Теория дискретных функций

2022/23 уч. год, 2-й семестр

1-й курс, I поток

21 мая 2023 г.

Лектор — проф. В. В. Кочергин

`Примерное содержание лекций 7-11`

Сложность булевых функций

Вспомогательные факты из теории графов

Графом будем называть тройку (V, E, ρ), где $V = \{v_1, v_2, \dots, \}$ — конечное или счетное множество, называемое *вершинами графа*, $E = (e_1, e_2, \dots\}$ — конечное или счетное множество, называемое *рёбрами графа*, а отображение ρ каждому ребру e из множества E сопоставляет элемент из множества $V_2 \cup V^2 \cup V$, где V_2 — множество всех двухэлементных подмножеств множества V, а $V^2 = V \times V$ — множество всех упорядоченных пар элементов из V.

Если $\rho(e) \in V_2$, т. е. $\rho(e)$ — неупорядоченная пара $\{v_1, v_2\}$ некоторых вершин v_1 и v_2 из V, то ребро e называется *неориентированным ребром*, а вершины v_1 и v_2 — *концами* ребра e.

Если $\rho(e) \in V^2$, т. е. $\rho(e)$ — упорядоченная пара (v_1, v_2) некоторых вершин v_1 и v_2 из V, то ребро e называется *ориентированным ребром* или *дугой*, а вершины v_1 и v_2 — *началом и концом* ребра e, соответственно. Говорят, что дуга e выходит из вершины v_1 и входит в вершину v_2.

В обоих случаях говорят, что вершины v_1 и v_2 *инцидентны* ребру e.

Если $\rho(e) = v$ или $\rho(e) = (v, v)$ для некоторой вершины $v \in V$, то ребро e называется *петлей* или *ориентированной петлей*, соответственно.

Граф, в котором все ребра неориентированные, называется *неориентированным*. Граф, в котором все ребра ориентированные, называется *ориентированным*. В графе могут одновременнно присутствовать как ориентированные ребра, так и неориентированные. Обычно такие графы называют смешанными.

Ребра $e_1, \ldots, e_s$, удовлетворяющие условию $\rho(e_1) = \ldots = \rho(e_s)$, называются *кратными* или *параллельными*.

Граф, в котором нет кратных ребер и петель, называется *простым*. Неориентированный граф, в котором нет кратных ребер и петель, называется *обыкновенным*.

Обозначим через $\deg v$ число ребер, инцидентных вершине v (при этом петли считаются дважды). Вершина v называется *изолированной*, если $\deg v = 0$. Вершина v называется *концевой* или *висячей*, если $\deg v = 1$.

В ориентированном графе через $\deg_+ v$ и $\deg_- v$ обозначим число дуг, входящих в вершину v и выходящих из нее, соответственно.

Последовательность $v_{s_1}, e_{t_1}, v_{s_2}, e_{t_2}, \ldots, v_{s_k}, e_{t_k}, v_{s_{k+1}}$ называется *путем* от вершины v_{s_1} (начало пути) к вершине $v_{s_{k+1}}$ (конец пути) длины k, $k \geq 1$, если для любого i, $i = 1, \ldots, k$, либо $\rho(e_{t_i}) = \{v_{s_i}, v_{s_{i+1}}\}$, либо $\rho(e_{t_i}) = (v_{s_i}, v_{s_{i+1}})$.

Путь, в котором нет повторяющихся вершин, называется *цепью*.

Путь, в котором нет повторяющихся ребер и совпадает начало и конец, называется *циклом*.

Неориентированный граф, в котором любые две вершины соединены путем, называется *связным*.

Ориентированный граф называется *сильно связным*, если для любой пары вершин есть путь от первой вершины ко второй и от второй вершины к первой.

Граф, который получается из ориентироваенного графа путем замены ориентированных ребер (дуг) на неориентрованные (формально отображение ρ земеняется на отображение ρ' следующим образом: если $\rho(e) = (v_1, v_2)$, то $\rho'(e) = \{v_1, v_2\}$), называется *соотнесенным*.

Ориентированный граф называется *слабо связным*, если соотнесенный граф связен.

Неориентированный связный граф без циклов называется *деревом*.

Теорема 1. *Пусть G — конечный обыкновенный граф. Тогда следующие высказывания равносильны*:

1. *Граф G является деревом.*

2. *В графе G любые две вершины соединены единственной цепью.*

3. *Граф G связен и число ребер на единицу меньше числа вершин.*

4. *Граф G связен, но при удалении любого ребра перестает быть связным.*

5. *Граф G не содержит циклов, но при добавлении любого ребра образуется цикл.*

6. *Граф G не содержит циклов и число ребер на единицу меньше числа вершин.*

Упражнение 1. Доказать теорему 1.

Упражнение 2. Пусть $G = (V, E, \rho)$ — конечный обыкновенный связный граф. Тогда в графе G можно выделить подграф $G_0 = (V, E_0, \rho_0)$, $E_0 \subset E$, $\rho_0 = \rho\big|_E$, являющийся деревом.

Лемма 1. *В любом конечном ориентированном графе без ориентированных циклов найдется вершина, из которой не выходит ни одно ребро.*

Лемма 2. *В любом конечном ориентированном графе без ориентированных циклов можно так занумеровать вершины первыми идущими подряд натуральными числами, что любое ребро направлено от вершины с меньшим номером к вершине с большим.*

Нумерацию вершин в конечном ориентированном графе без ориентированных циклов из леммы 2 будем называть *монотонной* или *правильной*. Отметим, что правильная нумерация, вообще говоря, не единственна.

Определение СФЭ

Вычисление — это последовательность элементарных действий (из заданного набора) над исходными (входными) данными и величинами, полученными в процессе вычисления. При этом последовательность действий может быть фиксированна, а может зависеть от результатов промежуточных вычислений. В первом случае вычисление называется неветвящимся, а во втором — ветвящимся. Таким образом, неветвящиеся вычисления — это фиксированные последовательности элементарных вычислительных команд. Важнейшим модельным классом неветвящихся вычислений являются схемы из функциональных элементов (СФЭ).

Пример 1. Рассмотрим задачу вычисления значения x^{15}, где x — входная переменная, а элементарной операцией является умножение. Последовательность

$$\begin{aligned} & z_1 = x \times x = x^2; \\ & z_2 = z_1 \times z_1 = x^4; \\ & z_3 = z_2 \times z_2 = x^8; \\ & z_4 = z_3 \times z_2 = x^{12}; \\ & z_5 = z_4 \times z_1 = x^{14}; \\ * \quad & z_6 = z_5 \times x = x^{15}, \end{aligned}$$

вычислительных шагов в соответствии с двоичным разложением числа 15 не является минимальной по числу шагов при вычислении x^{15}. Действительно, следующая последовательностей элементарных операций

$$\begin{aligned} & z_1 = x \times x = x^2; \\ & z_2 = z_1 \times z_1 = x^4; \\ & z_3 = z_2 \times x = x^5; \\ & z_4 = z_3 \times z_3 = x^{10}; \\ * \quad & z_5 = z_4 \times z_3 = x^{15}, \end{aligned}$$

вычисляет x^{15} на один шаг быстрее.

В случае вычисления булевых функций СФЭ являются довольно точной математической моделью электронных логических схем без обратной связи.

Чтобы избежать излишней громоздкости и неоправданного формализма, дадим одно из нескольких эквивалентных определений для случая вычисления булевых функций в конкретном базисе[1] $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$. В общем случае определение дается аналогично.

Итак, пусть есть:

1) множество «исходных данных» X (как правило это переменные и, быть может, константы; в нашем случае $X = \{x_1, \ldots, x_n\}$);

2) множество «базисных операций» B (в нашем случае $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$).

[1]Строго говоря, фраза «базис $\{x \vee y,\ x\&y,\ \overline{x}\}$» математически безграмотна — множество булевых функций $\{x \vee y,\ x\&y,\ \overline{x}\}$ не является минимальной по включению полной системой, т. е. базисом. Однако в данном контексте такое словосочетание стало устойчивым и это связано с тем, что слово «базис» здесь несет другой смысловой оттенок — это набор элементарных средств, «кирпичиков», из которых строится СФЭ.

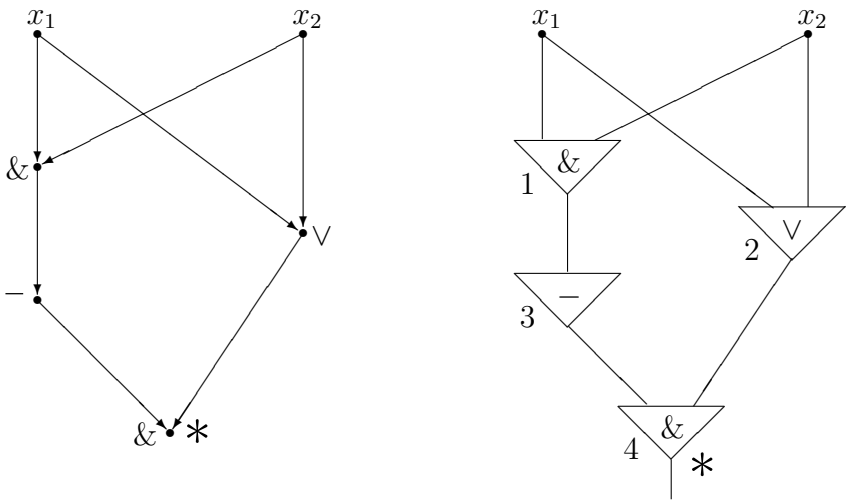


Рис. 1:

Схемой из функциональных элементов в базисе B_0 называется ориентированный граф (кратные ребра допускаются) без ориентированных циклов, в котором входные степени вершин могут быть равны только 0, 1 или 2, при этом если входная степень вершины равна 0, то вершине приписывается символ переменной из множества X (такие вершины называются *входами*), если входная степень вершины равна 1, то вершине приписывается функциональный символ, соответствующий операции отрицания, а если входная степень вершины равна 2, то вершине приписывается функциональный символ, соответствующий либо двухместной конъюнкции, либо двухместной дизъюнкции. Вершины с ненулевой входной степенью (т. е. вершины, которым приписаны символы операций), будем называть *функциональными элементами*. Кроме того, одна или несколько вершин помечены дополнительно «звездочкой» — эти вершины называются *выходами* (с них считывается информация).

