понятно, что $EQ(\sigma, \sigma) = EQ(\sigma', \sigma') = 1$. Теперь рассмотрим (σ, σ'). Для неё коммуникационное сообщение будет таким же, как либо для (σ, σ), либо для (σ', σ'). (Если вычислитель A отправляет бит первым, тогда этот бит будет таким же, как и для x, или для σ'. Если вычислитель B отправляет сообщение во втором раунде, тогда его бит должен быть таким же, как у обоих до тех пор, пока он получает такой же бит от вычислителя A). Таким образом, ответ протокола на наборе (σ, σ) должен совпадать с ответом протокола на наборе (σ, σ'). Но $EQ(\sigma, \sigma') = 0$, а $EQ(\sigma, \sigma) = 1$. Следовательно, $C(EQ) \geq n$.

□

Определение 1.6 *Полное множество (fooling set или FS) для функции f, которая отображает декартово произведение $\{0,1\}^n \times \{0,1\}^n$ в множество $\{0,1\}$ – это такое множество входных наборов $S \subseteq \{0,1\}^n \times \{0,1\}^n$, что существует $b \in \{0,1\}$ такое, что*

1) для любой пары $(\sigma, \gamma) \in S, f(\sigma, \gamma) = b$

2) для любых двух отличных пар $(\sigma_1, \gamma_1), (\sigma_2, \gamma_2) \in S$, *либо* $f(\sigma_1, \gamma_2) \neq b$, *либо* $f(\sigma_2, \gamma_1) \neq b$

Теорема 1.3 *Пусть для булевой функции* f *выполняется*$|FS_f| = n$, *тогда* $C(f) \geq logn$

Определим функцию $DISJ$ "непересечения". Пусть σ, γ – характеристические вектора подмножества $\{1, 2, ..n\}$, тогда

$$DISJ(\delta, \gamma) = \begin{cases} 1, \delta \cap \gamma = 0 \\ 0, \delta \cap \gamma \neq 0 \end{cases}$$

Теорема 1.4 $C(DISJ) \geq n$

Теорема 1.5 $C(MOD_m) \geq \log m$

Теорема 1.6 $C(EQ) \geq n$

Теорема 1.7 $C(NEQ) \geq n$

Теорема 1.8 $C(Parity) \geq 1$

1.2.2 Метод монохроматических прямоугольников

Рассмотрим коммуникационную матрицу функции f,размерности $2^n \times 2^n$, обозначим её $M(f)$.

Определение 1.7 *Комбинаторный прямоугольник в матрице – это подматрица, соответствующая* $A \times B$, *где* $A \subset \{0,1\}^n, B \subset \{0,1\}^n$

Если протокол начинает работу с сообщения от первого вычислителя, длиной в один бит, тогда $M(f)$ состоит из двух прямоугольников типа $A_0 \times \{0,1\}^n, A_1 \times B^n$, где A_b – подмножество строк для каждого бита сообщения первого вычислителя. Обозначают $A_0 \cup A_1 = \{0,1\}^n$. Если следующий бит послан вторым вычислителем, тогда каждый из двух прямоугольников дальше разбивается на два меньших прямоугольника, зависящих от того, какой бит был послан. Если протокол отработал k шагов, матрица будет состоять из 2^k прямоугольников. Обозначим каждый

такой прямоугольник в секции соответственно подмножеству входных пар, для которых значение функции одинаково.

Если протокол остановил свою работу, то значение f определёно внутри каждого прямоугольника, и оно должно быть одинаковым для всех

пар x, y в прямоугольнике. Значит, множество всех коммуникационных моделей должно привести к разбиению коммуникационной матрицы на монохроматические прямоугольники (прямоугольник $A \times B$ монохроматический, если для любого x из A, и для любого y из B, значение $f(x, y)$ одинаковое).

Определение 1.8 *Монохроматическое покрытие матрицы $M(f)$ – это разбиение $M(f)$ на непересекающиеся монохроматические прямоугольники. Обозначим через $\chi(f)$ минимальное количество монохроматических прямоугольников, на которые мы можем разбить $M(f)$.*

Следующая теорема следует непосредственно из наших рассуждений, сделанных выше:

Теорема 1.9 *Если f имеет коммуникационную сложность C, тогда её монохроматическое покрытие содержит не более чем 2^C прямоугольников. Т.е. $C \geq \log \chi(f)$*

Лемма 1.1 *Если f имеет множество «fooling set» с m парами, тогда $\chi(f) \geq m$*

Доказательство: если (x_1, y_1) и (x_2, y_2) две пары из множества «fooling set», тогда они не могут быть в монохроматическом прямоугольнике, т.к. не все $(x_1, y_1), (x_2, y_2), (x_1, y_2), (x_2, y_1)$ имеют одно и то же значение f. $\square$

1.2.3 Метод ранг коммуникационной матрицы

Теперь мы дадим оценку коммуникационной сложности $\chi(f)$ в терминах ранга коммуникационной матрицы. Напомним, что ранг матрицы в поле F — это максимальное число линейно независимых строк или столбцов. Для ранга матрицы существует и другое определение:

Определение 1.9 *Ранг матрицы M размера $n \times n$ – это минимальное значение l такое, что M может быть представлена в виде $M = \sum_{i=1}^{l} \alpha_i B_i$, где $\alpha_i \in F \backslash \{0\}$ и каждая матрица B_i — это $n \times n$ матрица ранга 1.*

Теорема 1.10 *Для любой функции* $f, \chi(f) \geq rank(M(f))$

Доказательство: каждый монохроматический прямоугольник может быть представлен как матрица ранга не больше чем 1. □

1.3 Многораундовые коммуникационные вычисления

Определение 1.10 *Многораундовыми или* $t-$*раундовыми коммуникационными вычислениями булевой функции* $f(X,Y)$ *называется алгоритм:*

1.Вычислитель A *получает на вход* $\sigma \in \{0,1\}^n$*,вычислитель* B *получает* $\gamma \in \{0,1\}^n$

2. Вычислитель A *начинает вычисления: по* σ *определяет сообщение*

$$m^1(\sigma) = m^1 = \{m^1_1, m^1_2, ...m^1_{t_1}\} \in \{0,1\}^{t_1}, t_1 = t_1(\sigma)$$

и передает его B*.*

3. Вычислитель B *получает* m^1*. Если по* m^1 *вычислитель* B *может вычислить значение функции, то он выдаёт значение функции* $f(\sigma,\gamma)$ *по* m^1 *и* γ*. В противном случае он формирует сообщение*

$$m^2 = {m^2}_1...{m^2}_{t_2} \in \{0,1\}^{t_2}$$

и отправляет вычислителю A *сообщение* m^2*.*

4. Вычислитель A *получает* m^2*. Если вычислитель* A *по* σ, m^1, m^2 *может вычислить значение функции, то он выдаёт значение функции* $f(\sigma,\gamma)$*. В противном случае он формирует* m^3 *и отправляет его вычислителю* B *и т.д.*

Согласно рассматриваемой модели коммуникационных вычислений, булева функция $f(X,Y)$ вычисляется с нечетным количеством раундов. В следующем разделе мы рассмотрим одноразундовые коммуникационные вычисления.

1.3.1 Односторонние коммуникационные вычисления (Однораундовые)

Определение 1.11 *Однораундовыми коммуникационными вычислениями булевой функции $f(X,Y)$ называется алгоритм:*

1.Вычислитель A получает на вход $\sigma \in \{0,1\}^n$,вычислитель B получает $\gamma \in \{0,1\}^n$

2. Вычислитель A начинает вычисления: по σ определяет сообщение

$$m^1(\sigma) = m^1 = \{m_1^1, m_2^1, ...m_{t_1}^1\} \in \{0,1\}^{t_1}, t_1 = t_1(\sigma)$$

и передает его B.

3. Вычислитель B получает m^1. Если по m^1 вычислитель B может вычислить значение функции, то он выдаёт значение функции $f(\sigma,\gamma)$ по m^1 и γ.

Рассмотрим функцию $f(x,y)$, которой соответствует коммуникационная матрица $CM(f)$. Обозначим через $nrow(CM(f)$ число различных строк матрицы $CM(f)$. Для удобства будем полагать, что $nrow(CM(f) = 2^l$. Через $C_1(f)$ обозначим одностороннюю коммуникационную сложность функции f.

Теорема 1.11 *Для любой булевой функции, зависящей от n переменных, выполняется $C_1(f) = \log nrow(CM(f))$.*

Доказательство: Докажем, что $C_1 \leq \log nrow(CM(f))$. Рассмотрим коммуникационную матрицу, где $1, 2, \ldots, 2^l$ – группы, внутри каждой группы строки одинаковые (это возможно сделать, так как от перестановки строк матрица не изменится). Строим протокол Ф: кодируем номер каждой группы:

$$\begin{aligned} &m_1 \to 00...00 \\ &m_2 \to 00...01 \\ &\vdots \\ &m_{2^l} \to 11...11 \end{aligned}$$

где l – длина кода.

Вычислитель A определяет номер группы, в которую попал входной набор σ и передает закодированный номер вычислителю B. Вычислитель

B получает на вход набор γ и, зная номер группы выдает ответ $f(\sigma, \gamma)$. Сложность этого протокола Ф: $C_1(f) = l$. Теперь докажем обратное неравенство: воспользуемся принципом Дирихле и методом от противного. Предположим, что $C_1 \leq lognrow(CM(f))$. Так как $C_1(\Phi) = \min C_1(\Phi)$, где Ф вычисляет f.

Построим протокол Φ_1, такой что односторонняя коммуникационная сложность $C_1 \leq t-1$, а $M = \{m_1 m_2 ... m_d\}$ – множество всех различных сообщений. Отсюда $d \leq 2^{t-1}$. Кодируем

$$\left.\begin{array}{l} m_1 \to 00...00 = \bar{m}_1 \\ m_2 \to 00...01 = \bar{m}_2 \\ \vdots \\ m_d \to 11...11 = \bar{m}_{2^{t-1}} \end{array}\right\} 2^{t-1}$$

Построим по Φ_1 протокол Φ_2, который будет работать следующим образом: Если в Φ_1 вычислитель передавал вычислителю B сообщение m_1, то в протоколе Φ_2 вычислитель A передает вычислителю B сообщение $\bar{m}_1$.

