Московский Государственный Технический Университет им. Н.Э. Баумана

Факультет ИУ
Кафедра ИУ-8

Чашкин А.В.

Булевы функции и преобразования

Оглавление

Глава 1.

Булевы функции

1.1. Булев куб

1. Константы 0 и 1 называются *булевыми* константами. Упорядоченные наборы из нулей и единиц будем называть *двоичными* или *булевыми наборами*. Символы 0 и 1, входящие в двоичный набор, называются разрядами набора. Число разрядов набора называется его длиной. Разряды каждого набора длины n нумеруются целыми числами от 1 до n слева направо: крайний левый разряд получает номер 1, крайний правый — номер n. Множество всех булевых наборов длины n называется булевым кубом размерности n и обозначается символом $\mathbb{B}^n$. Название булев куб возникло благодаря геометрической интерпретации множества $\mathbb{B}^n$. Если наборы из этого множества рассматривать как элементы действительного n-мерного пространства $\mathbb{R}^n$, то нетрудно убедиться, что они будут расположены в вершинах единичного куба. Поэтому часто булевы наборы называются также вершинами n-мерного единичного куба.

Номером и *весом* набора $\boldsymbol{u} = (u_1, \ldots, u_n)$ из $\mathbb{B}^n$ называются величины

$$|\boldsymbol{u}| = \sum_{i=1}^{n} u_i \cdot 2^{n-i}, \qquad \|\boldsymbol{u}\| = \sum_{i=1}^{n} u_i.$$

Номера наборов задают на множестве $\mathbb{B}^n$ отношение линейного порядка $\leq$. Будем говорить, что набор $\boldsymbol{u}$ не больше набора $\boldsymbol{v}$, если $|\boldsymbol{u}| \leq |\boldsymbol{v}|$. Порядок, определяемый отношением $\leq$, называется *лексикографическим*.

Множество всех наборов длины n и веса k образует *k-й слой* куба $\mathbb{B}^n$, обозначаемый через $\mathbb{B}_k^n$. Число наборов в k-м слое n-мерного единичного куба равно числу сочетаний из n элементов по k — столькими способами среди n разрядов произвольного набора можно выбрать k разрядов, равных единице. Следовательно,

$$|\mathbb{B}_k^n| = \binom{n}{k} = \frac{n!}{(n-k)!k!}.$$

Расстоянием Хемминга между вершинами $\boldsymbol{u}$ и $\boldsymbol{v}$ куба $\mathbb{B}^n$ называется число $d(\boldsymbol{u}, \boldsymbol{v}) = \sum_{i=1}^{n} |u_i - v_i|$, равное количеству несовпадающих разрядов $\boldsymbol{u}$ и $\boldsymbol{v}$. Нетрудно показать, что расстояние d является метрикой, т.е. d — положительная симметрическая функция двух аргументов, принимающая значение нуль тогда и только тогда, когда два ее аргумента совпадают, и для которой справедливо неравенство треугольника: $d(\boldsymbol{u}, \boldsymbol{v}) \leq d(\boldsymbol{u}, \boldsymbol{w}) + d(\boldsymbol{w}, \boldsymbol{v})$ для любых трех наборов $\boldsymbol{u}$, $\boldsymbol{v}$ и $\boldsymbol{w}$ из $\mathbb{B}^n$. Наборы $\boldsymbol{u}$ и $\boldsymbol{v}$ называются *соседними*, если $d(\boldsymbol{u}, \boldsymbol{v}) = 1$. Если $d(\boldsymbol{u}, \boldsymbol{v}) = n$, то наборы называются *противоположными*. Соседние наборы различаются между собой только в одном разряде, противоположные наборы — во всех разрядах.

Пример 1.1.1. В трехмерном булевом кубе $\mathbb{B}^3$ рассмотрим четыре набора $\boldsymbol{u}_1 = (000)$, $\boldsymbol{u}_2 = (100)$, $\boldsymbol{u}_3 = (110)$ и $\boldsymbol{u}_4 = (111)$. Для номеров и весов этих наборов справедливы равенства:

$$|\boldsymbol{u}_1| = 0, \quad |\boldsymbol{u}_2| = 4, \quad |\boldsymbol{u}_3| = 6, \quad |\boldsymbol{u}_4| = 7,$$
$$\|\boldsymbol{u}_1\| = 0, \quad \|\boldsymbol{u}_2\| = 1, \quad \|\boldsymbol{u}_3\| = 2, \quad \|\boldsymbol{u}_4\| = 3.$$

Попарные расстояния между рассматриваемыми наборами равны

$$d(\boldsymbol{u}_1,\boldsymbol{u}_2)=1,\quad d(\boldsymbol{u}_1,\boldsymbol{u}_3)=2,\quad d(\boldsymbol{u}_1,\boldsymbol{u}_4)=3,$$
$$d(\boldsymbol{u}_2,\boldsymbol{u}_3)=1,\quad d(\boldsymbol{u}_2,\boldsymbol{u}_4)=2,\quad d(\boldsymbol{u}_3,\boldsymbol{u}_4)=1.$$

Следовательно, наборы $\boldsymbol{u}_1$ и $\boldsymbol{u}_4$ — противоположные, а наборы $\boldsymbol{u}_i$ и $\boldsymbol{u}_{i+1}$ при $i=1,2,3$ — соседние. □

Пары соседних вершин булева куба называются *ребрами*. Если $\boldsymbol{u},\boldsymbol{v}$ — соседние вершины, различающиеся в i-м разряде, то говорят, что ребро $(\boldsymbol{u},\boldsymbol{v})$ проходит в i-м направлении и соединяет эти вершины. Пусть $i_1,\ldots,i_{n-k}$ — попарно различные натуральные числа, не превосходящие n, $u_1,\ldots,u_{n-k}$ — булевы константы. Множество $\{\boldsymbol{v}\in\mathbb{B}^n \mid v_{i_j}=u_j, j=1,2,\ldots,n-k\}$ называется *k-мерной гранью* куба $\mathbb{B}^n$. Легко видеть, что k-мерная грань n-мерного булева куба содержит 2^k различных вершин.

Пусть $\boldsymbol{u}\in\mathbb{B}^n$. *Шаром* радиуса k с центром в наборе $\boldsymbol{u}$ называется множество $B_{n,k}(\boldsymbol{u})$ состоящее из всех таких $\boldsymbol{v}\in\mathbb{B}^n$, что $d(\boldsymbol{u},\boldsymbol{v})\le k$. *Сферой* радиуса k с центром в наборе $\boldsymbol{u}$ называется множество $S_{n,k}(\boldsymbol{u})$ состоящее из всех таких $\boldsymbol{v}\in\mathbb{B}^n$, что $d(\boldsymbol{u},\boldsymbol{v})=k$. Легко видеть, что k-й слой $\mathbb{B}^n_k$ является сферой радиуса k с центром в нулевом наборе и сферой радиуса $n-k$ с центром в единичном наборе. Непосредственно из определений сферы и шара следует, что для любого $\boldsymbol{u}\in\mathbb{B}^n$ и любого $k=0,1,\ldots,n$

$$|S_{n,k}(\boldsymbol{u})|=\binom{n}{k},\qquad |B_{n,k}(\boldsymbol{u})|=\sum_{i=0}^{k}\binom{n}{i}.$$

Пример 1.1.2. Рассмотрим трехмерный булев куб $\mathbb{B}^3$, изображенный на расположенном слева рис. 1.1.1. Этот куб содержит восемь вершин, образующих в нем четыре слоя. Нулевой слой $\mathbb{B}^3_0$ состоит из единственной вершины (000). Первый слой $\mathbb{B}^3_1$ содержит три вершины: (100), (010) и (001). Второй слой $\mathbb{B}^3_2$ также содержит три вершины: (110), (101) и (011). И, наконец, третий слой $\mathbb{B}^3_3$ куба состоит из одной вершины (111). В кубе 12 ребер, изображенных на рисунке отрезками, соединяющими вершины. В каждом из трех направлений проходит по четыре ребра. В $\mathbb{B}^3$ содержится шесть двумерных граней, содержащих по четыре вершины. Вершины одной из этих граней отмечены на рисунке черными кружками. Легко видеть, что номера отмеченных вершин принадлежат множеству $\{1,3,5,7\}$. Шар $B_{3,1}(010)$ состоит из вершины (010) и трех вершин, соединенных с ней ребрами: (000), (110) и (011). Три последние вершины образуют сферу $S_{3,1}(010)$. □

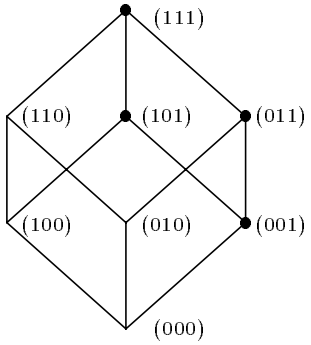


Рис. 1.1.1

2. Подмножество $G=\{\boldsymbol{g}_1,\ldots,\boldsymbol{g}_m\}$ булева куба $\mathbb{B}^n$ называется *двоичным кодом* с кодовым расстоянием d, если для любых двух его элементов $\boldsymbol{g}_i$ и $\boldsymbol{g}_j$ расстояние между ними не меньше d. Говорят, что код G исправляет t ошибок, если его кодовое расстояние не меньше чем $2t+1$. Заметим, что если вокруг каждого элемента $\boldsymbol{g}$ кода G с расстоянием $2t+1$ построит шар радиуса t с центром в этом элементе, то шары с центрами в разных элементах не будут пересекаться. Действительно, если некоторый набор $\boldsymbol{v}$ принадлежит пересечению шаров с центрами в элементах $\boldsymbol{g}_i$ и $\boldsymbol{g}_j$, то это означает, что $d(\boldsymbol{v},\boldsymbol{g}_i)\le t$ и $d(\boldsymbol{v},\boldsymbol{g}_j)\le t$. Но тогда в силу неравенства треугольника $d(\boldsymbol{g}_i,\boldsymbol{g}_j)\le d(\boldsymbol{v},\boldsymbol{g}_i)+d(\boldsymbol{v},\boldsymbol{g}_j)\le 2t$. Противоречие.

Пример 1.1.3. Снова рассмотрим трехмерный булев куб $\mathbb{B}^3$. В нем любые две противоположные вершины образуют код с расстоянием три, например, $\{(000),(111)\}$. Кодами с расстоянием два являются множество всех наборов четного веса и множество всех наборов нечетного веса. □

Обозначим через $m(n,d)$ максимально возможное число элементов в двоичном коде длины n, кодовое расстояние которого равно d. Для кодов с нечетным кодовым расстоянием имеет место следующий результат.

Теорема 1.1.1. *Справедливы неравенства*

$$\frac{2^n}{\sum_{i=1}^{2t}\binom{n}{i}} \le m(n,2t+1) \le \frac{2^n}{\sum_{i=1}^{t}\binom{n}{i}}.$$

Доказательство. Верхняя оценка величины $m(n,2t+1)$ легко следует из замечания, сделанного перед формулировкой теоремы. Рассмотрим в $\mathbb{B}^n$ произвольный код G с расстоянием $2t+1$, и построим вокруг каждого его элемента шар радиуса t. Так как эти шары не пересекаются, и каждый шар состоит ровно из $\sum_{i=1}^{t}\binom{n}{i}$ наборов, то все шары вместе содержат $|G|\sum_{i=1}^{t}\binom{n}{i}$ наборов. С другой стороны, шары лежат в $\mathbb{B}^n$ и, поэтому, содержат не более 2^n наборов. Следовательно, $|G|\sum_{i=1}^{t}\binom{n}{i} \le 2^n$.

Для доказательства нижней оценки опишем индуктивную процедуру построения в $\mathbb{B}^n$ кода G с расстоянием $2t+1$. Первый элемент кода выберем произвольно, вокруг выбранного элемента построим шар радиуса $2t$. В качестве второго элемента кода возьмем произвольный набор, не принадлежащий построенному шару. Очевидно, что расстояние между выбранными наборами не меньше $2t+1$. Вокруг второго элемента также построим шар радиуса $2t$. Теперь допустим, что выбраны k элементов, попарные расстояния между которыми не меньше $2t+1$, и вокруг каждого выбранного элемента построен шар радиуса $2t$. Если

$$k\sum_{i=1}^{2t}\binom{n}{i} < 2^n, \tag{1.1.1}$$

то можно выбрать очередной $(k+1)$-й элемент, не принадлежащий построенным шарам, и, следовательно, находящийся на расстоянии не меньшем $2t+1$ от каждого из выбранных ранее элементов. Легко видеть, что построение кода можно продолжать до тех пор пока справедливо неравенство (1.1.1). В тот момент, когда это неравенство впервые нарушится, код G будет состоять из k элементов, и для k будет справедливо неравенство $k\sum_{i=1}^{2t}\binom{n}{i} \ge 2^n$. Теорема доказана.

3. Вершины в булевом кубе распределены по слоям неравномерно. При больших n большинство вершин лежит в узкой полосе, состоящей из $2\sqrt{n\log_2 n}$ средних слоев. Более точно, имеет место следующее утверждение.

Теорема 1.1.2. *При $n\to\infty$ справедливо асимптотическое равенство*

$$\left|\bigcup_{k=\lfloor n/2-\sqrt{n\log_2 n}\rfloor}^{\lfloor n/2+\sqrt{n\log_2 n}\rfloor}\mathbb{B}_k^n\right| \sim \left|\mathbb{B}^n\right| = 2^n.$$

Доказательство. Оценим число наборов вес которых отличается от $\frac{n}{2}$ более чем на t единиц:

$$\left|\bigcup_{k:|n/2-k|>t}\mathbb{B}_k^n\right| = \sum_{k:|n/2-k|>t}\binom{n}{k} = \sum_{k:|n/2-k|>t}\frac{(n/2-k)^2}{(n/2-k)^2}\binom{n}{k} \le$$

$$\le \frac{1}{t^2}\sum_{k:|n/2-k|>t}\left(\frac{n}{2}-k\right)^2\binom{n}{k} \le \frac{1}{t^2}\sum_{k=0}^{n}\left(\frac{n}{2}-k\right)^2\binom{n}{k}. \tag{1.1.2}$$

Найдем сумму, стоящую в правой части неравенства (1.1.2). Легко видеть, что

$$\sum_{k=0}^{n}\binom{n}{k}\left(\frac{n}{2}-k\right)^2 = \sum_{k=0}^{n}\binom{n}{k}\left(\frac{n^2}{4}-nk+k^2\right) =$$

$$= \frac{n^2}{4}\sum_{k=0}^{n}\binom{n}{k} - \sum_{k=0}^{n}\binom{n}{k}(n-k)k. \tag{1.1.3}$$

Первая сумма в правой части (1.1.3) равна $n^2 2^{n-2}$. Найдем вторую сумму:

$$\sum_{k=0}^{n}(n-k)k\binom{n}{k} = \sum_{k=1}^{n-1}(n-k)k\binom{n}{k} = \sum_{k=1}^{n-1}\frac{(n-k)kn!}{(n-k)!k!} =$$
$$= \sum_{k=1}^{n-1}\frac{n(n-1)(n-2)!}{(n-k-1)!(k-1)!} = n(n-1)\sum_{k=0}^{n-2}\binom{n-2}{k} = n(n-1)2^{n-2}.$$

Из двух предыдущих неравенств следует, что

$$\sum_{k=0}^{n}\binom{n}{k}\left(\frac{n}{2}-k\right)^2 = n^2 2^{n-2} - n(n-1)2^{n-2} = n2^{n-2}.$$

Подставляя полученное равенство в правую часть (1.1.2) и полагая t равным $\sqrt{n\log_2 n}$, находим, что

$$\left|\bigcup_{k:|n/2-k|>\sqrt{n\log_2 n}}\mathbb{B}_k^n\right| \le \frac{n2^{n-2}}{n\log_2 n} = \frac{2^{n-2}}{\log_2 n} = o(2^n).$$

Таким образом, в сумме во всех слоях, номера которых отличаются от $\frac{n}{2}$ больше чем на $\sqrt{n\log_2 n}$, находится $o(2^n)$ булевых наборов длины n. Следовательно, почти все наборы лежат в узкой полосе, состоящей из $2\sqrt{n\log_2 n}$ средних слоев. Теорема доказана.

4. Кроме рассмотренного выше линейного порядка $\le$, на множестве наборов $\mathbb{B}^n$ существует естественный частичный порядок $\preceq$. Говорят, что набор $\boldsymbol{u}$ *не больше* набора $\boldsymbol{v}$ ($\boldsymbol{u}\preceq\boldsymbol{v}$), если $u_i \le v_i$ при всех $i = 1, 2, \ldots, n$. Если $\boldsymbol{u}\preceq\boldsymbol{v}$ и $\boldsymbol{u}\ne\boldsymbol{v}$, то говорят, что что набор $\boldsymbol{u}$ *строго меньше* набора $\boldsymbol{v}$ ($\boldsymbol{u}\prec\boldsymbol{v}$). Наборы $\boldsymbol{u}$ и $\boldsymbol{v}$ называются *сравнимыми*, если либо $\boldsymbol{u}\preceq\boldsymbol{v}$, либо $\boldsymbol{v}\preceq\boldsymbol{u}$. Если ни одно из этих отношений не выполняется, то наборы называются *несравнимыми*. Последовательность вершин $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k$ называется *цепью*, если $d(\boldsymbol{u}_i, \boldsymbol{u}_{i+1}) = 1$ и $\boldsymbol{u}_i \preceq \boldsymbol{u}_{i+1}$ для всех $i = 1, 2, \ldots, k-1$. Вершина $\boldsymbol{u}_k$ называется наибольшей вершиной цепи $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_k$, а вершина $\boldsymbol{u}_1$ — наименьшей вершиной этой цепи. Число вершин в цепи называется ее длиной. Говорят, что цепь связывает вершины $\boldsymbol{u}$ и $\boldsymbol{v}$ и проходит через вершину $\boldsymbol{w}$, если $\boldsymbol{u}$ и $\boldsymbol{v}$, являются, соответственно, первой и последней вершинами цепи, а $\boldsymbol{w}$ принадлежит этой цепи. Цепь называется *максимальной*, если она не является частью цепи большей длины. Множество попарно несравнимых вершин называется *антицепью*. Антицепь называется *максимальной*, если она не является подмножеством другой антицепи, состоящей из большего количества вершин.

Пример 1.1.4. В булевом кубе $\mathbb{B}^3$ (см. рис. 1.1.1) существует ровно шесть максимальных цепей. Длина каждой максимальной цепи равна четырем. Каждая максимальная цепь начинается в вершине (000), последовательно проходит через одну вершину первого и одну вершину второго слоя, и заканчивается в вершине (111). В $\mathbb{B}^3$ существуют: (*i*) две максимальные антицепи состоящие из трех вершин, это первый и второй слои; (*ii*) три максимальные антицепи состоящие из двух вершин, каждая такая антицепь состоит из одной вершины первого слоя и противоположной ей вершины второго слоя, например, $\{(100), (011)\}$; (*iii*) две максимальные антицепи состоящие из одной вершины, это вершины (000) и (111). □

Теорема 1.1.3. *Булев куб $\mathbb{B}^n$ можно покрыть $\binom{n}{\lfloor n/2\rfloor}$ непересекающимися цепями так, что число цепей длины $n-2p+1$, где $p = 0, 1, \ldots, \lfloor n/2\rfloor$, равно $\binom{n}{p} - \binom{n}{p-1}$.*

Доказательство. Теорему докажем индукцией по n. При $n = 1$ утверждение теоремы очевидно — куб $\mathbb{B}^1$ можно покрыть единственной цепью из двух вершин ($p = 0$).

Допустим, что теорема верна для $n = k$. В кубе $\mathbb{B}^{k+1}$ рассмотрим два подмножества B_0 и B_1, состоящие из всех наборов, у которых $(k+1)$-е разряды равны соответственно

нулю и единице. Каждое из множеств B_i изоморфно кубу $\mathbb{B}^k$, и поэтому в силу предположения индукции может быть покрыто непересекающимися цепями так, что что число цепей длины $k-2p+1$, где $p=0,1,\ldots,\lfloor k/2\rfloor$, равно $\binom{k}{p}-\binom{k}{p-1}$.

Рассмотрим одинаковые покрытия множеств B_1 и B_0, каждое из которых удовлетворяет условиям теоремы в кубе размерности k. Очевидно, что цепи этих покрытий не пересекаются и полностью покрывают куб $\mathbb{B}^{k+1}$. Пусть C_1 и C_0 — одинаковые цепи в в рассматриваемых покрытиях множеств B_1 и B_0, $\boldsymbol{v}_1$ и $\boldsymbol{v}_0$ — наибольшие элементы этих цепей. Очевидно, что $\boldsymbol{v}_0 \preceq \boldsymbol{v}_1$ и в наборах $\boldsymbol{v}_1$ и $\boldsymbol{v}_0$ все разряды кроме последнего совпадают.

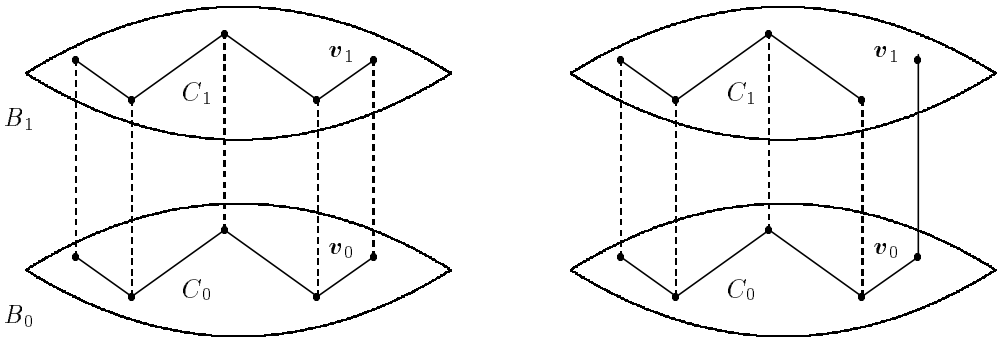


Рис. 1.1.2

Рассматриваемые объекты представлены в левой части рисунка 1.1.2. На этом рисунке штриховыми линиями изображены ребра, идущие в $(k+1)$-м направлении. Преобразуем цепи C_1 и C_0 следующим образом: набор $\boldsymbol{v}_1$ удалим из цепи C_1 и добавим к цепи C_0. В результате из двух цепей длины $k-2p+1$ получим цепь длины $k-2p=(k+1)-2p-1$ и цепь длины $k-2p+2=(k+1)-2(p-1)-1$. Новые цепи изображены в правой части рисунка 1.1.2. Выполним подобные преобразования над всеми парами одинаковых цепей в покрытиях B_1 и B_0.

Очевидно, что новые цепи покрывают $\mathbb{B}^{k+1}$ и не пересекаются. При этом цепь длины $(k+1)-2p-1$ получается либо удалением наибольшего набора из цепи длины $k-2(p-1)-1$, либо добавлением нового набора в цепь длины $k-2p-1$. Поэтому в преобразованном покрытии число цепей длины $(k+1)-2p-1$ равно

$$\left(\binom{k}{p}-\binom{k}{p-1}\right)+\left(\binom{k}{p-1}-\binom{k}{p-2}\right)=$$
$$=\left(\binom{k}{p}+\binom{k}{p-1}\right)-\left(\binom{k}{p-1}+\binom{k}{p-2}\right)=\binom{k+1}{p}-\binom{k+1}{p-1}.$$

Теорема доказана.

Так как цепь и антицепь не могут иметь более одного общего элемента, то из доказанной теоремы следует, что любая антицепь в кубе $\mathbb{B}^n$ состоит не более чем из $\binom{n}{\lfloor n/2\rfloor}$ наборов. Очевидно, что данная оценка не может быть усилена, так как слой $\mathbb{B}^n_{\lfloor n/2\rfloor}$ является антицепью и содержит ровно $\binom{n}{\lfloor n/2\rfloor}$ наборов. Информация о распределении вершин антицепи по слоям куба содержится в задаче 1.1.16.

Задачи

1.1.1. Найти максимально возможный периметр треугольника в $\mathbb{B}^n$.

1.1.2. Чему равно максимальное расстояние между наборами, принадлежащими:
a) $\mathbb{B}^n$; b) $\mathbb{B}^n_k$; c) $B_{n,k}$; d) $S_{n,k}$?

1.1.3. Найти среднее расстояние между наборами $\mathbb{B}^n$.

1.1.4. Найти число различных ребер в кубе $\mathbb{B}^n$.

1.1.5. Найти в $\mathbb{B}^n$: a) число различных k-мерных граней;
b) число различных k-мерных граней, проходящих через фиксированную вершину;
c) число всех граней.

1.1.6. Сколько вершин в среднем содержит одна грань n-мерного куба?

1.1.7. Найти число ребер, проходящих через вершины k-мерной грани n-мерного булева куба.

1.1.8. Найти число ребер, проходящих через вершины, лежащие в k-м слое n-мерного булева куба.

1.1.9. Найти: a) $\sum_{\boldsymbol{u}\in\mathbb{B}^n} \|\boldsymbol{u}\|$; b) $\sum_{\boldsymbol{u}\in\mathbb{B}^n} |\boldsymbol{u}|$; c) $\sum_{\boldsymbol{u}\in\mathbb{B}^n_k} |\boldsymbol{u}|$.

1.1.10. В n-мерном булевом кубе эллипсом с фокусами $\boldsymbol{\alpha}$ и $\boldsymbol{\beta}$ называется множество наборов, сумма расстояний от каждого из которых до $\boldsymbol{\alpha}$ и $\boldsymbol{\beta}$ равна фиксированному числу k. Сколько наборов содержится в эллипсе n-мерного булева куба?

1.1.11. Для любых сравнимых наборов $\boldsymbol{\alpha},\boldsymbol{\beta}\in\mathbb{B}^n$ интервалом с границами $\boldsymbol{\alpha}$ и $\boldsymbol{\beta}$ называется множество $I(\boldsymbol{\alpha},\boldsymbol{\beta}) = \{\boldsymbol{\gamma}\in\mathbb{B}^n \mid \boldsymbol{\alpha}\preceq\boldsymbol{\gamma}\preceq\boldsymbol{\beta}\}$. Показать, что любой интервал является гранью, а грань — интервалом.

1.1.12. Показать, что[1)]

$$\binom{n}{\lfloor n/2\rfloor} \asymp \frac{2^n}{\sqrt{n}}.$$

1.1.13. Найти в $\mathbb{B}^n$ число наборов несравнимых с данным набором $\boldsymbol{\alpha}$.

1.1.14. Найти число пар попарно несравнимых вершин в $\mathbb{B}^n$.

1.1.15. Найти: a) число различных максимальных цепей в $\mathbb{B}^n$;
b) число различных максимальных цепей, проходящих через фиксированную вершину k-го слоя куба $\mathbb{B}^n$.

1.1.16. Пусть T — антицепь в $\mathbb{B}^n$, $T_k = T\cap\mathbb{B}^n_k$. Показать, что

$$\sum_{k=0}^{n} |T_k| \Big/ \binom{n}{k} \le 1.$$

1.1.17. Показать, что в покрытии куба $\mathbb{B}^n$, построенном в доказательстве теоремы 1.1.3, все цепи длины $n-2p+1$ начинаются в p-м слое куба, а заканчиваются в его $(n-p)$-м слое.

1.2. Булевы функции

1. Функция $f(x_1,\ldots,x_n)$, отображающая $\mathbb{B}^n$ в $\mathbb{B}^1$, называется *n-местной булевой* функцией. Множество всех булевых функций обозначается через P_2, а множество всех булевых функций зависящих от n переменных — через $P_2(n)$. Каждая булева функция имеет конечную область определения, что позволяет полностью задать функцию f из $P_2(n)$, перечислив все наборы из $\mathbb{B}^n$ и указав значения f на этих наборах. В частности, булева функция $f(x_1,\ldots,x_n)$ может быть задана таблицей состоящей из 2^k строк, каждой из которых поставлен в соответствие булев набор длины k, и 2^{n-k} столбцов, каждому из которых поставлен в соответствие булев набор длины $n-k$. Параметр k принимает значения от 0 до n. В такой таблице (Таб. 1.2.1) значение функции f на наборе $(\sigma_1,\ldots,\sigma_k,\sigma_{k+1},\ldots,\sigma_n)$ помещается на пересечении строки, соответствующей набору $(\sigma_1,\ldots,\sigma_k)$, и столбца, соответствующего набору $(\sigma_{k+1},\ldots,\sigma_n)$. Если в таблице 1.2.1 параметр k равен нулю, то говорят, что функция задается вектором-строкой своих значений, а если $k=n$ — вектором-столбцом. Так как каждый элемент таблицы, задающей булеву функцию, равен либо нулю,

[1)] Выражение $f(n)\asymp g(n)$ означает существование таких констант a и b, что $f(n)\le ag(n)$ и $g(n)\le bf(n)$.

Таблица 1.2.1.

			0	0	...	σ_{k+1}	...	1	1	x_{k+1}
			0	0	...	σ_{k+2}	...	1	1	x_{k+2}
			...	...	...	...	...	...	...	...
			0	1	...	σ_n	...	0	1	x_n
x_1	...	x_k								
0	...	0				$\vdots$				
0	...	1				$\vdots$				
...	...	...				$\vdots$				
σ_1	...	σ_k	...	...	...	$f(\sigma)$				
...	...	...								
1	...	0								
1	...	1								

либо единице, то легко видеть, что число различных таблиц, и, соответственно, число различных булевых функций, зависящих от n переменных, равно 2^{2^n}. Число наборов из $\mathbb{B}^n$, на которых функция f принимает единичные значения, называется *весом* $\|f\|$ этой функции:

$$\|f\| = \sum_{\boldsymbol{u}\in\mathbb{B}^n} f(\boldsymbol{u}).$$

Пример 1.2.1. Зададим таблицей функцию $f(x_1,\ldots,x_5)$ равную единице только на тех наборах $\boldsymbol{\sigma} = (\sigma_1,\ldots,\sigma_5)$, номера которых являются простыми числами. Так как среди

Таблица 1.2.2.

		0	0	0	0	1	1	1	1	x_3
		0	0	1	1	0	0	1	1	x_4
		0	1	0	1	0	1	0	1	x_5
x_1	x_2									
0	0	0	1	1	1	0	1	0	1	
0	1	0	0	0	1	0	1	0	0	
1	0	0	1	0	1	0	0	0	1	
1	1	0	0	0	0	0	1	0	1	

целых положительных чисел, не превосходящих 31, простыми являются числа 1, 2, 3, 5, 7, 11, 13, 17, 19, 23, 29 и 31, то вес рассматриваемой функции f равен 12, и несложно поверить, что таблица 1.2.2 действительно задает функцию f. □

2. Подробно рассмотрим множества $P_2(1)$ и $P_2(2)$, состоящие из булевых функций, зависящих от одной и двух переменных. Первое множество состоит из четырех булевых функций. Вектор-столбцы этих функций, зависящих от переменной x, представлены в таблице 1.2.3. Первая и четвертая функции называются *тождественными константами*,

Таблица 1.2.3.

x	f_0	f_x	$f_\neg$	f_1
0	0	0	1	1
1	0	1	0	1

нулем и единицей, и обозначаются, соответственно, символами 0 и 1. Вторая функция называется *тождественной* и обозначается так же, как и ее аргумент, символом x. Третья функция называется *отрицанием* или *инверсией*. Каждую из перечисленных функций можно задать вектором длины два: $\binom{0}{0}$, $\binom{0}{1}$, $\binom{1}{0}$, $\binom{1}{1}$.

Теперь рассмотрим множество $P_2(2)$, состоящее из 16 функций, зависящих от переменных x и y. Среди этих функций две константы 0 и 1 и четыре функции x, $f_\neg(x)$, y, $f_\neg(y)$, каждая из которых зависит только от одной переменной. Векторы-столбцы семи из десяти оставшихся функций перечислены в таблице 1.2.4. Все эти функции имеют собственные названия. Первая функция $f_\&$ называется *конъюнкцией*. Эта функция часто также

Таблица 1.2.4.

$x\ y$	$f_\&$	$f_\vee$	$f_\oplus$	$f_\sim$	$f_\mid$	$f_\downarrow$	$f_\to$
0 0	0	0	0	1	1	1	1
0 1	0	1	1	0	1	0	1
1 0	0	1	1	0	1	0	0
1 1	1	1	0	1	0	0	1

называется умножением. Вторая функция называется *дизъюнкцией*. Нетрудно заметить, что $f_\&(x,y) = \min(x,y)$ и $f_\vee(x,y) = \max(x,y)$. Следующая функция $f_\oplus$ называется *суммой по модулю два*, иногда ее также называют исключающим или. Четвертая функция $f_\sim$ называется *эквивалентностью*, эта функция равна единице если значения ее аргументов совпадают. Пятая функция называется *штрихом Шеффера*, шестая — *стрелкой Пирса*, седьмая — *импликацией*. Три функции, не попавшие в таблицу 1.2.4, собственных названий не имеют.

3. Переменная x_i функции $f(x_1,\dots,x_n)$ называется *существенной*, если найдутся такие булевы постоянные $u_1,\dots,u_{i-1},u_{i+1},\dots,u_n$, что

$$f(u_1,\dots,u_{i-1},0,u_{i+1},\dots,u_n) \neq f(u_1,\dots,u_{i-1},1,u_{i+1},\dots,u_n)$$

Несущественная переменная называется также *фиктивной*. Нетрудно показать, что если функция f имеет фиктивную переменную, то $\|f\|$ — четное число. Действительно, пусть x_i — фиктивная переменная f. Тогда

$$\|f\| = \sum_{\boldsymbol{x}\in\mathbb{B}^n} f(x_1,\dots x_n) = \sum_{\boldsymbol{x}\in\mathbb{B}^n,x_i=0} f(x_1,\dots x_n) + \sum_{\boldsymbol{x}\in\mathbb{B}^n,x_i=1} f(x_1,\dots x_n) = 2\sum_{\boldsymbol{x}\in\mathbb{B}^n,x_i=0} f(x_1,\dots x_n).$$

Пример 1.2.2. Найдем число функций принадлежащих $P_2(n)$ и существенно зависящих от всех своих переменных. Искомое число обозначим через N. Пусть A_i — множество всех тех функций из $P_2(n)$ для которых переменная x_i не является существенной. Очевидно, что N равно $2^{2^n} - |\cup_{i=1}^n A_i|$. Мощность объединения множеств A_i вычислим при помощи формулы включений-исключений. Имеем

$$\Big|\bigcup_{i=1}^n A_i\Big| = \sum_{1\le i\le n}|A_i| - \sum_{1\le i_1<i_2\le n}|A_{i_1}\cap A_{i_2}| + \dots + (-1)^{k+1}\sum_{1\le i_1<\dots<i_k\le n}|A_{i_1}\cap\dots\cap A_{i_k}| + \dots + (-1)^{n+1}|A_1\cap\dots\cap A_n|.$$

Найдем мощность множества $A_1\cap\dots\cap A_k$. Это множество состоит из функций зависящих, не обязательно существенно, только от последних $n-k$ переменных $x_{k+1},\dots,x_n$. Поэтому, легко видеть, что мощность рассматриваемого множества равна числу функций в $P_2(n-k)$, т.е. $|A_1\cap\dots\cap A_k| = 2^{2^{n-k}}$. Очевидно, что мощность пересечения любых k множеств A_i также равна $2^{2^{n-k}}$. Учитывая, что k существенных переменных можно выбрать $\binom{n}{k}$ способами, видим, что

$$\Big|\bigcup_{i=1}^n A_i\Big| = \sum_{k=1}^n (-1)^{k+1}\binom{n}{k}2^{2^{n-k}}.$$

Следовательно,

$$N = 2^{2^n} - \sum_{k=1}^{n}(-1)^{k+1}\binom{n}{k}2^{2^{n-k}} = \sum_{k=0}^{n}(-1)^{k}\binom{n}{k}2^{2^{n-k}}.$$

Легко видеть, что $|\cup_{i=1}^{n} A_i| = o\left(2^{2^n}\right)$ при $n \to \infty$. Поэтому с ростом n почти все булевы функции из $P_2(n)$ существенно зависят от всех своих переменных. □

4. Определим несколько простейших преобразования булевых функций: подстановку констант, отождествление переменных, добавление и удаление фиктивных переменных.

Если для функций $f(x_1, \ldots, x_n)$ и $g(x_{k+1}, \ldots, x_n)$ при всех возможных значениях переменных $x_{k+1}, \ldots, x_n$ справедливо равенство

$$f(\alpha_1, \ldots, \alpha_k, x_{k+1}, \ldots, x_n) = g(x_{k+1}, \ldots, x_n),$$

то будем говорить, что функция g получена из функции f *подстановкой констант* $\alpha_1, \ldots, \alpha_k$ вместо переменных $x_1, \ldots, x_k$. Функцию g будем называть *подфункцией* функции f. Очевидным образом данное определение распространяется на случай подстановки констант $\alpha_1, \ldots, \alpha_k$ вместо произвольных переменных $x_{i_1}, \ldots, x_{i_k}$.

Пример 1.2.3. Найдем все возможные подфункции линейной функции $f_\oplus(x, y)$. Для этого подставим вместо первой переменной 0 и 1. В результате получим две функции одной переменной $f_\oplus(0, y)$ и $f_\oplus(1, y)$ значения которых легко находятся из таблицы 1.2.4:

$$f_\oplus(0,0) = 0, \qquad f_\oplus(1,0) = 1,$$
$$f_\oplus(0,1) = 1, \qquad f_\oplus(1,1) = 0.$$

Следовательно, $f_\oplus(0, y) = y$ и $f_\oplus(1, y) = f_\neg(y)$. Аналогичным образом подставляя константы вместо второй переменной получим функции x и $f_\neg(x)$. Очевидно, что подставляя константы одновременно вместо x и y, получим только константы 0 и 1. Таким образом, у функции $f_\oplus$ есть ровно шесть подфункций: 0, 1, x, y, $f_\neg(x)$ и $f_\neg(y)$. □

Если для функций $f(x_1, \ldots, x_n)$ и $g(x, x_{k+1}, \ldots, x_n)$ при всех возможных значениях переменных $x_{k+1}, \ldots, x_n$ справедливы равенства

$$f(0, \ldots, 0, x_{k+1}, \ldots, x_n) = g(0, x_{k+1}, \ldots, x_n),$$
$$f(1, \ldots, 1, x_{k+1}, \ldots, x_n) = g(1, x_{k+1}, \ldots, x_n),$$

то будем говорить, что функция g получена из функции f *отождествлением переменных* $x_1, \ldots, x_k$. Как и ранее, данное определение очевидным образом распространяется на случай отождествления произвольных переменных $x_{i_1}, \ldots, x_{i_k}$.

Пример 1.2.4. В булевой функции $f(x, y, z)$, заданной приводимой ниже таблицей, отождествим первую и третью переменные. Для этого в рассматриваемой таблице надо взять столбцы, в которых значения первой и третьей переменных совпадают. Такими столбцами будут[2)] первый, третий, шестой и восьмой. После этого из этих четырех столбцов составим новую таблицу. В каждом столбце из двух одинаковых компонент, первой и третьей, оставим только первую, поставив ее в соответствие новой переменной t. Таблица значений получившейся функции $g(t, y)$ расположена справа внизу.

x	0	0	0	0	1	1	1	1
y	0	0	1	1	0	0	1	1
z	0	1	0	1	0	1	0	1
f	0	1	1	1	1	1	0	0

t	0	0	1	1
y	0	1	0	1
g	0	1	1	0

Легко видеть, что отождествив у функции f первую и третью переменные, получили сумму по модулю два переменной y и новой переменной t. □

[2)] Здесь полагаем, что символы x, y, z и f располагаются в нулевом столбце таблицы.

Рассмотрим булеву функцию $f(x_1,\dots,x_n)$ с фиктивными переменными $x_1,\dots,x_k$ и булеву функцию $g(x_{k+1},\dots,x_n)$ получающуюся из f подстановкой констант $\alpha_1,\dots,\alpha_k$ вместо первых k переменных. Так как эти переменные у функции f фиктивные, то результат подстановки будет один и тот же для любых $\alpha_1,\dots,\alpha_k$. Поэтому можно полагать, что

$$g(x_{k+1},\dots,x_n)=f(0,\dots,0,x_{k+1},\dots,x_n).$$

Будем говорить, что функция g получена из функции f удалением фиктивных переменных $x_1,\dots,x_k$, а функция f получена из функции g добавлением фиктивных переменных $x_1,\dots,x_k$. Как и ранее, данное определение очевидным образом распространяется на случай удаления (добавления) произвольных фиктивных переменных $x_{i_1},\dots,x_{i_k}$.

Булевы функции f и g называются *равными*, если функция f получена из функции g удалением и добавлением фиктивных переменных.

5. Булева функция $f(x_1,\dots,x_n)$ называется *симметрической относительно переменных* x_i *и* x_j, если для любых $\alpha_1,\dots,\alpha_n$ справедливо равенство

$$f(\alpha_1,\dots,\alpha_i,\dots,\alpha_j,\dots,\alpha_n)=f(\alpha_1,\dots,\alpha_j,\dots,\alpha_i,\dots,\alpha_n),$$

т.е. значение функции не меняется при перестановке ее i-го и j-го аргументов.

Булева функция $f(x_1,\dots,x_n)$ называется *симметрической относительно переменных* $x_{i_1},\dots,x_{i_j}$, если она симметрическая относительно любых двух переменных из множества $\{x_{i_1},\dots,x_{i_j}\}$. Функция, симметрическая относительно всех своих переменных, называется *симметрической*. Все функции, перечисленные в таблице 1.2.4 кроме импликации, являются симметрическими. Из определения симметрической функции легко следует, что значение каждой такой функции на любом наборе $\boldsymbol{x}$ однозначно определяется весом этого набора.

Среди симметрических функций выделим симметрические пороговые функции, которые естественным образом часто возникают в различных задачах. *Симметрической пороговой функцией n переменных с порогом m* называется такая булева функция $\tau_m(x_1,\dots,x_n)$, что

$$\tau_m(x_1,\dots,x_n)=\begin{cases}1, & \text{если } \sum_{i=1}^n x_i\geq m,\\ 0, & \text{если } \sum_{i=1}^n x_i<m.\end{cases}$$

Легко видеть, что дизъюнкция и конъюнкция являются симметрическими функциями двух переменных с порогами 1 и 2 соответственно, т.е. $f_\vee(x_1,x_2)=\tau_1(x_1,x_2)$ и $f_\&(x_1,x_2)=\tau_2(x_1,x_2)$. Среди симметрических пороговых функций особое место занимают функции $\tau_n(x_1,\dots,x_{2n-1})$, $n\geq 2$, называемые функциями *голосования*, или функциями большинства. В таблице 1.2.5 определена функция голосования, зависящая от трех

Таблица 1.2.5.

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
τ_2	0	0	0	1	0	1	1	1

аргументов.

6. Пусть $D\subseteq\mathbb{B}^n$. Функция $f(x_1,\dots,x_n)$, определенная на области D и принимающая значения 0 и 1, называется *частичной булевой функцией*. Если набор $\boldsymbol{x}\in\mathbb{B}^n$ и $\notin D$, то будем говорить, что функция f не определена на этом наборе, или, что f принимает на нем неопределенное значение "$*$".

Как и обычную булеву функцию, частичную булеву функцию можно задать таблицей из 2^n строк. Однако в таблице значений частичной функции в последнем столбце булевы величины будут стоять только в $|D|$ строках, соответствующих наборам из области определения частичной функции. В остальных местах будут стоять символы $*$. Очевидно, что на области D можно определить $2^{|D|}$ различных частичных функций.

Доопределением частичной булевой функции $f(x_1, \ldots, x_n)$, определенной на области $D \subseteq \mathbb{B}^n$, называется такая булева функция $\hat{f} : \mathbb{B}^n \to \mathbb{B}$, что $\hat{f}(\boldsymbol{x}) = f(\boldsymbol{x})$ для любого $\boldsymbol{x}$ из D.

Доопределение частичной функции f можно получить заменив в таблице значений f каждый символ $*$ нулем или единицей. Поэтому легко видеть, что доопределение частичной функции не единственно. У каждой частичной функции, определенной на области D, есть $2^{2^n-|D|}$ различных доопределений.

Пример 1.2.5. Ниже приведены две частичные функции f_1 и f_2 и все их доопределения. Первая функция определена на двух наборах из четырех возможных, и поэтому имеет четыре доопределения, вторая функция определена на трех наборах и у нее два доопределения.

x	y
0	0
0	1
1	0
1	1

f_1	y	$f_\vee$	$f_\to$	$\mathbf{1}$
$*$	0	0	1	1
1	1	1	1	1
$*$	0	1	0	1
1	1	1	1	1

f_2	$f_\oplus$	$f_\mid$
$*$	0	1
1	1	1
1	1	1
0	0	0

□

7. Система булевых функций $f = \{f_1, \ldots, f_m\}$, зависящих от переменных $x_1, \ldots, x_n$, называется *булевым (m,n)-оператором*. Функция f_i называется *i-й компонентой* оператора f или его *i-й координатной функцией*. Каждый булев (m,n)-оператор задает некоторое отображение $\mathbb{B}^n$ в $\mathbb{B}^m$. Как и булеву функцию, булев оператор можно задать таблицей, содержащей все его значения. Так как значением (m,n)-оператора является двоичный набор длины m, то нетрудно заметить, что существует ровно 2^{m2^n} различных булевых (m,n)-операторов.

Пример 1.2.6. При помощи таблиц зададим два естественных оператора. Первый оператор S вычисляет сумму $\boldsymbol{s} = s_3 4 + s_2 2 + s_1$ двух двоичных двухразрядных чисел $\boldsymbol{x} = x_2 2 + x_1$ и $\boldsymbol{y} = y_2 2 + y_1$. Аргументами этого оператора являются величины x_1, x_2 и

Таблица 1.2.6.

x_2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
y_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
y_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
s_3	0	0	0	0	0	0	0	1	0	0	1	1	0	1	1	1
s_2	0	0	1	1	0	1	1	0	1	1	0	0	1	0	0	1
s_1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

y_1, y_2, а значениями — величины s_1, s_2 и s_3, т.е. $S(x_1, x_2, y_1, y_2) = (s_1, s_2, s_3)$. Этот оператор задается таблицей 1.2.6. Второй оператор R вычисляет разность $\boldsymbol{x} - \boldsymbol{y}$ двух двоичных

Таблица 1.2.7.

x_2	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
x_1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
y_2	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
y_1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
r_3	0	1	1	1	0	0	1	1	0	0	0	1	0	0	0	0
r_2	0	0	1	1	0	0	0	1	1	0	0	0	1	1	0	0
r_1	0	1	0	1	1	0	1	0	0	1	0	1	1	0	1	0

двухразрядных чисел $\boldsymbol{x} = x_2 2 + x_1$ и $\boldsymbol{y} = y_2 2 + y_1$. Разность представляется в виде трех булевых величин r_1, r_2 и r_3 так, что число $2r_2 + r_1$ равно модулю разности $\boldsymbol{x}$ и $\boldsymbol{y}$, т. е. $2r_2 + r_2 = |\boldsymbol{x} - \boldsymbol{y}|$, а булева величина r_3 определяет знак разности — $r_3 = 0$ если $\boldsymbol{x} \geq \boldsymbol{y}$, и $r_3 = 1$ если $\boldsymbol{x} < \boldsymbol{y}$. Оператор R задается таблицей 1.2.7. □

Задачи

1.2.1. Найти число различных булевых функций, зависящих от n аргументов и имеющих вес равный $1, 2, 3, \lceil n/2 \rceil$.

1.2.2. Найти число различных булевых функций, зависящих от n аргументов и имеющих четный вес.

1.2.3. Определить существенные и фиктивные переменные у функции, заданной вектором значений:
a) (11110110 11110110); b) (01010111 11010010); c) (10010111 11111011).

1.2.4. Пусть булева функция $f(x_1, \ldots, x_n)$ задана вектором значений $(f_0, \ldots, f_{2^n-1})$. Доказать, что если x_i является фиктивной переменной, то $f_j = f_{2^{n-i}+j}$ для всех целых j, принадлежащих множеству $\{k\,2^{n-i+1}, \ldots, (2k+1)\,2^{n-i+1} - 1\}$, где $k = 0, 1, \ldots 2^{i-1} - 1$.

1.2.5. Пусть $f(x_1, \ldots, x_n)$ существенно зависит ровно от k переменных. Показать, что $\|f\|$ делится на 2^{n-k}.

1.2.6. Найти в $P_2(n)$ число различных симметрических булевых функций.

1.2.7. Найти в $P_2(n)$ число различных булевых функций, симметрических относительно первых k аргументов.

1.2.8. Указать взаимно однозначное соответствие между множеством булевых функций от трех переменных и множеством симметрических функций от семи переменных.

1.2.9. Показать, что любая булева функция от трех переменных может быть получена из симметрической функции от семи переменных отождествлением переменных.

1.2.10. Показать, что любая булева функция может быть получена из симметрической функции отождествлением переменных.

1.2.11. Показать, что любая симметрическая функция отличная от константы существенно зависит от всех своих переменных.

1.2.12. Найти число неравных подфункций у функции, заданной вектором значений:
a) (00010110 10010000); b) (11010111 01010010); c) (10010111 11111000).

1.2.13. Какое минимальное число неравных подфункций может быть у n-местной булевой функции, существенно зависящей от всех своих аргументов?

1.2.14. Булев (n, n)-оператор называется взаимнооднозначным, если $f(\boldsymbol{x}) \neq f(\boldsymbol{y})$ для любых неравных $\boldsymbol{x}$ и $\boldsymbol{y}$ из $\mathbb{B}^n$. Найти число взаимнооднозначных булевых (n, n)-операторов.

1.2.15. Найти число булевых (m, n)-операторов таких, что $\|f(\boldsymbol{x})\| = 2$ для каждого $\boldsymbol{x} \in \mathbb{B}^n$.

1.3. Формулы. Реализация булевых функций формулами

Выше булевы функции задавались перечислением своих значений на всей области определения. При таком задании все функции, зависящие от одного и того же числа переменных, оказываются одинаково сложными — для определения функции n переменных требуется таблица из 2^n строк. В настоящем разделе рассматривается аналитический способ задания булевых функций посредством формул. Формульное представление булевых функций не только упрощает задание многих практически важных булевых функций, но и значительно облегчает различные действия с ними.