На рис. 1 слева приведен пример СФЭ в базисе B_0 с двумя входами и одним выходом. Справа приведена та же схема, но в этой схеме функциональные элементы изображены в виде треугольников, внутри которых изображен приписанный данной вершине-элементу функциональный символ. Такой подход позволяет избавиться от ориентации ребер в СФЭ — ребра выходят из вершин треугольников и входят в основания. Кроме того, функциональные элементы на схеме справа специальным образом пронумерованы.

Чтобы корректно определить функционирование СФЭ, дадим одно определение и сформулируем лемму, касающуюся такой нумерации.

Нумерация функциональных элементов схемы, содержащей n невходовых вершин, числами $1, 2, \ldots, n$ называется *правильной* (или *монотонной*), если каждое ребро СФЭ либо исходит из входа, либо направлено от функционального элемента, имеющего меньший номер, к функциональному элементу, имеющему бóльший номер.

Правильная нумерация вершин ориентированного графа без ориентированных циклов, которая существует в силу леммы 2, естественным образом индуцирует правильную нумерацию функциональных элементов схемы, порождающей такой граф. Поэтому справедлива

Лемма 3. *У любой СФЭ существует правильная нумерация невходовых вершин* (*функциональных элементов*).

Правильная нумерация вершин СФЭ, вообще говоря, не единственна — в схеме справа на рис. 1 свойство правильности нумерации сохранится, если поменять номера у первого и второго функциональных элементов.

Теперь зафиксируем какую-либо правильную нумерацию вершин схемы. Далее в порядке увеличения номера естественным образом приписываем вершине вычисляемую функцию. Тем самым каждой вершине будет приписана своя функция. Будем говорить, что СФЭ *реализует* (*вычисляет*) булеву функцию $f(x_1, \ldots, x_n)$ (систему функций $f_1(x_1, \ldots, x_n), \ldots, f_m(x_1, \ldots, x_n)$), если выходу (выходам) приписана эта функция (эта система функций). Удобно считать, что выходы СФЭ пронумерованы (упорядочены), и тем самым СФЭ вычисляет *булеву* (n, m)*-функцию* — вектор (набор) из m булевых функций от n переменных.

Функцию, вычисляемую i-м функциональным элементом схемы, представленной на рис. 4, обозначим через h_i. Тогда

$$
\begin{aligned}
&h_1 = x_1 x_2;\\
&h_2 = x_1 \vee x_2;\\
&h_3 = \overline{h}_1 = \overline{x}_1 \vee \overline{x}_2;\\
&h_4 = h_3 h_2 = (\overline{x}_1 \vee \overline{x}_2)(x_1 \vee x_2) = x_1 \oplus x_2.
\end{aligned}
$$

Таким образом, результатом работы этой схемы будет линейная функция $x_1 \oplus x_2$.

В качестве еще одного примера СФЭ отметим, что при всем формальном различии в определениях СФЭ и формулы, любую формулу можно интерпретировать как СФЭ, в которой выходы функциональных элементов не ветвятся, т. е. полустепень исхода любой вершины, отличной от входов и выхода, равна единице.

Теперь перейдем к понятию схемы вычислений. Пусть, по-прежнему, есть множество «исходных данных» X и множество «базисных операций» B. *Схема вычислений (над X) в базисе B* — это последовательность равенств

$$\begin{gathered} z_1 = \varphi_1(y_{11}, \ldots, y_{1r_1}); \\ \ldots \quad \ldots \quad \ldots \\ z_i = \varphi_i(y_{i1}, \ldots, y_{ir_i}); \\ \ldots \quad \ldots \quad \ldots \\ z_l = \varphi_l(y_{l1}, \ldots, y_{lr_l}), \end{gathered}$$

где каждая переменная y_{ij} $(i = 1, \ldots, l;\ j = 1, \ldots, r_i)$ — это либо одна из входных независимых переменных из множества X, либо одна из внутренних переменных $z_1, \ldots, z_{i-1}$, вычисленных на предыдущих шагах; $\varphi_1, \ldots, \varphi_l \in B$. Кроме того, одна или несколько внутренних переменных из множества $z_1, \ldots, z_l$ дополнительно помечены «звездочкой» — эти переменные называются *выходными* (с них считывается информация).

Отметим, что иногда под схемой вычислений понимают просто последовательность

$$z_1, \ldots, z_l,$$

удовлетворяющую такому условию: для каждого i, $1 \leq i \leq l$, найдется такая функция $\varphi_i \in B$, что $z_i = \varphi_i(y_{i1}, \ldots, y_{ir_i})$, где $y_{ij} \in X \cup \{z_1, \ldots, z_{i-1}\}$, $j = 1, \ldots, r_i$.

Понятия СФЭ и схемы вычислений эквивалентны — если в схеме вычислений исходить из того, что любая функция φ_i из B реализуется функциональным элементом, то естественным образом приходим к понятию СФЭ. В дальнейшем, как правило, будем говорить просто «схема».

Сложностью схемы S называется число функциональных элементов (или, что то же самое, число равенств в последовательности) схемы S. Сложность схемы S будем обозначать через $L(S)$.

Если функция f (произвольной природы) реализуется какой-либо схемой S в базисе B, то, очевидно, найдутся еще схемы, реализующие функцию f в базисе B. Определим величину $L_B(f)$ — *сложность реализации функции f схемами в базисе B* — равенством

$$L_B(f) = \min L(S),$$

где минимум берется по всем схемам S, реализующим функцию f в базисе B.

Аналогично определяется *сложность* $L_B(\{f_1, \ldots, f_m\})$ *реализации системы функций* $\{f_1, \ldots, f_m\}$ *схемами в базисе* B:

$$L_B(\{f_1, \ldots, f_m\}) = \min L(S),$$

где минимум берется по всем схемам S, реализующим систему функций $\{f_1, \ldots, f_m\}$ в базисе B.

Минимальной схемой называется такая схема, что не существует другой схемы в том же базисе, реализующей ту же самую функцию или систему функций, и имеющую меньшую сложность. Таким образом, если схема S, реализующая функцию f в базисе B, минимальная, то выполняется равенство $L(S) = L_B(f)$.

К сожалению, задача нахождения точного значения сложности функции или системы функций (и построения минимальной схемы), как правило, очень трудна и может быть решена только в отдельных простых случаях. Это связано с проблемами в доказательстве оценок сложности снизу. Действительно, для получения верхней оценки сложности функции достаточно предъявить конкретную схему, вычисляющую эту функцию, а вот для получения нижней оценки нужно доказать, что никакая схема меньшей сложности не реализует эту функцию, а это, вообще говоря, переборная задача огромного размера. Поэтому часто исследуют следующую упрощенную задачу.

Пусть F — класс (множество) функций, а $F(n)$ $(n = 1, 2, \ldots)$ — некоторое подмножество этого класса, например, множество всех функций от n фиксированных переменных из класса F. *Функцией Шеннона для класса* F, или более точно — функцией Шеннона сложности реализации функций из класса F схемами в базисе B, будем называть функцию $L_B^F(n)$, определяемую равенством

$$L_B^F(n) = \max L_B(f),$$

где максимум берется по всем функциям f из множества $F(n)$.

Нас будет интересовать асимптотическое поведение функции Шеннона в первую очередь для классической задачи, когда $F = P_2$. В этом случае вместо обозначения $L_B^{P_2}(n)$ будем использовать более простое $L_B(n)$. Кроме того, иногда будем у функционалов сложности опускать информацию о базисе (нижний индекс), если эта информация однозначно определяется из контекста.

Сложность возведения в степень

Временно отвлечемся от основного объекта изучения в текущей теме — сложности реализации схемами из функциональных элементов булевых

функций и систем булевых функций и рассмотрим в общем виде задачу из примера 1 через призму введенных определений.

Итак, пусть множество «исходных данных» X состоит из одной переменной x, а множество «базисных операций» B состоит только из ассоциативной[2] операции умножения, применимой к степеням переменной x. Функциями, реализуемыми схемами в рамках такой модели, являются степени переменной x.

В соответствии с введенными определениями сложность $L(x^n)$ вычисления степени x^n численно равна минимально возможному числу операций умножения, достаточному для вычисления x^n.

Тривиальный способ последовательного умножения на x дает грубую оценку $L(x^n) \leq n - 1$.

Существенно более экономный способ вычисления x^n, основан на представлении числа n в двоичной записи и заключается[3] в $\lfloor \log n \rfloor$ последовательных возведениях в квадрат с последующим перемножением тех степеней, которые соответствуют единицам в двоичном разложении числа n. Пусть s — количество единиц в двоичном разложении числа n, тогда предложенная схема дает такую оценку сложности:

$$L(x^n) \leq \lfloor \log n \rfloor + s \leq 2\lfloor \log n \rfloor.$$

Однако, как показывает пример 1, и эта оценка может быть улучшена.

Чтобы получить представление о качестве имеющейся верхней оценки, установим нижнюю оценку величины $L(x^n)$, которая основана на следующем простом утверждении.

Лемма 4. *Пусть схема S вычисляет x^n. Тогда выполняется неравенство*

$$n \leq 2^{L(S)}.$$

Доказательство. Проведем индукцию по величине $L(S)$.

При $L(S) = 0$ схема вычисляет саму переменную x и неравенство $1 \leq 2^0$ выполняется.

Теперь в схеме S рассмотрим последний элемент. Этот элемент вычисляет x^n, перемножая какие-то степени x^a и x^b, вычисляемые в свою очередь какими-то схемами сложности не более $L(S) - 1$. Дважды применяя предположение индукции, получаем

$$n = a + b \leq 2^{L(S)-1} + 2^{L(S)-1} = 2^{L(S)}.$$

[2]Требование ассоциативности операции умножения нужно для корректного определения степени: действительно, если двухместную операцию "$*$" определить равенством $x * y = \overline{x}$, то, с одной стороны, $(x * x) * x = \overline{x} * x = x$, а с другой стороны, $x * (x * x) = x * \overline{x} = \overline{x}$.