Очевидно, что протокол Φ_2 реализует ту же функцию, что и протокол Φ_1. Снова перекодируем строки коммуникационной матрицы:

Существует группа $\bar{m}_i$, такая что в ней есть 2 различные строки, то есть существует функция f, для которой $f(\sigma, \gamma) \neq f(\sigma', \gamma')$. Получаем, что A передает B одно и то же сообщение $\bar{m}_i$ на разных наборах вычислителя A: σ и σ'протокол вычисляет функцию неправильно. Отсюда следует, что наши предположения неверны и не существует протокола вычисляющего булеву функцию со сложностью меньшей $\log nrow(CM(f))$. □

1.3.2 Сравнительная сложность трех- и однораундовых коммуникационных вычислений

Рассмотрим трехраундовые коммуникационные вычисления.

Определение 1.12 *Трехраундовыми коммуникационными вычислениями булевой функции $f(X, Y)$ называется алгоритм:*

1.Вычислитель A получает на вход $\sigma \in \{0,1\}^n$,вычислитель B получает $\gamma \in \{0,1\}^n$.

2. Вычислитель A начинает вычисления: по σ определяет сообщение

$$m^1(\sigma) = m^1 = \{m_1^1, m_2^1, ...m_{t_1}^1\} \in \{0,1\}^{t_1}, t_1 = t_1(\sigma)$$

и передает его B.

3. Вычислитель B начинает вычисления: по сообщению m^1 и γ определяет сообщение

$$m^2(\gamma, m^1) = m^2 = \{m_1^2, m_2^2, ...m_{t_2}^2\} \in \{0,1\}^{t_2}, t_2 = t_2(\gamma, m^1)$$

и передает его A.

4. Вычислитель A по сообщению m^2 и σ определяет сообщение

$$m^3(\sigma, m^2) = m^3 = \{m_1^3, m_2^3, ...m_{t_2}^3\} \in \{0,1\}^{t_3}, t_3 = t_3(\sigma, m^2)$$

и передает его B.

5. Вычислитель B получает m^3. Если по m^3 и m^1 вычислитель B может вычислить значение функции, то он выдаёт значение функции $f(\sigma, \gamma)$ по m^3, m^1 и γ.

Проверим даёт ли использование нескольких раундов передачи сообщений преимущество в коммуникационной сложности.

ISA(X,Y) Рассмотрим функцию $ISA(X,Y), |X| = |Y| = n$, определяемую следующим образом

$$ISA(X,Y) = \begin{cases} x_i, \text{если Y содержит единственную единицу на i-ом месте} \\ 0, \text{иначе} \end{cases}$$

Теорема 1.12 $C_1(ISA) = n$

*Доказательство:*Построим коммуникационную матрицу для данной функции, увидим что $nrow(M) = 2^n$, так как для всякого набора σ строки будут различны, таким образом $C_1(ISA) \geq n$. Равенство получаем из теоремы о верхней оценки коммуникационной сложности для произвольной функции. □

Теорема 1.13 $C_3(ISA) \leq \log n + 1$

Доказательство: Для трехраундового коммуникационного вычисления можно построить протокол Φ с коммункационной сложностью $C_3(ISA) = \log n + 1$. Принцип работы протокола Φ следующий. На первом раунде вычислитель A передаёт пустое сообщение, затем вычислитель B просматривает входной набор γ, если в нём присутствует ровно одна единица, то вычислителю A передаётся номер этого символа, иначе выдается ответ 0. Вычислитель A, получив i, находит соответсвующий бит своего входного набора и отправляет его значениие σ_i вычислителю B. Таким образом сложность этого протокола равна $C(\Phi) = \log n + 1$, а значит коммуникационная сложность функции $C_3(ISA) \leq \log n + 1$. □

Теорема 1.14 $C(ISA) \geq \log n$

Далее покажем, что увеличение количества раундов, не всегда даёт улучшение в коммуникационной сложности. Рассмотрим уже известную функцию $EQ(X, Y)$.

Теорема 1.15 $C_1(EQ) = C_t(EQ) = n$, *для любого* $t \geq 1$

Глава 2

Недетерминированные коммуникационные протоколы

2.1 Недетерминированные коммуникационные вычисления

В данном разделе будем рассматривать следующую модель коммуникационных вычислений. Имеются два вычислителя A и B, им на вход подаются два входных набора σ и γ, соответственно. Вычислитель A, на основе входного набора строит множество возможных сообщений $M(\sigma) = \{m^1, \ldots, m^t\}$, которые можно отправить B, где $t = t(\sigma)$ также зависит от входа. Далее вычислитель выбирает сообщение из этого множества $m \in M(\sigma)$ и отправляет его B. Механизм выбора отправляемого сообщения недетерминирован (не определен). Вычислитель B, получив сообщение m и зная входной набор γ, если может выдать ответ, выдаёт значение функции. Иначе строит своё множество возможных сообщений $M(m, \gamma) = \ldots$ и также недетерминированно выбирает сообщение $m' \in M(m, \sigma)$ и отправляет его вычислителю B. Вычисления продолжаются до тех пор пока B не сможет выдать ответ. В следующем определении рассмотрим односторонний недетерминированный протокол.

Определение 2.1 *Протокол* Φ *недетерминированно вычисляет функцию* $f(\sigma, \gamma)$*, если:*

1. *Для любого входного набора* (σ, γ)*, такого что* $f(\sigma, \gamma) = 1$*, существует такое сообщение* m_i^σ*, что вычислитель* A *передает* m_i^σ

вычислителю B, а B выдает единицу.

2. *Для любого входного набора (σ, γ), такого что $f(\sigma, \gamma) = 0$ и для любого сообщения $m_i^\sigma \in \{m_1^\sigma ... m_t^\sigma\}$, которое вычислитель A передает B, вычислитель B выдает 0.*

Обозначим через Ф протокол, который недетерминированно вычисляет функцию f. Из определения получаем, что на входных наборах, на которых функция принимает значение 1, у протокола Ф существует такое сообщение, на котором протокол Ф выдаёт 1. А на тех входных наборах, на которых функция принимает значение 0, протокол Ф всегда выдаёт 0.

Далее дадим определение сложности недетерминированного коммуникационного протокола. На произвольном входном наборе (σ, γ) сложность протокола будет вычисляться следующим образом. Пусть $M(\sigma, \gamma) = \{m^1(\sigma, \gamma), m^2(\sigma, \gamma), \ldots, m^t(\sigma, \gamma), \}$ множество всех сообщений протокола Ф на этом входном наборе. Тогда сложность протокола для данного входа

$$C(\Phi(\sigma, \gamma)) = \max_{m \in M(\sigma, \gamma)} |m|$$

Определение 2.2 *Сложностью недетерминированного коммуникационного протокола* Ф *называется величина:*

$$C(\Phi) = \max_{(\sigma, \gamma)} C(\Phi(\sigma, \gamma)$$

Определение 2.3 *Недетерминированной сложностью булевой функции f называется величина:*

$$NC(f) = \min_{\Phi} C(\Phi)$$

Теорема 2.1 *Для произвольной булевой функции $f(X, Y)$ выполняется следующее равенство $NC(f) = NC_1(f)$.*

Доказательство: Проведём доказательство в два этапа. Сначала покажем, что $NC_1(f) \geq NC(f)$, это следует из определения. Так как односторонние коммуникационные вычисления, это частный случай коммуникационных вычислений. Теперь надо доказать, что $NC_1(f) \leq NC(f)$. Для этого возьмем недетерминированный протокол Ф, который вычисляет

функцию f с минимальной (наилучшей) сложностью, то есть $NC(f) = C(\Phi)$. По этому протоколу Φ построим односторонний недетерминированный протокол Φ_1 с той же коммуникационной сложностью, $C(\Phi) = C(\Phi_1)$. Если мы построим такой протокол Φ_1, тогда получим что $NC_1(f) \leq C(\Phi_1) = C(\Phi) = NC(f)$ и докажем теорему. $\square$

Теперь рассмотрим примеры недетерминированных протоколов для некоторых булевых функций.

Теорема 2.2 *Недетерминированная коммуникационная сложность функции NEQ:*

$$NC_1(NEQ) \leq \log n + 1$$

Доказательство: Построим для недетерминированный коммуникационный протокол для функции NEQ:

1. Вычислитель A недетерминированно выбирает i-тый бит δ_i набора δ и пересылает B номер i и значение δ_i.

2. Вычислитель B сравнивает $\delta_i = \gamma_i$, если они не равны, то выдает 1, иначе 0.

Проверим корректность данного протокола, правильно ли вычисляет наш протокол функцию NEQ в недетерминированном смысле. В соответствии с определением функции и недетерминированных коммуникационных вычислений:

1. $f(\delta, \gamma) = 1 \Rightarrow$ существует m_i^δ, такое что на наборе γ вычислитель B выдаст 1, иначе говоря есть i, для которого $\delta_i \neq \gamma_i$. Таким образом если A пошлёт i, δ_i, тогда B выдаст правильный ответ.

2. $f(\delta, \gamma) = 0 \Rightarrow \delta_i = \gamma_i \forall i \in [1, n]$, то есть для любого i вычислитель A посылает i и δ_i и B, зная весь δ, на любом сообщении будет выдавать 0, так как $\delta = \gamma$

Значит алгоритм корректен и сложность функции $NC(NEQ) \leq \log n + 1$. $\square$

Отметим, что в детерминированном случае, как было доказано ранее,

сложность функции NEQ будет $C(NEQ) = n$. То есть недетерминированность уменьшает сложность коммуникационных вычислений. Однако, на коммуникационную сложность некоторых функций недетерминированность никак не сказывается.

Теорема 2.3 *Недетерминированная сложность функции EQ:*

$$NC_1(EQ) \geq n$$

Теорема 2.4 *Недетерминированная сложность функции ISA:*

$$NC_1(ISA) \geq \log n + 1$$

2.1.1 Классы сложности и отношения между ними

Обозначим через $F_n = \{f_n(x_1, \ldots, x_n, y_1, \ldots, y_n)\}$ множество булевых функций от $2n$ переменных. Определим несколько классов сложности булевых функций.