1. Пусть $X_n = \{x_1, x_2, \ldots, x_n\}$ — множество булевых переменных, B — подмножество P_2. Выражение F, составленное из символов переменных из X_n и из символов функций

из B называется *булевой формулой* в *базисе* B над множеством переменных X_n, если F удовлетворяет следующему индуктивному определению:

1. Переменная x_i является формулой ($i \in \{1, 2, \ldots, n\}$);

2. Если f — k-местная функция из B и $\mathsf{F}_1, \ldots, \mathsf{F}_k$ — формулы, то выражение

$$f(\mathsf{F}_1, \ldots, \mathsf{F}_k) \tag{1.3.1}$$

так же является формулой. Формулы $\mathsf{F}_1, \ldots, \mathsf{F}_k$ называются *подформулами* формулы (1.3.1), а функция f — внешней функцией этой формулы. Любая подформула каждой формулы F_i так же называется подформулой формулы F.

Индуктивно определим значение формулы F на наборе переменных $x_1, x_2, \ldots, x_n$:

1. Если $\mathsf{F} = x_i$, то $\mathsf{F}(x_1, \ldots, x_n) = x_i$;

2. Пусть $f \in B$, $\mathsf{F} = f(\mathsf{F}_1, \ldots, \mathsf{F}_k)$ и значения формул $\mathsf{F}_1, \ldots, \mathsf{F}_k$ на переменных $x_1, \ldots, x_n$ определены. Тогда

$$\mathsf{F}(x_1, \ldots, x_n) = f(\mathsf{F}_1(x_1, \ldots, x_n), \ldots, \mathsf{F}_k(x_1, \ldots, x_n)).$$

Булева формула F над множеством переменных $x_1, \ldots, x_n$ *реализует* булеву функцию $f(x_1, \ldots, x_n)$, если

$$\mathsf{F}(x_1, \ldots, x_n) = f(x_1, \ldots, x_n)$$

при все наборах $(x_1, \ldots x_n)$ из $\mathbb{B}^n$.

Пусть формула F, реализующая функцию $f(x_1, \ldots, x_n)$, составлена из символов переменных $x_1, \ldots, x_n$ и символов функций $f_1, \ldots, f_m$. Тогда говорят, что формула F и функция f являются *суперпозициями* функций $f_1, \ldots, f_m$. Далее формулы и реализуемые ими булевы функции будем обозначать одними и теми же символами в тех случаях, когда это не будет приводить к неоднозначному пониманию.

Пример 1.3.1. Рассмотрим формулу $\mathsf{F} = f_\neg(f_\&(f_\neg(x), f_\neg(y)))$ над множеством из двух переменных x и y. В этой формуле вместо переменных x и y подставим различные булевы постоянные:

$$\begin{aligned}
\mathsf{F}(0,0) &= f_\neg(f_\&(f_\neg(0), f_\neg(0))) = f_\neg(f_\&(1,1)) = f_\neg(1) = 0,\\
\mathsf{F}(0,1) &= f_\neg(f_\&(f_\neg(0), f_\neg(1))) = f_\neg(f_\&(1,0)) = f_\neg(0) = 1,\\
\mathsf{F}(1,0) &= f_\neg(f_\&(f_\neg(1), f_\neg(0))) = f_\neg(f_\&(0,1)) = f_\neg(0) = 1,\\
\mathsf{F}(1,1) &= f_\neg(f_\&(f_\neg(1), f_\neg(1))) = f_\neg(f_\&(0,0)) = f_\neg(0) = 1.
\end{aligned}$$

Сравнивая полученные значения со значениями функций из таблицы 1.2.4 видим, что формула $f_\neg(f_\&(f_\neg(x), f_\neg(y)))$ реализует дизъюнкцию переменных x и y. □

2. Важными характеристиками любой формулы являются ее сложность и глубина[3)]. *Сложностью* $l(\mathsf{F})$ формулы F в базисе B называется число символов из B входящих в F. *Глубину* $d(\mathsf{F})$ формулы F определим индуктивно:

1. Если $\mathsf{F} = x_i$, то $d(\mathsf{F}) = 0$;

2. Если $\mathsf{F} = f(\mathsf{F}_1, \ldots, \mathsf{F}_k)$, где $f \in B$, то

$$d(\mathsf{F}) = 1 + \max_{1 \le i \le k} d(\mathsf{F}_i).$$

Из определений сложности и глубины формул в частности следует, что переменная всегда является формулой нулевой сложности и нулевой глубины, а сложность и глубина любой функции, принадлежащей базису формулы, равны единице. Из этих определений так же легко следует, что сложность и глубина любой формулы строго больше сложности и глубины каждой ее подформулы.

[3)] Существуют различные определения глубины и сложности формул. Так иногда сложностью формулы называют число символов переменных, входящих в формулу, а при определении глубины не учитывают отрицания.

Пример 1.3.2. Рассмотрим две двухместные булевы функции f_1 и f_2. Составленные из символов этих функций и символов переменных x, y и z выражения

$$\begin{aligned}
&\mathsf{F}_1 = f_1(x,y),\\
&\mathsf{F}_2 = f_2(\mathsf{F}_1,z) = f_2(f_1(x,y),z),\\
&\mathsf{F}_3 = f_2(\mathsf{F}_2,\mathsf{F}_1) = f_2(f_2(f_1(x,y),z),f_1(x,y)),
\end{aligned}$$

являются формулами над множеством переменных $\{x,y,z\}$ в базисе $\{f_1,f_2\}$. Нетрудно видеть, что сложность и глубина формулы F_1 равны единице, сложность и глубина формулы F_2 равны двум, а сложность и глубина формулы F_3 равны, соответственно, четырем и трем. □

Часто бывает удобно представлять формулы в виде помеченных корневых деревьев, в которых висячие вершины соответствуют переменным, а внутренние вершины — функциям. Такие деревья легко строятся в соответствии с индуктивным определением формул. Если формула состоит из одной переменной, то соответствующее дерево состоит из одной вершины, помеченной символом этой переменной. Если формулам $\mathsf{F}_1,\ldots,\mathsf{F}_k$ соответствуют деревья $\mathsf{D}_1,\ldots,\mathsf{D}_k$, то дерево D, соответствующее формуле $f(\mathsf{F}_1,\ldots,\mathsf{F}_k)$, получается из деревьев $\mathsf{D}_1,\ldots,\mathsf{D}_k$ следующим образом. К этим деревьям добавляется новая вершина, которая соединяется ребрами с корнями деревьев $\mathsf{D}_1,\ldots,\mathsf{D}_k$ и помечается символом f. Новая вершина будет корнем дерева, соответствующего формуле $f(\mathsf{F}_1,\ldots,\mathsf{F}_k)$. При таком построении дерева число его внутренних вершин всегда будет равно сложности формулы, а число ребер в самой длинной цепи среди цепей, связывающих корень с висячими вершинами, будет равно глубине формулы. Так формулам F_1, F_2 и F_3 из рассмотренного

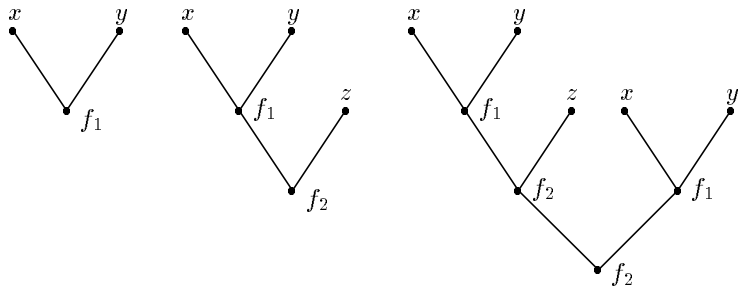


Рис. 1.3.1

выше примера, соответствуют деревья, изображенные на рисунке 1.3.1. Первое из этих деревьев содержит одну внутреннюю вершину, второе — две, третье — четыре. Глубины этих деревьев равны соответственно единице, двойке и тройке.

Если базис формулы состоит из конечного числа функций, то ее сложность и глубина связаны простым неравенством, установленном в следующей теореме.

Теорема 1.3.1. *Пусть* F *— формула в базисе* $B = \{f_1,\ldots,f_m\}$, k *— максимальное число аргументов у функций из* B. *Тогда*

$$d(\mathsf{F}) \geq \lceil \log_k((k-1)l(\mathsf{F})+1)\rceil.$$

Доказательство. Теорему докажем индукцией по глубине формулы. Для формул нулевой глубины неравенство теоремы очевидно. Допустим, что теорема верна для любой формулы глубины r. Пусть $\mathsf{F} = f(\mathsf{F}_1,\ldots,\mathsf{F}_s)$ — формула глубины $r+1$. Без ограничения общности будем полагать, что сложность подформулы F_1 не меньше сложности любой другой подформулы. Так как $s \leq k$, то очевидно, что сложность F и сложность подформулы F_1 связаны неравенством

$$l(\mathsf{F}) \leq kl(\mathsf{F}_1)+1.$$

Из этого неравенства легко получаем

$$(k-1)l(\mathsf{F})+1 \leq (k-1)(kl(\mathsf{F}_1)+1)+1 = k((k-1)l(\mathsf{F}_1)+1). \qquad (1.3.2)$$

По предположению индукции для подформулы F_1 выполняется неравенство

$$d(\mathsf{F}_1) \geq \lceil \log_k((k-1)l(\mathsf{F}_1)+1) \rceil.$$

Подставив последнее неравенство в (1.3.2) и прологарифмировав результат по основанию k получим

$$\lceil \log_k((k-1)l(\mathsf{F})+1) \rceil \leq \lceil 1+\log_k((k-1)l(\mathsf{F}_1)+1) \rceil \leq 1+d(\mathsf{F}_1) \leq d(\mathsf{F}).$$

Теорема доказана.

3. Пусть B — произвольный базис, состоящий из конечного числа функций. Оценим сверху число различных булевых формул над множеством переменных $x_1, \ldots, x_n$ в базисе B, при условии, что сложность рассматриваемых формул не превосходит заданной величины. Имеет место следующий результат.

Теорема 1.3.2. *Пусть m — константа, $B = \{f_1, \ldots, f_m\}$, k — максимальное число аргументов у функций из B. Тогда для $N(L,n,k)$ — числа различных формул в базисе B над множеством переменных $x_1, \ldots, x_n$ сложности не более L, справедливо неравенство*

$$N(L,n,k) \leq (c \cdot n)^{(k-1)L+1},$$

где c — константа, зависящая от базиса.

Доказательство. Прежде всего оценим величину $M(\mathsf{F})$, равную числу переменных, входящих в произвольную формулу F. Индукцией по глубине формулы покажем, что

$$M(\mathsf{F}) \leq (k-1)l(\mathsf{F})+1. \tag{1.3.3}$$

Очевидно, что неравенство (1.3.3) справедливо для любой формулы нулевой глубины. Предположим, что (1.3.3) справедливо для формул глубины r. Пусть $\mathsf{F} = f(\mathsf{F}_1, \ldots, \mathsf{F}_s)$ — формула глубины $r+1$. Тогда

$$M(\mathsf{F}) = \sum_{i=1}^{s} M(\mathsf{F}_i) \leq \sum_{i=1}^{s}((k-1)l(\mathsf{F}_i)+1) \leq$$
$$\leq (k-1)\left(\sum_{i=1}^{s} l(\mathsf{F}_i)+1\right)+1 = (k-1)l(\mathsf{F})+1.$$

Неравенство (1.3.3) доказано.

Из этого неравенства и определения формулы следует, что каждая формула F содержит $l(\mathsf{F})$ символов функций f_j, $l(\mathsf{F})$ правых скобок, $l(\mathsf{F})$ левых скобок и не более чем $(k-1)l(\mathsf{F})+1$ символов переменных x_j. Отметим, что все запятые, отделяющие в формулах аргументы друг от друга, можно безболезненно опустить. Поэтому общее число символов в формуле F не превосходит $(k+2)l(\mathsf{F})+1$. Таким образом, для числа различных формул сложности не более чем L справедлива следующая верхняя оценка

$$N(L,n,k) \leq \sum_{i=0}^{L} \binom{(k+2)i+1}{(k-1)i+1} \binom{3i}{i} \binom{2i}{i} n^{(k-1)i+1} m^i \leq$$
$$\leq \sum_{i=0}^{L} \left(2^{k+7}m\right)^i n^{(k-1)i+1} \leq (cn)^{(k-1)L+1},$$

где $c = \left(2^{k+7}m\right)^{\frac{1}{k-1}}$. Так как базис B состоит из конечного числа функций, то c является константой. Теорема доказана

Базис B называется *полным*, если любая функция из P_2 может быть реализована формулой в этом базисе. Вопросы полноты различных базисов рассматриваются далее в параграфе 1.6. Здесь мы покажем, что в любом полном конечном базисе с ростом n большинство функций из $P_2(n)$ реализуются только очень сложными формулами.

Теорема 1.3.3. *Пусть B — полный конечный базис, k — максимальное число аргументов у функций из B. Тогда для любой постоянной $\varepsilon > 0$ доля функций из $P_2(n)$ реализуемых формулами, сложность которых не превосходит $\frac{(1-\varepsilon)2^n}{(k-1)\log_2 n}$, стремится к нулю при $n \to \infty$.*

Доказательство. Так как неравные функции реализуются неравными формулами, то число функций, реализуемых формулами сложности не более L, не больше чем количество таких формул. Поэтому для доказательства теоремы достаточно показать, что число $N(L,n,k)$ различных формул сложности не более $L = \frac{(1-\varepsilon)2^n}{(k-1)\log_2 n}$ при возрастании n есть $o\left(2^{2^n}\right)$. Оценим логарифм $N(L,n,k)$. Используя оценку величины $N(L,n,k)$ из предыдущей теоремы при $n \to \infty$ имеем

$$\log_2 N(L,n,k) = \frac{(k-1)(1-\varepsilon)2^n}{(k-1)\log_2 n} \cdot \log_2(cn) = (1-\varepsilon)2^n\left(1 + \frac{c}{\log_2 n}\right) < \left(1 - \frac{\varepsilon}{2}\right)2^n.$$

Следовательно,

$$N(L,n,k) = 2^{\log_2 N(L,n,k)} = 2^{2^n(1-\varepsilon/2)} = o\left(2^{2^n}\right).$$

Теорема доказана.

В теореме 1.3.3 утверждается, что почти все булевы функции, зависящие от n переменных, реализуются очень сложными формулами. Иногда подобные утверждения удобно формулировать не для всего множества в целом, используя стандартное понятие "почти все", а для его отдельных элементов, говоря о "почти каждом" элементе множества. Пусть $\{A_n\}_{n=1}^{\infty}$ — последовательность конечных множеств возрастающей мощности. Будем говорить, что *почти каждый* элемент множества A_n обладает свойством $\mathcal{B}$, если при $n \to \infty$ отношение числа элементов множества A_n обладающих свойством $\mathcal{B}$ к мощности A_n стремится к единице.

Используя введенное понятие, переформулируем теорему 1.3.3 следующим образом: *при $n \to \infty$ почти каждая булева функция из $P_2(n)$ не может быть реализована в базисе B формулой, сложность которой асимптотически меньше чем $\frac{2^n}{(k-1)\log_2 n}$, где k — максимальное число аргументов у функций из B.*

4. Если базис B состоит только из двухместных и одноместных функций, то двухместные формулы $\mathsf{F}(x,y)$ будем записывать при помощи символов–связок в виде $(x \circ y)$, где $\circ$ — символ двухместной булевой функции, реализуемой формулой $\mathsf{F}(x,y)$. Наиболее часто встречающиеся символы двухместных булевых функций использованы в таблице 1.2.4 в качестве нижних индексов у символов соответствующих функций f. Так для обозначения конъюнкции чаще всего используется символ &, т.е. $f_{\&}(x,y) = (x\&y)$. Иногда конъюнкция обозначается так же через $\wedge$ и $\cdot$, или функциональный символ опускается. Формулы для других двухместных булевых функций, перечисленных в таблице 1.2.4, записываются следующим образом:

$$f_{\vee}(x,y) = (x \vee y), \qquad f_{\oplus}(x,y) = (x \oplus y), \qquad f_{\sim}(x,y) = (x \sim y),$$
$$f_{|}(x,y) = (x \,|\, y), \qquad f_{\downarrow}(x,y) = (x \downarrow y), \qquad f_{\to}(x,y) = (x \to y).$$

Для эквивалентности вместо символа $\sim$ иногда используется символ $\equiv$. Одноместную формулу, реализующую функцию отрицания, будем записывать при помощи горизонтальной черты покрывающей аргумент: $\mathsf{F}_{\neg}(x) = (\overline{x})$. Используя символы–связки можно записывать более сложные формулы. Например:

$$f_{\neg}(f_{\vee}(f_{\&}(x_1,x_2), f_{\&}(x_3,x_4))) = (\overline{(f_{\vee}(f_{\&}(x_1,x_2), f_{\&}(x_3,x_4)))} =$$
$$= (\overline{(f_{\&}(x_1,x_2) \vee f_{\&}(x_3,x_4))}) = (\overline{((x_1\&x_2) \vee (x_3\&x_4))}). \qquad (1.3.4)$$

Далее для упрощения записи сложных формул в некоторых случаях будем опускать скобки.

1. Полагая, что функция отрицания "сильнее" всех остальных функций, будем опускать скобки вокруг аргумента отрицания отрицания. Таким образом, если в формуле отсутствуют скобки, то сначала выполняется отрицание. Например, $\overline{(x_1 \to x_2)} = \overline{x_1 \to x_2}$.

2. Полагая, что функция & "сильнее" всех остальных двуместных функций, будем опускать скобки вокруг конъюнкции. Например, $(x_1\&x_2)\oplus x_3 = x_1\&x_2\oplus x_3 = x_1x_2\oplus x_3$.

3. Во всех формулах будем опускать внешние скобки. В этом случае формула (1.3.4) будет выглядеть так (с учетом предыдущих правил): $\overline{x_1x_2\vee x_3x_4}$.

4. Легко видеть, что для дизъюнкции справедливо равенство $(x_1\vee x_2)\vee x_3 = (x_1\vee x_2)\vee x_3$. Аналогичные равенства имеют место также для конъюнкции и суммы по модулю два. Поэтому будем опускать скобки, если одна из функций &, $\vee$ или $\oplus$ используется в формуле несколько раз подряд. Например, $(x_1\vee x_2)\vee x_3 = x_1\vee x_2\vee x_3$.

5. Булевы формулы F_1 и F_2 называются *эквивалентными*, если они реализуют одну и ту же булеву функцию. Замена формулы F_1 на эквивалентную ей формулу F_2 называется *эквивалентным преобразованием* формулы F_1. Заметим, что любое эквивалентное преобразование формул устанавливает равенство реализуемых этими формулами функций.

Приведем ряд соотношений, определяющих простейшие эквивалентные преобразования булевых формул в двухместных базисах. Из таблицы 1.2.3 следуют равенства

$$\overline{0} = 1, \quad \overline{1} = 0, \quad \overline{\overline{x}} = x,$$

последнее из которых называется *правилом двойного отрицания*. Справедливость приводимых далее равенств для формул над множеством из одной переменной x для дизъюнкции, конъюнкции, суммы, эквивалентности, отрицания и констант легко следует из таблицы 1.2.4:

$$\begin{array}{llll} x\vee x = x, & x\&x = x, & x\oplus x = 0, & x\sim x = 1,\\ x\vee\overline{x} = 1, & x\&\overline{x} = 0, & x\oplus\overline{x} = 1, & x\sim\overline{x} = 0,\\ x\vee 0 = x, & x\&0 = 0, & x\oplus 0 = x, & x\sim 0 = \overline{x},\\ x\vee 1 = 1, & x\&1 = x, & x\oplus 1 = \overline{x}, & x\sim 1 = x. \end{array} \tag{1.3.5}$$

Используя таблицу 1.2.4 нетрудно убедиться в эквивалентности различных формул над множеством из двух переменных. В частности справедливы соотношения

$$x\&y = \overline{\overline{x}\vee\overline{y}}, \quad x\vee y = \overline{\overline{x}\&\overline{y}}, \tag{1.3.6}$$

называемые *законами двойственности* или *законами де Моргана*. Из таблицы 1.2.4 также легко видеть, что

$$x\sim y = \overline{x\oplus y} = x\oplus y\oplus 1. \tag{1.3.7}$$

Подставляя в правые и левые части равенств (1.3.8) вместо переменных x, y и z различные булевы постоянные, видим, что конъюнкция связана с дизъюнкцией и сложением по модулю два законами дистрибутивности:

$$\begin{array}{l} (x\vee y)\&z = (x\&z)\vee(y\&z),\\ (x\&y)\vee z = (x\vee z)\&(y\vee z),\\ (x\oplus y)\&z = (x\&z)\oplus(y\&z). \end{array} \tag{1.3.8}$$

Так же легко подстановкой констант устанавливается справедливость следующих равенств:

$$x\oplus y = x\overline{y}\vee\overline{x}y, \qquad x\vee y = xy\oplus x\oplus y. \tag{1.3.9}$$

Правило двойного отрицания и соотношения (1.3.5)–(1.3.9) справедливы не только для переменных — они остаются верными и в случае, когда в (1.3.5)–(1.3.9) вместо переменных используются произвольные булевы функции. В частности, для любой булевой функции f справедливы равенства:

$$\begin{array}{lll} f\oplus f = 0, & f\vee\overline{f} = 1, & f\vee 1 = f,\\ f\oplus\overline{f} = 1, & f\,\&\overline{f} = 0, & f\,\&\,0 = 0. \end{array} \tag{1.3.10}$$

Пример 1.3.3. Преобразуем формулу $xy \vee xz \vee yz$ в эквивалентную формулу в базисе $\{\oplus, \&\}$. Для этого воспользуемся равенствами (1.3.8)—(1.3.10). Видим, что

$$\begin{aligned} xy \vee xz \vee yz &= (xyxz \oplus xy \oplus xz) \vee yz = \\ &= (xyz \oplus xy \oplus xz)yz \oplus (xyz \oplus xy \oplus xz) \oplus yz = \\ &= xyz \oplus xyz \oplus xyz \oplus xyz \oplus xy \oplus xz \oplus yz = xy \oplus xz \oplus yz. \end{aligned}$$

Подставляя в полученную формулу вместо переменных x, y и x булевы константы, видим, что значение формулы равно единице только тогда, когда среди ее аргументов есть хотя бы две единицы. Следовательно, формулы $xy \oplus xz \oplus yz$ и $xy \vee xz \vee yz$ реализуют функцию голосования $\tau_2(x, y, z)$. $\square$

Используя (1.3.5)–(1.3.10) можно получит новые полезные равенства, позволяющие производить эквивалентные преобразования формул. Например, для любых булевых функций f и g справедливо преобразование

$$f = f \cdot 1 = f \cdot (g \vee \overline{g}) = fg \vee f\overline{g},$$

называемое расщеплением. Обратное преобразование, переход от формулы $fg \vee f\overline{g}$ к формуле f, называется склеиванием. Следующая цепочка равенств задает преобразование, называемое поглощением:

$$f \vee fg = f \cdot 1 \vee fg = f \cdot (1 \vee g) = f \cdot 1 = f.$$

Наконец приведем еще одно преобразование, доказываемое при помощи расщепления и склеивания:

$$f \vee \overline{f}g = fg \vee f\overline{g} \vee \overline{f}g = fg \vee f\overline{g} \vee \overline{f}g \vee fg = f \vee g.$$

Другие полезные равенства, устанавливающие эквивалентность булевых формул, рассмотрим в следующем параграфе.

Задачи

1.3.1. Пусть $B = \{f_1, \ldots, f_m\}$. Оценить сверху число различных формул над множеством из n переменных в базисе B, при условии, что глубина формул не превосходит величины d.

1.3.2. Доказать формулу обобщенного склеивания: $xz \vee y\overline{z} \vee xy = xz \vee y\overline{z}$.

1.3.3. Определить существенные переменные у функции, заданной формулой:
a) $((x \to y) \vee z)(y \to x)x\overline{z}$; b) $(x \oplus y) \to z)(\overline{z \to y})$; c) $((x \vee y)(x \vee z) \to (\overline{x} \to yz))y$.

1.3.4. При помощи эквивалентных преобразований упростить формулы:
a) $xyz \vee xy\overline{z} \vee x\overline{y}z \vee \overline{x}yz$; b) $xyz \vee \overline{x}yz \vee x \vee yz \vee (x \oplus y)$; c) $x\overline{y}(x \to z) \sim \overline{z}$.

1.3.5. Показать, что любая формула в базисе $\{\vee, \&, \neg\}$ эквивалентна некоторой формуле в том же базисе, в которой отрицания появляются только над переменными.

1.3.6. Доказать, что:
a) $\overline{x}_1\&\overline{x}_2\& \ldots \&\overline{x}_n = \overline{x_1 \vee x_2 \vee \cdots \vee x_n}$;
b) $\overline{x}_1 \vee \overline{x}_2 \vee \cdots \vee \overline{x}_n = \overline{x_1\&x_2\& \ldots \&x_n}$.
c) $x_1 \vee x_2 \vee \cdots \vee x_n = (x_1 \oplus 1)(x_2 \oplus 1) \cdot \ldots \cdot (x_n \oplus 1) \oplus 1$.

1.3.7. Найти число попарно различных булевых функций, получающихся из функции $\bigvee_{1 \le i < j \le n} x_i x_j$ подстановкой констант вместо переменных $x_1, \ldots, x_n$.

1.3.8. На скольких наборах из $\mathbb{B}^n$ равна единице функция f:
a) $f(x_1, \ldots, x_n) = x_1 \vee \overline{x}_1x_2 \vee \overline{x}_1\overline{x}_2x_2 \vee \cdots \vee \overline{x}_1 \ldots \overline{x}_{n-1}x_n$;
b) $f(x_1, \ldots, x_n) = \bigvee_{1 \le i_1 < \ldots < i_k \le 2n} x_{i_1} \cdot \ldots \cdot x_{i_n}$.

1.3.9. Пусть $p(x_1, \ldots, x_n) = x_1x_2 \cdot \ldots \cdot x_k$. Найти $\|p(\boldsymbol{x})\|$.

1.3.10. На скольких наборах из $\mathbb{B}^n$ равна единице функция f:
a) $f(x_1, \ldots, x_n) = \bigoplus_{k=1}^{n} \bigoplus_{1 \le i_1 < \ldots < i_k \le n} x_{i_1} \cdot \ldots \cdot x_{i_k}$;
b) $f(x_1, \ldots, x_n) = \bigoplus_{k=1}^{n} x_1 \cdot \ldots \cdot x_k$;
c) $f(x_1, \ldots, x_n) = \bigoplus_{1 \le i_1 < \ldots < i_k \le n} x_{i_1} \cdot \ldots \cdot x_{i_k}$.

1.4. Специальные представления булевых функций

Ниже рассматриваются несколько стандартных способов представления булевых функций при помощи реализующих их формул, имеющих простую структуру.

1. Разложение функции по переменным. Для любого $x \in \mathbb{B}$ его *булевой степенью* называется функция[4)]

$$x^{(\sigma)} = \begin{cases} \overline{x}, & \text{при } \sigma = 0; \\ x, & \text{при } \sigma = 1. \end{cases}$$

Легко видеть, что $x^{(\sigma)} = x \oplus \sigma \oplus 1 = x\sigma \vee \overline{x}\,\overline{\sigma}$. Для любого $\boldsymbol{x} = (x_1, \ldots, x_n)$ и любого $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ произведение

$$k_{\boldsymbol{\sigma}}(\boldsymbol{x}) = x_1^{(\sigma_1)} \cdot \ldots \cdot x_n^{(\sigma_n)},$$

булевых степеней $\sigma_1, \ldots, \sigma_n$ переменных $x_1, \ldots, x_n$ называется *элементарной конъюнкцией*, ассоциированной с булевым набором $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$. Число переменных, входящих в конъюнкцию $k_{\boldsymbol{\sigma}}(\boldsymbol{x})$, называется *рангом* этой конъюнкции. Для функции $k_{\boldsymbol{\sigma}}(\boldsymbol{x})$ справедливо соотношение

$$k_{\boldsymbol{\sigma}}(\boldsymbol{\alpha}) = \begin{cases} 1, & \text{если } \boldsymbol{\alpha} = \boldsymbol{\sigma}, \\ 0, & \text{если } \boldsymbol{\alpha} \neq \boldsymbol{\sigma}. \end{cases} \tag{1.4.1}$$

Отсюда немедленно следует, что $k_{\boldsymbol{\alpha}}(\boldsymbol{x}) \cdot k_{\boldsymbol{\beta}}(\boldsymbol{x}) = 0$ при $\boldsymbol{\alpha} \neq \boldsymbol{\beta}$.

Теорема 1.4.1. *Для каждой булевой функции $f(x_1, \ldots, x_n)$ при любом m, $1 \leq m \leq n$, справедливо представление*

$$\begin{aligned} f(x_1,\ldots,x_m,x_{m+1},\ldots,x_n) &= \\ &= \bigvee_{(\sigma_1,\ldots,\sigma_m)} x_1^{(\sigma_1)} \cdot \ldots \cdot x_m^{(\sigma_m)} \cdot f(\sigma_1,\ldots,\sigma_m,x_{m+1},\ldots,x_n). \end{aligned} \tag{1.4.2}$$

Доказательство. Покажем, что для произвольного булева набора $(\alpha_1, \ldots, \alpha_n)$ значение функции, реализуемой формулой из правой части (1.4.2), равно $f(\alpha_1, \ldots, \alpha_n)$. Действительно, из (1.4.1) легко следует, что

$$\begin{aligned} &\bigvee_{(\sigma_1,\ldots,\sigma_m)} \alpha_1^{(\sigma_1)} \cdot \ldots \cdot \alpha_m^{(\sigma_m)} \cdot f(\sigma_2,\ldots,\sigma_m,\alpha_{m+1},\ldots,\alpha_n) = \\ &\quad = \alpha_1^{(\alpha_1)} \cdot \ldots \cdot \alpha_m^{(\alpha_m)} \cdot f(\alpha_1,\ldots,\alpha_m,\alpha_{m+1},\ldots,\alpha_n) = f(\alpha_1,\ldots,\alpha_n). \end{aligned}$$

Теорема доказана.

Формула (1.4.2) называется *разложением* функции f по переменным $x_1, \ldots, x_m$. Очевидным образом (1.4.2) обобщается на случай разложения булевой функции по переменным $x_{i_1}, \ldots, x_{i_k}$. Важным частным случаем такого разложения является разложение по одной переменной. Разложение f по первой переменной выглядит следующим образом:

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

Пример 1.4.1. Разложим по первой переменной линейную функцию $x \oplus y \oplus z$. Так как $0 \oplus y \oplus z = y \oplus z$ и $1 \oplus y \oplus z = y \sim z$, то

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

□

[4)] Обычно в математической литературе для булевой степени σ величины x используется обозначение x^{σ}, т. е. величина σ не заключается в скобки.

2. Дизъюнктивные нормальные формы. *Совершенной дизъюнктивной нормальной формой* (СДНФ) функции $f(x_1,\dots,x_n)$ называется ее разложение

$$f(x_1,\dots,x_n)=\bigvee_{(\sigma_1,\dots,\sigma_n)} x_1^{(\sigma_1)}\cdot\ldots\cdot x_n^{(\sigma_n)}\cdot f(\sigma_1,\dots,\sigma_n)$$

по всем ее переменным. Легко видеть, что СДНФ f есть дизъюнкция всех элементарных конъюнкций ассоциированных с теми наборами $\boldsymbol{\sigma}$, на которых функция f равна единице.

Пример 1.4.2. Найдем совершенные дизъюнктивные нормальные формы импликации и конъюнкции переменных x и y. Так как импликация принимает единичные значения на наборах $(00),(01)$ и (11), то в соответствии с определением СДНФ имеем:

$$x\to y=x^{(0)}y^{(0)}\vee x^{(0)}y^{(1)}\vee x^{(1)}y^{(1)}=\overline{x}\,\overline{y}\vee\overline{x}y\vee xy.$$

Конъюнкция $x\&y$ равна единице только если $x=y=1$ и, поэтому, сама является своей совершенной дизъюнктивной нормальной формой. □

Совершенные дизъюнктивные нормальные формы устроены очень просто. В некотором смысле, СДНФ булевой функции это ее вектор значений, записанный на языке формул. С другой стороны, часто СДНФ даже простых функций состоят из очень большого числа элементарных конъюнкций. Так, например, СДНФ дизъюнкции $x_1\vee\dots\vee x_n$ содержит 2^n-1 элементарных конъюнкций. Более экономными (с точки зрения числа символов переменных входящих в формулу) являются дизъюнктивные нормальные формы. *Дизъюнктивной нормальной формой* (ДНФ) булевой функции f называется реализующая f формула, являющаяся дизъюнкцией элементарных конъюнкций. В отличии от СДНФ дизъюнктивная нормальная форма функции f определяется неоднозначно, т.е. f может иметь несколько реализующих ее ДНФ. *Минимальной* дизъюнктивной нормальной формой функции f называется ДНФ, содержащая минимальное число символов переменных среди всех ДНФ функции f. Получить ДНФ можно из СДНФ при помощи эквивалентных преобразований.

Пример 1.4.3. Найдем минимальную ДНФ импликации. Сделаем это преобразуя ее СДНФ при помощи приведенных в предыдущем параграфе равенств:

$$\begin{aligned}x\to y=\overline{x}\,\overline{y}\vee\overline{x}y\vee xy=\overline{x}\,\overline{y}\vee\overline{x}y\vee\overline{x}y\vee xy=\\=\overline{x}(\overline{y}\vee y)\vee y(\overline{x}\vee x)=\overline{x}\vee y.\end{aligned}$$

Так как импликация существенно зависит от двух переменных, то очевидно, что ее любая ДНФ содержит символы обеих переменных. Таким образом, формула $\overline{x}\vee y$ будет минимальной ДНФ импликации. □

3. Конъюнктивные нормальные формы. Для любого $\boldsymbol{x}=(x_1,\dots,x_n)$ и любого $\boldsymbol{\sigma}=(\sigma_1,\dots,\sigma_n)$ дизъюнкция

$$d_{\boldsymbol{\sigma}}(\boldsymbol{x})=x_1^{(\sigma_1)}\vee\ldots\vee x_n^{(\sigma_n)},$$

булевых степеней $\sigma_1,\dots,\sigma_n$ переменных $x_1,\dots,x_n$ называется *элементарной дизъюнкцией*, ассоциированной с булевым набором $\boldsymbol{\sigma}=(\sigma_1,\dots,\sigma_n)$. Если все σ_i равны единице, то дизъюнкция $d_{\boldsymbol{\sigma}}(\boldsymbol{x})$ называется *монотонной*. Для функции $d_{\boldsymbol{\sigma}}(\boldsymbol{x})$ справедливо соотношение

$$d_{\boldsymbol{\sigma}}(\boldsymbol{\alpha})=\begin{cases}0, & \text{если } \boldsymbol{\alpha} \text{ и } \boldsymbol{\sigma} \text{ — противоположные наборы,}\\ 1, & \text{в остальных случаях.}\end{cases}$$

Пусть $f(x_1,\dots,x_n)$ — произвольная булева функция. Представим отрицание f в виде совершенной дизъюнктивной нормальной формы:

$$\overline{f}(x_1,\dots,x_n)=\bigvee_{\boldsymbol{\sigma}=(\sigma_1,\dots,\sigma_n)} x_1^{(\sigma_1)}\cdot\ldots\cdot x_n^{(\sigma_n)}\cdot\overline{f}(\sigma_1,\dots,\sigma_n),$$

где дизъюнкция берется по всем таким наборам $\boldsymbol{\sigma}$, что $f(\boldsymbol{\sigma}) = 0$. Взяв отрицания от обоих частей равенства и применив законы двойственности, видим, что для функции f справедливы равенства

$$f(x_1, \ldots, x_n) = \overline{\bigvee x_1^{(\sigma_1)} \cdot \ldots \cdot x_n^{(\sigma_n)}} = \bigwedge \left(x_1^{(\overline{\sigma}_1)} \vee \ldots \vee x_n^{(\overline{\sigma}_n)} \right), \tag{1.4.3}$$

в которых дизъюнкция и конъюнкция берутся по всем тем наборам $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$, для которых $f(\boldsymbol{\sigma}) = 0$. Формула, стоящая в правой части равенства (1.4.3), называется *совершенной конъюнктивной нормальной формой* (СКНФ) функции $f(x_1, \ldots, x_n)$.

Пример 1.4.4. Найдем совершенные конъюнктивные нормальные формы стрелки Пирса и дизъюнкции переменных x и y. Так стрелка Пирса принимает нулевые значения на наборах (01), (10) и (11), то в соответствии с формулой (1.4.3) имеем:

$$x \downarrow y = (x^{(\overline{0})} \vee y^{(\overline{1})})(x^{(\overline{1})} \vee y^{(\overline{0})})(x^{(\overline{1})} \vee y^{(\overline{1})}) = (x \vee \overline{y})(\overline{x} \vee y)(\overline{x} \vee \overline{y}).$$

Дизъюнкция $x \vee y$ равна нулю только если $x = y = 0$, и поэтому сама является своей совершенной конъюнктивной нормальной формой. □

Конъюнктивной нормальной формой (КНФ) булевой функции f называется реализующая f формула, являющаяся конъюнкцией элементарных дизъюнкций. Как и в случае дизъюнктивных нормальных форм КНФ называется *минимальной*, если она содержит минимальное число переменных среди всех КНФ функции f.

Пример 1.4.5. Найдем минимальную КНФ стрелки Пирса. Сделаем это преобразуя ее СКНФ. Легко видеть, что

$$x \downarrow y = (x \vee \overline{y})(\overline{x} \vee y)(\overline{x} \vee \overline{y}) = (\overline{x}\,\overline{y} \vee yx)(\overline{x} \vee \overline{y}) = \overline{x}\,\overline{y}.$$

Так как стрелка Пирса существенно зависит от двух переменных, то очевидно, что ее любая КНФ содержит символы обеих переменных. Таким образом, формула $\overline{x}\,\overline{y}$ будет минимальной КНФ стрелки Пирса. □

4. Алгебраическая нормальная форма. Для любого $x \in \mathbb{B}$ его *алгебраической степенью* (или просто степенью) называется функция[5)]

$$x^\sigma = \begin{cases} 1, & \text{при } \sigma = 0; \\ x, & \text{при } \sigma = 1. \end{cases}$$

Для любого $\boldsymbol{x} = (x_1, \ldots, x_n)$ и любого $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$ произведение

$$p_{\boldsymbol{\sigma}}(\boldsymbol{x}) = x_1^{\sigma_1} \cdot \ldots \cdot x_n^{\sigma_n}, \tag{1.4.4}$$

степеней $\sigma_1, \ldots, \sigma_n$ переменных $x_1, \ldots, x_n$ называется булевым одночленом, ассоциированным с булевым набором $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_n)$. Вес этого набора называется *степенью одночлена* (1.4.4) и обозначается $\deg p_{\boldsymbol{\sigma}}(\boldsymbol{x})$.

Булевы одночлены от переменных $x_1, \ldots, x_n$ естественным образом нумеруются номерами ассоциированных булевых наборов — номером одночлена (1.4.4) является величина $|\boldsymbol{\sigma}|$. Далее для обозначения одночлена $p_{\boldsymbol{\sigma}}(\boldsymbol{x})$ будем также использовать обозначение $p_{|\boldsymbol{\sigma}|}(\boldsymbol{x})$ или просто $p_{|\boldsymbol{\sigma}|}$, если из контекста понятно от каких переменных зависит рассматриваемый одночлен. Для функции $p_{\boldsymbol{\sigma}}(\boldsymbol{x})$ справедливо соотношение

$$p_{\boldsymbol{\sigma}}(\boldsymbol{\alpha}) = \begin{cases} 1, & \text{если } \boldsymbol{\sigma} \preceq \boldsymbol{\alpha}, \\ 0, & \text{в остальных случаях.} \end{cases} \tag{1.4.5}$$

[5)] См. примечание на стр. 21

Пример 1.4.6. Ниже приведены все булевы наборы длины два и ассоциированные с этими наборами одночлены от двух переменных:

$$\begin{aligned}
&\boldsymbol{\sigma} = (0,0), \quad |\boldsymbol{\sigma}| = 0, \quad p_0 = x_1^0 x_2^0 = 1 \cdot 1 = 1;\\
&\boldsymbol{\sigma} = (0,1), \quad |\boldsymbol{\sigma}| = 1, \quad p_1 = x_1^0 x_2^1 = 1 \cdot x_2 = x_2;\\
&\boldsymbol{\sigma} = (1,0), \quad |\boldsymbol{\sigma}| = 2, \quad p_2 = x_1^1 x_2^0 = x_1 \cdot 1 = x_1;\\
&\boldsymbol{\sigma} = (1,1), \quad |\boldsymbol{\sigma}| = 3, \quad p_3 = x_1^1 x_2^1 = x_2 \cdot x_1.
\end{aligned}$$

Легко видеть, что $\deg p_0 = 0$, $\deg p_1 = \deg p_2 = 1$ и $\deg p_3 = 2$. □

Теорема 1.4.2. *Каждая булева функция $f(x_1, \ldots, x_n)$ единственным образом представляется в виде*

$$f(x_1, \ldots, x_n) = \bigoplus_{\boldsymbol{\sigma}=(\sigma_1,\ldots,\sigma_n)} x_1^{\sigma_1} \cdot \ldots \cdot x_n^{\sigma_n} \cdot p_{\boldsymbol{\sigma}}, \tag{1.4.6}$$

где $p_{\boldsymbol{\sigma}} \in \{0,1\}$. Формула (1.4.6) называется алгебраической нормальной формой функции f или ее многочленом Жегалкина.

Доказательство. Прежде всего покажем, что каждая булева функция реализуется не более чем одним многочленом Жегалкина. Сделаем это методом от противного. Предположим, что некоторую булеву функцию $f(x_1, \ldots, x_n)$ реализуют два различных многочлена Жегалкина h_1 и h_2. Тогда $h_1(\boldsymbol{x}) \oplus h_2(\boldsymbol{x}) = f(\boldsymbol{x}) \oplus f(\boldsymbol{x}) = 0$. С другой стороны, так как $h_1 \neq h_2$, то многочлен $h_1 \oplus h_2$ содержит хотя бы один ненулевой одночлен. Следовательно, тождественный нуль реализуется многочленом $h_1 \oplus h_2$. В многочлене $h_1 \oplus h_2$ выберем одночлен минимальной степени, если таких одночленов несколько, то выберем любой. Пусть $p_{\boldsymbol{\sigma}}$ — выбранный одночлен. Легко видеть, что если одночлен $p_{\boldsymbol{\alpha}}$ принадлежит многочлену $h_1 \oplus h_2$, то либо $\boldsymbol{\sigma} \prec \boldsymbol{\alpha}$, либо эти наборы несравнимы. Из (1.4.5) следует, что $p_{\boldsymbol{\sigma}}(\boldsymbol{\sigma}) = 1$ и $p_{\boldsymbol{\alpha}}(\boldsymbol{\sigma}) = 0$ для любого $\boldsymbol{\alpha}$ большего $\boldsymbol{\sigma}$ или несравнимого с $\boldsymbol{\sigma}$. Поэтому, значение многочлена $h_1 \oplus h_2$ на любом наборе $\boldsymbol{\alpha}$ большем $\boldsymbol{\sigma}$ или несравнимым с $\boldsymbol{\sigma}$ равно единице. Противоречие. Следовательно, каждая булева функция реализуется не более чем одним многочленом Жегалкин.

Для окончательного доказательства теоремы осталось найти число различных многочленов от n переменных. Так как каждый одночлен, зависящий от n переменных (не обязательно существенно), однозначно определяется булевым набором длины n, то очевидно, что число одночленов равно 2^n. Каждый одночлен либо входит, либо не входит в многочлен. Следовательно, число различных многочленов равно 2^{2^n}, т.е. многочленов столько же, сколько и булевых функций, зависящих от n переменных. Так как каждая булева функция реализуется не более чем одним многочленом Жегалкина, то для каждой функции найдется реализующий ее многочлен. Теорема доказана.

Итак, каждая булева функция f единственным образом представляется в виде многочлена Жегалкина. *Степенью* функции f, обозначается через $\deg f$, называется максимальная степень одночленов, входящих в ее многочлен Жегалкина.

Пример 1.4.7. Методом неопределенных коэффициентов найдем многочлен Жегалкина дизъюнкции двух переменных. Для этого дизъюнкцию $x \vee y$ представим в виде многочлена

$$x \vee y = a \oplus bx \oplus cy \oplus dxy \tag{1.4.7}$$

с неизвестными коэффициентами a, b, c и d. Подставляя в левую и правую части (1.4.7) нули вместо переменных x и y, получаем, что $a = 0$. Полагая далее $x = 1$, $y = 0$, находим $a \oplus b = 1$. Подстановки $x = 0$, $y = 1$ и $x = y = 1$, дают, соответственно, $a \oplus c = 1$ и $a \oplus b \oplus c \oplus d = 1$. Таким образом, для определения четырех неизвестных коэффициентов получили систему из четырех уравнений. Решая эту систему, легко находим $a = 0$, $b = c = d = 1$. Следовательно, $x \vee y = x \oplus y \oplus xy$. □

Преобразуем СДНФ произвольной булевой функции f, заменяя в СДНФ дизъюнкции формулами в базисе $\{\oplus, \&\}$. Из рассмотренного примера и равенства нулю произведения двух различных элементарных конъюнкций, зависящих от одних и тех же переменных, легко получаем формулу

$$f(x_1, \ldots, x_n) = \bigoplus_{(\sigma_1, \ldots, \sigma_n)} x_1^{(\sigma_1)} \cdot \ldots \cdot x_n^{(\sigma_n)} \cdot f(\sigma_1, \ldots, \sigma_n), \tag{1.4.8}$$

которая в некоторых случаях оказывается более удобной, чем СДНФ или многочлен Жегалкина.

6. В заключении параграфа рассмотрим несколько примеров построения формул, реализующих различные булевы функции.

Пример 1.4.8. Для $\boldsymbol{u} \in \mathbb{B}^n$ положим $[\boldsymbol{u}] = \sum_{i=1}^{n} u_i 2^{i-1}$. Функцией сравнения булевых наборов $\boldsymbol{x} = (x_1, \ldots, x_n)$ и $\boldsymbol{y} = (y_1, \ldots, y_n)$ назовем функцию

$$f_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = \begin{cases} 0, & \text{если } [\boldsymbol{x}] \geq [\boldsymbol{y}], \\ 1, & \text{если } [\boldsymbol{x}] > [\boldsymbol{y}]. \end{cases}$$

Нетрудно видеть, что $f_1(x_1, y_1) = \overline{x}_1 y_1$, а при $n > 1$ функция f_n равна единице, если выполняется хотя бы одно из двух следующих условий:

- $x_n < y_n$;
- $x_n = y_n$ и $f_{n-1}(x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}) = 1$.

Из этих условий для f_n легко получается рекуррентная формула:

$$f_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = \overline{x}_n\, y_n \vee (x_n \sim y_n) f_{n-1}(x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}). \tag{1.4.9}$$

Используя (1.4.9) и формулу $\mathsf{F}_1(x_1, y_1) = \overline{x}_1 y_1$, реализующую функцию f_1, последовательно выпишем формулы F_2 и F_3, которые реализуют две следующие функции сравнения f_2 и f_3:

$$\begin{aligned} \mathsf{F}_2(x_1, x_2, y_1, y_2) &= \overline{x}_2\, y_2 \vee (x_2 \sim y_2)\mathsf{F}_1(x_1, y_1) = \overline{x}_2\, y_2 \vee (x_2 \sim y_2))\overline{x}_1\, y_1; \\ \mathsf{F}_3(x_1, x_2, x_3, y_1, y_2, y_3) &= \overline{x}_3\, y_3 \vee (x_3 \sim y_3)\mathsf{F}_2(x_1, x_2, y_1, y_2) = \\ &= \overline{x}_3\, y_3 \vee (x_3 \sim y_3)(\overline{x}_2\, y_2 \vee (x_2 \sim y_2)\overline{x}_1\, y_1). \end{aligned}$$

Аналогичным образом при любом конечном n для функции сравнения f_n можно построить реализующую ее формулу F_n. Найдем сложность построенной таким образом формулы. Так как сложность формулы F_1 равна трем, а из (1.4.9) следует, что сложность каждой последующей формулы по сравнению с предыдущей увеличивается на пять, то при $n > 1$ для сложности формулы F_n имеем

$$l(\mathsf{F}_n) = 5 + l(\mathsf{F}_{n-1}) = \ldots = 5(n-1) + l(\mathsf{F}_1) = 5n - 3.$$

Совершенно аналогично находится глубина формулы F_n. Так как глубина F_1 равна единице, а из (1.4.9) следует, что $d(\mathsf{F}_n) = 2 + d(\mathsf{F}_{n-1})$, то легко видеть, что при $n > 1$ глубина формулы F_n равна $2n - 1$. □

Пример 1.4.9. Найдем формулы, реализующие координатные функции s_1, s_2 и s_3 оператора сложения двух двоичных двухразрядных чисел $\boldsymbol{x} = x_2 2 + x_1$ и $\boldsymbol{y} = y_2 2 + y_1$ из примера 1.2.6. Для определения формул воспользуемся хорошо известным алгоритмом сложения целых чисел "столбиком". Нетрудно видеть, что в рассматриваемом случае справедливы равенства

$$s_1 = x_1 \oplus y_1, \qquad s_2 = x_2 \oplus y_2 \oplus x_1 y_1, \qquad s_3 = \tau_2(x_2, y_2, x_1 y_1).$$

Подставляя x_2, y_2 и $x_1 y_1$ в формулу для функции голосования, найденную в примере 1.3.3 на странице 20, для функции s_3 имеем

$$s_3 = x_2 y_2 \oplus x_2 x_1 y_1 \oplus y_2 x_1 y_1 = x_2 y_2 \oplus x_1 y_1 (x_2 \oplus y_2).$$

□

Пример 1.4.10. Найдем формулы, реализующие координатные функции оператора вычитания из примера 1.2.6. Напомним, что этот оператор вычисляет разность $\boldsymbol{x} - \boldsymbol{y}$ двух двоичных двухразрядных чисел $\boldsymbol{x} = x_2 2 + x_1$ и $\boldsymbol{y} = y_2 2 + y_1$. Разность представляется в виде трех булевых величин r_1, r_2 и r_3 так, что число $2r_2 + r_1$ равно модулю разности $\boldsymbol{x}$ и $\boldsymbol{y}$, а булева величина r_3 определяет знак разности — $r_3 = 0$ если $\boldsymbol{x} \geq \boldsymbol{y}$, и $r_3 = 1$ если $\boldsymbol{x} < \boldsymbol{y}$. Функции r_1, r_2 и r_3 задаются таблицей 1.2.7 на странице 13.