[3]Здесь и далее, если не оговорено противное, под записью $\log x$ понимается $\log_2 x$.

□

Оформим в виде теоремы следующую нижнюю оценку, непосредственно вытекающую из леммы.

Теорема 2. *Для любого натурального n справедливо неравенство*

$$L(x^n) \geq \log n.$$

Таким образом, установлены верхняя и нижняя оценки величины $L(x^n)$, отличающиеся асимптотически вдвое.

Теперь установим асимптотически неулучшаемую верхнюю оценку.

Теорема 3. *При $n \to \infty$ справедливо соотношение*

$$L(x^n) \leq \log n + \frac{\log n}{\log\log n}\left(1 + O\left(\frac{\log\log\log n}{\log\log n}\right)\right).$$

Доказательство. Пусть d — натуральный параметр, значение которого выберем позже. Представим n в системе счисления по основанию 2^d:

$$n = a_0\left(2^d\right)^0 + a_1\left(2^d\right)^1 + \ldots + a_s\left(2^d\right)^s,$$

где $0 \leq a_i \leq 2^d - 1$, $i = 0, 1, \ldots s$; $a_s \neq 0$. Отметим справедливость неравенств

$$2^{sd} \leq n < 2^{(s+1)d}.$$

На первом этапе, использовав sd операций умножения, путем последовательного возведения в квадрат вычисляем степени

$$x^2,\ x^4,\ \ldots, x^{2^d},\ \ldots,\ x^{2^{2d}},\ \ldots,\ x^{2^{sd}}.$$

Положим

$$u_0 = x^{2^{0d}} = x,\ u_1 = x^{2^d},\ \ldots,\ u_s = x^{2^{sd}}.$$

Отметим, что все степени u_i, $i = 0, 1, \ldots, s$, вычислены на первом этапе.

Для $k = 1, \ldots, 2^d - 1$ положим

$$I_k = \{i \mid a_i = k\}, \quad J_k = \{j \mid a_j \geq k\}.$$

Справедливы такие представления вычисляемой степени:

$$x^n = u_0^{a_0}u_1^{a_1}\ldots u_s^{a_s} =$$

$$= \left(\prod_{i\in I_{2^d-1}} u_i\right)^{2^d-1}\left(\prod_{i\in I_{2^d-2}} u_i\right)^{2^d-2}\ldots\left(\prod_{i\in I_1} u_i\right)^1 =$$

$$= \left(\prod_{i\in J_{2^d-1}} u_i\right)\left(\prod_{i\in J_{2^d-2}} u_i\right)\ldots\left(\prod_{i\in J_1} u_i\right).$$

По уже вычисленным степеням $u_0, u_1, \ldots, u_s$ ввиду вложений

$$J_{2^d-1} \subseteq J_{2^d-2} \subseteq \ldots \subseteq J_1$$

произведения

$$\prod_{i \in J_{2^d-1}} u_i, \quad \prod_{i \in J_{2^d-2}} u_i, \ \ldots, \ \prod_{i \in J_1} u_i$$

можно последовательно вычислить, используя не более s операций умножения (по одной операции для «присоединения» каждой новой переменной u_i). Для перемножения этих произведений требуется не более $2^d - 2$ операций умножения.

Таким образом, окончательно имеем:

$$L(x^n) \leq sd + s + 2^d - 2 \leq \log n + \frac{\log n}{d} + 2^d.$$

Теперь, полагая

$$d = \lfloor \log\log n - 2\log\log\log n \rfloor,$$

из предыдущего неравенства при $n \to \infty$ получаем

$$L(x^n) \leq \log n + \frac{\log n}{\log\log n \left(1 - \frac{2\log\log\log n + 1}{\log\log n}\right)} + \frac{\log n}{(\log\log n)^2} =$$

$$= \log n + \frac{\log n}{\log\log n}\left(1 + \frac{2\log\log\log n}{\log\log n} + \frac{2}{\log\log n} + o\left(\frac{1}{\log\log n}\right)\right).$$

Требуемая верхняя оценка установлена. □

Следствие 1. *При $n \to \infty$ справедливо соотношение*[4]

$$L(x^n) \sim \log n.$$

Упражнение 3. Доказать, что

$$L(\{x^{n_1}, x^{n_2}\}) \sim \log\max(n_1, n_2)$$

при $n_1 + n_2 \to \infty$.

Упражнение 4. Пусть выполняются условия $n_i \leq \frac{m^2}{\log m}$, $i = 1, \ldots, m$. Доказать, что при $m \to \infty$ справедливо соотношение

$$L(\{x^{n_1}, \ldots, x^{n_m}\}) \sim m.$$

Упражнение 5. Доказать, что

$$\max_{k:\, k \leq n} L(x^k) - \log n \to \infty$$

при $n \to \infty$.

[4]Для положительных при всех достаточно больших значениях n последовательностей $\{a_n\}$ и $\{b_n\}$ запись $a_n \sim b_n$ при $n \to \infty$ означает, что $\lim\limits_{n\to\infty} \frac{a_n}{b_n} = 1$.

Предварительные оценки сложности булевых функций

Договоримся, что в случае, когда набором элементарных операций является классический базис $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$, индекс B_0 у функционалов сложности будем опускать.

Лемма 5. *Для любого натурального n выполняется неравенство*

$$L(n) \leq n2^n.$$

Доказательство. Константы 0 и 1 можно реализовать в соответствии с формулами $x_1\overline{x}_1$ и $x_1 \vee \overline{x}_1$ схемами сложности 2. Для любой функции $f(x_1, \ldots, x_n)$, отличной от констант, промоделируем схемой представление этой функции в виде совершенной дизъюнктивной нормальной формы:

$$f(x_1, \ldots, x_n) = \bigvee_{\tilde{\sigma}:\ f(\tilde{\sigma})=1} x_1^{\sigma_1} \ldots x_n^{\sigma_n}.$$

Для этого потребуется не более n отрицаний, чтобы реализовать отрицания переменных, по $n-1$ операций конъюнкции на каждую из не более чем $2^n - 1$ элементарных конъюнкций из совершенной дизъюнктивной нормальной формы, а затем не более $2^n - 2$ операций дизъюнкции для реализации функции f. Таким образом,

$$L(n) \leq n + (n-1)(2^n - 1) + 2^n - 2 < n2^n.$$

□

Таким образом, мы получили некоторую верхнюю оценку функции Шеннона. Но пока мы не имеем никакой нижней оценки. Позже установим асимптотически совпадающие экспоненциальные верхнюю и нижнюю оценки, а пока поставим цель намного скромнее: установим, что функция Шеннона стремится к бесконечности при неограниченном росте аргумента.

Лемма 6. *Для любого полного в P_2 конечного базиса B при $n \to \infty$ справедливо соотношение $L_B(n) \to \infty$.*

Доказательство. Действительно, пусть утверждение о том, что $L_B(n) \to \infty$ при $n \to \infty$ неверно. Тогда в силу монотонности функции $L_B(n)$ (очевидно, что $L_B(n+1) \geq L_B(n)$) она ограничена, т. е. для некоторого числа C при всех n выполняется неравенство $L_B(n) \leq C$. Следовательно, с одной стороны, число входов у схем ограничено, а с другой стороны, количество переменных, от которых функция может зависеть существенно, неограничено — противоречие. □

Теперь установим, что рост функции Шеннонане менее, чем линеен.

Теорема 4. *Пусть B — конечное множество булевых функций, удовлетворяющее условию $[B] = P_2$. Тогда для произвольной булевой функции f, существенно зависящей от n переменных, выполняется неравенство*

$$L_B(f) \geq \left\lceil \frac{n-1}{r(B)-1} \right\rceil,$$

где $r(B)$ — наибольшее число существенных переменных у функций базиса B (или максимальное число входов у функциональных элементов из базиса B).

Доказательство. Рассмотрим произвольную минимальную схему S в базисе B для функции f. Обозначим через $R(S)$ число ребер в схеме S. В силу существенной зависимости функции f от всех n переменных и минимальности схемы S из всех вершин схемы S, кроме выходной, выходит по крайней мере по одному ребру, т. е.

$$R(S) \geq n + L(S) - 1.$$

С другой стороны, в каждый функциональный элемент входит не более $r(B)$ ребер, поэтому

$$R(S) \leq r(B)L(S).$$

Объединяя две оценки, получаем неравенство

$$n + L(S) - 1 \leq r(B)L(S).$$

Следовательно,

$$L_B(f) = L(S) \geq \frac{n-1}{r(B)-1}.$$

Из этого неравенства и целочисленности величины $L_B(f)$ следует нужная оценка. □

Оценку из теоремы 4 в отдельных случаях можно усиливать, но для конструктивно заданных[5] последовательностей булевых функций известны лишь не более чем линейные по числу переменных нижние оценки сложности.

[5]Под конструктивным заданием последовательности булевых функций можно понимать, например, такое задание, при котором ответ на вопрос, равно ли значение функции на полученном наборе единице, может быть получен за полиномиальное время.

Точная по порядку верхняя оценка сложности булевых функций

Обозначим через $\mathcal{K}_n(x_1, \ldots, x_n)$ множество всех 2^n элементарных конъюнкций от переменных $x_1, \ldots, x_n$:

$$\mathcal{K}_n(x_1, \ldots, x_n) = \{x_1^{\sigma_1} \ldots x_n^{\sigma_n} \mid (\sigma_1, \ldots, \sigma_n) \in E^n\}.$$

Лемма 7. *При $n \to \infty$ справедливо асимптотическое соотношение*

$$L(\mathcal{K}_n) \sim 2^n.$$

Доказательство. Построим схему, реализующую систему функций $\mathcal{K}_n(x_1, \ldots, x_n)$, как показано на рис. 2. Схема состоит из трех блоков (подсхем). Первая подсхема по переменным $x_1, \ldots, x_{\lfloor n/2 \rfloor}$ реализует систему конъюнкций $\mathcal{K}_{\lfloor n/2 \rfloor}(x_1, \ldots, x_{\lfloor n/2 \rfloor})$, вторая подсхема по переменным $x_{\lfloor n/2 \rfloor+1}, \ldots, x_n$ реализует систему конъюнкций $\mathcal{K}_{\lceil n/2 \rceil}(x_{\lfloor n/2 \rfloor+1}, \ldots, x_n)$, а третья — каждую элементарную конъюнкцию из множества $\mathcal{K}_n(x_1, \ldots, x_n)$ «собирает» (с помощью одной операции конъюнкции) из двух ее «половинок», реализованных на некоторых выходах первой и второй подсхемы соответсвенно.