Определение 2.4 *Класс* $P-CC_1 = \{f \in F_n$ *: такие что существует детерминированный односторонний коммуникационный протокол* Φ*, вычисляющий* f *и сложность этого протокола* $C(\Phi) \leq (\log n)^k$*, где* $k \geq 0\}$

Определение 2.5 *Класс* $NP-CC_1 = \{f \in F_n$ *: такие что существует недетерминированный односторонний коммуникационный протокол* Φ*, вычисляющий* f *и сложность этого протокола* $NC(\Phi) \leq (logn)^k$*, где* $k \geq 0\}$

Определение 2.6 *Класс* $co-NP-CC_1 = \{g \in F_n$ *: такие что отрицание этой функции* $\neg g \in NP-CC_1\}$

Теорема 2.5

$$P-CC_1 \subset NP-CC_1$$

Доказательство: $P-CC_1 \subseteq NP-CC_1$, так как любой детерминированный коммуникационный протокол, есть частный случай недетерминированного коммуникационного протокола. С другой стороны в предыдущем разделе было показано, что $NEQ \in NP-CC_1 \setminus P-CC_1$. □

2.2 Обобщения модели $k-$ вычислителей

Существует несколько способов обобщить рассмотренную ранее коммуникационную модель на модели, в которую входит более двух вычислителей. Рассмотрим наиболее интересную: « number on the forehead ». Она заключается в математической головоломке, в ходе которой людей собирают в одной комнате, у каждого человека на голове есть бит, который могут видеть все остальные, а он сам не видит. Фактически: существует функция $f : (\{0,1\}^n)^k \to \{0,1\}$ и входной вектор $(x_1, x_2...x_k)$, где $x_i \in \{0,1\}^n$, $i-$ый игрок может видеть все x_j, $i \neq j$. Как и в случае двух игроков, у игроков есть фиксированный коммуникационный протокол, и все это общение основано на принципе « public blackboard». В завершении протокола все участники должны знать $f(x_1...x_k)$.

Пример 1 *Рассмотрим вычисление функции*

$$f(x_1x_2x_3) = \bigoplus_{i=1}^{n} maj(x_{1i}, x_{2i}, x_{3i})$$

в модели с тремя участниками, где x_1, x_2, x_3 вектора размерности n бит. Коммуникационная сложность этой функции =3: каждый игрок читая число i, может определить большинство из x_{1i}, x_{2i}, x_{3i} , рассматривая биты доступные ему. Он записывает $\oplus$ сумму этих чисел на доску, и конечный ответ это $\oplus$ из битов игроков. Этот протокол верен так как большинство из каждого ряда известно 1-ему или 3-ему игроку (для нечетного номера).

Пример 2 *Функция «inner product» (IP)*

$$IP_{n,k} = \bigoplus_{i=1}^{n} \bigwedge_{j=1}^{k} x_{ij} \tag{2.1}$$

Заметим, что к=2 ведет к mod 2 внутреннему делению произведению примера 5.

В модели с двумя вычислителями мы ввели понятие монохромотических прямоугольников, чтобы доказать нижнюю оценку. Для $k-$вычислительной машины мы будем использовать цилиндрические пересечения.

Определение 2.7 *Цилиндр в i измерениях — это множество S входных значений, таких что если $(x_1...x_k) \in S$, тогда для всех x_i мы получим $(x_1, ..., x_{i-1}, x_i, x_{i+1}, ..., x_k)$ также $\in S$.*

Определение 2.8 *Цилиндрическое пересечение — это* $\bigcap_{i=1}^{k} T_i$*, где* T_i *— цилиндр в* i *измерениях.*

Как было замечено в случае двух вычислителей, коммуникационный протокол может быть рассмотрен как путь преобразований матрицы $M(f)$. Здесь $M(f)$ это $k-$размерный куб, i передача не зависит от x_i. Тем не менее мы заключаем, что если у f есть multiparty протокол, который передает бит, тогда у его матрицы есть разбиение, которое использует более 2^c монохромотических цилиндрических пересечений.

Лемма 2.1 *Если каждое преобразование* $M(f)$ *в монохроматические цилиндрические пересечения занимает по меньшей мере* R *цилиндрических пересечений, тогда* $k-$*вычислительная коммуникационная сложность* $\leq \log_2 R$*.*

Глава 3

Вероятностные коммуникационные протоколы

3.1 Вероятностные однораундовые коммуникационные вычислителения

В этом разделе будем рассматривать следующую модель коммуникационных вычислений. Даны два вычислителя A и B, на вход подаются два входных набора σ и γ, соответственно. Вычислитель A, на основе входного набора строит множество возможных сообщений $M(\sigma) = \{m^1, \ldots, m^t\}$, которые можно отправить B, где $t = t(\sigma)$ также зависит от входа. Далее вычислитель выбирает сообщение из этого множества $m \in M(\sigma)$ и отправляет его B. Механизм выбора отправляемого сообщения вероятностный, то есть имеется генератор случайных чисел, и вычислитель A, в соответствии, с полученной случайной величиной выбирает равновероятно одно из сообщений. Вычислитель B, получив сообщение m и зная входной набор γ, если может выдать ответ, выдаёт значение функции. Иначе строит своё множество возможных сообщений $M(m, \gamma) = \ldots$ и также вероятностно выбирает сообщение $m' \in M(m, \sigma)$, затем отправляет его вычислителю B. Вычисления продолжаются до тех пор пока B не сможет выдать ответ. Если для обоих вычислителей используется один генератор случайных чисел, тогда эта модель называется вероятностными коммуникационными вычислениями с открытым ключом "public coin". Если у каждого вычислителя свой генератор, то модель с закрытым ключом "private coin".

Определение 3.1 *Будем говорить, что протокол* Φ $\frac{1}{2}$ *вычисляет функцию* $f(\sigma,\gamma)$*, если :*

1. *Для любого входного набора* (σ,γ)*:* $f(\sigma,\gamma)=1$*, вероятность принять входной набор* $\Pr_{accept}(\delta,\gamma)>\frac{1}{2}$
2. *Для любого входного набора* (σ,γ)*:* $f(\sigma,\gamma)=0$*, вероятность принять входной набор* $\Pr_{accept}(\delta,\gamma)\leq\frac{1}{2}$

Определение 3.2 *Будем говорить, что протокол* Φ $\left(\frac{1}{2}+\varepsilon\right)$ *вычисляет функцию* $f(\sigma,\gamma)$*, если существует такое* $\varepsilon\in\left(0,\frac{1}{2}\right]$*, что:*

1. *Для любого входного набора* (σ,γ)*:* $f(\sigma,\gamma)=1$*, вероятность принять входной набор* $\Pr_{accept}(\delta,\gamma)>\frac{1}{2}+\varepsilon$
2. *Для любого входного набора* (σ,γ)*:* $f(\sigma,\gamma)=0$*, вероятность принять входной набор* $\Pr_{accept}(\delta,\gamma)\leq\frac{1}{2}+\varepsilon$

Определение 3.3 *Вероятностной коммуникационной сложностью функции* $f(\sigma,\gamma)$ *называется величина*

$$PCC_1(f)=\min C(\Phi)$$

сложность наилучшего протокола, который $\Phi-\frac{1}{2}$ *вычисляет функцию* $f(x,y)$

$$RCC_1(f)=\min C(\Phi)$$

сложность наилучшего протокола, который $\Phi-\left(\frac{1}{2}+\varepsilon\right)$ *вычисляет функцию* $f(x,y)$

Свойство 1

$$CC(f)\geq RCC_{\frac{1}{2}+\varepsilon}(f)\geq PCC_{\frac{1}{2}}(f)$$

Теперь рассмотрим пример вероятностного протокола для известной нам функции EQ. Построим вероятностный протокол Φ, который с большой вероятностью правильного ответа вычисляет функцию $EQ(x,y)$, используя $O(\log n)$ битов при коммунникации. Идея построения подобного вероятностного алгоритма принадлежит Р.В.Фрейвалду

Входы трактуются как два натуральных числа x и y, $0 \leq x, y \leq 2^n - 1$. Вычислитель P_x выбирает (равновероятностно) простое число $p \leq n^2$, вычисляет $x' = x \pmod p$ и пересылает пару (x', p) вычислителю P_y. Вычислитель P_y вычисляет $y' = y \pmod p$ и сравнивает y' с x'. Если $y' \neq x'$, то вычислитель P_y выдает ответ $x \neq y$. Если $y' = x'$, то вычислитель P_y выдает ответ $x = y$.

Теперь посчитаем вероятности возможных ошибок. Если два числа x, y равны, то x', y' равны, и протокол Ф выдаст правильный ответ. Если два числа x, y различны, то тем не менее может оказаться, что $y' = x'$, и протокол Ф выдаст неверный ответ. Это может произойти, если p является делителем числа $x - y$. Заметим, что $|x - y| < 2^n$, следовательно $x - y$ может иметь не более n различных простых делителей. С другой стороны вычислитель P_x выбирает простые числа среди $O(\frac{n^2}{\log n})$ простых чисел, таким образом вероятность выбора делителя числа $x - y$ очень мала.

3.2 Оценки сложности коммуникационных вычислений

3.2.1 Энтропийная оценка

В данном параграфе вводится новая характеристика булевой функции — *коммуникационная структура*, основанная на понятии теста коммуникационной матрицы. Приведем содержательные соображения, лежащие в основе энтропийного метода доказательства нижней оценки коммуникационной сложности.

Чем сложнее структра коммуникационной матрицы (чем она гуще) тем меньше мощность теста по отношению к числу различных строк матрицы CM_f. Чем регулярнее структра коммуникационной матрицы (чем она режее), тем больше мощность теста по отношению к числу различных строк матрицы CM_f. Чем сложнее структура коммуникационной матрицы CM_f (коммуникационная структура булевой функции f), тем меньше возможность сэкономить при переходе от детерминированного вычисления к вероятностному в зависимости от велчины требумой надежности (вероятности правильного ответа). Приведенные интуитивные

соображения лежат в основе определяемого ниже понятия детерминированной и вероятностной коммуникационной характеристики булевой функции и основанного на этом понятии энтропиного метода доказательства.

Определение 3.4 *Для булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *положим* $dcc(f,n) = \frac{ts(f,n)}{dim(f,n)}$. *Назовем* $dcc(f,n)$ *детерминированной коммуникационной характеристикой булевой функции* f.

Наряду с обозначением $dcc(f,n)$ мы будем использовать обозначение $dcc(f)$. В последующем, когда будем рассматривать разбиение $pat(n)$ входов функции f, отличное от традиционного, будем использовать обозначение $dcc(f, pat(n))$.

