Д. Ю. Черухин

СРАВНЕНИЕ БУЛЕВЫХ БАЗИСОВ

(конспект лекций спецкурса)

МОСКВА • 2008

Черухин Д. Ю. Сравнение булевых базисов. — М.: График Без Границ, 2008. — 76 с.

Брошюра содержит обработанный конспект лекций спецкурса "Сравнение булевых базисов", который автор прочёл на мехмате МГУ в 2006/7 уч. г. Тематика спецкурса относится к дискретной математике, точнее, к теории сложности булевых функций. Спецкурс посвящён вопросам сравнения различных мер сложности формул исчисления высказываний, вычисляющих булевы функции. Исследуются несколько естественных мер сложности, так или иначе связанных с длиной формулы, при этом в качестве базиса рассматривается произвольная полная конечная система булевых функций. В спецкурсе излагаются результаты предшественников и автора, достаточно полно представляющие данную проблематику.

Автор выражает признательность своей жене Светлане, которая прослушала спецкурс и на основании своих конспектов и аудиозаписи лекций набрала и отредактировала эту брошюру.

Для студентов, специализирующихся в области дискретной математики.

Черухин Дмитрий Юрьевич
Сравнение булевых базисов

Набор, обработка: Черухина С. Е.
Вёрстка и издание автора

Неформальное некоммерческое издательство «График Без Границ».
Свёрстано с помощью системы TEX. Подписано в печать 07.02.2008.
Отпечатано на принтере. Объём 4,75 п. л. Тираж 2 экз.

Содержание

Глава I. Введение

Лекция 1 (4.10.06)

Начнем с определений. Что такое булева функция? *Булева функция* (*б. ф.*) *n переменных* — это следующее отображение $f\colon \{0,1\}^n \to \{0,1\}$. Такое определение годится для $n \geqslant 1$. Если $n = 0$, то f является константой (т. е. 0 или 1).

Приведём примеры б. ф. Основные функции — $\&, \vee, \neg$. Зададим их таблицей:

x	y	$\&$	$\vee$	$\overline{x}$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Табл. 1

Эти три функции образуют базис $B_0 = \{\&, \vee, \neg\}$, т. е. через них можно выразить любые функции. Например:

$x \to y = \overline{x} \vee y$ (*импликация*);

$x \leftrightarrow y = (x \to y) \,\&\, (y \to x)$ (*эквивалентность*);

$x \oplus y = \overline{x}y \vee x\overline{y}$ (*исключающее "или"* — истинно ровно одно).

Эти функции общеупотребительны. Существуют специальные функции. Например, *штрих Шеффера*: $x/y = \overline{x \,\&\, y}$. Эта функция замечательна тем, что через неё можно выразить все остальные функции. А также *стрелка Пирса*: $x \downarrow y = \overline{x \vee y}$. Это пример двойственности. Стрелка Пирса является двойственной функцией к штриху Шеффера.

Что такое *двойственность*? Это когда мы все 0 меняем на 1, и наоборот. Т. е., если в таблице для x/y мы сделаем все (!) замены $0 \to 1$ и $1 \to 0$, то получим (с точностью до перестановки строк) таблицу, задающую $x \downarrow y$.

Еще пример специальной функции (уже от трех переменных):

$m(x, y, z) = xy \vee xz \vee yz$ (*медиана*, или *функция голосования*). Она принимает значение 1, когда большинство из её переменных принимает значение 1.

Далее, что такое "функция принимает значение" и т. д. мы уточнять не будем, предполагая, что такие вещи известны, а понятию "формула", являющемуся в нашем курсе не только средством, но и объектом изучения, дадим подробное определение. Это определение не является самым общим (почему, будет ясно из дальнейшего).

Пусть задано множество функций $M = \{f_1, \ldots, f_k\}$. Имеются переменные, обозначаемые $x_1, x_2, \ldots$ или $x, y, z, \ldots$ Определим *формулы* над M:

1) x_i — формула, т. е. любая переменная является формулой (если $M = \{0, 1\}$, то x нельзя выразить через функции из M, поэтому наше определение не самое общее; у нас тождественная функция, по сути, всегда присутствует в M);

2) если $f_i \in M$, f_i имеет k_i аргументов и $F_1, \ldots, F_{k_i}$ — формулы, то $f(F_1, \ldots, F_{k_i})$ — формула;

3) других формул нет.

Данные выше примеры формул записаны не канонически. Например, мы подразумеваем эквивалентность:

$\overline{x} \vee y \equiv \vee(\neg(x), y)$.

Существуют следующие правила сокращения:

$\varphi(F, G) \equiv (F \varphi G)$;

$\neg(F) \equiv \overline{F}$;

$F \,\&\, G \equiv F \cdot G \equiv FG$.

В формуле без скобок самый высокий приоритет имеет &, на втором месте — $\vee, \oplus$, на третьем — $\to$ и на последнем — $\leftrightarrow$.

Многие понятия можно определить с помощью индукции по построению формулы. Например, каждая формула определяет (или реализует) некоторую б. ф.

Существуют некоторые тонкости: пусть есть формула $\overline{x} \vee y$. Переменные входят в формулу несимметрично. Что считать первой переменной, а что второй? Можно сказать, что эта формула реализует функцию с точностью до перестановки некоторых её переменных. Ещё момент: фиктивные переменные. Можно считать, что $\overline{x} \vee y = \varphi(x, y, z)$. Мы рассматриваем функции с точностью до порядка переменных и фиктивных переменных.

Определение. *M — базис, если любую б. ф. можно выразить некоторой формулой над M.*

Приведём примеры базисов.

1) $B_0 = \{\&, \vee, \neg\}$. Докажем, что это действительно базис.

Доказательство. Верно следующее разложение:

$$f(x_1, \ldots, x_n) = \overline{x}_1 f(0, x_2, \ldots, x_n) \vee x_1 f(1, x_2, \ldots, x_n).$$

Убедимся в верности формулы непосредственной подстановкой 0 и 1 вместо переменной x_1. Продолжая таким образом разложение по индукции, в конце концов сведём функцию к константам $f(c_1, \ldots, c_n)$. Осталось выразить константы в базисе B_0:

$$0 = x \cdot \overline{x}, \quad 1 = x \vee \overline{x}. \qquad \square$$

Базис B_0 считается основным, так как он является максимальным (по сложности) в иерархии базисов. Приведём другие примеры. Если добавить к базису ещё что-то, то это также будет являться базисом.

2) $P_2(n) = \{f : \{0,1\}^n \to \{0,1\}\}$ является базисом при $n \geqslant 2$. Почему? Поскольку $B_0 \subseteq P_2(2)$. Уточним, так как $\overline{x}$ — функция одного переменного, добавим к ней фиктивную переменную. Аналогично,

$$B_0 \subseteq P_2(2) \subseteq P_2(3) \subseteq P_2(4) \ldots$$

Мы построили бесконечную цепочку вложенных базисов. В связи с этим возникает естественный вопрос: возможно ли построить цепочку строго убывающих базисов в смысле сложности формул в них (точное определение будет дано позже), в частности, верно ли, что

$$B_0 \succ P_2(2) \succ P_2(3) \succ P_2(4) \ldots ?$$

Это исследование было начато Б. Субботовской. Вначале она показала [10], что $B_1 = B_0 \cup \{\oplus\}$ "лучше" B_0. Поясним, в базисе B_0

линейная функция имеет сложность по порядку больше, чем в базисе B_1. Таким образом, задача сравнения базисов неразрывно связана с проблемой так называемых "нижних оценок": показать, что некая функция в конкретном базисе имеет сложность, не меньшую некой величины, что является совсем непростой задачей. Например, в данном случае решение задачи (по тем или иным причинам) затянулось практически на 40 лет.

Бесконечная цепочка базисов следует из работы Н. Перязева [6](Иркутск, 1995). К сожалению, подробное доказательство им не было представлено, материалы опубликованы в виде препринта и тезисов. По существу, работа использует идеи Субботовской, которая смогла довести своей техникой длину цепочки до трёх. Дальнейшее использование её приемов оказывалось чрезмерно сложным, требовался какой-то другой подход. Как это оказывалось раньше, необходимо было ввести упрощающий доказательство функционал (т. е. заметить некоторые обобщающие свойства), что и удалось сделать Перязеву.

Заметим (для полноты картины), этой задачей занимались и другие. Так, Стеценко опубликовал в 1988 г. [8]свои результаты о сравнении различных базисов (неполных), им были построены различные цепочки, но окончательного решения ещё не было.

В 1996 г., не зная о результатах Перязева, это решение (неожиданно быстро) получил Д. Черухин [14]. Решение было основано на других, нежели Субботовской и Перязева, методах и даже давало критерий для сравнения базисов. В дальнейшем, независимо от Перязева, Черухину также удалось довести начатое Субботовской до конца, показав, что её методами также можно получить бесконечную цепочку.

Продолжим примеры базисов.

3) $B_1 = B_0 \cup \{\oplus\}$ — пример базиса, "существенно" отличного от B_0.

4) $B_{Ш} = \{/\}$ (штрих Шеффера). Проверим, что это — базис. Имеем

$$\overline{x} = x/x \text{ (вспомним, что } x/x = \overline{x \,\&\, x} = \overline{x}).$$

У нас теперь есть отрицание. Выразим, используя его, конъюнкцию и дизъюнкцию:

$$x \mathbin{\&} y = \overline{x/y} = (x/y)/(x/y).$$

Вспоминая законы де Моргана, получим

$$x \vee y = \overline{\overline{x} \mathbin{\&} \overline{y}}$$

(здесь иллюстрация того, что функции двойственны). Далее не выписываем до конца в силу громоздкости.

Мы показали, что $B_Ш$ — базис. Как будет ясно из дальнейшего, $B_Ш \sim B_0$ (базисы эквивалентны, поскольку их сложностные свойства одинаковы). Базис B_0 считается основным среди базисов, ему эквивалентных, поскольку он связан с контактными схемами, которые возникли исторически первыми.

Слово "сложность" уже несколько раз появлялось в нашей речи, а поскольку основная проблематика спецкурса как раз связана со сложностью в отличие, скажем, от полноты, то пора дать строгое определение этому понятию.

Определение. *Пусть $\mathcal{F}$ — множество формул над M, Тогда отображение $L : \mathcal{F} \to \mathbb{R}_+ = [0, \infty)$ называется мерой сложности.*

Термин "*функционал сложности*" используется в качестве искусственной меры, помогающей при доказательстве нижних оценок сложности.

Есть естественные меры сложности, например:

1) $L(F)$ — число вхождений в формулу F символов переменных. Правильнее определить сложность по индукции, но лучше приведем пример:

$$L(x\overline{y} \vee \overline{x}y) = 4.$$

Сложность функции определяется так: $L_M(f) = \min L(F)$, где $\min$ берется по всем формулам F над M, реализующим f.

2) Еще пример меры сложности: L^C — число вхождений в формулу функциональных символов (эта мера пришла из СФЭ).

3) $L^O(F)$ — общее число символов в формуле (L — от Length (англ.) — длина (формулы)).

Рассмотрим конкретную формулу:

$$F = \vee(\&(x, \neg(y)), \&(\neg(x), y)).$$

Для неё $L(F) = 4$, $L^C(F) = 5$, $L^O(F) = 4 + 5 + 5 \cdot 2 + 3 = 22$.

Как можно заметить,

$$L^O(F) \leqslant c \cdot L(F).$$

Над B_0 последнее неравенство верно для любой формулы, а если в базисе есть константы, то нет. Вообще, все выписанные меры (для одного и того же базиса B) эквивалентны: одна мера отличается от другой в константу раз. Это верно для функций, но не для формул. Почему? Приведем пример.

В качестве базиса рассмотрим $B_0 \cup \{0\}$. Тогда в этом базисе для формулы

$$F = 0 \,\&\, 0 \,\&\, 0 \ldots$$

имеем $L^C, L^O \to \infty$, а $L = 0$. У минимальных формул (в данном случае, $F = 0$) такого быть не может.

Для формул наиболее естественна мера L. Наиболее красивый результат в теории сложности — теорема Храпченко [13] — получается при использовании именно этой меры.

Теперь определим отношение порядка между разными мерами. У на есть множество $P_2 = \bigcup\limits_{n \geqslant 0} P_2(n)$ (все б. ф.). Пусть даны меры $L_1\colon P_2 \to \mathbb{R}_+$ и $L_2\colon P_2 \to \mathbb{R}_+$. Скажем, что L_1 *предшествует* мере L_2 ($L_1 \preceq L_2$), если

$$\exists C, D \ \ \forall f \in P_2 \ \ L_1(f) \leqslant C L_2(f) + D.$$

Зачем нужно D? Для случая, если $L_2(f) = 0$, а $L_1 \neq 0$.

Это отношение является отношением частичного порядка. Для проверки этого достаточно заметить, что $L_1 \preceq L_2 \,\&\, L_2 \preceq L_3 \Longrightarrow L_1 \preceq L_3$. Этот порядок не полный: меры могут быть несравнимы.

Дадим подробные определения производных отношений:

$L_1 \sim L_2 := L_1 \preceq L_2 \,\&\, L_2 \preceq L_1$;

$L_1 \prec L_2 := L_1 \preceq L_2 \,\&\, L_2 \not\preceq L_1$;

L_1 *не сравнима* с $L_2 := L_1 \npreceq L_2 \,\&\, L_2 \npreceq L_1$.

Таким образом, две меры могут находиться в 4-х отношениях:
1) $L_1 \sim L_2$;
2) $L_1 \prec L_1$;
3) $L_2 \prec L_1$;
4) L_1 и L_2 несравнимы.

Перенесем определение предшествования на базисы:

$$B_1 \preceq B_2 := L_{B_1} \preceq L_{B_2}.$$

При этом не важно, какую из мер (L, L^C или L^O) выбрать, как как позже мы докажем, что $\forall B\ L_B \sim L_B^C \sim L_B^O$.

Заметим, что имеется алгоритм, с помощью которого можно ответить на вопрос: "в каком отношении находятся 2 базиса?". Но есть и другая проблематика (не сложностная): описать всю иерархическую структуру базисов. Что известно на этот счет?

Мы уже упоминали Субботовскую, она рассмотрела базис B_0 ([10], 1961) и показала, что он "хуже" B_1, а позднее построила вложенную цепочку базисов длины 3, используя медиану ([4], 1970).

Стеценко ([9], 1992) описал первый ярус базисов, следующих непосредственно за B_0, с помощью 5-ти семейств; каждый из этих базисов представляет из себя B_0, в который добавлена некоторая функция. Второй ярус, видимо, также состоит из конечного числа семейств, образующихся при помощи двух функций 1-го яруса и еще каких-то функций. Построением этого яруса занимаются в Иркутске Перязев с учениками [2, 20]. По его словам, описание яруса есть, но работы по этому вопросу имеют разный уровень доказательности.

Относительно 1-го яруса можно сказать следующее: еще в 80-е гг. Стеценко было быстро (на PC) получено описание семейств, а строгое доказательство было изложено лишь в 1992 г. и заняло 40–50 стр. Как описать дальнейшие ярусы, на каком языке это сделать, является предметом так называемой текущей научной работы.

Приведём пример несравнимых базисов. Каждый базис из 1-го яруса эквивалентен базису вида $B_0 \cup \{f\}$, где f — S-функция. Таких функций имеется 5 типов (семейств). Вот один из них:

$$f_n = x_1 x_2 \ldots x_n \vee \overline{x}_1 \overline{x}_2 \ldots \overline{x}_n, \quad n \geqslant 2.$$

Базисы $B_0 \cup \{f_n\}$ и $B_0 \cup \{f_m\}, n \neq m$ — несравнимы.

Если вернуться к предыстории, в 1961 г. Субботовская привела пример неэквивалентных базисов, показав, что $B_0 \succ B_0 \cup \{\oplus\} = B_1$. Очевидно, что $L_{B_1}(x_1 \oplus \ldots \oplus x_n) = n$. Она показала, что $L_{B_0}(x_1 \oplus \ldots \oplus x_n) \succeq n^{3/2}$. В 1963 г. она же дала критерий того, что заданный базис эквивалентен B_0, т. е. описание всех B таких, что $B \sim B_0$ [11]. Например, $\{/\} \sim B_0$, а $B_1 \nsim B_0$.

Также в 1970 г. ей были описаны некоторые базисы, неэквивалентные B_1. А именно, рассматривалась сложность линейной функции в базисах вида $B = B_0 \cup \{$некоторая монотонная функция$\}$ и для неё получалась нелинейная нижняя оценка, т. е. $L_B(x_1 \oplus \ldots x_n) \succeq n^{1+c_B}, c_B > 0$. Таким образом, получалась цепочка из трех убывающих базисов $B_0 \succ B_1 \succ B \cup \{\oplus\}$.

Лекция 2 (11.10.06)

На прошлой лекции были даны основные определения. Сейчас будут выведены простейшие следствия из них. Их доказательства в некоторых работах совсем опускали, считая очевидными или просто выводимыми. На самом деле, строгое док-во общего случая (как мы увидим далее) потребует значительного кол-ва времени и сил.

Мы докажем эквивалентность трех мер сложности функций: L_B — число вхождений переменных, L_B^C — число вхождений функций, L_B^O — общая длина формулы. Вначале выпишем очевидные неравенства:

$$L_B \leqslant L_B^O, \quad L_B^C \leqslant L_B^O.$$

Далее докажем, что L_B^O не намного превышает L_B^C.

Вспомним индуктивное определение формулы. Базис: переменные x_i и константы c; индуктивный переход: $f(F_1, \ldots, F_k)$. Функциональному символу f припишем некоторые символы формулы $F = f(F_1, \ldots, F_k)$: саму функцию f, а также те скобки и запятые формулы F, которые не содержатся в подформулах $F_1, \ldots, F_k$. Дополнительно, если подформула F_i является переменной, то припишем её функции f. Константе c припишем её саму.

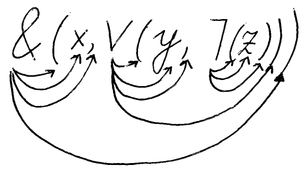

Рис. 1

На рис. 1 стрелка, ведущая от функции к другому символу, означает, что этот символ приписан функции. Итак, все символы мы приписали некоторым функциям. Какова максимальная мощность множества символов, приписанных одной функции? Введём следуюшее понятие.

Определение. *Порядок базиса — наибольшее число переменных у функций из этого базиса.*

Обозначим порядок базиса через $n(B)$ (заметим, что всегда $n(B) \geqslant 2$). Тогда одной функции может быть приписано не больше, чем:

$$2\,\text{скобки} + (n(B)-1)\,\text{запятых} + n(B)\,\text{переменных} + \text{сама функция}.$$

Следовательно, общую длину формулы можно оценить выражением:

$$L^O(F) \leqslant (2n(B)+2)L^C(F).$$

Отсюда $L_B^C \sim L_B^O$.