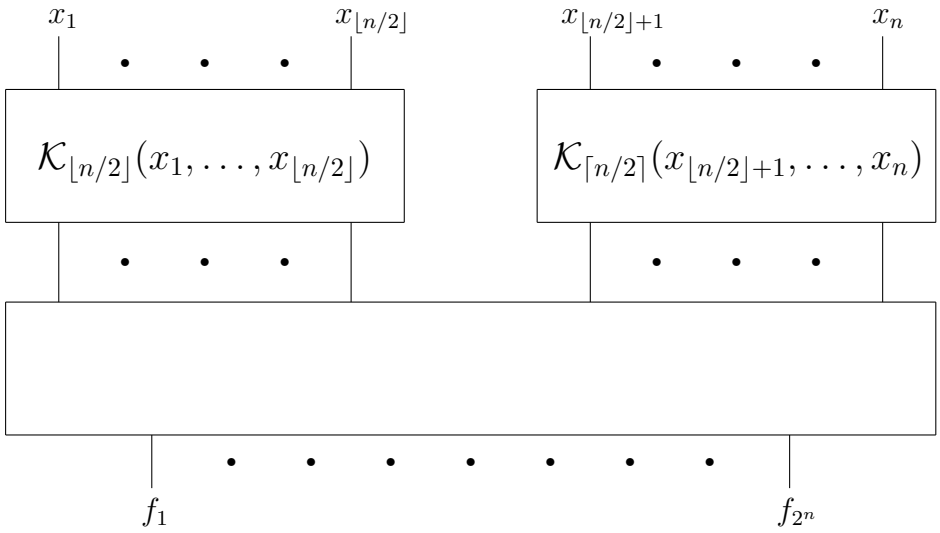


Рис. 2:

Используя тривиальные верхние оценки сложности первых двух под-

схем, получаем:

$$L(\mathcal{K}_n) \le L(\mathcal{K}_{\lfloor n/2 \rfloor}) + L(\mathcal{K}_{\lceil n/2 \rceil}) + 2^n \le$$
$$\le \left\lfloor \frac{n}{2} \right\rfloor + \left(\left\lfloor \frac{n}{2} \right\rfloor - 1 \right) 2^{\lfloor n/2 \rfloor} + \left\lceil \frac{n}{2} \right\rceil + \left(\left\lceil \frac{n}{2} \right\rceil - 1 \right) 2^{\lceil n/2 \rceil} + 2^n \le$$
$$\le 2^n + O\left(n\sqrt{2^n} \right).$$

□

Теорема 5. *При $n \to \infty$ верна следующая верхняя оценка функции Шеннона*

$$L(n) \lesssim 6\frac{2^n}{n}.$$

Доказательство. Пусть $f(x_1, \ldots, x_n)$ — самая сложная функция от n переменных. Опишем два различных метода построения схем, позволяющих получить указанную оценку сложности. Один метод принято называть *методом Шеннона*, другой — *методом каскадов*.

`Метод Шеннона`.

Введем натуральный параметр $k = k(n)$, значение которого выберем позже. Пока потребуем только, чтобы выполнялись условия $k < n$ и $n - k \to \infty$ при $n \to \infty$.

Разложим функцию $f(x_1, \ldots, x_n)$ по первым $n - k$ переменным:

$$f(x_1, \ldots, x_n) = \bigvee_{(\sigma_1, \ldots, \sigma_{n-k})} x_1^{\sigma_1} \ldots x_{n-k}^{\sigma_{n-k}} f(\sigma_1, \ldots, \sigma_{n-k}, x_{n-k+1}, \ldots, x_n).$$

Построим схему, реализующую функцию f как показано на рис. 3. Схема состоит из трех блоков (подсхем). Первая подсхема по переменным $x_1, \ldots, x_{n-k}$ реализует систему конъюнкций $\mathcal{K}_{n-k}(x_1, \ldots, x_{n-k})$, вторая по переменным $x_{n-k+1}, \ldots, x_n$ реализует систему всех 2^{2^k} функций от этих k переменных, а третья в соответствии с указанным разложением функции f реализует саму функцию f.

Применяя леммы 5 и 7, получаем:

$$L(n) = L(f) \le 2^{n-k} + o\left(2^{n-k}\right) + k2^k 2^{2^k} + 2 \cdot 2^{n-k} = \frac{3 \cdot 2^n}{2^k} + o\left(\frac{2^n}{2^k}\right) + k2^k 2^{2^k}.$$

Теперь положим $k = \lfloor \log(n - 3\log n) \rfloor$. Тогда

$$\frac{n - 3\log n}{2} < 2^k \le n - 3\log n.$$

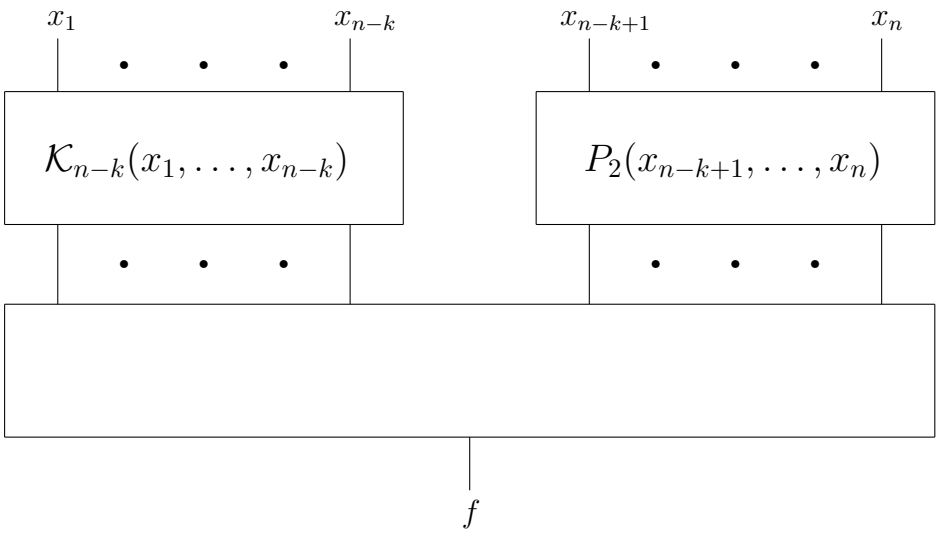


Рис. 3:

Поэтому

$$L(n) \leq 6\frac{2^n}{n} + o\left(\frac{2^n}{n}\right).$$

Доказательство теоремы 5 методом Шеннона завершено.

`Метод каскадов`.

Последовательно определим множества функций $G_0, G_1, \ldots, G_{n-1}$, имеющие вид

$$G_i = \{g_{i,1}(x_{i+1}, \ldots, x_n), \ldots, g_{i,r_i}(x_{i+1}, \ldots, x_n)\}, \quad i = 0, 1, \ldots, n-1,$$

следующим образом. Положим сначала

$$G_0 = \{g_{0,1}(x_1, \ldots, x_n\} = \{f(x_1, \ldots, x_n)\}.$$

Пусть уже определено множество

$$G_{i-1} = \{g_{i-1,1}(x_i, \ldots, x_n), \ldots, g_{i-1,r_{i-1}}(x_i, \ldots, x_n)\}.$$

Тогда множество

$$G_i = \{g_{i,1}(x_{i+1}, \ldots, x_n), \ldots, g_{i,r_i}(x_{i+1}, \ldots, x_n)\}$$

будут составлять все различные функции из списка

$$g_{i-1,1}(0, x_{i+1}, \ldots, x_n), \ldots, g_{i-1,r_{i-1}}(0, x_{i+1}, \ldots, x_n),$$
$$g_{i-1,1}(1, x_{i+1}, \ldots, x_n), \ldots, g_{i-1,r_{i-1}}(1, x_{i+1}, \ldots, x_n),$$

занумерованные произвольным образом.

Отметим два важных свойства множеств G_i.

1. Для любой функции g из множества G_{i-1} найдутся такие функции $g^{(1)}$ и $g^{(2)}$ из множества G_i, что справедливо равенство

$$g(x_i, \ldots, x_n) = x_i g^{(1)}(x_{i+1}, \ldots, x_n) \vee \overline{x}_i g^{(2)}(x_{i+1}, \ldots, x_n).$$

2. При $i = 1, 2, \ldots, n-1$ для количества r_i элементов в множестве G_i выполняются неравенства $r_i \leq 2r_{i-1} \leq 2^i$ и $r_{n-i} \leq 2^{2^i}$.

Перейдем к описанию схемы, последовательно реализующей множества функций $G_{n-1}, G_{n-2}, \ldots, G_0$ и, следовательно, на последнем шаге вычисляющей функцию $f(x_1, \ldots, x_n)$.

Так как $G_{n-1} \subseteq \{0, 1, x_n, \overline{x}_n\}$, то для реализации функций из множества G_{n-1} потребуется не более трех элементов.

После этого вычислим отрицания всех остальных переменных, затратив еще $n-1$ инвертор (т. е. элемент, реализующий отрицание функции, подаваемой на единственный вход этого элемента).

Далее, если уже реализованы все функции из множества G_i, то для вычисления любой функции из множества G_{i-1} в соотвествии со свойством 1 достаточно трех элементов. Поэтому

$$L(f) \leq 3 + (n-1) + \sum_{i=0}^{n-2} 3r_i.$$

Так же как и в методе Шеннона введем натуральный параметр $k = k(n)$, значение которого выберем позже. Для оценки сверху величины r_i в зависимости от выполнения условия $i \leq n-k-1$ применим разные оценки из свойства 2:

$$L(f) \leq n + 2 + 3\left(1 + 2 + \ldots + 2^{n-k-1}\right) + 3\left(2^{2^k} + 2^{2^{k-1}} + \ldots + 2^{2^1}\right) \leq$$
$$\leq n + 3 \times 2^{n-k} + 6 \times 2^{2^k}.$$

Полагая $k = \lfloor \log(n - 2\log n) \rfloor$, получаем

$$\frac{n - 2\log n}{2} < 2^k \leq n - 2\log n$$

и поэтому

$$L(n) \leq 6\frac{2^n}{n} + o\left(\frac{2^n}{n}\right).$$

Доказательство теоремы 5 методом каскадов завершено.