Из приведенного оределения и свойства следует, что $dcc(f,n) = \frac{ts(f,n)}{nrow(CM_f)}$, т.е. детерминированная коммуникационная характеристика булевой функции f является характеристикой структуры ее коммуникационной матрицы.

Определение 3.5 *Д Для булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для числа* $p \in \left[\frac{1}{2}, 1\right]$, *положим* $pcc_p(f,n) = dcc(f,n)h(p)$, *где* $h(p) = -p\log p - (1-p)\log(1-p)$ *— это Шенноновская эптропия. Величину* $pcc_p(f,n)$ *будем называть* p*-вероятностной коммуникационной характеристикой булевой функции* f.

Наряду с обозначением $pcc_p(f,n)$ мы будем использовать обозначение $pcc_p(f)$ и обозначение $pcc(f)$. В последующем, когда будем рассматривать разбиение $pat(n)$ входов функции f, отличное от традиционного, будем использовать обозначение $pcc(f, pat(n))$.

Из определения и свойства следует следующее свойство.

Теорема 3.1 *Для произвольной булевой функции* f *выполняется*

1. $1 \le dcc(f) \le \frac{2^n}{n}$.
2. $h(p) \le pcc(f) \le \frac{2^n h(p)}{n}$ *для* $p \in [1/2, 1]$

Для произвольной булевой функции $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для произвольного* $\varepsilon \in (0, 1/2]$, $p = \frac{1}{2} + \varepsilon$ *выполняется*

$$PC_p(f) \ge DC(f)(1 - pcc_p(f)) - 1.$$

Доказательство. Обозначим ϕ вероятностный протокол p-вычисляющий булеву функцию f. Пусть множество X — это множество представителей функции f, а множество Y — это тест для функции f. Заметим, что $|X| = dim(f)$.

На множестве X зададим случайную велечину ξ равномерным распределением вероятностей $P(\xi = x) = 1/dim(f)$.

На множестве M_ϕ сообщений протокола ϕ зададим случайную велечину θ условным распределением вероятностей $P(\theta = m/\xi = x) = Pr($*вычислитель* P_0 *посылает сообщение* m *при условии, что на вход* P_0 *была подана последовательность* $x)$.

Пользуясь определением и свойствами Шенноновской энтропии и количества информации, все используемые в данном доказательстве свойства энтропии и количества информации приведены в первой главе этой же книги) мы можем записать следующие соотношения

$$\begin{aligned} I(\xi,\theta) = H(\theta) - H(\theta/\xi) \le H(\theta) &\le \log dim(\phi) && (3.1)\\ &\le C_\phi, && (3.2)\\ I(\xi,\theta) = H(\xi) - H(\xi/\theta) &= \log dim(f) - H(\xi/\theta) && (3.3)\\ &\ge DC(f) - 1 - H(\xi/\theta). && (3.4) \end{aligned}$$

Для доказательства утверждения теоремы достаточно доказать, что

$$H(\xi/\theta) \le ts(f)h(p). \tag{3.5}$$

Обозначим $t = ts(f)$. Для каждой входной последовательности $x \in X$ определим характеристическую двоичную последовательность $b(x) = b_1 b_2 \ldots b_t$ следующим образом: для $b_i = 1(0)$ тогда и только тогда, когда $f(x, y_i) = 1(0)$ для $y_i \in Y,\ i \in \{1, 2, \ldots, t\}$.

Обозначим $B(X) = \{b(x)/x \in X\}$.

На множестве $B(X)$ определим случайный вектор $\pi = (\pi_1, \pi_2, \ldots, \pi_t)$ следующим распределением вероятностей: $P(\pi = b(x)) = P(\xi = x)$. Из определения случйного вектора π имеем

$$H(\xi/\theta) = H(\pi/\theta) \le \sum_{i=1}^{t} H(\pi_i/\theta).$$

Таким образом для доказательства неравенства (3.5) достаточно доказать, что для всех $i \in \{1, 2, \ldots, t\}$ выполняется неравенство

$$H(\pi_i/\theta) \leq h(p). \tag{3.6}$$

Рассмотрим случайные велечины η_i, $i \in \{1, 2, ..., t\}$,принимающие значения 0 и 1 и заданные распределением вероятностей $P(\eta_i = 1/\theta = m) = Pr($*протокол ϕ выдает значение 1 при условии, что вычислитель P_0 послал сообщение m и на вход вычислителя P_1 подана входная последовательность y_i*$)$. В силу определения случайная велечина η_i зависит лишь от значения случайной велечины θ, поэтому в силу соответсвующего свойства энтропии имеем

$$H(\pi_i/\theta) = H(\pi_i/\theta, \eta_i).$$

Далее применяя свойства функции энтропии получаем

$$H(\pi_i/\theta) = H(\pi_i/\theta, \eta_i) \leq H(\pi_i/\eta_i). \tag{3.7}$$

В силу определения энтропии получаем

$$H(\pi_i/\eta_i) = P(\eta_i = 1)H(\pi_i/\eta_i = 1) + P(\eta_i = 0)H(\pi_i/\eta_i = 0).$$

Положим $\omega = P(\pi_i = 1/\eta_i = 1)$, $\omega' = P(\pi_i = 0/\eta_i = 0)$. Используя введенное обозначение, мы можем записать последнее равентсво в следующем виде

$$H(\pi_i/\eta_i) = P(\eta_i = 1)h(\omega) + P(\eta_i = 0)h(\omega').$$

Далее, используя свойство функции h получаем

$$H(\pi_i/\eta_i) \leq h(P(\eta_i = 1)\omega + P(\eta_i = 0)\omega') = h(P(\pi_i = \eta_i)). \tag{3.8}$$

Вероятность $P(\pi_i = \eta_i)$, $i \in \{1, 2, ..., t\}$ — это вероятность следующего события: коммуникационный протокол ϕ выдает правильный результат, если вычислитель P_0 получает входнуюпоследовательность $x \in X$, а вычислитель P_1 получает входную последовательность $y_i \in Y$. Так как по условию теоремы протокол ϕ p-вычисляет функцию f, то для всех $i \in \{1, 2, \ldots, t\}$ выполняется

$$P(\pi_i = \eta_i) \geq p.$$

Функция $h(p)$ монотонно убывает от 1 до 0, когда p меняется от $1/2$ до 1. Таким образом для всех $i \in \{1, 2, \ldots, t\}$ выполняется

$$h(P(\pi_i = \eta_i)) \leq h(p).$$

Таким образом последнее неравенство и неравенства (3.8), (3.7) и (3.6) доказывают неравенство (3.5). Применяя неравенство (3.5) совместно с неравенствами (3.2), (3.4), получаем утверждение теоремы. □

Теорема 3.2 *Для произвольной булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для произвольного* $\varepsilon \in (0, 1/2]$, $p = \frac{1}{2} + \varepsilon$ *выполняется*

$$dim_p(f) \geq dim(f)^{(1-pcc_p(f))}.$$

Доказательство. т В отличие от теоремы окончательное утверждение получается применением неравенства (3.5) совместно с неравенствами (3.1), (3.3). □

Заметим, что несколько менее точную оценку теоремы можно получить и как непосредственное следствие утверждения теоремы **??**, так как для произвольной функции f ее вероятностная p-размерность $dim_p(f)$ связана с величиной $PC_p(f)$ p-коммуникационной сложности соотношем $2^{PC_p(f)} \geq dim_p(f) > 2^{PC_p(f)-1}$.

Используя разложение функции энтропии $h(1/2+\varepsilon)$ в ряд Тейлора в окрестности точки $\frac{1}{2}$, имеем:

$$h(1/2 + \varepsilon) \;=\; 1 \;-\; \varepsilon^2(2/\ln 2) \;+\; \cdots$$

В качестве следствия теоремы, используя разложение функции $h(1/2+\varepsilon)$ в ряд Тейлора в окрестности точки $\frac{1}{2}$, получаем:

Свойство 2 *Пусть для булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *выполняется* $dcc(f,n) = 1$. *Пусть* $\varepsilon(n) \in (0, 1/2]$ *и* $\varepsilon(n) \to 0$ *при* $n \to \infty$, *пусть* $p(n) = 1/2 + \varepsilon(n)$. *Тогда*

$$PC_{p(n)}(f,n) \geq O(DC(f,n)\varepsilon^2(n)).$$

В качестве следующего следствия теоремы, и факта, что $h(p) \sim (1-p)\log(1-p)^{-1}$ при $p \to 1$ получаем:

Свойство 3 *Пусть для булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *выполняется* $dcc(f,n) = 1$. *Пусть* $er(n) \in [0,1/2)$, *и* $er(n) \to 0$ *при* $n \to \infty$, *пусть* $p(n) = 1 - er(n)$. *Тогда*

$$PC_{p(n)}(f,n) \geq DC(f,n) - O(DC(f,n)er(n)\log er(n)^{-1}).$$

3.2.2 Топологическая оценка

Теорема 3.3 *Для произвольной булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для произвольного* $\varepsilon \in (0,1/2]$, $p = 1/2 + \varepsilon$ *выполняется*

$$PC_p(f) \geq \log(DC(f) - 1) - \log\log(1 + 1/\varepsilon).$$

Введем необходимые понятия из теории множеств и метрических пространств.

Пусть $\mathcal{S}$ — это метрическое пространство с метрикой ρ. Конечное множество элементов $s_1, s_2, \ldots, s_t$ пространства $\mathcal{S}$ называется ε-цепью, если $\rho(s_i, s_{i+1}) < \varepsilon$ для $i \in \{1, 2, \ldots, t-1\}$. Говорят, что элементы s_1 и s_t соединимы ε-цепью.

Подмножество $\mathcal{C}_\varepsilon$ пространства $\mathcal{S}$ называется ε-компонентой пространства $\mathcal{S}$, если два любых элемента $s, s' \in \mathcal{C}_\varepsilon$ соединимы ε-цепью. Метрическое подпространство $\mathcal{S}'$ пространства $\mathcal{S}$ называется ограниченным, если существует такая константа c, что для произвольных двух элементов $s, s' \in \mathcal{S}'$ выполняется $\rho(s, s') \leq c$. Для произвольного $\varepsilon > 0$ ограниченное подпространство конечномерного векторного пространства разбивается на конечное число своих ε-компонент.