Теперь будем сравнивать меры L_B и L_B^C. В одну сторону:

$$L(F) \leqslant L^O(F) \leqslant (2n(B)+2)L^C(F).$$

Следовательно, $L_B \preceq L_B^C$.

В другую сторону док-во сложнее. Так, в упомянутой на лекции 1 формуле $0\&0\&\ldots$, а также в формуле $f(f(f\ldots f(x)\ldots))$, функциональных символов может быть сколь угодно много, а переменных не более одной. С такими неприятностями мы будем бороться. Мы видим, что неравенство $L_B^C \preceq L_B$ неверно для мер сложности формул. Поэтому мы будем его доказывать для мер сложности функций (!). Наша цель — показать, что $L_B^C(f) \leqslant c_1 L_B(f) + c_2$, где константы c_1 и c_2 зависят только от базиса B.

Приступим к доказательству. Рассмотрим для данной функции f минимальную формулу F_0 в смысле стандартной меры (L_B), т. е. $L(F_0) = L_B(f)$. В базисе B могут быть не все константы, поэтому мы расширим базис. Определим замыкание базиса.

Сначала дадим ещё одно определение. Две функции от одного числа переменных $f(x_1, \ldots, x_n)$ и $g(x_1, \ldots, x_n)$ называются *обобщённо-однотипными* (*о.-о.*), если одну можно получить из другой следующими операциями:

1) перестановка переменных;
2) "навешивание" отрицания на переменную ($x_i \mapsto \overline{x}_i$);
3) "навешивание" отрицания на функцию ($f \mapsto \overline{f}$).

Это понятие и в дальнейшем будет часто использоваться. Поясним геометрический смысл о.-о. Просто *однотипность* (соответствующая св-вам 1 и 2) геометрически эквивалентна тому, что на булевом кубе вращением (движением) куба можно перевести одну функцию в другую. В о.-о. добавляется возможность изменять все значения функции на противоположные ($0 \leftrightarrow 1$).

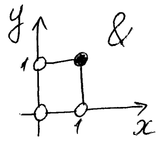

Рис. 2

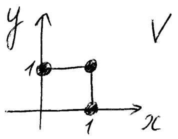

Рис. 3

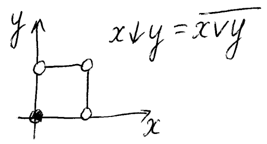

Рис. 4

На рис. 2–4 незакрашенные кружки соответствуют значению 0, а закрашенные — 1. Функции & и $\downarrow$ однотипны (поворот на 180°), а функции & и $\vee$ о.-о. (поворот и замена значений), но не однотипны.

Наконец, определим $[B]$ — *замыкание базиса* B. Множество $[B]$ получается из B с помощью следующих двух операций:

1) для любой f берём все ее подфункции, т. е. $f(x_1, \ldots, x_n) \mapsto f(x_1, \ldots, c_i, \ldots, x_n)$;

2) для любой полученной f берём все о.-о. с ней.

Приведём примеры.

1) $B_0 = \{\&, \vee, \neg\}$, $[B_0] = \{(x^\sigma y^\tau)^\pi, c, x^{c'}\}$ $\left(x^\sigma = \begin{cases} \overline{x}, & \sigma = 0; \\ x, & \sigma = 1. \end{cases}\right)$ Всего в $[B_0]$ 12 функций: 8 нелинейных и еще 4.

2) $B' = \{/\}$ (штрих Шеффера). Так как $x/y = \overline{xy}$, то при $y = 1$ получим $\overline{x}$. В итоге $[B'] = [B_0]$. Это не случайный факт (как выяснится позже). Алгоритмический критерий сравнения базисов [16]звучит так: базисы эквивалентны тогда и только тогда, когда их ядра (Ker) совпадают (в данном случае Ker почти совпадает с замыканием).

Продолжим док-во. У нас есть минимальная формула с точки зрения стандартной меры. Преобразуем её, перейдя к замыканию базиса $[B]$. Полезно рассмотреть такое понятие как дерево формулы. Каждой формульной переменной и функциональному символу соответствует вершина дерева (см. рис. 5).

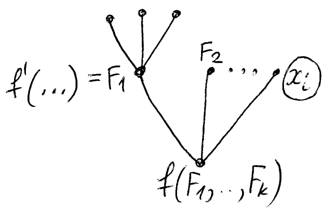

Рис. 5

Дерево отражает процесс построения формулы. Приведем пример для иллюстрации: формуле $\&(x, \vee(y, \neg(z))$ соответствует дерево на рис. 6.

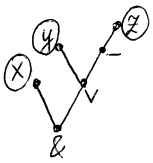

Рис. 6

В чем состоит наша цель? Мы хотим исключить нежелательные цепочки из функций одного аргумента $f(g(h(\ldots)))$ (см. рис. 7) и константы.

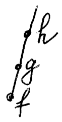

Рис. 7

С этой целью будем проводить с формулой F_0 следующие преобразования, результат которых обозначим через F_1.

1) Если где-то встретилась константа, мы удалим её, заменив функцию, где аргументом является данная константа, её подфункцией из $[B]$, например:

$$f(0, x, g(1, y)) \mapsto f|_{x_1=0}(x, g|_{x_1=1}(y)).$$

2) Теперь удалим функции одного аргумента (не всегда это можно сделать, но остается конечное (!) число случаев, когда это невозможно). Пусть $g \in \{0, 1, x, \overline{x}\}$; если $g(x) \equiv \text{const}$, то случай сводится к п. 1; если $g(x) \equiv x$, то просто удалим функцию g. Остается случай $g(x) \equiv \overline{x}$. Здесь возможны две ситуации.

а) Отрицание применяется к переменной (рис. 8). Тогда отрицание спускаем ниже; оно поглощается предыдущей функцией f, которая превращается в функцию $f'(\ldots, x) = f(\ldots, \overline{x})$:

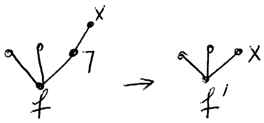

Рис. 8

б) Отрицание применяется к подформуле, но не к переменной (рис. 9). Тогда отрицание "навешиваем" на всю функцию:

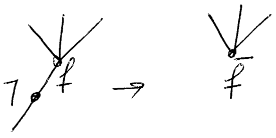

Рис. 9

Когда ни одно из преобразований не применимо, а цель не достигнута? В случаях, если $F_0 = \mathrm{const}$ или $F_0 = \neg(x_i)$. Следовательно, есть три возможных исхода преобразований:

а) c;

б) $\overline{x}_i$;

в) формула F_1, не содержащая констант и функций одного аргумента.

Теперь для F_1 можно доказать нужную оценку о сравнении мер сложности L и L^C. Рассмотрим дерево формулы F_1. Мы будем использовать факты из теории графов (ТГ). В ТГ есть понятие *степени* вершины: $d(v)$ — число ребер, инцидентных вершине v.

Рассмотрим случай в). Тогда в дереве есть *листовые* вершины (т. е. вершины, для которых $d(v) = 1$), в листьях находятся переменные и для любой вершины v, за исключением корня, $d(v) \neq 2$ (см. рис. 10).

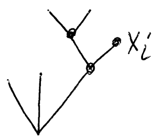

Рис. 10

А для такого типа деревьев есть "хорошие" оценки. Воспользуемся некоторыми формулами из ТГ:

1) $\sum\limits_v d(v) = 2E$ (формула Эйлера, E — число ребер);

2) в дереве $V = E + 1$ (V — число вершин).

Нам надо доказать, что $L^C(F_1) \leqslant c_1 L(F_1) + c_2$. Введем обозначения: L — число листовых вершин (сложность формулы F_1), L^c — число нелистовых вершин ($L^C(F_1)$). Тогда $L + L^c = V$ — общее число вершин. Применим формулы ТГ и преобразуем, учитывая, что степень корня может равняться двум:

$$2L + 2L^c - 2 = 2(V-1) = 2E = \sum_v d(v) \geqslant 1 \cdot L + 2 + 3(L^c - 1),$$

$$2L + 2L^c - 2 \geqslant L + 2 + 3L^c - 3,$$

$$L^c \leqslant L - 1.$$

Заметим, что если базис состоит из двуместных функций, то верно просто равенство.

Возвратимся к базису B. А именно, преобразуем формулу F_1 в формулу F_2 в базисе B, не намного увеличив её сложность L^C (заметим, что при переходе от F_0 к F_1 переменные мы не удаляли, поэтому сложность L не менялась). Рассмотрим произвольную нелистовую вершину дерева формулы F_1; пусть ей соответствует функция $f \in [B]$. Тогда

$$f(x_1, \dots, x_k) = g^\sigma(x_1^{\sigma_1} \dots, x_k^{\sigma_k}, c_{k+1}, \dots, c_n), \quad g \in B. \qquad (1)$$

Преобразуем окрестность данной вершины в соответствии с равенством (1): заменим функцию в вершине на g; добавим новые листовые вершины, соответствующие константам $c_{k+1}, \dots, c_n$, и рёбра, ведущие к ним; на всех рёбрах, соответствующих переменным с отрицаниями, введём по новой вершине и пометим их функциями отрицания; наконец, если на функцию g "навешено" отрицание, то введём новую вершину ниже данной, пометив её отрицанием. Весь процесс преобразования, включая и последующие шаги, изображён на рис. 11.

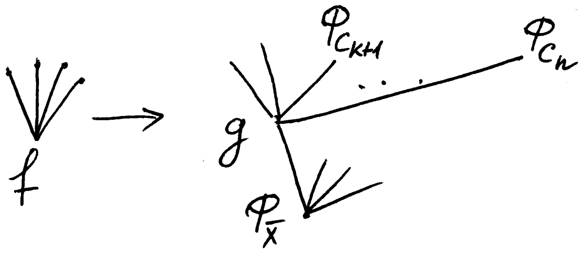

Рис. 11

Проведя указанное преобразование для каждой нелистовой вершины, мы получим формулу над базисом $B \cup \{0, 1, \neg\}$. Однако, константы и отрицание не обязательно содержатся в B, поэтому продолжим преобразования. В силу полноты базиса, константы выражаются в нём. Поэтому есть формулы Φ_0, Φ_1 в B такие, что $\Phi_0 \equiv 0$, $\Phi_1 \equiv 1$. Заменим всюду в дереве константы c на соответствующие поддеревья Φ_c.

Далее, отрицание тоже выражается некоторой формулой в B, но нам подойдёт только такая формула $\Phi_{\overline{x}}$, $\Phi_{\overline{x}} \equiv \overline{x}$, в которую переменная x входит ровно один раз (иначе придётся дублировать поддеревья, соответствующие аргументу отрицания, что может привести к существенному росту сложности).

Построим формулу $\Phi_{\overline{x}}$. Для этого вспомним теорему Поста: в любом (полном) базисе есть немонотонная функция, т. е. существует такая $\varphi \in B$, что

$$\exists c_1, \ldots, c_n\colon \quad \varphi(c_1, \ldots, 0, \ldots, c_n) = 1,\ \varphi(c_1, \ldots, 1, \ldots, c_n) = 0.$$

Тогда

$$\varphi(c_1, \ldots, x, \ldots, c_n) = \overline{x}.$$

Наконец, положим

$$\Phi_{\overline{x}} = \varphi(\Phi_{c_1}, \ldots, x, \ldots, \Phi_{c_n})$$

(если в формулы Φ_{c_i} входит переменная x, то мы её в этих формулах заменим другой ($x \mapsto y$) для того, чтобы x входила только один раз).

Все функции отрицания в нашем дереве заменим на поддеревья $\Phi_{\overline{x}}$, а именно, каждое отрицание представляет собой вершину степени 2 с одним входящим и одним исходящим ребром. Изменим функцию в этой вершине на φ, добавим новые исходящие рёбра и подвесим к ним поддеревья Φ_{c_i}, соответствующие константам.

Подведём итог нашим преобразованиям $F_1 \mapsto F_2$. В какое максимальное число функциональных символов (нелистовых вершин) в формуле F_2 может раскрыться одна функция f из формулы F_1? Обозначим $c' = \max\{L_B^C(\Phi_0), L_B^C(\Phi_1), L_B^C(\Phi_{\overline{x}})\}$. Когда мы вместо констант $c_{k+1}, \ldots, c_n$ подставим реализующие их Φ_{c_i} (см. (1)), сложность L^C увеличится на не более чем на $(n-k)c'$. При этом может возникнуть максимум $(k+1)$ отрицание, а каждое отрицание раскрывается в не более, чем c' функций.

Итого, каждая функция из F_1 может раскрыться не более, чем в

$$1 + (n-k)c' + (k+1)c' \leqslant (n+2)c' \leqslant (n(B)+2)c' = c''$$

функций в формуле F_2. Заметим, что мы рассмотрели случай в). В случаях а) и б) коэффициент увеличения $\leqslant c' \leqslant c''$. Поэтому

$$L_B^C(f) \leqslant L^C(F_2) \leqslant c'' L^C(F_1) \leqslant$$

$$\leqslant c''(L(F_1)+1) = c''(L(F_0)+1) = c''(L_B(f)+1).$$

Следовательно, $L_B^C \preceq L_B$. Мы доказали лемму:

Лемма 1. $\forall B\ L_B^O \sim L_B^C \sim L_B$.

Лекция 3 (18.10.06)

Мы продолжим доказательство несложных (т. е. не использующих нижние оценки сложности) утверждений о сравнении базисов. Общее заглавие сегодняшней лекции — бесповторность.

Что такое бесповторная формула? Пусть функция $f(x_1, \ldots, x_n)$ задана формулой F, т. е. $F \equiv f(x_1, \ldots, x_n)$. Какие-то переменные в формуле могут быть несущественными, т. е. F реализует f с точностью до удаления несущественных переменных.

Определение. *Формула F бесповторна, если каждая из её существенных переменных входит в F ровно 1 раз.*

Приведём примеры бесповторных и не являющихся таковыми формул. Бесповторные формулы:

1) $x \mathbin{\&} \overline{x}$ $[= 0]$; x — несущественная переменная, количество несущественных переменных может быть любым, поэтому данная формула является бесповторной;

2) $y \vee x \mathbin{\&} \overline{x}$ $[= y]$ (аналогично).

Не бесповторные формулы:

1) $x \vee x \mathbin{\&} \overline{x}$ $[= x]$; существенная переменная x входит 3 раза. Как превратить эту формулу в бесповторную? Заменим некоторые x на y $(x \mapsto y)$: $x \vee y \mathbin{\&} \overline{y}$. Если бы в базисе были константы, можно было бы заменить все несущественные переменные ими.

Определение. *Функция f бесповторно выразима* ($\equiv$ *реализуется*) *в базисе B, если найдётся бесповторная формула в базисе B, которая реализует f.*

Примеры.

1) $B_{Ш} = \{/\}$. Реализуется ли бесповторно $\overline{x}$ в этом базисе? Выразим отрицание: $\overline{x} = x/x$. Эта формула не является бесповторной. Не очевидно, можно ли предъявить бесповторную формулу. Надо использовать несущественные переменные.

Константы можно выразить бесповторно, поскольку базис полный и входящие в формулу для константы переменные не являются существенными. Так как $x \mathbin{\&} y = \overline{x/y}$, то

$$1 = x/\overline{x} = x/(x/x).$$

Отрицание можно выразить следующим образом $\overline{x} = x/1$. Тогда, учитывая представление 1 через / и заменяя несущественную переменную x на z, получаем бесповторную формулу для $\overline{x}$ в базисе $B_{Ш}$:

$$\overline{x} = x/(z/(z/z)).$$

2) Функция $x \oplus y$ бесповторно не выражается ни в $B_Ш$, ни в B_0. Почему? Предположим противное. Пусть она выражается бесповторно в B_0. Подставим в бесповторную формулу вместо несущественных переменных константы и воспользуемся правилами сокращения:

$$x \,\&\, 1 = x, \quad x \,\&\, 0 = 0, \text{ и т. д.}$$

В итоге, мы получим формулу, содержащую только две существенные переменные и имеющую вид $(x^\sigma \circ y^\pi)^\sigma$, где $\circ$ есть & или $\vee$. Эта формула реализует нелинейную функцию. Противоречие. Следовательно, линейная функция бесповторно невыразима в базисе B_0. Это причина того, что B_1 не эквивалентен B_0 (в дальнейшем об этом будет сформулирована теорема).

Чем хороша бесповторность? Сравним два базиса: B_0 и $B_Ш$. Мы хотим доказать, что $B_Ш \preceq B_0$. Раскрывая определение $\preceq$, необходимо доказать следующее:

$$\forall f \; L_{B_{\text{Ш}}}(f) \leqslant C L_{B_0}(f) + D.$$

Каков естественный путь доказательства? По формуле F, реализующей f в базисе B_0, мы должны построить формулу G в базисе $B_Ш$ так, чтобы

$$L(G) \leqslant CL(F) + D.$$

Если мы воспользуемся следующей (не бесповторной) формулой, выражающей конъюнкцию через функцию Шеффера:

$$F_1 \,\&\, F_2 = (F_1/F_2)/(F_1/F_2),$$

то сложность формулы может возрасти более, чем в константу раз, так как на каждом ярусе мы должны удваивать подформулы F_1 и F_2, а число ярусов может быть неограничено.

Бесповторная формула такого эффекта не производит. Сформулируем лемму, дающую основной признак сравнения базисов. Это, наверное, самая первая лемма, сформулированная в работе Субботовской [11]по этой проблематике и доказанная, по ее словам, Лупановым.

Лемма 2. *Пусть даны базисы B_1 и B_2. Если $\forall f \in B_2$ f бесповторно выразима в B_1, то $B_1 \preceq B_2$.*

Доказательство. Как мы уже говорили, будем строить по формуле F над базисом B_2, реализующей функцию f, формулу G над базисом B_1. Для удобства доказательство проведем в мере сложности L^C, т. е. покажем, что

$$L^C(G) \leqslant c' L^C(F)$$

и воспользуемся леммой 1.

Пусть $F = f(F_1, \ldots, F_k)$, где $f \in B_2$. По условию, f бесповторно выразима в B_1, т. е. есть бесповторная в B_1 формула H такая, что $f(x_1 \ldots x_k) \equiv H(x_1, \ldots, x_k, z)$, где z — несущественная переменная. Тогда мы заменим F формулой $H(F_1, \ldots, F_k, z)$; z можно выбрать так, чтобы она не совпадала с другими переменными. Применим эту операцию рекурсивно к $F_1, \ldots, F_k$.

Как при этом изменится сложность L^C? Один функциональный символ заменится на константное число других функциональных символов. В итоге, так как базис состоит из конечного числа функций, получим $L^C_{B_1} \preceq L^C_{B_2}$. Наконец, воспользуемся леммой 1:

$$L_{B_1} \sim L^C_{B_1} \preceq L^C_{B_2} \sim L_{B_2}. \qquad \square$$

Извлечём следствия из полученного результата. Сначала немного забежим вперёд: более сильного признака нет. В конечном счете, справедлива

Основная теорема. $B_1 \preceq B_2 \leftrightarrow \forall f \in B_2$ *f бесповторно выразима в B_1.*

Её доказательство занимает несколько десятков страниц [16]. Доказана она была постепенно с 1996 по 1999 гг. [14, 15]

Продемонстрируем применение леммы 2. Сравним два базиса: B_0 и $B_Ш$. Поскольку $x/y = \overline{x \,\&\, y}$, то функция / бесповторно выразима в B_0, а значит, $B_0 \preceq B_Ш$.