Прежде всего заметим, что формула для функции r_3 была найдена в примере 1.4.8. Из этого примера имеем:

$$r_3(x_1, x_2, y_1, y_2) = \overline{x}_2\, y_2 \vee (x_2 \sim y_2))\overline{x}_1\, y_1.$$

Из таблиц 1.2.6 и 1.2.7 видно, что функция r_1 равна функции s_1 из предыдущего примера. Следовательно, $r_1 = x_1 \oplus y_1$.

Наконец найдем формулу для функции r_2. Сначала для r_2 выпишем СДНФ, затем преобразуем ее при помощи эквивалентных преобразований:

$$\begin{aligned} r_2 &= \overline{x}_2\overline{x}_1 y_2\overline{y}_1 \vee \overline{x}_2\overline{x}_1 y_2 y_1 \vee \overline{x}_2 x_1 y_2 y_1 \vee x_2\overline{x}_1\overline{y}_2\overline{y}_1 \vee x_2 x_1\overline{y}_2\overline{y}_1 \vee x_2 x_1\overline{y}_2 y_1 = \\ &= \overline{x}_2\overline{x}_1 y_2(\overline{y}_1 \vee y_1) \vee \overline{x}_2 x_1 y_2 y_1 \vee x_2\overline{x}_1\overline{y}_2\overline{y}_1 \vee x_2 x_1\overline{y}_2(\overline{y}_1 \vee y_1) = \\ &= \overline{x}_2 y_2(\overline{x}_1 \vee x_1 y_1) \vee x_2\overline{y}_2(\overline{x}_1\overline{y}_1 \vee x_1) = \\ &= \overline{x}_2 y_2(\overline{x}_1 \vee y_1) \vee x_2\overline{y}_2(\overline{y}_1 \vee x_1) = \overline{x}_2 y_2(x_1 \to y_1) \vee x_2\overline{y}_2(y_1 \to x_1). \end{aligned}$$

Таким образом, $r_2 = \overline{x}_2 y_2(x_1 \to y_1) \vee x_2\overline{y}_2(y_1 \to x_1)$. □

Задачи

1.4.1. Разложить по первой переменной функцию:
a) $x_1 \vee x_2 \vee x_3$, b) $1 \oplus x_1 \oplus x_2 \oplus x_3$, c) $x_1 x_2 \to x_3$.

1.4.2. Показать, что каждая булева функция трех переменных имеет симметрическую подфункцию двух переменных.

1.4.3. Найти совершенные дизъюнктивные нормальные формы всех булевых функций двух переменных.

1.4.4. Найти число различных дизъюнктивных нормальных форм над множеством переменных $x_1, \ldots, x_n$.

1.4.5. Показать, что любая ДНФ функции $x_1 \oplus \ldots \oplus x_n$ состоит из:
a) 2^{n-1} элементарных конъюнкций; b) $n2^{n-1}$ символов переменных.

1.4.6. Найти совершенные конъюнктивные нормальные формы всех булевых функций двух переменных.

1.4.7. Найти число различных конъюнктивных нормальных форм над множеством переменных $x_1, \ldots, x_n$.

1.4.8. Показать, что любая КНФ функции $x_1 \oplus \ldots \oplus x_n$ состоит из:
a) 2^{n-1} элементарных дизъюнкций; b) $n2^{n-1}$ символов переменных.

1.4.9. Найти число одночленов степени k от n переменных.

1.4.10. Найти число одночленов четной степени от n переменных.

1.4.11. Найти число одночленов степени $\leq \frac{n}{2}$ от n переменных.

1.4.12. Найти многочлены Жегалкина всех булевых функций двух переменных.

1.4.13. Найти многочлен Жегалкина функции f если:
a) $f = x \vee y \vee z$, b) $f = (x \downarrow y) \vee z$, c) $f = (x \to y)\&(y \to z)$.

1.4.14. Функция $f(x, y, z, t)$ равна единице, если выполняется хотя бы одно из трех условий: 1) $x < y$; 2) $z < t$; 3) число $8x + 4y + 2z + t$ без остатка делится на пять. Написать формулу в базисе $P_2(2)$, которая реализует f. Полученную формулу максимально упростить.

1.4.15. Функция $f(x,y,z,t)$ равна единице, если выполняется хотя бы одно из двух условий: 1) $2x+y \geq 2z+t$; 2) число $4y+2z+t$ — простое. Написать формулу в базисе $P_2(2)$, которая реализует f. Полученную формулу максимально упростить.

1.4.16. Функция $f(x,y,z,t)$ равна единице, если выполняется хотя бы одно из двух условий: 1) $x \leq y \leq z \leq t$; 2) число $x+y+z+t$ — четное. Написать формулу в базисе $\{\&, \vee, \neg\}$, которая реализует f. Полученную формулу максимально упростить.

1.4.17. Функция $f(x,y,z,t)$ равна единице, если $4x+2y+z-5t > 0$. Написать формулу в базисе $\{\&, \oplus, 1\}$, которая реализует f. Полученную формулу максимально упростить.

1.4.18. Написать формулу, реализующую частичную функцию, заданную вектором значений:
a) $(000*\ 0***\ 100*\ 0000)$; b) $(11*1\ *111\,010*\ 0010)$; c) $(100*\ 0111\ 1111\ 1***)$.

1.4.19. Показать, что функция сравнения f_{2^n} из примера 1.4.8 может быть реализована формулой, глубина которой асимптотически не превосходит $2n$.

1.5. Замкнутые классы булевых функций

Пусть R — произвольное множество булевых функций. *Замыканием* множества R называется множество всех функций, которые можно реализовать формулами в базисе R. Замыкание множества R будем обозначать через $[R]$.

Множество булевых функций R называется (*функционально*) *замкнутым* множеством, если оно совпадает со своим замыканием, т.е. $R = [R]$.

Рассмотрим пять важнейших замкнутых множеств в P_2. Часто рассматриваемые ниже замкнутые множества называются так же замкнутыми классами.

1. Будем говорить, что функция $f(x_1,\ldots,x_n)$ сохраняет нуль, если

$$f(0,\ldots,0) = 0.$$

Множество, состоящее из всех булевых функций сохраняющих нуль, обозначается через T_0. Легко видеть, что функции 0, x, $x\&y$, $x \vee y$ и $x \oplus y$ принадлежат T_0, а функции 1, $\overline{x}$, $x \sim y$, $x \,|\, y$, $x \downarrow y$ и $x \to y$ не принадлежат T_0.

Так как тождественная функция сохраняет нуль, и для любых сохраняющих нуль функций $f_0, f_1, \ldots, f_k$ справедливо равенство

$$f(0,\ldots,0) = f_0(f_1(0,\ldots,0),\ldots,f_k(0,\ldots,0)) = f_0(0,\ldots,0) = 0,$$

т.е. реализуемая формулой $f_0(f_1,\ldots,f_k)$ функция f так же сохраняет нуль, то легко видеть, что множество T_0 замкнуто.

Любой булев вектор длины 2^n с первой нулевой компонентой будет вектором значений функции из T_0. Поэтому, в T_0 содержится ровно 2^{2^n-1} функций из $P_2(n)$. Множество $T_0 \cap P_2(n)$ будем обозначать через $T_0(n)$.

2. Будем говорить, что функция $f(x_1,\ldots,x_n)$ сохраняет единицу, если

$$f(1,\ldots,1) = 1.$$

Множество, состоящее из всех булевых функций сохраняющих единицу, обозначается через T_1.

Легко видеть, что функции 1, x, $x\&y$, $x \vee y$, $x \sim y$ и $x \to y$ принадлежат T_1, а функции 0, $\overline{x}$, $x \oplus y$, $x \,|\, y$ и $x \downarrow y$ не принадлежат T_1. Доказательство замкнутости множества T_1 аналогично доказательству замкнутости множества T_0. Также легко видеть, что в T_1 содержится ровно 2^{2^n-1} функций из $P_2(n)$. Множество $T_1 \cap P_2(n)$ будем обозначать через $T_1(n)$.

3. Будем говорить, что булева функция $f(x_1,\ldots,x_n)$ является *двойственной* к функции $g(x_1,\ldots,x_n)$, если

$$f(x_1,\ldots,x_n) = \overline{g}(\overline{x}_1,\ldots,\overline{x}_n).$$

Функцию двойственную к функции f будем обозначать через f^*. Легко видеть, что $(f^*)^* = f$ для любой булевой функции f. Из законов двойственности следует, что $(x\&y)^* = x \vee y$ и $(x \vee y)^* = x\&y$. Функция f называется *самодвойственной*, если $f = f^*$. Множество, состоящее из всех самодвойственных булевых функций, обозначается через S. Самодвойственными являются функции x, $\overline{x}$, $x_1 \oplus x_2 \oplus x_3$. Среди булевых функций, существенно зависящих ровно от двух переменных, нет ни одной самодвойственной функции.

Докажем замкнутость множества самодвойственных функций. Пусть $f_0, f_1, \ldots, f_k$ — произвольные самодвойственные функции. Рассмотрим новую функцию $f = f_0(f_1, \ldots, f_k)$. Так как добавление фиктивной переменной оставляет самодвойственную функцию самодвойственной, то без ограничения общности будем полагать, что все функции f_i зависят от одних и тех же переменных $x_1, \ldots, x_n$. Тогда

$$\begin{aligned} f(\overline{x}_1, \ldots, \overline{x}_n) &= f_0(f_1(\overline{x}_1, \ldots, \overline{x}_n), \ldots, f_k(\overline{x}_1, \ldots, \overline{x}_n)) = \\ &= f_0(\overline{f}_1(x_1, \ldots, x_n), \ldots, \overline{f}_k(x_1, \ldots, x_n)) = \\ &= \overline{f}_0(f_1(x_1, \ldots, x_n), \ldots, f_k(x_1, \ldots, x_n)) = \overline{f}(x_1, \ldots, x_n). \end{aligned}$$

Следовательно, функция f — самодвойственная. Таким образом, множество S замкнуто.

Так как каждая самодвойственная функция на противоположных наборах принимает противоположные значения, то для определения любой самодвойственной функции достаточно задать ее значения только на половине из 2^n наборов. Следовательно, в S содержится ровно $2^{2^{n-1}}$ функций из $P_2(n)$. Далее множество самодвойственных функций, зависящих от n переменных, будем обозначать через $S(n)$.

4. Функция $f(x_1, \ldots, x_n)$ называется *линейной*, если степень ее многочлена Жегалкина не превосходит единицу, т. е.

$$f(x_1, \ldots, x_n) = \alpha_1 x_1 \oplus \ldots \oplus \alpha_n x_n \oplus \alpha_0,$$

где α_i — булевы постоянные. Множество, состоящее из всех линейных булевых функций, обозначается через L. Очевидно, что среди функций из $P_2(2)$ линейными являются только 0, 1, x, $\overline{x}$, $x \oplus y$ и $x \sim y$. Непосредственно из определения линейной функции следует замкнутость множества L.

Так как каждая булева функция однозначно определяется коэффициентами своего многочлена Жегалкина, а у каждой линейной функции все коэффициенты при одночленах степени два и выше равны нулю, то легко видеть, что в L содержится ровно 2^{n+1} функций из $P_2(n)$. Далее множество $L \cap P_2(n)$ будем обозначать через $L(n)$.

5. Функция $f(x_1, \ldots, x_n)$ называется *монотонной*, если

$$f(\alpha_1, \ldots, \alpha_n) \le f(\beta_1, \ldots, \beta_n)$$

для любых наборов $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ и $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_n)$ таких, что $\boldsymbol{\alpha} \preceq \boldsymbol{\beta}$. Множество, состоящее из всех монотонных булевых функций, обозначается через M. В $P_2(2)$ монотонными являются функции 0, 1, x, $x\&y$ и $x \vee y$.

Докажем замкнутость множества монотонных функций. Пусть $f_0, f_1, \ldots, f_k$ — произвольные монотонные функции. Очевидно, что добавление фиктивной переменной оставляет монотонную функцию монотонной. Поэтому без ограничения общности будем полагать, что все функции f_i зависят от одних и тех же переменных $x_1, \ldots, x_n$. Пусть $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ — такие наборы из $\mathbb{B}^n$, что $\boldsymbol{\alpha} \preceq \boldsymbol{\beta}$. Рассмотрим новую функцию $f = f_0(f_1, \ldots, f_k)$. Так как $f_i(\boldsymbol{\alpha}) \le f_i(\boldsymbol{\beta})$, то $(f_1(\boldsymbol{\alpha}), \ldots, f_k(\boldsymbol{\alpha})) \preceq (f_1(\boldsymbol{\beta}, \ldots, f_k(\boldsymbol{\beta}))$, и поэтому

$$f(\boldsymbol{\alpha}) = f_0(f_1(\boldsymbol{\alpha}), \ldots, f_k(\boldsymbol{\alpha})) \le f_0(f_1(\boldsymbol{\beta}), \ldots, f_k(\boldsymbol{\beta})) = f(\boldsymbol{\beta}).$$

Следовательно, функция f — монотонная. Таким образом, множество M замкнуто.

В отличие от множеств $T_0(n)$, $T_1(n)$, $S(n)$ и $L(n)$, мощности которых легко были найдены выше, точное число монотонных функций в $P_2(n)$ при больших n не известно. Поэтому

здесь без доказательства приведем только формулу для асимптотики логарифма числа монотонных функций в $P_2(n)$. Обозначим множество монотонных функций n переменных через $M(n)$. Тогда

$$\log_2 |M(n)| \sim \binom{n}{\lfloor \frac{n}{2} \rfloor} \sim \frac{2^n}{\sqrt{\pi n/2}}.$$

Пример 1.5.1. Найдем мощность множества $L(n) \cap S(n)$. Из определений линейных и самодвойственных функций следует, что если $f \in L(n) \cap S(n)$, то для f справедливы равенства

$$f(\boldsymbol{x}) = \alpha_0 \oplus \alpha_1 x_1 \oplus \cdots \oplus \alpha_n x_n = 1 \oplus \alpha_0 \oplus \alpha_1(x_1 \oplus 1) \oplus \cdots \oplus \alpha_n(x_n \oplus 1) =$$
$$= (\alpha_0 \oplus \alpha_1 x_1 \oplus \cdots \oplus \alpha_n x_n) \oplus (1 \oplus \alpha_1 \oplus \cdots \oplus \alpha_n) = f(\boldsymbol{x}) \oplus (1 \oplus \alpha_1 \oplus \cdots \oplus \alpha_n).$$

Откуда немедленно следует, что $\alpha_1 \oplus \cdots \oplus \alpha_n = 1$. Таким образом, линейная функция будет самодвойственной тогда и только тогда, когда она существенно зависит от нечетного числа переменных. Следовательно, $|L(n) \cap S(n)| = 2^n$. $\square$

Задачи

1.5.1. Найти число монотонных функций в $P_2(3)$.

1.5.2. Показать, что функция двойственная к монотонной так же будет монотонной.

1.5.3. Найти число монотонных булевых функций, зависящих от n переменных и принимающих единичное значение ровно на 7 наборах.

1.5.4. Пусть функция $f(x_1, \ldots, x_n)$ принадлежит множеству $[x \to y]$ и существенно зависит не менее чем от двух переменных. Доказать, что она принимает единичные значения более чем на 2^{n-1} наборах.

1.5.5. Будет ли множество симметрических функций замкнутым?

1.5.6. Показать, что $\log_2 |M(n)| \geq \binom{n}{\lfloor n/2 \rfloor}$.

1.5.7. Показать, что $\log_2 |M(n)| \leq \binom{n}{\lfloor n/2 \rfloor} \log_2 n$.

1.5.8. Найти

a) $|L(n) \cup T_0(n) \cup T_1(n)|$; b) $|L(n) \cap S(n) \cap T_1(n)|$;
c) $|L(n) \cap M(n) \cup T_0(n)|$; d) $|(S(n) \cup T_0(n)) \setminus L(n)|$;
e) $|(S(n) \cup T_0(n) \cup T_1(n)) \cap L(n)|$; f) $|S(n) \setminus (T_0(n) \setminus L(n))|$;
g) $|S(n) \cup L(n) \cup T_0(n) \cup T_1(n)|$; h) $|(L(n) \setminus S(n)) \setminus M(n)|$.

1.6. Критерий полноты системы булевых функций

Система булевых функций $F = \{f_1, f_2, \ldots, f_i, \ldots\}$ называется *функционально полной*, если любая булева функция может быть реализована формулой в базисе F. Так как каждая булева функция реализуется своими совершенной дизъюнктивной нормальной формой и алгебраической нормальной формой, то системы функций $\{\&, \vee, \neg\}$ и $\{\&, \oplus, 1\}$ будут полными. Для установления полноты других систем можно воспользоваться следующей теоремой.

Теорема 1.6.1. *Пусть F и G — системы булевых функций, система F полная и каждая ее функция реализуется формулой в базисе G. Тогда G — полная система.*

Доказательство. Покажем, что произвольная булева функция h может быть реализована формулой в базисе G. Сделаем это индукцией по глубине формул, реализующих булевы функции в базисе F. В основание индукции положим формулы нулевой глубины, т.е. переменные. Далее предположим, что любая функция, реализуемая в базисе F формулой глубины l, может быть реализована формулой в базисе G. Пусть h — произвольная булева функция для которой существует реализующая ее в базисе F формула H глубины l. Тогда

$\mathsf{H} = f(\mathsf{H}_1, \ldots, \mathsf{H}_k)$, где $f \in F$ и $\mathsf{H}_1, \ldots, \mathsf{H}_k$ формулы в базисе F. Очевидно, что $d(\mathsf{H}_i) < d(\mathsf{H})$ для каждого i из $\{1, \ldots, k\}$ и поэтому по предположению индукции реализуемые формулами $\mathsf{H}_1, \ldots, \mathsf{H}_k$ функции $h_1, \ldots, h_k$ реализуются так же формулами $\mathsf{H}'_1, \ldots, \mathsf{H}'_k$ в базисе G. По условию теоремы функция $f(y_1, \ldots, y_k)$ реализуется формулой F в базисе G. В этой формуле каждую переменную y_i заменим формулой H'_i. Полученная формула H' реализует функцию h и является формулой в базисе G. Теорема доказана.

Из доказанной теоремы и законов двойственности немедленно следует полнота систем $\{\&, \neg\}$ и $\{\vee, \neg\}$.

Пример 1.6.1. Покажем, что система $\{x \to y, \overline{x}\}$ является полной. Так как $x \to y = \overline{x} \vee y$, то $\overline{x} \to y = x \vee y$, т.е. дизъюнкция двух переменных реализуется формулой в базисе из импликации и отрицания. Ранее было отмечено, что дизъюнкция и отрицание образуют полную систему. Следовательно, в силу предыдущей теоремы система $\{x \to y, \overline{x}\}$ является полной. $\square$

Теорема 1.6.2. *Для того чтобы система функций F была полной, необходимо и достаточно, чтобы она не содержалась целиком ни в одном из пяти замкнутых классов T_0, T_1, S, M и L.*

Доказательство. Необходимость. Пусть система функций F полна в P_2. Предположим, что F целиком содержится в замкнутом классе $R \in \{T_0, T_1, L, S, M\}$. Тогда из свойств операции замыкания, включения $F \subseteq R$ и равенства $[F] = P_2$ следует, что $P_2 = [F] = R$. Поэтому, $R = P_2$. С другой стороны каждый из классов T_0, T_1, L, S, M отличен от P_2, т.е. $R \neq P_2$. Противоречие. Необходимость доказана.

Достаточность. Так как система функций F не содержится целиком ни в одном из пяти замкнутых классов перечисленных в условии теоремы, то в этой системе найдутся пять (не обязательно различных) функций f_{T_0}, f_{T_1}, f_S, f_M и f_L таких, что

$$f_{T_0} \notin T_0, \quad f_{T_1} \notin T_1, \quad f_S \notin S, \quad f_M \notin M, \quad f_L \notin L.$$

Если $f_{T_0}(1, \ldots, 1) = 0$, то $f_{T_0}(x, \ldots, x) = \overline{x}$. Если $f_{T_1}(0, \ldots, 0) = 1$, то $f_{T_1}(x, \ldots, x) = \overline{x}$. Если же $f_{T_0}(1, \ldots, 1) = 1$ и $f_{T_1}(0, \ldots, 0) = 0$, то

$$f_{T_0}(x, \ldots, x) = 1, \quad f_{T_1}(x, \ldots, x) = 0.$$

Следовательно, после отождествления переменных у функций f_{T_0} и f_{T_1} получаем либо (i) отрицание, либо (ii) две тождественные константы 0 и 1.

Последовательно рассмотрим эти возможности.

(i) Так как f_S не является самодвойственной, то найдутся такие противоположные наборы $\alpha_1, \ldots, \alpha_k$ и $\overline{\alpha}_1, \ldots, \overline{\alpha}_k$, что

$$f_S(\alpha_1, \ldots, \alpha_k) = f_S(\overline{\alpha}_1, \ldots, \overline{\alpha}_k).$$

Рассмотрим функцию $\varphi_1(x) = f_S(x^{(\alpha_1)}, \ldots, x^{(\alpha_k)})$. Легко видеть, что

$$\begin{aligned}\varphi_1(0) &= f_S(0^{(\alpha_1)}, \ldots, 0^{(\alpha_k)}) = f_S(\overline{\alpha}_1, \ldots, \overline{\alpha}_k) = \\ &= f_S(\alpha_1, \ldots, \alpha_k) = f_S(1^{(\alpha_1)}, \ldots, 1^{(\alpha_k)}) = \varphi_1(1),\end{aligned}$$

т.е. функция φ_1 является константой. Вторая константа получается из φ_1 и отрицания.

(ii) Так как f_M не является монотонной, то найдутся такие соседние наборы $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k)$ и $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_k)$, что $\boldsymbol{\alpha} \prec \boldsymbol{\beta}$ и

$$f_M(\alpha_1, \ldots, \alpha_k) > f_M(\alpha_1, \ldots, \alpha_k).$$

При каждом $1 \leq i \leq k$ определим функцию

$$g_i(x) = \begin{cases} 0, & \text{если } \alpha_i = \beta_i = 0; \\ 1, & \text{если } \alpha_i = \beta_i = 1; \\ x, & \text{если } \alpha_i = 0,\ \beta_i = 1. \end{cases}$$

Рассмотрим функцию $\varphi_2(x) = f_M(g_1(x), \ldots, g_k(x))$. Легко видеть, что

$$\varphi_2(0) = f_M(\alpha_1, \ldots, \alpha_k) > f_4(\beta_1, \ldots, \beta_k) = \varphi_2(1),$$

т. е. φ_2 — отрицание.

Таким образом, из функций f_{T_0}, f_{T_1}, f_S и f_M получены константы и отрицание.

Теперь, рассмотрим нелинейную функцию f_L. В многочлене Жегалкина этой функции найдется слагаемое, содержащее не менее двух переменных. Без ограничения общности будем полагать, что это x_1 и x_2. Группируя слагаемые, содержащие x_1, x_2 и x_1x_2, преобразуем многочлен Жегалкина функции f_L к виду

$$f_l(x_1, \ldots, x_k) = x_1x_2f_0(x_3, \ldots, x_k) \oplus x_1f_1(x_3, \ldots, x_k) \oplus x_2f_2(x_2, \ldots, x_k) \oplus f_3(x_2, \ldots, x_k),$$

где функция f_0 отлична от тождественного нуля. Далее воспользуемся равенством $x \oplus 1 = \overline{x}$. Пусть набор $(\alpha_3, \ldots, \alpha_k)$ таков, что $f_0(\alpha_3, \ldots, \alpha_k) = 1$. Введем функцию

$$\psi(x_1, x_2) = f_M(x_1, x_2, \alpha_3, \ldots, \alpha_k) = x_1x_2 \oplus \gamma_1x_1 \oplus \gamma_2x_2 \oplus \gamma_3,$$

где $\gamma_i = f_i(\alpha_3, \ldots, \alpha_k)$ — константы. Положим

$$\varphi_3(x_1, x_2) = \psi(x_1 \oplus \gamma_2, x_2 \oplus \gamma_1) \oplus \gamma_1\gamma_2 \oplus \gamma_3.$$

Легко видеть, что

$$\begin{aligned}\varphi_3(x_1, x_2) &= \psi(x_1 \oplus \gamma_2, x_2 \oplus \gamma_1) \oplus \gamma_1\gamma_2 \oplus \gamma_3 = \\ &= (x_1 \oplus \gamma_2)(x_2 \oplus \gamma_1) \oplus \gamma_1(x_1 \oplus \gamma_2) \oplus \gamma_2(x_2 \oplus \gamma_1) \oplus \gamma_3 \oplus \gamma_1\gamma_2 \oplus \gamma_3 = x_1x_2\end{aligned}$$

Таким образом, $\varphi_3(x_1, x_2) = x_1x_2$. Теперь утверждение теоремы следует из теоремы 1.6.1 и полноты системы функций $\{\&, \neg\}$.

Пример 1.6.2. Снова, как и в примере 1.6.1, докажем полноту системы $\{x \to y, \overline{x}\}$. Сделаем это при помощи теоремы 1.6.2. Легко видеть, что отрицание не сохраняет нуль, не сохраняет единицу и не является монотонной функцией. Многочлен Жегалкина импликации равен $xy \oplus x \oplus 1$. Следовательно, импликация не принадлежит классу линейных функций. Наконец заметим, что вес импликации равен трем, и поэтому, она не является самодвойственной функцией. Таким образом, система $\{x \to y, \overline{x}\}$ не содержится целиком ни в одном пяти замкнутых классов T_0, T_1, S, M и L и в силу теоремы 1.6.2 является полной. □

Пример 1.6.3. Используя теорему 1.6.2, исследуем полноту системы $\{x \to y, xy\}$. Так как $1 \to 1 = 1$ и $1 \cdot 1 = 1$, то очевидно, что импликация и конъюнкция сохраняют единицу. Следовательно, система $\{x \to y, xy\}$ содержится в классе T_1 и в силу теоремы 1.6.2 не является полной. □

Пусть F — замкнутый класс в P_2. Система булевых функций $\{f_1, \ldots, f_i, \ldots\}$ называется *базисом* в F, если ее замыкание совпадает с F, а любая ее собственая подсистема не является полной в F. Рассмотренная в примере 1.6.2 система функций $\{x \to y, \overline{x}\}$ является базисом в P_2. Эта система полна в P_2, ее первая функция сохраняет единицу, а вторая яаляется линейной и самодвойственной. Другая полная в P_2 система функций $\{\&, \vee, \neg\}$ базисом не является, так как две ее собственные подсистемы $\{\&, \neg\}$ и $\{\vee, \neg\}$ полны в P_2.

Булева функция f называется *шефферовой*, если $[f] = P_2$. Очевидно, что каждая шефферова функция является базисом в P_2.

Пример 1.6.4. Найдем все шефферовы функции в $P_2(2)$. Допустим, что f — шефферова, и $\boldsymbol{f} = (f_1, f_2, f_3, f_4)$ — вектор ее значений. Так как $f \notin T_0$ и $f \notin T_1$, то $f_1 = 1$ и $f_4 = 0$. Отсюда немедленно следует, что $f \notin M$. Теперь посмотрим, какие значения могут принимать вторая и третья компоненты вектора $\boldsymbol{f}$. Если $f_2 \neq f_3$, то либо $\boldsymbol{f} = (1100)$, либо $\boldsymbol{f} = (1010)$. В обоих случаях f будет линейной самодвойственной функцией. Если $f_2 = f_3$, то либо $\boldsymbol{f} = (1000)$, либо $\boldsymbol{f} = (1110)$. Непосредственной проверкой легко убедиться, что каждый из этих векторов задает нелинейную и несамодвойственную функцию. Таким образом, в $P_2(2)$ содержится ровно две шефферовы функции: стрелка Пирса и штрих Шеффера. □

Задачи

1.6.1. Выяснить, является ли множество A базисом в B:
a) $A = \{xy \sim z\},\ B = T_1$;
b) $A = \{xy \vee z\},\ B = T_0$;
c) $A = \{x \sim y, x \oplus y\},\ B = L$;
d) $A = \{x_1 \oplus x_2 \oplus \cdots \oplus x_k, 1\}, k$ — константа, $B = L$;

1.6.2. Выяснить, при каких n функция f является шефферовой:
a) $f(x_1, \ldots, x_n) = 1 \oplus x_1x_2 \oplus \ldots \oplus x_ix_{i+1} \oplus \ldots \oplus x_{n-1}x_n \oplus x_nx_1$;
b) $f(x_1, \ldots, x_n) = 1 \oplus x_1x_2 \oplus \ldots \oplus x_ix_{i+1} \oplus \ldots \oplus x_{n-1}x_n$;
c) $f(x_1, \ldots, x_n) = 1 \oplus \sum_{1 \le i < j \le n} x_ix_j$;
d) $f(x_1, \ldots, x_n) = 1 \oplus (x_1|x_2) \oplus \ldots \oplus (x_i|x_{i+1}) \oplus \ldots \oplus (x_{n-1}|x_n) \oplus (x_n|x_1)$;
e) $f(x_1, \ldots, x_n) = 1 \oplus (x_1|x_2) \oplus \ldots \oplus (x_i|x_{i+1}) \oplus \ldots \oplus (x_{n-1}|x_n)$;
f) $f(x_1, \ldots, x_n) = 1 \oplus (x_1 \to x_2) \oplus \ldots \oplus (x_i \to x_{i+1}) \oplus \ldots \oplus (x_{n-1} \to x_n)$.

1.6.3. Доказать, что если f монотонна и существенно зависит не менее чем от двух переменных, то система $\{0, \overline{f}\}$ полна в P_2.

1.6.4. Набор $\boldsymbol{\alpha}$ из $\mathbb{B}^n$ назовем нижней единицей монотонной функции f, если $f(\boldsymbol{\alpha}) = 1$ и $f(\boldsymbol{\beta})=0$ для каждого $\boldsymbol{\beta} \prec \boldsymbol{\alpha}$. Пусть монотонная функция f имеет ровно две нижние единицы. Доказать, что $\overline{f}$ — шефферова.

1.6.5. Найти число шефферовых функций в:
a) $P_2(3)$; b) $P_2(4)$; c) $P_2(5)$; d) $P_2(n)$.

Глава 2.

Линейные булевы пространства

В различных областях непрерывной математики и в ее многочисленных приложениях широко используются методы линейной алгебры. Не является исключением и дискретная математика — введение структуры линейного пространства в дискретные множества позволяет решать многие задачи, решение которых иными методами было бы проблематично.

В настоящей главе вводятся линейные булевы пространства и исследуются их основные свойства. Большая часть из приводимых ниже понятий и утверждений имеют прямые аналоги в традиционных курсах линейной алгебры. Однако, обычно, в этих курсах утверждения доказываются только для линейных пространств определенных над полем действительных или комплексных чисел, и, поэтому, их применение в булевом случае требует дополнительных обоснований. Более того, иногда прямые аналогии между непрерывным и булевым случаями не имеют места. Ярким примером отсутствия такой аналогии является понятие ортогональности. Если в линейных пространствах над полем действительных чисел ортогональные векторы линейно независимы, то в булевом случае вектор четного веса ортогонален сам себе. Поэтому некоторые утверждения, которые используют понятие ортогональности и кажутся очевидными в действительном линейном пространстве, могут быть неверными в булевом случае.

2.1. Линейные булевы пространства

1. Конечное множество $\mathbb{V}$, замкнутое относительно коммутативной и ассоциативной операции сложения $+$, называется линейным булевым пространством, если:

(i) в $\mathbb{V}$ существует нулевой элемент $\mathbf{0}$ такой, что $\boldsymbol{v}+\mathbf{0}=\boldsymbol{v}$ для каждого $\boldsymbol{v}\in\mathbb{V}$;

(ii) каждый элемент $\boldsymbol{v}\in\mathbb{V}$ является своим обратным относительно операции сложения, т. е. $\boldsymbol{v}+\boldsymbol{v}=\mathbf{0}$;

(iii) определено умножение элементов $\mathbb{V}$ на булевы константы так, что $0\cdot\boldsymbol{v}=\mathbf{0}$ и $1\cdot\boldsymbol{v}=\boldsymbol{v}$ для каждого $\boldsymbol{v}\in\mathbb{V}$;

Элементы линейного пространства называются векторами. Из (i)–(iii) легко следует единственность нулевого элемента $\mathbf{0}$, а также ряд других свойств. В частности для всех $\alpha,\beta\in\mathbb{B}$ и всех $\boldsymbol{v},\boldsymbol{u}\in\mathbb{V}$ справедливы законы дистрибутивности относительно констант и относительно векторов:

$$(\alpha\oplus\beta)\boldsymbol{v}=\alpha\boldsymbol{v}+\beta\boldsymbol{v},\qquad \alpha(\boldsymbol{v}+\boldsymbol{u})=\alpha\boldsymbol{v}+\alpha\boldsymbol{u}.$$

Рассмотрим некоторые примеры линейных булевых пространств.

Пример 2.1.1. На множестве наборов из $\mathbb{B}^n$ введем две операции: операцию покомпонентного сложения $\oplus$, отображающую булевы наборы $\boldsymbol{u}$ и $\boldsymbol{v}$ в их сумму $\boldsymbol{u}\oplus\boldsymbol{v}$, и операцию покомпонентного умножения $\cdot$ набора на константу из $\mathbb{B}$. Для i-го разряда суммы $\boldsymbol{u}\oplus\boldsymbol{v}$ и i-го разряда произведения $\alpha\cdot\boldsymbol{u}$ $(i=1,2,\ldots,n)$ положим

$$(\boldsymbol{u}\oplus\boldsymbol{v})_i=u_i\oplus v_i,\quad (\alpha\cdot\boldsymbol{u})_i=\alpha u_i.$$

Легко видеть, что множество $\mathbb{B}^n$ с операцией $\oplus$ будет линейным булевым пространством. Так же легко видеть, что нулевым элементом этого пространства будет нулевой набор

$\mathbf{0} = (0, \ldots, 0)$. *Скалярным произведением* $(\boldsymbol{x}, \boldsymbol{y})$ векторов $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{B}^n$ называется величина

$$(\boldsymbol{x}, \boldsymbol{y}) = x_1 y_1 \oplus \ldots \oplus x_i y_i \oplus \ldots \oplus x_n y_n.$$

Векторы называются ортогональными, если их скалярное произведение равно нулю. Отметим, что линейную функцию с нулевым свободным членом $\alpha_1 x_1 \oplus \ldots \oplus \alpha_n x_n$ можно рассматривать как скалярное произведение вектора коэффициентов $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ и вектора переменных $\boldsymbol{x} = (x_1, \ldots, x_n)$. □

Пример 2.1.2. При любом натуральном n линейным булевым пространством является множество $P_2(n)$ всех булевых функций зависящих от n переменных с обычной операцией сложения $\oplus$. Нулевым элементом этого пространства будет тождественный нуль. Легко видеть, при что любом натуральном $k \leq n$ множество всех булевых функций из $P_2(n)$ степени не больше k так же будет линейным булевым пространством. □

Пусть $\mathbb{V}$ — линейное булево пространство, $\mathbb{V}' \subseteq \mathbb{V}$. Множество $\mathbb{V}'$ называется *линейным подпространством* пространства $\mathbb{V}$, если для любых $\boldsymbol{x}$ и $\boldsymbol{y}$ из $\mathbb{V}'$ их сумма $\boldsymbol{x} + \boldsymbol{y}$ также принадлежит $\mathbb{V}'$.

Пусть $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \in \mathbb{V}$, $\alpha_1, \ldots, \alpha_k \in \mathbb{B}$. Вектор $\alpha_1 \boldsymbol{x}_1 + \cdots + \alpha_k \boldsymbol{x}_k$ называется *линейной комбинацией* векторов $\boldsymbol{x}_i$ с коэффициентами α_i. Множество всех линейных комбинаций векторов $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ называется *линейной оболочкой* этих векторов и обозначается через $\langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$. Легко видеть, что для любых векторов $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ их линейная оболочка будет линейным подпространством в $\mathbb{V}$.

Пример 2.1.3. Рассмотрим тождественные функции $x_1, \ldots, x_n$ и тождественную единицу. Линейная оболочка этих функций состоит из всех линейных функций n переменных, т.е $\langle x_1, \ldots, x_n, 1 \rangle = L(n)$, и является линейным подпространством в пространстве $P_2(n)$. □

2. Векторы $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ называются *линейно зависимыми*, если найдутся такие одновременно не равные нулю булевы постоянные $\alpha_1, \ldots, \alpha_k$, что линейная комбинация векторов $\boldsymbol{x}_i$ с коэффициентами α_i равна нулевому набору: $\alpha_1 \boldsymbol{u}_1 + \ldots + \alpha_k \boldsymbol{u}_k = \mathbf{0}$. Если при любых одновременно не равных нулю постоянных α_i линейная комбинация векторов $\boldsymbol{x}_i$ с коэффициентами α_i не равна нулевому набору, то векторы $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ называются *линейно независимыми*.

Лемма 2.1.1. *Пусть $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ и $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ — линейно независимые системы векторов, и каждый $\boldsymbol{y}_i \in \langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$. Тогда $m \leq k$.*

Доказательство. Допустим, что утверждение леммы не верно и $m > k$. Так как каждый набор $\boldsymbol{y}_i \in \langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$, то $\langle \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m \rangle \subseteq \langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$. Очевидно, что $\langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$ содержит ровно $2^k - 1$ ненулевых векторов, а $\langle \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m \rangle$ — $2^m - 1$ ненулевых векторов. Поэтому при $m > k$ в $\langle \boldsymbol{y}_1, \ldots, \boldsymbol{y}_m \rangle$ найдутся две различные ненулевые линейные комбинации, равные одной и той же линейной комбинации из $\langle \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k \rangle$:

$$\alpha_1 \boldsymbol{y}_1 + \cdots + \alpha_m \boldsymbol{y}_m = \gamma_1 \boldsymbol{x}_1 + \cdots + \gamma_k \boldsymbol{x}_k,$$
$$\beta_1 \boldsymbol{y}_1 + \cdots + \beta_m \boldsymbol{y}_m = \gamma_1 \boldsymbol{x}_1 + \cdots + \gamma_k \boldsymbol{x}_k.$$

Складывая левые и правые части этих равенств, получаем, что

$$(\alpha_1 \oplus \beta_1) \boldsymbol{y}_1 + \cdots + (\alpha_m \oplus \beta_m) \boldsymbol{y}_m = \mathbf{0},$$

причем среди чисел $\alpha_i \oplus \beta_i$ обязательно найдется хотя бы одно равное единице. Следовательно, векторы $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_m$ линейно зависимы. Противоречие. Таким образом $m \leq k$. Лемма доказана.

Пусть $\mathbb{V}$ — линейное булево пространство. Система векторов $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ называется *базисом* в $\mathbb{V}$, если эти векторы линейно независимы и их линейная оболочка совпадает с $\mathbb{V}$.

Например, в пространстве $\mathbb{B}^n$ базисом будет следующая система векторов E_n:

$$\begin{aligned}
&\boldsymbol{e}_1 = (1,0,0,\ldots,0,0),\\
&\boldsymbol{e}_2 = (0,1,0,\ldots,0,0),\\
&\boldsymbol{e}_3 = (0,0,1,\ldots,0,0),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\\
&\boldsymbol{e}_n = (0,0,0,\ldots,0,1),
\end{aligned}$$

в которой каждый из векторов содержит ровно одну единичную компоненту. Базис E_n называется стандартным базисом в $\mathbb{B}^n$.

Пример 2.1.4. Найдем число различных базисов в $\mathbb{B}^n$. Прежде всего заметим, что каждый базис в $\mathbb{B}^n$ состоит ровно из n векторов. Это легко следует из леммы 2.1.1 и существования базиса E_n.

В $\mathbb{B}^n$ выберем n линейно независимых векторов. Сделаем это последовательно выбирая векторы из $\mathbb{B}^n$ так, чтобы выбранный на очередном шаге вектор не принадлежал линейной оболочке ранее выбранных векторов. Первый вектор $\boldsymbol{v}_1$ можно выбрать 2^n-1 способами: подойдет любой ненулевой вектор. Второй вектор $\boldsymbol{v}_2$ можно выбрать 2^n-2 способами: подойдет любой ненулевой вектор отличный от $\boldsymbol{v}_1$. На k-м шаге можно выбрать любой вектор не принадлежащий линейной оболочке векторов $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_{k-1}$. Так как линейная оболочка $\langle \boldsymbol{v}_1,\ldots,\boldsymbol{v}_{k-1}\rangle$ состоит из 2^{k-1} векторов, то вектор $\boldsymbol{v}_k$ можно выбрать 2^n-2^{k-1} способами. Таким образом, все n векторов можно выбрать

$$(2^n-1)(2^n-2)\cdot\ldots\cdot(2^n-2^k)\cdot\ldots\cdot(2^n-2^{n-1}) \tag{2.1.1}$$

способами. Наконец заметим, что одна и та же система векторов $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n$ будет выбрана $n!$ раз, так как векторы могут выбираться в разном порядке. Следовательно, число различных базисов в $\mathbb{B}^n$ равно произведению (2.1.1) деленному на $n!$. $\square$

Так как линейное булево пространство $\mathbb{V}$ состоит из конечного числа векторов ($\mathbb{B}^n$ состоит из 2^n векторов), то очевидно, что в $\mathbb{V}$ найдутся векторы, линейная оболочка которых совпадает с $\mathbb{V}$. Отсюда и из леммы 2.1.1 легко получаем следующее утверждение.

Теорема 2.1.1. *В каждом линейном булевом пространстве $\mathbb{V}$ найдется базис. Все базисы $\mathbb{V}$ состоят из одинакового числа векторов.*

Будем говорить, что векторы $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_k$ порождают линейное булево пространство $\mathbb{V}$, если $\mathbb{V}=\langle \boldsymbol{v}_1,\ldots,\boldsymbol{v}_k\rangle$. Очевидно, что любое линейное пространство порождается своим базисом, но не всякая система векторов, порождающая линейное пространство, является базисом этого пространства. Например система векторов $\boldsymbol{e}_1,\ldots,\boldsymbol{e}_n,\boldsymbol{e}_1\oplus\boldsymbol{e}_2$ порождает пространство $\mathbb{B}^n$, но не является его базисом.

Число векторов в базисе пространства $\mathbb{V}$ называется *размерностью* пространства $\mathbb{V}$ и обозначается через $\dim\mathbb{V}$. Пространство размерности k часто называют k-мерным пространством.

Очевидно, что каждый вектор $\boldsymbol{v}$, лежащий в k-мерном пространстве $\mathbb{V}$, представляется в виде линейной комбинации

$$\boldsymbol{v}=\alpha_1\boldsymbol{v}_1+\cdots+\alpha_k\boldsymbol{v}_k \tag{2.1.2}$$

базисных векторов $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_k$ этого пространства. Величины α_i называются координатами вектора $\boldsymbol{v}$ в базисе $\{\boldsymbol{v}_1,\ldots,\boldsymbol{v}_k\}$. Нетрудно видеть, что представление (2.1.2) единственно. Действительно, допустим, что найдутся две различные линейные комбинации базисных векторов, каждая из которых равна вектору $\boldsymbol{v}$. Тогда

$$\alpha_1\boldsymbol{v}_1+\cdots+\alpha_k\boldsymbol{v}_k=\beta_1\boldsymbol{v}_1+\cdots+\beta_k\boldsymbol{v}_k.$$

Складывая эти линейные комбинации, получаем, что

$$(\alpha_1\oplus\beta_1)\boldsymbol{v}_1+\cdots+(\alpha_k\oplus\beta_k)\boldsymbol{v}_k=\boldsymbol{0},$$

причем среди сумм $\alpha_i \oplus \beta_i$ обязательно найдется хотя бы одна равная единице. Следовательно, векторы $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ линейно зависимы, т.е. эти векторы не образуют базис в $\mathbb{V}$. Противоречие.

3. Линейное булево пространство $\mathbb{V}$ с операцией сложения $+$ и линейное булево пространство $\mathbb{W}$ с операцией сложения $\star$ называются *изоморфными*, если существует такое взаимнооднозначное отображение $f : \mathbb{V} \to \mathbb{W}$, что

$$f(\boldsymbol{v} + \boldsymbol{u}) = f(\boldsymbol{v}) \star f(\boldsymbol{u})$$

для любых $\boldsymbol{v}, \boldsymbol{u} \in \mathbb{V}$.

Теорема 2.1.2. *Любое линейное булево пространство $\mathbb{V}$ с операцией сложения $+$ при подходящем n изоморфно пространству $\mathbb{B}^n$ с операцией сложения $\oplus$.*

Доказательство. Из теоремы 2.1.1 следует, что в пространстве $\mathbb{V}$ найдется некоторый базис $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$. Определим отображение $f : \mathbb{V} \to \mathbb{B}^n$ так, что

$$f(\alpha_1 \boldsymbol{v}_1 + \cdots + \alpha_n \boldsymbol{v}_n) = \alpha_1 \boldsymbol{e}_1 \oplus \cdots \oplus \alpha_n \boldsymbol{e}_n$$

для каждого $\alpha_1 v_1 + \cdots + \alpha_n \boldsymbol{v}_n$ из $\mathbb{V}$. Тогда для любых $\boldsymbol{v}$ и $\boldsymbol{u}$ из $\mathbb{V}$ имеем

$$\begin{aligned} f(\boldsymbol{v} + \boldsymbol{u}) &= f((\alpha_1 \boldsymbol{v}_1 + \cdots + \alpha_n \boldsymbol{v}_n) + (\beta_1 \boldsymbol{v}_1 + \cdots + \beta_n \boldsymbol{v}_n)) = \\ &= f((\alpha_1 \oplus \beta_1)\boldsymbol{v}_1 + \cdots + (\alpha_n \oplus \beta_n)\boldsymbol{v}_n) = (\alpha_1 \oplus \beta_1)\boldsymbol{e}_1 \oplus \cdots \oplus (\alpha_n \oplus \beta_n)\boldsymbol{e}_n = \\ &= (\alpha_1 \boldsymbol{e}_1 \oplus \cdots \oplus \alpha_n \boldsymbol{e}_n) \oplus (\beta_1 \boldsymbol{e}_1 \oplus \cdots \oplus \beta_n \boldsymbol{e}_n) = f(\boldsymbol{v}) \oplus f(\boldsymbol{u}). \end{aligned}$$

Теорема доказана.

4. Рассмотрим произвольное линейное пространство $\mathbb{V}$ и его подпространство $\mathbb{W}$. *Смежным классом* пространства $\mathbb{V}$ по подпространству $\mathbb{W}$ называется множество $\boldsymbol{\alpha} + \mathbb{W}$, состоящее из всех векторов вида $\boldsymbol{\alpha} + \boldsymbol{w}$, где $\boldsymbol{\alpha}$ — фиксированный вектор из $\mathbb{V}$, а $\boldsymbol{w}$ — вектор из $\mathbb{W}$. Вектор $\boldsymbol{\alpha}$ называется *представителем* смежного класса $\boldsymbol{\alpha} + \mathbb{W}$.

Теорема 2.1.3. *Любые два смежных класса пространства $\mathbb{V}$ по подпространству $\mathbb{W}$: (i) не пересекаются или совпадают; (ii) содержат одинаковое число векторов.*

Доказательство. Допустим, что смежные классы $\boldsymbol{\alpha} + \mathbb{W}$ и $\boldsymbol{\beta} + \mathbb{W}$ имеют общий вектор $\boldsymbol{\gamma}$. Тогда в $\mathbb{W}$ найдутся векторы $\boldsymbol{w}_1$ и $\boldsymbol{w}_2$ такие, что $\boldsymbol{\gamma} = \boldsymbol{\alpha} + \boldsymbol{w}_1 = \boldsymbol{\beta} + \boldsymbol{w}_2$. Таким образом, $\boldsymbol{\beta} = \boldsymbol{\alpha} + \boldsymbol{w}_1 + \boldsymbol{w}_2$, и для любого вектора $\boldsymbol{\beta} + \boldsymbol{w}$ из смежного класса $\boldsymbol{\beta} + \mathbb{W}$ имеем

$$\boldsymbol{\beta} + \boldsymbol{w} = \boldsymbol{\alpha} + (\boldsymbol{w}_1 + \boldsymbol{w}_2 + \boldsymbol{w}),$$

т.е. $\boldsymbol{\beta} + \boldsymbol{w} \in \boldsymbol{\alpha} + \mathbb{W}$ и, следовательно, $\boldsymbol{\beta} + \mathbb{W} \subseteq \boldsymbol{\alpha} + \mathbb{W}$. Аналогично доказывается включение $\boldsymbol{\alpha} + \mathbb{W} \subseteq \boldsymbol{\beta} + \mathbb{W}$. Таким образом, $\boldsymbol{\beta} + \mathbb{W} = \boldsymbol{\alpha} + \mathbb{W}$. Утверждение ($i$) доказано.

Второе утверждение теоремы легко следует из того, что все суммы $\boldsymbol{\alpha} + \boldsymbol{w}$ различны при фиксированном $\boldsymbol{\alpha}$ и различных $\boldsymbol{w}$. Теорема доказана.

Пример 2.1.5. В пространстве $P_2(2)$, состоящем из всех булевых функций, зависящих от двух переменных x и y, функции, не зависящие от переменной y, образуют линейное подпространство, которое, очевидно, состоит из четырех функций: 0, 1, x и $x \oplus 1$. Обозначим это подпространство через $\mathbb{V}_x$. Построим смежные классы пространства $P_2(2)$ по подпространству $\mathbb{V}_x$.

Так как $P_2(2)$ состоит из 16 функций, то в $P_2(2)$ будет ровно четыре смежных класса по $\mathbb{V}_x$. Первым смежным классом будет само подпространство $\mathbb{V}_x$. Для определения второго смежного класса в качестве его представителя выберем не принадлежащую $\mathbb{V}_x$ функцию — y. Тогда $y \oplus \mathbb{V}_x = \{y, y \oplus 1, y \oplus x, y \oplus x \oplus 1\}$. Представителем третьего смежного класса будет функция не принадлежащая объединению первых двух. Такой функцией будет, например, xy. Тогда $xy \oplus \mathbb{V}_x = \{xy, xy \oplus 1, xy \oplus x, xy \oplus x \oplus 1\}$. Наконец последний четвертый смежный класс состоит из четырех оставшихся функций, т.е. из $y \oplus xy, 1 \oplus y \oplus xy, x \oplus y \oplus xy$ и $1 \oplus x \oplus xy \oplus$. $\square$

Задачи

2.1.1. Показать, что в любом булевом линейном пространстве $\mathbb{V}$ для всех $\alpha, \beta \in \mathbb{B}$ и всех $\boldsymbol{v}, \boldsymbol{u} \in \mathbb{V}$ справедливы законы дистрибутивности относительно констант и относительно векторов:

$$(\alpha \oplus \beta)\boldsymbol{v} = \alpha\boldsymbol{v} + \beta\boldsymbol{v}, \qquad \alpha(\boldsymbol{v} + \boldsymbol{u}) = \alpha\boldsymbol{v} + \alpha\boldsymbol{u}.$$

2.1.2. Показать, что n-мерное линейное булево пространство состоит ровно из 2^n элементов.