В d-мерном векторном пространстве R^d определим метрику ρ следующим образом. Для элементов $\mu = (p_1, p_2, ..., p_d)$ и $\mu' = (p'_1, p'_2, ..., p'_d)$ пространства R^d положим

$$\rho(\mu, \mu') = \sum_{i=1}^{d} \mid p_i - p'_i \mid .$$

Пусть $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ — произвольная булева функция, а Φ — односторонний вероятностный протокол, p-вычисляющий функцию f, $p = 1/2 + \varepsilon$, $\varepsilon \in (0, 1/2]$. Пусть $dim(\Phi) = d$. Обозначим R^d_Φ подпространство пространства R^d, состоящее из всевозможных распределений вероятностей сообщений протокола Φ.

$$R^d_\Phi = \left\{\mu \in R^d \ / \ \mu = \mu(x), x \in \{0,1\}^n\right\}.$$

В силу определения метрическое подпространство R^d_Φ ограничено.

Мы докажем нижнюю оценку для p-коммуникационной размерности булевой функции, которая связана с величиной p-коммуникационной сложности свойством ?? (см. стр. ??).

Теорема 3.4 *Для произвольной булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для произвольного* $\varepsilon \in \left(0, \frac{1}{2}\right]$, $p = 1/2 + \varepsilon$ *выполняется*

$$dim_p(f) \geq \frac{\log dim(f)}{\log(1 + 1/\varepsilon)}.$$

Доказательство. Пусть Φ — это произвольный односторонний вероятностный протокол, p-вычисляющий функцию f. Для произвольных двух слов x, x' из множества представителей X функции f точки $\mu(x) = \{p_1(x), p_2(x), \ldots, p_d(x)\}$ и $\mu(x') = \{p_1(x'), p_2(x'), \ldots, p_d(x')\}$ принадлежат различным 2ε-компонентам пространства R^d_Φ.

Действительно, предположим, что существует 2ε-компонента $\mathcal{C}_{2\varepsilon} \in R^d_\Phi$ такая, что $\mu(x), \mu(x') \in \mathcal{C}_\varepsilon$. Положим $x_1 = x$, а $x_2 = x'$. Пусть точки $\mu(x_1), \mu(x_2), \ldots, \mu(x_t)$ образуют 2ε-цепь. Последнее означает, что для $i \in \{1, 2, \ldots, t-1\}$ выполняется

$$\rho(\mu(x_i), \mu(x_{i+1})) < 2\varepsilon. \tag{3.9}$$

Применяя последнее неравенство, получаем, что для произвольного слова y из множества Y тест функции f выполняется

$$\mu(x)\nu(y) - \mu(x')\nu(y) \leq \sum_{i=1}^{d} \mid p_i(x) - p_i(x') \mid q_i(y) \leq \rho(\mu(x), \mu(x')) < 2\varepsilon. \tag{3.10}$$

В силу условия теоремы 3.3 вероятностный протокол имеет надежность ε, поэтому для произвольных последовательностей $u, v \in \{0,1\}^n$ выполняется либо $\mu(u)\nu(v) \geq \frac{1}{2} + \varepsilon$, либо $\mu(u)\nu(v) \leq \frac{1}{2} + \varepsilon$. Содержательно $\mu(u)\nu(v)$ — это вероятность выдачи 1 протоколом Φ на входе uv.

Из сказанного определения вероятностного коммуникационного протокола и соотношений следует, что $f(x, y) = f(x', y)$ для всех последовательностей $y \in \{0,1\}^n$. Это противоречит тому, что $x, x' \in X$.

Оценим теперь число K 2ε-компонент пространства R_f^d. Нам достаточно оценить K следующим образом. В каждую 2ε-компоненту $\mathcal{C}_{2\varepsilon}$ пространства R_f^d поместим сферу радиуса ϵ с центром в соответствующей точке $\mu(x)$, $x \in X$. Все эти сферы могут пересекаться разве что по границе. Пространство R_f^d вместе со своими сферами радиуса ϵ поместим в большую сферу радиуса $1+\epsilon$ с центром в точке $(0, 0, \ldots, 0)$.

Объем сферы радиуса r в пространстве R^d равен cr^d, где константа c зависит от используемой метрики ρ. Таким образом справедлива оценка

$$K \leq \frac{c(1+\epsilon)^d}{c\epsilon^d} = \left(1 + \frac{1}{\epsilon}\right)^d.$$

Так как $K \geq dim(f)$, то теорема доказана. □

Так как для произвольной функции f ее вероятностная p-размерность $dim_p(f)$ связана с величиной $PC_p(f)$ p-коммуникационной сложности соотношением $2^{PC_p(f)} \geq dim_p(f)$, то из теоремы и того, что $\log dim(f) > DC(f) - 1$ следует утверждение теоремы.

3.2.3 Геометрическая оценка

Теорема 3.5 *Для произвольной булевой функции* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, *для* $p = 1/2$ *выполняется*

$$PC_p(f) \geq \log(DC(f) - 1) - \log\log ts(f).$$

В последующей части излагается доказательство теоремы, которое состоит из двух утверждений. Введем необходимые определения и обозначения.

Обозначим $\mathbf{R}^d$ d-мерное Евклидово пространство. $(d-1)$-мерная гиперплоскость

$$a_1 z_1 + a_2 z_2 + \cdots + a_d z_d = b$$

делит $\mathbf{R}^d$ на две связные области. Условимся, что эти две связные области определяются следующими двумя неравенствами

$$a_1 z_1 + a_2 z_2 + \cdots + a_d z_d > b$$

и

$$a_1 z_1 + a_2 z_2 + \cdots + a_d z_d \leq b.$$

Обозначим $k(d,t)$ максимальное число связных областей, которые могут быть образованы в d-мерном Евклидовом пространстве $\mathbf{R}^d$ t $(d-1)$-мерными гиперплоскостями. Следующий факт установлен О.Б.Лупановым.

Свойство 4 *Если* $d = 1$ *и* $t \geq 1$, *тогда* $k(d,t) = t+1$.
Если $d \geq 2$ *и* $t \geq 2$, *тогда* $k(d,t) \leq t^d$.

Для полноты изложения приведем доказательство леммы.

Доказательство. 1. Если $d = 1$, тогда прямая разбивается t различными точками (0-мерными гиперплоскостями) на $k(d,t) = t+1$ части.

2. Доказательство второй части проведем по индукции числа t гиперплоскостей. Легко видеть, что для произвольного $d \geq 2$ выполняется $k(d,2) \leq 2^d$.

Пусть $t > 2$ и $d > 2$. Рассмотрим t произвольных $(d-1)$-мерных гиперплоскости α_1, α_2, ..., α_t. $t-1$ гиперплоскости α_1, α_2, ..., α_{t-1} могут определить в пространстве $\mathbf{R}^d$ самое большее $k(d,t-1)$ различных связных области. Обозначим $D_1, D_2, ..., D_l$ эти области $(l \leq k(d,t-1))$.

Гиперплоскость α_t можно рассматривать как $(d-1)$-мерное Евклидово пространство. Число связных областей в пространстве α_t, определяемое гиперплоскостями α_1, α_2, ..., α_{t-1} равно числу связных областей в пространстве α_t, образованное пересечением α_t с гиперплоскостями α_1, α_2, ..., α_{t-1} (каждое такое пересечение является $(d-2)$-мерной гиперплоскостью). Следовательно это число не превосходит $k(d-1,t-1)$.

Каждое из этих связных областей пространства α_t лежит в некоторой области D_i. Следовательно

$$k(d,t) \leq k(d,t-1) + k(d-1,t-1).$$

По предположению индукции для произвольного $d \geq 2$ имеем $k(d,t-1) \leq (t-1)^d$. Поэтому, для произвольного $d \geq 3$ выполняется

$$k(d,t) \leq (t-1)^d + (t-1)^{d-1} \leq t^d.$$

В случае $d = 2$ мы имеем $k(d,t) \leq (t-1)^2 + (t-1) + 1 \leq t^2 = t^d$. Таким образом для $d \geq 2$ мы получаем $k(d,t) \leq t^d$. □

3.3 Сложность индивидуальных функций. Иерархии сложности

В данном параграфе мы определим две булевы функции и далее их модификации. Эти две функции обладают различными коммуникационными характеристиками и, как следствие этого имеют принципиально различные коммуникационные свойства.

Всюду в этом параграфе для простоты будем будем считать, что число n имеет вид $n = 2^k$, где k — целое число (обобщение на общий случай производится легко).

Первая функция f_1, определяемая условием $f_1(x, y) = 1$ тогда и только тогда, когда $x = y$, хорошо известна. Эту функцию в англоязычной литературе называют функция равенство. Непосредственно из определения функции равенство следует, что ее коммуникационная матрица CM_1 — это единичная матрица размерности $2^n \times 2^n$.

Следующее свойство непосредственно следует из вида коммуникационной матрицы CM_1.

Свойство 3.3.1 *Для функции равенство f_1 выполняются следующие равенства:*

1. $ts(f_1) = 2^n - 1$.
2. $DC(f_1) = n$.
3. $dcc(f_1) = \frac{2^n-1}{n}$.
4. *Для $p \in [1/2, 1]$ выполняется* $pcc_p(f_1) = \frac{2^n-1}{n} h(p)$.

Теорема 3.3.1 *Для произвольного $\varepsilon \in (0, 1/2)$, $p = 1/2 + \varepsilon$ выполняется*

$$\log n - \log\log(1 + 1/\epsilon) - 1 \ \leq PC_p(f_1) \ \leq 4 \log n.$$

Доказательство. Верхняя оценка для $PC_p(f_1)$ хорошо известна и основана на методе "малых простых чисел предложенного Р.Фрейвалдом [38] для построения вероятностной машины Тьюринга, распознающей симметрию. Для случая вероятностных односторонних коммуникационных протоколов этот метод интерпретирован в обзоре [71]. Для произвольного $\varepsilon \in (0, 1/2)$, $p = 1/2 + \varepsilon$ в обзоре [71] предложен вероятностный протокол Φ, p-вычисляющий функцию f_1, такой что

$$PC_{\Phi}(n) \leq 4 \log n.$$

Нижняя оценка теоремы непосредственно следует из теоремы 3.3 и свойства 3.3.1. □

Теорема 3.3.1 позволяет сделать следующие выводы.

- При фиксированном $\varepsilon \in (0, 1/2)$ нижняя оценка теоремы 3.3 является точной.
- Теоремы 3.5 и **??** дают худшие (тривиальные) по сравнению с теоремой 3.3 нижние оценки вероятностной коммуникационной сложности.