Теперь, наоборот, выразим бесповторно через функцию Шеффера элементы базиса B_0, т. е. $\&, \vee, \neg$. Бесповторную формулу для отрицания мы уже строили: $\overline{x} = x/1 = x/(z/(z/z))$. Конъюнкция выражается следующим образом: $x \,\&\, y = \overline{x/y}$.

Поскольку "штрих" и отрицание представляются бесповторными формулами, то и композиция таких формул есть бесповторная формула. Это утверждение можно сформулировать более строго: бесповторная композиция бесповторных формул бесповторна. Поэтому и дизъюнкцию можно представить бесповторной формулой в силу $x \vee y = \overline{\overline{x} \,\&\, \overline{y}}$. Следовательно, $B_{\text{Ш}} \preceq B_0$ и $B_{\text{Ш}} \sim B_0$.

Так же можно показать, что $B_1 = B_0 \cup \{\oplus\} \sim P_2(2)$. В одну сторону доказательство очевидно, поскольку B_1 вложен в $P_2(2)$. В другую — надо показать, что любую двуместную функцию можно выразить через B_1. Какие бывают двуместные функции? Линейные и нелинейные. Нелинейные функции можно бесповторно выразить в B_0. А для линейных понадобится также $\oplus$. Для одноместных функций всё ещё очевиднее.

Вообще, если ограничиться базисами B такими, что $B \subseteq P_2(2)$, то имеется всего два класса эквивалентности: базисы, эквивалентные B_0, и эквивалентные B_1. Точнее, мы пока не можем ничего сказать, как соотносятся между собой базисы B_0 и B_1. Лемма 2 здесь не помогает, и нужны более сложные вещи, связанные с нижними оценками сложности.

Если ограничиться $P_2(3)$, то уже будет 40 классов эквивалентности. Вообще, число этих классов растёт очень быстро — как "тройная" экспонента $(2^{2^{2^n}})$. И как описывать эти классы, неясно. Начальные ярусы еще можно как-то описать, а дальше?..

Продолжим выводить следствия из леммы 2. Почему базис B_0 максимален?

Лемма 3. $\forall B \; B \preceq B_0$.

Доказательство. Поскольку $x \vee y = \overline{\overline{x} \,\&\, \overline{y}}$, нам достаточно показать, что можно бесповторно выразить в любом базисе конъюнкцию и отрицание. Почему отрицание бесповторно выразимо в B? Сначала выразим константы: они бесповторно выразимы по определению. Пусть они выражаются в базисе B формулами Φ_0 и Φ_1.

Далее, поскольку базис полный, в нём есть немонотонная функция f. Это означает, что

$$\exists i, c_1, \ldots, c_n\ f(c_1, \ldots, 0, \ldots, c_n) > f(c_1, \ldots, 1, \ldots, c_n).$$

Тогда

$$\overline{x} = f(c_1, \ldots, \overset{i}{x}, \ldots, c_n) = f(\Phi_{c_1}, \ldots, x, \ldots, \Phi_{c_n}).$$

Как бесповторно выразить конъюнкцию? Можно сослаться на теорему Поста из учебника Яблонского [23]. Мы пользуемся тем фактом, что базис полный и тогда в нём есть нелинейная функция и тем, что любую функцию можно представить в виде полинома Жегалкина.

Представим нелинейную функцию $f \in B$ в таком виде:

$$f(x_1, \ldots, x_n) = c_0 \oplus c_1 x_1 \oplus \ldots \oplus c_n x_n \oplus x_1 x_2 \ldots x_k \oplus \ldots,$$

т. е. у неё есть линейная часть, минимальный по длине нелинейный моном (здесь он состоит для удобства из первых k, $k \geqslant 2$, переменных) и другие мономы.

Подставим вместо последних $(n-k)$ переменных нули, т. е. положим $x_{k+1} = \ldots = x_n = 0$. Тогда останется лишь линейная часть и один моном длины k. Затем положим $x_3 = \ldots = x_k = 1$. Получим $x_1 x_2 \oplus c_1 x_1 \oplus c_2 x_2 \oplus c_0$. А это есть функция $(x_1^\sigma \,\&\, x_2^\pi)^\tau$. Из неё можно получить, поскольку у нас уже есть отрицание, &, а затем и $\vee$. $\square$

Ещё несложные факты. Мы ввели понятие замыкания базиса. Следующее утверждение часто используется при доказательстве нижних оценок.

Утверждение. $B \sim [B]$.

Доказательство. В одну сторону очевидно, так как $B \subseteq [B]$ и $L_{[B]} \leqslant L_B$. Для доказательства в другую сторону воспользуемся леммой 2. Как выглядит произвольная функция из замыкания B?

$$\forall f \in [B]\ f \equiv \varphi^\sigma(x_1^{\sigma_1}, \ldots, x_k^{\sigma_k}, c_{k+1}, \ldots, c_n), \text{ где } \varphi \in B.$$

Поскольку $\varphi \in B$, константы и отрицание бесповторно выразимы в любом полном базисе, то f бесповторно выразима в B, следовательно, $L_B \preceq L_{[B]}$. $\square$

Далее, пусть у нас есть два базиса. Как их сравнить? Как узнать, выразим ли бесповторно один базис через другой? Есть ли такой алгоритм? Поскольку с ходу непонятно, как нам действовать, в силу того, что несущественных переменных может быть слишком много. Оказывается, можно дать алгоритмический критерий, используя понятие ядра базиса. Давайте введём это понятие. Оно связано с понятием простой функции.

Сначала определим, что такое разложимая функция. Функция f называется *разложимой*, если

$$\exists \varphi, \psi \ f(x_1, \ldots, x_n) \equiv \psi(\varphi(x_{i_1}, \ldots, x_{i_k}), x_{i_{k+1}}, \ldots, x_{i_n}),$$

где $k \geqslant 2$, $k < n$ и $(i_1, \ldots, i_n)$ — перестановка набора $(1, \ldots, n)$.

Пример разложимой функции: $x \vee yz = \vee(\&(y, z), x)$. Пример неразложимой функции: $m(x, y, z) = xy \vee xz \vee yz$ (пока мы не можем это доказать). Впервые неразложимые функции упоминаются, возможно, в работе Кузнецова [3].

Теперь, скажем, что f — *простая*, если:
1) f существенно зависит от всех переменных;
2) f неразложима.

Определение. *Ядром базиса B* (Ker(B)) *называется множество всех простых функций из замыкания базиса:*

$$Ker(B) = \{f \in [B] \colon f \text{ — \textit{простая}}\}.$$

Следствие из основной теоремы.

$$B_1 \preceq B_2 \leftrightarrow Ker(B_2) \subseteq Ker(B_1).$$

Понятно, что ядро строится эффективно. Имеем лаконичный критерий, к тому же алгоритмический. Сравнение сложностных характеристик базисов сводится к теоретико-множественному вложению.

Глава II. Сравнение обобщенно-монотонных базисов

Лекция 4 (8.11.06)

Эта лекция начинает новый цикл, посвящённый нижним оценкам сложности. Предыдущие три можно рассматривать как обширное введение, в котором, по существу, сложных, серьёзных доказательств не было.

Сегодня будет первое доказательство, правда, не очень сложное, поскольку это самая первая нелинейная нижняя оценка, которая вообще была получена за всю историю и, разумеется, она доказывается не очень сложно. Это известная теорема Субботовской [10]о том, что сложность линейной функции в классе π-схем, т. е. параллельно-последовательных контактных схем, или, что то же самое, в классе формул в базисе B_0, нелинейна, а именно, растёт по порядку не меньше, чем $n^{3/2}$.

Вначале изложим доказательство, а затем будем обсуждать вещи исторического плана, потому что проблема нахождения сложности линейной функции в классе π-схем — это самостоятельная задача, интересная и с точки зрения истории математики. На языке сравнения базисов мы установим существование неэквивалентных базисов, а именно, покажем, что $B_1 = B_0 \cup \{\oplus\} \prec B_0 = \{\&, \vee, \neg\}$ (ранее мы доказали, что $B_1 \preceq B_0$). Это мы сделаем следующим образом: рассмотрим линейную функцию $x_1 \oplus \ldots \oplus x_n$ и докажем, что в базисе B_0 она имеет нелинейную сложность.

Каким методом мы будем это доказывать? Имеются некоторые вариации. Есть первоначальное доказательство Субботовской: целенаправленно выбирается переменная, выбирается константа и делается подстановка. Есть современное толкование в вероятностном духе: рассматривается случайная подстановка, а именно, все переменные, независимо друг от друга, с одной и той же вероятностью заменяются константами 0 или 1 или с некоторой другой вероятностью остаются переменными. Сейчас на Западе рассуждают исключительно в таком духе [25, 24]. Мы выберем нечто среднее: будем каждый раз выбирать одну переменную и подставлять вместо неё константу, но переменную и константу будем выбирать случайно.

В чём состоит идея доказательства? Эта общая идея, получившая название "забивание переменных", состоит в следующем. Предположим, что сложность данной функции в базисе B_0 равна $c \cdot n$, т. е. каждая переменная в среднем входит в формулу c раз. Тогда выберем случайную переменную и подставим вместо неё случайную константу. При этом все вхождения этой переменной мы потеряем, т. е. сложность уменьшится на c. А за счет того, что наряду с этой переменной мы потеряем вхождения ещё и других переменных, сложность уменьшится более, чем на c. В результате сложность будет иметь нелинейный рост.

Почему мы потеряем ещё какие-то переменные? Потому что, например, подставив константу 0 вместо x_i в формулу $x_i x_j$, мы избавимся не только от x_i, но и от x_j. А когда мы делаем случайную подстановку, то с какой-то вероятностью константа 0 будет подставлена вместо x_i. Значит, мы потеряем какую-то долю других переменных. Вот, на макроуровне, почему сложность будет нелинейной. Теперь давайте проведём это рассуждение подробнее.

Пусть есть некоторая формула F в базисе B_0 и пусть выполнены условия:

1) $L(F) \geqslant 2$;

2) F минимальна (никакие упрощения в F невозможны, она уже максимально упрощена);

3) в F входят только переменные $x_1, \ldots, x_n$ (нужно для проведения индукции);

4) $\forall c\ F \not\equiv c$.

Теперь определим случайную подстановку. Обозначим её A. С вероятностью $\frac{1}{2n}$ вместо переменной x_i подставляется константа σ:

$$P(x_i = \sigma) = \frac{1}{2n}, \quad 1 \leqslant i \leqslant n,\ 0 \leqslant \sigma \leqslant 1,$$

т. е. всегда выбирается только одна переменная, одна константа, но они выбираются случайно. Также мы будем использовать событие $x_i = *$. Оно означает, что вместо x_i ничего не подставляется, т. е. константа подставляется вместо какой-то другой переменной.

Соответственно, можно рассмотреть формулу, которая получится в результате этой подстановки, а именно, $F' = F|_A$ (в формулу F производится подстановка A и затем осуществляются всевозможные упрощения). В формуле F' будем допускать константы, т. е. рассматривать её над базисом $B_0 \cup \{0,1\}$. Заметим, что если F' удовлетворяет условию 4), то от констант можно избавиться. Обозначим $L_F = L(F)$ и $L'_F = L(F')$. Последнее выражение есть случайная величина; для неё мы будем рассматривать матожидание $M(L'_F)$.

Лемма 4. *Пусть F удовлетворяет условиям* 1)–4). *Тогда*

$$M(L'_F) \leqslant L_F\Big(1 - \frac{3}{2n}\Big).$$

Доказательство. Доказывать будем индукцией по построению формулы F. В качестве базиса индукции мы не можем взять переменную потому, что сложность переменной равна 1. Поэтому будем рассматривать формулы вида $x_i^\sigma \circ x_j^\pi, \circ \in \{\&, \vee\}$. В силу симметрии, достаточно рассмотреть формулу $F = x_i x_j$.

Выпишем вероятности и подсчитаем соответствующие сложности. С вероятностью $\frac{1}{2n}$ вместо x_i подставляется 0. Тогда $F' = 0$, $L'_F = 0$. С вероятностью $\frac{1}{2n}$ вместо x_i подставляется 1. Получаем $F' = x_j$, $L'_F = 1$. Для x_j всё аналогично. Далее, с вероятностью $1 - \frac{2}{n}$ наступает событие $x_i = *$, $x_j = *$, при котором $F' = x_i x_j$, $L'_F = 2$.

Теперь всё это нужно просуммировать и найти матожидание полученной сложности.

$$M(L'_{x_i x_j}) = 2\Big(1 - \frac{2}{n}\Big) + \frac{2}{2n} = 2 - \frac{3}{n} = 2\Big(1 - \frac{3}{2n}\Big) = L_{x_i x_j}\Big(1 - \frac{3}{2n}\Big).$$

Индуктивный переход. Здесь есть несколько случаев. Начнём с тривиальных.

1) Допустим, что $F = \overline{G}$. К формуле G мы можем применить предположение индукции. Очевидно, что сложности G и F равны, они также будут равны после подстановки констант и упрощений, поэтому в данном случае требуемое неравенство дословно переносится с G на F.

2) $F = G \circ H$, $\circ \in \{\&, \vee\}$, $L(G) \geqslant 2$, $L(H) \geqslant 2$. Достаточно одной выкладки:

$$M(L'_F) \leqslant M(L'_G)+M(L'_H) \leqslant L_G\Big(1-\frac{3}{2n}\Big)+L_H\Big(1-\frac{3}{2n}\Big) = L_F\Big(1-\frac{3}{2n}\Big).$$

3) Остался случай, когда G является переменной или отрицанием переменной. Здесь существуют 4 возможности; мы рассмотрим один вариант, остальные рассматриваются аналогично. Пусть $F = x_i \cdot H$. Сначала докажем вспомогательное утверждение.

Утверждение. *Если F минимальна, то H не содержит x_i.*

Доказательство. Заметим, что

$$x_i \cdot H \equiv x_i \cdot H|_{x_i=1},$$

т. е. если H содержит x_i, то вместо x_i можно подставить 1 и сделать соответствующие упрощения формулы. □

Теперь выпишем вероятности. С вероятностью $\frac{1}{2n}$ в x_i подставляем константу 0, получаем $F' = 0$, $L'_F = 0$. С вероятностью $\frac{1}{2n}$ в x_i подставляем константу 1, получаем $F' = H$, $L'_F = L'_H$. Наконец, с вероятностью $1 - \frac{1}{n}$ переменная x_i не меняется, и мы после упрощений в H получим формулу x_iH', сложность которой равна $1 + M(L'_H|_{x_i=*})$. Обозначим последнее матожидание через X. Тогда $L'_F \leqslant 1 + X$, так как не исключено, что H' является константой и переменную x_i также можно удалить.

Оценим X. Для того, чтобы найти условное матожидание, применим формулу полной вероятности.

$$M(L'_H) = \frac{1}{2n}M(L'_H|_{x_i=0}) + \frac{1}{2n}M(L'_H|_{x_i=1}) + \Big(1-\frac{1}{n}\Big)X =$$
$$= \frac{1}{2n}L_H + \frac{1}{2n}L_H + \Big(1-\frac{1}{n}\Big)X = \frac{1}{n}L_H + \Big(1-\frac{1}{n}\Big)X.$$

Мы использовали то, что H не зависит от x_i. Далее, в силу предположения индукции для H, получим

$$\frac{1}{n}L_H + \Big(1-\frac{1}{n}\Big)X = M(L'_H) \leqslant L_H\Big(1-\frac{3}{2n}\Big). \qquad (2)$$

Окончательно,

$$M(L'_{x_iH}) \leqslant \frac{1}{2n}L_H + \Big(1-\frac{1}{n}\Big)(X+1) \overset{(2)}{\leqslant}$$

$$\overset{(2)}{\leqslant} \frac{1}{2n}L_H + \Big(1-\frac{1}{n}\Big) + L_H\Big(1-\frac{3}{2n}\Big) - \frac{1}{n}L_H =$$

$$= (L_H+1)\Big(1-\frac{3}{2n}\Big) - \frac{1}{2n}L_H + \frac{1}{2n} \leqslant$$

(поскольку $L_H \geqslant 1$)

$$\leqslant (L_H+1)\Big(1-\frac{3}{2n}\Big) = L_F\Big(1-\frac{3}{2n}\Big). \qquad \square$$

Обозначим $\lambda_n = L_{B_0}(x_1 \oplus \ldots \oplus x_n)$. С помощью леммы 4 получим нижнюю оценку для λ_n. Пусть F — минимальная формула для функции $x_1 \oplus \ldots \oplus x_n$. Заметим, что в результате случайной подстановки A в эту функцию получаются функции вида $x_1 \oplus \ldots \oplus x_{i-1} \oplus x_{i+1} \oplus \ldots \oplus x_n \oplus c$, имеющие одну и ту же сложность λ_{n-1}, поэтому

$$\lambda_n = L(F) \geqslant \frac{M(L'_F)}{1-\frac{3}{2n}} \geqslant \frac{\lambda_{n-1}}{1-\frac{3}{2n}}.$$

Отсюда по индукции можно получить (как это сделано в моей книжке [19]) оценку $\lambda_n \geqslant n^{3/2}$. Но мы приведём первоначальное доказательство Субботовской, использующее факты из анализа, а именно, неравенство $\ln(1+x) \leqslant x$ и оценку $\sum\limits_{i=1}^{n} \frac{1}{n} = \ln n + c + o(1)$. Имеем

$$\lambda_n \geqslant \frac{\lambda_{n-1}}{1-\frac{3}{2n}} \geqslant \frac{\lambda_{n-2}}{(1-\frac{3}{2n})(1-\frac{3}{2(n-1)})} \geqslant \ldots \geqslant \frac{\lambda_1}{(1-\frac{3}{2n})\ldots(1-\frac{3}{2\cdot 2})} =$$

(поскольку $\lambda_1 = 1$)

$$= \exp\{-\sum_{i=2}^{n}\ln(1-\frac{3}{2i})\} \geqslant \exp\{-\sum_{i=2}^{n}(-\frac{3}{2i})\} \sim e^{3/2(\ln n + c')} = c_1 n^{3/2}.$$

Таким образом, основное утверждение этой лекции доказано.

Сейчас немного поговорим об истории вопроса. Доказанный нами результат о нелинейной нижней оценке сложности в базисе B_0 для линейной функции, из которого следует существование неэквивалентных базисов, был получен Субботовской в 1961 г. Сама задача о сложности линейной функции в классе π-схем, или что то же самое, в базисе B_0, представляет самостоятельный интерес как в математическом, так и в историческом плане. По истории этого вопроса можно написать целый очерк, тем более, что заключительная точка здесь ещё не поставлена. Расскажем вкратце основные моменты.