2.1.3. Пусть $\mathbb{V}$ — подмножество $\mathbb{B}^n$, состоящее из всех наборов четного веса. Показать, что $\mathbb{V}$ является линейным подпространством в $\mathbb{B}^n$ и указать какой-либо его базис.

2.1.4. Пусть $\mathbb{W} \subset \mathbb{V}$ — подпространство пространства $\mathbb{V}$, $W = \{\boldsymbol{w}_1, \ldots, \boldsymbol{w}_k\}$ — базис в $\mathbb{W}$. Показать, что базис W можно дополнить до базиса пространства $\mathbb{V}$.

2.1.5. Пусть $\mathbb{W}$ и $\mathbb{V}$ такие пространства, что $\mathbb{W} \subseteq \mathbb{V}$ и $\dim\mathbb{W} = \dim\mathbb{V}$. Показать, что $\mathbb{W} = \mathbb{V}$.

2.1.6. Показать, что если $\mathbb{W}$ и $\mathbb{V}$ такие подпространства $\mathbb{B}^n$, что $\mathbb{W} \cap \mathbb{V} = \{\mathbf{0}\}$, то $\dim\mathbb{W} + \dim\mathbb{V} \leq n$.

2.1.7. Найти в $\mathbb{B}^n$ число различных линейных подпространств размерности k, $1 \leq k \leq n-1$.

2.1.8. Найти в $\mathbb{B}^n$ число различных линейных подпространств размерности k, $1 \leq k \leq n-1$, содержащих данный ненулевой набор $\boldsymbol{\alpha}$.

2.1.9. Будет ли множество симметрических функций линейным подпространством в $P_2(n)$? Если будет, то найти его размерность и указать какой-либо базис.

2.2. Линейные операторы

1. Оператор $f : \mathbb{B}^n \to \mathbb{B}^m$ называется *линейным*, если каждая его компонента является линейной функцией с нулевым свободным членом. Множество линейных операторов из $\mathbb{B}^n$ в $\mathbb{B}^m$ обозначим через $\mathcal{L}(m,n)$. Каждый оператор из $\mathcal{L}(m,n)$ будем называть линейным (m,n)-оператором. Легко видеть, что для каждого линейного оператора f, определенного на $\mathbb{B}^n$, и всех $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{B}^n$ справедливо равенство

$$f(\boldsymbol{u} \oplus \boldsymbol{v}) = f(\boldsymbol{u}) \oplus f(\boldsymbol{v}). \tag{2.2.1}$$

Для доказательства этого равенства достаточно убедится в его справедливости отдельно для каждой компоненты оператора f.

Пусть $V = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\}$ — произвольный базис в $\mathbb{B}^n$, $\boldsymbol{u}$ — произвольный вектор из $\mathbb{B}^n$, $u_1, \ldots, u_n$ — координаты вектора $\boldsymbol{u}$ в базисе V. Тогда из (2.2.1) имеем

$$f(\boldsymbol{u}) = f(u_1\boldsymbol{v}_1 \oplus \ldots \oplus u_n\boldsymbol{v}_n) = u_1 f(\boldsymbol{v}_1) \oplus \ldots \oplus u_n f(\boldsymbol{v}_n).$$

Следовательно, значение линейного оператора однозначно определяется его значениями на векторах базиса.

Оператор $e_n : \mathbb{B}^n \to \mathbb{B}^n$ называется тождественным, если $e_n(\boldsymbol{x}) = \boldsymbol{x}$ для каждого $\boldsymbol{x} \in \mathbb{B}^n$. Легко видеть, что тождественный оператор является линейным. Оператор $f : \mathbb{B}^n \to \mathbb{B}^n$ называется *невырожденным*, если $f(\boldsymbol{x}) \neq f(\boldsymbol{y})$ для всех $\boldsymbol{x} \neq \boldsymbol{y}$. Для всякого невырожденного оператора f найдется единственный *обратный* оператор f^{-1} такой, что

$$f^{-1}(f(\boldsymbol{x})) = f(f^{-1}(\boldsymbol{x})) = \boldsymbol{x}$$

для каждого x из $\mathbb{B}^n$. Можно показать, что оператор обратный к линейному невырожденному оператору так же будет линейным.

Композицией $h \circ f$ линейного (k,m)-оператора h и линейного (m,n)-оператора f называется такой (k,n)-оператор g, что

$$g(\boldsymbol{x}) = h(f(\boldsymbol{x})) \tag{2.2.2}$$

для каждого $\boldsymbol{x}$ из $\mathbb{B}^n$. Заметим, что композиция $h \circ f$ операторов h и f определена только в том случае, когда число компонент оператора f равно числу переменных оператора h.

Покажем, что композиция линейных операторов является линейным оператором. Рассмотрим композицию $g(\boldsymbol{x}) = h(f(\boldsymbol{x}))$. Пусть компоненты операторов f и h определяются следующими равенствами:

$$f_i = f_{i1}x_1 \oplus \ldots \oplus f_{ij}x_j \oplus \ldots \oplus f_{in}x_n, \qquad i = 1,2,\ldots,m; \tag{2.2.3}$$

$$h_i = h_{i1}y_1 \oplus \ldots \oplus h_{ij}y_j \oplus \ldots \oplus h_{im}y_m, \qquad i = 1,2,\ldots,k. \tag{2.2.4}$$

Выразим компоненты g_i оператора g через коэффициенты операторов h и f. Для этого компоненты оператора f из (2.2.3) подставим в (2.2.4) вместо переменных $y_1,\ldots,y_m$. В результате при $i = 1,2,\ldots,k$ для i-компоненты g получим

$$\begin{aligned} g_i &= h_{i1}(f_{11}x_1 \oplus \ldots \oplus f_{1t}x_t \oplus \ldots \oplus f_{1n}x_n) \oplus \ldots \\ &\quad \ldots \oplus h_{im}(f_{m1}x_1 \oplus \ldots \oplus f_{mt}x_t \oplus \ldots \oplus f_{mn}x_n) = \\ &= (h_{i1}f_{11} \oplus \ldots \oplus h_{it}f_{t1} \oplus \ldots \oplus h_{im}f_{m1})x_1 \oplus \ldots \\ &\quad \ldots \oplus (h_{i1}f_{1n} \oplus \ldots \oplus h_{it}f_{tn} \oplus \ldots \oplus h_{im}f_{mn})x_n. \end{aligned}$$

Следовательно, каждая компонента g_i оператора g является линейной функцией, для j-го коэффициента которой справедливо равенство

$$g_{ij} = h_{i1}f_{1j} \oplus \ldots \oplus h_{it}f_{tj} \oplus \ldots \oplus h_{im}f_{mj}. \tag{2.2.5}$$

Таким образом, композиция линейных операторов является линейным оператором.

Суммой линейных (m,n)-операторов f и h называется такой (m,n)-оператор $f \oplus h$, что

$$(f \oplus h)(\boldsymbol{x}) = f(\boldsymbol{x}) \oplus h(\boldsymbol{x})$$

для каждого $\boldsymbol{x} \in \mathbb{B}^n$. Очевидно, что сумма линейных операторов так же будет линейным оператором.

2. *Ядром* линейного (m,n)-оператора f называется множество всех таких $\boldsymbol{x} \in \mathbb{B}^n$, для которых $f(\boldsymbol{x}) = \boldsymbol{0}$. Ядро оператора f обозначается через $\ker f$. *Образом* оператора f называется множество всех таких $\boldsymbol{y} \in \mathbb{B}^m$, что $\boldsymbol{y} = f(\boldsymbol{x})$. Образ оператора f обозначается через $\operatorname{Im} f$. Нетрудно показать, что ядро и образ любого линейного (m,n)-оператора являются линейными подпространствами в $\mathbb{B}^n$ и $\mathbb{B}^m$ соответственно. Размерность образа линейного оператора f называется его *рангом* и обозначается через $\operatorname{rank} f$.

Теорема 2.2.1. *Для любого линейного (m,n)-оператора f*

$$n = \dim\ker f + \dim\operatorname{Im} f.$$

Доказательство. Пусть векторы $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_k$ образуют базис в $\ker f$. Произвольным образом дополним этот базис до базиса всего пространства $\mathbb{B}^n$. Новые базисные векторы обозначим через $\boldsymbol{v}_{k+1},\ldots,\boldsymbol{v}_n$. Теперь для доказательства теоремы достаточно показать, что образ оператора f порождается векторами $f(\boldsymbol{v}_{k+1}),\ldots,f(\boldsymbol{v}_n)$, т.е.

$$\operatorname{Im} f = \langle f(\boldsymbol{v}_{k+1}),\ldots,f(\boldsymbol{v}_n)\rangle.$$

Прежде всего убедимся, что векторы $f(\boldsymbol{v}_{k+1}),\ldots,f(\boldsymbol{v}_n)$ линейно независимы. Действительно, если это не так, то найдутся такие одновременно не равные нулю постоянные $\alpha_{k+1},\ldots,\alpha_n$, что

$$\alpha_{k+1}f(\boldsymbol{v}_{k+1}) \oplus \ldots \oplus \alpha_n f(\boldsymbol{v}_n) = \boldsymbol{0}.$$

Откуда, используя линейность оператора f, легко получаем, что

$$\mathbf{0} = \alpha_{k+1} f(\boldsymbol{v}_{k+1}) \oplus \ldots \oplus \alpha_n f(\boldsymbol{v}_n) = f(\alpha_{k+1}\boldsymbol{v}_{k+1} \oplus \ldots \oplus \alpha_n \boldsymbol{v}_n),$$

т.е. вектор $(\alpha_{k+1}\boldsymbol{v}_{k+1} \oplus \ldots \oplus \alpha_n \boldsymbol{v}_n)$ принадлежит ядру оператора f. Противоречие с выбором векторов $\boldsymbol{v}_{k+1}, \ldots, \boldsymbol{v}_n$. Следовательно, векторы $f(\boldsymbol{v}_{k+1}), \ldots, f(\boldsymbol{v}_n)$ линейно независимы.

Теперь покажем, что каждый вектор из образа f выражается в виде линейной комбинации векторов $f(\boldsymbol{v}_{k+1}), \ldots, f(\boldsymbol{v}_n)$. Для этого произвольный вектор $\boldsymbol{\alpha}$ из $\mathbb{B}^n$ разложим по базису $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ и применим к этому вектору оператор f:

$$\begin{aligned} f(\boldsymbol{\alpha}) &= f(\alpha_1\boldsymbol{v}_1 \oplus \ldots \oplus \alpha_n\boldsymbol{v}_n) = \\ &= f(\alpha_1\boldsymbol{v}_1 \oplus \ldots \oplus \alpha_k\boldsymbol{v}_k) \oplus f(\alpha_{k+1}\boldsymbol{v}_{k+1} \oplus \ldots \oplus \alpha_n\boldsymbol{v}_n) = \\ &= \alpha_{k+1} f(\boldsymbol{v}_{k+1}) \oplus \ldots \oplus \alpha_n f(\boldsymbol{v}_n), \end{aligned}$$

Таким образом, любой элемент из $\operatorname{Im} f$ выражается в виде линейной комбинации векторов $f(\boldsymbol{v}_{k+1}), \ldots, f(\boldsymbol{v}_n)$. Так как эти векторы линейно независимы, то они образуют базис в $\operatorname{Im} f$. Следовательно, $\dim \operatorname{Im} f = n - k$. Теорема доказана.

Теорема 2.2.2. *Для любого подпространства $\mathbb{V}$ пространства $\mathbb{B}^n$ найдется определенный на $\mathbb{B}^n$ линейный оператор f такой, что $\mathbb{V} = \ker f$.*

Доказательство. Пусть $k = \dim \mathbb{V}$, и пусть векторы $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ образуют базис в $\mathbb{V}$. Рассмотрим линейный (k, n)-оператор $v = (v_1, \ldots, v_k)$, где

$$v_i = v_{i1}x_1 \oplus \ldots \oplus v_{ij}x_j \oplus \ldots \oplus v_{in}x_n,$$

$i = 1, 2, \ldots, k$ и v_{ij} — j-я координата вектора $\boldsymbol{v}_i$. Пусть $\mathbb{W}$ — ядро оператора v. Из предыдущей теоремы следует, что $\dim \mathbb{W} = n - k$. Пусть $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{n-k}$ — базис в $\mathbb{W}$. Очевидно, что

$$(\boldsymbol{v}_i, \boldsymbol{w}_j) = 0, \tag{2.2.6}$$

для всех $i = 1, \ldots, k$ и всех $j = 1, \ldots, n - k$. Рассмотрим линейный $(n - k, n)$-оператор $w = (w_1, \ldots, w_{n-k})$, где

$$w_i = w_{i1}x_1 \oplus \ldots w_{ij}x_j \oplus \ldots \oplus w_{in}x_n,$$

$i = 1, 2, \ldots, n - k$ и w_{ij} — j-я координата вектора $\boldsymbol{w}_i$. Из (2.2.6) следует, что $\mathbb{V} \subseteq \ker w$. В силу теоремы 2.2.1 имеет место равенство $\dim w = k$. Поэтому $\mathbb{V} = \ker w$. Теорема доказана.

Задачи

2.2.1. Оператор $f : \mathbb{B}^n \to \mathbb{B}^m$ называется *аффинным* (m, n)-оператором, если каждая его компонента является линейной функцией (т.е. ее свободный член не обязательно равен нулю). Найти число аффинных (m, n)-операторов.

2.2.2. Найти число различных линейных булевых операторов, отображающих $\mathbb{B}^n$ в $\mathbb{B}^m$, которые переводят фиксированный набор $\boldsymbol{\alpha}$ в нулевой набор.

2.2.3. Показать, что ядро любого линейного (m, n)-оператора является подпространством в $\mathbb{B}^n$.

2.2.4. Показать, что образ любого линейного (m, n)-оператора является подпространством в $\mathbb{B}^m$.

2.2.5. Доказать, что оператор обратный к линейному невырожденному оператору так же будет линейным оператором.

2.3. Матрицы

1. Удобным средством описания линейных операторов являются матрицы. *Матрицей линейного* (m,n)*-оператора* $f=(f_1,\dots,f_m)$ с компонентами

$$\begin{aligned}
f_1 &= f_{11}x_1 \oplus f_{12}x_2 \oplus \dots\dots \oplus f_{1n}x_n,\\
f_2 &= f_{21}x_1 \oplus f_{22}x_2 \oplus \dots\dots \oplus f_{1n}x_n,\\
&\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\\
f_m &= f_{m1}x_1 \oplus f_{m2}x_2 \oplus \dots\dots \oplus f_{mn}x_n,
\end{aligned}$$

называется прямоугольная таблица из m строк и n столбцов

$$\mathbf{F}=\begin{pmatrix}
f_{11} & \cdots & f_{1j} & \cdots & f_{1n}\\
\dots\dots & \dots\dots & \dots\dots & \dots\dots & \dots\dots\\
f_{i1} & \dots & f_{ij} & \dots & f_{in}\\
\dots\dots & \dots\dots & \dots\dots & \dots\dots & \dots\dots\\
f_{m1} & \cdots & f_{mj} & \cdots & f_{mn}
\end{pmatrix}, \tag{2.3.1}$$

составленная из коэффициентов f_{ij} оператора f. Величины f_{ij} называются элементами матрицы (2.3.1). Булеву матрицу из m строк и n столбцов будем называть *булевой* (m,n)*-матрицей* $\mathbf{F}=(f_{ij})$ или булевой матрицей размера $m\times n$. Если $m=n$, то булеву (n,n)-матрицу будем называть квадратной матрицей порядка n, или просто матрицей порядка n. В частности, легко видеть, что матрицей тождественного оператора будет квадратная матрица $\mathbf{E}_n=(e_{ij})$, элементы которой определяется равенством

$$e_{ij}=\begin{cases}1, & \text{при } i=j,\\ 0, & \text{при } i\neq j.\end{cases}$$

Матрица $\mathbf{E}_n$ называется *единичной* матрицей порядка n.

Матрица $\mathbf{F}^{\mathrm{T}}=(f_{ij}^{\mathrm{T}})$ размера $m\times n$ называется транспонированной матрицей $\mathbf{F}=(f_{ij})$ размера $n\times m$, если $f_{ij}^{\mathrm{T}}=f_{ji}$. Легко видеть, что для любой матрицы $\mathbf{F}$ справедливо равенство $\left(\mathbf{F}^{\mathrm{T}}\right)^{\mathrm{T}}=\mathbf{F}$.

2. Определим операции сложения и умножения булевых матриц. Сделаем это так, что бы матрица суммы $h\oplus g$ линейных операторов h и g была равна сумме матриц этих операторов, а матрица композиции $h\circ f$ линейных операторов h и f — произведению матриц этих операторов.

Суммой $\mathbf{A}\oplus\mathbf{B}$ двух (m,n)-матриц $\mathbf{A}=(a_{ij})$ и $\mathbf{B}=(b_{ij})$ называется такая (m,n)-матрица $\mathbf{C}=(c_{ij})$, что $c_{ij}=a_{ij}\oplus b_{ij}$. Нетрудно видеть, что матрица суммы двух линейных операторов равна сумме матриц этих операторов.

Произведением $\mathbf{BA}$ (m,n)-матрицы $\mathbf{A}=(a_{ij})$ и (k,n)-матрицы $\mathbf{B}=(b_{ij})$ называется такая (k,n)-матрица $\mathbf{C}=(c_{ij})$, что[1)]

$$c_{ij}=b_{i1}a_{1j}\oplus\dots\oplus b_{it}a_{tj}\oplus\dots\oplus b_{im}a_{mj}.$$

Если матрицы $\mathbf{A}$ и $\mathbf{B}$ являются матрицами операторов f и h, заданных равенствами (2.2.3) и (2.2.4), то из определения произведения матриц и равенства (2.2.5) легко следует, что матрица $\mathbf{BA}$ действительно будет матрицей композиции $h\circ f$.

Если рассматривать набор $\boldsymbol{x}$ из $\mathbb{B}^n$ в качестве матрицы, состоящей из единственного столбца высоты n, то из определения матриц легко следует, что значение линейного (m,n)-оператора f на наборе $\boldsymbol{x}$ можно найти вычислив произведение $\mathbf{F}\boldsymbol{x}$ матрицы этого оператора и набора $\boldsymbol{x}$. Далее, говоря о произведении $\mathbf{F}\boldsymbol{x}$, набор $\boldsymbol{x}$ будем иногда называть *вектором-столбцом* для того, что бы подчеркнуть его "вертикальное" положение в этом произведении.

[1)] Заметим, что элемент c_{ij} равен скалярному произведению i-й строки матрицы $\mathbf{B}$ и j-го столбца матрицы $\mathbf{A}$.

Пример 2.3.1. Рассмотрим оператор $f = (f_1, f_2)$ с компонентами $f_1 = x_1 \oplus x_2 \oplus x_3$ и $f_2 = x_1 \oplus x_2$, Найдем его значение при $x_1 = x_2 = x_3 = 1$. Сделаем это, умножив матрицу оператора f на вектор-столбец (111):

$$f(111) = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Следовательно, образом вектора (111) является вектор (10), т.е. $f_1 = 1$, $f_2 = 0$. Аналогичный результат получается после вычисления $f(1,1,1)$ по формулам его компонент. □

Как и при вычислении композиции линейных операторов, произведение $\mathbf{BA}$ матриц $\mathbf{B}$ и $\mathbf{A}$ определено не всегда, а только в том случае, когда число столбцов матрицы $\mathbf{B}$ равно числу строк матрицы $\mathbf{A}$. Поэтому легко видеть, что произведения $\mathbf{AB}$ и $\mathbf{BA}$ матриц $\mathbf{B}$ и $\mathbf{A}$ определены одновременно только для квадратных матриц одного порядка. Отметим, что операция умножения квадратных матриц не коммутативна, т.е. найдутся такие матрицы $\mathbf{A}$ и $\mathbf{B}$ одного порядка, что $\mathbf{AB} \neq \mathbf{BA}$.

Пример 2.3.2. Рассмотрим две треугольных булевых матрицы второго порядка $\mathbf{A} = \left(\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right)$ и $\mathbf{B} = \left(\begin{smallmatrix} 1 & 1 \\ 0 & 1 \end{smallmatrix}\right)$. Для произведений $\mathbf{AB}$ и $\mathbf{BA}$ этих матриц справедливы равенства:

$$\mathbf{AB} = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}, \qquad \mathbf{BA} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Таким образом, $\mathbf{AB} \neq \mathbf{BA}$. □

Непосредственно из определений операции умножения и сложения булевых матриц следует, что каждая из этих операций ассоциативна, и кроме того эти операции связаны законами дистрибутивности: для любой матрицы $\mathbf{A}$ размера $m \times n$, любых матриц $\mathbf{B}$ и $\mathbf{C}$ размера $n \times k$ и любой матрицы $\mathbf{D}$ размера $k \times l$ справедливы равенства

$$\mathbf{A}(\mathbf{B} \oplus \mathbf{C}) = \mathbf{AB} \oplus \mathbf{AC}, \qquad (\mathbf{B} \oplus \mathbf{C})\mathbf{D} = \mathbf{BD} \oplus \mathbf{CD}.$$

Матрицы можно разбивать на блоки — матрицы меньших размеров, и, при условии подходящего выбора размеров блоков, выполнять поблочное сложение и умножение матриц. Например, рассмотрим разбитые на четыре блока (m,n)-матрицу $\mathbf{A}$ и (n,k)-матрицу $\mathbf{B}$:

$$\mathbf{A} = \left(\begin{array}{c|c} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \hline \mathbf{A}_{21} & \mathbf{A}_{22} \end{array}\right), \qquad \mathbf{B} = \left(\begin{array}{c|c} \mathbf{B}_{11} & \mathbf{B}_{12} \\ \hline \mathbf{B}_{21} & \mathbf{B}_{22} \end{array}\right)$$

Если эти матрицы разбиты на блоки так, что для всех $1 \leq i,t,j \leq 2$ определены произведения $\mathbf{A}_{it}\mathbf{B}_{tj}$, то для произведения $\mathbf{C} = \mathbf{AB}$ справедливо равенство

$$\mathbf{C} = \left(\begin{array}{c|c} \mathbf{A}_{11}\mathbf{B}_{11} \oplus \mathbf{A}_{12}\mathbf{B}_{21} & \mathbf{A}_{11}\mathbf{B}_{12} \oplus \mathbf{A}_{12}\mathbf{B}_{22} \\ \hline \mathbf{A}_{21}\mathbf{B}_{11} \oplus \mathbf{A}_{22}\mathbf{B}_{21} & \mathbf{A}_{21}\mathbf{B}_{12} \oplus \mathbf{A}_{22}\mathbf{B}_{22} \end{array}\right).$$

3. С каждой булевой матрицей свяжем три линейных пространства: пространство строк, пространство столбцов, и ортогональное пространство. *Пространством строк* булевой матрицы $\mathbf{A}$ называется линейное пространство $\mathbb{A}$, порожденное строками этой матрицы. Пространство строк матрицы $\mathbf{A}$ будем обозначать также через $\langle \mathbf{A} \rangle$. Если $\mathbb{A}$ — пространство строк матрицы $\mathbf{A}$, то матрица $\mathbf{A}$ называется *порождающей* матрицей пространства $\mathbb{A}$. *Пространством столбцов* булевой матрицы $\mathbf{A}$ называется линейное пространство, порожденное столбцами этой матрицы. Легко видеть, что для любой булевой матрицы $\mathbf{A}$ пространство ее столбцов совпадает с пространством строк транспонированной матрицы.

Поэтому пространство столбцов матрицы $\mathbf{A}$ будем обозначать через $\langle \mathbf{A}^{\mathrm{T}} \rangle$ и $\mathbb{A}^{\mathrm{T}}$. *Ортогональным пространством* булевой (m,n)-матрицы $\mathbf{A}$ называется линейное пространство $\mathbb{A}^{\perp}$, состоящее из всех тех векторов $\boldsymbol{v} \in \mathbb{B}^n$, для которых $\mathbf{A}\boldsymbol{v} = \mathbf{0}$.

Нетрудно видеть, что ядро и образ любого линейного оператора f совпадают с ортогональным пространством и пространством столбцов матрицы $\mathbf{F}$ этого оператора, т.е.

$$\mathbb{F}^{\perp} = \ker f, \qquad \mathbb{F}^{\mathrm{T}} = \operatorname{Im} f. \tag{2.3.2}$$

Поэтому из теоремы 2.2.1 вытекает следующее утверждение.

Теорема 2.3.1. *Для любой булевой (m,n)-матрицы $\mathbf{A}$ справедливо равенство*

$$\dim \mathbb{A}^{\perp} + \dim \mathbb{A}^{\mathrm{T}} = n.$$

Введем два элементарных преобразования строк булевых матриц: (1) перестановку i-й и j-й строк матрицы и (2) прибавление i-й строки матрицы к ее j-й строке. Если матрица $\mathbf{B}$ получена из матрицы $\mathbf{A}$ при помощи элементарных преобразований строк, то будем говорить, что эти матрицы *эквивалентны*. Эквивалентность матриц $\mathbf{A}$ и $\mathbf{B}$ будем обозначать через $\mathbf{A} \sim \mathbf{B}$.

Теорема 2.3.2. *Для любых эквивалентных булевых матриц $\mathbf{A}$ и $\mathbf{B}$ их ортогональные пространства совпадают, т. е. $\mathbb{A}^{\perp} = \mathbb{B}^{\perp}$.*

Доказательство. Очевидно, что при перестановке компонент линейного оператора его ядро не изменяется. Поэтому в силу первого равенства в (2.3.2) перестановка строк матрицы не изменяет ее ортогональное пространство. Следовательно, для доказательства теоремы достаточно показать, что второе элементарное преобразование строк также не изменяет ортогонального пространства.

Рассмотрим линейное пространство $\mathbb{V}$, порожденное векторами $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$, и линейное пространство $\mathbb{V}'$, порожденное векторами $\boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_k$. Будем полагать, что вторая система получена из первой прибавлением ее i-го вектора к j-му, т.е. $\boldsymbol{v}'_t = \boldsymbol{v}_t$ при $t \neq j$ и $\boldsymbol{v}'_j = \boldsymbol{v}_i \oplus \boldsymbol{v}_j$. Легко видеть, что в этом случае $\boldsymbol{v}_j = \boldsymbol{v}'_i \oplus \boldsymbol{v}'_j$.

Докажем равенство $\mathbb{V}^{\perp} = (\mathbb{V}')^{\perp}$. Для этого рассмотрим произвольный вектор $\boldsymbol{u}$ из пространства $\mathbb{V}^{\perp}$. Очевидно, что $(\boldsymbol{u}, \boldsymbol{v}_t) = 0$ для каждого вектора $\boldsymbol{v}_t$ первой системы. Поэтому,

$$0 = (\boldsymbol{u}, \boldsymbol{v}_j) \oplus (\boldsymbol{u}, \boldsymbol{v}_i) = (\boldsymbol{u}, \boldsymbol{v}_j \oplus \boldsymbol{v}_i) = (\boldsymbol{u}, \boldsymbol{v}'_j).$$

Следовательно, $\boldsymbol{u} \in (\mathbb{V}')^{\perp}$, и, таким образом, пространство $\mathbb{V}^{\perp}$ содержится в пространстве $(\mathbb{V}')^{\perp}$. С другой стороны, если $\boldsymbol{u} \in (\mathbb{V})^{\perp}$, то $(\boldsymbol{u}, \boldsymbol{v}'_t) = 0$ для каждого вектора $\boldsymbol{v}'_t$ второй системы. Поэтому,

$$0 = (\boldsymbol{u}, \boldsymbol{v}'_j) \oplus (\boldsymbol{u}, \boldsymbol{v}'_i) = (\boldsymbol{u}, \boldsymbol{v}'_j \oplus \boldsymbol{v}'_i) = (\boldsymbol{u}, \boldsymbol{v}_j).$$

Следовательно, $\boldsymbol{u} \in \mathbb{V}^{\perp}$, и пространство $(\mathbb{V}')^{\perp}$ содержится в пространстве $\mathbb{V}^{\perp}$. Таким образом, $\mathbb{V}^{\perp} = (\mathbb{V}')^{\perp}$. Теорема доказана.

Теорема 2.3.3. *Для любой булевой (m,n)-матрицы $\mathbf{A}$ справедливо равенство*

$$\dim \mathbb{A} = \dim \mathbb{A}^{\mathrm{T}}. \tag{2.3.3}$$

Число (2.3.3) *называется рангом матрицы $\mathbf{A}$ и обозначается через* $\operatorname{rank} \mathbf{A}$.

Доказательство. Пусть $\dim \mathbb{A} = k$. Без ограничения общности будем полагать, что первые k строк матрицы $\mathbf{A}$ линейно независимы, а остальные $m-k$ строк матрицы являются их линейными комбинациями. При помощи элементарных преобразований строк преобразуем матрицу $\mathbf{A}$ в эквивалентную ей матрицу $\mathbf{B}$, в которой только первые k строк не будут нулевыми. Для этого к каждой из последних $m-k$ строк матрицы $\mathbf{A}$ надо прибавить равную ей линейную комбинацию первых k строк. В силу предыдущей теоремы $\mathbb{A}^{\perp} = \mathbb{B}^{\perp}$. Так как все ненулевые компоненты столбцов матрицы $\mathbf{B}$ сосредоточены в ее

первых k строках, то легко видеть, что пространство столбцов матрицы $\mathbf{B}$ будет изоморфно некоторому подпространству пространства $\mathbb{B}^k$. Поэтому, $\dim \mathbb{B}^{\mathrm{T}} \le k$. Таким образом, из предыдущих рассуждений и теоремы 2.3.1 имеем

$$\dim \mathbb{A}^{\mathrm{T}} = n - \dim \mathbb{A}^{\perp} = n - \dim \mathbb{B}^{\perp} = \dim \mathbb{B}^{\mathrm{T}} \le k = \dim \mathbb{A}.$$

Следовательно, $\dim \mathbb{A}^{\mathrm{T}} \le \dim \mathbb{A}$. Применяя приведенные рассуждения к матрице $\mathbf{A}^{\mathrm{T}}$, легко получаем неравенство $\dim \mathbb{A}^{\mathrm{T}} \ge \dim \mathbb{A}$. Теорема доказана.

Задачи

2.3.1. Показать, что $\operatorname{rank} \mathbf{AB} \le \min(\operatorname{rank} \mathbf{A}, \operatorname{rank} \mathbf{B})$ для любых квадратных матриц одного порядка.

2.3.2. Квадратная матрица $\mathbf{A} = (a_{ij})$ называется верхнетреугольной, если $a_{ij} = 1$ при $i \le j$ и $a_{ij} = 0$ при $i > j$. Квадратная матрица $\mathbf{B} = (b_{ij})$ называется нижнетреугольной, если $b_{ij} = 1$ при $i \ge j$ и $a_{ij} = 0$ при $i < j$. Найти $\operatorname{rank} \mathbf{AB}$, если $\mathbf{A}$ — верхнетреугольная, а $\mathbf{B}$ — нижнетреугольная матрицы порядка n.

2.3.3. Найти число булевых (m, n)-матриц ранга k, если $k \le m \le n$.

2.4. Определители

В этом параграфе ведем функцию det — определитель системы векторов $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n \in \mathbb{B}^n$, равную единице если векторы линейно независимы, и равную нулю если векторы линейно зависимы. Для этого потребуются три леммы. В первой лемме установим необходимые условия, которым должна удовлетворять каждая такая функция. Во второй лемме покажем, что необходимым условиям, сформулированным в первой лемме, удовлетворяет единственная функция, и для этой функции укажем вычисляющую ее формулу. Наконец, в третьей лемме покажем, что определенная во второй лемме функция действительно равна единице на линейно независимых векторах и равна нулю на линейно зависимых.

Прежде всего дадим необходимое определение. Функция m аргументов

$$f : \underbrace{\mathbb{B}^n \times \ldots \times \mathbb{B}^n}_{m \text{ раз}} \to \mathbb{B}$$

называется *линейной* по i-у аргументу, если равенство

$$\begin{aligned} f(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}, \boldsymbol{u}'_i \oplus \boldsymbol{u}''_i, \boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_m) = \\ = f(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}, \boldsymbol{u}'_i, \boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_m) \oplus f(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{i-1}, \boldsymbol{u}''_i, \boldsymbol{u}_{i+1}, \ldots, \boldsymbol{u}_m) \end{aligned} \tag{2.4.1}$$

справедливо при всех $\boldsymbol{u}_j, \boldsymbol{u}'_i, \boldsymbol{u}''_i$ из $\mathbb{B}^n$.

Лемма 2.4.1. *Пусть функция* $\det : (\mathbb{B}^n)^n \to \mathbb{B}^n$ *равна единице если ее аргументы линейно независимые векторы, и равна нулю, если линейно зависимые. Тогда* det *является симметрической, линейной по каждому своему аргументу и* det *равна нулю если среди ее аргументов найдутся два одинаковых.*

Доказательство. Первое и третье свойства очевидны. Симметричность функции следует из того, что линейная независимость системы не зависит от перестановки векторов в системе. Третье свойство следует из линейной зависимости любой системы с двумя одинаковыми векторами. Необходимость функции быть линейной по каждому аргументу менее очевидна.

Покажем, что функция det должна быть линейной по последнему аргументу. Линейность по другим аргументам доказывается аналогично. Пусть $\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n, \boldsymbol{u}'_n \in \mathbb{B}^n$. Рассмотрим три системы векторов $U = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n\}$, $U' = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}'_n\}$ и $U'' = \{\boldsymbol{u}_1, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n \oplus \boldsymbol{u}'_n\}$, отличающиеся только последним вектором. Покажем, что либо среди

этих систем линейно независимыми будут ровно две, либо все три будут линейно зависимыми. Другими словами, если две какие-либо системы одновременно линейно зависимы или линейно независимы, то третья система обязательно будет линейно зависимой.

Не ограничивая общности рассуждений, рассмотрим два случая: (i) системы U и U' линейно зависимы; (ii) системы U и U' линейно независимы. В первом случае найдутся такие наборы $\mathbf{w}, \mathbf{w}' \in \mathbb{B}^n$, что

$$\Big(\bigoplus_{i=1}^{n-1} w_i \boldsymbol{u}_i\Big) \oplus w_n \boldsymbol{u}_n = 0, \quad \Big(\bigoplus_{i=1}^{n-1} w'_i \boldsymbol{u}_i\Big) \oplus w'_n \boldsymbol{u}'_n = 0$$

Если хотя бы одно из чисел w_n и w'_n равно нулю, то очевидно, что в этом случае U'' также линейно зависима. Если $w_n = w'_n = 1$, то тогда

$$\Big(\bigoplus_{i=1}^{n-1} (w \oplus w'_i) \boldsymbol{u}_i\Big) \oplus (\boldsymbol{u}_n \oplus \boldsymbol{u}'_n) = 0.$$

Следовательно, U'' линейно зависима. Во втором случае из линейной независимости систем U и U' следует существование таких наборов $\mathbf{w}, \mathbf{w}' \in \mathbb{B}^{n-1}$, что

$$\Big(\bigoplus_{i=1}^{n-1} w_i \boldsymbol{u}_i\Big) \oplus \boldsymbol{u}_n = \boldsymbol{u}'_n, \quad \Big(\bigoplus_{i=1}^{n-1} w'_i \boldsymbol{u}_i\Big) \oplus \boldsymbol{u}'_n = \boldsymbol{u}_n.$$

Прибавляя первое равенство ко второму, убеждаемся, что система U'' линейно зависима.

Теперь нетрудно показать, что функция det линейна по своему последнему аргументу. Так как среди систем U, U' и U'' линейно независимых четное число, то справедливо равенство

$$\det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) \oplus \det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}'_n) \oplus \det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \oplus \boldsymbol{u}'_n) = 0,$$

которое, как легко видеть, эквивалентно равенству (2.4.1) при $i = n$. Лемма доказана.

В следующей лемме устанавливается, что существует единственная функция, удовлетворяющая сформулированным выше трем необходимым условиям.

Лемма 2.4.2. *Пусть* $\det : (\mathbb{B}^n)^n \to \mathbb{B}^n$ *— симметрическая, линейная по каждому своему аргументу функция, которая равна нулю если среди ее аргументов найдутся два одинаковых. Тогда ее значение на векторах* $\{\boldsymbol{u}_i = (u_{i1}, \ldots, u_{in})\}$ *вычисляется по формуле*

$$\det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) = \bigoplus_{1 \le j_1 \ne \cdots \ne j_n \le n} u_{1j_1} \cdot \ldots \cdot u_{nj_n}. \tag{2.4.2}$$

Доказательство. Так как det — линейная по всем своим аргументам функция, то представляя $\boldsymbol{u}_1$ в виде суммы базисных векторов, имеем

$$\det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) = \det\Big(\bigoplus_{j_1=1}^{n} u_{1j_1} \mathbf{e}_{j_1}, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n\Big) = \bigoplus_{j_1=1}^{n} u_{1j_1} \cdot \det(\mathbf{e}_{j_1}, \boldsymbol{u}_2, \ldots, \boldsymbol{u}_n).$$

Выполняя аналогичные преобразования с векторами $\boldsymbol{u}_2, \ldots, \boldsymbol{u}_n$, получаем

$$\begin{aligned} \det(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) &= \bigoplus_{j_1=1}^{n} u_{1j_1} \cdot \det\Big(\boldsymbol{e}_{j_1}, \bigoplus_{j_2=1}^{n} u_{2j_2} \boldsymbol{e}_{j_2}, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_n\Big) = \\ &= \bigoplus_{j_1=1}^{n} \bigoplus_{j_2=1}^{n} u_{1j_1} u_{2j_2} \cdot \det(\boldsymbol{e}_{j_1}, \boldsymbol{e}_{j_2}, \boldsymbol{u}_3, \ldots, \boldsymbol{u}_n) = \ldots \\ &= \bigoplus_{j_1=1}^{n} \cdots \bigoplus_{j_n=1}^{n} u_{1j_1} \cdot \ldots \cdot u_{nj_n} \cdot \det(\boldsymbol{e}_{j_1}, \ldots, \boldsymbol{e}_{j_n}). \end{aligned} \tag{2.4.3}$$

Так как det — симметрическая функция, принимающая единичное значение на различных базисных векторах, и равная нулю если среди ее аргументов найдутся равные, то из последнего равенства получаем, что

$$\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=\bigoplus_{1\le j_1\ne\dots\ne j_n\le n} u_{1j_1}\cdot\ldots\cdot u_{nj_n}.$$

Лемма доказана.

Далее покажем, что сформулированные выше необходимые условия разделения функцией линейно зависимых и линейно независимых систем векторов являются также достаточными, т.е. единственная функция, удовлетворяющая этим условиям, действительно равна единице на любой линейно независимой системе и нулю на любой линейно зависимой системе.

Лемма 2.4.3. *Пусть* $\boldsymbol{u}_1,\dots,\boldsymbol{u}_n\in\mathbb{B}^n$. *Тогда для функции* det, *вычисляемой по формуле* (2.4.2), *справедливо равенство*

$$\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=\begin{cases}1, & \text{если } \boldsymbol{u}_1,\dots,\boldsymbol{u}_n \text{ линейно независимы;}\\ 0, & \text{если } \boldsymbol{u}_1,\dots,\boldsymbol{u}_n \text{ линейно зависимы.}\end{cases}$$

Доказательство. Полагаем, что $i_1,\dots,i_n$ — целые положительные несовпадающие индексы каждый из которых не превосходит n. Сначала покажем, что $\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=0$, если векторы $\boldsymbol{u}_1,\dots,\boldsymbol{u}_n$ линейно зависимы. Из линейной зависимости рассматриваемых векторов следует, что найдутся такие одновременно не равные нулю постоянные w_i, что $\oplus_{i=0}^n w_i\boldsymbol{u}_i=0$. Допустим, что $w_1=1$. Тогда $\boldsymbol{u}_1=\oplus_{i=2}^n w_i\boldsymbol{u}_i$ и

$$\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=\det\Bigl(\bigoplus_{i=2}^n w_i\boldsymbol{u}_i,\boldsymbol{u}_2,\dots,\boldsymbol{u}_n\Bigr)\bigoplus_{i=2}^n w_i\det(\boldsymbol{u}_i,\boldsymbol{u}_2,\dots,\boldsymbol{u}_n).$$

Каждый определитель, входящий в последнюю сумму, имеет по два одинаковых аргумента. Следовательно, вся сумма равна нулю. Таким образом, мы показали, что если векторы $\boldsymbol{u}_1,\dots,\boldsymbol{u}_n$ линейно зависимы, то значение определителя на этих векторах равно нулю.

Пусть теперь $\boldsymbol{u}_1,\dots,\boldsymbol{u}_n$ — линейно независимые векторы из $\mathbb{B}^n$. Покажем, что $\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=1$. Так как рассматриваемые векторы образуют в $\mathbb{B}^n$ базис, то найдутся такие постоянные $w_{i,j}$, что

$$\boldsymbol{e}_i=\bigoplus_{j=1}^n w_{i,j}\boldsymbol{u}_j.$$

Вычислим $\det(\mathbf{e}_1,\dots,\mathbf{e}_n)$. Выполняя преобразования, аналогичные преобразованиям в (2.4.3), получаем, что

$$\det(\boldsymbol{e}_1,\dots,\boldsymbol{e}_n)=\bigoplus_{1\le j_1\ne\dots\ne j_n\le n} w_{1,j_1}\cdot\ldots\cdot w_{n,j_n}\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n).$$

Если $\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=0$, то немедленно получаем $\det(\boldsymbol{e}_1,\dots,\boldsymbol{e}_n)=0$. Противоречие. Следовательно, $\det(\boldsymbol{u}_1,\dots,\boldsymbol{u}_n)=1$. Лемма доказана.

Задачи

2.4.1. Определителем квадратной матрицы $\mathbf{A}$ называется определитель ее строк. Показать, что $\det\mathbf{A}=1$ тогда и только тогда, когда $\mathbf{A}$ невырождена.

2.4.2. Найти $\det\mathbf{A}$, если:

a) $\mathbf{A}=\begin{pmatrix}0&1&\dots&\dots&1\\1&0&1&\dots&1\\ \dots&\dots&\dots&\dots&\dots\\1&\dots&\dots&1&0\end{pmatrix}$, b) $\mathbf{A}=\begin{pmatrix}1&1&0&\dots&\dots&0\\0&1&1&0&\dots&0\\ \dots&\dots&\dots&\dots&\dots&\dots\\1&0&\dots&\dots&0&1\end{pmatrix}$.

2.4.3. Показать, что $\det \mathbf{AB} = \det \mathbf{A} \cdot \det \mathbf{B}$ для любых квадратных матриц $\mathbf{A}$ и $\mathbf{B}$ одного порядка.

2.4.4. Показать, что $\det \mathbf{A} = \det \mathbf{A}^{\mathrm{T}}$ для любой квадратной матрицы $\mathbf{A}$.

Глава 3.

Линейные булевы пространства и булевы функции

В этой главе рассматриваются примеры использования структуры линейного булева пространства для решения различных задач, возникающих как в теории булевых функций, так и в некоторых смежных областях.

3.1. Системы линейных булевых уравнений

1. Рассмотрим систему из m линейных булевых уравнений

$$\begin{cases} a_{11}x_1 \oplus a_{12}\,x_2 \oplus \ldots \oplus a_{1n}\,x_n = b_1, \\ a_{21}x_1 \oplus a_{22}\,x_2 \oplus \ldots \oplus a_{1n}\,x_n = b_2, \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 \oplus a_{m2}x_2 \oplus \cdots \oplus a_{mn}x_n = b_m \end{cases} \tag{3.1.1}$$

с булевыми коэффициентами a_{ij}, булевыми свободными членами b_i и с n булевыми неизвестными x_i. Эту систему можно записать в виде матричного уравнения

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \tag{3.1.2}$$

с булевой (m,n)-матрицей $\mathbf{A} = (a_{ij})$, составленной из коэффициентов функций системы (3.1.1), вектором-столбцом свободных членов $\boldsymbol{b} = (b_i)$ и вектором-столбцом неизвестных $\boldsymbol{x} = (x_i)$. Матричное уравнение называется *согласованным*, если оно имеет хотя бы одно решение. Далее будем изучать решения уравнения (3.1.2), а, следовательно, и решения системы (3.1.1).

Прежде всего выясним, в каких случаях уравнение (3.1.2) согласовано. Для этого из матрицы $\mathbf{A}$ и вектора $\boldsymbol{b}$ рассматриваемого уравнения составим новую матрицу $(\mathbf{A}\,|\,\boldsymbol{b})$, добавив к матрице $\mathbf{A}$ в качестве нового столбца вектор $\boldsymbol{b}$ и отделив его при этом от $\mathbf{A}$ вертикальной линией. Получившаяся матрица

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} & b_m \end{array}\right).$$

называется *расширенной матрицей* уравнения (3.1.2) (и системы (3.1.1)). Имеет место следующий критерий согласованности матричного уравнения.

Теорема 3.1.1. *Уравнение* $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ *с булевой* (m,n)*-матрицей* $\mathbf{A}$ *имеет решение тогда и только тогда, когда ранг матрицы* $\mathbf{A}$ *равен рангу расширенной матрицы* $(\mathbf{A}|\boldsymbol{b})$.

Доказательство. Если $\operatorname{rank}\mathbf{A} = \operatorname{rank}(\mathbf{A}\,|\,\boldsymbol{b})$, то вектор $\boldsymbol{b}$ является линейной комбинацией столбцов $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ матрицы $\mathbf{A}$, т.е.

$$\boldsymbol{b} = \alpha_1\boldsymbol{a}_1 \oplus \cdots \oplus \alpha_n\boldsymbol{a}_n. \tag{3.1.3}$$

Так как равенство (3.1.3) эквивалентно матричному равенству $\mathbf{A}\boldsymbol{\alpha} = \boldsymbol{b}$, то очевидно, что вектор $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ будет решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$.

С другой стороны, если вектор-столбец $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ является решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$, то, очевидно, справедливо равенство (3.1.3), из которого немедленно следует равенство рангов матриц $\mathbf{A}$ и $(\mathbf{A}\,|\,\boldsymbol{b})$. Теорема доказана.

Теорема 3.1.2. *Если уравнение* $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ *с булевой* (m,n)*-матрицей* $\mathbf{A}$ *имеет хотя бы одно решение, то все решения этого уравнения образуют смежный класс пространства* $\mathbb{B}^n$ *по ортогональному пространству матрицы* $\mathbf{A}$.

Доказательство. Допустим, что уравнение $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ имеет решение $\boldsymbol{x}_0$. В этом случае для доказательства теоремы достаточно показать, что любой вектор, лежащий с вектором $\boldsymbol{x}_0$ в одном смежном классе пространства $\mathbb{B}^n$ по ортогональному пространству матрицы $\mathbf{A}$, является решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$, и наоборот, любое решение рассматриваемого уравнения принадлежит тому же смежному классу пространства $\mathbb{B}^n$ по ортогональному пространству матрицы $\mathbf{A}$, что и вектор $\boldsymbol{x}_0$.

Рассмотрим ортогональное пространство $\mathbb{A}^\perp$ матрицы $\mathbf{A}$. Для каждого вектора $\boldsymbol{v}$ из пространства $\mathbb{A}^\perp$ справедливы равенства

$$\mathbf{A}(\boldsymbol{x}_0 \oplus \boldsymbol{v}) = \mathbf{A}\boldsymbol{x}_0 \oplus \mathbf{A}\boldsymbol{v} = \boldsymbol{b} \oplus \mathbf{0} = \boldsymbol{b}.$$

Следовательно, вектор $\boldsymbol{x}_0 \oplus \boldsymbol{v}$ является решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$. С другой стороны, если $\boldsymbol{y}$ — решение рассматриваемого уравнения, то

$$\mathbf{A}(\boldsymbol{y} \oplus \boldsymbol{x}_0) = \mathbf{A}\boldsymbol{y} \oplus \mathbf{A}\boldsymbol{x}_0 = \boldsymbol{b} \oplus \boldsymbol{b} = \mathbf{0}.$$

Поэтому, $\boldsymbol{y} \oplus \boldsymbol{x}_0 \in \mathbb{A}^\perp$. Следовательно, $\boldsymbol{y}$ принадлежит тому же смежному классу пространства $\mathbb{B}^n$ по $\mathbb{A}^\perp$, что и вектор $\boldsymbol{x}_0$. Теорема доказана.

Из доказанной теоремы следует, что для нахождения всех решений уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ достаточно решить две задачи: (1) найти хотя бы одно решение $\boldsymbol{x}_0$ этого уравнения, такое решение называется *частным*; (2) найти ортогональное пространство матрицы $\mathbf{A}$. Эти две задачи изучаются в следующих пунктах.

2. Матрица $\mathbf{A}$ невырожденного линейного оператора A называется *невырожденной* матрицей. Так как размерность образа линейного невырожденного (n,n)-оператора равна n, а сам образ является линейной оболочкой столбцов матрицы этого оператора, то легко видеть, что столбцы невырожденной матрицы линейно независимы.

Каждая невырожденная булева матрица порядка n имеет единственную *обратную* матрицу $\mathbf{A}^{-1}$ такую, что

$$\mathbf{A}^{-1}\mathbf{A} = \mathbf{A}\mathbf{A}^{-1} = \mathbf{E}_n.$$

Нетрудно показать, что матрица $\mathbf{E}_n$ является единственной матрицей порядка n такой, что для любой матрицы $\mathbf{A}$ того же порядка справедливы равенства $\mathbf{A}\mathbf{E}_n = \mathbf{A}$ и $\mathbf{E}_n\mathbf{A} = \mathbf{A}$. Следовательно, если для квадратных матриц $\mathbf{A}$ и $\mathbf{B}$ одного порядка n выполняется хотя бы одно из равенств $\mathbf{A}\mathbf{B} = \mathbf{A}$ или $\mathbf{B}\mathbf{A} = \mathbf{A}$, то $\mathbf{B} = \mathbf{E}_n$. Отсюда в свою очередь легко следует, что если квадратные матрицы $\mathbf{A}$ и $\mathbf{B}$ таковы, что $\mathbf{A}\mathbf{B} = \mathbf{E}_n$ или $\mathbf{B}\mathbf{A} = \mathbf{E}_n$, то $\mathbf{A}^{-1} = \mathbf{B}$.