Определим вторую функцию $f_2(x, y)$. Введем следующие обозначения. Пусть Z_1 — это следующее подмножество множества $\{0,1\}^n$. $Z_1 = \{x(i)/i \in \{1, 2, \ldots, n\}\}$, где $x(i)$, $i \in \{1, 2, ..., n\}$, это двоичное слово, i-ый бит которого есть 1, а остальные биты — 0. Обозначим S подмножество $\{0,1\}^n \times Z_1 \bigcup Z_1 \times \{0,1\}^n$ множества $\{0,1\}^n \times \{0,1\}^n$.

Положим

$$f_2(x,y) = \begin{cases} \bigvee_{i=1}^{n} x_i \wedge y_i, & \text{если } x, y \in S \\ 0, & \text{иначе} \end{cases}$$

Из определения функции f_2 следует:

- коммуникационная матрица CM_2 — это матрица размерности $2^n \times 2^n$,
- все 2^n строк матрицы CM_2 попарно различны,
- множество Z_1 является тестом для функции f_2 (тестом для 2^n строк матрицы CM_2).

Из сказанного следует следующее свойство.

Свойство 3.3.2 *Для функции f_2 выполняется*

1. $ts(f_2) = n$.
2. $DC(f_2) = n$.

3. $dcc(f_2) = 1$.

4. Для $p \in [\frac{1}{2}, 1]$ *справедливо* $pcc_p(f_2) = h(p)$.

Теорема 3.3.2 *Существует 2-х раундовый детерминированный протокол* ϕ*, вычисляющий булеву функцию* f_2 *такой, что*

$$C_\phi(n) \leq 2\log n + 1.$$

Доказательство. Конструкция коммуникационного протокола ϕ проста.

Пусть x, y — это входные последовательности протокола ϕ.

Первый раунд. Вычислитель P_0 посылает вычислителю P_1 номер m_i i-ого бита входной последовательности x, равного единице, *если* $x \in Z_1$. Вычислитель P_0 посылает вычислителю P_1 соответсвующее сообщение m_0 *если* $x \notin Z_1$.

Второй раунд. Вычислитель P_1, получив сообщение m_i от вычислителя P_0, либо выдает ответ сам (в случае $i \neq 0$, а также в случае $i = 0$ и $y \notin Z_1$), либо передает вычислителю P_0 сообщение m_i, *если* $y \in Z_1$. □

Опишем вероятностный односторонний протокол Ф, вычисляющий функцию f_2.

Если $x \in Z_1$, $x = x(i)$, $i \in \{1, 2, \ldots, n\}$ *тогда* протокол Ф функционирует следующим образом. Вычислитель P_0 посылает сообщение $x \in Z_1$ и сообщение i вычислителю P_1. После получения этой информации вычислитель P_1 выдает верный ответ с вероятностью 1.

Если $x \notin Z_1$, *тогда* протокол Ф функционирует следующим образом. Предположим для простоты, что $z = 2^t$, где t — это целое число (данный протокол легко обобщается на общий случай).

Вычислитель P_0 делит входную последовательность x на z равных частей длины n/z каждая.Затем вычислитель P_0 равновероятно выбирает одну из частей $x^{(j)}$, $j \in \{1, 2, \ldots, z\}$ входной последовательности x и посылает вычислителю P_1 сообщение, содержащее информацию $x \notin Z_1$, число j и подпоследовательность $x^{(j)}$.

Вычислитель P_1 функционирует следующим образом. P_1 проверяет свою входную последовательность y.

Если $y \notin Z_1$,*тогда* P_1 выдает правильный ответ с вероятностью 1.

Если $y \in Z_1$, $y = y(i)$, $i \in \{1, 2, ..., n\}$, $x^{(j)}$ содержит i-ый бит в общей нумерации битов последовательности x и этот бит равен 1, *тогда* вычислитель P_1 выдает ответ $b = 1$, *иначе* с вероятностью $q = 1/2 - 1/(4z - 2)$

вычислитель P_1 выдает ответ $b = 1$, а с вероятностью $1 - q$ выдает ответ $b = 0$.

Из описания протокола Φ имеем

$$Pr(\Phi \; outputs \; b = 1 \; when \; f_2(x, y) = 1) \; \geq \; 1/z + (1 - 1/z)q \; = \; 1/2 + 1/(4z - 2),$$

$$Pr(\Phi \; outputs \; b = 0 \; when \; f_2(x, y) = 0) \; \geq \; 1 - q \; = \; 1/2 + 1/(4z - 2).$$

Очевидно следующее свойство.

Свойство 3.3.3 *Для булевой функции f_2, для $p = 1/2 + 1/(4z - 2)$ выполняется*

$$PC_p(f_2, n) \; \leq \; n/z \; + \log z \; + \; 2.$$

Теорема 3.3.3 *Для булевой функции f_2, для $p = 1/2$ выполняется*

$$\log n - \log \log n - 1 \; \leq \; PC_p(f_2) \; \leq \; \log n \; + \; 3.$$

Доказательство. Верхняя оценка теоремы следует из свойства 3.3.3 для случая $z = n$. В данном случае часть x_i входной последовательности x — тривиальна и равна в точности i-му биту последовательности x.

Из теоремы 3.5 и свойства 3.3.2 следует нижняя оценка. □

Теорема 3.3.3 позволяет сделать следующие выводы.

- Оценка теоремы 3.5 — точна.
- Применение нижних оценок теорем **??**, 3.3 совместно со свойством 3.3.2 показывает, что оценка теоремы 3.5 является наилучшей в данном случае.

Теорема 3.3.4 *Для булевой функции f_2, для константы $z \geq 1$, для $p = 1/2 + 1/(4z - 2)$ выполняется*

$$n(1 - h(p)) - 1 \; \leq \; PC_p(f_2) \; \leq \; n/z \; + \log z \; + \; 2.$$

Доказательство. Верхняя оценка доказана в свойстве 3.3.3.

Из свойства 3.3.2 имеем, что $pcc_p(f_2) = h(p)$. Из теоремы **??** следует нижняя оценка. □

Теорема 3.3.4 позволяет сделать следующие выводы.

- Оценка теоремы **??** точна.
- Применение нижних оценок теорем 3.5, 3.3 совместно со свойством 3.3.2 показывают, что оценка теоремы **??** является наилучшей в данном случае.

Далее в утверждении 3.3.2 доказывается зависимость вероятностной односторонней коммуникационной сложности от величины надежности вероятностного вычисления. Результат такого типа является первым результатам подобного рода для коммуникационной сложности булевых функций.

Для последовательности $\{f(x,y) : x,y \in \{0,1\}^n\}$ единообразно определяемых функций, чисел $p, p' \in [0,1]$, будем писать $PC_p(f) < PC_{p'}(f)$, если для некоторого n_0, для всех $n \geq n_0$ выполняется $PC_p(f,n) < PC_{p'}(f,n)$.

Следствие 3.3.1 *Для булевой функции f_2, для числа $t \geq 2$, для $z(t,n) = \lceil n^{1/t} \rceil$, $p(t,n) = 1/2 + 1/(4z(t,n)-2)$ выполняется*

$$O(n^{(1-2/t)}) \;\leq\; PC_{p(t,n)}(f_2) \;\leq\; O(n^{(1-1/t)}).$$

Доказательство. Верхняя оценка следует из теоремы 3.3.4.

Нижняя оценка. В силу свойства 3.3.2 мы можем применить следствие **??** (стр. **??**) энтропийной оценки (теоремы **??**).

$$PC_{p(t,n)}(f_2) \geq O(n/(4z(t,n)-2)^2) = O(n^{(1-2/t)}).$$

□

Для последовательности булевых функций $\{f\}$ и для функций $p(n), p'(n) \in \{0,1\}$, будем писать $PC_{p(n)}(f) \prec PC_{p'(n)}(f)$, если $PC_{p(n)}(f,n)/PC_{p'(n)}(f,n) \to 0$, при $n \to \infty$.

Следствие 3.3.2 *Существует бесконечная последовательность чисел* $2 < t_1 < t_2 < \cdots < t_i < \cdots$ *такая, что для каждого* $i \geq 1$ *выполняется*

$$PC_{p(t_i,n)}(f_2) \prec PC_{p(t_{i+1},n)}(f_2).$$

В заключительной части параграфа рассматриваются две последовательности функций $\{f_1^g(x,y) : x,y \in \{0,1\}^n\}$ и $\{f_2^g(x,y) : x,y \in \{0,1\}^n\}$, которые демонстрируют неожиданный факт. Детерминированно функции $f_1^g(x,y)$ сложнее функций $f_2^g(x,y)$, а вероятностно — наоборот функции $f_2^g(x,y)$ сложнее функций $f_1^g(x,y)$. Этот экзотический факт объясняется влиянием коммуникационной характеристики функции на величину вероятностной коммуникационной сложности.

Пусть $g : \{1,2,\ldots\} \to \{1,2,\ldots\}$ — это целочисленная функция, удовлетворяющая требованию $g(n) \leq n$ для всех $n \geq 1$. Пусть $k(n) = n - g(n)$.

Определим функцию $f_1^g(x,y)$ следующим образом. $f_1^g(x,y) = 1$ тогда и только тогда, когда $x = y$ и первые $g(n)$ битов x — нули. Эта функция подобна функции f_1 и обладает аналогичными с функцией f_1 свойствами.