На заре кибернетики основным изучаемым классом "управляющих систем" были контактные схемы, а их важным подклассом — параллельно-последовательные (π-) схемы. Пример π-схемы, реализующей функцию $x \oplus y = x\overline{y} \vee y\overline{x}$, приведён на рис. 12.

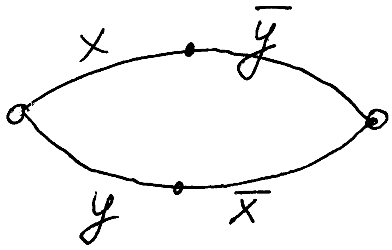

Рис. 12

Задачей о сложности линейной функции занимался ещё Шеннон в 30–40-е гг. [21]Он рассмотрел эту функцию в классе π-схем и получил для неё точную (!) оценку сложности. Она была экспоненциальной по n, что-то вроде $3 \cdot 2^{n-1}$, причём доказательство занимало один абзац. Позднее его соавтор по следующей статье Риордан указал на ошибку в его рассуждении и показал, что сложность на самом деле есть $O(n^2)$. Давайте продемонстрируем, почему это действительно так. Будем использовать метод "деления пополам".

Например, как реализовать сумму четырёх переменных?

$$(x \oplus y) \oplus (z \oplus t) = (x \oplus y)(\overline{z \oplus t}) \oplus (\overline{x \oplus y})(z \oplus t).$$

Мы сгруппировали четыре переменных по две, затем применили известную формулу. Далее следует применить эту же формулу для каждой скобки, и в результате мы получим формулу сложности 16. Если $n = 2^k$, то сложность формулы будет равна (проделав всё то же самое) $4^k = n^2$. Если n не есть степень двойки, то можно взять ближайшую к ней сверху степень двойки, построить формулу для неё, и затем лишние переменные обнулить. При этом мы уменьшим число переменных не более, чем в 2 раза, а сложность по порядку есть n^2. Итого, сложность равна $O(n^2)$.

Затем, когда кибернетику в Советском Союзе разрешили, этой проблемой занимался Яблонский. Он в 1954 г. получил оценку $\lambda_n \leqslant \frac{9}{8}n^2$ [22]и хотел получить нижнюю оценку (по частному сообщению О. Б. Лупанова) так, чтобы она по порядку была равна n^2. Но этого ему сделать не удалось, а оценку $n^{3/2}$ он якобы знал, но публиковать не стал, поскольку считал, что должна быть более высокая оценка n^2. Как это принято в некоторых научных школах, студентам дают уже кем-то решённые задачи, неопубликованные или опубликованные в недоступном месте, чтобы, например, гарантированно была дипломная работа. Теперь этот результат закреплён за Субботовской. На Западе метод "забивания переменных" широко известен, имя Субботовской вошло в историю.

Именно метод "забивания" применяется для получения нижних оценок сложности в любом базисе. Хастад получил оценку n^3 [25]по большей части с помощью именно его. Может быть, и метод Храпченко можно сформулировать на языке "забивания переменных". Также и у Нечипорука в его методе [5]встречаются подфункции, которые получаются подстановкой констант вместо переменных.

Вообще, в кибернетике есть такое основное понятие как "черный ящик", т. е. устройство, неизвестно (для нас) как внутренне устроенное, но известно, как реагирующее на внешние сигналы. Так вот формулу можно рассматривать как "черный ящик": мы не можем представлять себе её целиком, потому что она может быть устроена слишком сложно, хаотично. Но что мы можем делать с формулой? Подставлять вместо переменных константы, некие входы "обнулять" и смотреть, что будет получаться в итоге. Придумать какой-то принципиально отличный от этого метод можно, наверно, в очень узких классах схем.

В методе Храпченко [13]делается не так, но Хастад в своем доказательстве использует его соображения; он об этом прямо говорит в одной своей лемме, где вместо всех переменных, кроме одной, подставляются константы, а затем подсчитывается средняя сложность получаемой подформулы. Если этот момент проанализировать, то оба метода, видимо, можно свести воедино, тем более, что теорема Храпченко — не последнее слово. Её неоднократно обобщали; наиболее полный вариант метода Храпченко пока не сформулирован.

Итак, оценку n^2, которую хотел получить Яблонский, удалось в 1971 г. доказать Храпченко [12]. Его теорема — наиболее красивый результат в теории нижних оценок сложности. По существу, эта нижняя оценка не была повышена. Правда, в 1994 г. Рычков (Новосибирск) немного повысил нижнюю оценку [7], т. е. для некоторых n нижняя оценка стала n^2+2, для некоторых — n^2+3. Работа очень большая, порядка 20 страниц. В моей книжке содержится таблица, в которой указаны все известные оценки для λ_n при малых n.

Замечание. Уже после прочтения спецкурса, летом 2007 года, было найдено точное значение сложности линейной функции в базисе B_0 [26]. Как и предполагалось, схемы Яблонского являются оптимальными.

Лекция 5 (15.11.06)

Сегодняшняя лекция продолжает вторую часть курса, посвящённую сложности линейной функции в различных базисах. В прошлый раз мы обсуждали основной базис B_0. Сегодня будет рассмотрен ещё один базис, а в следующий раз, возможно, все остальные обобщённо-монотонные базисы, поскольку для них рассуждения будут аналогичны сегодняшним.

Через B_1 обычно обозначают базис $B_0 \cup \{\oplus\}$, а через B_2 — базис $B_0 \cup \{m(x,y,z)\}$, где $m(x,y,z)$ — медиана (функция, равная 1, если большинство переменных равны 1). Наша цель — получить нелинейную нижнюю оценку сложности линейной функции $(x_1 \oplus \ldots \oplus x_n)$ в базисе B_2; тем самым мы будем иметь результат о сравнимости базисов B_1 и B_2. Действительно, поскольку линейная функция в B_1 имеет линейную сложность, а в B_2 — нелинейную, то B_2 не предшествует B_1 ($B_2 \not\preceq B_1$). Если же мы рассмотрим объединение базисов B_1 и B_2, то получим, что $B_1 \cup B_2 \prec B_2$.

Начнем, как всегда с существенного — нижних оценок. Затем рассмотрим верхние, и наконец, дадим комментарии. Будем опираться на некоторые вещи, которые были в прошлый раз. Техника доказательства будет такой же, т. е. мы рассмотрим случайную подстановку констант и докажем похожее неравенство.

Напомним, что все рассматриваемые формулы зависят от переменных $x_1, \ldots, x_n$. Введём следующую подстановку констант A: с вероятностью $\frac{1}{2n}$ в переменную x_i подставляется константа σ ($1 \leqslant i \leqslant n$; $0 \leqslant \sigma \leqslant 1$), т. е. мы имеем полный набор вероятностей.

В отличие от прошлой лекции, базис другой: он содержит медиану. Поясним, чем он отличается от предыдущего и каким образом устроено забивание в данном случае; забивание не переменных, а неких других объектов. Как ведет себя медиана при подстановке констант?

$$m(0, x, y) = xy, \quad m(1, x, y) = x \vee y.$$

Мы видим, что подстановка константы вместо одной переменной не удаляет других переменных; они остаются. Поэтому такие простые рассуждения, как для базиса B_0, здесь не проходят. И не сразу было осознано, что здесь нужно делать, вернее, м. б., сначала было сделано, а потом осознано.

Как же нужно организовать забивание в данном случае? Идея такая: забивается сама функция медианы, поскольку при любой подстановке она исчезает; вместо неё остаются конъюнкция или дизъюнкция, а они уже являются функциями простыми, для них возможно забивание переменных. Другими словами, если мы в медиану подставим вторую константу, то с какой-то вероятностью третья переменная окажется несущественной. Таким образом, мы будем учитывать число медиан в формуле; по сути, мы введём новую меру сложности, которая будет учитывать число переменных, входящих в формулу, и (с каким-то коэффициентом) число медиан. Относительно новой меры будет происходить забивание — гарантированное уменьшение сложности на величину, пропорциональную количеству подставляемых констант, а дальше уже все аналогично ранее рассмотренному.

Какой вес нужно придать медиане для того, чтобы получилась максимально возможная оценка? Посмотрим, что было в прошлый раз. Если в формулу xy вместо x подставлять случайную константу, то в половине случаев, а именно, когда константа равна 0, вторая переменная y удаляется. Можно сказать, что подстановка одной константы вместо одного вхождения переменной удаляет ещё $\frac{1}{2}$ вхождения некой другой переменной (в вероятностном смысле).

В нашем случае при подстановке одной константы вместо одной переменной удаляется медиана. Поэтому разумнее считать, что медиана весит $\frac{1}{2}$, чтобы и в том и другом случае сложность уменьшалась на $\frac{1}{2}$. Это можно пояснить принципом сообщающихся сосудов: если здесь сделаешь уровень меньше — там он станет больше. При этом максимальный уровень минимизируется, когда жидкость и здесь и там на одном уровне. Так и определяется новая мера сложности: число вхождений переменных плюс $\frac{1}{2}$ от числа медиан.

Давайте сразу сформулируем лемму. Обозначения те же. Четыре условия, бывшие в прошлый раз в лемме, остаются теми же самыми. Какие это условия? То, что формула зависит только от переменных $x_1, \ldots, x_n$, не равна константе, существенно зависит хотя бы от двух переменных и формула является минимальной. Мы рассматриваем формулу F, удовлетворяющую этим четырем условиям. Напомним, F — формула; L_F — её сложность. У нас есть дополнительная мера: число медиан в формуле F; обозначим её через m_F. Когда мы действуем на формулу F случайной подстановкой констант A, она переходит в случайную формулу F', т. е. $F \xrightarrow{A} F'$. Соответственно, у F' есть сложность — случайная величина L'_F. У F' также есть случайное число медиан, которое обозначим через m'_F. Наконец, сформулируем лемму.

Лемма 5. *При соблюдении вышеизложенных четырех условий справедливо следующее неравенство:*

$$M(L'_F) + \frac{1}{2}M(m'_F) \leqslant L_F\Big(1 - \frac{3}{2n}\Big) + \frac{1}{2}m_F.$$

Заметим, что, в отличие от коэффициента при сложности L_F, константа $1/2$ при m_F не изменяется. Действительно, медиана — функциональный символ, а не переменная. Поэтому, если мы подставим константу в формулу $m(H_1, H_2, H_3)$, то со внешним вхождением медианы мы ничего сделать не сможем (не всегда сможем его забить), потому что формулы $H_1, \ldots, H_3$ могут быть большими и при нашей подстановке могут не обратиться в константу. Но все равно при забивании сложность будет уменьшаться: не в $3/2$ раза, а немного меньше.

Доказательство. Вначале выпишем все случаи, которые нам нужно рассмотреть.

Базис индукции: $F = xy$ и другие похожие формулы (можно брать отрицание и конъюнкцию и дизъюнкцию менять местами), а также (отдельно) медиана $F = m(x, y, z)$.

Индуктивный переход: формулы GH, xH (подразумевается, что у G и H сложность не меньше двух, поскольку предположение индукции можно применять только к таким формулам) и четыре случая с медианой, которые мы рассмотрим совместно: $m(x, y, z)$ (несмотря на то, что этот случай из базиса индукции), $m(H_1, H_2, H_3)$, $m(x, H_2, H_3)$, $m(x, y, H)$.

Случай xy непосредственно следует из того, что было в прошлый раз. Действительно, данная формула не содержит медиан, т. е. это формула над базисом B_0, который был уже рассмотрен. Случаи GH и xH сложнее, поскольку G и H могут содержать медиану. Мы не всё будем выписывать, а частично сошлемся на то, что было доказано в предыдущий раз.

Разберём случай $F = GH$. Он доказывается просто, поскольку для обеих мер справедлива аддитивность:

$$L_{GH} = L_G + L_H, \quad m_{GH} = m_G + m_H.$$

Кроме того, для "штрихованных" мер верны неравенства

$$M(L'_{GH}) \leqslant M(L'_G) + M(L'_H), \quad M(m'_{GH}) \leqslant M(m'_G) + M(m'_H).$$

(неравенства могут быть строгими в том случае, когда формулу GH после подстановки можно упростить за счёт влияния подформул G и H друг на друга).

Тогда по предположению индукции можно выписать следующее:

$$M(L'_{GH})+\frac{1}{2}M(m'_{GH})\leqslant M(L'_G+L'_H+\frac{1}{2}m'_G+\frac{1}{2}m'_H)\leqslant$$

$$\leqslant (L_G+L_H)\Big(1-\frac{3}{2n}\Big)+\frac{1}{2}(m_G+m_H)=L_F\Big(1-\frac{3}{2n}\Big)+\frac{1}{2}m_F.$$

Теперь рассмотрим случай $F=xH$. Всё повторять не будем. Почему же лемма будет верна? В прошлый раз мы вывели, что из $M(L'_H)\leqslant L_H(1-\frac{3}{2n})$ следует $M(L'_{xH})\leqslant L_{xH}(1-\frac{3}{2n})$. При этом мы использовали только то, что x не входит в H, а структура H и то, над каким базисом рассматривается формула, оказалось неважным.

Теперь у нас появились дополнительные слагаемые. Давайте выпишем их. По предположению индукции имеем:

$$M(L'_H)\leqslant L_H\Big(1-\frac{3}{2n}\Big)+\frac{1}{2}m_H-\frac{1}{2}M(m'_H).$$

При этом доказать нужно следующее:

$$M(L'_{xH})\leqslant L_{xH}\Big(1-\frac{3}{2n}\Big)+\frac{1}{2}m_{xH}-\frac{1}{2}M(m'_{xH}).$$

Заметим, что $m_H=m_{xH}$ и $m'_H\geqslant m'_{xH}$. То есть нам нужно из неравенства $M(L'_H)\leqslant L_H(1-\frac{3}{2n})+c_1$ вывести неравенство $M(L'_{xH})\leqslant L_{xH}(1-\frac{3}{2n})+c_2$, где $c_1\geqslant c_2$. Достаточно вывести второе неравенство для $c_2=c_1$.

Как мы вообще выводим одно линейное неравенство из других? С линейными неравенствами мы можем делать следующие операции: умножать их на положительные коэффициенты и складывать с другими неравенствами. В этом и состоит теорема двойственности линейного программирования: одно линейное неравенство есть следствие других тогда и только тогда, когда найдутся такие положительные веса, на которые можно умножить наши данные неравенства и сложить их, чтобы получить требуемое.

Можно явно посмотреть в прошлой лекции, что доказательство устроено именно так: берётся данное нам неравенство, складывается с другими, в которых мы уже используем подстановки вместо переменной x и получается требуемое. Еще важно, на какой коэффициент мы умножим наше исходное неравенство. Оказывается, что этот коэффициент равен 1. Почему? Сложность формулы H, этого "черного ящика", о котором мы ничего не знаем, должна входить в неравенство с коэффициентом 1. При этом константа c_1 также умножится на 1, т. е. если мы произведем все нужные выкладки для неравенства с "добавкой" c_1, то для константы коэффициент не изменится. Подчеркнём, что это некие наводящие соображения, а если кто хочет проделать все строго, тому оставим это в качестве упражнения.

Сейчас мы перейдем к основному случаю, который мы сразу будем доказывать в общем виде. Это будет прообраз самого общего рассуждения, которое будет в следующий раз проведено для произвольного базиса. Рассмотрим следующую формулу: $F = m(x_1^{\sigma_1}, \dots, x_k^{\sigma_k}, H_{k+1}, \dots, H_3)$, $0 \leqslant k \leqslant 3$, т. е. имеется всего три подформулы. Сначала уточним, могут ли среди переменных $x_1, \dots, x_k$ быть одинаковые? Заметим, что $m(x, x, y) = x$ и $m(x, \overline{x}, y) = y$. Следовательно, если есть одинаковые переменные, даже с отрицанием, то формула не является минимальной.

Рассмотрим два случая:

а) с вероятностью $\frac{1}{n}$ вместо x_i подставляется константа σ $(1 \leqslant i \leqslant k,\ 0 \leqslant \sigma \leqslant 1)$; всего $2k$ возможностей;

б) с вероятностью $1 - \frac{k}{n}$ вместо первых k переменных ничего не подставляется, т. е. $x_1 = *, \dots, x_k = *$; это событие обозначим через $*$.

Что получается в случае а)? Если хотя бы одну константу подставили, то внешняя медиана удалится. Будет конъюнкция или дизъюнкция. Как это записать в общем виде? Для единообразия обозначений положим $x_1^{\sigma_1} = H_1, \dots, x_k^{\sigma_k} = H_k$. Тогда после подстановки какой-либо константы останется формула $H'_{i_1} \circ H'_{i_2}$ $(i_1, i_2 \neq i$, $\circ \in \{\&, \vee\}$, $\{i, i_1, i_2\} = \{1, 2, 3\})$.

В случае б) внешняя медиана сохранится и мы получим формулу $m(x_1, \dots, x_k, H'_{k+1}, \dots, H'_3)$ (вернее, до такого уровня мы сможем сократить формулу заведомо). Выпишем меры сложности, точнее, их матожидания. Для случая а):

$$M(L'_F \mid x_i = \sigma) \leqslant k - 1 + \sum_{j=k+1}^{3} M(L'_{H_j} \mid x_i = \sigma),$$

$$M(m'_F \mid x_i = \sigma) \leqslant \sum_{j=k+1}^{3} M(m'_{H_j} \mid x_i = \sigma).$$

Для случая б):

$$M(L'_F|*) \leqslant k + \sum_{j=k+1}^{3} M(L'_{H_j} \mid *), \quad M(m'_F|*) \leqslant 1 + \sum_{j=k+1}^{3} M(m'_{H_j} \mid *).$$

Нам остается все это просуммировать:

$$M(L'_F) + \frac{1}{2} M(m'_F) \leqslant$$

$$\leqslant \frac{1}{2n} \sum_{i=1}^{k} \sum_{\sigma=0}^{1} \Big[k - 1 + \sum_j M(L'_{H_j} | x_j = \sigma) + \frac{1}{2} \sum_j M(m'_{H_j} \mid x_i = \sigma) \Big] +$$
$$+ \Big(1 - \frac{k}{n}\Big) \Big[k + \sum_j M(L'_{H_j} \mid *) + \frac{1}{2} \Big(1 + \sum_j M(m'_{H_j} \mid *)\Big) \Big] =$$
$$= \sum_j M(L'_{H_j}) + \frac{1}{2} \sum_j M(m'_{H_j}) + \Big[\frac{1}{2n} \cdot 2k(k-1) + \Big(1 - \frac{k}{n}\Big)\Big(k + \frac{1}{2}\Big) \Big] \leqslant$$

(применяем предположение индукции)

$$\leqslant \sum_j \Big[L_{H_j} \Big(1 - \frac{3}{2n}\Big) + \frac{1}{2} m_{H_j} \Big] + [\ldots] =$$

(т. к. $L_F = k + \sum\limits_j L_{H_j}$, $m_F = 1 + \sum\limits_j m_{H_j}$)

$$= L_F \Big(1 - \frac{3}{2n}\Big) + \frac{1}{2} m_F - k\Big(1 - \frac{3}{2n}\Big) - \frac{1}{2} + [\ldots].$$

Для доказательства леммы мы должны показать, что

$$M(L'_F) + \frac{1}{2}M(m'_F) \leqslant L_F\Big(1 - \frac{3}{2n}\Big) + \frac{1}{2}m_F.$$

Нам осталось проверить, что

$$-k\Big(1 - \frac{3}{2n}\Big) - \frac{1}{2} + \frac{2k(k-1)}{2n} + \Big(1 - \frac{k}{n}\Big)\Big(k + \frac{1}{2}\Big) = 0.$$

Проделав выкладки, убедимся в этом. Лемма доказана. □

Что делать дальше? Пока неясно, как применять доказанное неравенство рекурсивно. Нужно ещё одно неравенство (на второй лекции было похожее). Мы хотим получить связь между медианной мерой и обычной.