□

Отметим, что оценка сложности реализации системы всех булевых функций от фиксированных переменных, использованная при доказательстве теоремы 5, очень груба. На самом деле нетрудно найти точное значение этой величины, причем при реализации в любом полном базисе.

Упражнение 6. Пусть $[B] = P_2$. Доказать, что для любого натурального n верно равенство

$$L_B(P_2(x_1, \ldots, x_n)) = 2^{2^n} - n.$$

Теперь сформулируем два утверждения о сложности реализации булевых функций в произвольном базисе.

Теорема 6. *Пусть B_1 и B_2 — конечные множества булевых функций, причем $[B_1] = [B_2] = P_2$. Тогда найдутся такие положительные константы c_1 и c_2, что для любой булевой функции f выполняются неравенства*

$$c_1 L_{B_1}(f) \leq L_{B_2}(f) \leq c_2 L_{B_1}(f).$$

Доказательство. Положим

$$c_2 = \max_{\varphi \in B_1} L_{B_2}(\varphi).$$

Теперь, если в какой-либо минимальной схеме S, реализующей произвольную функцию f в базисе B_1, заменить все функциональные элементы, соответствующие функциям из базиса B_1, минимальными схемами в базисе B_2, реализующие те же самые функции, получим схему S', реализующую функцию f в базисе B_2 и имеющую сложность не более $c_2 L(S)$. Следовательно,

$$L_{B_2}(f) \leq L(S') \leq c_2 L(S) = c_2 L_{B_1}(f),$$

и второе неравенство доказано.

Полагая

$$c_1 = \left(\max_{\varphi \in B_2} L_{B_1}(\varphi) \right)^{-1},$$

можно аналогично устанавливить первое неравенство. □

Теоремы 5 и 6 устанавливают верхнюю оценку сложности реализации булевых функций в произвольном базисе. Сформулируем эту оценку как отдельную теорему.

Теорема 7. *Пусть* $[B] = P_2$. *Тогда при* $n \to \infty$ *выполняется соотношение*

$$L_B(n) \leq O\left(\frac{2^n}{n}\right).$$

Ниже будет показано, что эта оценка по порядку неулучшаема, почти все булевы функции имеют экспоненциальную по числу переменных сложность. В связи с этим возникает естественная и тесно связанная с практическими применениями задача выявления различных классов булевых функций, допускающих существенно более простую схемную реализацию. Одним из таких классов является множество симметрических булевых функций, к изучению которого со сложностной точки зрения мы и переходим.

Реализация симметрических функций

Прежде чем непосредственно заняться реализацией симметрических функций, рассмотрим две задачи, вспомогательные для исходной задачи, но имеющие серьезное самостоятельное значение.

Обозначим через Σ_n булев оператор суммирования n-разрядных чисел, т. е. булеву $(2n, n+1)$-функцию (систему из $n+1$ булевой функции от $2n$ переменных), которая по двум n-разрядным двоичным числам вычисляет $(n+1)$-разрядное двоичное представление их суммы, а через N_n — булев оператор подсчета числа единиц в наборе длины n, т. е. булеву $(n, \lceil \log(n+1) \rceil)$-функцию, которая по n-разрядному двоичному набору вычисляет $\lceil \log(n+1) \rceil$-разрядное двоичное представление количества единиц в этом наборе.

Лемма 8. *Для любого конечного полного базиса* B *при* $n \to \infty$ *верно равенство*

$$L_B(\Sigma_n) = O(n).$$

Доказательство. Утверждение леммы в силу теоремы 6 достаточно доказать для какого-нибудь конкретного конечного базиса. Пусть базис B содержит функции $x\&y$, $x \oplus y$, $x \oplus y \oplus z$ и $xy \vee xz \vee yz$.

Построим схему S, которая по двум группам входов — $(x_1, \ldots, x_n)$ и $(y_1, \ldots, y_n)$, на которые подаются двоичные n-разрядные числа (младшие разряды x_1 и y_1), вычисляет набор $(z_1, \ldots, z_{n+1})$, представляющий двоичную запись их суммы. Тогда, обозначив через u_i, $i = 2, \ldots, n+1$,

значение переноса в i-й разряд, получаем:

$$z_1 = x_1 \oplus y_1, \quad u_2 = x_1 y_1$$
$$z_i = x_i \oplus y_i \oplus u_i, \quad u_{i+1} = x_i y_i \vee x_i u_i \vee y_i u_i, \qquad i = 2, \ldots, n-1;$$
$$z_n = x_n \oplus y_n \oplus u_n, \quad z_{n+1} = u_{n+1} = x_n y_n \vee x_n u_n \vee y_n u_n.$$

Следовательно, $L_B(\Sigma_n) \leq 2n$. □

Лемма 9. *Для любого конечного полного базиса B при $n \to \infty$ верно равенство*

$$L_B(N_n) = O(n).$$

Доказательство. Очевидно, что результат применения оператора N_n равен двоичной записи суммы n подаваемых на входы оператора одноразрядных двоичных чисел. Опишем способ вычисления этой суммы схемой линейной сложности.

Сначала будем считать, что $n = 2^k$ для некоторого k. Построим схему S, имеющую k ярусов. Ярус с номером t, $t = 1, \ldots, k$, будет состоять из 2^{k-t} подсхем, каждая из которых реализует оператор Σ_t и, следовательно, имеет две группы по t входов, а также $t+1$ выходов. Таким образом, считая в силу леммы 8, что $L_B(\Sigma_t) \leq ct$, в случае, когда $n = 2^k$, имеем:

$$L_B(N_n) \leq L_B(S) \leq \sum_{t=1}^{k} 2^{k-t} ct = c2^k \sum_{t=1}^{k} \frac{t}{2^t} < 2c2^k = 2cn.$$

Переходя к общему случаю, полагаем $n' = 2^{\lceil \log n \rceil}$. Очевидно, что $n \leq n' < 2n$. Схему, реализующую оператор N_n, можно получить из схемы S', реализующей оператор $N_{n'}$, подав на $n' - n$ входов схемы S' константу 0. Поэтому

$$L_B(N_n) \leq L_B(0) + L_B(N_{n'}) \leq L_B(0) + 2cn' \leq L_B(0) + 4cn = O(n).$$

□

Теперь рассмотрим реализацию симметрических булевых функций. Напомним, что функция $f(x_1, \ldots, x_n)$ называется *симметрической*, если для любой перестановки σ из симметрической группы S_n выполняется равенство $f(x_{\sigma(1)}, \ldots, x_{\sigma(n)}) = f(x_1, \ldots, x_n)$.

Теорема 8. *Для произольного конечного полного базиса B найдутся такие положительные константы c_1 и c_2, что для любой симметрической булевой функции $f(x_1, \ldots, x_n)$, отличной от константы, выполняются неравенства*

$$c_1 n \leq L_B(f(x_1, \ldots, x_n)) \leq c_2 n.$$

Доказательство. Нижнюю оценку в силу существенной зависимости от всех своих переменных любой отличной от константы симметрической функции дает неравенство

$$L_B(f) \geq \frac{n-1}{r(B)-1},$$

где $r(B)$ — максимальное число существенных переменных у функций из базиса B.

Переходя к доказательству верхней оценки отметим, что произвольная симметрическая функция f от n переменных может быть задана двоичной последовательностью $\tilde{\pi}(f) = (\pi_0(f), \pi_1(f), \ldots, \pi_n(f))$, где $\pi_k(f)$ — значение функции f на наборах, состоящих из k единиц и $n-k$ нулей. Значение симметрической функции $f(x_1, \ldots, x_n)$ однозначно определяется по числу единиц в наборе $x_1, \ldots, x_n$, а следовательно, по двоичной записи этого числа. На этом и основан метод построения схемы S, вычисляющей функцию $f(x_1, \ldots, x_n)$.

Схема S состоит из подсхем S_1 и S_2. Подсхема S_1 реализует оператор N_n, на выходах подсхемы S_1 вычисляется двоичная запись длины $\lceil \log(n+1) \rceil$ числа единиц во входном наборе. Подсхема S_2 по двоичной записи числа единиц во входном наборе вычисляет значение функции f на этом наборе. В силу леммы 9 и теоремы 11 получаем:

$$L_B(f) \leq L_B(N_n) + L_B(\lceil \log(n+1) \rceil) = O(n) + O\left(\frac{n}{\log n}\right) = O(n).$$

□

Нижние оценки сложности булевых функций

На прошлой лекции мы двумя способами (методом Шеннона и методом каскадов) установили следующую верхнюю оценку функции Шеннона сложности реализации булевых функций от n переменных в базисе $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$:

$$L(n) \lesssim 6\ \frac{2^n}{n}.$$

Насколько хороша эта оценка? Чтобы ответить на этот вопрос, нужно получить как можно более высокую нижнюю оценку функции Шеннона $L(n)$. А пока мы имеем только оценку $L(n) \geq n-1$, так как любая функция от n переменных, существенно зависящая от n переменных, требует для своей реализации в базисе B_0 не менее $n-1$ двухвходового элемента. Чуть усилим эту оценку функции Шеннона. Заметим, что

любая немонотонная функция от n переменных, существенно зависящая от n переменных (например, функция $\overline{x_1 \vee \ldots \vee x_n}$), требует для своей реализации в базисе B_0 не менее $n-1$ двухвходового элемента и, как минимум, один инвертор (элемент, реализующий отрицание). Поэтому справедливо следующее утверждение.

Лемма 10. *Для любого натурального n справедливо неравенство*

$$L(n) \geq n.$$

Оценку из леммы 10, конечно, можно усилить. Однако, для конструктивно (явным образом) задаваемых последовательностей булевых функций известны лишь не более чем линейные по числу переменных нижние оценки сложности.