Свойство 3.3.4 *Для функции* $f_1^g(x,y)$ *справедливы следующие соотношения*

1. $ts(f_1^g) = 2^{k(n)}$.
2. $DC(f_1^g) = k(n)$.
3. $dcc(f_1^g) = 2^{k(n)}/k(n)$.
4. *для* $\varepsilon \in (0,1/2)$, *для* $p = 1/2 + \varepsilon$ *выполняется*

$$\log(k(n)) - \log\log(1+1/\epsilon) - 1 \leq PC_p(f_1^g) \leq 4\log(k(n)).$$

Определим функцию $f_2^g(x,y)$ подобно функции $f_2(x,y)$ следующим образом. Пусть $Z_1^g = \{x(i) \in Z_1 / i \in \{1,2,\ldots,k(n)\}\}$, и $S^g = \{0,1\}^n \times Z_1^g$ Положим

$$f_2^g(x,y) = \begin{cases} \bigvee_{i=1}^n x_i \wedge y_i, & \text{если } x,y \in S^g \\ 0, & \text{иначе} \end{cases}$$

Непосредственно из определения функции $f_2^g(x,y)$ видно, что она обладает следующими свойствами подобными свойствам функции $f_2^g(x,y)$

Свойство 3.3.5 *Для $k(n) = n - g(n)$, для функции f_2^g справедливы следующие соотношения.*

1. $ts(f_2^g) = k(n)$.
2. $DC(f_2^g) = k(n)$.
3. $dcc(f_2^g) = 1$,
4. *Для $p \in (1/2, 1)$ выполняется* $pcc_p(f_2^g) = h(p)$.
5. *для $p = 1/2$ выполняется*

$$\log k(n) - \log\log k(n) - 1 \leq PC_p(f_2^g) \leq 2\log k(n) + 1.$$

6. *Для $z \geq 2$, для $p = 1/2 + 1/(4z - 2)$ выполняется*

$$k(n)(1 - h(p)) - 1 \leq PC_p(f_2^g) \leq k(n)/z + \log z + \ 2.$$

7. *Существует 2-раундовый детерминированный протокол ϕ, вычисляющий функцию f_2^g, такой, что*

$$C_\phi(n) \leq 2\log k(n) + 1.$$

Из свойств 3.3.4 и 3.3.5 следуют следующее утверждение.

Теорема 3.3.5 *Для целочисленных функций $g(n) = n - \lceil n^{1/4} \rceil$ и $g'(n) = n - \lceil n^{1/2} \rceil$ выполняется*

$$DC(f_2^g) \prec DC(f_1^{g'}),$$

но для произвольного фиксированного $p \in (1/2, 1)$ выполняется

$$PC_p(f_2^g)0 \succ PC_p(f_1^{g'}).$$

Доказательство. Для $k(n) = n - g(n) = \lceil n^{1/4} \rceil$ и $k'(n) = n - g'(n) = \lceil n^{1/2} \rceil$ имеем

1. $DC(f_2^g) = k(n)$.

2. Для $z \geq 2$, для $p = 1/2 + 1/(4z-2)$ выполняется

$$k(n)(1-h(p)) - 1 \leq PC_p(f_2^g) \leq k(n)/z + \log z + \ 2.$$

Далее

1. $DC(f_1^{g'}) = k'(n)$.
2. Для $\varepsilon \in (0, 1/2)$, для $p = 1/2 + \varepsilon$ выполняется

$$\log(k'(n)) - \log\log(1+1/\varepsilon) - 1 \leq PC_p(f_1^{g'}) \leq 4\log(k'(n)).$$

□

3.4 Вероятностная сложность почти всех функций

Обозначим $\mathbf{F}(n,n)$ множество всех булевых функций от $2n$ переменных. Пусть E — это некоторое свойство функций из $\mathbf{F}(n,n)$. Обозначим $\mathbf{F}^E(n,n)$ подмножество функций из $\mathbf{F}(n,n)$, не обладающих свойством E.

Будем говорить, что почти все функции из множества $\mathbf{F}(n,n)$ обладают свойством E, если

$$\frac{|\mathbf{F}^E(n,n)|}{|\mathbf{F}(n,n)|} \to 0 \quad (\text{при } \ n \to \infty).$$

Лемма 3.4.1 *Почти все функции из* $\mathbf{F}(n,n)$ *обладают следующими свойствами.*

1. $DC(f) = n$.
2. *Для произвольного* $\theta \in (0,1)$ *выполняется* $n \leq ts(f) < (2+\theta)n$.

Доказательство. Пусть $N = 2^n$. Обозначим $\mathbf{M}(N \times N)$ множество $N \times N$ булевых матриц.

1. Достаточно показать, что почти все матрицы из множества $\mathbf{M}(N \times N)$ имеют попарно различные строки. Пусть в множестве $\mathbf{M}(N \times N)$ задано равномерное распределение вероятностей. Пусть $M \in \mathbf{M}(N \times N)$ — это случайно выбранная матрица. Тогда для произвольных $i, j \in \{1, 2, ..., N\}$, $i \neq j$, выполняется.

$$Pr(\textit{в матриц } M \textit{ строка } i \textit{ равен строке } j) = 2^{-N},$$

$$Pr(\textit{существует две одинаковые строки в матрице} M) \leq C_N^2 2^{-N}.$$

2. Так как для почти всех функций $DC(f) = n$, то для почти всех функций справедливо $n \leq ts(f)$.

   Далее выберем произвольное $\theta \in (0, 1)$, пусть $t = (2 + \theta)n$. Достаточно показать, что для почти все матрицы из $\mathbf{M}(N \times N)$ имеют $N \times t$ подматрицы $M^{(t)}$ с N попарно различными строками.

   Пусть $M \in \mathbf{M}(N \times N)$ — это случайная матрица. Пусть $M_1^{(t)}$ — подматрица матрицы M, образованная N строками и первыми t столбцами. Для произвольных $i, j \in \{1, 2, ..., N\}$, $i \neq j$, выполняется.

$$Pr(\textit{в матрице } M_1^{(t)} \textit{ строка } i \textit{ равна строке } j) = 2^{-t},$$

$$Pr \quad (\textit{не менее, чем две строки в матрице } M_1^{(t)} \textit{ совпадают}) \leq C_N^2 2^{-t} < N^2 2^{-t}.$$

□

Замечание Используя более точный счет, можно уточнить значение константы $(2 + \theta)$. Но для дальнейшего использования нам достаточно доказанного утверждения. Отметим, что вопрос о мощности (длине) минимального теста для почти всех таблиц исследовался подробно различными авторами. В книге [86] приводятся результаты (с соответсвующими ссылками) о мощности минимальных тестов для различных типов таблиц.

В качестве следствия из леммы 3.4.1 и из следствия 3 теоремы **??** мы имеем следующую теорему.

Теорема 3.4.1 *Для почти всех функций из* $\mathbf{F}(n,n)$*, для* $er(n) \to 0$*,* $p(n) = 1 - er(n)$ *выполняется*

$$PC_{p(n)}(f,n) \geq n - O(n\ er(n) \log er(n)^{-1})$$

Яо [84] Доказал, что для почти всех функций из $\mathbf{F}(n,n)$, для произвольной фиксированной константы $\varepsilon \in (0,1/2)$, для $p = 1/2 + \varepsilon$ выполняется

$$PC_{p(n)}(f,n) \geq n - \log n - 2.$$

Оценка теоремы 3.4.1 при $en(n) \prec 1/n$ точнее оценки Яо.

Оценка теоремы 3.4.1 на порядок точнее оценки Яо при переходе к рассмотрению коммуникационной размерности вместо коммуникационной сложности.

Литература

[1] Аблаев Ф.М. Возможности вероятностных машин по представлению языков в реальное время. — Известия ВУЗов, Математика, 1985, вып. 7, с. 32-40.

[2] Аблаев Ф.М. Влияние степени изолированности точки сечения на число состояний вероятностного автомата. — Математические заметки, 1988, т.44. вып. 3, с. 289-297.

[3] Аблаев Ф.М. Сравнительная сложность представления языков в вероятностных автоматах. — Кибернетика, 1989, с. 21-26.

[4] Аблаев Ф.М. О сравнительной сложности вероятностных и детерминированных автоматов. — Дискретная математика, 1991, т. 3, вып. 2, с. 114-120.

[5] Аблаев Ф.М. Нижние оценки для вероятностной односторонней коммуникационной сложности булевых функций. — Тезисы докладов X-ой международной конференции по проблемам теоретической кибернетики. В сборнике Методы исистемы технической диагностики, Саратовский университет, 1993, вып. 18, с. 3.

[6] Агафонов В.Н. Сложность алгоритмов и вычислений. — Новосибирск, изд. Новосибирского университета, 1975.

[7] Александров П.С. Введение в теорию множеств и общую топологию. — М.: Наука, 1977.

[8] Асарин Е.А. О сложности равномерных приближений непрерывных функций. — УМН, 1984, т. 39, вып. 3, с. 157-170.

[9] Проблемы Гильберта. Сборник под общей редакцией П.С. Александрова. М.: Наука, 1969.

[10] Арнольд В.И., О функциях трех переменных. — ДАН, 1957, т. 114, вып. 4, с. 679-681.

[11] Ахо А., Ульман Д. Сложность вычислений в СБИС. — М: Мир, 1991.

[12] Ахо А., Хопкрофт Д., Ульман Д. Построение и анализ вычислительных алгоритмов. — М: Мир, 1979.

[13] Барздинь Я.М. Сложность распознавании симметрии на машинах Тьюринга. — Проблемы кибернетики, 1965, вып 15, с. 245-248.

[14] Барздинь Я.М., Трахтенброт Б.А. Конечные автоматы. Поведение и синтез. — М.: Наука, 1970.

[15] Бухараев Р.Г., Захаров В.М. Управляемые генераторы случрйных кодов. — Казань, изд. Казанского университета, 1978.

[16] Витушкин А.Г. К тринадцатой проблеме Гильберта. — ДАН, 1954, т. 95, вып. 4, с. 243-250.

[17] Витушкин А.Г. Оценка сложности задачи табулироваеия. — М.: Наука, 1959.

[18] Габбасов Н.З. Замечание об одной оценке, относящейся к теореме редукции Рабина. — Рукопись деп. в ВИНИТИ 25 февраля 1988, N 1532-В88

[19] Галлагер Р. Теория информации и надежная связь. — М.: Мир, 1972

[20] Гашков С.Б. Сложность приближенного вычисления действительных чисел, непрерывных функций и линейных функционалов. — автореферат на соискание ученой степени доктора физико-математических наук, МГУ, Москва, 1992.

[21] Кузьмин В.А. Реализация функций алгубры логики автоматами, нормальными алгоритмами и машинами Тьюринга. — Проблемы кибернетики, 1979, вып. 36, с. 181-194.

[22] Колмогоров А.Н. О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного и сложения. — ДАН, 1957, т. 114, вып. 5, с. 953-956.

[23] Колмогоров А.Н. Оценки минимального числа элементов ε-сетей в различных функциональных классах и их применение к вопросу о представимости функций нескольких переменных суперпозициями функций меньшего числа переменных. — ДАН, 1955, т. 101, вып. 2, с. 192-194.

[24] Колмогоров А.Н. Различные подходы к оценке трудности приближенного задания и вычисления функций. — В сб.: Proceedings of the International Congress of Mathrmaticians 1962, Stockholm, 1963, p. 352-356.