Утверждение 2. $m_F \leqslant \frac{1}{2}(L_F - 1)$.

Доказательство. Рассмотрим дерево формулы F (см. рис. 13).

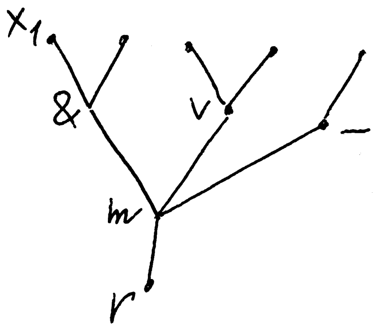

Рис. 13

Добавим ещё одно ребро ниже корня. Это нужно для того, чтобы у всех вершин с медианой была одна и та же степень (число инцидентных ребер). Будем использовать известную формулу:

$$\sum d(v) = 2P = 2(B - 1). \tag{3}$$

Подсчитаем отдельно количество вершин с одной и той же степенью. Вершины с $d(v) = 1$ — это переменные (листья дерева) и корень (root) r. Их число равно $L_F + 1$. Вершинами с $d(v) = 2$ являются отрицания $\neg$. Обозначим их количество через x. Пусть y — число вершин с $d(v) = 3$. Ими будут являться & и $\vee$. И наконец, остались вершины с $d(v) = 4$. Это медианы m; их число равно m_F.

Выпишем полностью сумму степеней вершин: $L_F + 1 + 2X + 3Y + 4m_F$. Известно также количество вершин: $L_F + 1 + X + Y + m_F$. Тогда по формуле (3) имеем:

$$L_F + 1 + 2X + 3Y + 4m_F = 2(L_F + 1 + X + Y + m_F - 1).$$

Выделим m_F:

$$2m_F = L_F - Y - 1 \leqslant L_F - 1.$$

Утверждение доказано. $\square$

Теперь, используя полученную оценку, мы сможем найти в неравенстве

$$L_F\Big(1 - \frac{3}{2n}\Big) + \frac{1}{2}m_F \leqslant \Big(L_F + \frac{1}{2}m_F\Big)\Big(1 - \frac{\alpha}{n}\Big) \tag{4}$$

неизвестное α. Этот параметр будет немного меньше $\frac{3}{2}$, а именно $\frac{6}{5}$. Докажем это.

Преобразуем неравенство (4), перенеся все слагаемые в левую часть:

$$L_F\Big(-\frac{3}{2n} + \frac{\alpha}{n}\Big) + \frac{1}{2}m_F \cdot \frac{\alpha}{n} \leqslant 0.$$

Мы хотим доказать, что при определенном α левая часть неравенства не больше 0. Для этого воспользуемся утверждением 2: $m_F \leqslant \frac{1}{2}L_F$. Имеем:

$$L_F\Big(-\frac{3}{2n} + \frac{\alpha}{n}\Big) + \frac{1}{2}m_F \cdot \frac{\alpha}{n} \leqslant L_F\Big(\frac{3}{2n} + \frac{\alpha}{n} + \frac{\alpha}{4n}\Big) \leqslant 0.$$

Неравенство верно при $\alpha \leqslant \frac{6}{5}$.

Дальше буквально воспроизводятся рассуждения Субботовской из предыдущей лекции, которые можно даже сформулировать в общем виде.

Утверждение 3. *Пусть последовательность* $a_1, a_2, \ldots, a_n$ *удовлетворяет условиям*

$$a_1 = 1, \quad a_{n-1} \leqslant a_n\Big(1 - \frac{\alpha}{n}\Big).$$

Тогда $a_n \succeq n^{\alpha}$.

Мы доказали это для $\alpha = \frac{3}{2}$. Для других α все совершенно аналогично.

Применим утверждение 3. В нашем случае a_n — это комбинированная мера сложности. Т. е. для каждой формулы F рассматриваем меру $P_F = L_F + \frac{1}{2}m_F$, затем рассматриваем минимальную формулу с точки зрения этой меры для линейной функции и полагаем $a_n = P(x_1 \oplus \ldots \oplus x_n)$.

Мы знаем, что при подстановке константы в линейную функцию снова получаем линейную функцию, уже от $n-1$ переменной. Тогда по доказанному выше $a_{n-1} \leqslant a_n(1 - \frac{\alpha}{n})$. Следовательно, $P(x_1 \oplus \ldots \oplus x_n) \succeq n^{\frac{6}{5}}$. Заметим, что по ранее доказанному о связи между различными мерами сложности (Лемма 1) $m_F \preceq L_F$. В итоге получим $L_{B_2}(x_1 \oplus \ldots \oplus x_n) \succeq n^{1,2}$.

Дадим некоторые комментарии. Первой эту задачу рассматривала Субботовская. Под фамилией Мучник в 1970 году в киевском журнале "Кибернетика" она опубликовала свои результаты [4]. Оценку, которую мы сейчас доказали, она получила, рассматривая не только базис B_2, но и более широкий класс базисов. С ростом числа переменных у базисных функций показатель α при n в оценке сложности все более стремится к 1. Так это и остается по сегодняшний день, в следующий раз мы это рассмотрим. Если положить $k = n(B)$, где $n(B)$ — ранг базиса B (максимальное число переменных у базисных функций), то можно показать, что $\alpha = \frac{3k-3}{3k-4}$. У Субботовской эта оценка была получена не для всех возможных базисов, а для бесконечного семейства базисов, обладающих определенным свойством. Для получения более общего результата и облегчения доказательства требовалось введение новой меры сложности.

Затем исторически за эту задачу взялся Перязев, который в 1993 году приехал из Иркутска в Москву и, видимо, пообщавшись, узнал об этой задаче. В 1995 г. у него вышел препринт [6], в котором он распространил метод Субботовской уже на все допустимые базисы и получил критерий нелинейности сложности линейной функции для всех базисов. Он догадался, что нужна другая, вспомогательная, мера и ввел ее. Эта мера (если не изменяет память) отличается от нашей. Но основная лемма у Перязева доказывается нестрого. Возможно, сама мера возникала где-то и в тексте Субботовской, но явно не была ей выделена.

Затем в конце 1996 г., не зная о результатах Перязева, автором была получена [14]бесконечная цепочка базисов методами, отличными от Субботовской и Перязева. В 1999 г. автор, всё ещё не зная о результатах Перязева, переоткрыл их [17], обобщив результат Субботовской на все возможные базисы. Также можно добавить, что у Перязева базисы, в которых линейная функция имеет нелинейную сложность, не достаточно прозрачно описаны, а именно, с помощью неких предикатов. Это описание можно значительно упростить; речь идёт всего лишь о семействе обобщенно-монотонных базисов (каждая функция в которых возрастает или убывает по каждой переменной).

История с переоткрытиями на этом не закончилась. В 2001 г. Цвиг с ученицей опубликовали работу [24], в которой вероятностными методами получалась нижняя оценка с той же константой α и тот же критерий нелинейности (исключение: для B_2 у них получилась более высокая оценка). По словам Цвига, работа Субботовской 1970 г. была ему неизвестна.

Для базиса B_2 показатель α можно повысить. Каким образом? Когда Субботовская опубликовала свои результаты, еще не был известен результат Храпченко для базиса B_0. Поэтому, когда я узнал об известной нижней оценке для B_2 ($n^{1,2}$) и для B_0 (n^2), я предложил своему приятелю Безрукову Александру следующую идею: возьмем формулу в базисе B_2, раскроем там скобки для медианы по формуле $m(x,y,z) = x(y \vee z) \vee yz$, причем в качестве x возьмем формулу наибольшей сложности, получим формулу в B_0, а в этом базисе есть оценка Храпченко. При таких преобразованиях сложность увеличится в некоторую степень, т. е. $(n^\alpha)^\beta \geqslant n^2$. Константа β — фиксированная величина, зависящая от метода преобразования. Оценим α: $\alpha \geqslant \frac{2}{\beta}$. Уменьшая β, можно таким образом увеличивать значение для α. Вначале я предположил, что можно получить $\alpha = 1,3$, но у Безрукова получалось $\alpha = 1,26$. Он это сделал, применяя вариационное исчисление. И затем ученица Цвига действительно довела результат до $\alpha = 1,33$. Отметим, что метод Храпченко не перенесен ни на какой базис, отличный от B_0. Если бы это удалось, то, конечно, оценку можно было бы еще повысить.

Теперь, что касается верхних оценок. В 2001 г. я показал [18], что в любом базисе, кроме B_0 (и эквивалентных ему) сложность линейной функции не больше n^α, $\alpha < 2$. Можно сказать, что Перязев в своем препринте описал те базисы, в которых линейная функция имеет линейную (минимальную) сложность (мы это на следующей лекции сделаем), я же рассмотрел базисы с другой стороны, с точки зрения максимальной сложности.

Какие оценки известны в базисе B_2? Мы начали решать эту задачу опять вместе с Безруковым. Для $n = 2$ сложность $L = 4$, здесь ничего нельзя изменить. Нужно, как минимум, три переменные, чтобы применить медиану. Вначале я предложил Безрукову формулу сложности $L = 9$:

$$x \oplus y \oplus z = m(m(\overline{x}, y, z), m(x, \overline{y}, z), m(x, y, \overline{z}))$$

по аналогии с $x \oplus y = \overline{x}y \vee x\overline{y}$. Исходя из этой формулы удалось понизить верхнюю оценку для суммы n слагаемых до $o(n^2)$, что асимптотически ниже, чем в базисе B_0 (заметим, что в базисе B_0 сложность функции $x \oplus y \oplus z$ равна 10).

На следующий год встал вопрос: чему равна точная оценка для линейной функции от трех переменных в базисе B_2? Сначала Безрукову удалось показать, что сложность не менее 7 (вернее, он полагал, что не менее 8, но это оказалось ошибкой). Затем, анализируя вместе с ним его доказательство, мы пришли к формуле

$$x \oplus y \oplus z = m(m(\overline{x}, y, z), m(x, \overline{y}, z), \overline{z}), \tag{5}$$

т. е. удалось понизить верхнюю оценку с 9 до 7.

Формулу (5) можно объяснить следующим образом (позже Касим-заде сказал, что это некий общий метод). Рассмотрим две формулы в базисе B_0:

$$x \oplus y \oplus 0 = \overline{x}y \vee x\overline{y}, \tag{6}$$

$$x \oplus y \oplus 1 = (\overline{x} \vee y)(x \vee \overline{y}). \tag{7}$$

Заметим, что эти формулы отличаются только тем, что & и $\vee$ меняются местами. Вспомним, как меняется медиана при подстановке вместо какой-то переменной константы. Если подставить вместо z константу 0, получим xy; если $z = 1$, то получим $x \vee y$. Теперь можно легко проверить формулу (5) для медианы, используя вспомогательные формулы (6) и (7). Так же можно действовать для любой пары двойственных формул, построив формулу над базисом B_2. Позже я увидел формулу (5) в работе Субботовской ([4], 1970) в качестве примера формулы, представленной в виде дерева. Тем не менее, Безруков курсовые и дипломную работы [1]защитил нормально

Сейчас я расскажу верхнюю оценку, которая следует из формулы (5). Сформулируем общим факт, который пригодится нам в дальнейшем.

Лемма 6. *Пусть* $F \equiv x_1 \oplus \ldots \oplus x_k$ *— формула над базисом* B. *Тогда*

$$L_B(x_1 \oplus \ldots \oplus x_n) \leqslant O(n^{\log_k L(F)}).$$

В частности, для базиса B_2 получается $n^{\log_3 7} = n^{1{,}77\ldots}$.

Доказательство.

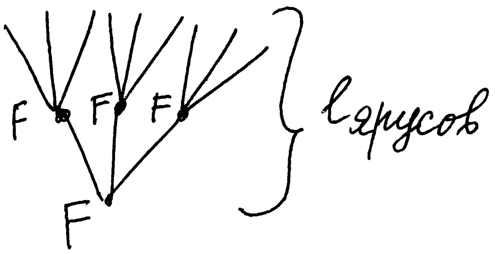

Рис. 14

Строим формулу по дереву (рис. 14), в котором l ярусов. Эта формула реализует функцию от $n = k^l$ переменных, а её сложность равна $(L(F))^l = L(F)^{\log_k n} = n^{\log_k L(F)}$ (мы использовали свойство логарифмов: $a^{\log_b c} = c^{\log_b a}$).

Таким образом, мы доказали лемму для случая, когда n — степень числа k. Если это не так, берем ближайшую сверху к n степень, строим для неё формулу и вместо лишних переменных подставляем константы. Итоговая сложность возрастёт (относительно n) не более, чем в константу раз. Лемма доказана. □

В формуле (5) можно заметить вот еще что: x и y входят по 2 раза, а z — 3 раза. Поэтому дерево, используемое в лемме 6, можно выращивать несимметрично: вместо x и y больше подставлять, а вместо z меньше. За счет этого оказывается, что можно немного сэкономить, то есть получить оценку $n^{1,73\dots}$. Константу $1,73\dots$ мы посчитали на компьютере с Безруковым, а потом, гораздо позже, я теоретически это показал [18]. Эта константа является корнем некоего трансцендентного уравнения. Т. е. для каждой формулы, куда переменные входят несимметрично, можно, строя неравномерное дерево, немного уменьшать сложность. В частности, таким способом я доказал, что для любого базиса сложность линейной функции меньше по порядку, чем n^2.

Резюмируем: в базисе B_2 сложность линейной функции находится в пределах $n^{1,33}\dots n^{1,73}$. Для других базисов известно еще меньше. На самом деле, для некоторых базисов нижнюю оценку Субботовской-Перязева можно повысить. Но с этим надо повозиться. Интересующиеся могут этой задачей заняться.

Лекция 6 (06.12.06)

Эта лекция закрывает второй цикл: нижние оценки сложности "в духе Субботовской". (Первый цикл был введением в нижние оценки сложности.) Что в этот цикл входит? Сам результат Субботовской, его обобщение на случай базиса с медианой, и наконец, сегодня — обобщение на обобщенно-монотонные базисы, т. е. на все базисы, где это обобщение возможно. Напомним, что значит, что функция монотонна.

Определение. *Функция монотонна ($f \in M$), если она возрастает по каждой из переменных.*

Определение. *Функция называется обобщенно-монотонной (о.-м.), если по любой переменной она убывает или возрастает.*

Определение. *Базис называется о.-м., если все функции в нем о.-м.*

Примеры о.-м. базисов:

1) $B_0 = \{\&, \vee, \neg\}$. Здесь & и $\vee$ — монотонны, а $\neg$ — о.-м., т. к. убывает по своей переменной.

2) $B_2 = B_0 \cup \{m\}$.

Вообще, что представляет собой о.-м. функция? Её можно представить в следующем виде:

$$f(x_1^{\sigma_1}, \ldots, x_n^{\sigma_n}), \quad f \in M. \tag{8}$$

Утверждение 4. $\forall$ *о.-м. базиса* B $\exists$ *базис* $B' = \{f_1, \ldots, f_k, \overline{x}\}$ $(f_i \in M)$, *эквивалентный* B.

Как найти базис B'? Нужно каждую функцию из базиса B представить в виде (8). Почему эти базисы эквивалентны? Это следует из доказанной ранее леммы 2: если функции из одного базиса бесповторно выражаются через функции другого базиса, то эти базисы эквивалентны. Почему функции выражаются бесповторно? Понятно, что когда в бесповторной формуле переменные заменяются на отрицания, формула останется бесповторной (т. к. отрицание бесповторно выражается в любом базисе). Поэтому каждая функция φ из базиса B бесповторно выражается через f и отрицание из B': $\varphi(x_1, \ldots, x_n) = f(x_1^{\sigma_1}, \ldots, x_n^{\sigma_n})$ и, наоборот, $f(x_1, \ldots, x_n) = \varphi(x_1^{\sigma_1}, \ldots, x_n^{\sigma_n})$.

На класс о.-м. базисов можно непосредственно распространить результат Субботовской. Почему класс базисов именно такой? Сформулируем ответ в виде леммы.

Лемма 7. *Если* B *— не о.-м. базис, то* $L_B(x_1 \oplus \ldots \oplus x_n) = O(n)$.

(Поэтому в данном случае метод забивания переменных константами не действует; первым об этом написал Перязев [6], но по-другому, более сложным языком.)

Доказательство. Поскольку B — не о.-м. базис, то в нём есть не о.-м. функция $f \in B$, т. е. по какой-то переменной (пусть по x_1)

она не возрастает и не убывает. Это значит, что найдутся четыре таких набора, что

$$f(0, \alpha_2, \ldots, \alpha_n) = 1, \quad f(1, \alpha_2, \ldots, \alpha_n) = 0,$$

$$f(0, \beta_2, \ldots, \beta_n) = 0, \quad f(1, \beta_2, \ldots, \beta_n) = 1.$$

Через функцию f выразим сумму двух переменных. Для этого введем

$$\varphi_i(x) = \begin{cases} \beta_i, & x = 0, \\ \alpha_i, & x = 1. \end{cases}$$

Тогда $x \oplus y = f(x, \varphi_2(y), \ldots, \varphi_n(y)) = \Phi(x, y)$. Это можно непосредственно проверить. Мы видим, что x входит в формулу Φ только один раз. Следовательно, можно подставлять в формулу Φ вместо x такие же формулы Φ:

$$\Phi(\ldots(\Phi(\Phi(x_1, x_2), x_3)\ldots), x_n).$$

В результате мы получим формулу для линейной функции от n переменных, сложность которой линейна. (Функции φ_i зависят от одного аргумента и реализуются в B с константной сложностью). Лемма доказана. $\square$

Теперь покажем обратное: если базис — о.-м., то линейная функция в нём имеет нелинейную сложность. Это также придумал Перязев [6], но в других терминах, и его доказательство не до конца прозрачно. Поэтому я представлю свое доказательство. Делается это так же, как и в случае базиса с медианой B_2, что было рассмотрено на прошлой лекции. Вводится дополнительная мера сложности, которая учитывает вхождение функций. Но сегодня мы будем излагать другим способом, более простым и самым детерминированным, который предложила еще Субботовская.