Нетрудно показать, что при помощи элементарных преобразований строк любую невырожденную булеву матрицу порядка n можно преобразовать в единичную матрицу того же порядка. Приведем соответствующий алгоритм, состоящий из n шагов. После выполнения i-го шага алгоритма первые i столбцов исходной матрицы будут преобразованы в первые i столбцов единичной матрицы.

Рассмотрим произвольную невырожденную матрицу

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix}. \tag{3.1.4}$$

Так как матрица невырождена, то в ее первом столбце найдется хотя бы один единичный элемент. В матрице $\mathbf{A}$ переставим строки так, чтобы после перестановки первый элемент первой строки стал равным единице. Затем ко всем строкам, в которых первый элемент равен единице, прибавим первую строку. В результате получим новую матрицу

$$\mathbf{A}' = \begin{pmatrix} 1 & a'_{12} & a'_{13} & \cdots & a'_{1n} \\ 0 & a'_{22} & a'_{23} & \cdots & a'_{2n} \\ 0 & a'_{32} & a'_{33} & \cdots & a'_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a'_{n2} & a'_{n3} & \cdots & a'_{nn} \end{pmatrix},$$

в которой в первом столбце единица встречается только один раз — в первой строке. Отметим, что среди элементов $a'_{22}, a'_{32}, \ldots, a'_{n2}$ найдется хотя бы один равный единице. Если это не так, то второй столбец матрицы $\mathbf{A}'$ будет либо совпадать с первым столбцом, либо состоять из одних нулей, и, следовательно, матрица $\mathbf{A}'$ не будет невырожденной. Пусть таким элементом будет a'_{i2}. В матрице $\mathbf{A}'$ поменяем местами вторую и i-ю строки. Затем ко всем строкам, в которых второй элемент равен единице, прибавим вторую строку. В результате получим новую матрицу

$$\mathbf{A}'' = \begin{pmatrix} 1 & 0 & a'_{13} & \cdots & a'_{1n} \\ 0 & 1 & a'_{23} & \cdots & a'_{2n} \\ 0 & 0 & a'_{33} & \cdots & a'_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a'_{n3} & \cdots & a'_{nn} \end{pmatrix},$$

в которой во втором столбце единица встречается только один раз — во второй строке. Вновь отметим, что среди элементов $a'_{33}, a'_{43}, \ldots, a'_{n3}$ найдется хотя бы один равный единице. Если это не так, то третий столбец матрицы $\mathbf{A}''$ будет принадлежать линейной оболочке первого и второго столбцов этой матрицы.

Легко видеть, что, применяя описанную процедуру к оставшимся $n-2$ столбцам, можно преобразовать матрицу $\mathbf{A}$ в единичную матрицу порядка n. Таким образом, каждая невырожденная матрица эквивалентна единичной матрице того же порядка.

Теперь заметим, что применение первого элементарного преобразования к данной булевой матрице сводится к умножению этой матрицы слева на единичную матрицу с переставленными i-й и j-й строками, а применение второго элементарного преобразования — к умножению слева на единичную матрицу с дополнительной единицей, стоящей на пересечении j-й строки и i-го столбца. Например, для перестановки первой и четвертой строк нижнетреугольной матрицы четвертого порядка имеем

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

а для прибавление к первой строке этой матрицы ее четвертой строки

$$\begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

Таким образом, для каждой невырожденной булевой матрицы $\mathbf{A}$ найдется булева матрица $\mathbf{B}$, являющаяся произведением матриц соответствующих выполненным элементарным преобразованиям, и такая, что $\mathbf{BA} = \mathbf{E}$. Следовательно, $\mathbf{A}^{-1} = \mathbf{B}$.

Из сказанного выше следует простой способ обращения невырожденной матрицы $\mathbf{A}$. Сначала элементарными преобразованиями матрица $\mathbf{A}$ преобразуется в единичную матрицу, или, что эквивалентно, матрица $\mathbf{A}$ умножается слева на некоторую матрицу $\mathbf{B}$ такую, что $\mathbf{BA} = \mathbf{E}$. Затем аналогичные преобразования производим над единичной матрицей, т.е. умножаем единичную матрицу на ту же матрицу $\mathbf{B}$ и, следовательно, имеем $\mathbf{BE} = \mathbf{B}$. Очевидно, что матрица, получившаяся из единичной матрицы, будет обратной к матрице $\mathbf{A}$.

Пример 3.1.1. Применим сформулированный выше алгоритм обращения невырожденных матриц к матрице $\left(\begin{smallmatrix}0&0&1\\0&1&1\\1&1&1\end{smallmatrix}\right)$. Элементарные преобразования над обращаемой и единичной матрицами будем выполнять одновременно, разделив эти матрицы вертикальной линией. Легко видеть,что справедливы следующие соотношения:

$$\left(\begin{array}{ccc|ccc}0&0&1&1&0&0\\0&1&1&0&1&0\\1&1&1&0&0&1\end{array}\right) \sim \left(\begin{array}{ccc|ccc}1&1&1&0&0&1\\0&1&1&0&1&0\\0&0&1&1&0&0\end{array}\right) \sim$$

$$\sim \left(\begin{array}{ccc|ccc}1&0&0&0&1&1\\0&1&1&0&1&0\\0&0&1&1&0&0\end{array}\right) \sim \left(\begin{array}{ccc|ccc}1&0&0&0&1&1\\0&1&0&1&1&0\\0&0&1&1&0&0\end{array}\right).$$

Таким образом, $\left(\begin{smallmatrix}0&0&1\\0&1&1\\1&1&1\end{smallmatrix}\right)^{-1} = \left(\begin{smallmatrix}0&1&1\\1&1&0\\1&0&0\end{smallmatrix}\right)$. □

Приведенный алгоритм обращения невырожденных матриц позволяет легко решить уравнение $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ с невырожденной матрицей $\mathbf{A}$. Для решения этого уравнения достаточно обратить матрицу $\mathbf{A}$ и умножить найденную матрицу $\mathbf{A}^{-1}$ на вектор $\boldsymbol{b}$. Действительно, умножая матрицу $\mathbf{A}^{-1}$ на левую и правую части рассматриваемого уравнения имеем

$$\mathbf{A}^{-1}\boldsymbol{b} = \mathbf{A}^{-1}\mathbf{A}\boldsymbol{x} = \mathbf{E}\boldsymbol{x} = \boldsymbol{x}.$$

Следовательно, решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ с невырожденной матрицей $\mathbf{A}$ является вектор $\mathbf{A}^{-1}\boldsymbol{b}$. Рассмотрим простой пример.

Пример 3.1.2. Решим матричное уравнение

$$\begin{pmatrix}0&0&1\\0&1&1\\1&1&1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix} = \begin{pmatrix}1\\0\\1\end{pmatrix}.$$

Матрица из этого уравнения была обращена в предыдущем примере. Используя его результат, имеем

$$\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix} = \begin{pmatrix}0&1&1\\1&1&0\\1&0&0\end{pmatrix}\begin{pmatrix}1\\0\\1\end{pmatrix} = \begin{pmatrix}1\\1\\1\end{pmatrix}.$$

Таким образом, решением рассматриваемого уравнения является вектор-столбец (111). □

Заметим, что для решения уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ с невырожденной матрицей $\mathbf{A}$ необязательно эту матрицу обращать. Достаточно выполнить над вектором $\boldsymbol{b}$ действия, преобразующие матрицу $\mathbf{A}$ в единичную матрицу.

Пример 3.1.3. Решим уравнение $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$, в котором матрица $\mathbf{A}$ такая же как и в предыдущем примере, а вектор $\boldsymbol{b}$ равен (110). Элементарные преобразования над матрицей $\mathbf{A}$ и вектором $\boldsymbol{b}$ будем выполнять одновременно, добавив вектор $\boldsymbol{b}$ к столбцам матрицы $\mathbf{A}$. Выполняя такие же преобразования как и в примере 3.1.1, имеем:

$$\left(\begin{array}{ccc|c}0&0&1&1\\0&1&1&1\\1&1&1&0\end{array}\right) \sim \left(\begin{array}{ccc|c}1&1&1&0\\0&1&1&1\\0&0&1&1\end{array}\right) \sim \left(\begin{array}{ccc|c}1&0&0&1\\0&1&1&1\\0&0&1&1\end{array}\right) \sim \left(\begin{array}{ccc|c}1&0&0&1\\0&1&0&0\\0&0&1&1\end{array}\right).$$

Следовательно, $\boldsymbol{x} = (101)$. $\square$

3. В этом и в следующих разделах будем рассматривать матричные уравнения, в которых ранг матрицы меньше числа ее столбцов, т. е. меньше числа искомых неизвестных.

Будем говорить, что булевы (m,n)-матрицы $\mathbf{A}$ и $\mathbf{A}'$ *комбинаторно-эквивалентны*, (обозначается через $\mathbf{A} \overset{c}{\sim} \mathbf{A}'$), если матрица $\mathbf{A}'$ получена из матрицы $\mathbf{A}$ при помощи элементарных преобразований строк, перестановки столбцов и удаления нулевых строк. Покажем, что каждая булева (m,n)-матрица $\mathbf{A}$ ранга k $(k \le m \le n)$ комбинаторно-эквивалентна некоторой матрице

$$\mathbf{A}' = \begin{pmatrix} 1 & 0 & \dots & 0 & 0 & a'_{1\,k+1} & \cdots & a'_{1n} \\ 0 & 1 & \dots & 0 & 0 & a'_{2\,k+1} & \cdots & a'_{2n} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & 1 & a'_{k\,k+1} & \cdots & a'_{mn} \end{pmatrix}, \tag{3.1.5}$$

состоящей из двух блоков — единичной матрицы порядка m и следующей за ней матрицы размера $m \times (n-m)$. Матрица вида (3.1.5) называется *систематической* матрицей.

Сначала рассмотрим булеву (m,n)-матрицу $\mathbf{A}$ ранга m у которой первые m столбцов линейно независимы. Матрица $\mathbf{A}_m$, составленная из первых m столбцов матрицы $\mathbf{A}$, будет невырожденной, и, следовательно, найдется последовательность элементарных преобразований R, переводящая матрицу $\mathbf{A}_m$ в единичную матрицу порядка m. Легко видеть, что эта же последовательность R преобразует матрицу $\mathbf{A}$ в эквивалентную ей систематическую матрицу вида (3.1.5).

Допустим теперь, что ранг булевой (m,n)-матрицы $\mathbf{A}$ равен k, $k < m$, и ее первые k столбцов линейно независимы. Тогда каждый из оставшихся столбцов является линейной комбинацией первых k столбцов. В этом случае найдется последовательность элементарных преобразований R, переводящая матрицу $\mathbf{A}_k$, составленную из первых k столбцов матрицы $\mathbf{A}$, в матрицу $\mathbf{A}'_k$, состоящую из двух блоков — единичной матрицы порядка k и находящейся под ней нулевой $(m-k,k)$-матрицы. Преобразуем матрицу $\mathbf{A}$ при помощи указанной последовательности преобразований R в эквивалентную ей матрицу $\mathbf{A}'$. Так как в исходной матрице $\mathbf{A}$ каждый из последних $n-k$ столбцов является линейной комбинацией первых k столбцов, то и в преобразованной матрице $\mathbf{A}'$ каждый из ее последних $n-k$ столбцов будет линейной комбинацией ее первых k столбцов. Поэтому, в матрице $\mathbf{A}'$ первые k строк образуют систематическую (k,n)-матрицу, а последние $(m-k)$ строк состоят только из нулей. После удаления нулевых строк матрица $\mathbf{A}'$ станет систематической.

Наконец заметим, что любая (m,n)-матрица $\mathbf{B}$ ранга k может быть получена перестановкой столбцов из подходящей (m,n)-матрицы $\mathbf{A}$ ранга k с первыми k линейно независимыми столбцами. Следовательно, любая булева (m,n)-матрица ранга k комбинаторно-эквивалентна некоторой систематической (k,n)-матрице.

Для преобразования булевой матрицы в комбинаторно-эквивалентную ей систематическую матрицу удобно использовать модификацию приведенного в предыдущем разделе алгоритма преобразования невырожденной матрицы в единичную. Отличие модификации от исходного алгоритма состоит только в том, что при выполнении очередного шага алгоритма над произвольной матрицей может возникнуть ситуация, когда преобразуемый на этом шаге столбец будет равен линейной комбинации предыдущих столбцов. В этом случае преобразуемый столбец надо просто пропустить и продолжить выполнение алгоритма со следующим столбцом. Заканчивается выполнение алгоритма удалением нулевых строк (если такие строки появятся) и перестановкой пропущенных столбцов в конец матрицы.

Рассмотрим алгоритм преобразования булевой матрицы в комбинаторно-эквивалентную ей систематическую матрицу на следующем простом примере.

Пример 3.1.4. Преобразуем указанную далее матрицу $\mathbf{A}$ в комбинаторно-эквивалентную ей систематическую матрицу $\mathbf{A}'$. Применяя для этого описанный выше алгоритм, последовательно получим пять приведенных ниже матриц. Первые четыре матрицы эквивалентны матрице $\mathbf{A}$, а последняя, являющаяся систематической, комбинаторно-эквива-

лентна:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \sim$$
$$\sim \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \overset{\mathrm{c}}{\sim} \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

На первом шаге алгоритма первая строка матрицы $\mathbf{A}$ прибавлена к ее второй и третьей строкам. На втором шаге вторая строка новой матрицы прибавлена к ее остальным строкам. На третьем шаге алгоритма переставлены местами третья и четвертая строки матрицы. Легко видеть, что в полученной матрице третий столбец равен сумме первых двух столбцов. Поэтому на четвертом шаге третий столбец оставлен без изменений, преобразован четвертый столбец — третья строка матрицы прибавлена к первой. Наконец на последнем шаге переставлены третий и четвертый столбцы, а последняя строка, состоящая только из нулей, удалена. Полученная в результате преобразований матрица является систематической. □

4. Приведем алгоритм, который находит базис ортогонального пространства произвольной булевой (m, n)-матрицы.

Сначала рассмотрим частный случай — опишем алгоритм решающий рассматриваемую задачу для систематической матрицы. Для этого систематической матрице $\mathbf{A} = \left(\mathbf{E}_m \widetilde{\mathbf{A}}\right)$ поставим в соответствие $(n-m, n)$-матрицу $\mathbf{A}' = \left(\widetilde{\mathbf{A}}^{\mathrm{T}} \mathbf{E}_{n-m}\right)$, которая состоит из двух блоков — транспонированной матрицы $\widetilde{\mathbf{A}}$ и следующей за ней единичной матрицы порядка $(n-m)$. Легко видеть, что для произведения $\mathbf{A}\mathbf{A}'^{\mathrm{T}}$ справедливы равенства:

$$\left(\mathbf{E}_m \widetilde{\mathbf{A}}\right)\left(\widetilde{\mathbf{A}}^{\mathrm{T}} \mathbf{E}_{n-m}\right)^{\mathrm{T}} = \left(\mathbf{E}_m \widetilde{\mathbf{A}}\right)\begin{pmatrix} \widetilde{\mathbf{A}} \\ \mathbf{E}_{n-m} \end{pmatrix} = \left(\widetilde{\mathbf{A}} \oplus \widetilde{\mathbf{A}}\right) = \mathbf{0}.$$

Таким образом, пространство $\mathbb{A}'$, порожденное строками матрицы $\mathbf{A}'$, входит в ортогональное пространство матрицы $\mathbf{A}$. Так как размерность этого пространства равна $n-m$, а размерность пространства строк матрицы $\mathbf{A}$ равна m, т.е. $\dim \mathbb{A}' + \dim \mathbb{A} = n$, то из теоремы 2.3.1 легко следует, что $\mathbb{A}^{\perp} = \mathbb{A}'$. Таким образом, в качестве базиса ортогонального пространства матрицы $\mathbf{A}$ можно взять строки матрицы $\mathbf{A}'$.

Приведенный алгоритм очевидным образом модифицируется в алгоритм нахождения базиса ортогонального пространства произвольной матрицы. Матрицу $\mathbf{A}$ надо преобразовать в комбинаторно-эквивалентную ей систематическую матрицу $\mathbf{A}'$. При этом следует запомнить все выполненные в процессе преобразования перестановки столбцов. Затем для матрицы $\mathbf{A}'$ описанным выше способом строится матрица $\mathbf{B}'$, строки которой порождают ортогональное пространство матрицы $\mathbf{A}'$. Наконец в матрице $\mathbf{B}'$ все переставленные ранее столбцы возвращаются на свои места. Полученная матрица $\mathbf{B}$ будет порождать ортогональное пространство исходной матрицы $\mathbf{A}$.

Пример 3.1.5. Для матрицы $\mathbf{A}$ из рассмотренного выше примера 3.1.4 построим матрицу, порождающую ее ортогональное пространство. Сначала преобразуем матрицу $\mathbf{A}$ в комбинаторно-эквивалентную ей систематическую матрицу $\mathbf{A}'$. Из примера 3.1.4 имеем:

$$\mathbf{A} = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}, \qquad \mathbf{A}' = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}.$$

При преобразовании $\mathbf{A}$ в $\mathbf{A}'$ переставлялись третий и четвертый столбцы. Для матрицы $\mathbf{A}'$ легко находим порождающую ее ортогональное пространство матрицу $\mathbf{B}'$: транспонируя

матрицу $\left(\begin{smallmatrix}1&1&1\\1&1&1\\0&1&0\end{smallmatrix}\right)$ получаем матрицу $\left(\begin{smallmatrix}1&1&0\\1&1&1\\1&1&0\end{smallmatrix}\right)$, к которой справа приписываем единичную матрицу третьего порядка. Затем в матрице $\mathbf{B}'$ выполняем обратную к ранее выполненной перестановку столбцов — снова меняем местами третий и четвертый столбцы. В результате имеем:

$$\mathbf{B}' = \begin{pmatrix}1&1&0&1&0&0\\1&1&1&0&1&0\\1&1&0&0&0&1\end{pmatrix}, \qquad \mathbf{B} = \begin{pmatrix}1&1&1&0&0&0\\1&1&0&1&1&0\\1&1&0&0&0&1\end{pmatrix}.$$

Непосредственной проверкой легко убедиться в том, что $\mathbf{AB} = \mathbf{0}$. □

5. Теперь приведем алгоритм нахождения частного решения уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$. Частное решение уравнения с систематической матрицей $\mathbf{A} = (\mathbf{E}_m\mathbf{B})$ находится легко. Например, из равенств

$$(\mathbf{E}_m\mathbf{B})\begin{pmatrix}\boldsymbol{b}\\\mathbf{0}\end{pmatrix} = \mathbf{E}_n\boldsymbol{b} \oplus \mathbf{B0} = \boldsymbol{b}$$

следует, что вектор-столбец $(\boldsymbol{b}, \mathbf{0})$ будет решением рассматриваемого уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$.

Далее рассмотрим уравнение с несистематической (m, n)-матрицей $\mathbf{A}$ ранга m, в которой первые m столбцов линейно независимы. Легко видеть, что при помощи только элементарных преобразований строк матрица $\mathbf{A}$ может быть преобразована в эквивалентную ей систематическую матрицу $\mathbf{B}$. Ранее было показано (стр. 49), что выполнение любого элементарного преобразования строк квадратной матрицы сводится к умножению этой матрицы слева на некоторую матрицу определенного вида. Аналогичное утверждение справедливо и для матриц произвольных размеров. Например, легко видеть, что умножение произвольной (m, n)-матрицы $\mathbf{A}$ слева на квадратную матрицу порядка m, получающуюся из единичной матрицы того же порядка добавлением единицы на пересечение i-й строки и j-го столбца, соответствует добавлению к i-й строке матрицы $\mathbf{A}$ ее j-й строки. Следовательно, для любой (m, n)-матрицы $\mathbf{A}$ ранга m, в которой первые m столбцов линейно независимы, найдется квадратная матрица $\mathbf{C}$ порядка m умножение на которую слева преобразует матрицу $\mathbf{A}$ в эквивалентную ей систематическую матрицу $\mathbf{B}$. Так как

$$\mathbf{B}\boldsymbol{x} = \mathbf{CA}\boldsymbol{x} = \mathbf{C}\boldsymbol{b},$$

то одновременное выполнение над строками матрицы $\mathbf{A}$ и над координатами вектора $\boldsymbol{b}$ элементарных преобразований, переводящих $\mathbf{A}$ в $\mathbf{B}$, преобразует уравнение $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ в уравнение $\mathbf{B}\boldsymbol{x} = \mathbf{C}\boldsymbol{b}$, решения которого совпадают с решениями исходного уравнения. Так как матрица $\mathbf{B}$ систематическая, то вектор-столбец $(\boldsymbol{b}', \mathbf{0})$, где $\boldsymbol{b}' = \mathbf{C}\boldsymbol{b}$, будет частным решением уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$. Очевидным образом приведенный алгоритм можно использовать и для решений уравнений с произвольными матрицами.

Пример 3.1.6. Найдем все решения уравнения

$$\begin{pmatrix}1&1&0&1&1&0\\1&0&1&1&0&1\\0&1&1&0&1&1\\1&0&1&0&1&1\end{pmatrix}\begin{pmatrix}x_1\\x_2\\\vdots\\x_6\end{pmatrix} = \begin{pmatrix}1\\1\\0\\0\end{pmatrix}. \tag{3.1.6}$$

Решения однородного уравнения с такой же матрицей как и в (3.1.6) были найдены в предыдущем примере. Поэтому для определения всех решений уравнения (3.1.6) осталось найти какое-нибудь его частное решение. Сделаем это описанным выше способом. Элементарные преобразования над строками матрицы и координатами вектора будем выполнять одновременно, добавив в матрицу коэффициентов в качестве дополнительного столбца вектор свободных членов. Легко видеть, что следующие преобразования аналогичны преобразо-

ваниям из предыдущего примера:

$$\left(\begin{array}{cccccc|c} 1&1&0&1&1&0&1\\ 1&0&1&1&0&1&1\\ 0&1&1&0&1&1&0\\ 1&0&1&0&1&1&0 \end{array}\right) \sim \left(\begin{array}{cccccc|c} 1&1&0&1&1&0&1\\ 0&1&1&0&1&1&0\\ 0&1&1&0&1&1&0\\ 0&1&1&1&0&1&1 \end{array}\right) \sim \left(\begin{array}{cccccc|c} 1&0&1&1&0&1&1\\ 0&1&1&0&1&1&0\\ 0&0&0&0&0&0&0\\ 0&0&0&1&1&0&1 \end{array}\right) \sim$$

$$\sim \left(\begin{array}{cccccc|c} 1&0&1&1&0&1&1\\ 0&1&1&0&1&1&0\\ 0&0&0&1&1&0&1\\ 0&0&0&0&0&0&0 \end{array}\right) \sim \left(\begin{array}{cccccc|c} 1&0&1&0&1&1&0\\ 0&1&1&0&1&1&0\\ 0&0&0&1&1&0&1\\ 0&0&0&0&0&0&0 \end{array}\right).$$

Таким образом, получаем уравнение

$$\begin{pmatrix} 1&0&1&0&1&1\\ 0&1&1&0&1&1\\ 0&0&0&1&1&0\\ 0&0&0&0&0&0 \end{pmatrix} \begin{pmatrix} x_1\\ x_2\\ \vdots\\ x_6 \end{pmatrix} = \begin{pmatrix} 0\\ 0\\ 1\\ 0 \end{pmatrix},$$

частное решение которого находится легко. В матрице этого уравнения четвертый столбец равен вектору, стоящему в его правой части. Поэтому легко видеть, что вектор (000100) будет частным решением последнего уравнения, а, следовательно, и уравнения (3.1.6). Наконец каждое решение этих уравнений является суммой найденного частного решения и некоторого вектора из ортогонального пространства матрицы из (3.1.6), т. е. представляется в виде суммы

$$(000100) \oplus \lambda_1(111000) \oplus \lambda_2(110110) \oplus \lambda_3(110001),$$

где λ_1, λ_2 и λ_3 — произвольные булевы константы. □

Задачи

3.1.1. Пусть $\mathbf{A}$, $\mathbf{B}$ — булевы матрицы порядка n такие, что $\mathbf{AB} = \mathbf{E}_n$. Показать, что $\mathbf{BA} = \mathbf{E}_n$.

3.1.2. Доказать, что каждая невырожденная квадратная булева матрица имеет единственную обратную матрицу.

3.1.3. Показать, что квадратная матрица невырождена тогда и только тогда, когда ее строки линейно независимы.

3.1.4. Найти число различных невырожденных булевых матриц порядка n.

3.1.5. Пусть $m \le n$. Найти число различных булевых (m, n)-матриц ранга m.

3.1.6. Найти ранг $(2^n, 2^n)$-матрицы $\mathbf{A}_n$, если $\mathbf{A}_0 = (1)$, а при бóльших n имеет место рекуррентная формула:

a) $\mathbf{A}_{n+1} = \begin{pmatrix} \mathbf{A}_n & \mathbf{A}_n \\ \mathbf{0} & \mathbf{A}_n \end{pmatrix}$; b) $\mathbf{A}_{n+1} = \begin{pmatrix} \mathbf{A}_n & \mathbf{A}_n \\ \overline{\mathbf{A}}_n & \mathbf{A}_n \end{pmatrix}$; c) $\mathbf{A}_{n+1} = \begin{pmatrix} \mathbf{A}_n & \overline{\mathbf{A}}_n \\ \overline{\mathbf{A}}_n & \mathbf{A}_n \end{pmatrix}$.

Здесь через $\mathbf{0}$ обозначена матрица, все элементы которой равны нулю, а через $\overline{\mathbf{A}}$ — матрица получающаяся из $\mathbf{A}$ отрицанием всех ее элементов.

3.1.7. Найти все решения уравнения $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ если:

a) $\mathbf{A} = \begin{pmatrix} 1&1&0&0\\ 0&1&1&0\\ 0&0&1&1\\ 1&0&0&1 \end{pmatrix}$, $\boldsymbol{b} = \begin{pmatrix} 1\\ 1\\ 0\\ 1 \end{pmatrix}$; b) $\mathbf{A} = \begin{pmatrix} 1&1&0&1&1\\ 1&0&1&0&1\\ 0&1&1&1&1\\ 1&1&1&1&1 \end{pmatrix}$, $\boldsymbol{b} = \begin{pmatrix} 0\\ 0\\ 0\\ 1 \end{pmatrix}$

c) $\mathbf{A} = \begin{pmatrix} 0&1&1&1\\ 1&0&1&1\\ 1&1&0&1\\ 1&1&1&0 \end{pmatrix}$, $\boldsymbol{b} = \begin{pmatrix} 1\\ 0\\ 0\\ 1 \end{pmatrix}$; d) $\mathbf{A} = \begin{pmatrix} 1&1&0&0&0\\ 0&1&1&0&0\\ 0&0&1&1&0\\ 0&0&0&1&1 \end{pmatrix}$, $\boldsymbol{b} = \begin{pmatrix} 1\\ 1\\ 1\\ 1 \end{pmatrix}$.

3.2. Вычисление коэффициентов АНФ

1. Пусть $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ и $\boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_n$ два разных базиса в $\mathbb{B}^n$. Тогда каждый вектор $\boldsymbol{u} \in \mathbb{B}^n$ является линейной комбинацией векторов этих базисов, т. е.

$$\boldsymbol{u} = u_1\boldsymbol{v}_1 \oplus u_2\boldsymbol{v}_2 \oplus \cdots \oplus u_n\boldsymbol{v}_n = u'_1\boldsymbol{v}'_1 \oplus u'_2\boldsymbol{v}'_2 \oplus \cdots \oplus u'_n\boldsymbol{v}'_n. \tag{3.2.1}$$

Рассмотрим задачу определения координат вектора $\boldsymbol{u} \in \mathbb{B}^n$ в базисе $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$, если известны его координаты в базисе $\boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_n$. Допустим, что векторы $\boldsymbol{v}'_i$ второго базиса выражаются через векторы первого базиса следующими формулами:

$$\begin{aligned}
\boldsymbol{v}'_1 &= \alpha_{11}\boldsymbol{v}_1 \oplus \alpha_{12}\boldsymbol{v}_2 \oplus \cdots\cdots \oplus \alpha_{1n}\boldsymbol{v}_n,\\
\boldsymbol{v}'_2 &= \alpha_{21}\boldsymbol{v}_1 \oplus \alpha_{22}\boldsymbol{v}_2 \oplus \cdots\cdots \oplus \alpha_{2n}\boldsymbol{v}_n,\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
\boldsymbol{v}'_n &= \alpha_{n1}\boldsymbol{v}_1 \oplus \alpha_{n2}\boldsymbol{v}_2 \oplus \cdots\cdots \oplus \alpha_{nn}\boldsymbol{v}_n.
\end{aligned} \tag{3.2.2}$$

Подставим эти формулы в правую часть равенства (3.2.1). Тогда

$$\boldsymbol{u} = \bigoplus_{i=1}^{n} u'_i\boldsymbol{v}'_i = \bigoplus_{i=1}^{n} u'_i\left(\bigoplus_{j=1}^{n} \alpha_{ij}\boldsymbol{v}_j\right) = \bigoplus_{j=1}^{n}\bigoplus_{i=1}^{n} \alpha_{ij}u'_i\boldsymbol{v}_j = \bigoplus_{j=1}^{n}\left(\bigoplus_{i=1}^{n} \alpha_{ij}u'_i\right)\boldsymbol{v}_j.$$

Следовательно,

$$\begin{aligned}
u_1 &= \alpha_{11}u'_1 \oplus \alpha_{21}u'_2 \oplus \cdots\cdots \oplus \alpha_{n1}u'_n,\\
u_2 &= \alpha_{12}u'_1 \oplus \alpha_{22}u'_2 \oplus \cdots\cdots \oplus \alpha_{n2}u'_n,\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
u_n &= \alpha_{1n}u'_1 \oplus \alpha_{2n}u'_2 \oplus \cdots\cdots \oplus \alpha_{nn}u'_n.
\end{aligned}$$

Последние равенства можно переписать в матричной форме

$$\begin{pmatrix} \alpha_{11} & \alpha_{21} & \cdots & \alpha_{n1} \\ \alpha_{12} & \alpha_{22} & \cdots & \alpha_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{1n} & \alpha_{2n} & \cdots & \alpha_{nn} \end{pmatrix} \begin{pmatrix} u'_1 \\ u'_2 \\ \vdots \\ u'_n \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}. \tag{3.2.3}$$

Матрица в (3.2.3) называется матрицей перехода от базиса $\boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_n$ к базису $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$. Ее i-й столбец состоит из координат вектора $\boldsymbol{v}'_i$ в базисе $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$. Таким образом, для определения координат вектора $\boldsymbol{u}$ в базисе $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ по его известным координаты в базисе $\boldsymbol{v}'_1, \ldots, \boldsymbol{v}'_n$ надо определить координаты каждого вектора $\boldsymbol{v}'_i$ в первом базисе, из найденных координат составить матрицу перехода, и умножить ее на вектор $\boldsymbol{u}$.

2. Воспользуемся приведенным выше методом определения координат вектора в новом базисе для вычисления коэффициентов многочлена Жегалкина произвольной булевой функции.

Рассмотрим линейное 2^n-мерное пространство $\mathbb{B}^{2^n}$, состоящее из векторов значений всех булевых функций, зависящих от n переменных. Далее для краткости будем говорить не о векторах значений булевых функций, а просто о функциях. В пространстве $\mathbb{B}^{2^n}$ введем два естественных базиса K_n и P_n. Базис $K_n = \{\boldsymbol{k}_0, \ldots, \boldsymbol{k}_{2^n-1}\}$ состоит из всех 2^n элементарных конъюнкций $\boldsymbol{k}_{|\boldsymbol{v}|} = x_1^{(v_1)} \cdot \ldots \cdot x_n^{(v_n)}$ переменных $x_1, \ldots, x_n$, а базис $P_n = \{\boldsymbol{p}_0, \ldots, \boldsymbol{p}_{2^n-1}\}$ — из всех 2^n булевых одночленов $\boldsymbol{p}_{|\boldsymbol{v}|} = x_1^{v_1} \cdot \ldots \cdot x_n^{v_n}$ тех же переменных.

В качестве примера рассмотрим линейные пространства $\mathbb{B}^2$ и $\mathbb{B}^4$. Первое пространство состоит из четырех булевых функций, зависящих от одной переменной x, второе — из 16 булевых функций, зависящих от двух переменных x_1 и x_2. В пространстве $\mathbb{B}^2$ базис K_1 состоит из переменной x и ее отрицания:

$$\boldsymbol{k}_0 = x^{(0)} = \overline{x} = (1,0), \qquad \boldsymbol{k}_1 = x^{(1)} = x = (0,1);$$

а базис P_1 — из тождественной единицы и переменной x:

$$\boldsymbol{p}_0 = x^0 = 1 = (1,1), \qquad \boldsymbol{p}_2 = x^1 = x = (0,1).$$

В пространстве $\mathbb{B}^4$ базисы K_2 и P_2 состоят из следующих функций:

$$\begin{aligned}
&\boldsymbol{k}_0 = \overline{x}_1\overline{x}_2 = (1,0,0,0); && \boldsymbol{p}_0 = 1\cdot 1 \ = (1,1,1,1);\\
&\boldsymbol{k}_1 = \overline{x}_1 x_2 = (0,1,0,0); && \boldsymbol{p}_1 = 1\cdot x_2 = (0,1,0,1);\\
&\boldsymbol{k}_2 = x_1\overline{x}_2 = (0,0,1,0); && \boldsymbol{p}_2 = x_1\cdot 1 = (0,0,1,1);\\
&\boldsymbol{k}_3 = x_1 x_2 = (0,0,0,1); && \boldsymbol{p}_3 = x_1 x_2 \ = (0,0,0,1).
\end{aligned}$$

Обозначим через $\mathbf{P}_m$ матрицу перехода от базиса P_m к базису K_m. Нетрудно видеть, что для матриц $\mathbf{P}_1$ и $\mathbf{P}_2$ справедливы равенства:

$$\mathbf{P}_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \qquad \mathbf{P}_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

Сравнение этих матриц показывает, что $\mathbf{P}_2$ состоит из четырех блоков — трех матриц $\mathbf{P}_1$ и одной нулевой матрицы, состоящей из одних нулей. Покажем, что аналогичным образом устроены все матрицы $\mathbf{P}_n$.

Для этого составим таблицу $\mathbf{T}_n$, состоящую из 2^n строк и 2^n столбцов, занумерованных целыми числами от нуля до 2^n-1. Пусть $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{B}^n$. В таблице $\mathbf{T}_n$ $|\boldsymbol{u}|$-й строке поставим в соответствие набор $\boldsymbol{u}$, а $|\boldsymbol{v}|$-му столбцу — одночлен $p_{\boldsymbol{v}} = x_1^{v_1}\cdots x_n^{v_n}$. На пересечении $|\boldsymbol{u}|$-й строки, соответствующей набору $(u_1,\ldots,u_n)$, и $|\boldsymbol{v}|$-го столбца, соответствующего одночлену $x_1^{v_1}\cdots x_n^{v_n}$, поместим значение одночлена $x_1^{v_1}\cdots x_n^{v_n}$ на наборе $u_1,\ldots,u_n$. Например таблицы $\mathbf{T}_1$ и $\mathbf{T}_2$ выглядят следующим образом:

x_1	1	x_1
0	1	0
1	1	1

x_1	x_2	1	x_2	x_1	x_1x_2
0	0	1	0	0	0
0	1	1	1	0	0
1	0	1	0	1	0
1	1	1	1	1	1

Так как $\boldsymbol{p}_{(0,v_2,\ldots,v_n)}(x_1,x_2,\ldots,x_n) = \boldsymbol{p}_{(v_2,\ldots,v_n)}(x_2,\ldots x_n)$, и при любых $\boldsymbol{u}$ и $\boldsymbol{v}$ из $\mathbb{B}^n$ справедливо равенство

$$u_1^{v_1}u_2^{v_2}\cdots u_n^{v_n} = \begin{cases} u_2^{v_2}\cdots u_n^{v_n}, & \text{если } v_1 \le u_1 \\ 0, & \text{если } v_1 > u_1, \end{cases}$$

то в общем случае (см. таблицу 3.2.1, где $\boldsymbol{u} = (u_2,\ldots,u_n)$, $\boldsymbol{v} = (v_2,\ldots,v_n)$, и, следовательно, одночлен $\boldsymbol{p}_{\boldsymbol{v}}$ не зависит от x_1) таблица $\mathbf{T}_n$ естественным образом распадается на четыре блока размера $2^{n-1}\times 2^{n-1}$, три из которых совпадают с таблицей $\mathbf{T}_{n-1}$, а четвертый, определяемый равенствами $u_1 = 0$, $v_1 = 1$, состоит из одних нулей.

Из равенства (1.4.8) следует, что для любой булевой функции $f \in P_2(n)$ ее вектор значений является ее вектором координат в базисе элементарных конъюнкций. Поэтому построенная выше таблица $\mathbf{T}_n$, в которой j-й столбец состоит из координат вектора $\boldsymbol{p}_j$ в базисе K_n, фактически является матрицей перехода от базиса P_n к базису K_n. Следовательно, для каждой матрицы $\mathbf{P}_n$, при $n \ge 2$, справедливо рекуррентное представление

$$\mathbf{P}_n = \begin{pmatrix} \mathbf{P}_{n-1} & \mathbf{0} \\ \mathbf{P}_{n-1} & \mathbf{P}_{n-1} \end{pmatrix}, \tag{3.2.4}$$

позволяющее достаточно просто находить матрицы перехода от базиса одночленов к базису элементарных конъюнкций. Для определения матрицы перехода от базиса элементарных конъюнкций к базису одночленов надо обратить матрицу $\mathbf{P}_n$.

Таблица 3.2.1.

$x_1\ x_2\ \ldots\ x_n$	$\cdots\ \boldsymbol{p_v}\ \cdots$	$\cdots\ x_1\boldsymbol{p_v}\ \cdots$
$0\ 0\ \cdots\ 0$		
$\ldots\ldots\ldots\ldots$		
$0\ u_2\ \cdots\ u_n$	$\boldsymbol{p_v}(\boldsymbol{u})$	0
$\ldots\ldots\ldots\ldots$		
$0\ 1\ \cdots\ 1$		
$1\ 0\ \cdots\ 0$		
$\ldots\ldots\ldots\ldots$		
$1\ u_2\ \cdots\ u_n$	$\boldsymbol{p_v}(\boldsymbol{u})$	$\boldsymbol{p_v}(\boldsymbol{u})$
$\ldots\ldots\ldots\ldots$		
$1\ 1\ \cdots\ 1$		

Лемма 3.2.1. *Для каждой матрицы* $\mathbf{P}_n$, $n \geq 1$, *справедливо равенство*

$$\mathbf{P}_n^{-1} = \mathbf{P}_n. \tag{3.2.5}$$

Доказательство. Лемму докажем индукцией по n. При $n = 1$ утверждение леммы легко следует из очевидного равенства

$$\mathbf{P}_1\mathbf{P}_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Предположим, что утверждение леммы справедливо для всех натуральных n не превосходящих некоторое m. Пусть $\mathbf{E}_k$ — единичная матрица размера $k \times k$. По предположению индукции $\mathbf{P}_m^2 = \mathbf{E}_{2^m}$. Вычислим $\mathbf{P}_{m+1}^2$ используя предположение индукции и равенство (3.2.4):

$$\begin{aligned}\mathbf{P}_{m+1}^2 &= \begin{pmatrix} \mathbf{P}_m & \mathbf{0} \\ \mathbf{P}_m & \mathbf{P}_m \end{pmatrix}\begin{pmatrix} \mathbf{P}_m & \mathbf{0} \\ \mathbf{P}_m & \mathbf{P}_m \end{pmatrix} = \\ &= \begin{pmatrix} \mathbf{P}_m^2 & \mathbf{0} \\ (\mathbf{P}_m^2 \oplus \mathbf{P}_m^2) & \mathbf{P}_m^2 \end{pmatrix} = \begin{pmatrix} \mathbf{E}_{2^m} & \mathbf{0} \\ \mathbf{0} & \mathbf{E}_{2^m} \end{pmatrix} = \mathbf{E}_{2^{m+1}}.\end{aligned}$$

Лемма доказана.

Таким образом, для вычисления коэффициентов многочлена Жегалкина функции f из $P_2(n)$ надо умножить матрицу $\mathbf{P}_n$ на вектор значений функции f. Если в векторе значений $(f_0, f_1, \ldots, f_{2^n-1})$ координата $f_{|\boldsymbol{u}|}$ равна $f(u_1, \ldots, u_n)$, то в векторе $(f_0^\star, f_1^\star, \ldots, f_{2^n-1}^\star)$, состоящем из коэффициентов многочлена Жегалкина, его $|\boldsymbol{v}|$-я координата будет равна коэффициенту при одночлене $x_1^{v_1}\cdots x_n^{v_n}$.

Пример 3.2.1. Применим полученную формулу для вычисления коэффициентов многочлена Жегалкина дизъюнкции двух переменных. Легко видеть, что

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

Следовательно, $x_1 \vee x_2 = x_2 \oplus x_1 \oplus x_1x_2$. Ранее, методом неопределенных коэффициентов такую же формулу получили в примере 1.4.7. □

Задачи

3.2.1. Найти все булевы функции двух и трех переменных, для которых вектор значений совпадает с вектором коэффициентов многочлена Жегалкина.

3.2.2. Для скольких булевых функции четырех переменных вектор значений совпадает с вектором коэффициентов многочлена Жегалкина.

3.2.3. Для скольких n-местных булевых функции вектор значений совпадает с вектором коэффициентов многочлена Жегалкина.

3.3. Линейное хеширование

1. *Хешированием* множества D, лежащего в n-мерном булевом кубе, называется отображение этого множества в булев куб меньшей размерности. Оператор, задающий такое отображение, называется оператором хеширования множества D. Если при хешировании образы различных элементов множества D различны, то хеширование называется *совершенным*. Соответственно, оператор f называется *оператором совершенного хеширования* множества D, если $f(\boldsymbol{x}) \neq f(\boldsymbol{y})$ для любых $\boldsymbol{x}, \boldsymbol{y} \in D$. Оператор совершенного хеширования области D будем так же называть инъективным оператором на области D и будем говорить, что этот оператор действует на области D инъективно. Хеширование при помощи линейных операторов называется *линейным*. Ниже рассматривается задача совершенного линейного хеширования произвольного подмножества в $\mathbb{B}^n$.

Для любой области $D \subseteq \mathbb{B}^n$ через $D^\star$ обозначим множество всех попарных сумм различных элементов этой области, т.е.

$$D^\star = \{\boldsymbol{y} \mid \boldsymbol{y} = \boldsymbol{x}_i \oplus \boldsymbol{x}_j, \text{ где } \boldsymbol{x}_i, \boldsymbol{x}_j \in D \text{ и } \boldsymbol{x}_i \neq \boldsymbol{x}_j\}.$$

Справедливо следующее утверждение о существовании линейного оператора совершенного хеширования.

Теорема 3.3.1. *Пусть для множества $D^\star$ попарных сумм элементов области $D \subseteq \mathbb{B}^n$ справедливо неравенство*

$$2^{n-m+1} > |D^\star| + 1.$$

Тогда существует инъективный на области D линейный $(n-m, n)$-оператор.

Доказательство теоремы 3.3.1 основано на последовательном применении ее частного случая — доказываемой ниже леммы.

Лемма 3.3.1. *Пусть для множества $D^\star$ попарных сумм элементов области $D \subseteq \mathbb{B}^n$ справедливо неравенство*

$$2^n > |D^\star| + 1.$$

Тогда существует инъективный на области D линейный $(n-1, n)$-оператор.

Доказательство. Если линейный оператор f инъективно действует на области D, т.е. $f(\boldsymbol{x}_i) \neq f(\boldsymbol{x}_j)$ для любых $\boldsymbol{x}_i$ и $\boldsymbol{x}_j$ из D, то

$$f(\boldsymbol{x}_i \oplus \boldsymbol{x}_j) = f(\boldsymbol{x}_i) \oplus f(\boldsymbol{x}_j) \neq \boldsymbol{0}. \tag{3.3.1}$$

Следовательно, $\boldsymbol{x}_i \oplus \boldsymbol{x}_j \notin \ker f$. Поэтому из (3.3.1) следует, что множество $D^\star$ и ядро оператора f не пересекаются. Легко видеть, что верно и обратное: если множество $D^\star$ и подпространство $\mathbb{H}$ не пересекаются, то $\mathbb{H}$ является ядром линейного оператора отображающего несовпадающие наборы области D в несовпадающие наборы ее образа. Действительно, рассмотрим подпространство $\mathbb{H}$, не имеющее общих наборов с $D^\star$, и линейный оператор f, ядром которого является $\mathbb{H}$. Пусть $\boldsymbol{x}_i$ и $\boldsymbol{x}_j$ — произвольные наборы из D. Так как $\boldsymbol{x}_i \oplus \boldsymbol{x}_j \notin \mathbb{H} = \ker f$, то

$$f(\boldsymbol{x}_i) \oplus f(\boldsymbol{x}_j) = f(\boldsymbol{x}_i \oplus \boldsymbol{x}_j) \neq \boldsymbol{0},$$

т.е. образы наборов $\boldsymbol{x}_i$ и $\boldsymbol{x}_j$ различны. Поэтому для построения требуемого линейного оператора достаточно найти в $\mathbb{B}^n$ подпространство $\mathbb{H}$, которое не пересекается с множеством $D^\star$ и размерность которого равна единице. Существование такого пространства легко следует из условий леммы. Так как $2^n > |D^\star| + 1$, то среди элементов $\mathbb{B}^n$ найдется ненулевой набор не принадлежащий $D^\star$. Этот набор вместе с нулевым набором будут составлять требуемое одномерное подпространство. Лемма доказана.

Доказательство теоремы 3.3.1. Воспользуемся леммой 3.3.1. Из этой леммы следует существование такого линейного $(n-1,n)$-оператора f_1, что $f_1(\boldsymbol{x}) \neq f_1(\boldsymbol{y})$ для любых неравных наборов $\boldsymbol{x}$ и $\boldsymbol{y}$ из D. Далее для множества D будем использовать обозначение D_0. Через D_1 обозначим образ области D_0 при действии f_1. Легко видеть, что мощность множества $D_1^\star$, состоящего из попарных сумм различных элементов множества D_1, не превосходит мощности множества $D_0^\star$. Действительно, если это не так, то в D_0 должны присутствовать такие наборы $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ и $\boldsymbol{y}_1$, $\boldsymbol{y}_2$, что $\boldsymbol{x}_1 \oplus \boldsymbol{x}_2 = \boldsymbol{y}_1 \oplus \boldsymbol{y}_2$ и $f_1(\boldsymbol{x}_1) \oplus f_1(\boldsymbol{x}_2) \neq f_1(\boldsymbol{y}_1) \oplus f_1(\boldsymbol{y}_2)$. Однако очевидно, что только одно из этих соотношений может быть справедливым.

Если $2^{n-1} > |D_0^\star|$, то $2^{n-1} > |D_1^\star|$ и поэтому можно снова воспользоваться леммой 3.3.1, применив ее к новому множеству D_1. Из этой леммы следует существование линейного $(n-2,n-1)$-оператора f_2 такого, что $f_2(\boldsymbol{x}) \neq f_2(\boldsymbol{y})$ для любых неравных наборов $\boldsymbol{x}$ и $\boldsymbol{y}$ из D_1. Положим $D_2 = f_1(D_1)$. Как и в предыдущем случае, легко видеть, что $|D_2^\star| \leq |D_1^\star|$. Заметим, что композиция $f_2 \circ f_1$ операторов f_2 и f_1 будет инъективным на D линейным $(n-2,n)$-оператором.

Предположим, что описанную процедуру выполнили в общей сложности $k-1$ раз и для каждого целого i от единицы до $k-1$ получили инъективный на области D_{i-1} линейный $(n-i+1,n-i)$-оператор f_i и лежащее в $\mathbb{B}^{n-i}$ множество D_i такие, что $D_i = f_i(D_{i-1})$, $|D_i^\star| \leq |D_{i-1}^\star|$, а композиция композиция $f_{k-1} \circ \cdots \circ f_1$ является инъективным на области D линейным $(n-k+1,n)$-оператором.

Если $2^{n-k+1} > |D_{k-1}^\star| + 1$, то лемму 3.3.1 можно применить еще раз. Так как по предположению линейный $(n-k+1,n)$-оператор $f_{k-1} \circ \cdots \circ f_1$ отображает разные наборы области D в разные наборы ее образа D_{k-1}, а линейный $(n-k,n-k+1)$-оператор f_k действует на D_{k-1} инъективно, то легко видеть, что композиция $f_k \circ (f_{k-1} \circ \cdots \circ f_1)$, полученных в результате применения леммы 3.3.1 операторов f_i, будет инъективным на области D линейным $(n-k,n)$-оператором.

Наконец заметим, что при $k \leq m$ из условий теоремы и сделанного предположения следуют неравенства

$$2^{n-k+1} \geq 2^{n-m+1} > |D_0^\star| \geq |D_{k-2}^\star| \geq |D_{k-1}^\star|.$$

Поэтому очевидно, что леммой 3.3.1 можно воспользоваться в общей сложности не менее m раз, а получившийся в результате линейный $(n-m,n)$-оператор $f_m \circ \cdots \circ f_1$ будет инъективным на области D. Теорема доказана.

Пример 3.3.1. Найдем линейный оператор, действующий инъективно на множестве

$$D = \left\{ \begin{pmatrix}1\\0\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\0\\1\end{pmatrix} \right\}.$$

Так как $\binom{4}{2} + 1 < 2^3$, то теорема 3.3.1 гарантирует существование инъективного на D линейного $(2,4)$-оператора. Этот оператор найдем в соответствии с алгоритмом, изложенном в доказательстве теоремы 3.3.1.