Тем не менее, можно доказать, что верхняя оценка функции Шеннона $L(n)$, доказываемая методом Шеннона или методом каскадов, неулучшаема по порядку роста величины, а для этого нужно установить экспоненциальную нижнюю оценку функции Шеннона. Эта оценка будет неконструктивная и будет получаться из мощностных соображений, идею которых проиллюстрируем на примере одной задачи из теории чисел.

Как доказать, что есть трансцендентные действительные числа (не являющиеся корнями ненулевых многочленов с рациональными коэффициентами)? Да, можно доказать, что $e, \pi, \sin 1, 2^{\sqrt{2}}, \log 3$ трансцендентны. Однако эти доказательства отнюдь не просты. В то же время элементарные мощностные рассуждения — множество действительных чисел континуально, а множество корней ненулевых многочленов с рациональными коэффициентами счетно — дают не только простейшее (правда, некоструктивное) доказательство этого факта, но и устанавливают, что почти все действительные числа трансцендентны.

Аналогичные рассуждения лежат и в основе экспоненциальной нижней оценки функции Шеннона. Сформулируем соответствующий очевидный факт в виде леммы, но сначала напомним определение минимальной схемы.

Схема, реализующая некоторую булеву функцию (или систему функций) в базисе B, называется *минимальной схемой в базисе B*, если никакая схема в базисе B меньшей сложности не реализует ту же самую функцию (систему функций).

Лемма 11. *Если число различных минимальных в базисе B схем S с n входами и одним выходом, удовлетворяющих условию $L_B(S) \leq k$, меньше величины 2^{2^n}, то выполняется неравенство*

$$L_B(n) > k.$$

Это утверждение непосредственно следует из того, что в условиях леммы число булевых функций $f(x_1, \ldots, x_n)$, удовлетворяющих условию $L_B(f) \leq k$, меньше величины 2^{2^n} — числа всех булевых функций от n переменных.

Таким образом, при доказательстве нижней оценки сложности функции Шеннона на первый план выходит задача о получении приемлемых верхних оценок числа минимальных схем, к которой мы и переходим.

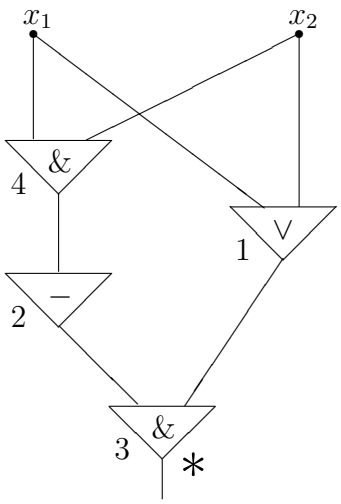
Для того, чтобы оценить (сверху) количество минимальных схем оценим количество схем в более широком классе — в классе так называемых приведенных (или правильных) схем.

Схему будем называть *приведенной* (*правильной*), если в ней нет двух разных элементов, реализующих одну и ту же функцию. Очевидно, что из любой схемы путем удаления некоторых элементов и «переподключения» выходящих из этих элементов ребер можно получить приведенную схему. Поэтому справедливо такое утверждение.

Лемма 12. *Любая минимальная схема является приведенной.*

Обозначим через $N_=(k, n)$ число приведенных схем в базисе $B_0 = \{x \vee y, x\&y, \overline{x}\}$ со входами, которым приписаны переменные $x_1, \ldots, x_n$, и одним выходом, имеющих сложность в точности k, а через $N_\leq(k, n)$ — число приведенных схем в базисе B_0 со входами, которым приписаны переменные $x_1, \ldots, x_n$, и одним выходом, имеющих сложность не более k.

Пусть S — приведенная схема в базисе B_0 сложности k со входами $x_1, \ldots, x_n$ и одним выходом. Занумеруем в произвольном порядке числами $1, 2, \ldots, k$ элементы схемы S. Обозначим эту нумерацию NUM. Схеме S с выбранной нумерацией элементов NUM сопоставим таблицу $T(S, NUM)$ высоты k и ширины 3 следующим образом. Пусть элемент E схемы S получил номер i. Тогда в i-й строке таблицы $T(S, NUM)$ в первом столбце указывается функция из базиса B_0, приписанная элементу E, а в других столбцах этой строки — символы из множества $\{x_1, \ldots, x_n\} \cup \{1, \ldots, k\}$: в $(j+1)$-й столбец, $j = 1, 2$, этой строки помещается информация о том, из какой вершины ведет ребро, соответствующее j-му входу элемента E (считаем, что входы элементов пронумерованы). Если ребро ведет из входа, помеченного переменной x_i, то в соответствующую клетку помещается символ x_i, а если ребро ведет из элемента с номером k, то — число k. Если у элемента E не два входа, а один (т. е. элемент E — инвертор), то оставшуюся пустой третью клетку i-й строки заполним для определенности так же, как и вторую клетку этой строки. Кроме того, если элемент E является выходом схемы, то i-я строка помечается дополнительно символом $*$ (если выходом является переменная, то помечается эта переменная).



∨	x_1	x_2	
−	4	4	
&	2	1	*
&	x_1	x_2	

Рис. 4:

На рис. 4 представлена схема, все элементы которой занумерованы в некотором порядке, а также таблица, соответствующая этой схеме с заданной нумерацией элементов.

Лемма 13. *Пусть S — приведенная схема в базисе B_0, а NUM_1 и NUM_2 — две отличные друг от друга нумерации элементов схемы S. Тогда*

$$T(S, NUM_1) \neq T(S, NUM_2).$$

Доказательство. Предположим противное, т. е. $NUM_1 \neq NUM_2$, но при этом $T(S, NUM_1) = T(S, NUM_2)$. Рассмотрим для схемы S еще одну нумерацию ее элементов — монотонную (правильную) нумерацию NUM_0. Нумерация NUM_0 обладает следующим свойством: для любого i, $1 \leq i \leq L(S)$, на каждый вход i-го относительно нумерации NUM_0 элемента подается либо переменная, либо выход элемента с номером (относительно нумерации NUM_0), меньшим i.

Среди всех элементов схемы S, которые имеют разные номера в нумерациях NUM_1 и NUM_2, выделим тот элемент E, который имеет наименьший номер относительно нумерации NUM_0. Пусть элемент E в нумерации NUM_1 имеет номер p, а в нумерации NUM_2 — номер q, $q \neq p$.

В силу выбора элемента E элементы, которые соединены ребрами со входами элемента E (если такие элементы есть), имеют одни и те же номера относительно нумераций NUM_1 и NUM_2, поэтому строка с номером p таблицы $T(S, NUM_1)$ и строка с номером q таблицы $T(S, NUM_2)$ заполнены идентично. Но по предположению, справедливо равенство

$T(S, NUM_1) = T(S, NUM_2)$, а значит, строки с номерами p и q таблицы $T(S, NUM_1)$ (а также таблицы $T(S, NUM_2)$) совпадают. Отсюда непосредственно следует, что в схеме S есть два элемента, вычисляющие одну и ту же функцию, что противоречит неприводимости схемы S. $\square$

Лемма 14. *Найдется $c > 0$, такое что при всех значениях k и n при выполнении условия $k \geq n$ справедливо неравенство*

$$N_{\leq}(k, n) \leq c^k k^k.$$

Доказательство. Оценим сверху число таблиц, которые соответствуют всевозможным нумерациям элементов всех приведенных схем со входами $x_1, \ldots, x_n$ и одним выходом сложности k. Каждую клетку первого столбца можно заполнить не более чем тремя способами, каждую клетку второго и третьего столбцов таблицы — не более чем $k+n$ способами. Кроме того, выбор места для пометки $*$ осуществляется $k+n$ способами. Поэтому всего таких таблиц не более $3^k(k+n)^{2k}(k+n)$ штук. Каждой приведенной схеме сложности k в силу леммы 13 соответствует $k!$ различных таблиц. Следовательно,

$$N_{=}(k, n) \leq \frac{3^k(k+n)^{2k}(k+n)}{k!}.$$

Используя условие $k \geq n$, соотношение $k! \geq (k/3)^k$ и неравенство $2k \leq 2^k$, получаем:

$$N_{=}(k, n) \leq \frac{3^k(2k)^{2k}(2k)}{\left(\frac{k}{3}\right)^k} \leq 72^k k^k.$$

Из последней оценки с учетом справедливости при $l \leq k$ неравенства $N_{=}(l, n) \leq N_{=}(k, n)$ легко получается верхняя оценка величины $N_{\leq}(k, n)$:

$$N_{\leq}(k, n) = \sum_{l=0}^{\lfloor k \rfloor} N_{=}(l, n) \leq \sum_{l=0}^{\lfloor k \rfloor} 72^l l^l \leq (k+1)72^k k^k \leq 144^k k^k.$$

$\square$

Теорема 9 (мощностная нижняя оценка)**.** *Для любого $\varepsilon > 0$ доля булевых функций f от n фиксированных переменных, удовлетворяющих условию*

$$L(f) \geq (1 - \varepsilon)\frac{2^n}{n},$$

стремится к 1 при $n \to \infty$.

Доказательство. Положим

$$k_\varepsilon = (1-\varepsilon)\frac{2^n}{n}.$$

Число функций f от n фиксированных переменных, удовлетворяющих условию $L(f) \le k_\varepsilon$, не превосходит величины $N_{\le}(k_\varepsilon, n)$. Поэтому достаточно установить, что при $n \to \infty$

$$\frac{N_{\le}(k_\varepsilon, n)}{2^{2^n}} \to 0$$

или, что то же самое,

$$\log_2 \frac{N_{\le}(k_\varepsilon, n)}{2^{2^n}} \to -\infty.$$

Действительно, применяя лемму 14, имеем:

$$\log_2 \frac{N_{\le}(k_\varepsilon, n)}{2^{2^n}} \le k_\varepsilon \log_2 c + k_\varepsilon \log_2 k_\varepsilon - 2^n \le$$
$$\le (1-\varepsilon)\frac{2^n}{n}\log_2 c + (1-\varepsilon)\frac{2^n}{n}\log_2 (2^n) - 2^n = -\varepsilon 2^n + O\left(\frac{2^n}{n}\right).$$

Последнее выражение стремится к $-\infty$ при $n \to \infty$. □

Следствие 2. *При $n \to \infty$ выполняется асимптотическое неравенство*

$$L(n) \gtrsim \frac{2^n}{n}.$$

Теорема 9 вместе с верхними оценками, полученными методом Шеннона и методом каскадов, устанавливает порядок роста функции Шеннона $L(n)$.