Введём меру сложности. Каждой формуле F соответствует мера $L(F)$ — число вхождений символов переменных. Теперь введём величину $M(F)$ — сумму весов функций, входящих в формулу. Для каждой функции $f(x_1, \ldots, x_l)$ определим её вес следующим образом:

$$M(f) = \begin{cases} \frac{l-2}{2}, & l \geqslant 3 \\ 0, & l \leqslant 2. \end{cases}$$

Вспомним пример с медианой: $M(m) = \frac{1}{2}$. Почему для медианы вес вводится именно так? Потому что в формуле xy переменная y (например) забивается в половине случаев, а медиана $m(x, y, z)$ всегда, при любой подставленной константе. Для уравновешивания выбирается коэффициент $\frac{1}{2}$.

Положим $M(F) = \sum M(f)$, и комбинированная мера есть $P(F) = L(F) + M(F)$. Мера P будет у нас основной, по ней будет вестись индукция. Как связаны меры L и P? Докажем следующую оценку.

Утверждение 5.

$$L(F) \geqslant \frac{2k-2}{3k-4}P(F) + \frac{k-2}{3k-4},$$

где $k = n(B)$ — порядок базиса.

Доказательство. Будем считать, что в формуле F отрицания стоят только над переменными. Почему? Если это не так, то заменим отрицание функции $\overline{f(\ldots)}$ на двойственную функцию и отрицание спустим на её аргументы. При этом мы расширим базис (добавив в него двойственные функции), однако значения сложности $P(F)$ и $L(F)$ не изменятся.

Будем доказывать индукцией по построению формулы. Основанием индукции будет переменная или ее отрицание. Тогда сложность в обоих смыслах равна 1.

Индуктивный переход. Пусть $F = f(x_1^{\sigma_1}, \ldots, x_s^{\sigma_s}, F_{s+1}, \ldots, F_t)$. Тогда $L(F) = s + \sum L(F_i)$, $P(F) = \frac{t-2}{2} + s + \sum P(F_i)$. Отметим, что в силу того, что отрицания в формуле навешаны только над переменными, $t \geqslant 2$.

Выпишем неравенство, которое нам нужно доказать:

$$s + \sum L(F_i) \geqslant \frac{2k-2}{3k-4}\Big(\frac{t-2}{2} + s + \sum P(F_i)\Big) + \frac{k-2}{3k-4}.$$

Воспользуемся предположением индукции об оценке для внутренней формулы F_i:

$$L(F_i) \geqslant \underbrace{\frac{2k-2}{3k-4}}_{=\alpha} P(F_i) + \underbrace{\frac{k-2}{3k-4}}_{=\beta}.$$

Суммируя по всем i, получим

$$s + \sum L(F_i) \geqslant s + \alpha \sum P(F_i) + (t-s)\beta.$$

Нам осталось доказать, что последнее выражение не больше, чем

$$\alpha \sum P(F_i) + \alpha \cdot \frac{t-2}{2} + s\alpha + \beta.$$

Покажем, что

$$s + (t-s)\beta \geqslant \alpha \cdot \frac{t-2}{2} + s\alpha + \beta.$$

Пользуясь тем, что $\alpha + \beta = 1$, и опуская промежуточные выкладки, приходим к

$$\alpha\Big(\frac{3}{2}t - 2\Big) \leqslant t - 1,$$

$$\frac{2k-2}{3k-4} = \alpha \leqslant \frac{t-1}{\frac{3}{2}t-2} = \frac{2t-2}{3t-4}.$$

Последнее неравенство верно в силу того, что $t \leqslant k$ и $t \geqslant 2$. Утверждение доказано. □

Теперь перейдем к оценке сложности. Далее будем считать, что базис B замкнут ($B = [B]$); ранее мы показали, что $B \sim [B]$. Докажем следующую лемму.

Лемма 8. *Пусть формула F над о.-м. базисом B удовлетворяет условиям:*

1) $L(F) \geqslant 2$;

2) F — минимальна;

3) $F \not\equiv c$;

4) формула является приведенной, т. е. отрицания в F навешаны только над переменными.

Тогда $\forall x_i$ $\exists c$

$$P(F) - P(F|_{x_i=c}) \geqslant \frac{3}{2} L_{x_i}(F)$$

($L_{x_i}(F)$ — *число вхождений* x_i *в* F).

Доказательство. Выкладок здесь почти не будет. Нужно усмотреть, почему это неравенство верно. Как x_i может входить в формулу? $F(x_i^\sigma, \ldots)$ Здесь возможны 2 случая.

1) $n(F) = 2$. Тогда $F = x_i^\sigma \circ H$. Поскольку B — замкнутый о.-м. базис, то можно считать, что $\circ \in \{\&, \vee\}$. Для каждого из четырех возможных случаев F имеем свое забивающее значение переменной: для $x_i H$ — 0, для $\overline{x}_i H$ — 1, $x_i \vee H$ соотв. 1 и для $\overline{x}_i \vee H$ — 0. Из условия минимальности F следует, что H не содержит переменной x_i, поэтому каждое вхождение x_i в формулу F забивает хотя бы одну другую переменную.

2) $n(F) \geqslant 3$. Тогда $F = f(x_i^\sigma, H_2, \ldots, H_t)$. При подстановке вместо x_i произвольной константы c формула F переходит в $\varphi(H_2, \ldots, H_t)$ (где φ — подфункция f, $\varphi \in [B]$), и вес формулы F снижается на $\frac{1}{2}$.

Теперь мы выберем константу c исходя из условия, что она является забивающим значением для большинства вхождений переменной x_i из случая 1), обозначим это число вхождений за m_1. Число вхождений переменной x_i из случая 2) обозначим за m_2. Общее число вхождений x_i есть $m = m_1 + m_2$. Тогда

$$P(F) - P(F|_{x_i=c}) \geqslant m + \frac{m_1}{2} + \frac{m_2}{2} = \frac{3}{2}m.$$

Лемма доказана. □

Сведем вместе лемму 8 и утверждение 5. Имеем:

$$P(F) - P(F|_{x_i=c}) \geqslant \frac{3}{2} L_{x_i}(F) \geqslant$$

(Выберем переменную x_i: в качестве неё возьмем переменную, входящую в F наибольшее число раз. Заметим, что число различных переменных, входящих в F, не больше, чем n.)

$$\geqslant \frac{3}{2n} L(F) \geqslant \frac{3}{2n} \frac{2k-2}{3k-4} P(F) + \ldots \geqslant \frac{1}{n} \frac{3k-3}{3k-4} P(F)$$

(мы воспользовались утверждением 5).

Преобразуем получившееся в итоге неравенство:

$$P(F)\left(1-\frac{c_k}{n}\right) \geqslant P(F|_{x_i=c}), \quad c_k = \frac{3k-3}{3k-4}.$$

Вспоминая утв. 4 из прошлой лекции, в итоге получим, что $L_B(x_1 \oplus \ldots \oplus x_n) \succeq n^{c_k}$, где $c_k > 1$, т. е. мы доказали нелинейную нижнюю оценку (учитывается, что меры P и L эквивалентны). По существу, это основной результат.

Осталось получить бесконечно убывающую цепочку базисов. Напомним уже доказанную лемму 6. Пусть $F \equiv x_1 \oplus \ldots \oplus x_k$, $L(F) = c$. Тогда $L_B(x_1 \oplus \ldots \oplus x_n) = O(n^{\log_k c})$.

Рассмотрим специальное семейство о.-м. базисов. Введем монотонную функцию:

$$f_k(x_1,\ldots,x_k,y_1,\ldots,y_k) = \begin{cases} 0, & \sum x_i + \sum y_i < k \\ 1, & \sum x_i + \sum y_i > k \\ x_1 \oplus \ldots \oplus x_k, & \sum x_i + \sum y_i = k \end{cases}$$

Тогда

$$f_k(x_1,\ldots,x_k,\overline{x}_1,\ldots,\overline{x}_k) = x_1 \oplus \ldots \oplus x_k.$$

Рассмотрим базис $B_k = B_0 \cup \{f_1,\ldots,f_k\}$. Среди этих базисов можно выбрать интересующую нас цепочку. Почему? У нас есть такие оценки:

$$n^{\log_k 2k} \succeq L_{B_k}(x_1 \oplus \ldots \oplus x_n) \succeq n^{c_k}, \quad n \to \infty.$$

Можно так выбрать последовательность чисел $\{k_i\}$, что каждая следующая верхняя оценка будет ниже, чем предыдущая нижняя оценка. Покажем это. Последовательности $\{c_k\}$ и $\{\log_k 2k\}$ убывают и стремятся к 1 при $k \to \infty$. Выберем $\{k_i\}$ следующим образом. Положим $k_1 = 2$; нижняя оценка для этого k_1 равна $n^{c_{k_1}}$. Число k_2 выберем так, чтобы $\log_{k_2} 2k_2 < c_{k_1}$ и т. д. k_i выбирается из такого же соображения $\log_{k_i} 2k_i < c_{k_{i-1}}$.

Тогда $B_{k_1} \succ B_{k_2} \succ \ldots$, т. к. в каждом последующем базисе линейная функция реализуется существенно проще, чем в предыдущем (в силу выбора соответствующего k_i). Кроме того, каждый последующий базис не хуже предыдущего, т. к. включает его.

Глава III. Сравнение произвольных базисов

Лекция 7 (13.12.06)

Этой лекцией мы начинаем третий цикл нашего курса, который продолжится в следующем семестре и завершится каким-то крупным результатом, возможно, ослабленным вариантом критерия сравнения произвольных базисов. Техника этого раздела берет свое начало с работы Субботовской 1963 г., которая была затем мной развита и применена к другим базисам. Она рассматривала базис B_0 и получила критерий эквивалентности произвольного базиса базису B_0. А именно, мы уже говорили, что есть признак сравнения базисов:

$B_1 \preceq B_2 \leftrightarrow \forall f \in B_2$ бесповторно выражается в B_1.

Ранее мы доказывали, что любой базис B не хуже B_0 ($B \preceq B_0$). Встает вопрос: для каких базисов B верно обратное, что $B_0 \preceq B$? Если для базисов выполнены оба включения, то базисы эквивалентны. Т. о., спрашивается, какие базисы эквивалентны B_0? И Субботовская доказала, что этот признак справедлив также и в обратную сторону: $B_0 \preceq B \leftrightarrow \forall f \in B$ бесповторно выразима в B_0.

В одну сторону доказательство следует из признака. Мы докажем Утверждение в другую сторону: $B_0 \preceq B \Rightarrow \forall f \in B$ бесповторно выразима в B_0. И тем самым, мы получим классификацию всех базисов, эквивалентных B_0.

При доказательстве мы будем в основном следовать работе Субботовской 1963 г.

Пусть дана некоторая функция $f(x_1, \ldots, x_n)$. Определим, что значит, что одна переменная этой функции забивает и другую.

Определение. x_i забивает x_j, если $\exists c$, что $f|_{x_i=c}$ существенно не зависит от x_j.

Мы уже встречались с забиванием в методе Субботовской применительно к формулам. Здесь же мы рассматриваем забивание применительно к функциям. Т. е. есть такое отношение на множестве переменных данной функции — одна переменная забивает другую. Это отношение можно задать в виде стрелочек.

Рис.

Это отношение транзитивно: если x_i забивает x_j и x_j забивает x_k, то x_i забивает x_k. Покажем это.

По определению, $f|_{x_i=c}$ не зависит от x_j, $f|_{x_j=d}$ не зависит от x_k. Тогда $f|_{x_i=c,x_j=d} = f|_{x_i=c}$, поскольку последняя функция не зависит от x_j. Первая же функция, являясь подфункцией $f|_{x_j=d}$, не зависящей от x_k, сама не зависит от x_k.

Еще одно определение. x_i называется отмеченной переменной, если $\forall j$ x_i не забивает x_j.

Пример 1.

$f(x,y,z) = xy \vee \overline{x}z$, y,z — отмеченные переменные.

Затем Субботовская вводит множество $\Phi = \{f :$ никакая подфункция f не содержит отмеченных переменных, в том числе, и сама f $\}$.

Может ли какая-нибудь функция не содержать отмеченных переменных, а ее подфункции - наоборот, содержать?

Пример 2.

Возьмем f из примера 1 и рассмотрим следующую функцию $f(x_1x_2, y_1y_2, z_1z_2)$. Тогда эта функция не принадлежит Φ, поскольку x_i друг друга забивают, то же относится и к y_i и z_i. А ее подфункция, получающаяся при $x_2 = y_2 = z_2 = 1$ имеет отмеченные переменные.

Какие же функции содержатся в Φ?

Лемма.

$f \in \Phi \leftrightarrow f$ бесповторно выразима в B_0.

Доказательство.

$\Leftarrow$ f бесповторно выразима. Почему она не содержит отмеченных переменных?

Вхождение произвольной переменной x_i в бесповторно выразимую формулу имеет такой вид: $(x_i^\sigma \circ F)$. При какой-то константе x_i забивает подформулу F, поэтому X_i уже не будет отмеченной переменной. Следовательно, бесповторно выразимая в B_0 функция f не содержит отмеченных переменных, и никакая ее подфункция, также бесповторно выразимая в B_0, не может содержать отмеченных переменных. Т. е. $f \in \Phi$.

В обратную сторону. $\Rightarrow$

Допустим, что $f \in \Phi$.

Докажем сначала вспомогательное утверждение.

Утверждение.

Пусть $f(x_1, \dots, x_n)$, $n > 2$.

Тогда $\exists f_1, \varphi \in \Phi$, что $f(x_1, \dots, x_n) = f_1(\varphi(x_{i_1}, x_{i_2}), x_{i_3}, \dots, x_{i_n})$.

Т. е. функцию можно записать в виде формулы, куда входит подформула данного вида, зависящая только от двух переменных. Такая группа из двух переменных называется выделимой.

Доказательство.

Рассмотрим систему стрелочек. У нас нет отмеченных переменных, из каждой переменной отходит какая-то стрелка.

Рис. 1

Поскольку переменных конечное число, в какой-то момент мы вернемся в ту переменную, где уже побывали.

Рис. 2

Рассмотрим такой цикл. x_{i_1} забивает x_{i_2} и т. д. по цепочке, завершает цикл x_{i_k}, забивающая X_{i_1}. Тогда, в силу транзитивности, x_{i_2} забивает x_{i_1}. Т. О., мы получили пару переменных, которые забивают друг друга.

Эта пара переменных и будет выделенной парой. Почему это будет так? Надо убедиться, что при подстановке вместо этих переменных различных констант у нас может получиться не более двух различных подфункций.

Рис.

x_{i_1} забивает x_{i_2} с константой c, а x_{i_2} забивает x_{i_1} с константой d.

Тогда $f|_{x_{i_1}=c, x_{i_2}=d} = f|_{x_{i_1}=c} =$ (поскольку последняя функция не зависит от x_{i_2})$= f|_{x_{i_1}=c, x_{i_2}=\overline{d}} = f_{x_{i_1}=\overline{c}, x_{i_2}=d}$ (последнее равенство доказывается по аналогии). Т. е. на трех наборах подфункции совпадают.

φ мы определим так:

$\varphi(x_{i_1}, x_{i_2}) = 0 \leftrightarrow x_{i_1} = \overline{c} \,\&\, x_{i_2} = \overline{d}$.

Видно, что это нелинейная функция двух переменных, а такие функции, как легко проверить, принадлежат множеству Φ.

Как определим f_1?

$$f_1(y, x_{i_3}, \ldots, x_{i_n}) = \begin{cases} f|_{x_{i_1}=c, x_{i_2}=d}, & \text{если } y = 1 \\ f|_{x_{i_1}=\overline{c}, x_{i_2}=\overline{d}}, & \text{если } y = 0 \end{cases}$$

Отсюда уже видно, что $f \equiv f_1(\varphi, \ldots)$.

Почему $f_1 \in \Phi$?

Поскольку φ является нелинейной функцией, из нее можно получить подстановкой некоторой константы вместо одной переменной y или $\overline{y}$. Тогда заметим, что $f_1(y, \ldots)$ есть функция, обобщенно-однотипная с некоторой подфункцией f, а любая подфункция функции, принадлежащей Φ, также принадлежит Φ.

Утверждение доказано.

Закончим доказательство леммы.

Итак, мы выделили группу переменных, теперь применим утверждение к функции f_1, у нее также выделим группу переменных и т. д. Получаем бесповторную формулу над базисом B_0. Это можно увидеть, если вспомнить, что $\varphi(x, y) = x^{\sigma} \circ y^{\tau}$, $\circ \in \{\&, \vee\}$.

Если $n = 2$, то непосредственно доказывается, что f — нелинейна, и для нее лемма также справедлива.

Лемма доказана.

Эта лемма представляет и самостоятельный интерес, поскольку дает более естественное описание функций, бесповторно выразимых в базисе B_0. В этом направлении относительно других базисов тоже что-то делалось и делается до сих пор, например, некий критерий вырабатывается в Иркутске Перязевым и др. По сути, здесь нужно проверять наличие отмеченных переменных, т. е. комбинаторное свойство самой функции, в отличие от бесповторной выразимости, где надо проверять, существует или нет требуемая форма, т. е. осуществлять некоторый перебор. Здесь же рассматривается внутренне свойство функции.

Теперь мы приступаем к получению нижней оценки сложности.

Как мы будем доказывать сегодняшнюю основную теорему?

По ранее доказанному следует, что если все функции из B бесповторно выразимы в B_0, то $B \sim B_0$.

Теперь предположим, что какая-то функция f из B бесповторно невыразима в B_0. И нам надо показать, что $B \prec B_0$.

Известно, что $B \preceq B_0$. А в обратную сторону, $B_0 \neq\preceq B$, не будет выполнено. Т. е. мы покажем, что некая последовательность функций имеет в B_0 высокую и в B низкую, (по порядку ниже), сложности.

Эту последовательность можно построить на основе той функции, которая бесповторно невыразима в B_0. В B эта функция и построенная на ее основе последовательность бесповторно выразимы, а следовательно, имеют в B линейную сложность. Мы же покажем, что в B_0 последовательность имеет нелинейную сложность.

Давайте определим эту последовательность. Впервые эта последовательность была рассмотрена в работе Субботовской, а затем для многих базисов первоначальное доказательство было мною существенно расширено, но это продвижение было сделано в том же направлении.

Вспомним предыдущую лемму.

Что значит, что f бесповторно невыразима в B_0? Это значит, что $f \neq\in \Phi$. Тогда $\exists$ подфункция ψ функции f такая, что ψ имеет отмеченную переменную.

Пусть $\psi(x_1, \ldots, x_n)$, x_1 — отмеченная переменная.

Исходя из этой функции ψ, мы построим нужную нам последовательность.

Положим $\psi_1 = \psi$. Это базис индукции.