Множество попарных сумм $D^\star$ состоит из всех векторов веса два, и, следовательно, вектор (1000) не принадлежит $D^\star$. Поэтому в качестве $(3,4)$-оператора f_1 возьмем оператор, ядро которого порождается вектором (1000). Вектор (1000) будем рассматривать как систематическую матрицу с одной строкой и четырьмя столбцами. В соответствии с алгоритмом построения ортогонального пространства систематической матрицы, изложенном на стр. 52, пространство, ортогональное вектору (1000), порождается строками матрицы

$$\mathbf{F}_1 = \begin{pmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix},$$

которая, таким образом будет матрицей оператора f_1. Теперь для определения $(2,3)$-оператора f_2 рассмотрим множество $D_1 = f_1(D)$, которое, как легко видеть, состоит из

столбцов матрицы $\mathbf{F}_1$. Так же легко видеть, что множество его попарных сумм D_1^* состоит из всех векторов длины три, вес которых не превосходит двух. Следовательно, вектор (111) не принадлежит этому множеству. В качестве $(2,3)$-оператора f_2 возьмем оператор, ядро которого порождается вектором (111). Рассматривая вектор (111) как систематическую матрицу с одной строкой и тремя столбцами, находим, что его ортогональное пространство порождается матрицей

$$\mathbf{F}_2 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}.$$

Следовательно, матрица $\mathbf{F}_2$ будет матрицей оператора f_2. Наконец умножая матрицы $\mathbf{F}_1$ и $\mathbf{F}_2$ найдем матрицу $\mathbf{F}$ композиции $f_2 \circ f_1$ операторов f_1 и f_2:

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{pmatrix}.$$

Таким образом, инъективный на множестве D линейный $(2,4)$-оператор задается найденной матрицей $\mathbf{F}$. $\square$

Из теоремы 3.3.1 легко извлекается верхняя оценка на ранг инъективного на произвольном множестве линейного оператора. Соответствующую теорему приведем без доказательства.

Теорема 3.3.2. *Для любой области $D \subseteq \mathbb{B}^n$, состоящей не более чем из $\sqrt{2^n}$ наборов, найдется линейный (m,n)-оператор совершенного хеширования, для числа компонент которого справедливо неравенство*

$$m \leq \lfloor 2\log_2 |D| \rfloor - 1.$$

2. Множество $\mathcal{F}$, состоящее из (m,n)-операторов, называется *универсальным хеширующим множеством* множества $\mathcal{D}$, состоящего из областей $D \subseteq \mathbb{B}^n$, если для любой области D из $\mathcal{D}$ в $\mathcal{F}$ найдется инъективный на D оператор f.

Для множества $D_{n,d}$, состоящего из всех d-элементных подмножеств n-мерного булева куба, оценим мощность его универсального хеширующего множества. Имеет место следующий результат.

Теорема 3.3.3. *При любом d, не превосходящем $\sqrt{2^n}$, для $D_{n,d}$ существует универсальное хеширующее множество $\mathcal{L}_{n,d}$, состоящее не более чем из $\lceil \log_2 \binom{2^n}{d} \rceil$ линейных $(\lceil 2\log_2 d \rceil, n)$-операторов.*

Доказательство теоремы опирается на вспомогательную лемму.

Лемма 3.3.2. *Пусть $D \subseteq \mathbb{B}^n$, $|D| \leq \sqrt{2^n}$, $m = \lceil 2\log_2 |D| \rceil$. Тогда не менее половины линейных (m,n)-операторов являются инъективными на области D.*

Доказательство. Допустим, что утверждение леммы неверно. Тогда для каждой области D, удовлетворяющей условиям леммы, найдется более $\frac{1}{2}2^{mn}$ линейных операторов, каждый из которых отображает какую-либо пару наборов из D в один набор. Так как число пар различных элементов области D равно $\frac{1}{2}|D|(|D|-1)$, то в D обязательно найдутся два набора $\boldsymbol{x}$ и $\boldsymbol{y}$, которые отображаются в один набор одновременно более чем

$$\frac{2^{mn}}{|D|(|D|-1)}$$

линейными (m,n)-операторами. Очевидно, что набор $\boldsymbol{x} \oplus \boldsymbol{y}$ принадлежит ядру каждого из этих операторов. С другой стороны, фиксированный набор принадлежит ядру ровно 2^{mn-m} линейных (m,n)-операторов. Следовательно,

$$\frac{2^{mn}}{|D|(|D|-1)} < 2^{mn-m}.$$

Откуда после несложных преобразований получаем неравенство

$$2^m < |D|(|D|-1),$$

которое очевидно противоречит условию леммы, наложенному на m. Следовательно, сделанное предположение ложно. Лемма доказана.

Доказательство теоремы 3.3.3. Положим $m = \lceil 2\log_2 d \rceil$. Построим прямоугольную таблицу $\mathbf{T}$, состоящую из 2^{mn} строк и $\binom{2^n}{d}$ столбцов. Каждой строке поставим в соответствие линейный (m,n)-оператор, а каждому столбцу — область из $D_{n,d}$. В таблице на пересечении строки, соответствующей оператору f, и столбца, соответствующего области D, поставим единицу, если оператор f действует на D инъективно, в противном случае поставим нуль. Будем говорить, что i-я строка таблицы $\mathbf{T}$ покрывает ее j-й столбец, если элемент t_{ij}, стоящий в таблице на пересечении i-й строки и j-го столбца, равен единице. Из леммы 3.3.2 следует, что в построенной таблице каждый столбец содержит не менее $\frac{1}{2}2^{mn}$ единиц.

Нетрудно видеть, что для доказательства теоремы достаточно установить существование в таблице $\mathbf{T}$ таких $\lceil \log_2 \binom{2^n}{d} \rceil$ строк, которые вместе покрывают все столбцы.

Будем выбирать строки по одной так, чтобы очередная выбираемая строка покрывала наибольшее число еще не покрытых столбцов. Обозначим через γ_k долю столбцов, покрытых первыми k выбранными строками, т.е. первые k строк покрывают $\gamma_k\binom{2^n}{d}$ столбцов. Тогда в оставшихся непокрытыми $(1-\gamma_k)\binom{2^n}{d}$ столбцах содержится не менее $\frac{1}{2}(1-\gamma_k)\binom{2^n}{d}2^{mn}$ единиц. Следовательно, среди невыбранных строк найдется строка, в которой на пересечении с непокрытыми столбцами содержится не менее чем

$$\frac{\frac{1}{2}(1-\gamma_k)\binom{2^n}{d}2^{mn}}{2^{mn}-k} > \frac{1}{2}(1-\gamma_k)\binom{2^n}{d}$$

единиц. Тогда, добавив эту строку в покрытие, видим, что для числа столбцов, покрытых $(k+1)$ строками, справедливо неравенство

$$\gamma_{k+1}\binom{2^n}{d} > \gamma_k\binom{2^n}{d} + \frac{1}{2}(1-\gamma_k)\binom{2^n}{d}.$$

Следовательно,

$$\gamma_{k+1} > \frac{1}{2}(1+\gamma_k).$$

Откуда для числа непокрытых столбцов $(1-\gamma_{k+1})\binom{2^n}{d}$ немедленно получаем неравенства

$$(1-\gamma_{k+1})\binom{2^n}{d} < \frac{1}{2}(1-\gamma_k)\binom{2^n}{d} < \cdots < \frac{1}{2^k}(1-\gamma_1)\binom{2^n}{d} < \frac{1}{2^{k+1}}\binom{2^n}{d}.$$

Полагая $k = \lceil \log_2 \binom{2^n}{d} \rceil$, видим, что $(1-\gamma_k)\binom{2^n}{d} < 1$, т.е. в таблице нет ни одного не покрытого столбца. Теорема доказана.

Использованный в доказательстве теоремы метод называется градиентным.

3. Теперь покажем, что для любого целого m, не превосходящего $\frac{n}{2}$, в n-мерном булевом кубе найдется область D_m, состоящая из $2^{m+1}-1$ наборов, и такая, что число компонент любого линейного оператора совершенного хеширования этой области не меньше $2m$.

Пусть $m \le \frac{n}{2}$ и $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{2m}$ — первые $2m$ базисных векторов стандартного базиса E_n. Положим

$$D_m = \langle \boldsymbol{e}_1, \ldots, \boldsymbol{e}_m \rangle \cup \langle \boldsymbol{e}_{m+1}, \ldots, \boldsymbol{e}_{2m} \rangle.$$

Легко видеть, что D_m состоит из $2^{m+1}-1$ различных наборов, а множество $D_m^\star$ попарных сумм наборов из D_m вместе с нулевым набором образуют подпространство размерности $2m$ в $\mathbb{B}^n$. Поэтому, очевидно, что размерность любого подпространства, не пересекающегося с $D_m^\star$, не превосходит $n-2m$. Следовательно, ранг любого линейного оператора совершенного хеширования области D_m не меньше чем $2m$.

Так как $\lfloor 2\log_2 |D_m| \rfloor - 1 = 2m$, то теорема 3.3.2 гарантирует существование линейного $(2m, n)$-оператора совершенного хеширования D_m. Таким образом в общем случае неравенство теоремы 3.3.2 является точным и усилить его нельзя.

Интересно отметить, что существует достаточно простой нелинейный $(m+1, n)$-оператор совершенного хеширования области D_m с единственной нелинейной компонентой. Этот оператор задается следующими равенствами:

$$y_1 = x_1 \oplus x_{m+1}, \ldots\ldots\ldots, \; y_m = x_m \oplus x_{2m},$$
$$y_{m+1} = (x_1 \vee \ldots \vee x_m)\&\overline{(x_{m+1} \vee \ldots \vee x_{2m})}.$$

Далее покажем, что неравенство теоремы 3.3.2 является асимптотически точным для почти всех областей из $D_{n,d}$ при условии, что $\frac{\log_2 d}{\log_2 n}$ неограниченно возрастает при $n \to \infty$. Сделаем это следующим образом. Сначала для произвольной области D из $\mathbb{B}^n$ введем функцию μ, определив ее равенством

$$\mu(D) = \min \operatorname{rank} f,$$

в котором минимум берется по всем инъективным на области D линейным операторам. Затем величину $\mu(D)$ оценим снизу для почти всех областей из $D_{n,d}$.

Теорема 3.3.4. *Пусть $n \le d \le \sqrt{n^2 2^n}$. Тогда при $n \to \infty$ для почти всех $D \in D_{n,d}$*

$$\mu(D) \ge 2\log_2 d - 2\log_2 n - 2.$$

Доказательство. Пусть f — произвольный линейный (m, n)-оператор. Через $M(f, d)$ обозначим число областей из $D_{n,d}$, для которых оператор f является оператором совершенного хеширования. Если f — оператор совершенного хеширования области D, то легко видеть, что никакие два набора из D не принадлежат одному и тому же смежному классу пространства $\mathbb{B}^n$ по ядру оператора f. Поэтому

$$M(f, d) = \binom{2^m}{d} 2^{(n-m)d}. \tag{3.3.2}$$

Теперь предположим, что при некоторой постоянной α не менее чем для $\alpha\binom{2^n}{d}$ областей из $D_{n,d}$ среди линейных (m, n)-операторов найдутся операторы совершенного хеширования. Так как число различных линейных (m, n)-операторов равно 2^{mn}, то в среднем каждый линейный (m, n)-оператор является оператором совершенного хеширования не менее чем для $\alpha\binom{2^n}{d}2^{-mn}$ областей мощности d. Следовательно, найдется оператор, который будет оператором совершенного хеширования по крайней мере для

$$P = \alpha\binom{2^n}{d}2^{-mn}$$

различных областей. С другой стороны, необходимо, чтобы величина P не превосходила $M(f, d)$. Поэтому из (3.3.2) и последнего неравенства

$$\alpha\binom{2^n}{d}2^{-mn} \le \binom{2^m}{d}2^{(n-m)d}.$$

Откуда после несложных преобразований получаем

$$\binom{2^n}{d} \Big/ \binom{2^m}{d} \le \frac{1}{\alpha} 2^{mn} 2^{(n-m)d}.$$

Легко видеть, что

$$\binom{2^n}{d} \Big/ \binom{2^m}{d} = \frac{2^n(2^n-1)\ldots(2^n-d+1)}{2^m(2^m-1)\ldots(2^m-d+1)}$$

Оценим снизу правую часть последнего равенства. Для этого в числителе заменим каждый из последних $(d-1)$ сомножителей величиной (2^n-d). Оценивая знаменатель, воспользуемся тем, что среднее арифметическое не меньше среднего геометрического. Поэтому в знаменателе произведение последних $(d-1)$ сомножителей заменим $(d-1)$-й степенью их среднего арифметического. При этом будем полагать, что m не больше $n-2$. В противном случае

$$m \geq n-2 \geq 2\log_2 d - 2\log_2 n - 2,$$

и неравенство леммы справедливо. Далее, так как $\frac{1}{d-1}\sum_{j=1}^{d-1} j = \frac{d(d-1)}{2(d-1)} = \frac{d}{2}$, $\frac{1-x}{1-y} \geq 1-x+y$ при $1 > y \geq x \geq 0$ и $m+2 \leq n$, то

$$\binom{2^n}{d} \Big/ \binom{2^m}{d} = \frac{2^n}{2^m}\left(\frac{2^n(1-d/2^n)}{2^m(1-d/2^{m+1})}\right)^{d-1} \geq$$

$$\geq 2^{(n-m)d}\left(1+d\left(\frac{1}{2^{m+1}}-\frac{1}{2^n}\right)\right)^{d-1} \geq 2^{(n-m)d}\left(1+\frac{d}{2^{m+2}}\right)^{d-1}.$$

Следовательно,

$$\left(1+\frac{d}{2^{m+2}}\right)^{d-1} \leq \frac{1}{\alpha}2^{mn}.$$

Так как $\left(1+\frac{1}{x}\right)^x > 2$ при $x > 1$ и $m \leq n-2$, то для любой постоянной α при $n \to \infty$ из последнего неравенства имеем

$$2^{n(n-1)} \geq \frac{1}{\alpha}2^{mn} \geq \left(1+\frac{d}{2^{m+2}}\right)^{\frac{2^{m+2}}{d}\cdot\frac{d(d-1)}{2^{m+2}}} > 2^{\frac{d(d-1)}{2^{m+2}}}.$$

Логарифмируя полученное неравенство и выделяя 2^m, после простых преобразований при $d \geq n$ получаем

$$2^m > \frac{d(d-1)}{4n(n-1)} \geq \left(\frac{d}{2n}\right)^2.$$

Теорема доказана.

Задачи

3.3.1. Доказать теорему 3.3.2.

3.3.2. Пусть $D \subset \mathbb{B}^n$, $|D| \leq \sqrt{2^n}$. Показать, что существует такой набор $\boldsymbol{\alpha} \in \mathbb{B}^n$, что $D \cap (\boldsymbol{\alpha} \oplus D) = \varnothing$.

3.3.3. Показать, что если универсальное хеширующее множество множества $D_{n,d}$ состоит только из линейных операторов, то оно состоит не менее чем из $d-1$ операторов.

3.4. Линейные коды

1. Напомним, что подмножество n-мерного булева куба $G = \{\boldsymbol{g}_1, \ldots, \boldsymbol{g}_m\}$ называется кодом с кодовым расстоянием d, если для любых двух его элементов $\boldsymbol{g}_i$ и $\boldsymbol{g}_j$ расстояние между ними не меньше d. Так же говорят, что код G исправляет t ошибок, если его кодовое расстояние не меньше чем $2t+1$.

Определение для произвольного вектора $\boldsymbol{v} \in \mathbb{B}^n$ векторов $\boldsymbol{g} \in G$ и $\boldsymbol{c} \in \mathbb{B}^n$ таких, что расстояние от вектора $\boldsymbol{v}$ до вектора $\boldsymbol{g}$ меньше чем расстояние от $\boldsymbol{v}$ до любого другого элемента G, а $\boldsymbol{c} = \boldsymbol{g} \oplus \boldsymbol{v}$, называется *декодированием* вектора $\boldsymbol{v}$, или, *исправлением ошибок* в векторе $\boldsymbol{v}$. Вектор $\boldsymbol{c}$ называется *вектором ошибок*.

Код G называется *линейным* (n,k)-кодом, если он является k-мерным линейным подпространством пространства $\mathbb{B}^n$. Булева $(n-k,n)$-матрица $\mathbf{H}$ называется *проверочной* матрицей линейного кода G, если $\mathbf{H}\boldsymbol{g} = \mathbf{0}$ для каждого $\boldsymbol{g} \in G$ и $\mathbf{H}\boldsymbol{x} \neq \mathbf{0}$ для каждого $\boldsymbol{x} \notin G$.

Булева (k,n)-матрица $\mathbf{G}$ называется *порождающей* матрицей линейного кода G, если линейная оболочка $\langle \boldsymbol{g}_1, \ldots, \boldsymbol{g}_k \rangle$ строк матрицы $\mathbf{G}$ совпадает с G. Легко видеть, что проверочная и порождающая матрицы любого линейного кода связаны равенством $\mathbf{H}\mathbf{G}^{\mathrm{T}} = \mathbf{0}$.

Теорема 3.4.1. *В каждом линейном коде G кодовое расстояние t равно весу его минимального ненулевого элемента*:

$$t = \min_{\boldsymbol{g} \neq \mathbf{0}, \boldsymbol{g} \in G} \|\boldsymbol{g}\|.$$

Доказательство. Так как нулевой набор всегда принадлежит линейному коду, то очевидно, что кодовое расстояние не превосходит веса минимального ненулевого элемента.

Теперь допустим, что $t < \min\|\boldsymbol{g}\|$. В этом случае в G найдутся два элемента $\boldsymbol{g}_1$ и $\boldsymbol{g}_2$, расстояние между которыми меньше t. Следовательно,

$$\|\boldsymbol{g}_1 \oplus \boldsymbol{g}_2\| = d(\boldsymbol{g}_1, \boldsymbol{g}_2) < t.$$

С другой стороны, сумма $\boldsymbol{g}_1 \oplus \boldsymbol{g}_2$ обязательно принадлежит G. Поэтому $\|\boldsymbol{g}_1 \oplus \boldsymbol{g}_2\| \geq t$. Пришли к противоречию. Теорема доказана.

Теорема 3.4.2. *Пусть f — линейный (m,n)-оператор совершенного хеширования шара $B_{n,t}(\mathbf{0})$ радиуса t с центром в нулевом наборе. Тогда ядро оператора f является линейным $(n, n-m)$-кодом с кодовым расстоянием $2t+1$.*

Доказательство. Пусть L — линейный (m,n)-оператор, удовлетворяющий условиям теоремы, $\boldsymbol{x}$ и $\boldsymbol{y}$ — произвольные наборы из его ядра. Так как ядро (m,n)-оператора содержит не меньше чем 2^{n-m} наборов, то для доказательства теоремы достаточно показать, что расстояние между $\boldsymbol{x}$ и $\boldsymbol{y}$ не меньше $2t+1$. Если $d(\boldsymbol{x},\boldsymbol{y}) \leq 2t$, то в $\mathbb{B}^n$ найдутся наборы $\boldsymbol{x}'$ и $\boldsymbol{y}'$ такие, что $\|\boldsymbol{x}'\| \leq t$, $\|\boldsymbol{y}'\| \leq t$ и $\boldsymbol{x}' \oplus \boldsymbol{y}' = \boldsymbol{x} \oplus \boldsymbol{y}$. Тогда

$$L(\boldsymbol{x}') \oplus L(\boldsymbol{y}') = L(\boldsymbol{x}' \oplus \boldsymbol{y}') = L(\boldsymbol{x} \oplus \boldsymbol{y}) = L(\boldsymbol{x}) \oplus L(\boldsymbol{y}) = \mathbf{0}.$$

Следовательно, $L(\boldsymbol{x}') = L(\boldsymbol{y}')$. С другой стороны, наборы $\boldsymbol{x}'$ и $\boldsymbol{y}'$ лежат в шаре $B_{n,t}(\mathbf{0})$. Поэтому, $L(\boldsymbol{x}') \neq L(\boldsymbol{y}')$. Противоречие. Следовательно, $d(\boldsymbol{x},\boldsymbol{y}) \geq 2t+1$. Теорема доказана.

Из предыдущей теоремы и теоремы 3.3.2 легко следует утверждение о существовании достаточно хороших линейных кодов.

Теорема 3.4.3. *Существует линейный (n,m)-код с минимальным расстоянием $2t+1$, параметры которого n, m и t удовлетворяют неравенству*

$$2^{n-m+1} > \sum_{i=0}^{2t} \binom{n}{i}.$$

2. Построим линейный $(2^n-1, 2^n-n-1)$-код H_n, исправляющий одну ошибку в наборах длины 2^n-1. Искомый код зададим при помощи его проверочной матрицы $\mathbf{H}_n$.

Пусть $\mathbf{H}_n = (h_{ij})$ — булева $(n, 2^n-1)$-матрица, у которой j-й столбец $\boldsymbol{h}_j = (h_{1j}, \ldots, h_{nj})$ совпадает с двоичным разложение числа j. Например, матрица $\mathbf{H}_3$ выглядит следующим образом:

$$\mathbf{H}_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Легко видеть, что $\mathbf{H}_n \boldsymbol{e}_j = \boldsymbol{h}_j$, и, следовательно, $\mathbf{H}_n \boldsymbol{e}_i \neq \mathbf{H}_n \boldsymbol{e}_j$ если $i \neq j$. Поэтому соответствующий матрице $\mathbf{H}_n$ линейный $(n, 2^n-1)$-оператор будет оператором совершенного хеширования шара единичного радиуса с центром в нулевом наборе. Таким образом, из теоремы 3.4.2 следует, что матрица $\mathbf{H}_n$ является проверочной матрицей кода, исправляющего одну ошибку в наборах длины 2^n-1. Код с проверочной матрицей $\mathbf{H}_n$ называется кодом Хемминга. Исправление ошибки в векторе $\boldsymbol{v}$ при использовании кода Хемминга выполняется очень просто. Легко видеть, что вектор $\mathbf{H}_m \boldsymbol{v}$ состоит из одних нулей, если ошибки не было, а если одиночная ошибка присутствует, то вектор $\mathbf{H}_m \boldsymbol{v}$ является двоичным представлением номера ошибочного разряда.

Задачи

3.4.1. Показать, что если линейный (n,k)-код имеет минимальное расстояние d, то $n-k+1 \geq d$.

3.4.2. Построить порождающую матрицу кода Хемминга длины 7.

3.4.3. Пусть G — линейный (n,k)-код. Доказать, что если код G имеет хотя бы одно слово нечетного веса, то все кодовые слова четного веса образуют $(n,k-1)$-код.

3.4.4. Пусть в (k,n)-матрице $\mathbf{H}$ любые $t-1$ столбцов линейно независимы. Показать, что матрица $\mathbf{H}$ является проверочной матрицей линейного $(n,n-k)$-кода с кодовым расстоянием не меньшим t.

3.4.5. (Граница Варшамова–Гилберта) Показать, что существует линейный (n,k)-код с минимальным расстоянием, не меньшим d, параметры которого n, k и d удовлетворяют неравенству $\sum_{i=0}^{d-2} \binom{n}{i} \geq 2^{n-k}$.

3.5. Коды Рида–Малера

Кодом Рида–Малера $RM(n,k)$ длины 2^n порядка k называется множество векторов значений всех n-местных булевых функций, степени многочленов Жегалкина которых не превосходят k.[1)]

Пример 3.5.1. Множество $RM(2,0)$ состоит из векторов значений двух булевых констант, т. е. $RM(2,0) = \{(0000),(1111)\}$. Множество $RM(2,1)$ состоит из векторов значений всех линейных функций двух переменных x_1 и x_2:

$$\begin{aligned} 0 &= (0000), & x_1 &= (0011), \\ 1 &= (1111), & 1 \oplus x_1 &= (1100), \\ x_2 &= (0101), & x_1 \oplus x_2 &= (0110), \\ 1 \oplus x_2 &= (1010), & 1 \oplus x_2 \oplus x_1 &= (1001). \end{aligned}$$

Наконец множество $RM(2,2)$ совпадает с $P_2(2)$. Нетрудно видеть, что рассмотренные множества являются линейными кодами длины четыре с кодовыми расстояниями равными, соответственно, четырем, двум и единице. □

Теорема 3.5.1. *Код Рида–Малера $RM(n,k)$ длины 2^n порядка k является линейным кодом длины 2^n с кодовым расстоянием 2^{n-k}.*

Доказательство. Теорему докажем индукцией по числу переменных n. При $n=2$ утверждение теоремы следует из рассмотренного выше примера.

Предположим, что теорема верна при всех n не превосходящих некоторое $m-1 \geq 2$. Покажем, что из этого предположения следует утверждение теоремы для $n=m$.

Так как сумма двух функций степени $\leq k$ так же будет функцией степени $\leq k$, то очевидно, что множество $RM(n,k)$ является линейным пространством. Поэтому в силу теоремы 3.4.1 достаточно показать, что вес каждой ненулевой функции из $RM(n,k)$ не меньше чем 2^{m-k}.

Пусть f — произвольная m-местная булева функция степени k. Если $k=m$, то утверждение теоремы очевидно, так как $\|f\| \geq 1$. Поэтому далее полагаем, что $k<m$. В многочлене Жегалкина функции f соберем вместе одночлены содержащие и не содержащие x_m. В результате получим равенство

$$f(x_1,\ldots,x_m) = x_m f_1(x_1,\ldots,x_{m-1}) \oplus f_2(x_1,\ldots,x_{m-1}), \tag{3.5.1}$$

[1)] Здесь, как и в разделе 3.2, для краткости будем говорить не о векторах значений булевых функций, а просто о функциях.

где $f_1 \in RM(m-1,k-1)$ и $f_2 \in RM(m-1,k)$. Нетрудно видеть, что $f_1 \oplus f_2 \in RM(m-1,k)$. Если функций f_1, f_2 и $f_1 \oplus f_2$ отличны от тождественного нуля, то по предположению индукции для этих функций справедливы неравенства

$$\|f_2\| \geq 2^{m-k}, \quad \|f_2\| \geq 2^{m-1-k}, \quad \|f_1 \oplus f_2\| \geq 2^{m-1-k}. \tag{3.5.2}$$

Для каждого $\boldsymbol{x}$ из $\mathbb{B}^m$ через $\boldsymbol{x}'$ обозначим первые $(m-1)$ координат набора $\boldsymbol{x}$. Булев куб $\mathbb{B}^m$ разобьем на два непересекающихся подмножества B_0^m и B_1^m, первое из которых состоит из всех наборов с последним разрядом равным нулю, а второе — из всех наборов с последним разрядом равным единице. Оценим вес функции f. Из (3.5.1) имеем

$$\begin{aligned}\|f\| &= \sum_{\boldsymbol{x} \in B_1^m} (x_m f_1(\boldsymbol{x}') \oplus f_2(\boldsymbol{x}')) + \sum_{\boldsymbol{x} \in B_0^m} (x_m f_1(\boldsymbol{x}') \oplus f_2(\boldsymbol{x}')) = \\ &= \sum_{\boldsymbol{x}' \in \mathbb{B}^{m-1}} (f_1(\boldsymbol{x}') \oplus f_2(\boldsymbol{x}')) + \sum_{\boldsymbol{x}' \in \mathbb{B}^{m-1}} f_2(\boldsymbol{x}') = \|f_1 \oplus f_2\| + \|f_2\|.\end{aligned}$$

Далее рассмотрим три случая.

1. Функции $f_1 \oplus f_2$ и f_2 отличны от тождественного нуля. Из (3.5.2) имеем

$$\|f\| \geq \|f_1 \oplus f_2\| + \|f_2\| \geq 2^{m-1-k} + 2^{m-1-k} = 2^{m-k}.$$

2. Функция $f_1 \oplus f_2$ отлична от тождественного нуля и $f_2 \equiv 0$. Тогда $f_1 \oplus f_2 = f_1 \in RM(m-1,k-1)$. Из (3.5.2) имеем

$$\|f\| = \|f_1 \oplus f_2\| = \|f_1\| \geq 2^{m-k}.$$

3. Функция f_2 отлична от тождественного нуля и $f_1 \oplus f_2 \equiv 0$. Тогда $f_2 \equiv f_1 \in RM(m-1,k-1)$. Из (3.5.2) имеем

$$\|f\| = \|f_2\| = \|f_1\| \geq 2^{m-k}.$$

Теорема доказана.

Для кодов Рида–Малера существует простой алгоритм исправления ошибок, основанный на рассматриваемом ниже методе определения коэффициентов многочлена Жегалкина булевой функции. Для функции f, степень которой не превосходит k, где $1 \leq k \leq n-2$, рассматриваемый метод позволяет правильно определять коэффициенты многочлена Жегалкина даже в том случае, когда вместо вектора значений функции f известен вектор значений функции $f \oplus c$, где c — булева функция, вес которой не превосходит $2^{n-k-1}-1$. Опишем этот метод.

Сначала рассмотрим способ определения коэффициента при одночлене $x_1 \cdot \ldots \cdot x_k$. Для этого одночлена определим разбиение булева куба $\mathbb{B}^n$ на 2^{n-k} подмножеств B_i, $i = 0, 1, \ldots, 2^{n-k}-1$, так, что подмножество B_i состоит из всех тех наборов, у которых на последних $n-k$ местах стоят такие константы $\beta_{k+1}, \ldots, \beta_n$, что $i = \sum_{j=1}^{n-k} \beta_{k+j} 2^{n-k-j}$. Очевидно, что каждое из этих подмножеств является подкубом размерности k. Например, для одночлена $x_1 x_2$, рассматриваемого как функция переменных x_1, x_2, x_3 и x_4, описываемое разбиение куба состоит из подмножеств

$$\begin{aligned}B_0 &= \{(0000), (0100), (1000), (1100)\}, \quad B_1 = \{(0001), (0101), (1001), (1101)\}, \\ B_2 &= \{(0010), (0110), (1010), (1110)\}, \quad B_3 = \{(0011), (0111), (1011), (1111)\}.\end{aligned}$$

Ограничением булевой функции $f(x_1, \ldots, x_n)$ на множество B_i, где $i = \sum_{j=1}^{n-k} \beta_{k+j} 2^{n-k-j}$, называется функция $f(x_1, \ldots, x_k, \beta_{k+1}, \ldots, \beta_n)$, получающаяся из f подстановкой констант $\beta_{k+1}, \ldots, \beta_n$ вместо переменных $x_{k+1}, \ldots, x_n$. Очевидно, что ограничение одночлена $x_1 \cdot \ldots \cdot x_k$ на любое из множеств B_i получается из исходного одночлена удалением всех его фиктивных переменных, равно единице на единственном наборе $x_1 = 1, \ldots, x_k = 1$, и поэтому для каждого $i \in \{0, 1, \ldots, 2^{n-k}-1\}$ справедливы равенства

$$\bigoplus_{\boldsymbol{x} \in B_i} x_1 \cdot \ldots \cdot x_k = \bigoplus_{x_1, \ldots, x_k \in \mathbb{B}^k} x_1 \cdot \ldots \cdot x_k = 1. \tag{3.5.3}$$

Легко видеть, что ограничение любого другого одночлена $x_{i_1}\cdot\ldots\cdot x_{i_m}$ степени не больше k на любое множество B_i будет одночленом степени строго меньшей k. Поэтому, такое ограничение обязательно будет иметь хотя бы одну фиктивную переменную, его вес будет четным числом, и, следовательно, для каждого $i\in\{0,1,\ldots,2^{n-k}-1\}$ будут справедливы равенства

$$\bigoplus_{\boldsymbol{x}\in B_i} x_{i_1}\cdot\ldots\cdot x_{i_m} = \bigoplus_{\substack{x_1,\ldots,x_k\in\mathbb{B}^k\\ x_{k+1}=\beta_{k+1},\ldots,x_n=\beta_n}} x_{i_1}\cdot\ldots\cdot x_{i_m}=0. \tag{3.5.4}$$

Для любой булевой функции $f(x_1,\ldots,x_k)$, степень которой не превосходит k, равенства (3.5.3) и (3.5.4) позволяют определить входит ли одночлен $x_1\cdot\ldots\cdot x_k$ в многочлен Жегалкина этой функции. Действительно, для любой f и для любого $\boldsymbol{x}$ значение $f(\boldsymbol{x})$ равно взятой по модулю два сумме значений одночленов, входящих в многочлен Жегалкина f. Поэтому, сумма значений функции f на любом из множеств B_i, взятая по модулю два, равна единице только в том случае, когда одночлен $x_1\cdot\ldots\cdot x_k$ входит в ее многочлен Жегалкина. Таким образом, если степень булевой функции f не превосходит k, то существуют 2^{n-k} независимых соотношений для определения вхождения одночлена $x_1\cdot\ldots\cdot x_k$ в многочлен Жегалкина f.

Теперь рассмотрим сумму $f(\boldsymbol{x})\oplus c(\boldsymbol{x})$ булевых функций $f(\boldsymbol{x})$ и $c(\boldsymbol{x})$ таких, что $\deg f\le k$ и $\|c\|<2^{n-k-1}$. Для каждого $i\in\{0,1,\ldots,2^{n-k}-1\}$ вычислим две суммы

$$\bigoplus_{\boldsymbol{x}\in B_i}(f(\boldsymbol{x})\oplus c(\boldsymbol{x})),\qquad \bigoplus_{\boldsymbol{x}\in B_i} f(\boldsymbol{x}), \tag{3.5.5}$$

которые очевидно различаются не более чем на $\|c\|$ множествах B_i. Поэтому для определения коэффициента при одночлене $x_1\cdot\ldots\cdot x_k$ в многочлен Жегалкина функции f можно использовать вектор значений функции $f\oplus c$. Достаточно вычислить все суммы $\bigoplus_{\boldsymbol{x}\in B_i}(f\oplus c)(\boldsymbol{x})$. Эти суммы называются проверочными суммами для одночлена $x_1\cdot\ldots\cdot x_k$. Если больше половины проверочных сумм равны единице, то одночлен $x_1\cdot\ldots\cdot x_k$ входит в многочлен Жегалкина функции f, а если больше половины вычисленных сумм равны нулю, то не входит.

Определение коэффициента при произвольном одночлене $x_{i_1}\cdot\ldots\cdot x_{i_k}$ степени k отличается от приведенного выше метода только способом определения множеств B_i — в случае одночлена $x_{i_1}\cdot\ldots\cdot x_{i_k}$ каждое B_i состоит из всех тех наборов, у которых принимают фиксированные значения разряды с индексами, не принадлежащими множеству $\{i_1,\ldots,i_k\}$. Например, для одночлена x_1x_3, рассматриваемого как функция переменных x_1,x_2,x_3 и x_4, множества B_i определяются следующим образом:

$$B_0=\{(0000),(0010),(1000),(1010)\},\quad B_1=\{(0001),(0011),(1001),(1011)\},$$
$$B_2=\{(0100),(0110),(1100),(1110)\},\quad B_3=\{(0101),(0111),(1101),(1111)\}.$$

После того как для функции f определены коэффициенты многочлена Жегалкина при всех одночленах степени k, к функции $f\oplus c$ прибавим все одночлены степени k, коэффициенты при которых равны единице. В результате получим новую функцию $f'\oplus c$, где $\deg f'\le k-1$, а вес c по прежнему не превосходит $2^{n-k-1}-1$. Теперь, используя вектор значений функции $f'\oplus c$, описанным выше способом определим в многочлене Жегалкина функции f коэффициенты при одночленах степени $k-1$. Затем одночлены степени $k-1$, коэффициенты при которых равны единице, прибавим к функции $f'\oplus c$. Очевидно, что степень новой функции $f''\oplus c$ не превосходит $k-2$.

Нетрудно убедиться в том, что повторяя приведенные вычисления для всех остальных степеней вплоть до нулевой, в результате получим функцию c. Итак, если в сумме $f\oplus c$ двух функций n переменных степень функции f не превосходит k, а вес функции c не превосходит $2^{n-k-1}-1$, то приведенный метод позволяет по известной сумме $f\oplus c$ определить функции f и c.

Пример 3.5.2. Рассмотрим вектор $\boldsymbol{v} = (1100\,1011\,0000\,0101)$. Будем полагать, что вектор $\boldsymbol{v}$ равен сумме $\boldsymbol{g} \oplus \boldsymbol{c}$ вектора $\boldsymbol{g}$, принадлежащего коду Рида–Малера второго порядка длины 16, и вектора ошибок $\boldsymbol{c}$, вес которого не превосходит единицы. Найдем векторы $\boldsymbol{g}$ и $\boldsymbol{c}$. Вектор $\boldsymbol{g}$ является вектором значений булевой функции четырех переменных степень которой не превосходит двух. Поэтому $\boldsymbol{g}$ равен некоторой линейной комбинации следующих векторов:

$$\begin{aligned}
1 &= (1111\,1111\,1111\,1111), & x_4x_3 &= (0001\,0001\,0001\,0001),\\
x_4 &= (0101\,0101\,0101\,0101), & x_4x_2 &= (0000\,0101\,0000\,0101),\\
x_3 &= (0011\,0011\,0011\,0011), & x_4x_1 &= (0000\,0011\,0000\,0011),\\
x_2 &= (0000\,1111\,0000\,1111), & x_3x_2 &= (0000\,0000\,0101\,0101),\\
x_1 &= (0000\,0000\,1111\,1111), & x_3x_1 &= (0000\,0000\,0011\,0011),\\
& & x_2x_1 &= (0000\,0000\,0000\,1111),
\end{aligned}$$

образующих базис рассматриваемого кода Рида–Малера. Определим проверочные суммы для одночлена x_4x_3. Фиксируя первый и второй разряды в двоичных наборах длины четыре, находим множества наборов B_i и множества B'_i номеров этих наборов:

$$\begin{aligned}
B_0 &= \{(0000), (0001), (0010), (0011)\}, & B'_0 &= \{0,1,2,3\},\\
B_1 &= \{(0100), (0101), (0110), (0111)\}, & B'_1 &= \{4,5,6,7\},\\
B_2 &= \{(1000), (1001), (1010), (1011)\}, & B'_2 &= \{8,9,10,11\},\\
B_3 &= \{(1100), (1101), (1110), (1111)\}, & B'_3 &= \{12,13,14,15\}.
\end{aligned}$$

Соответствующие найденным множествам проверочные суммы для слова $\boldsymbol{v}$ равны:

$$\begin{aligned}
S_0 &= v_0 \oplus v_1 \oplus v_2 \oplus v_3 = 0, & S_1 &= v_4 \oplus v_5 \oplus v_6 \oplus v_7 = 1,\\
S_2 &= v_8 \oplus v_9 \oplus v_{10} \oplus v_{11} = 0, & S_3 &= v_{12} \oplus v_{13} \oplus v_{14} \oplus v_{15} = 0.
\end{aligned}$$

Три суммы из четырех равны нулю. Следовательно, коэффициент при одночлене x_4x_3 в разложении вектора $\boldsymbol{v}$ в базисе одночленов равен нулю. Теперь определим множества B_i и B'_i для одночлена x_4x_2. Фиксируя первый и третий разряды двоичных наборов, имеем:

$$\begin{aligned}
B_0 &= \{(0000), (0001), (0100), (0101)\}, & B'_0 &= \{0,1,4,5\},\\
B_1 &= \{(0010), (0011), (0110), (0111)\}, & B'_1 &= \{2,3,6,7\},\\
B_2 &= \{(0000), (1001), (1100), (1101)\}, & B'_2 &= \{8,9,12,13\},\\
B_3 &= \{(1010), (1011), (1110), (1111)\}, & B'_3 &= \{10,11,14,15\}.
\end{aligned}$$

Соответствующие проверочные суммы для вектора $\boldsymbol{v}$ равны:

$$\begin{aligned}
S_0 &= v_0 \oplus v_1 \oplus v_4 \oplus v_5 = 1, & S_1 &= v_2 \oplus v_3 \oplus v_6 \oplus v_7 = 0,\\
S_2 &= v_8 \oplus v_9 \oplus v_{12} \oplus v_{13} = 1, & S_3 &= v_{10} \oplus v_{11} \oplus v_{14} \oplus v_{15} = 1.
\end{aligned}$$

Теперь три суммы из четырех равны единице. Следовательно, коэффициент при одночлене x_4x_2 в разложении вектора $\boldsymbol{v}$ в базисе одночленов равен единице. Проводя аналогичные вычисления для остальных одночленов второй степени, находим для x_4x_1:

$$\begin{aligned}
S_0 &= v_0 \oplus v_2 \oplus v_4 \oplus v_6 = 1, & S_1 &= v_1 \oplus v_3 \oplus v_5 \oplus v_7 = 0,\\
S_2 &= v_8 \oplus v_{10} \oplus v_{12} \oplus v_{14} = 0, & S_3 &= v_9 \oplus v_{11} \oplus v_{13} \oplus v_{15} = 0;
\end{aligned}$$

для x_3x_2:

$$\begin{aligned}
S_0 &= v_0 \oplus v_1 \oplus v_8 \oplus v_9 = 0, & S_1 &= v_2 \oplus v_3 \oplus v_{10} \oplus v_{11} = 0,\\
S_2 &= v_4 \oplus v_5 \oplus v_{12} \oplus v_{13} = 0, & S_3 &= v_6 \oplus v_7 \oplus v_{14} \oplus v_{15} = 1;
\end{aligned}$$

для x_3x_1:

$$S_0 = v_0 \oplus v_2 \oplus v_8 \oplus v_{10} = 1, \quad S_1 = v_1 \oplus v_3 \oplus v_9 \oplus v_{11} = 1,$$
$$S_2 = v_4 \oplus v_6 \oplus v_{12} \oplus v_{14} = 0, \quad S_3 = v_5 \oplus v_7 \oplus v_{13} \oplus v_{15} = 1;$$

для x_2x_1:

$$S_0 = v_0 \oplus v_4 \oplus v_8 \oplus v_{12} = 0, \quad S_1 = v_1 \oplus v_5 \oplus v_9 \oplus v_{13} = 0,$$
$$S_2 = v_2 \oplus v_6 \oplus v_{10} \oplus v_{14} = 1, \quad S_3 = v_3 \oplus v_7 \oplus v_{11} \oplus v_{15} = 0.$$

Таким образом, коэффициенты при одночленах x_4x_2 и x_3x_1 равны единице, а при одночленах x_4x_3, x_4x_1, x_3x_2 и x_2x_1 равны нулю. Найдем сумму $\boldsymbol{v}' = \boldsymbol{v} \oplus x_4x_2 \oplus x_3x_1$. Так как $x_4x_2 \oplus x_3x_1 = (0000\,0101\,0011\,0110)$, то легко видеть, что

$$\boldsymbol{v}' = \boldsymbol{v} \oplus x_1x_4 \oplus x_2x_3 = (1100\,1110\,0011\,0011).$$

Теперь, используя найденный вектор $\boldsymbol{v}'$, найдем коэффициенты при одночленах первой степени. Сначала определим проверочные суммы для x_4. Фиксируя второй, третий и четвертый разряды находим множества наборов B_i:

$$B_0 = \{(0000), (0001)\}, \qquad B_1 = \{(0010), (0011)\},$$
$$B_2 = \{(0100), (0101)\}, \qquad B_3 = \{(0110), (0111)\},$$
$$B_4 = \{(1000), (1001)\}, \qquad B_5 = \{(1010), (1011)\},$$
$$B_6 = \{(1100), (1101)\}, \qquad B_7 = \{(1110), (1111)\},$$

и множества B'_i номеров наборов этих множеств:

$$B'_0 = \{0, 1\}, \quad B'_1 = \{2, 3\}, \quad B'_2 = \{4, 5\}, \quad B'_3 = \{6, 7\},$$
$$B'_4 = \{8, 9\}, \quad B'_5 = \{10, 11\}, \quad B'_6 = \{12, 13\}, \quad B'_7 = \{14, 15\}.$$

Для соответствующих этим множествам проверочных сумм имеем:

$$S_0 = v'_0 \oplus v'_1 = 0, \quad S_1 = v'_2 \oplus v'_3 = 0, \quad S_2 = v'_4 \oplus v'_5 = 0, \quad S_3 = v'_6 \oplus v'_7 = 1,$$
$$S_4 = v'_8 \oplus v'_9 = 0, \quad S_5 = v'_{10} \oplus v'_{11} = 0, \quad S_6 = v'_{12} \oplus v'_{13} = 0, \quad S_7 = v'_{14} \oplus v'_{15} = 0.$$

Семь из восьми сумм равны нулю, следовательно, коэффициент при x_4 равен нулю. Проводя аналогичные вычисления для остальных одночленов первой степени, находим проверочные суммы для x_3:

$$S_0 = v'_0 \oplus v'_2 = 1, \quad S_1 = v'_1 \oplus v'_3 = 1, \quad S_2 = v'_4 \oplus v'_6 = 0, \quad S_3 = v'_5 \oplus v'_7 = 1,$$
$$S_4 = v'_8 \oplus v'_{10} = 1, \quad S_5 = v'_9 \oplus v'_{11} = 1, \quad S_6 = v'_{12} \oplus v'_{14} = 1, \quad S_7 = v'_{13} \oplus v'_{15} = 1;$$

для x_2:

$$S_0 = v'_0 \oplus v'_4 = 0, \quad S_1 = v'_1 \oplus v'_5 = 0, \quad S_2 = v'_2 \oplus v'_6 = 1, \quad S_3 = v'_3 \oplus v'_7 = 0,$$
$$S_4 = v'_8 \oplus v'_{12} = 0, \quad S_5 = v'_9 \oplus v'_{13} = 0, \quad S_6 = v'_{10} \oplus v'_{14} = 0, \quad S_7 = v'_1 \oplus v'_{15} = 0;$$

для x_1:

$$S_0 = v'_0 \oplus v'_8 = 1, \quad S_1 = v'_1 \oplus v'_9 = 1, \quad S_2 = v'_2 \oplus v'_{10} = 1, \quad S_3 = v'_3 \oplus v'_{11} = 1,$$
$$S_4 = v'_4 \oplus v'_{12} = 0, \quad S_5 = v'_5 \oplus v'_{13} = 1, \quad S_6 = v'_6 \oplus v'_{14} = 0, \quad S_7 = v'_7 \oplus v'_{15} = 1.$$

Таким образом, коэффициенты при одночленах x_4 и x_2 равны нулю, а коэффициенты при одночленах x_3 и x_1 равны единице. Теперь найдем сумму $\boldsymbol{v}'' = (\boldsymbol{v} \oplus x_4x_2 \oplus x_3x_1) \oplus x_3 \oplus x_1$. Так как $x_3 \oplus x_1 = (0011\,0011\,1100\,1100)$, то легко видеть, что

$$\boldsymbol{v}'' = (\boldsymbol{v} \oplus x_4x_2 \oplus x_3x_1) \oplus x_3 \oplus x_1 = (1111\,1101\,1111\,1111).$$

Для свободного члена каждый разряд является отдельной проверочной суммой. Поэтому из последнего равенства видно, что единица входит в разложение вектора $\boldsymbol{g}$. Следовательно, $\boldsymbol{c} = \boldsymbol{v}'' \oplus 1$, $\boldsymbol{g} = \boldsymbol{v} \oplus \boldsymbol{c}$, и, таким образом,

$$\boldsymbol{c} = \boldsymbol{v} \oplus x_4x_2 \oplus x_3x_1 \oplus x_3 \oplus x_1 \oplus 1 = (0000\,0010\,0000\,0000),$$
$$\boldsymbol{g} = x_4x_2 \oplus x_3x_1 \oplus x_3 \oplus x_1 \oplus 1 = (1100\,1001\,0000\,0101).$$

□

Задачи

3.5.1. Пусть $\boldsymbol{g} \in RM(4,2)$, $\boldsymbol{c} \in \mathbb{B}^{16}$ и $\|\boldsymbol{c}\| \leq 1$. Найти $\boldsymbol{g}$ и $\boldsymbol{c}$ если:
a) $\boldsymbol{g} \oplus \boldsymbol{c} = (1100\,1001\,1101\,1010)$; b) $\boldsymbol{g} \oplus \boldsymbol{c} = (0111\,1001\,1111\,1010)$.

3.5.2. Построить проверочную и порождающую матрицы кода $RM(n,k)$ если:
a) $n = 4$, $k = 2$; b) $n = 5$, $k = 3$; c) $n = 5$, $k = 2$; d) n и k — произвольные целые.

3.5.3. Показать, что любой код длины $n = 2k$ с минимальным расстоянием k содержит не более $2n$ элементов.

Глава 4.

Сложность вычисления булевых функций

Любое вычисление можно представить в виде последовательности шагов, каждый из которых состоит в выполнении некоторого простого действия над исходными данными или над величинами, полученными на предыдущих шагах. Список команд, описывающих эти шаги и определяющих порядок их выполнения, обычно называется программой или схемой вычисления. Как правило, программы состоят из команд двух видов: вычислительных и управляющих. Вычислительные команды производят некоторые безусловные действия, например, складывают числа. Управляющие команды определяют порядок выполнения вычислительных команд. К таким командам, в частности, относятся команды условного перехода и команда условной остановки вычислений. Число шагов, выполняемых в процессе вычисления, называется его сложностью. Если один и тот же объект, например булева функция, может быть вычислен различными способами, то его сложностью называется сложность самого простого вычисления.

Вычисления, программы которых состоят только из вычислительных команд или из вычислительных команд и команд остановки, называются неветвящимися. В любом неветвящемся вычислении команды выполняются последовательно одна за другой в том порядке, в котором они расположены в программе. Команды условной остановки могут прервать вычисления, но изменить порядок выполнения команд не могут.

Ниже рассматривается сложность вычисления булевых функций. Булевы функции вычисляются при помощи схем из функциональных элементов. Эти схемы представляют собой наиболее общую математическую модель неветвящихся вычислений, на каждом шаге которых выполняются только безусловные команды[1)]. Для подавляющего большинства существующих электронных схем именно схемы из функциональных элементов являются наиболее адекватной математической моделью.

В настоящей главе определяются схемы из функциональных элементов и рассматриваются некоторые свойства этих схем. Изучается сложность вычисления ряда простых булевых функций, в том числе функций, зависящих от небольшого числа переменных.

4.1. Схемы из функциональных элементов

1. Пусть B — подмножество множества булевых функций, зависящих не более чем от двух переменных.

Схемой из функциональных элементов (или *булевой схемой*) с n входами и m выходами называется ориентированный ациклический граф S, обладающий следующими свойствами:

(1) S содержит n вершин с входной степенью равной нулю. Такие вершины называются входами схемы.

(2) Входные степени остальных вершин не превосходят двух. Эти вершины называются элементами схемы. Ребра, входящие в один элемент, нумеруются числами от 1 до 2.

[1)] В седьмой главе будут рассмотрены неветвящиеся вычисления с условной остановкой

Каждому элементу схемы с входной степенью равной j ($j = 0, 1, 2$) приписана j-местная функция из B. Множество B называется базисом схемы.

(3) m вершин помечены целыми числами от 1 до m. Эти вершины называются выходами схемы.

Если вершины w и u схемы S связаны ребром, ориентированным от u к w, то будем говорить, что вершина w подключена к вершине u. Вершину u будем называть предком вершины w, а вершину w — потомком вершины u. Если вершина w имеет двух предков и номер ребра (uw) равен единице, то вершину u будем называть первым предком вершины w, если номер (uw) равен двойке — вторым предком этой вершины.