Теперь тем же самым способом на основе леммы 14 с некоторым увеличением технических выкладок усилим доказанную нижнюю оценку функции Шеннона. Покажем, что функция Шеннона $L(n)$ при всех достаточно больших n превосходит величину $2^n/n$ (этот факт не следует ни из теоремы 9, ни из следствия к этой теореме), причем разность этих величин растет экспоненциально.

Теорема 10 (усиленная мощностная нижняя оценка)**.** *Для любого $\varepsilon > 0$ доля булевых функций f от n фиксированных переменных, удовлетворяющих условию*

$$L(f) \ge \frac{2^n}{n}\left(1 + (1-\varepsilon)\frac{\log_2 n}{n}\right),$$

стремится к 1 при $n \to \infty$.

Доказательство. Положим

$$k_\varepsilon = \frac{2^n}{n} + (1-\varepsilon)\frac{2^n \log_2 n}{n^2}.$$

Установим, что при $n \to \infty$

$$\log_2 \frac{N_{\leq}(k_\varepsilon, n)}{2^{2^n}} \to -\infty.$$

Применяя лемму 14, имеем:

$$\log_2 \frac{N_{\leq}(k_\varepsilon, n)}{2^{2^n}} \leq k_\varepsilon \log_2 c + k_\varepsilon \log_2 k_\varepsilon - 2^n \leq$$
$$\leq O\left(\frac{2^n}{n}\right) + \left(\frac{2^n}{n} + (1-\varepsilon)\frac{2^n \log_2 n}{n^2}\right) \log_2 \left(2\,\frac{2^n}{n}\right) - 2^n =$$
$$= -\varepsilon \frac{2^n \log_2 n}{n} + O\left(\frac{2^n}{n}\right).$$

Последнее выражение стремится к $-\infty$ при $n \to \infty$. □

Следствие 3. *При $n \to \infty$ выполняется асимптотическое неравенство*

$$L(n) - \frac{2^n}{n} \gtrsim \frac{2^n \log_2 n}{n^2}.$$

Теперь вернемся к оценкам роста функции Шеннона сложности реализации булевых функций схемами из функциональных элементов в произвольном полном конечном базисе.

Из теоремы 6 и полученных верхней и нижней оценок функции Шеннона сложности реализации булевых функций схемами из функциональных элементов в базисе $B_0 = \{x \vee y,\, x\&y,\, \overline{x}\}$ непосредственно устанавливается порядок роста функции Шеннона для произвольного полного конечного базиса.

Теорема 11. *Пусть B — конечное множество булевых функций, причем $[B] = P_2$. Тогда найдутся такие положительные константы a и b, что при $n \to \infty$ выполняются асимптотические неравенства*

$$a\frac{2^n}{n} \lesssim L_B(n) \lesssim b\frac{2^n}{n}.$$

Асимптотически наилучший метод О. Б. Лупанова

Теперь на примере базиса $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$ рассмотрим предложенный О. Б. Лупановым метод построения схем, который является асимптотически наилучшим для почти всех булевых функций.

Теорема 12 (О. Б. Лупанов)**.** *Пусть* $n \to \infty$. *Тогда*

$$L(n) \leq \frac{2^n}{n}\left(1 + O\left(\frac{\log n}{n}\right)\right).$$

Доказательство. Опишем метод, который позволяет для произвольной (в том числе и для самой сложной) функции от n переменных построить схему, состоящую не более чем из $\frac{2^n}{n}\left(1 + O\left(\frac{\log n}{n}\right)\right)$ элементов.

Пусть $k = k(n)$ — натуральный параметр, удовлетворяющий при $n \to \infty$ условиям $k \to \infty$ и $n - k \to \infty$. Точное значение этого параметра укажем позже.

Таблицу из 2^n значений произвольной функции $f(x_1, \ldots, x_n)$ представим в виде прямоугольной таблицы высоты 2^k и ширины 2^{n-k} как показано на рис. 5.

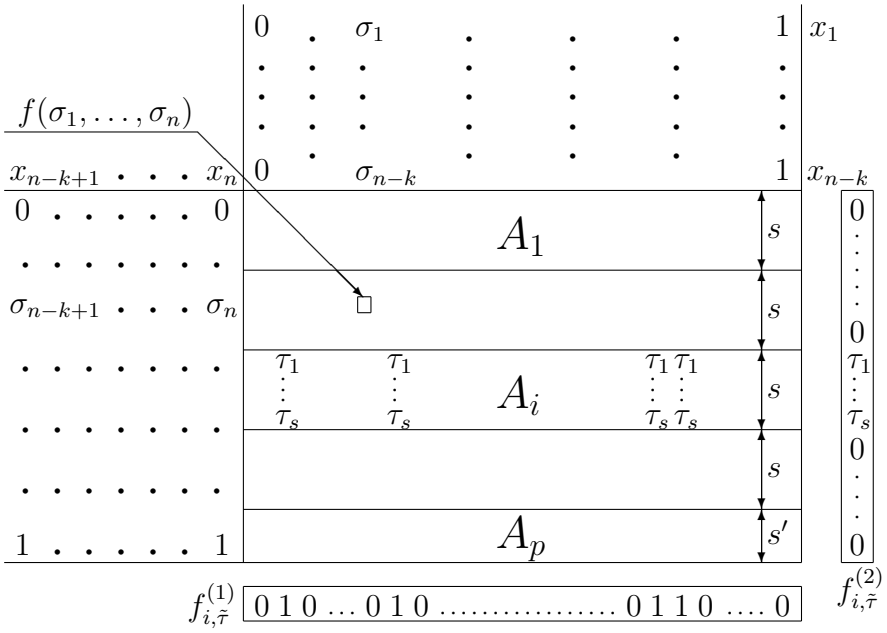


Рис. 5:

Пусть $s = s(n)$ — также некоторый натуральный параметр, удовлетворяющий при $n \to \infty$ условиям $s \to \infty$ и $\frac{2^k}{s} \to \infty$. Точное значение

этого параметра укажем позже. Таблицу разобъем на горизонтальные полосы $A_1, \ldots, A_p$ высоты s (полоса A_p имеет высоту $s' \leq s$), $p = \left\lceil \frac{2^k}{s} \right\rceil$. Для $i = 1, \ldots, p$ через $f_i(x_1, \ldots, x_n)$ обозначим функцию, значения которой совпадают со значениями функции $f(x_1, \ldots, x_n)$ на полосе A_i, и равны 0 на остальных полосах. Тогда

$$f(x_1, \ldots, x_n) = \bigvee_{i=1}^{p} f_i(x_1, \ldots, x_n).$$

Теперь для каждой пары $(i, \tilde{\tau})$, $i = 1, \ldots, p$, $\tilde{\tau} \in \{0,1\}^s$ (или $\tilde{\tau} \in \{0,1\}^{s'}$ при $i = p$), обозначим через $f_{i,\tilde{\tau}}(x_1, \ldots, x_n)$ функцию, таблица которой получается из таблицы функции $f_i(x_1, \ldots, x_n)$ путем обнуления всех столбцов полосы A_i, значения в которых не сопадают с набором $\tilde{\tau}$. Тогда

$$f(x_1, \ldots, x_n) = \bigvee_{i=1}^{p} \bigvee_{\tilde{\tau}} f_{i,\tilde{\tau}}(x_1, \ldots, x_n).$$

Наконец, у каждой функции $f_{i,\tilde{\tau}}(x_1, \ldots, x_n)$ можно разделить переменные, точнее представить эту функцию в виде

$$f_{i,\tilde{\tau}}(x_1, \ldots, x_n) = f^{(1)}_{i,\tilde{\tau}}(x_1, \ldots, x_{n-k}) f^{(2)}_{i,\tilde{\tau}}(x_{n-k+1}, \ldots, x_n),$$

где функция $f^{(1)}_{i,\tilde{\tau}}(x_1, \ldots, x_{n-k})$ обращается в единицу только на таких наборах $(\sigma_1, \ldots, \sigma_{n-k})$, что в соответствующих этим наборам столбцах полосы A_i находится набор $\tilde{\tau}$, а столбец значений функции $f^{(2)}_{i,\tilde{\tau}}(x_{n-k+1}, \ldots, x_n)$ совпадает с набором $\tilde{\tau}$ на полосе A_i, а на наборах вне полосы A_i функция $f^{(2)}_{i,\tilde{\tau}}(x_{n-k+1}, \ldots, x_n)$ равна 0.

Возвращаясь к представлению функции f, окончательно получаем:

$$f(x_1, \ldots, x_n) = \bigvee_{i=1}^{p} \bigvee_{\tilde{\tau}} f^{(1)}_{i,\tilde{\tau}}(x_1, \ldots, x_{n-k}) f^{(2)}_{i,\tilde{\tau}}(x_{n-k+1}, \ldots, x_n).$$

Схема S, реализующая функцию $f(x_1, \ldots, x_n)$, будет состоять из подсхем S_i, $i = 1, \ldots, 6$, — см. рис. 6.