Индуктивный переход.

$\psi_i = \psi_{i-1}(\psi(x'_1, \ldots, x'_n), \ldots, \psi(x_1^{n^{i-1}}, \ldots, x_n^{n^{i-1}}))$. Всего переменных у ψ n^i.

Функцию ψ_i можно реализовать с помощью такого равномерного дерева глубины i.

Рис.

Функция ψ_i бесповторно выразима в базисе B, поскольку ψ является подфункцией какой-то функции из базиса B, константы бесповторно выразимы в любом базисе и ψ_i является бесповторной композицией бесповторно выразимых функций.

Почему эта последовательность будет иметь нелинейную сложность? ψ имеет отмеченную переменную, следовательно, она не забивает никакую переменную и, подставляя вместо отмеченной переменной константу, остальные переменные остаются в функции существенными, а в формуле B_0 эта переменная какую-то переменную забьет. На этом эффекте и основана нижняя оценка. Единственное, что можно сказать, если у Субботовской доказательство занимает 4 страницы, то в расширенном варианте у меня — 40 страниц. Но если базис содержит линейную функцию, забивание не произойдет, нужно смотреть на нелинейные подформулы, но об этом разговор будет в следующем семестре.

Итак, введем некоторые понятия, связанные с функцией ψ_i.

Часть переменных функции ψ_i, обозначаемая $(x_1^j, \ldots, x_n^j)$, называется поднабором.

Специальным набором $(x_{j_1}^1, x_{j_2}^2, \ldots, x_{j_{n^{i-1}}}^{n^{i-1}})$ называется набор, содержащий по одной переменной из каждого поднабора.

Теперь наша задача: разбить все множество переменных на n специальных наборов $r_1, \ldots, r_n$.

По сути, мы выберем директивно лишь r_1 набор, а остальные можно выбрать произвольно, как получится, это всегда можно сделать.

Далее, построим три множества переменных A, B, C.

Что такое A? Мы будем выбирать какие-то отмеченные переменные из разных поднаборов, т. е. в каждом поднаборе есть первая переменная, являющаяся отмеченной x_1^j. Мы будем некоторые такие отмеченные переменные выбирать и из них строить множество A.

Параллельно будем строить B. Т. е. выбранные отмеченные переменные будут забивать какие-то другие переменные, их мы и будем относить к B.

Пусть у нашей функции ψ_i есть какая-то минимальная формула F в B_0. Тогда $x_1^{\prime} \in A$, а $x_{i_1}^{j_1} \in B$, ($x_1^{\prime}$ забивает $x_{i_1}^{j_1}$, т. е. $F_1 = F|_{x_1^{\prime}=c_1}$).

Продолжим далее. У нас есть формула F_1. Возьмем еще какую-нибудь отмеченную переменную, которая не попала ни в A, ни в B. Обозначим ее $x_1^{l_2}$ и отнесем ее к A. В F_1 эта переменная забивает еще одну переменную $X_{i_2}^{j_2}$, (которая, может и совпадать с $x_{i_1}^{j_2}$) и отнесем ее к B. Получим $F_2 = F_1|_{x_1^{l_2}=c_2}$. И т. д. Отметим еще, что поскольку все переменные, которые мы относим к A отмеченные, оставшиеся переменные являются существенными.

Как долго может продолжаться процесс построения A?

Теоретически, переменная, например, из первого поднабора забивает переменную из второго поднабора. Из третьего — из четвертого. В худшем случае, в A можем положить не менее половины отмеченных переменных.

Введем параметр $\nu = \lfloor\frac{n^{i-1}}{2}\rfloor$. n^{i-1} — число поднаборов, в каждом есть одна отмеченная переменная. $|A| = \nu$; когда сделаем ν шагов, мы остановимся.

$|B| \leqslant \nu$ в любом случае.

Построенное множество A включим в набор r_1. А затем дополним его с помощью третьего множества C. К C отнесем ту переменную из каждого оставшегося поднабора, которая входит в B минимальное число раз. (Напомним, что в B могут входить одинаковые переменные.) Тогда $A \vee C = r_1$.

Оценим, сколько раз переменные из C могут входить в набор B?

Всего в наборе B имеется ν вхождений, мы каждый раз брали переменную с наименьшим вхождением, следовательно, $|B \cup C| \leqslant \frac{\nu}{n}$.

Сколько имеется в B вхождений переменных, не содержащихся в C (а также в A)? Соответственно, не менее $\nu - \frac{\nu}{n}$ переменных. Они попадают в другие специальные наборы: $r_2, \ldots, r_n$.

Мы расмотрели всю последовательность подстановок $F, F_1, \ldots, F_\nu$. Все переменные из $r_2, \ldots, r_n$ остались существенными. Возьмем, например, набор r_2. Можно подставить такие константы вместо остальных переменных, чтобы из каждого поднабора осталась ровно та переменная, которая принадлежит r_2. Тогда получим функцию, обобщенно-однотипную с ψ_{i-1}. Тем самым, мы осуществим индуктивный переход. То же самое можно сказать относительно любого набора r_i. Запишем это.

$\forall r_i$ $(i \in \{2, \ldots, n\})$ существует такая подстановка констант A_i вместо всех переменных $\neq\in r_i$, что $\psi_i|_{A_i}$ — обобщенно-однотипна с ψ_{i-1}.

Одну из этих подстановок мы будем использовать, но какую именно предстоит выяснить. Мы подсчитаем среднюю сложность, которая при этом получится, и возьмем ту, которая будет оптимальной.

Вспомним, когда мы подставляли константы c_i, мы получали функции $\psi_i^1, \psi_1^2, \ldots, \psi_1^\nu$. И затем из последней функции еще какой-то подстановкой оставляем по одной переменной из каждого поднабора.

Поэтому обозначим нашу функцию, получившуюся при подстановке A_i, через ψ_i^ν.

Начнем проводить выкладки.

Всего в наборах $r_2, \ldots, r_n$ не менее $\nu - \frac{\nu}{n}$ переменных из B. Нам нужно выбрать тот набор, где пересечение с B будет наибольшим. Почему ?

Что такое, по существу, B? Это те переменные, которые хотя бы по одному вхождению удаляются, но сами остаются существенными. Они как-то распределены по наборам $r_2, \ldots, r_n$. Нам надо выбрать тот специальный набор, куда попадет наибольшее число этих переменных, значит, сложность оставшейся формулы будет как можно меньшей.

При получении A у нас была какая-то подстановка констант, при этом переменные из B теряли свои вхождения, сами оставаясь существенными. Пусть мы выбрали тот r_i, который с B имеет наибольшее пересечение. При подстановке A_i получили подфункцию ψ_{i-1}.

$\frac{\nu - \frac{\nu}{n}}{n-1} \leqslant |r_i \cup B|$.

Какова будет сложность полученной формулы $F' \equiv \psi_{i-1}$?

Рассмотрим такую формулу $F^s = F_\nu|_{A_s}$ $(s = 2, \ldots, n)$. И введем величину $q_s = |r_s \cup B|$.

Мы знаем, что $q_2 + \ldots + q_n \geqslant \nu - \frac{\nu}{n} = (1 - \frac{1}{n})\nu \geqslant (1 - \frac{1}{n})(\frac{n^{i-1}}{2} - 1)$(здесь используется определение ν)$\geqslant n^{i-2}$ при $i \geqslant 3$.

Далее, обозначим $L_s(F)$ — число вхождений в F переменных из r_s. При подстановке констант вместо переменных из B по крайней мере по одному вхождению переменной удалилось, поэтому

$L_s(F) \geqslant L(F^s) + q_s \geqslant L_{B_0}(\psi_{i-1}) + q_s$ (последнее верно в силу того, что F^s — обобщенно-однотипная с ψ_{i-1}), $s = 2, \ldots, n$.

Просуммируем эти неравенства:

$L_{B_0}(\psi_i) = L(F) \geqslant \sum\limits_{s=1}^{n} L_s(F) \geqslant nL_{B_0}(\psi_{i-1}) + \sum\limits_{s=2}^{n} q_s$(заметим, что из r_1 можно также получить функцию, обобщенно-однотипную с ψ_{i-1}, поэтому $L_1(F) \geqslant L_{B_0}(\psi_{i-1})) \geqslant nL_{B_0}(\psi_{i-1}) + n^{i-2}$.

Мы получили рекуррентное неравенство. Обозначив $\alpha_i = L_{B_0}(\psi_i)$, продолжим его:

$\alpha_i \geqslant n\alpha_{i-1} + n^{i-2} \geqslant n(n\alpha_{i-2} + n^{i-3}) + n^{i-2} = n^2\alpha_{i-2} + 2n^{i-2} \geqslant (i-3)n^{i-2} \geqslant\geqslant (i-3)n^{i-2}$.

Это и есть нелинейная нижняя оценка, поскольку у функции ψ_i имеется $N = n^i$ переменных, n — константа, тогда $\alpha_i \succeq N \log N$.

Лекция 8 (28.02.07)

Мы продолжаем третью главу нашего спецкурса, посвященную более изощренной технике забивания. Предыдущая лекция была основана на работе Субботовской 1963 г., связанной с критерием эквивалентности произвольного базиса базису B_0, где уже были применены менее очевидные приемы забивания в отличие от обобщенно-монотонных базисов и базиса с медианой.

С этой лекции мы будем рассматривать базисы, содержащие линейную функцию, и для них использовать более сложную технику забивания. Это уже будут мои результаты, полученные начиная с 1997 г. Сегодня мы докажем частный случай моего результата 1997 г. о том, что существует бесконечно убывающая цепочка базисов $P_2(2) \succ P_2(3) \succ P_(4) \succ \ldots$, где $P_2(n)$ — все функции двузначной логики , зависящие от n переменных. Из результатов Перязева также извлекается бесконечная цепочка, но другая, не самого общего вида, как здесь.

Сегодня мы начнем доказывать первый шаг: $P_2(2) \succ P_2(3)$. Остальные шаги рассматриваются аналогично. Сейчас мы докажем вспомогательные леммы и затем сформулируем общую идею доказательства.

Итак, нам нужно предъявить $f_1, f_2 \ldots$ такую, что

$L_{P_2(3)}(f_n) \asymp nes(f_n)$, (где $nes(f_n)$ — число существенных переменных), а

$\frac{L_{P_2(2)}(f_n)}{nes(f_n)} \to \infty$.

Для упрощения доказательства будем рассматривать базис $\{\oplus, \&, 1\}$, эквивалентный $P_2(2)$. Этим базисом удобно пользоваться, с алгебраической точки зрения это естественный базис: функции сложения и умножения над полем Z_2. И мы просто имеем дело с многочленами, где выполнена какая-то группировка переменных, и еще взята константа 1.

Какую мы выберем последовательность функций? Берется некая базовая функция и далее конструируется последовательность функций, как у Субботовской в 7 лекции, т. е. берутся степени этой базовой функции.

Базовой функцией у нас будет $f(x, y, z) = xyz \oplus x \oplus y \oplus z$. Эта функция трех аргументов, следовательно, она вложена в базис $P_2(3)$ и, значит, реализуется бесповторно в этом базисе. Также и ее степени будут реализоваться бесповторно, а значит, ее сложность будет равна числу ее аргументов.

Напомним, что такое степень произвольной функции.

Пусть $g(x_1, \ldots, x_n)$ — произвольная функция.

Тогда степени определяются следующим образом:

$g^{(1)} = g$

$\vdots$

$g^{(m)} = g(g^{(m-1)}(\tilde{x}^1), \ldots, g^{(m-1)}(\tilde{x}^n))$, где $\tilde{x}^i$ — попарно различные наборы и в каждом различные переменные.

То есть мы видим, что степени базовой функции бесповторно выразимы и, тем самым, первая часть утверждения очевидна.

Переходим к доказательству нижней оценки. Почему в качестве базовой функции мы выбрали именно функцию f?

Степень ее полинома Жегалкина максимальна, а именно, равна числу переменных. И второе, она почти совпадает с линейной функцией, отличаясь от нее только в одной точке. Чем хороша линейная функция? При подстановке констант она остается существенно зависящей от оставшихся переменных. Поэтому наша базовая функция для многих подстановок будет также оставаться зависящей от оставшихся переменных. Выпишем явно эти подстановки. Во-первых, подстановки от произвольного числа нулей:

$f(0, y, z) = y \oplus z,$

$f(0, 0, z) = z.$

И подстановки с одной единицей:

$f(1, y, z) = y \vee z$ и

подстановки с одной единицей и одним нулем:

$f(1, 0, z) = z$.

С двумя единицами подстановки уже не подойдут:

$f(1, 1, z) = 0$.

Это же распространяется и на степень исходной функции и даже на любую функцию, бесповторно выразимую в базисе, состоящем только из f. Сформулируем это в виде леммы.

Пусть $\mathcal{F}$ — все функции, строго бесповторно (т. е. нет несущественных переменных и все переменные входят в формулу по одному разу) выразимые в базисе $\{f\}$.

Как можно представить это множество $\mathcal{F}$?

Возьмем любое корневое дерево, у которого исходящая степень каждой вершины равна 3. И в каждой вершине находится одна и та же функция f, а листьям соответствуют разные переменные.

Рис. 1

Частным случаем является степень функции, когда дерево равномерное, т. е. полностью заполнено до какого-то яруса:

Рис. 2

$f^{(2)}$ Например, двухярусное дерево соответствует второй степени.

Таким образом, мы можем сформулировать общее свойство функций из множества $\mathcal{F}$ — устойчивость относительно подстановок нулей вместо каких-то переменных и, возможно, одной единицы.

Утверждение 1.

Пусть $g \in \mathcal{F}, A$ — подстановка констант вместо переменных, причем среди констант не более одной единицы. Тогда $g|_A$ существенно зависит от всех оставшихся переменных.

Мы не будем строго по индукции доказывать это утверждение , а попробуем объяснить на пальцах. Мы можем подставлять константы по очереди и на каждом шаге видеть, что переменные будут оставаться существенными. Вначале подставляем нули. Подставим один нуль. Наша функция превратилась в сумму по модулю два от оставшихся переменных. Если подставили еще один нуль, то останется просто одна переменная и всю функцию f, в которую подставлены два нуля, можно убрать и заменить одной листовой вершиной, а именно, оставшейся переменной. А если 3 нуля подставить? Уже понятно, что делать. Из f при данной подстановке могут получаться или переменные, или линейная функция, т. е. мы приходим на круги своя, к исходному виду. Теперь подстановка — одна единица. Если она просто в f подставлена, то получится существенная функция дизъюнкция, а если она подставлена в сумму по модулю два, то — отрицание переменной (эта подстановка одной единицы может произойти только на последнем шаге).

Итак, мы разобрали первое свойство устойчивости базовой функции и ее производных.

Теперь второе свойство — степень полинома Жегалкина.

Утверждение 2.

Пусть у функций $g, g_1, \ldots, g_n$ степень полинома Жегалкина максимальна, т. е. равна числу переменных. Тогда у $g(g_1(\tilde{x}^1), \ldots, g_n(\tilde{x}^n))$ степень полинома Жегалкина максимальна.

Как это можно объяснить? Просто мы получаем максимальный моном и он ни с чем не сокращается. Почему мы его получаем? Давайте выпишем старшие члены у g и остальных функций:

$g = \hat{g} + \ldots$, где $\hat{g} = x_1 \cdot \ldots \cdot x_n$

$g_i = \hat{g}_i + \ldots$.

Представим, что все функции — многочлены и мы подставляем в многочлен g другие какие-то многочлены вместо переменных:

$g(g_1, \ldots, g_n) = g_1 \cdot \ldots \cdot g_n + \ldots = \hat{g}_1 \cdot \ldots \cdot \hat{g}_n + \ldots$.

А это и есть произведение всевозможных переменных. Т. е. максимальный член имеется, но почему он ни с чем не сокращается в дальнейшем? Потому что в $\hat{g}_1 \ldots \hat{g}_n$ все остальные члены имеют меньшее число переменных хотя бы в одном сомножителе, значит, и в сумме меньшее. В "хвосте" также присутствует меньшее количество функций, не все переменные в произведении будут иметься. Вообще, это алгебраическая вещь, универсальная. Если подумать немного, то вполне это можно понять.

Следствие из утв. 2.

$\forall g \in \mathcal{F}$ степень полинома Жегалкина функции g максимальна.

Как это доказывается?

Любая функция из $\mathcal{F}$ получается следующим образом:

$f(f_1, f_2, f_3)$, $f_i \in \mathcal{F}$. Далее следует непосредственно индукция, для f_i мы доказали по предположению индукции, для f следует из определения и применяем утв. 2.

Зачем нам нужно, чтобы степень была максимальной? Для того, чтобы в любой формуле, реализующей эту функцию, было много умножений.

Утверждение 3.

Пусть G — формула в базисе $\{\oplus, \&, 1\}$, реализующая функцию g, и $d(g)$ — степень полинома Жегалкина функции g. Тогда в G найдется не менее $d(g) - 1$ функций $\&$.

Доказательство по индукции по построению формулы G , поскольку это уже не алгебра в чистом виде.

Базис индукции: $d(g) = 1$, $G = x_i$, ни одного элемента & нет. Очевидно. А также не забыть случай $G = 1$.

Индуктивный переход.

a) $G = G_1 \cdot G_2$

b) $G = G_1 \oplus G_2$

Пусть g_1 и g_2 — функции, которые реализуются формулами G_1 и G_2.

Начнем с пункта b).

Обозначим число умножений в формуле G за $L_\$(G)$.

Тогда $d(g_1 \oplus g_2) \leqslant \max(d(g_1), d(g_2)) \leqslant$ (по предположению индукции)$\leqslant \max(L_\&(G_1) + 1, L_\&(G_2) + 1) = \max(L_\&(G_1), L(G_2)) + 1 \leqslant L_\&(G_1) + L_\&(G_2) + 1 = L_\&(G) + 1$.

a) $G = G_1 \cdot G_2$

$d(g) = d(g_1 g_2) \leqslant d(g_1) + d(g_2) \leqslant$ (снова предположение индукции)$\leqslant L_\&(G_1) + 1 + L_\&(G_2) + 1 = L_\&(G) + 1$ (одна из единиц соответствует внешнему умножению в формуле).

Как будет устроено доказательство основного результата?

Начнем с самой функции $f(x, y, z)$. Сейчас мы покажем, что $L(f) \geqslant 4$, т. е. хотя бы одна переменная входит два раза. (На самом деле, $L(f) = 5$, но это нам не нужно.)

Напомним вид функции f.

$f = xyz \oplus x \oplus y \oplus z = x(yz \oplus 1) \oplus y \oplus z$. Можно показать, что последняя формула для f минимальна. Нам достаточно показать, что сложность не менее 4.

Мы будем использовать то, что умножений в формуле не менее 2, поскольку степень полинома Жегалкина 3.