Как во всяком ориентированном ациклическом графе, на множестве вершин любой схемы существует естественный частичный порядок, определяемый ориентацией ребер. Будем говорить, что вершина u находится в схеме S *выше* вершины v, а вершина v *ниже* вершины u, если в S существует ориентированный путь, начинающийся в вершине u и заканчивающийся в вершине v. В частности, предок всегда находится в схеме выше любого своего потомка.

Если вершине u приписана функция h, то будем говорить, что вершина u *реализует* функцию h.

Пусть $X_n = \{x_1, \ldots, x_k\}$ — множество независимых переменных, $P = \{p_1, \ldots, p_n\}$ — множество булевых функций, зависящих от переменных из X_n. Будем говорить, что входы схемы S подключены к функциям из множества P, если каждому входу схемы S приписана некоторая функция p_i из P, причем разным вершинам приписаны разные функции.

Граф традиционно изображается на плоскости в виде множества точек, соответствующих вершинам, и соединяющих их линий, соответствующих ребрам. Стрелки на ребрах указывают их ориентацию. Именно так в левой части рисунка 4.1.1 изображен граф, являющийся схемой из функциональных элементов с двумя входами, подключенными к функциям x и y, и пятью элементами, реализующими функции $\&$, $\oplus$, $\to$, $\vee$ и $\neg$. Четвертый и пятый элементы являются выходами схемы, поэтому каждый из этих элементов помечен двумя символами — символом реализуемой функции и, через запятую, номером выхода. В правой части рисунка 4.1.1 изображена та же самая схема. При ее изображении были использованы следующие правила. 1) Предок всегда располагается выше потомка, т.е. все ребра ориентированы сверху вниз. Поэтому стрелки, указывающие ориентацию ребер, отсутствуют. 2) Элементы схем изображены треугольниками, в середине каждого треугольника помещен символ функции, реализуемой элементом. Если входная степень элемента, изображенного треугольником, равна двум, то ребра, связывающие элемент с его предками, присоединяются к разным точкам одной стороны треугольника — точка присоединения ребра от первого предка располагается левее точки присоединения ребра от второго предка. При таком изображении элементов отпадает необходимость в расстановке номеров ребер — нумерация однозначно восстанавливается по изображению. 3) Каждая вершина, являющаяся выходом схемы, отмечается ребром, выходящим из этой вершины, и не входящим ни в какую другую вершину.

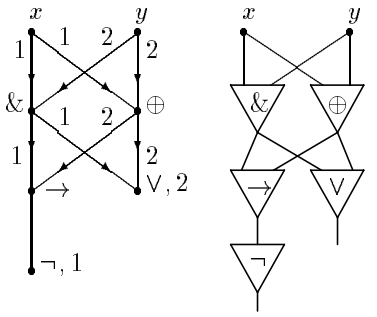


Рис. 4.1.1

Далее при изображении схем, как правило, будем придерживаться перечисленных выше правил.

Пусть входы схемы S подключены к функциям из P. Для каждой вершины w этой схемы определим функцию $S(\mathsf{w})$, *вычисляемую* в вершине w. Сделаем это индуктивно.

(1) Если w — вход схемы, которому приписана функция p_i, то $S(\mathsf{w}) = p_i$.

(2) Если вершине w приписана нульместная функция f, то $S(\mathsf{w}) = f$.

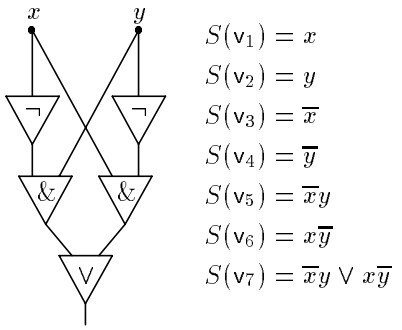
(3) Если вершине w приписана одноместная функция f и в вершине u — предке вершины w, вычисляется функция $S(\mathsf{u})$, то $S(\mathsf{w}) = f(S(\mathsf{u}))$.

(4) Если вершине w приписана двухместная функция f и в вершине u — первом предке

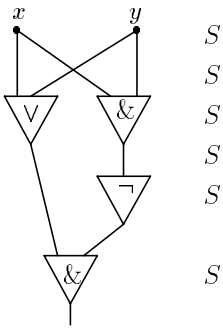
вершины w, вычисляется функция $S(\mathsf{u})$, а в вершине v — втором предке вершины w, вычисляется функция $S(\mathsf{v})$, то $S(\mathsf{w}) = f(S(\mathsf{u}), S(\mathsf{v}))$.

Будем говорить, что схема S вычисляет систему булевых функций $f = \{f_i\}_{i=1}^m$, зависящих от функций $p_1, \ldots, p_n$, если в схеме S входы подключены к функциям $p_1, \ldots, p_n$, а в выходах u_i $(i = 1, \ldots, m)$ вычисляются функции f_i, т.е. $S(\mathsf{u}_i) = f_i$. Изображая схемы, иногда рядом с выходом схемы вместо его номера будем указывать вычисляемую в этом выходе функцию.

Пример 4.1.1. На рисунке 4.1.2 изображены две схемы S_1 и S_2, вычисляющие одну и ту же функцию $x \oplus y$ в базисе $\{\&, \vee, \neg\}$. Вершины обоих схем перенумерованы слева направо–сверху вниз целыми числами начиная с единицы. Левая схема S_1 состоит из семи вершин



$S(\mathsf{v}_1) = x$
$S(\mathsf{v}_2) = y$
$S(\mathsf{v}_3) = \overline{x}$
$S(\mathsf{v}_4) = \overline{y}$
$S(\mathsf{v}_5) = \overline{x}y$
$S(\mathsf{v}_6) = x\overline{y}$
$S(\mathsf{v}_7) = \overline{x}y \vee x\overline{y}$



$S(\mathsf{v}_1) = x$
$S(\mathsf{v}_2) = y$
$S(\mathsf{v}_3) = x \vee y$
$S(\mathsf{v}_4) = x\&y$
$S(\mathsf{v}_5) = \overline{x\&y} =$
$= \overline{x} \vee \overline{y}$
$S(\mathsf{v}_6) = (x \vee y)(\overline{x} \vee \overline{y})$

Рис. 4.1.2

$\mathsf{v}_1, \ldots, \mathsf{v}_7$. Среди этих вершин два входа x и y и пять элементов. Символы функций, реализуемых элементами, указаны в соответствующих треугольниках. Функции, вычисляемые в вершинах схемы, перечислены в стоящем рядом столбце. В первых двух вершинах v_1 и v_2 вычисляются переменные x и y, в последней седьмой вершине v_7 — функция $x \oplus y$. Вершина v_7 является единственным выходом схемы. Правая схема S_2 состоит из шести вершин — двух входов x и y и четырех элементов. Вычисляемые в вершинах этой схемы функции так же перечислены в стоящем справа от схемы столбце. □

Две схемы S_1 и S_2 над множеством переменных $x_1, \ldots, x_n$ называются *эквивалентными*, если они вычисляют одинаковые функции (системы функций). Схемы на рисунке 4.1.2 — эквивалентные.

2. Важнейшими характеристиками любой схемы являются ее сложность и глубина. *Сложностью* $L(\mathsf{S})$ схемы S называется число элементов этой схемы. Для определения глубины схемы S рассмотрим различные ориентированные цепи, связывающие ее входы и выходы. Длиной цепи называется число элементов, через которые эта проходит цепь. Цепь, проходящую через максимальное число элементов, назовем максимальной цепью схемы S, а ее длину назовем *глубиной* $D(\mathsf{S})$ схемы S.

Особое значение имеет случай, когда входы схемы подключены к независимым переменным, т.е. когда $P = X_n$. В следующих определениях рассматриваются именно такие схемы.

Если среди всех схем, имеющих базис B и вычисляющих систему f, схема S содержит наименьшее число элементов, то S называется минимальной (по сложности) схемой системы f. Число элементов в минимальной (по сложности) схеме системы f называется *сложностью системы функций* f и обозначается через $L_B(f)$.

Аналогичным образом для произвольной системы булевых функций определяются ее минимальная (по глубине) схема и ее глубина. Если среди всех схем, имеющих базис B и вычисляющих систему f, схема S имеет наименьшую глубину D, то S называется минимальной. Глубина минимальной (по глубине) схемы системы f называется *глубиной системы функций* f и обозначается символом $D_B(f)$. Рассматривая сложность функций в базисе $P_2(2)$, будем, как правило, опускать индекс, указывающий базис схемы, т.е. $L_{P_2(2)}(f) = L(f)$.

Далее под минимальными схемами будем, как правило, понимать схемы, минимальные по сложности.

Пример 4.1.2. Снова рассмотрим изображенные рисунке 4.1.2 схемы, вычисляющие функцию $x \oplus y$ в базисе $\{\&, \vee, \neg\}$. Сложность левой схемы S_1 равна пяти, сложность правой схемы S_2 — четырем. Глубина обоих схем равна трем. В левой схеме есть две цепи длины три. Одна из них начинается во входе x и проходит через левый элемент отрицания, левый элемент дизъюнкции и левый элемент конъюнкции[2)]. В правой схеме цепь длины три начинается во входе y и проходит через два конъюнктора и элемент отрицания. Очевидно, что схема S_1 не является минимальной схемой для функции $x \oplus y$ среди схем с базисом $\{\&, \vee, \neg\}$ — схема S_2 имеет такой же базис, вычисляет такую же функцию что и S_1 и при этом состоит из меньшего числа элементов. □

3. При построении больших схем, вычисляющих сложные функции, часто бывает удобным сначала построить схемы вычисляющие некоторые вспомогательные функции, а потом из этих схем собрать требуемую схему S. Такие вспомогательные схемы будем называть подсхемами схемы S. Подсхемы с более чем одним выходом будем изображать прямоугольниками, а подсхемы с одним выходом либо треугольниками большого размера, либо четырехугольником с верхними прямыми и нижними закругленными углами. Внутрь фигуры изображающей подсхему будем помещать либо символ функции, вычисляемой подсхемой, либо символ–имя подсхемы. Рассмотрим два примера.

Пример 4.1.3. Рассмотрим две схемы в базисе $\{\&, \vee, \neg\}$: вычисляющую систему функций $\{x \oplus y, x \oplus y \oplus 1\}$ схему A_2, и вычисляющую систему функций $\{x \oplus y \oplus z, x \oplus y \oplus z \oplus 1\}$ схему A_3. Изображенная в левой части рисунка 4.1.3 схема A_2 состоит из вычисляющей сумму $x \oplus y$ подсхемы A, и подключенного к выходу этой подсхемы элемента отрицания, вычисляющего функцию $x \oplus y \oplus 1$. В схеме A_2 в качестве подсхемы A используется рассмотренная в примере 4.1.1 схема S_2. Сложность схемы A_2 равна пяти, а глубина — четырем. В правой части рисунка 4.1.3 изображена схема A_3. Эта схема состоит из трех подсхем. Верхняя подсхема является схемой A_2, подключенной к входам x и y. Две нижние подсхемы являются экземплярами схемы S_2 из примера 4.1.1. Левая подсхема подключена к первому выходу подсхемы A_2 и входу z, правая — ко второму выходу A_2 и входу z. Легко видеть, что сложность схемы A_3 равна 13, а глубина — семи. В схеме A_3 есть две цепи длины семь связывающих ее второй выход с входами x и y. □

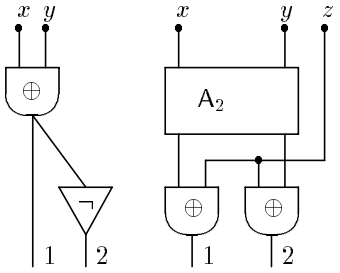


Рис. 4.1.3

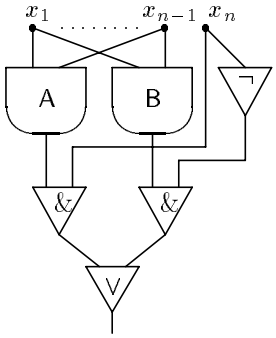


Рис. 4.1.4

Пример 4.1.4. Покажем, что любая булева функция n переменных может быть вычислена такой схемой S в базисе $\{\vee, \&, \neg\}$, что

$$L(\mathsf{S}) \le 3 \cdot 2^n - 4, \quad D(\mathsf{S}) \le 2n. \tag{4.1.1}$$

Сделаем это индукцией по числу переменных n. В основание индукции положим схемы, вычисляющие функции одной переменной. Самыми сложными и глубокими будут вычисляющие константы 0 и 1 схемы S_0 и S_1. Так как $0 = x\&\overline{x}$ и $1 = x \vee \overline{x}$, то легко видеть, что $L(\mathsf{S}_i) = 2$ и $D(\mathsf{S}_i) = 2$, т. е. неравенства (4.1.1) справедливы при $n = 1$. Теперь допустим, что эти неравенства справедливы при всех n не превосходящих некоторое целое $k \ge 1$. Покажем, что тогда неравенства (4.1.1) имеют место и при $n = k + 1$. Для этого построим

[2)] Часто элемент дизъюнкции называется дизъюнктором, элемент конъюнкции — конъюнктором, элемент отрицания — инвертором.

схему, вычисляющую произвольную булеву функцию $f(x_1, \ldots, x_n)$, и оценим ее сложность и глубину. Функцию f разложим по последней переменной:

$$f(x_1, \ldots, x_n) = \overline{x}_n f(x_1, \ldots, x_{n-1}, 0) \vee x_n f(x_1, \ldots, x_{n-1}, 1).$$

В соответствием с этим разложением построена изображенная на рисунке 4.1.4 схема S. Эта схема вычисляет функцию f и состоит из двух подсхем A и B, одного элемента отрицания, двух конъюнкторов и одного дизъюнктора. Подсхема A вычисляет функцию $f_0 = f(x_1, \ldots, x_{n-1}, 0)$, подсхема B — функцию $f_1 = f(x_1, \ldots, x_{n-1}, 1)$, элемент отрицания вычисляет $\overline{x}_n$. Далее два конъюнктора умножают функции f_0 и f_1, соответственно на $\overline{x}_n$ и x_n. Затем дизъюнктор вычисляет дизъюнкцию двух произведений. Легко видеть, что сложность и глубина схемы S выражаются через сложности и глубины подсхем A и B следующим образом

$$L(\mathsf{S}) = L(\mathsf{A}) + L(\mathsf{B}) + 4, \quad D(\mathsf{S}) = \max(D(\mathsf{A}), D(\mathsf{B})) + 2.$$

По предположению индукции каждая из подсхем A и B состоит не более чем из $3 \cdot 2^{n-1} - 4$ элементов, а их глубина не превосходит $2(n-1)$. Поэтому,

$$L(\mathsf{S}) \leq (3 \cdot 2^{n-1} - 4) + (3 \cdot 2^{n-1} - 4) + 4 = 3 \cdot 2^n - 4,$$
$$D(\mathsf{S}) \leq 2(n-1) + 2 = 2n.$$

Неравенства (4.1.1) доказаны. □

4. Оценим сверху число различных булевых схем над множеством переменных $x_1, \ldots, x_n$ в базисе B, при условии, что сложность рассматриваемых схем не превосходит заданной величины. Имеет место следующий результат.

Теорема 4.1.1. *Пусть $B \subseteq P_2(2)$, $N(L, n)$ — число неэквивалентных схем в базисе B над множеством переменных $x_1, \ldots, x_n$ сложности не более L. Тогда*

$$N(L, n) \leq (c(L + n))^{L+1},$$

где c — константа, зависящая от базиса.

Доказательство. Рассмотрим произвольную схему S, состоящую из n входов и L элементов. Элементы схемы пометим произвольным образом различными символами. Преобразуем S, применив к каждой вершине, из которой выходит более одного ребра, следующее преобразование. Если из вершины v выходит k ребер $(\mathsf{v}, \mathsf{u}_i), \ldots, (\mathsf{v}, \mathsf{u}_k)$, входящих в вершины $\mathsf{u}_1, \ldots, \mathsf{u}_k$, то:

1. Удалим ребра $(\mathsf{v}, \mathsf{u}_2), \ldots, (\mathsf{v}, \mathsf{u}_k)$;
2. Добавим $k-1$ новых вершин $\mathsf{v}_2, \ldots, \mathsf{v}_k$, приписав каждой из них символ v;
3. Каждую новую вершину v_p, $p = 2, 3, \ldots, k$, свяжем ребром $(\mathsf{v}_p, \mathsf{u}_p)$ с вершиной u_p.

Отметим, что данное преобразование не добавляет в схему внутренние вершины. Поэтому легко видеть, что преобразованная схема будет деревом, с L внутренними вершинами. Висячие вершины этого дерева будут помечены не более чем $L + n$ разными символами (не более L символов, соответствующих элементам S, и n символов переменных). Получившееся дерево будет представлять некоторую формулу в двухместном базисе над множеством из $L + n$ переменных (см. стр. 16). При этом из схем, вычисляющих разные функции, будут получаться разные формулы. Следовательно, число неэквивалентных схем в базисе B над множеством из n переменных, каждая из которых состоит не более чем из L элементов, не превосходит числа различных формул в том же двухместном базисе над множеством из $L+n$ переменных, при этом сложность каждой формулы не больше L. Таким образом, для оценки количества схем можно воспользоваться теоремой 1.3.2. Из этой теоремы следует, что

$$N(L, n) \leq N(L, L + n, 2) \leq (c(L + n))^{L+1}.$$

Теорема доказана.

Как и в случае формул (стр. 18, теорема 1.3.3), доказанная теорема позволяет установить, что большинство булевых функций можно вычислить только при помощи очень сложных схем. Имеет место следующий результат.

Теорема 4.1.2. *Пусть B — полный базис. Тогда для любой постоянной $\varepsilon > 0$ доля функций из $P_2(n)$ вычисляемых схемами, сложность которых не превосходит*

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

стремится к нулю при $n \to \infty$.

Доказательство. Так как неравные функции вычисляются разными схемами, то число функций, вычисляемых схемами сложности не более L, не больше чем количество таких схем. Поэтому для доказательства теоремы достаточно показать, что число $N(L,n)$ неэквивалентных схем сложности не более $L = \frac{(1-\varepsilon)2^n}{n}$ при возрастании n есть $o\bigl(2^{2^n}\bigr)$. Оценим логарифм $N(L,n)$. Используя оценку величины $N(L,n)$ из предыдущей теоремы, и учитывая, что $n \ll \frac{2^n}{n}$, при $n \to \infty$ имеем

$$\log_2 N(L,n) \lesssim \frac{(1-\varepsilon)2^n}{n} \cdot \log_2\bigl(c\frac{(1-\varepsilon)2^n}{n}\bigr) \le (1-\varepsilon)2^n.$$

Следовательно,

$$N(L,n) = 2^{\log_2 N(L,n)} = 2^{2^n(1-\varepsilon)} = o\Bigl(2^{2^n}\Bigr).$$

Теорема доказана.

Теорема 4.1.2 является частным случаем более общего результата, связывающего сложность вычисления функций, принадлежащих какому-либо множеству, и мощность этого множества. Этот результат приведем без доказательства, которое почти полностью аналогично доказательству теоремы 4.1.2.

Теорема 4.1.3. *Пусть B — полный базис, $R \subseteq P_2$, $R(n) = R \cap P_2(n)$. Тогда для любой постоянной $\varepsilon > 0$ доля функций из $R(n)$ вычисляемых схемами, сложность которых не превосходит*

$$\frac{(1-\varepsilon)\log_2 R(n)}{\log_2\log_2 R(n)},$$

стремится к нулю при $n \to \infty$.

Пример 4.1.5. Рассмотрим множество $Q_{n,2}$, состоящее из всех n-местных булевых функций, степень многочлена Жегалкина которых не превосходит двух. Очевидно, что в рассматриваемом множестве содержится ровно $2^{1+\binom{n}{1}+\binom{n}{2}}$ функций. Поэтому из теоремы 4.1.3 легко следует, что при $n \to \infty$ в $Q_{n,2}$ найдется функция f, для сложности $L(f)$ которой справедливо асимптотическое неравенство

$$L(f) \gtrsim \frac{1+\binom{n}{1}+\binom{n}{2}}{\log_2(1+\binom{n}{1}+\binom{n}{2})} \sim \frac{n^2}{4\log_2 n}.$$

Более того, это неравенство справедливо для почти всех функций из $Q_{n,2}$. □

Задачи

4.1.1. Показать, что для любой схемы S справедливо неравенство $D(\mathrm{S}) > \log_2 L(\mathrm{S})$.

4.1.2. Дать определение схемы из функциональных элементов в случае, когда базис схемы содержит более чем двухместные функции.

4.1.3. Построить схемы в базисе $\{\&, \oplus, 1\}$, которые вычисляют функции:
a) $x \vee y \vee z$; b) $(x \downarrow y) \vee (x \downarrow z)$; c) $(x \mid y) \vee (x \to z)$; d) $(x \to yz) \vee (z \to x) \vee y$.

4.1.4. Построить схемы в базисе $\{\&, \vee, \neg\}$, которые вычисляют функции:

a) $x \oplus y \oplus z$; b) $(x \oplus y) \vee (x \oplus z)$; c) $(x \mid y) \sim (x \downarrow z)$; d) $(x \oplus yz) \vee (z \to x)$.

4.1.5. Построить схемы в базисе $P_2(2)$, которые вычисляют системы функций:
a) $\{x \oplus y \oplus z, \tau_2(x,y,z)\}$; b) $\{x \oplus y \oplus z, x \oplus y \oplus z \oplus 1\}$; c) $\{x \vee y \vee z, x\&y\&z\}$.

4.1.6. Показать, что для любых полных базисов $B_1, B_2 \subseteq P_2(2)$ найдутся такие константы c_1 и c_2, что для любой функции f из $P_2(n)$ будут справедливы неравенства $L_{B_1}(f) \leq c_1 L_{B_2}(f)$ и $L_{B_2}(f) \leq c_2 L_{B_1}(f)$.

4.1.7. Оценить константы c_1 и c_2 из предыдущей задачи, если базисы B_1 и B_2 равны соответственно:
a) $\{\&, \vee, \neg\}$ и $\{\downarrow\}$; b) $\{\&, \oplus, 1\}$ и $\{\&, \neg\}$; c) $\{\to, \neg\}$ и $P_2(2)$; d) $\{\vee, \neg\}$ и $\{\mid\}$.

4.1.8. Показать, что число неэквивалентных схем, вычисляющих булевы (m,n)-операторы и состоящих не более чем из L элементов, не превосходит $(c(L+n))^{L+m}$, где c — константа.

4.1.9. Показать, что любая монотонная булева функция n переменных, отличная от тождественной константы, может быть вычислена такой схемой S в базисе $\{\vee, \&\}$, что $L(\mathrm{S}) \leq 2 \cdot 2^n - 2$ и $D(\mathrm{S}) \leq 2n$.

4.1.10. Доказать теорему 4.1.3.

4.1.11. Доказать аналоги теорем 4.1.2 и 4.1.3 для глубины функций.

4.1.12. Пусть $m = \mathcal{O}(2^n)$. Показать, что для любой постоянной $\varepsilon > 0$ доля булевых (m,n)-операторов вычисляемых схемами, сложность которых не превосходит $\frac{(1-\varepsilon)2^n m}{n + \log_2 m}$, стремится к нулю при $n \to \infty$.

4.1.13. Показать, что при $n \to \infty$ в $P_2(n)$ найдется функция f такая, что $\|f\| \leq 2^{n-1}$ и $L(f)$ асимптотически не меньше чем $\frac{2^n}{n}$.

4.2. Булевы функции трех переменных. Построение схем

Ниже рассматривается сложность вычисления булевых функций трех переменных схемами из функциональных элементов в трех базисах B_0, B_1 и B_2. Базис B_0 состоит из всех не более чем двухместных булевых функций, базис B_1 — из всех не более чем двухместных булевых функций, за исключением всех линейных функций, существенно зависящих от двух аргументов, базис B_2 — из всех не более чем двухместных линейных булевых функций и конъюнкции. В этом параграфе приведены конструкции схем, вычисляющих функции трех переменных, и, как следствие, получены верхние оценки сложности указанных функций. Нижние оценки, совпадающие с верхними, устанавливаются в двух следующих параграфах.

Утверждение 4.2.1. *Для сложности каждой булевой функции f, зависящей от трех переменных, справедливо неравенство*

$$L_{B_0}(f) \leq 4.$$

Доказательство. В многочлене Жегалкина произвольной булевой функции $f(x,y,z)$ соберем вместе все одночлены содержащие и все одночлены не содержащие переменную x. В результате получим следующее равенство:

$$\begin{aligned} f(x,y,z) &= a_0 \oplus a_1 x \oplus a_2 y \oplus a_3 z \oplus a_4 xy \oplus a_5 xz \oplus a_6 yz \oplus a_7 xyz = \\ &= x(a_1 \oplus a_4 y \oplus a_5 z \oplus a_7 yz) \oplus (a_0 \oplus a_2 y \oplus a_3 z \oplus a_6 yz) = \\ &= xh(y,z) \oplus g(y,z). \end{aligned}$$

Схема, построенная в соответствии с этим представлением, состоит из элемента реализующего функцию h, элемента реализующего функцию g, конъюнкции и элемента сложения. Утверждение доказано.

Утверждение 4.2.2. *Для сложности каждой булевой функции f, зависящей от трех переменных, справедливо неравенство*

$$L_{B_1}(f) \le 6.$$

Доказательство. Прежде всего покажем, что каждая булева функция двух переменных может быть вычислена в базисе B_1 схемой, состоящей не более чем из трех элементов. Так как каждая нелинейная функция входит в базис B_1, то в этом базисе сложность любой нелинейной функции равна единице. Поэтому достаточно показать, что сложности линейных функций $\oplus$ и $\sim$ не превосходят трех.

Положим $\varphi(x,y) = \overline{x}\,y$ и $\psi(x,y) = x \downarrow y = \overline{x}\,\overline{y}$. Легко видеть, что для линейных функций двух переменных справедливы равенства

$$x \oplus y = \overline{x}y \vee x\overline{y} = \varphi(x,y) \vee \varphi(y,x),$$
$$x \sim y = x\,y \vee \overline{x}\,\overline{y} = x\,y \vee \psi(x,y).$$

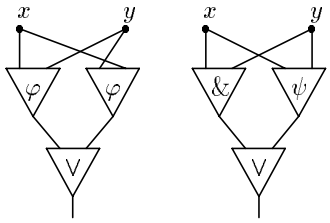


Рис. 4.2.1

На рисунке слева изображены схемы, построенные в соответствии с этими равенствами. Таким образом, если h — булева функция двух переменных, то ее сложность удовлетворяет неравенству

$$L_{B_1}(h) \le 3. \tag{4.2.1}$$

Произвольную булеву функцию трех переменных $f(x,y,z)$ разложим по первой переменной x:

$$f(x,y,z) = \overline{x}f(0,y,z) \vee xf(1,y,z) = \overline{x}h(y,z) \vee xg(y,z).$$

Рассмотрим два случая, первый, когда каждая из функций h и g существенно зависит от двух переменных, и второй, когда хотя бы одна из них существенно зависит не более чем от одной переменной. Первый случай распадается на следующие три:

(i) обе функции h и g нелинейные;

(ii) одна из этих функций нелинейная, вторая линейная;

(iii) обе функции линейные.

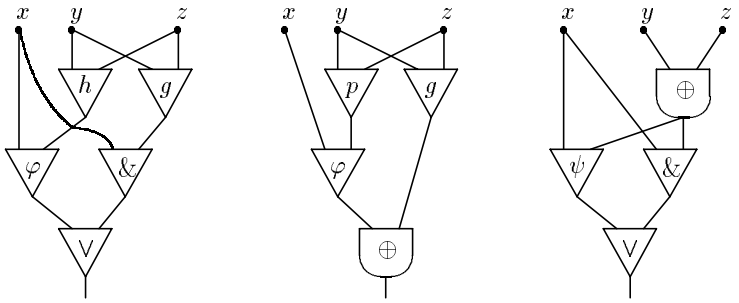


Рис. 4.2.2

(i) Схема, вычисляющая функцию f в рассматриваемом случае, изображена в левой части рисунка 4.2.2. Очевидно, что сложность этой схемы равна пяти.

(ii) Без ограничения общности будем полагать, что линейной является функция h, а нелинейной — g. Тогда

$$\overline{x}h \vee xg = \overline{x}hxg \oplus \overline{x}h \oplus xg = \overline{x}h \oplus xg =$$
$$= (x \oplus 1)h \oplus (x \oplus 1)g \oplus g = \overline{x}(g \oplus h) \oplus g.$$

Положим $p = g \oplus h$. Очевидно, что функция p будет нелинейной, и, следовательно, будет принадлежать базису B_1. Схема, вычисляющая f в соответствии с приведенной выше формулой, изображена в центре рисунка 4.2.2. Сложность этой схемы равна шести.

(*iii*) Если $g = h$, то в этом случае переменная z не является существенной для функция $f(x, y, z)$, и, поэтому, сложность функции f удовлетворяет неравенству (4.2.1). Если $g \neq h$, то тогда одна из этих функций равна $\oplus$, а вторая — $\sim$. В правой части рисунка 4.2.2 приведена схема, вычисляющая f в предположении $h = \oplus$. Сложность этой схемы равна шести.

Теперь рассмотрим второй случай. Без ограничения общности будем полагать, что функция h существенно зависит только от одной переменной y. Тогда для вычисления произведения $\overline{x}h(y, z)$ достаточно одного элемента, вычисляющего подходящую двухместную функцию ρ из базиса B_1. Из (4.2.1) следует, что сложность функции g не превосходит трех. Поэтому произведение $xg(y, z)$ можно вычислить схемой сложность которой не больше четырех. Схема, вычисляющая функцию f в рассматриваемом случае, изображена на рисунке 4.2.3. Легко видеть, что сложность этой схемы равна шести. Утверждение доказано.

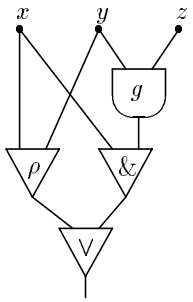


Рис. 4.2.3

Утверждение 4.2.3. *Для сложности каждой булевой функции f, зависящей от трех переменных, справедливо неравенство*

$$L_{B_2}(f) \leq 5.$$

Доказательство. Прежде всего покажем, что любая булева функция двух переменных может быть вычислена схемой, сложность которой не превосходит трех. Через $l(y, z)$ будем далее обозначать подходящую ненулевую линейную функцию двух переменных, т. е.

$$l(y, z) \in \{1, y, z, y \oplus 1, z \oplus 1, y \oplus z, y \oplus z \oplus 1\}.$$

Тогда для любой, отличной от конъюнкции нелинейной булевой функции $h(y, z)$ справедливо равенство

$$h(y, z) = yz \oplus ay \oplus bz \oplus c = yz \oplus l(y, z). \tag{4.2.2}$$

Поэтому такая функция h может быть вычислена схемой, состоящей из трех элементов: одной конъюнкции, одного элемента, реализующего функцию $l(y, z)$, и одного элемента $\oplus$, входы которого подключены к выходам первых двух элементов. Если в (4.2.2) хотя бы одна из постоянных a и b равна нулю, то для вычисления f достаточно двух элементов: сначала вычисляется конъюнкция yz, затем к ней прибавляется линейная часть.

Далее покажем, что любые две непостоянные булевы функции $h(y, z)$ и $g(y, z)$, одна из которых линейная, могут быть вычислены схемой, сложность которой не превосходит трех. Пусть $h(y, z)$ — линейная функция, $g(y, z)$ — нелинейная функция. Без ограничения общности, будем полагать, что h существенно зависит от y. Тогда

$$h(y, z) = \alpha \oplus y \oplus \alpha_1 z,$$
$$g(y, z) = \beta \oplus \beta_1 y \oplus \beta_2 z \oplus yz.$$

Если $h(y, z) = y$, то вычислять h не надо, а для вычисления g достаточно трех элементов.

Далее полагаем, что h отлична от тождественной переменной. Рассмотрим несколько случаев. В каждом рассматриваемом случае сначала вычисляем линейную функцию h. Для этого потребуется один элемент.

1. $\beta_1\beta_2 = 0$. В этом случае, как было показано выше, для вычисления g достаточно двух элементов.

2. $\beta_1\beta_2 = 1$, $\alpha_1 = 1$. В этом случае $g(y, z) = yz \oplus h(y, z) \oplus (\alpha \oplus \beta)$. Поэтому для вычисления g достаточно двух элементов — одного конъюнктора для вычисления yz, и одного линейного элемента, для сложения yz и $h(y, z)$. Если $(\alpha \oplus \beta) = 0$, то для сложения используется элемент $\oplus$, если $(\alpha \oplus \beta) = 1$ — элемент $\sim$.

3. $\beta_1\beta_2 = 1$, $\alpha_1 = 0$. В этом случае $h(y,z) = y \oplus 1$. Следовательно,

$$g(y,z) = \beta \oplus y \oplus z \oplus yz = \beta \oplus y \oplus z(y \oplus 1) = \beta \oplus y \oplus zh.$$

Как и в предыдущем случае, легко видеть, что для вычисления g достаточно двух элементов — одного конъюнктора для вычисления zh, и одного линейного элемента.

Таким образом, при всех возможных значениях постоянных α, α_1 и β, β_1, β_2 для вычисления пары функций h и g достаточно трех элементов.

Разложим функцию $f(x,y,z)$ по переменной x, собрав в многочлене Жегалкина этой функции вместе одночлены содержащие и не содержащие x:

$$\begin{aligned} f(x,y,z) &= a_0 \oplus a_1x \oplus a_2y \oplus a_3z \oplus a_4xy \oplus a_5xz \oplus a_6yz \oplus a_7xyz = \\ &= x(a_1 \oplus a_4y \oplus a_5z \oplus a_7yz) \oplus (a_0 \oplus a_2y \oplus a_3z \oplus a_6yz) = \\ &= xh_1(y,z) \oplus g_1(y,z). \end{aligned} \tag{4.2.3}$$

Легко видеть, что если $a_6a_7 = 0$, то одна из функций h_1 или g_1 будет линейной. Если $a_5a_7 = 0$, то раскладывая f по переменной y, получим разложение $f = yh_2(x,z) \oplus g_2(x,z)$, в котором одна из функций h_2 или g_2 будет линейной. Точно также, если $a_4a_7 = 0$, то раскладывая f по переменной z, получим разложение $f = zh_3(x,y) \oplus g_3(x,y)$ в котором одна из функций h_3 или g_3 будет линейной.

Таким образом, если $a_4a_5a_6a_7 = 0$, то функцию f можно вычислить схемой, состоящей не более чем из пяти элементов. Для этого надо разложить f по подходящей переменной так, чтобы в разложении хотя бы одна из функций h или g была линейной. Тогда эти функции вычисляются схемой из трех элементов. Для окончательного вычисления f потребуется один элемент умножения и один элемент сложения.

Теперь рассмотрим случай когда $a_4a_5a_6a_7 = 1$. Из (4.2.3) следует, что

$$\begin{aligned} h_1(y,z) &= a_1 \oplus y \oplus z \oplus yz, \\ g_1(y,z) &= a_0 \oplus a_2y \oplus a_3z \oplus yz. \end{aligned}$$

Рассмотрим три случая.

1. $a_2a_3 = 1$. В этом случае $g_1(y,z) = h_1(y,z) \oplus (a_0 \oplus a_1)$. Поэтому,

$$f(x,y,z) = xh_1(y,z) \oplus h_1(y,z) \oplus (a_0 \oplus a_1).$$

Легко видеть, что для вычисления f достаточно пяти элементов — три элемента для вычисления h_1, один элемент для умножения h_1 и x, и один элемент для вычисления суммы xh_1 и $h_1(y,z) \oplus (a_0 \oplus a_1)$.

2. $a_2a_3 = 0$, $a_2 \vee a_3 = 1$. Без ограничения общности полагаем, что $a_2 = 1$, $a_3 = 0$. В этом случае

$$\begin{aligned} h_1(y,z) &= a_1 \oplus y \oplus z \oplus yz = a_0 \oplus a_1 \oplus z \oplus (a_0 \oplus y \oplus yz) \\ &= (a_0 \oplus a_1) \oplus z \oplus g_1(y,z) \end{aligned}$$

Для вычисления f снова достаточно пяти элементов. Два элемента потребуются для вычисления $g_1(y,z)$, один элемент для вычисления суммы g_1 и $(a_0 \oplus a_1) \oplus z$, т. е. для вычисления h_1, один элемент для умножения h_1 и x, и один элемент для сложения xh_1 и g_1.

3. $a_2 \vee a_3 = 0$. В этом случае

$$f(x,y,z) = x(a_1 \oplus z \oplus y \oplus yz) \oplus a_0 \oplus yz.$$

Для вычисления $h_1 = (a_1 \oplus z \oplus y \oplus yz)$ достаточно трех элементов. При этом будет также вычислена конъюнкция yz. Один элемент потребуется для умножения h_1 и x, и еще один для вычисления суммы xh_1 и $a_0 \oplus yz$. Утверждение доказано.

Задачи

4.2.1. Построить схему, вычисляющую в базисе $\{\&, \vee\}$ одновременно все непостоянные монотонные функции двух переменных.

4.2.2. Построить схему, вычисляющую в базисе $\{\oplus, 1\}$ одновременно все линейные функции трех переменных.

4.2.3. Построить минимальную схему, вычисляющую в базисе B одновременно все функции трех переменных, если:
a) $B = B_0$, b) $B = B_1$, c) $B = \{\vee, \&, \neg\}$, d) $B = \{\&, \oplus, 1\}$, e) $B = \{\downarrow\}$.

4.2.4. Для произвольной трехместной булевой функции f оценить сверху:
a) $L_{\{\&,\neg\}}(f)$; b) $L_{\{\vee,\neg\}}(f)$; c) $L_{\{\&,\vee,\neg\}}(f)$; d) $L_{\{\&,\oplus,1\}}(f)$; e) $L_{\{\downarrow\}}(f)$; f) $L_{\{|\}}(f)$.

4.2.5. Для произвольной трехместной булевой функции f оценить сверху:
a) $D_{B_0}(f)$; b) $D_{B_1}(f)$; c) $D_{B_2}(f)$; d) $D_{\{\&,\vee,\neg\}}(f)$; e) $D_{\{\&,\oplus,1\}}(f)$; f) $D_{\{|\}}(f)$.

4.3. Преобразования схем

Сформулируем и докажем несколько утверждений о свойствах минимальных схем в базисах B_0 и B_1. В дальнейшем эти утверждения будут использованы для установления нижних оценок сложности конкретных булевых функций, в частности рассмотренных в предыдущем параграфе функций трех переменных.

Утверждение 4.3.1. *Пусть $B \in \{B_0, B_1\}$, S — схема в базисе B. Если в S есть вершина, в которой вычисляется тождественная константа, то эта вершина может быть удалена из схемы, а сама схема S преобразована таким образом, что ее сложность не увеличится, а вычисляемая ею функция не изменится.*

Доказательство. Пусть S — удовлетворяющая условиям леммы схема. Допустим, что в схеме S есть вершины, в которых вычисляется константы. Пусть s — самая нижняя такая вершина, и пусть в этой вершине вычисляется постоянная α. Рассмотрим фрагмент схемы содержащий s, элементы $\mathsf{s}_1, \ldots, \mathsf{s}_p$, — потомки вершины s, и вершины $\mathsf{u}_1, \ldots, \mathsf{u}_p$, — предки элементов $\mathsf{s}_1, \ldots, \mathsf{s}_p$. Функции, реализуемые элементами $\mathsf{s}_1, \ldots, \mathsf{s}_p$ будем обозначать через $v_1, \ldots, v_p$. Заметим, что в силу выбора вершины s, среди элементов $\mathsf{s}_1, \ldots, \mathsf{s}_p$ нет элементов, реализующих одноместные функции. Рассматриваемый фрагмент изображен в левой части рисунка 4.3.1. Преобразуем этот фрагмент следующим образом. Удалим элемент s, а элементы $\mathsf{s}_1, \ldots, \mathsf{s}_p$ заменим элементами $\mathsf{s}'_1, \ldots, \mathsf{s}'_p$, реализующими такие одноместные функции $v'_1, \ldots, v'_p$, что $v'_i(x) = v_i(\alpha, S(\mathsf{u}_i))$ (здесь полагаем, что в схеме S вершина s является первым предком элемента s_i). Преобразованный фрагмент показан в правой части рисунка 4.3.1. Легко видеть, что в элементах $\mathsf{s}'_1, \ldots, \mathsf{s}'_p$ вычисляются те же функции, что и ранее в элементах $\mathsf{s}_1, \ldots, \mathsf{s}_p$. Утверждение доказано.

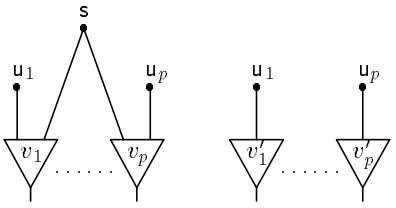


Рис. 4.3.1

Утверждение 4.3.1 используется в часто применяемом преобразовании схем — подстановке константы вместо какого-либо входа. Рассмотрим это преобразование на примере схемы, вычисляющей функцию $x \oplus y$. Эта схема изображена в левой части рисунка 4.3.2. В схеме вместо переменной y подставим единицу. После такой подстановки правый элемент отрицания превратится в элемент, реализующий тождественный нуль, а левый конъюнктор — в тождественный элемент, который вычисляет такую же функцию, как и левый инвертор, и поэтому может быть удален из схемы. Преобразованная схема изображена справа от исходной схемы. В новой схеме первый вход дизъюнктора подключен к инвертору вместо удаленного конъюнктора. Второй вход оставшегося конъюнктора подключен

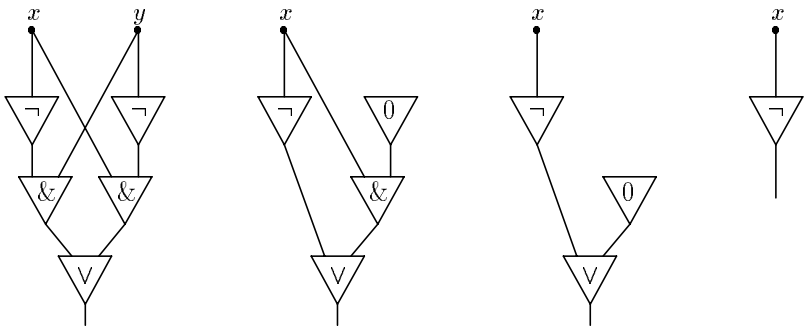


Рис. 4.3.2

к тождественному нулю, и, следовательно, сам вычисляет тождественный нуль. Поэтому во второй схеме удалим элемент, вычисляющий тождественный нуль, а конъюнктор заменим элементом, вычисляющим тождественный нуль. В результате получится третья слева схема рисунка 4.3.2. В этой схеме второй вход дизъюнктора подключен к тождественному нулю, и, следовательно, вычисляет функцию, к которой подключен его первый вход. Поэтому конъюнктор вместе с тождественным нулем можно удалить из схемы, а элемент отрицания объявить выходом схемы. Получившаяся схема изображена в правой части рисунка 4.3.2. Эта схема вычисляет отрицание переменной x.

Утверждение 4.3.2. *Пусть $B \in \{B_0, B_1\}$, S — схема в базисе B, вычисляющая функцию существенно зависящую не менее чем от двух переменных. Если в S есть вершина, в которой реализуется отрицание, то эта вершина может быть удалена из схемы, а сама схема S преобразована таким образом, что ее сложность не увеличится, а вычисляемая ею функция не изменится.*

Доказательство. Пусть S — удовлетворяющая условиям леммы схема. Допустим, что в этой схеме есть элементы, в которых реализуется отрицание. Пусть s — самый нижний такой элемент и s не является выходом схемы. Рассмотрим фрагмент схемы, содержащий s, элементы $\mathsf{s}_1, \ldots, \mathsf{s}_p$, — потомки вершины s, вершину s_0 — предка s, и вершины $\mathsf{u}_1, \ldots, \mathsf{u}_p$, — предки элементов $\mathsf{s}_1, \ldots, \mathsf{s}_p$. Рассматриваемый фрагмент изображен в левой части рисунка 4.3.3. Преобразуем его следующим образом. Удалим элемент s. Элементы $\mathsf{s}_1, \ldots, \mathsf{s}_p$, реализующие функции $v_1, \ldots, v_p$, подключим к вершине s_0, после чего заменим элементами $\mathsf{s}'_1, \ldots, \mathsf{s}'_p$, реализующими такие функции $v'_1, \ldots, v'_p$, что $v'_i(x, y) = v_i(\overline{x}, y)$ (здесь как и ранее полагаем, что в схеме S вершина s_0 является первым предком элемента s_i). Преобразованный фрагмент схемы S изображен в правой части рис. 4.3.3. Легко видеть, что в элементах $\mathsf{s}'_1, \ldots, \mathsf{s}'_p$ вычисляются те же функции, что и ранее в элементах $\mathsf{s}_1, \ldots, \mathsf{s}_p$.

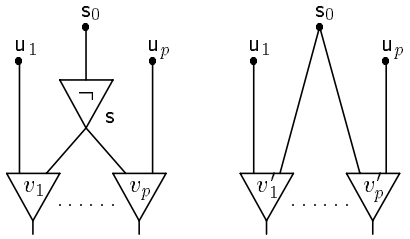


Рис. 4.3.3

Теперь рассмотрим случай, когда элемент s является выходом схемы. Так как s является единственным выходом схемы, то очевидно, что s — последний элемент этой схемы (в противном случае просто удалим все элементы расположенные ниже s). Пусть s_0 — вершина к которой подключена вершина s. Так как схема S вычисляет функцию, существенно зависящую не менее чем от двух переменных, то вершина s_0 обязательно будет функциональным элементом. Реализуемую элементом s_0 функцию обозначим через v_0. В рассматриваемом фрагменте удалим элемент s, элемент s_0 заменим элементом s'_0, реализующим функцию $\overline{v}_0$, и объявим элемент s'_0 выходом схемы.

Утверждение доказано.

Наконец без доказательства приведем следующее очевидное утверждение.

Утверждение 4.3.3. *Если в схеме* S *есть две вершины, в которых вычисляется одна и та же функция, то одна из этих вершин может быть удалена из схемы, а сама схема* S *преобразована таким образом, что ее сложность не увеличится, вычисляемая ею функция не изменится.*

Пример 4.3.1. Последовательное применение приведенных выше утверждений показано на рисунке 4.3.4. Исходная схема в базисе B_0 изображена слева. Вершина, в которой вы-

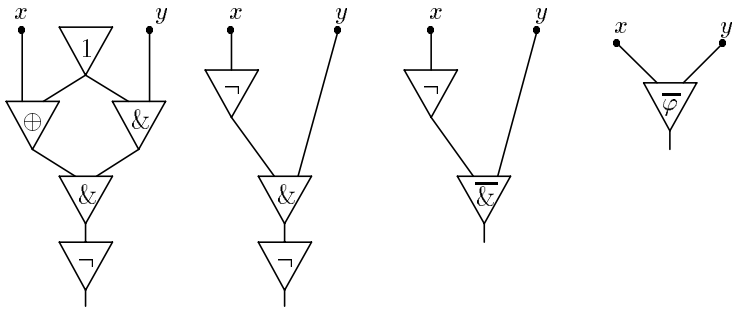


Рис. 4.3.4

числяется тождественная единица, удаляется в силу утверждения 4.3.1. При этом элемент сложения заменяется элементом отрицания, а верхняя конъюнкция — тождественным элементом, который в свою очередь удаляется в силу утверждения 4.3.3. Преобразованная схема изображена на втором слева фрагменте рисунка. Затем, в силу утверждения 4.3.2, удаляется нижний элемент отрицания и конъюнкция заменяется элементом, реализующим $\overline{\&}$. Наконец, удаляется последний элемент отрицания, при этом элемент, реализующий $\overline{\&}$, заменяется элементом, реализующим функцию $\overline{\varphi}$. Результатом всех преобразований является одноэлементная схема, изображенная на правом фрагменте рисунка 4.3.4. □

Функция $f(x, y)$, существенно зависящая от переменных x и y, называется $\&$-*функцией*, если

$$f(x, y) = (x^{(\alpha)} \& y^{(\beta)})^{(\gamma)},$$

где α, β, γ — булевы постоянные. Функция $g(x, y)$, существенно зависящая от переменных x и y, называется $\oplus$-*функцией*, если

$$g(x, y) = x \oplus y \oplus \alpha,$$

где α — булева постоянная. Элементы, реализующие $\&$- и $\oplus$-функции, называются $\&$- и $\oplus$-элементами, соответственно.

Легко проверить, что различные α, β, γ определяют различные $\&$-функции, а различные α — различные $\oplus$-функции. Поэтому общее число различных $\&$- и $\oplus$-функций равно десяти. Так как среди 16 функций, зависящих от переменных x и y есть две константы — 0 и 1, и четыре функции одного существенного аргумента — x, $\overline{x}$, y, $\overline{y}$, то каждая из десяти оставшихся функций существенно зависит от x и y. Следовательно, каждая двуместная булева функция, существенно зависящая от двух своих аргументов, будет либо $\&$-функцией, либо $\oplus$-функцией.

Утверждение 4.3.4. *Пусть в схеме* S : (1) *к входу* x_i *подключен только один элемент* s *и этот элемент реализует* $\&$-*функцию*; (2) *второй вход* s *подключен к вершине* v, *в которой вычисляется функция, существенно зависящая только от переменных* $x_{i_1}, \ldots, x_{i_k}$. *Тогда найдутся такие постоянные* $\alpha_{i_1}, \ldots, \alpha_{i_k}$, *что после подстановки их вместо переменных* $x_{i_1}, \ldots, x_{i_k}$ *выход схемы* S *не зависит от* x_i.