Подсхема S_1, на входы которой подаются переменные $x_1, \ldots, x_{n-k}$, реализует систему функций $\mathcal{K}_{n-k}(x_1, \ldots, x_{n-k})$. В силу леммы о сложности реализации системы элементарных конъюнкций можно считать, что при $n \to \infty$

$$L(S_1) = 2^{n-k} + o\left(2^{n-k}\right) \leq 2 \times 2^{n-k}.$$

Подсхема S_2, на входы которой подаются переменные $x_{n-k+1}, \ldots, x_n$, реализует систему функций $\mathcal{K}_k(x_{n-k+1}, \ldots, x_n)$. Также в силу леммы о

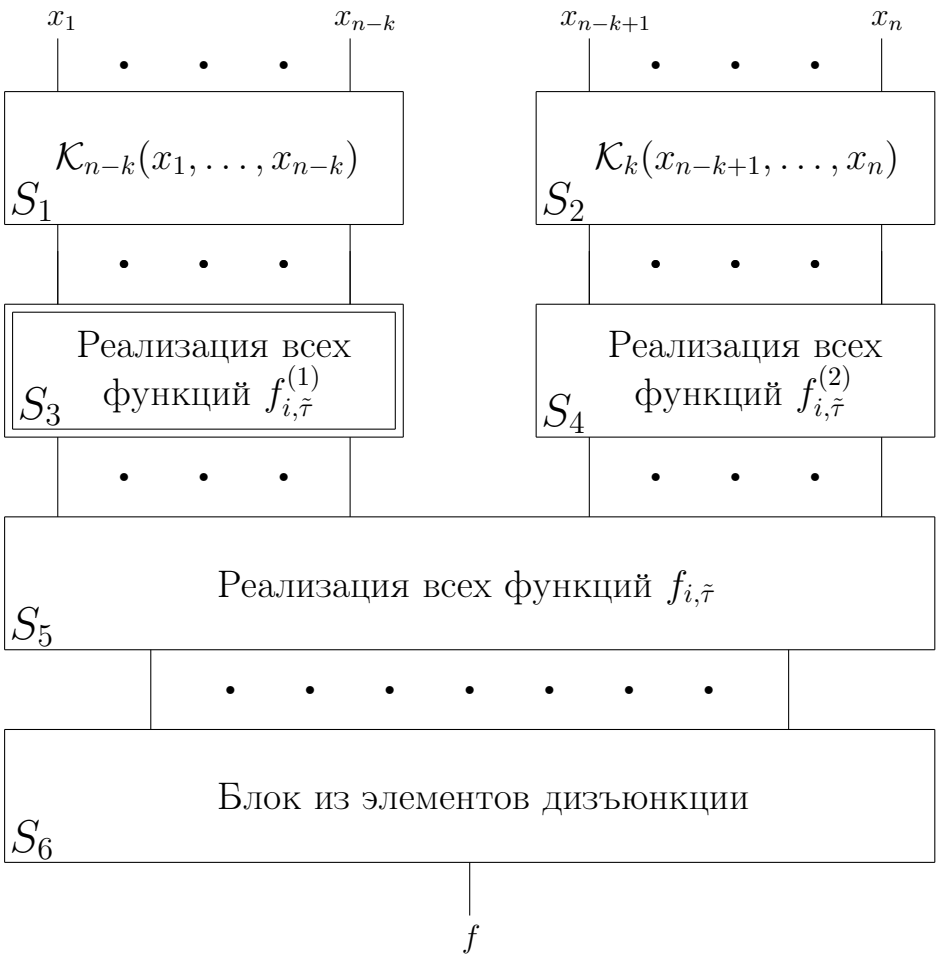


Рис. 6:

сложности реализации системы элементарных конъюнкций можно считать, что при $n \to \infty$

$$L(S_2) = 2^k + o\left(2^k\right) \leq 2 \times 2^k.$$

Подсхема S_3, на входы которой подаются элементарные конъюнкции из системы $\mathcal{K}_{n-k}(x_1, \ldots, x_{n-k})$, состоит только из дизъюнкторов и реализует функции $f^{(1)}_{i,\tilde{\tau}}(x_1, \ldots, x_{n-k})$ для всех i и $\tilde{\tau}$ в соответствии с представлением этих функций в виде совершенной дизъюнктивной нормальной

формы. Учитывая равенство[6]

$$\sum_{\tilde{\tau}} \left| N_{f^{(1)}_{i,\tilde{\tau}}} \right| = 2^{n-k},$$

справедливое для всех $i = 1, \ldots, p$, получаем следующую оценку:

$$L(S_3) \leq p2^{n-k}.$$

Подсхема S_4, на входы которой подаются элементарные конъюнкции из системы $\mathcal{K}_k(x_{n-k+1}, \ldots, x_n)$, состоит только из дизъюнкторов и реализует функции $f^{(2)}_{i,\tilde{\tau}}(x_{n-k+1}, \ldots, x_n)$ для всех i и $\tilde{\tau}$ также в соответствии с представлением этих функций в виде совершенной дизъюнктивной нормальной формы. Используя очевидное равенство

$$\left| N_{f^{(2)}_{i,\tilde{\tau}}} \right| \leq s,$$

выполняющееся для всех допустимых значений i и $\tilde{\tau}$, получаем:

$$L(S_4) \leq p2^s s.$$

Подсхема S_5, на входы которой подаются выходы подсхем S_3 и S_4, состоит только из конъюнкторов и реализует функции $f_{i,\tilde{\tau}}(x_1, \ldots, x_n)$ для всех i и $\tilde{\tau}$. Сложность подсхемы S_5 можно оценить сверху числом различных пар $(i, \tilde{\tau})$:

$$L(S_5) \leq p2^s.$$

Подсхема S_6, на входы которой подаются все функции$f_{i,\tilde{\tau}}(x_1, \ldots, x_n)$ для всех i и $\tilde{\tau}$, состоит только из дизъюнкторов и реализует функцию $f(x_1, \ldots, x_n)$.Сложность подсхемы S_6 также можно оценить сверху числом различных пар $(i, \tilde{\tau})$:

$$L(S_6) < p2^s.$$

Таким образом,

$$\begin{aligned} L(f) \leq L(S) = \sum_{i=1}^{6} L(S_i) \leq \\ \leq 2 \times 2^{n-k} + 2 \times 2^k + p2^{n-k} + p2^s s + 2p2^s \leq \\ \leq \left(\frac{2^k}{s} + 1 \right) 2^{n-k} + \left(\frac{2^k}{s} + 1 \right) 2^s (s+2) + 2^{n-k+1} + 2^{k+1} \leq \\ \leq \frac{2^n}{s} + 4 \times 2^{s+k} + 3 \times 2^{n-k} + 2 \times 2^k. \end{aligned}$$

[6]Здесь и далее через N_g обозначается множество наборов переменных функции g, на которых эта функция равна 1.

Теперь полагая

$$k = \lfloor 3 \log n \rfloor, \quad s = \lfloor n - 5 \log n \rfloor,$$

легко проверить, что условия $k \to \infty$, $k \to \infty$, $s \to \infty$, $\frac{2^k}{s} \to \infty$ выполнены. Подставляя значения k и s в полученную оценку, имеем:

$$L(f) \leq \frac{2^n}{n} \left(1 + O\left(\frac{\log n}{n}\right)\right).$$

Утверждение теоремы следует из справедливости этой оценки для функции $f(x_1, \ldots, x_n)$, удовлетворяющей условию $L(f) = L(n)$. □

Упражнение 7. Доказать, что при $n \to \infty$ для любого фиксированного $r \geq 2$ справедливо асимптотическое неравенство

$$L_{\{x_1 \& \ldots \& x_r, \overline{x}\}}(n) \sim \frac{1}{r-1} \frac{2^n}{n}.$$

Отметим важный факт, который вытекает из теорем 9 и 12: почти все функции от n переменных имеют сложность, асимптотически совпадающую со сложностью самой сложной функции. Такой эффект называется *эффектом Шеннона.*

Реализация самодвойственных функций

В качестве одного из разнообразных применений теоремы Лупанова рассмотрим задачу о сложности реализации самодвойственных функций в базисе $B_0 = \{x \vee y,\ x\&y,\ \overline{x}\}$.

Определим функцию Шеннона сложности реализации самодвойственных функций в базисе B_0 равенством

$$L^S(n) = \max_{f(x_1, \ldots, x_n) \in S} L(f).$$

Теорема 13. *При $n \to \infty$ справедливо асимптотическое равенство*

$$L^S(n) \sim \frac{2^{n-1}}{n}.$$

Доказательство. Если в доказательстве теоремы 9 положить

$$k_\varepsilon = (1 - \varepsilon) \frac{2^{n-1}}{n}$$

и сравнить величину $N_{\leq}(k_\varepsilon, n)$ не с числом 2^{2^n} всех булевых функций от n переменных, а с числом $2^{2^{n-1}}$ всех самодвойственных функций от n переменных, то получится такая нижняя оценка:

$$L^S(n) \gtrsim \frac{2^{n-1}}{n-1} \sim \frac{2^{n-1}}{n}.$$

Построим схему в базисе $B_0 = \{\vee, \&, {}^-\}$ для произвольной самодвойственной функции $f(x_1, \ldots, x_n)$. Положим

$$g(x_1, \ldots, x_{n-1}) = f(x_1, \ldots, x_{n-1}, 0).$$

Тогда

$$f(x_1, \ldots, x_{n-1}, 1) = \overline{f(\overline{x}_1, \ldots, \overline{x}_{n-1}, 0)} = \overline{g(\overline{x}_1, \ldots, \overline{x}_{n-1})}.$$

Таким образом, для реализации функции $f(x_1, \ldots, x_n)$ достаточно при $x_n = 0$ реализовать функцию $g(x_1, \ldots, x_{n-1})$, а при $x_n = 1$ — функцию $\overline{g(\overline{x}_1, \ldots, \overline{x}_{n-1})}$.

Преобразуем минимальную схему S_g, реализующую в базисе B_0 функцию $g(x_1, \ldots, x_{n-1})$, в схему S_f, реализующую функцию $f(x_1, \ldots, x_n)$. На i-й вход схемы, $i = 1, \ldots, n-1$, вместо переменной x_i подадим выход подсхемы, реализующей функцию $x_i \oplus x_n$ (схема для функции $x_1 \oplus x_2$ приведена на рис. 4). Выход схемы S_g, а также переменную x_n, подадим на входы еще одной подсхемы, реализующей сумму по модулю 2 своих входов. Получим схему S_f, реализующую функцию f, причем

$$L(S_f) = L(S_g) + nL(S_\oplus) = L(S_g) + 4n.$$

Применяя для оценки сложности минимальной схемы, реализующей функцию g от $n-1$ переменной, теорему 12, получаем требуемую верхнюю оценку. □