Допустим, что все переменные входят по одному разу: x, y, z. Между переменными помещается две операции. Как мы выяснили — обе они умножение. Вернее, где-то могут быть единицы. Например, $(x \oplus 1)(y(z \oplus 1) \oplus 1)$. Посмотрим на самое внешнее умножение формулы общего вида : $(x[\oplus 1] \cdot (\ldots)[\oplus 1]$. Возьмем какую-нибудь константу вместо переменной x, чтобы первая скобка уничтожилась. Если единица в ней есть, возьмем константу 1, если — нет, то возьмем 0. В любом случае, остальные переменные исчезают. Но это не так, поскольку ранее мы показали, что надо хотя бы две единицы подставить, чтобы получить 0. Следовательно, одна переменная входит хотя бы два раза и сложность формулы для f не менее 4.

Из этого факта следует, что для последовательности степеней оценка сложности асимптотически не менее $2n$. Вернее, мы будем работать с параметром максимальное вхождение одной переменной.

Покажем это. Эта процедура напоминает процедуру Рамсея, т. е. мы выбираем нужную нам подформулу. Рассмотрим сначала $f^{(2)}$.

Рис.

Здесь все 9 переменных сгруппированы в 3 группы по 3 переменных. Допустим, в какой-то группе все три переменные входят по одному разу. Может ли такое быть? Нет, не может. Подставим какие-то константы в остальные переменные, так чтобы была существенная зависимость от этой группы. Такая возможность, если есть существенная зависимость, всегда есть. Т. е. получится либо f, либо $\overline{f}$, и в ней сохранятся все вхождения переменных по одному разу, что невозможно. Значит, в каждой группе не более, чем две переменные могут входить по одному разу.

Допустим, в первой группе 2 переменные входят по 1 разу, во второй — по 1 разу, а в третьей могут 2 переменные входить по 1 разу? Нет, не могут. Рассмотрим такую ситуацию: в каждой группе найдется переменная, входящая 1 раз. Уже такого быть не может. Почему? Опять же вместо оставшихся переменных подставим константы так, чтобы переменные, входящие по предположению 1 раз, сохранились. Получим снова функцию $f(x^{\sigma_1}, y^{\sigma_2}, z^{\sigma_3})$. Это невозможно, поскольку сложность не меньше 4. Поэтому ясно, что в одной из групп не должно быть ни одной переменной, входящей 1 раз. Следовательно, максимум 4 переменные входят один раз в формулу. Если мы рассмотрим $f^{(3)} = f(f^{(2)})$, то в каждой большой группе из 9 переменных не могут все переменные входить по 1 разу, а в каждой группе не более 4 переменных входят 1 раз, как следует из предыдущего шага. Итого, не более 8 переменных, а всего их 27. Уже видно, что остается все меньшая и меньшая доля переменных, входящих 1 раз. Точнее, всего 3^m переменных, а входящих 1 раз не более, чем $2^m = o(3^m)$.

Сформулируем это же в некоем общем виде.

Утверждение 4. Пусть $g(x_1 \ldots x_n)$ некоторая функция и в любой формуле для g над некоторым базисом B найдется переменная, входящая не менее k раз. Далее, пусть g' — функция, строго бесповторно выразимая над базисом $\{g\}$, причем дерево (не обязательно равномерное), соответствующее функции g', имеет глубину (максимальное количество ярусов) m.

Рис.

Пример 3-хярусного дерева.

Тогда в любой формуле, реализующей функцию g' над B, число переменных, входящих не более k раз, не больше, чем $(n-1)^m$.

Как это соответствует предыдущему частному случаю?

Здесь в качестве g выступала функция f, $n = 3$, $m = 2, 3$, $k = 2$, в качестве G' — равномерное дерево. Тогда $(n-1)^m = 2^m = o(3^m)$.

Докажем по индукции.

Базис индукции. (Индукция по построению формулы g').

$g' = g$, $m = 1$. Утверждается, что в любой формуле, реализующей g, число переменных, входящих не более k раз, не больше $n-1$, т. е. существует переменная, входящая более k раз, что является переформулировкой условия.

И еще случай $g' = x_i$, $m = 0$.

Индуктивный переход.

Пусть g' представляется в виде $g' = g(g'_1, \ldots, g'_m)$, где к функциям g'_i можем применить предположение индукции. У g'_i глубина не более $m-1$. Рассмотрим какую-нибудь формулу над базисом B , о которой идет речь в утверждении, и посмотрим, как расположены те переменные, которые входят меньше k раз. Утверждается, что среди переменных хотя бы одной из функций g'_i таких переменных нет.

Почему? Предположим, что у всех g'_i есть переменная, входящая меньше k раз. Тогда вместо оставшихся переменных подставим какие-то константы, чтобы эти переменные остались. И получим функцию $g(x_1^{\sigma_1}, \dots, x_n^{\sigma_n})$. Тогда мы имеем формулу для g, в которой существует переменная, входящая менее k раз, что противоречит условию. Значит, найдется g'_i, не содержащая таких переменных.

Пусть, скажем, $i = n$. Последняя функция из g'_i не содержит таких переменных, остальные могут содержать. Но сколько? Опять же по предположению индукции $g'_j(j < n)$ содержат не более $(n-1)^{m-1}$ таких переменных. Итого, не больше $(n-1)(n-1)^{m-1} = (n-1)^m$ "плохих" переменных.

Утверждение сложное, но доказывается оно просто.

Каким же будет общий вид доказательства? Пока что мы получили оценку асимптотически не меньше $2n$.

Теперь мы выберем какую-то большую степень, число m. Наша задача: показать, что при каком-то достаточно большом m хотя бы одна переменная будет входить уже не меньше трех раз. Предположим, что все переменные входят по 2 раза. Тогда произведем серию забиваний и при каждом забивании какая-то переменная будет с двух вхождений переменной перескакивать на единицу. И после того, как этих единиц наберется 2^m, а это есть "о малое" от общего числа переменных, получим противоречие с утв.4. Значит, есть переменная, которая входит хотя бы 3 раза. Тогда в качестве g берем $f^{(m)}$ и применяем снова утв.4. Понятно, что степени g будут и степенями функции f. Тогда сразу получаем оценку асимптотически не меньше $3n$. Далее ищем переменную, входящую не меньше 4 раз. Это будет происходить при m_2. И т. д.

Лекция 9 (14.03.07)

Эта лекция является непосредственным продолжением предыдущей лекции, поэтому сразу же приступим к делу.

Мы рассматриваем формулы в базисе $\{\&, \oplus, 1\}$. Пусть G — формула в этом базисе. Среди подформул формулы G выделим максимальные линейные формулы, которые назовем центральными.

Пусть H — подформула формулы G. H назовем центральной, если внутри H содержатся только линейные функции, но любая формула, содержащая H, уже содержит умножение.

Т. е. центральную подформулу можно представить так:

$H = (x_{i_1} \oplus \ldots \oplus x_{i_k}[\oplus 1]) \cdot (\ldots)$.

Она обладает следующими свойствами:

1) содержит хотя бы одну переменную

2) линейна

3) максимально линейна.

Число таких центральных подформул связано с числом умножений.

В прошлый раз мы оценивали число умножений в формуле, выражали его через степень полинома Жегалкина и т. д. И вот, для забивания, нас интересуют такие центральные подформулы. Как будет забивание осуществляться? Мы будем подставлять какие-то константы вместо одной какой-то скобки и вторая скобка забьется. Поэтому нам важно знать число таких подформул.

Давайте обозначим $C(G)$ — число центральных подформул в G. И, у нас уже было обозначение, $L_\&(G)$ — чисдо умножений в G.

Утверждение 5.

Если G минимальна и по сложности L, и по числу единиц, (которые не входят в сложность), то $C(G) \geqslant L_\& + 1$.

(Условие минимальности вводится, чтобы исключить вырожденный случай, когда скобки состоят только из констант.)

Попробуем объяснить без индукции.

Во-первых, все "единички" из формулы уберем. Поскольку утверждение синтаксическое, что будет происходить с функцией, нам неинтересно. Останутся только операции $\&, \oplus$. Затем рассмотрим центральные подформулы и заменим каждую из этих подформул на введенную новую переменную (каждую для своей подформулы). Получим некоторую формулу.

Может получиться такой случай: $(A)(B) \oplus x_{i_1} \ldots$. Две скобки перемножаются и добавляется "хвост"переменных. Его мы отбрасываем также. Он не является центральной подформулой, он ни на что не умножается потом.

Получившаяся новая формула будет состоять из переменных, число которых равно числу центральных подформул. Сложность ее будет равна $C(G)$ и все умножения при этом сохранятся, их число $L_\&(G)$. Дальше уже легко сообразить, что операций & или $\oplus$ на единицу меньше, чем $C(G)$.

Сформулируем следствие из предыдущих утверждений и только что доказанного.

Ранее мы доказали следствие из утв.2: $forall g \in \mathcal{F} d(g) = n(g)$, утв.3: если G реализует g в B, то $L_\&(G) \geqslant d(g) - 1$.

Соберем эти утв. вместе.

Следствие из утв. 2,3,5

Если $g \in \mathcal{F}$, G реализует g в B, тогда $C(G) \geqslant L_\&(G) + 1 \geqslant d(g) = n(g)$.

Теперь сформулируем утверждение, объясняющее то, как использовать центральные формулы для забивания.

Утверждение 6.

Пусть G — минимальная формула в B, H — центральная подформула в G и $L_\&(G) \geqslant 1$. Тогда существует подстановка констант A, обладающая свойствами:

1) A определена только на переменных, входящих в H

2) среди подставляемых констант не больше 1 единицы

3) формула $G|_A$ упрощаема.

Определение упрощаемой формулы: если мы при подстановке констант вместо переменных, можем удалить еще какие-то вхождения других переменных, то такая формула называется упрощаемой.

Собственно все забивание, которое будет в окончательной теореме, сосредоточено здесь, в утв. 6. (Заметим, что ранее в "желтой книжке" и статье в МВК доказывалось сложнее, чем здесь. Будет рассмотрено всего несколько случаев.)

В G, по условию, есть хотя бы одно умножение, поэтому $H \cdot H'$.

Случай 1.

В H' есть переменная, не входящая в H. Это основной случай. Мы вместо всех переменных из H подставим какие-то константы и получим упрощаемую формулу. Выберем A так, чтобы $H = 0$. Все условия утв. 6 выполнены, поскольку H представляет из себя сумму переменных и, возможно, одной единицы. Отсюда следует, что единиц достаточно взять не больше одной.

Случай 2.

Все переменные из H' содержатся в H. Т. е. в этой формуле $H \cdot H'$ присутствуют только переменные, входящие в H. Поэтому мы можем в этой формуле выбирать любой фрагмент и делать там подстановку, и условие 1 будет выполнено.

Случай 2.1.

H' является центральной подформулой.

Случай 2.1.1.

В H есть переменная, не входящая в H'. Этот случай аналогичен случаю 1.

Случай 2.1.2.

Все переменные из H входят в H'. Ранее было условие: все переменные из H' входят в H. Т. е. мы получаем, что либо $X \cdot X$, либо $X(X \oplus 1)$. А эти формулы не являются минимальными.

Случай 2.2.

H' — не центральная подформула.

Тогда в H' есть подформула вида $H_1 \cdot H_1'$, где H_1 и H_1' — центральные. Почему это так? Возьмем самое первое умножение, которое производится при построении формулы H'. Если до этого умножений не было, а были только линейные операции, значит, умножение действует на две линейные скобки. Их обозначим через H_1 и H_1'. И здесь мы попадаем в ситуацию, аналогичную сл.1 и сл.2.1. Важно не упускать из виду, что в H' все переменные входят в H, поэтому никакие другие переменные подстановкой не будут затронуты.

Замечание. Это довольно неочевидный разбор случаев, пришел я к нему не сразу, а уже после опубликования статьи.

Утверждение 6 доказано.

Приступим к основной лемме. Она будет достаточно длинной. Из нее будет непосредственно следовать нижняя оценка.

Лемма.

$\forall k \exists m_k$: в любой формуле в B, реализующей $f^{(m_k)}$, найдется переменная, входящая больше k раз.

Базис индукции: $k = 1, m_k = 1$.

(было следствие из утв. 3, что сложность самой функции f не меньше 4, а у нее 3 переменные, значит, найдется переменная, входящая в формулу хотя бы 2 раза, т. е. больше, чем 1 раз).

Индуктивный переход.

$k - 2 \to k$, $k \geqslant 2$.

Обозначим через $g = f^{(m_{k-1})}$. И далее будем рассматривать степени функции g. Они же будут и степенями функции f. А именно, рассмотрим функции $g^{(m)} = f^{(m \cdot m_{k-1})}$.

Т. е. m_k будем искать в виде некоторого m, умноженного на m_{k-1}, а m будем выбирать из условий, которые в процессе доказательства выпишем.

Введем класс $\mathcal{F}_g$ — класс бесповторно выразимых функций над базисом $\{g\}$, $\mathcal{F} \subseteq \mathcal{F}$. Поэтому мы будем использовать утверждения, которые доказывали для $\mathcal{F}$.

Будем считать m фиксированной величиной, а потом выберем ее. Теперь для данного m предположим противное доказываемому.

$\exists G$ в B, реализующая $g^{(m)}$ такая, что $\forall$ переменная входит в G не более k раз.

Теперь уже по внутренней индукции произведем цепочку подстановок констант, в результате которой получим последовательность функций и формул. Т. е. построим функции

$g^{(m)} = g_0, g_1, \ldots, g_t$ и формулы

$G = G_0, G_1, \ldots, G_t$, причем

$g_i = g_{i-1}|_{A_i}$, $G_i = G_{i-1}|_{A_i}$.

Подстановки A_i выберем так, чтобы выполнялись следующие условия:

1) g_i — обобщенно-однотипна с некоторой функцией из $\mathcal{F}$

2) $|A_i| \leqslant 2kn$ $(n = n(g))$

3) в A_i не больше 1 единицы

4) формула $G_{i-1}|_{A_i}$ упрощаема

5) A_i определена только на неособых переменных.

Определим, что такое неособые переменные.

Общая идея такая: на каждом шаге будет производиться какое-то упрощение, т. е. хотя бы у одной переменной из остающихся будет на одно вхождение меньше. И нам надо эту переменную беречь, т. е. не допустить, чтобы на следующем шаге вместо нее была подставлена константа и этот эффект нивелировался. Поэтому мы такие переменные, которые входят не k раз, а меньше, не трогаем, считаем их особыми. Ну, и еще некоторые переменные.

g_i реализуется почти деревом над базисом g. В этом дереве в каждой нелистовой вершине сидит функция g или отрицание, в нем не больше m ярусов, поскольку g_i получена из функции $g^{(m)}$, исключая, быть может, некоторые переменные. Так вот, особыми также будем считать переменные, находящиеся на не последнем ярусе, а где-то ниже. Изначально у нас все переменные находились на последнем ярусе, а потом некоторые ушли внутрь, их тоже трогать не будем, будем считать особыми. Суть я объяснил. Теперь давайте изложим формально.

Введем множества P_i и Q_i.

Цитируемая литература

[1] Безруков А. Е. Дипломная работа. — М.: мехмат МГУ, 1998.

[2] Кириченко К. Д. Бесповторные и слабоповторные булевы функции в предэлементарных базисах // Канд. дисс.. — Иркутск, 2000.

[3] Кузнецов А. В. О бесповторных контактных схемах и бесповторных суперпозициях функций алгебры логики // Труды МИАН СССР. — 1958. — Т. 51. — С. 186–225.

[4] Мучник Б. А. Оценка сложности реализации линейной функции формулами в некоторых базисах // Кибернетика. — 1970. — № 4. — С. 29–38.

[5] Нечипорук Э. И. Об одной булевской функции // ДАН СССР. — 1966. — Т. 169, № 4. — С. 765–766.

[6] Перязев Н. А. Слабоповторные булевы функции в бинарном базисе // Иркутский ун-т. Серия: Дискретная математика и информатика. Вып. 4. — Иркутск, 1998.

[7] Рычков К. Л. О нижних оценках сложности параллельно-последовательных контактных схем, реализующих линейные булевы функции // Сибирский журнал исследования операций. — 1994. — Т. 1, № 4. — С. 33–52.

[8] Стеценко В. А. О сравнении булевых базисов // Изв. вузов. Математика. — 1988. — № 7. — С. 72–79.

[9] Стеценко В. А. О предплохих базисах в P_2 // Математические вопросы кибернетики. Вып. 4. — М.: Наука, 1992. — С. 139–177.

[10] Субботовская Б. А. О реализации линейных функций формулами в базисе $\vee$, $\&$, $-$ // ДАН СССР. — 1961. — Т. 136, № 3. — С. 784–787.

[11] Субботовская Б. А. О сравнении базисов при реализации функций алгебры логики формулами // Докл. АН СССР. — 1963. — Т. 149, № 4. — С. 784–787.

[12] Храпченко В. М. О сложности реализации линейной функции в классе П-схем // Матем. заметки. — 1971. — Т. 9, № 1. — С. 35–40.

[13] Храпченко В. М. Об одном методе получения нижних оценок сложности П-схем // Матем. заметки. — 1972. — Т. 10, № 1. — С. 83–92.

[14] Черухин Д. Ю. Об одной бесконечной последовательности улучшающихся булевых базисов // Дискретный анализ и исследование операций. — 1997. — Т. 4, № 4. — С. 79–95.

[15] Черухин Д. Ю. О предплохих булевых базисах // Дискретная математика. — 1999. — Т. 11, вып. 2. — С. 118–160.

[16] Черухин Д. Ю. Алгоритмический критерий сравнения булевых базисов // Математические вопросы кибернетики. Вып. 8. — М.: Наука, 1999. — С. 77–122.

[17] Черухин Д. Ю. О сложности реализации линейной функции формулами в конечных булевых базисах // Дискретная математика. — 2000. — Т. 12, вып. 1. — С. 135–144.

[18] Черухин Д. Ю. О реализации линейной функции формулами в различных базисах // Вестник Моск. ун-та. Сер 1. Математика. Механика. — 2001. — № 6. — С. 15–19.

[19] Черухин Д. Ю. Нижние оценки сложности булевых функций. — М.: График Без Границ, 2005.

[20] Шаранхаев И. К. Слабоповторные булевы функции в предэлементарных базисах // Канд. дисс.. — Иркутск, 2003.

[21] Шеннон К. Работы по теории информации и кибернетике. — М.: ИЛ, 1963.

[22] Яблонский С. В. Реализация линейной функции в классе П-схем // ДАН СССР. — 1954. — Т. 94, № 5. — С. 805–806.

[23] Яблонский С. В. Введение в дискретную математику. — 2-е изд. — М.: Наука, 1986.

[24] Chockler H., Zwick U. Which bases admit non-trivial shrinkage of formulae? // Computational complexity. — 2001. — V. 10, No 1. — C. 28–40.

[25] Hastad J. The shrinkage exponent of de Morgan formulas is 2 // SIAM J. Comput.. — 1998. — V. 27. — C. 48–64.

[26] Lee T. A new rank technique for formula size lower bounds // Proceedings of STACS-2007.