[25] Колмогоров А.Н., Тихомиров В.М. ε-энтропия и ε-емкость множеств и в функциональных пространствах. — УМН, 1959, 14:2, с. 3-86.

[26] Кочкарев Б.С. Об утсойчивости вероятностных автоматов. — Кибернетика, 1968, вып 2, с. 95-111.

[27] Лупанов О.Б. О возможности синтеза схем из произвольных элементов. — Труды Математического Института имени В.А.Стеклова АН СССР, Москва, 1958, том 51, с. 158-173.

[28] Лупанов О.Б. Асимптотические оценки сложности управляющих систем. — М.: изд. Московского университета, 1984.

[29] Леу К., Мур Э.Ф., Шеннон К.Э., Шапиро Н. Вычислимость на вероятностных машинах. — в кн.: "Автоматы". М.: Мир, 1956, с. 242-278.

[30] Марченков С.С. Об одном методе анализа суперпозиции непрерывных функций. — Проблемы кибернетики, 1980, вып. 37, с. 5-17.

[31] Нигматуллин Р.Г. Сложность булевых функций. — М.: Наука, 1991.

[32] Никольский С.М. Ряды Фурье функций с данным модулем непрерывности. — ДАН, 1946, т. 52, с. 191-194.

[33] Офман Ю.П. О приближенной реализации непрерывных функций на автоматах. — ДАН, 1963, т. 152, вып. 4, с. 823-826.

[34] Покровская И.А. Некоторые оценки числа состояний вероятностных автоматов, представляющих регулярные языки. — Проблемы кибернетики, 1979, вып 36, с. 181-194.

[35] Рабин М. Вероятностные автоматы. — Кибернетический сборник, 1964, вып. 9, с. 123-141.

[36] Тиман А.Ф. Теория приближений функций действительного переменного. — М.: Наука, 1960.

[37] Тихомиров В.М. Некоторые вопросы теории приближений. — М.: изд. Московского университета, 1976.

[38] Фрейвалд Р.В. Ускорение распознавания некоторых множеств применением датчика случайных чисел. — Проблемы Кибернетики, 1979, вып. 36, с. 209-224.

[39] Фрейвалд Р.В. Об увеличении числа состояний при детерминизации конечных вероятностных автомтатов. — Автоматика и вычислительная техника, 1982, 3, с. 39-42.

[40] Фрейвалд Р.В. Сложность вычислений на вероятностных и детерминированных машинах односторонних машинах Тьюринга. — Кибернетика и вычислительная техника, 1986б вып. 2, с. 147-179.

[41] Храпченко В.М. Об одном методе получения нижних оценок сложности П-схем. — Математические заметки, 1971, т. 10, вып. 1, с. 83-92.

[42] Чегис И.А., Яблонский С.В. Логические способы контроля работы электрических схем. — Труды Математического Института имени В.А.Стеклова АН СССР, Москва, 1958, том 51, с. 270-360.

[43] Яблонский С.В. Об алгоритмических трудностях синтеза минимальных контактных схем. — Проблемы кибернетики, 1959, вып. 2, с. 75-121.

[44] Яблонский С.В. Введение в дискретную математику. — М: Наука, 1986.

[45] Ablayev F., Freivalds R. Why sometimes probabilistic algorithms can be more effective. — in Proc. of the MFCS'86, Lecture Notes in Computer Science, 1986, 233, p. 1-14.

[46] Ablayev F. Possibilities of probabilistic one-way counting machines. — in Proc. of the FCT'87, Lecture Notes in Computer Science, 1987, 278, p. 1-4.

[47] Ablayev F. The complexity properties of probabilistic automata with isolated cut point. — Theoretical Computer Science, 1988, 57, p. 87-95.

[48] Ablayev F. On Comparing Probabilistic and Deterministic Automata Complexity of Languages. — in Proc. of the MFCS'89, Lecture Notes in Computer Science, 1989, v. 379, p. 599-605.

[49] Ablayev F. Lower bounds for probabilistic space complexity: automata approach. — University of Rochrster (USA), Technical Report 423, May 1992, p. 1-13.

[50] Alayev F. Lower bounds for one-way probabilistic communication complexity. — in Proc. of the ICALP'93, Lect. Lecture Notes in Computer Science, 1993, v. 700, p. 241-252.

[51] Aho A., Ulman J., Yanakakis M. On notion of information transfer in VLSI circuits. — in Proc. of the 15th Annual ACM Symposium on the Theory of Computing, 1983, p. 133-139.

[52] Chor B., Goldreich O. Unibased bits from sources of weak randomness and probabilistic communication complexity. — Siam J. Comput. 1988, v. 17, 2, p. 230-260.

[53] Gill J. Computational complexity of probabilistic Turing machines. — SIAM J. Comput., 1977, 6, p. 675-695.

[54] Phan Dinh Dieu On a Necessary Condition for Stochastic Languages. — Elektronische Informationsverarbeitung und Kybernetik, 1972, v. 8, p. 575-588.

[55] Duris P., Galil Z., Schnitger G. Lower bounds on Communication Complexity. — in Proc. of the 16th Annual ACM Symposium on the Theory of Computing, 1984, 81-89.

[56] Gamal A., Orlitsky A. Communication complexity. — in Complexity in Information Theory. Editor Yaser S. Abu-Mostafa, Springer-Verlag, 1988, p. 16-61.

[57] Freivalds R. Fast Probabilistic Algorithms. — in Proc. of the Conference Mathematical Foundation of Computer Science 1979, Lect. Notes in Comput. Science, 1979, v. 74, p. 57-69.

[58] Freivalds R. Probabilistic two-way machines. — in Proc. of MFCS'81, Lecture Notes in Computer Science, 1981, 118, p. 33-45.

[59] Freivalds R. Complexity of Probabilistic Versus Deterministic Automata. — Baltic Computer Science Selected Papers, Lecture Notes in Computer Science, 1991, v. 502, p. 565-613.

[60] Goldman M., Hastad J., Razborov A. Majority gates vs. general weighted threshold gates. — in Proc. of 7-th Anual conf. Structure in Complexity Theory, 1992, p. 2-13.

[61] Halstenberg B., Reischuk R. On Different Modes of Communication. — in Proc. of the 20th Annual ACM Symposium on the Theory of Computing, 1988, p. 162-172.

[62] Hilbert D. Mathematische Probleme. — Nachr. Akad. Wyss. Gottingen 1900, p. 253-297; Gesammelete Abhandlungen, Bd. 3 1935, p. 290-329.

[63] J. Hromkovich. Communication complexity and parallel computing, *Springer-Verlag, Berlin, Heidelberg, New York*, 1997.

[64] Ja'Ja' J., Prasana Kumar V. K., Simon J. Information transfer under different sets of protocols. — SIAM J. Comput. 1984, v. 13, p. 840-849.

[65] Karchmer M., Wigderson A. Monotone circuits for connectivity require superlogarithmic depth. — in Proc. of the 20th ACM STOC'88, 1988, p539-550.

[66] Karchmer M., Wigderson A. Characterizing non-deterministic circuits size. in Proc. of the ACM STOC'93, 1993, p. 532-540.

[67] E. Kushilevitz and N. Nisan. Communication Complexity, *Cambridge University Press*, 1997.

[68] T. Wah Lam, Ruzzo W. Results on Communication Complexity Classes. — Journal of Computer and System Sciences, 1992, v. 44, p. 324-342.

[69] Lengauer T. VLSI Theory. — in Handbook of Theoretical Computer Science, Edited by J. van Leeuwen, Elsevier Science Publishers B. V. 1990, p. 837-868.

[70] Lorentz G. Metric Entropy, Widths and Superpositions Functions. — Amer. Math. Monthly 1962, v. 69, 6, p. 469-485.

[71] Lovasz L. Communication Complexity: A Survey. — in "Paths, Flows and VLSI Layout", Korte, Lovasz, Promel, Schrijver Eds., Springer-Verlag 1990, p. 235-266.

[72] Lipton R., Sedgewick R. Lower bounds for VLSI. — in Proc. of the 13-th STOC, 1981, p. 300-307.

[73] Mehlhorn K., Schmidt E. Las Vegas is better than determinism in VLSI and distributed computing. — in Proc. of the 14th ASM STOC, 1982, p. 330-337.

[74] Handbook of Theoretical Computer Science, Edited by J. van Leeuwen, Elsevier Science Publishers B. V. 1990,

[75] Papadimitrio C. H., Sipser M. Communication Complexity. — in Proc. of the 14th ACM STOC 1983, p. 196-200.

[76] Paturi H., Simon J. Probabilistic Communication Complexity. — Journal of Computer and System Sciences, 1986, 33, p. 106-123.

[77] A. Paz A. Introduction to probabilistic automata. — N.Y., Wesly, 1971.

[78] Razborov A. The gap between the chromatic number of a gap and the rank of its adjacency matrix is superlinear. — Discr. Mathematics, 1992, 108, p. 393-396.

[79] Razborov A. On the distributional complexity of disjointness. — Theor. Comput. Sci., 1992, 106, p. 385-390.

[80] Hopcroft J., Ulman J. Introduction to automata theory, languages, and computation. — N.Y. Wesley, 1979.

[81] Rabin M. Probabilistic Algorithms for Testing Primality. — J. Number Theory, 1980, 12, p. 128-138.

[82] Stearns R.E., Hartmanis J., Lewis P.M. II. Hierarchies of memory limited computation — 1965 IEEE Conference Record on Switching Circuit Theory and Logical Design, 1965, p. 179-190.

[83] Thompson C. Area-time complexity for VLSI. — in Proc. of the 11th ACM STOC'79, 1979, p. 81-88.

[84] Yao A. Some Complexity Questions Related to Distributive Computing. — in Proc. of the 11th ACM Symposium on the Theory of Computing, 1979, p. 209-213.

[85] Yao A. Lower Bounds by Probabilistic Arguments. — in Proc. of the 24th IEEE Symposium on Foundations of Computer Science, 1983, p. 420-428.

[86] Соловьев Н.А. Тесты (теория, построение, применение). — Новосибирск, Наука, 1978.

[87] Фрейвалд Р.В, Икауниекс Э. О некоторых преимуществах вероятностных машин по сравнению с детерминированными. — Изв. вузов. Математика. 1977, 2, с. 118-123.