Доказательство. Допустим, что к входу x_i подключен элемент s. Этот элемент реализует двухместную функцию $h(u,v) = (u^{(\alpha)}\&v^{(\beta)})^{(\gamma)}$, существенно зависящую от двух своих аргументов. Будем полагать, что к переменной x_i подключен второй вход элемента s. Тогда первый вход s подключен к некоторой вершине s_1, в которой вычисляется функция $h_1(x_{i_1},\ldots,x_{i_k})$, не зависящая от x_i, и, в силу утверждения 4.3.1, не равная тождественной постоянной. Подобная схема изображена на рисунке 4.3.5. В этой схеме условиям, наложенным на вход x_i, удовлетворяют входы x_2 и x_3. Если в качестве x_i взять вход x_2, то вершиной s_1 будет вход x_1. Подставляя вместо x_1 тождественную единицу видим, что функция, вычисляемая первым элементом схемы не зависит от x_2, так как $x_2 \vee 1 = 1$. Если в качестве x_i взять вход x_3, то вершиной s_1 будет вершина, реализующая конъюнкцию. После подстановки $x_1 = 0$, $x_2 = 0$ первый элемент схемы будет вычислять тождественный нуль, и, следовательно, тождественный нуль так же будет вычислять конъюнктор, т.е. вычисляемая им функция не зависит от x_3. Легко видеть, что и в общем случае найдутся значения $\alpha_{i_1},\ldots,\alpha_{i_k}$, переменных $x_{i_1},\ldots,x_{i_k}$, при которых $h_1(\alpha_{i_1},\ldots,\alpha_{i_k}) = \overline{\alpha}$. Тогда

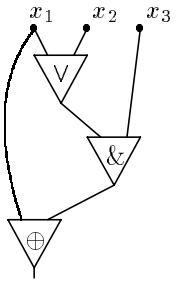


Рис. 4.3.5

$$\mathsf{S}(s) = h(h_1(\alpha_{i_1},\ldots,\alpha_{i_k}),x_i) = ((\overline{\alpha})^{(\alpha)}\&x_i^{(\beta)})^{(\gamma)} = (0\&x_i^{(\beta)})^{(\gamma)} = 0^{(\gamma)},$$

т.е. элемент s вычисляет функцию независящую от x_i. Следовательно, от x_i так же не зависит и выход схемы S. Утверждение доказано.

Задачи

4.3.1. Сформулировать и доказать аналог леммы 4.3.1 для схем, вычисляющих булевы операторы.

4.3.2. Сформулировать и доказать аналог леммы 4.3.2 для схем, вычисляющих булевы операторы.

4.4. Булевы функции трех переменных. Нижние оценки сложности.

Используя полученные в предыдущем параграфе результаты, покажем, что верхние оценки сложности вычисления булевых функций трех переменных в базисах B_0, B_1 и B_2, установленные в утверждениях 4.2.1–4.2.3, являются наилучшими из возможных.

Утверждение 4.4.1. *Справедливо неравенство*

$$L_{B_1}(x \oplus y \oplus z) \geq 6.$$

Доказательство. Пусть S — минимальная схема в базисе B_1, вычисляющая функцию $x \oplus y \oplus z$. Прежде всего покажем, что последний элемент схемы не может быть подключен ни к какому ее входу. Допустим, что это не так и первый вход последнего элемента, реализующего функцию $h(x,v) = (x^{(\alpha)}\&v^{(\beta)})^{(\gamma)}$, подключен к входу x. Легко видеть, что при $x = \overline{\alpha}$ выход схемы не зависит от значений двух оставшихся входов. С другой стороны, после подстановки $\overline{\alpha}$ вместо переменной x схема S должна вычислять функцию $\overline{\alpha} \oplus y \oplus z$, которая существенно зависит от y и z. Противоречие.

Покажем теперь, что в S к каждой переменной подключено не менее двух различных элементов. Допустим, что это не так, и к переменной x подключен только один элемент s. Из утверждения 4.3.4 следует, что найдутся значения y_0 и z_0 переменных y и z, при которых выход схемы S не зависит от x. С другой стороны, после подстановки постоянных y_0 и z_0 вместо переменных y и z схема S должна вычислять функцию $x \oplus y_0 \oplus z_0$, которая всегда существенно зависит от x. Противоречие.

Легко видеть, что доказанное свойство не зависит от числа входов схемы и справедливо не только для суммы, но и для ее отрицания. Поэтому в схемах S_2 и S_2', вычисляющих сумму $x \oplus y$ и эквивалентность $x \sim y$, к каждому входу также будет подключено не менее двух элементов. Отсюда немедленно следует, что каждая из этих схем содержит по крайней мере три элемента. Для того, что бы убедиться в этом достаточно сравнить число входящих и выходящих ребер в схеме из двух элементов. В такой схеме из двух входов выходят четыре ребра и еще одно ребро обязательно выходит из первого элемента — всего пять ребер. С другой стороны, в два элемента может входит не более четырех ребер. Противоречие.

Рассмотрим фрагмент схемы S, состоящий из входа x, двух его потомков s_1 и s_2 (здесь полагаем, что элемент s_2 расположен в схеме не выше элемента s_1), и элемента s_3 — потомка s_2. Из установленных выше свойств схемы S следует, что в S обязательно найдутся все три элемента $\mathsf{s}_1, \mathsf{s}_2, \mathsf{s}_3$. Пусть элемент s_2 реализует функцию $h(u,v) = (u^{(\alpha)} \& v^{(\beta)})^{(\gamma)}$. Рассматриваемый фрагмент изображен на рисунке 4.4.1, где $h = \&$. Подставим вместо переменной x постоянную $\overline{\alpha}$. После такой подстановки элемент s_1 будет реализовать функцию, существенно зависящую только от своего второго входа. Элемент s_2 будет вычислять константу $0^{(\gamma)}$, и, следовательно, элемент s_3 будет реализовать функцию, существенно зависящую только от своего второго входа. Из утверждений 4.3.1 и 4.3.2 следует, что элементы $\mathsf{s}_1, \mathsf{s}_2$ и s_3 можно удалить из схемы не изменяя вычисляемой схемой функции. После подстановки константы $\overline{\alpha}$ и удаления трех элементов, преобразованная схема будет вычислять либо $y \oplus z$, либо $y \sim z$, и, следовательно, состоять не менее чем из трех элементов. Поэтому в исходной схеме было не менее шести элементов. Утверждение доказано.

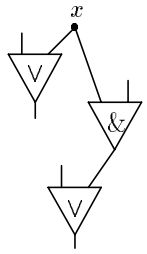


Рис. 4.4.1

Утверждение 4.4.2. *Справедливо неравенство*

$$L_{B_2}(x \vee y \vee z) \geq 5.$$

Доказательство. Пусть S — минимальная схема в базисе B_2, вычисляющая дизъюнкцию переменных x, y, z

$$x \vee y \vee z = x \oplus y \oplus z \oplus xy \oplus xz \oplus yz \oplus xyz. \tag{4.4.1}$$

Так как степень дизъюнкцию трех переменных равна трем, то S обязательно содержит два элемента конъюнкции. Покажем, что S также содержит не менее трех линейных элементов. Сделаем это методом от противного, используя следующее простое свойство дизъюнкции: если $x \vee y \vee z = f(x,y,z) \& g(x,y,z)$, то одна из функций f или g равна дизъюнкции $x \vee y \vee z$.

Действительно, из равенства $f \& g = x \vee y \vee z$ следует, что $f, g \geq x \vee y \vee z$. Но легко видеть, что это неравенство справедливо только для двух функций — самой дизъюнкции $x \vee y \vee z$ и тождественной единицы.

Из указанного свойства немедленно следует, что в схеме S последний элемент должен быть линейным.

Итак, допустим, что в схеме S есть только два линейных элемента: s_1— первый линейный элемент S; s_2 — второй линейный элемент S, являющийся ее выходом. Вычисляемая элементом s_1 функция v_1 будет суммой двух одночленов h_1 и h_2 и постоянной α:

$$v_1 = h_1 \oplus h_2 \oplus \alpha.$$

Тогда вычисляемую элементом s_2 функцию v_2 можно представить в одном из двух следующих видов:

$$v_1 = p(h_1 \oplus h_2 \oplus \alpha) \oplus q \oplus \beta, \tag{4.4.2}$$

$$v_1 = p(h_1 \oplus h_2 \oplus \alpha) \oplus q(h_1 \oplus h_2 \oplus \alpha) \oplus \beta, \tag{4.4.3}$$

где p и q — одночлены, β — константа. Если имеет место равенство (4.4.2), то легко видеть, что многочлен Жегалкина функции v_2 состоит не более чем из пяти одночленов, что противоречит равенству (4.4.1).

Теперь рассмотрим равенство (4.4.3). Возможны несколько случаев.

1. $\beta = 0$. В этом случае после раскрытия скобок в (4.4.3) получим многочлен состоящий не более чем из шести одночленов. Противоречие с (4.4.1).

2. $\beta = 1, \alpha = 0$. В этом случае после раскрытия скобок в (4.4.3) получим многочлен состоящий не более чем из пяти одночленов. Противоречие с (4.4.1).

3. $\beta = 1, \alpha = 1, p, q \neq 1$. После раскрытия скобок в (4.4.3) получим многочлен с ненулевым свободным членом. Противоречие с (4.4.1).

4. $\beta = 1, \alpha = 1, q = 1$. После раскрытия скобок в (4.4.3) получим многочлен состоящий не более чем из пяти одночленов. Противоречие с (4.4.1).

Таким образом, во всех возможных случаях вычисляемая схемой S функция не является дизъюнкцией. Следовательно, минимальная схема содержит не менее трех линейных элементов. Утверждение доказано.

Теперь покажем, что среди трехместных булевых функций есть функция, сложность которой в базисе из всех двухместных булевых функций не меньше четырех. Такой функцией является функция голосования

$$\tau_2(x, y, z) = xy \oplus xz \oplus yz \tag{4.4.4}$$

Утверждение 4.4.3. *Справедливо неравенство*

$$L_{B_0}(\tau_2(x, y, z)) \geq 4.$$

Доказательство. Прежде всего отметим, что подстановка произвольной константы на место любого аргумента функции голосования преобразует ее в функцию, существенно зависящую от двух оставшихся аргументов. Подставляя константы вместо переменной x в (4.4.4), легко видеть, что

$$\tau_2(0, y, z) = yz, \quad \tau_2(1, y, z) = y \oplus z \oplus yz. \tag{4.4.5}$$

Так как функция голосования является симметрической функцией, то равенства, аналогичные (4.4.5), справедливы и при подстановке констант вместо переменных y и z.

Пусть S — минимальная схема, вычисляющая функцию голосования. Перенумеруем элементы схемы S так, чтобы первый номер получил элемент, входы которого подключены только к независимым переменным, а последний номер получил выход схемы.

Допустим, что минимальная схема S состоит из двух элементов. Так как функция голосования является симметрической функцией, то без ограничения общности будем полагать, что входы первого элемента схемы подключены к переменным x и y. В этом случае схема S выглядит так, как это изображено на рис. 4.4.2. Пусть первый элемент реализует функцию f, а второй — функцию g. Покажем, что ни f, ни g не могут быть &-функциями. Действительно, если f является &-функцией, то из утверждения 4.3.3 следует существование такой постоянной α, что после ее подстановки вместо переменной y выход схемы не зависит от переменной x. Получили противоречие с (4.4.5). Если &-функцией является функция $g = (u^{(\alpha)} \& z^{(\beta)})^{(\gamma)}$, то подставляя вместо переменной z константу $\overline{\beta}$ убеждаемся, что вычисляемая схемой S функция будет константой. Снова противоречие с (4.4.5).

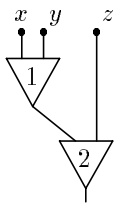


Рис. 4.4.2

Таким образом, функции f и g должны быть $\oplus$-функциями. Следовательно, вычисляемая схемой S функция должна быть линейной. Противоречие с (4.4.4).

Теперь предположим, что вычисляющая функцию голосования минимальная схема S состоит из трех элементов. Снова без ограничения общности будем полагать, что входы первого элемента схемы подключены к переменным x и y.

Общее число входов второго и третьего элементов схемы равно четырем. Три из этих четырех входов должны быть обязательно подключены: (i) к первому элементу; (ii) к

второму элементу; (iii) к переменной z. Следовательно, оставшейся четвертый вход может быть подключен либо к одной из переменных x и y (без ограничения общности будем полагать, что такой переменной будет y), либо к переменной x, либо к первому элементу схемы. Таким образом в схеме S два элемента могут подключаться либо к переменной y, либо к переменной z, либо к первому элементу. К переменной x обязательно подключен только первый элемент схемы.

Далее рассмотрим два случая:

(i) к первому элементу схемы S подключен элемент, который не является выходом схемы;

(ii) к первому элементу схемы S не подключен элемент, который не является выходом схемы.

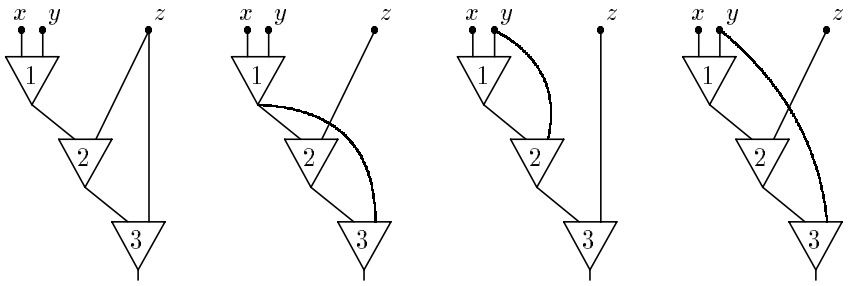


Рис. 4.4.3

Рассмотрим первый случай. С учетом сделанных выше предположений, в первом случае возможны только четыре различных конструкции схемы S. Все они представлены на рисунке 4.4.3. Из рисунка видно, что три левые схемы не являются минимальными. Каждая из этих схем может быть преобразована в схему из двух элементов так, что старая и новая схемы будут вычислять одну и туже функцию. В качестве примера рассмотрим самую левую схему. В этой схеме второй и третий элементы вычисляют функцию, зависящую от двух аргументов — переменной z и функции, вычисляемой первым элементом. Поэтому второй и третий элементы можно заменить одним, реализующим подходящую двуместную функцию. Преобразованная схема будет выглядеть так, как это изображено на рис. 4.4.2.

Теперь рассмотрим правую схему. Как и при анализе двухэлементной схемы легко показать, что ни один из элементов не может быть &-элементом. Подстановка вместо переменной y или z подходящей константы преобразует схему S в схему, вычисляющую константу или функцию одной переменной, что противоречит (4.4.5). Но если все элементы являются $\oplus$-элементами, то вычисляемая схемой S функция будет линейной. Противоречие с (4.4.4). Таким образом, ни одна из изображенных на рис. 4.4.3 схем не вычисляет функцию голосования. Случай (i) рассмотрен полностью.

Рассмотрим второй случай. С учетом сделанных предположений второй элемент схемы подключен к переменным y и z. Поэтому схема S выглядит так, как это изображено на рис. 4.4.4. Как и ранее, легко показать, что первый и второй элементы не могут быть &-элементами. Подстановка вместо переменной y подходящей константы преобразует схему S в схему, вычисляющую константу или функцию одной переменной, что противоречит (4.4.5). Допустим, что первый и второй элементы являются $\oplus$-элементами, реализующими функции $x \oplus y \oplus \alpha$ и $y \oplus z \oplus \beta$. Преобразуем схему S, заменив переменные y и z переменной x. В преобразованной схеме первый элемент вычисляет константу α, второй элемент — константу β. Следовательно, сама схема также вычисляет тождественную константу. Пришли к противоречию с (4.4.4), так как подстановка $y = x$, $z = x$ преобразует функцию голосования в функцию x. Случай (ii) рассмотрен полностью.

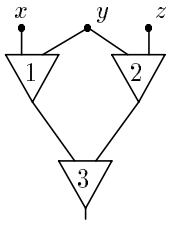


Рис. 4.4.4

Таким образом, минимальная схема функции голосования состоит не менее чем из

четырех элементов, т.е. $L_{B_0}(\tau_2(x,y,z)) \geq 4$. Утверждение доказано.

Задачи

4.4.1. Доказать равенства

a) $L_{\{\vee,\&,\neg\}}(1) = 2$; b) $L_{\{\vee,\&,\neg\}}(0) = 2$; c) $L_{\{\vee,\&,\neg\}}(x \to y) = 2$;
d) $L_{\{\vee,\&,\neg\}}(x|y) = 2$; e) $L_{\{\vee,\&,\neg\}}(x \downarrow y) = 2$; f) $L_{\{\vee,\&,\neg\}}(x \oplus y) = 4$;
g) $L_{\{\vee,\&,\neg\}}(x \sim y) = 4$.

4.4.2. Доказать равенства

a) $L_{\{\oplus,\&,1\}}(x \sim y) = 3$; b) $L_{\{\oplus,\&,1\}}(0) = 1$; c) $L_{\{\oplus,\&,1\}}(x|y) = 2$;
d) $L_{\{\oplus,\&,1\}}(x \downarrow y) = 4$; e) $L_{\{\oplus,\&,1\}}(x \vee y) = 3$; f) $L_{\{\oplus,\&,1\}}(x \to y) = 4$.

4.4.3. Доказать равенства

a) $L_{\downarrow}(x \vee y) = 2$; b) $L_{\downarrow}(0) = 2$; c) $L_{\downarrow}(x \to y) = 3$; d) $L_{\downarrow}(x\&y) = 3$;
e) $L_{\downarrow}(1) = 3$; f) $L_{\downarrow}(x|y) = 4$; g) $L_{\downarrow}(x \sim y) = 4$; h) $L_{\downarrow}(x \oplus y) = 5$.

4.4.4. Доказать равенства

a) $L_{|}(1) = 2$; b) $L_{|}(x \to y) = 2$; c) $L_{|}(x\&y) = 2$; d) $L_{|}(x \vee y) = 3$;
e) $L_{|}(0) = 3$; f) $L_{|}(x \downarrow y) = 4$; g) $L_{|}(x \oplus y) = 4$; h) $L_{|}(x \sim y) = 5$.

4.4.5. Найти $L_{B_0}(xyz \vee \overline{x}\,\overline{y}\,\overline{z})$.

4.4.6. Найти $L_{\{\&,\vee,\neg\}}(x \oplus y \oplus z)$.

4.5. Сложность функций, зависящих от большого числа аргументов

Будем рассматривать булевы функции f_n, число аргументов которых зависит от натурального параметра n, меняющегося от некоторой положительной константы до бесконечности[3)]. Например, такими функциями будут линейная функций $l_n = x_1 \oplus \cdots \oplus x_n$ и дизъюнкция $\vee_n = x_1 \vee \cdots \vee x_n$. Естественно, что сложность $L(f_n)$ каждой такой функции f_n зависит от n. Как правило точное значение сложности таких функций найти не удается. Более того, даже приблизительное определение сложности является очень трудной задачей, решение которой находится только в редких случаях. При этом основные трудности возникают при доказательстве нижних оценок сложности. Однако для некоторых функций удается найти точное или асимптотически точное (при $n \to \infty$) значение их сложности. Ниже рассматриваются две такие функции.

1. Рассмотрим n-местную дизъюнкцию $x_1 \vee \cdots \vee x_n$. Легко видеть, что дизъюнкция сохраняет нуль, а функции $\&$ и $\oplus$ образуют базис в T_0. Поэтому дизъюнкцию $x_1 \vee \cdots \vee x_n$ можно вычислить схемой в базисе $\{\&, \oplus\}$. Далее найдем сложность такого вычисления, а затем покажем, что использование функций, не сохраняющих нуль, позволит уменьшить сложность вычисления дизъюнкции.

Утверждение 4.5.1. *Справедливы соотношения*

$$L_{\{\&,\oplus\}}(x_1 \vee \ldots \vee x_n) = 3n - 3,$$
$$D_{\{\&,\oplus\}}(x_1 \vee \ldots \vee x_n) \leq 2\lceil \log_2 n \rceil.$$

Доказательство. Верхняя оценка. Положим $m = \lfloor n/2 \rfloor$, $l = \lceil n/2 \rceil$. Схема S_n, вычисляющая функцию $x_1 \vee \ldots \vee x_n$, легко строится индуктивно в соответствии с формулой

$$x_1 \vee \ldots \vee x_n = (x_1 \vee \ldots \vee x_m)(x_{m+1} \vee \ldots \vee x_n) \oplus \oplus (x_1 \vee \ldots \vee x_m) \oplus (x_{m+1} \vee \ldots \vee x_n).$$

[3)] Фактически речь идет последовательности функций $\{f_n\}$, в которой n-я функция зависит от n переменных, и все функции последовательности вычисляются при помощи некоторого одного алгоритма.

Полагая, что $m, l \geq 2$, $L(\mathsf{S}_m) = 3m - 3$, $L(\mathsf{S}_l) = 3l - l$, видим, что

$$L(\mathsf{S}_n) = L(\mathsf{S}_m) + L(\mathsf{S}_l) + 3 = (3m - 3) + (3l - 3) + 3 = 3n - 3.$$

Полагая, что $D(\mathsf{S}_m) \leq D(\mathsf{S}_l) = 2\lceil \log_2 l \rceil$ и учитывая, что для любого нечетного n большего единицы $\lceil \log_2 n \rceil = \lceil \log_2 (n+1) \rceil$, а для любого четного — $2\lceil n/2 \rceil = n$, имеем

$$D(\mathsf{S}_n) = D(\mathsf{S}_l) + 2 = 2\lceil \log_2 \lceil n/2 \rceil \rceil + 2 = 2\lceil \log_2 2\lceil n/2 \rceil \rceil = 2\lceil \log_2 n \rceil.$$

Верхние оценки доказаны.

Нижняя оценка сложности. Пусть S — минимальная схема в базисе $\{\&, \oplus\}$ для дизъюнкции n переменных. Покажем, что

$$L_{\{\&,\oplus\}}(x_1 \vee \ldots \vee x_n) \geq 3n - 3. \tag{4.5.1}$$

Прежде всего, покажем, что в S к каждому входу обязательно должен быть подключен хотя бы один элемент, реализующий функцию $\oplus$. Действительно, допустим, что к n-му входу подключены только элементы $\mathsf{s}_1, \ldots, \mathsf{s}_p$, реализующие конъюнкции. В S положим $x_1 = \ldots = x_{n-1} = 0$. Так как базис схемы S сохраняет нуль, то все ее вершины, находящиеся в схеме выше элементов $\mathsf{s}_1, \ldots, \mathsf{s}_p$ и не являющиеся n-м входом, будут вычислять тождественный нуль. Поэтому тождественный нуль будет вычисляться в вершинах, к которым подключены вторые входы элементов $\mathsf{s}_1, \ldots, \mathsf{s}_p$, а, следовательно, и в самих элементах $\mathsf{s}_1, \ldots, \mathsf{s}_p$. Последнее означает, что схема S вычисляет тождественный нуль при любом значении переменной x_n. Пришли к противоречию.

Теперь покажем, что в S обязательно найдется вход, к которому подключено не менее двух элементов. Предположим, что это не так. Тогда в схеме найдутся два входа, к которым подключен один и тот же элемент s. Без ограничения общности будем полагать, что такими входами будут $(n-1)$-й и n-й входы. В S положим $x_1 = \ldots = x_{n-2} = 0$, $x_{n-1} = x_n = x$. После такой подстановки все элементы S будут вычислять тождественный нуль при любом значении переменной x. Снова пришли к противоречию.

Пусть в S к n-му входу подключены два элемента v и u. Без ограничения общности будем полагать, что элемент u реализует $\oplus$. В S положим $x_n = 0$. Покажем, что после такой подстановки из S можно удалить не менее двух элементов, реализующих $\oplus$. Для этого рассмотрим всевозможные цепочки элементов $\mathsf{v} = \mathsf{v}_0, \mathsf{v}_1, \ldots, \mathsf{v}_k$ в которых каждый элемент v_{i+1} подключен к элементу v_i. Среди этих цепочек обязательно найдется такая, в которой первые k элементов реализуют конъюнкции, а последний — $\oplus$. (Если элемент v_0 реализует $\oplus$, то $k = 0$.) Слева на рисунке 4.5.1 изображена подобная цепочка из трех элементов. Если такой цепочки нет, то в S существует цепочка из одних конъюнкций, которая связывает n-й вход и выход схемы. В этом случае подстановка нуля вместо x_n приведет к тому, что в последнем элементе схемы будет вычисляться нуль независимо от значения первых $n-1$ переменных. Итак, пусть $\mathsf{w} = \mathsf{v}_k$ — элемент, реализующий $\oplus$ и связанный с v цепочкой из одних конъюнкций. Если w и u различные элементы, то именно они будут удалены после подстановки $x_n = 0$. Допустим теперь, w и u совпадают (подобная ситуация изображена в правой части рисунка 4.5.1). Сохраним за этим элементом обозначение w. Легко видеть, что после подстановки $x_n = 0$ вычисляемая в w функция будет тождественным нулем. Как и выше, рассматривая всевозможные цепочки конъюнкций, начинающиеся в w, найдем элемент z, реализующий $\oplus$.

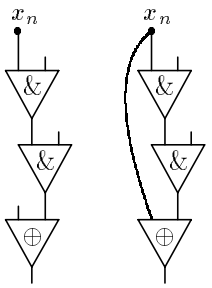


Рис. 4.5.1

Преобразование схемы, заключающееся в подстановке вместо x_i тождественного нуля и последующее удаление из схемы i-го входа и всех вершин, вычисляющих тождественный нуль, назовем операцией удаления i-го входа схемы.

Из доказанного выше следует, что из схемы S можно удалять входы до тех пор, пока в ней не останется единственный не удаленный вход. Следовательно, операция удаления входа может быть выполнена $n-1$ раз. При этом из S будет удалено не менее чем $2n -$

2 элемента $\oplus$. Кроме того, легко видеть, что S содержит не менее чем $n-1$ элемент конъюнкции. Это следует из того, что степень дизъюнкции n переменных равна n. Таким образом, общее число элементов в схеме S не меньше чем $3n-3$. Неравенство (4.5.1) доказано. Утверждение доказано полностью.

Дизъюнкция принадлежит классу T_0, и как видно из доказанного утверждения, достаточно просто вычисляется схемами в базисе, целиком содержащемся в T_0. С другой стороны, легко видеть, что $x_1 \vee \cdots \vee x_n = (x_1 \oplus 1) \cdot \ldots \cdot (x_n \oplus 1) \oplus 1$. Откуда легко следует неравенство

$$L_{\{\&,\oplus,1\}}(x_1 \vee \cdots \vee x_n) \le 2n+1.$$

Таким образом, расширение базиса схемы до базиса полного в P_2 позволяет уменьшить вычисления в полтора раза.

2. Рассмотрим функцию $\tau_2(x_1,\ldots,x_n)$, равную единице, если среди ее аргументов найдется не менее двух единиц, и равную нулю в противном случае. Как и рассмотренная выше дизъюнкция, τ_2 содержится в T_0. Однако в отличие от дизъюнкции, сложность вычисления τ_2 практически одинакова в полном и неполном базисах.

Утверждение 4.5.2. *Пусть $n \to \infty$. Тогда*

$$L(\tau_2(x_1,\ldots,x_n)) \sim 2n, \qquad D(\tau_2(x_1,\ldots,x_n)) \sim \log_2 n.$$

Утверждение следует из четырех приводимых далее лемм. В первых трех устанавливается верхние оценки сложности и глубины, в четвертой — нижняя оценка сложности.

Лемма 4.5.1. *Пусть m — целое, $n = 2^m$. Тогда*:
(i) существуют зависящие от булевых переменных $x_1,\ldots,x_n$ такие булевы функции $f_0, f_1, \ldots, f_m$, что

$$f_0 = x_1 \vee \cdots \vee x_n, \quad \tau_2(x_1,\ldots,x_n) = f_1 \vee \ldots \vee f_m; \tag{4.5.2}$$

(ii) существует вычисляющая функции $f_0, f_1, \ldots, f_m$ схема S *для сложности и глубины которой справедливы неравенства*

$$L(\mathsf{S}) \le 3n-1, \quad D(\mathsf{S}) = m.$$

Доказательство. Лемму докажем индукцией по m. При $m=1$ утверждение леммы очевидно, искомыми функциями являются дизъюнкция $x_1 \vee x_2$ и конъюнкция $x_1 \& x_2$ двух переменных. Вычисляющая их схема S_1 состоит из двух элементов и ее глубина равна единице. Предположим, что лемма справедлива для всех целых не превосходящих $k-1$.

(i) По предположению индукции существуют зависящие от переменных $x_1,\ldots,x_{2^{k-1}}$ функции $f_0^1, f_1^1, \ldots, f_k^1$ и зависящие от переменных $x_{2^{k-1}+1},\ldots,x_{2^k}$ функции $f_0^2, f_1^2, \ldots, f_k^2$ для которых справедливы соотношения (4.5.2). Новые функции $f_0, f_1, \ldots, f_k$ определим следующим образом:

$$f_i = f_i^1 \vee f_i^2, \ \text{ при } i = 0,1,\ldots,k-1; \quad f_k = f_0^1 \& f_0^2. \tag{4.5.3}$$

Очевидно, что $f_0 = x_1 \vee \ldots \vee x_{2^k}$ и

$$f_1 \vee \ldots \vee f_k = (f_1^1 \vee \ldots \vee f_{k-1}^1) \vee (f_1^2 \vee \ldots \vee f_{k-1}^2) \vee f_0^1 \& f_0^2. \tag{4.5.4}$$

По построению функция $f_1 \vee \cdots \vee f_k$ — монотонная, поэтому для доказательства второго равенства в (4.5.2) достаточно убедиться, что эта функция равна нулю на любом наборе веса 1, и равна единице на любом наборе веса 2.

Пусть $\boldsymbol{\alpha}$ — произвольный булев набор длины 2^k и веса 2. Его первую половину обозначим через $\boldsymbol{\alpha}_1$, вторую — через $\boldsymbol{\alpha}_2$. Две единицы набора $\boldsymbol{\alpha}$ могут располагаться (1) в первой половине этого набора, (2) во второй половине, (3) одна единица может находиться в первой половине набора, а вторая — во второй. По предположению индукции в

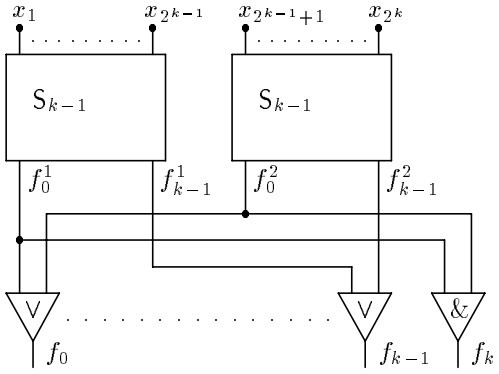


Рис. 4.5.2

случае (1) $(f_1^1 \vee \cdots \vee f_{k-1}^1)(\boldsymbol{\alpha}_1) = 1$, в случае (2) $(f_1^2 \vee \cdots \vee f_{k-1}^2)(\boldsymbol{\alpha}_2) = 1$, в случае (3) $f_0^1(\boldsymbol{\alpha}_1)\&f_0^2(\boldsymbol{\alpha}_2) = 1$. Следовательно, в силу (4.5.4)

$$(f_1 \vee \cdots \vee f_k)(\boldsymbol{\alpha}) = 1.$$

Теперь рассмотрим произвольный булев набор $\boldsymbol{\beta}$ длины 2^k и веса 1. Без ограничения общности будем полагать, что единственная единица $\boldsymbol{\beta}$ находится в его первой половине, т.е. $\|\boldsymbol{\beta}_1\| = 1$, $\|\boldsymbol{\beta}_2\| = 0$. Тогда по предположению индукции

$$(f_1^1 \vee \ldots \vee f_{k-1}^1)(\boldsymbol{\beta}_1) = 0, \quad f_0^1(\boldsymbol{\beta}_1) = 1,$$
$$(f_1^2 \vee \ldots \vee f_{k-1}^2)(\boldsymbol{\beta}_2) = 0, \quad f_0^2(\boldsymbol{\beta}_2) = 0.$$

Из последних равенств и (4.5.4) легко следует, что

$$(f_1 \vee \cdots \vee f_k)(\boldsymbol{\beta}) = 0.$$

Первое утверждение леммы доказано.

(ii) Докажем второе утверждение. Для построения схемы S_k, вычисляющей функции $f_0, f_1, \ldots, f_k$ воспользуемся равенствами (4.5.3). Схема S_k состоит из двух подсхем, каждая из которых является экземпляром схемы S_{k-1}, k дизъюнкторов и одного конъюнктора. Конструкция схемы показана на рис. 4.5.2. Легко видеть, что

$$L(\mathsf{S}_k) = 2L(\mathsf{S}_{k-1}) + (k+1), \quad D(\mathsf{S}_k) = D(\mathsf{S}_{k-1}) + 1. \tag{4.5.5}$$

По предположению индукции глубина схем, вычисляющих функции $f_0^1, f_1^1, \ldots, f_{k-1}^1$ и $f_0^2, f_1^2, \ldots, f_{k-1}^2$, равна $k-1$. Следовательно,

$$D(\mathsf{S}_k) = D(\mathsf{S}_{k-1}) + 1 = k.$$

Для оценки сложности схемы S_k последовательно $(k-1)$ раз применим первое равенство в (4.5.5). Учитывая, что $L(\mathsf{S}_1) = 2$ получаем

$$L(\mathsf{S}_k) = 2L(\mathsf{S}_{k-1}) + (k+1) = 2(2L(\mathsf{S}_{k-2}) + k) + (k+1) = \ldots =$$
$$= 2^{k-1}L(\mathsf{S}_1) + \sum_{i=3}^{k+1} 2^{k+1-i} i < 2^k + 2^{k+1} \sum_{i=3}^{\infty} i \cdot 2^{-i} = 3 \cdot 2^k.$$

Лемма доказана.

Лемма 4.5.2. *Справедливы неравенства*

$$L(\tau_2(x_1, \ldots, x_n) \le 6n + \log_2 n + 1,$$
$$D(\tau_2(x_1, \ldots, x_n) \le \lceil \log_2 n \rceil + \lceil \log_2 \log_2 n \rceil.$$

Доказательство. Пусть $k = \lceil \log_2 n \rceil$. Из предыдущей леммы следует, что существует схема сложности не более $3 \cdot 2^k$ и глубины не более k, которая вычисляет такие функции $f_1, \ldots, f_k$, что

$$\tau_2(x_1, \ldots, x_{2^k}) = f_1 \vee \ldots \vee f_{2^k}.$$

Дизъюнкция k функций легко вычисляется схемой сложности k и глубины $\lceil \log_2 k \rceil$. Следовательно, найдется схема S, вычисляющая функцию $\tau_2(x_1, \ldots, x_{2^k})$, сложность и глубина которой не превосходят, соответственно, величин $3 \cdot 2^k + k$ и $k + \lceil \log_2 k \rceil$. Воспользуемся этой схемой, подставив вместо недостающих $(2^k - n)$ аргументов нули. Так как $n \leq 2^k < 2n$, то легко видеть, что оба неравенства из утверждения леммы справедливы. Лемма доказана.

Лемма 4.5.3. *Пусть $n \to \infty$. Тогда*

$$L(\tau_2(x_1, \ldots, x_n)) \leq 2n + \mathcal{O}\left(\sqrt{n}\right),$$
$$D(\tau_2(x_1, \ldots, x_n)) \leq \log_2 n + \log_2 \log_2 n + 3.$$

Доказательство. Положим $m = \lceil \sqrt{n} \rceil$. Каждое целое $k = 1, 2, \ldots, n$ представим парой чисел (k_1, k_2), $1 \leq k_1, k_2 \leq m$, так, что $k = (k_1 - 1)m + k_2$. Положим $a_i = x_{i,1} \vee \cdots \vee x_{i,m}$, $b_j = x_{1,j} \vee \cdots \vee x_{m,j}$. Тогда имеет место равенство

$$\tau_2(x_1, \ldots, x_n) = \tau_2(a_1, \ldots, a_m) \vee \tau_2(b_1, \ldots, b_m). \tag{4.5.6}$$

В правой части равенства (4.5.6) стоит монотонная функция. Поэтому для того, чтобы убедиться в справедливости (4.5.6) достаточно показать, что стоящая справа функция равна нулю на любом наборе веса 1, и равна единице на любом наборе веса 2.

Пусть $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ — булев набор веса 1. Будем полагать, что $\alpha_{p,q} = 1$. Тогда среди функций $a_i(\boldsymbol{\alpha})$ и $b_j(\boldsymbol{\alpha})$ равны единице только $a_p(\boldsymbol{\alpha})$ и $b_q(\boldsymbol{\alpha})$. Следовательно, правая часть (4.5.6) равна нулю. Пусть $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_n)$ — булев набор веса 2. Будем полагать, что $\beta_{p,q} = \beta_{s,t} = 1$. Очевидно, что либо $p \neq s$, либо $q \neq t$. Если имеет место первое из этих неравенств, то $a_p(\beta) = a_s(\beta) = 1$. Если справедливо второе, то $b_q(\beta) = b_t(\beta) = 1$. Легко видеть, что в обоих случаях правая часть (4.5.6) равна единице. Таким образом, справедливость равенства (4.5.6) установлена.

Воспользуемся этим равенством для построения схемы S, вычисляющей τ_2. Эта схема состоит из: (1) m подсхем A_i, вычисляющих функции a_i; (2) m подсхем B_j, вычисляющих функции b_j; (3) подсхемы S_1, вычисляющей функцию τ_2 аргументами которой являются функции a_i; (4) подсхемы S_2, вычисляющей функцию τ_2 аргументами которой являются функции b_j; (5) дизъюнктора, вычисляющего дизъюнкцию функций, вычисленных подсхемами S_1 и S_2.

Очевидно, что сложность каждой из подсхем A_i и B_j равна $m-1$, а глубина — $\lceil \log_2 m \rceil = \lceil \frac{1}{2} \log_2 n \rceil \leq \frac{1}{2} \log_2 n + 1$. Из леммы 4.5.2 следует, что для каждой S_i

$$L(\mathsf{S}_i) \leq 6m + \log_2 m + 1 = \mathcal{O}\left(\sqrt{n}\right),$$
$$D(\mathsf{S}_i) \leq \lceil \log_2 m \rceil + \lceil \log_2 \log_2 m \rceil \leq \frac{1}{2} \log_2 n + \log_2 \log_2 n + 2.$$

Следовательно,

$$L(\mathsf{S}) = 2m(m-1) + \mathcal{O}\left(\sqrt{n}\right) = 2n + \mathcal{O}\left(\sqrt{n}\right),$$
$$D(\mathsf{S}) \leq \log_2 n + \log_2 \log_2 n + 3.$$

Лемма доказана.

Лемма 4.5.4. *Пусть $n \geq 3$. Тогда*

$$L(\tau_2(x_1, \ldots, x_n)) \geq 2n - 2.$$

Доказательство. Утверждение докажем индукцией по числу переменных функции τ_2. В основание индукции положим функцию трех переменных. В утверждении 4.4.3 было показано, что $L(\tau_2(x_1, x_2, x_3)) = 4$. Предположим, что для произвольного числа переменных n, не превосходящего $k-1$, утверждение доказано. Покажем, что оно справедливо при $n = k$.

Пусть S — минимальная схема для функции $\tau_2(x_1, \ldots, x_k)$. Допустим, что в схеме S найдется вход к которому подключено не менее двух элементов. Без ограничения общности полагаем, что таким входом будет последний k-й вход, и к нему подключены элементы s_1 и s_2. Вместо переменной x_k подставим тождественный нуль. Легко видеть, что после подстановки нуля схема будет вычислять функцию $\tau_2(x_1, \ldots, x_{k-1})$, а элементы s_1 и s_2 будут реализовать не более чем одноместные функции. Из утверждения 4.3.1 следует, что элементы s_1 и s_2 можно так удалить из S, что новая схема S′ будет вычислять функцию $\tau_2(x_1, \ldots, x_{k-1})$ и содержать по крайней мере на два элемента меньше, чем исходная схема. Для функции $\tau_2(x_1, \ldots, x_{k-1})$ справедливо предположение индукции

$$L(\tau_2(x_1, \ldots, x_{k-1}) \geq 2(k-1) - 2 = 2k - 4.$$

Следовательно,

$$L(\tau_2(x_1, \ldots, x_k) \geq L(\tau_2(x_1, \ldots, x_{k-1}) + 2 \geq 2k - 2.$$

Теперь рассмотрим случай когда в схеме S к каждому входу подключен ровно однин элемент. Без ограничения общности будем полагать, что к входам x_{k-1} и x_k подключен один и тот же элемент s, реализующий функцию v. Допустим, что v — $\oplus$-функция. Тогда отождествим переменные x_{k-1} и x_k, положив $x = x_{k-1} = x_k$. Легко видеть, что после такого отождествления функция, вычисляемая элементом s, а, следовательно, и функция, вычисляемая всей схемой в целом, не зависит от новой переменной x. В тоже время отождествление любых двух аргументов функции $\tau_2(x_1, \ldots, x_k)$ при $k \geq 3$ приводит к функции существенно зависящей от всех $k-1$ аргументов. Пришли к противоречию.

Далее рассмотрим случай, когда v — &-функция. Тогда в элементе s вычисляется функция $v(x_{k-1}, x_k) = \left(x_{k-1}^{(\alpha)} \& x_k^{(\beta)}\right)^{(\gamma)}$. Преобразуем схему S подставив вместо переменной x_k постоянную $\overline{\beta}$. Так как $v(x_{k-1}, \overline{\beta}) = 0^{(\gamma)}$ — константа, то новая схема S′ вычисляет функцию, для которой переменная x_{k-1} не является существенной. Противоречие. Лемма доказана.

Задачи

4.5.1. Показать, что $L_{\{\&,\neg\}}(x_1 \vee \ldots \vee x_n) = 2n$.

4.5.2. Показать, что $L_{\{\vee,\neg\}}(x_1 \& \ldots \& x_n) = 2n$.

4.5.3. Показать, что $L_{B_1}(x_1 \oplus \ldots \oplus x_n) = 3n - 3$.

4.5.4. Показать, что $L_{\{\&,\vee,\neg\}}(x_1 \oplus \ldots \oplus x_n) = 4n - 4$.

4.5.5. Найти $L_{\{\&,\oplus,1\}}(x_1 \vee \ldots \vee x_n)$.

4.5.6. Показать, что для любой булевой функции $f(x_1, \ldots, x_n)$ имеют место неравенства
а) $L_{\{\&,\neg\}} f(x_1, \ldots, x_n) \leq 2L_{\{\&,\vee,\neg\}}(f) + n$; b) $L_{\{\vee,\neg\}} f(x_1, \ldots, x_n) \leq 2L_{\{\&,\vee,\neg\}}(f) + n$.

Глава 5.

Специальные булевы функции и операторы

В этой главе рассматриваются эффективные схемы, вычисляющие некоторые важные с теоретической и практической точек зрения булевы функции и операторы. В частности, подробно изучается сложность и глубина вычисления суммы, разности и произведения целых чисел. Базис всех встречающихся в этой главе схем состоит из всех не более чем двухместных булевых функций.

5.1. Вычисление суммы и разности двух целых чисел

1. Рассмотрим булев $(n+1, 2n)$-оператор *сложения* S_n, вычисляющий сумму двух n-разрядных целых положительных чисел, представленных в двоичной системе счисления. Пусть

$$\boldsymbol{x} = \sum_{i=1}^{n} x_i 2^{i-1}, \quad \boldsymbol{y} = \sum_{i=1}^{n} y_i 2^{i-1}, \quad \boldsymbol{z} = \sum_{i=1}^{n+1} z_i 2^{i-1},$$

где $\boldsymbol{x} + \boldsymbol{y} = \boldsymbol{z}$. Тогда

$$S_n(x_1, \ldots, x_n, y_1, \ldots, y_n) = (z_1, \ldots, z_{n+1}).$$

Схему, вычисляющую оператор S_n, назовем n-разрядным *сумматором*. Справедливо следующее утверждение.

Лемма 5.1.1. *Существует n-разрядный сумматор Σ_n, для сложности и глубины которого справедливы равенства*

$$L(\Sigma_n) = 5n - 3, \quad D(\Sigma_n) = 2n - 1.$$

Доказательство. Для построения схемы Σ_n воспользуемся хорошо известным алгоритмом сложения целых чисел "столбиком". В этом алгоритме j-й разряд суммы z_j равен сумме j-х разрядов слагаемых и переноса q_j из предыдущих $j-1$ разрядов, т.е.

$$\begin{array}{cc} & q_{n+1}\ q_n\ \ldots\ q_2 \\ + & x_n\ \ldots\ x_2\ x_1 \\ & y_n\ \ldots\ y_2\ y_1 \\ \hline & z_{n+1}\ z_n\ \ldots\ z_2\ z_1 \end{array}$$

$$z_j = x_j \oplus y_j \oplus q_j. \tag{5.1.1}$$

Легко видеть, что для переноса в $(j+1)$-й разряд справедлива следующая формула

$$q_{j+1} = x_j y_j \oplus x_j q_j \oplus y_j q_j = x_j y_j \oplus q_j(x_j \oplus y_j). \tag{5.1.2}$$

Так как перенос в первый разряд отсутствует, то функции z_1 и q_2 вычисляются по формулам

$$z_1 = x_1 \oplus y_1, \quad q_2 = x_1 \& y_1. \tag{5.1.3}$$

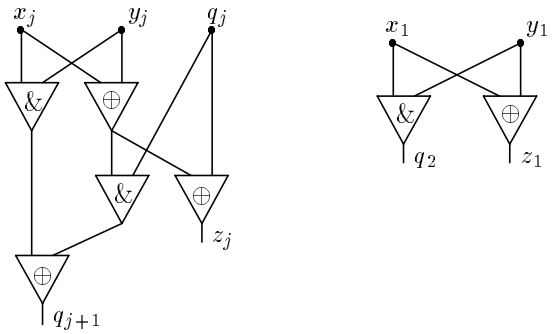


Рис. 5.1.1

В соответствии с формулами (5.1.1)–(5.1.3) построены схемы S_j, показанные на рисунке 5.1.1. Слева расположена схема S_j, вычисляющая функции z_j и q_{j+1} при $j = 2, 3, \ldots, n$, справа — схема S_1, вычисляющая функции z_1 и q_2. Используем эти схемы в качестве подсхем при построении схемы Σ_n.

Схема Σ_n состоит из последовательно соединенных подсхем $\mathsf{S}_1, \ldots, \mathsf{S}_n$. Входы подсхемы S_1 подключены к функциям x_1 и y_1. На первом выходе S_1 вычисляется перенос во второй разряд — функция q_2, на втором выходе функция z_1. При $j = 2, 3, \ldots, n$, два входа

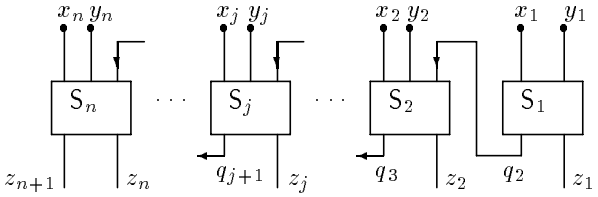


Рис. 5.1.2

подсхемы S_j подключены к функциям x_j, y_j, третий — к вычисляемой подсхемой S_{j-1} функции q_j. На первом выходе S_j вычисляется перенос во $(j+1)$-й разряд — функция q_2, на втором выходе функция z_j. Общий вид схемы Σ_n показан на рисунке 5.1.2.

Сложность подсхемы S_1 равна двум. При $j = 2, 3, \ldots, n$, сложность каждой подсхемы S_j равна пяти. Поэтому

$$L(\Sigma_n) = \sum_{j=1}^{n} \mathsf{S}_j = 5(n-1) + 2 = 5n - 3.$$

Теперь найдем глубину схемы Σ_n. Из конструкции этой схемы (см. рис. 5.1.1 и 5.1.2) видно, что каждая максимальная цепь в Σ_n связывает один из входов x_1 или y_1 с выходом z_{n+1}, и проходит через один элемент подсхемы S_1 и через два элемента каждой подсхемы S_j, при $j > 1$. Следовательно,

$$D(\Sigma_n) = 2(n-1) + 1 = 2n - 1.$$

Лемма доказана.

В дальнейшем, нам потребуется *усеченный сумматор* Σ'_n, складывающий n-разрядные числа не превосходящие 2^{n-1}. Сумматор Σ'_n легко получается из сумматора Σ_n. Достаточно заметить, что если каждое из слагаемых не превосходит 2^{n-1}, то $(n+1)$-й разряд суммы равен единице в единственном случае когда слагаемые в точности равны 2^{n-1}. Следовательно, для вычисления старшего разряда суммы $\boldsymbol{x} + \boldsymbol{y}$ достаточного одного элемента умножения, так как $z_{n+1} = x_n \& y_n$. Поэтому удалив в подсхеме S_n сумматора Σ_n (левая схема на рисунке 5.1.1) нижний элемент сложения и находящийся в центре схемы элемент умножения получим требуемый усеченный сумматор Σ'_n для сложности и глубины которого при $n \geq 2$ справедливы равенства

$$L(\Sigma'_n) = 5n - 5, \quad D(\Sigma'_n) = 2n - 3. \tag{5.1.4}$$

При $n = 1$ схемы Σ_1 и Σ'_1 совпадают.

Пусть $k = \lceil \log_2(n+1) \rceil$. Булев (k,n)-оператор $W(x_1,\ldots,x_n)$ назовем оператором *подсчета*, если

$$W(\alpha_1,\ldots,\alpha_n) = (\beta_1,\ldots,\beta_k),$$

где $\sum_{i=1}^{n} \alpha_i = \sum_{i=1}^{k} 2^{i-1}\beta_i$. Для всякого набора $\boldsymbol{\alpha}$ оператор W вычисляет его вес $\|\boldsymbol{\alpha}\|$. Схему C_n, вычисляющую (k,n)-оператор подсчета назовем *n-счетчиком*.

Лемма 5.1.2. *Существует счетчик C_{2^n}, для сложности и глубины которого справедливы неравенства*

$$L(\mathsf{C}_{2^n}) \le 6 \cdot 2^n, \quad D(\mathsf{C}_{2^n}) \le n^2.$$

Доказательство. Схему C_{2^n} построим в соответствии со следующим алгоритмом. Переменные $x_1,\ldots,x_n$ разобъем на пары и для каждой пары найдем ее сумму используя усеченный сумматор Σ'_1. В результате получится 2^{n-1} двухразрядных чисел, каждое из которых не превосходит 2. Новые числа снова разобьем на пары и для каждой пары найдем ее сумму используя усеченный сумматор Σ'_2. В результате получим 2^{n-2} трехразрядных чисел, каждое из которых не превосходит 4. Подобную процедуру повторим еще $(n-2)$ раза. На i-м шаге будет использовано 2^{n-i} усеченных сумматоров Σ'_i сложность и глубина каждого из которых равны

$$L(\Sigma'_i) = 5i - 5, \quad D(\Sigma'_n) = 2i - 3.$$

Тогда для сложности всей схемы C_{2^n} при $n \ge 2$ получаем

$$\begin{aligned} L(\mathsf{C}_{2^n}) &= \sum_{i=1}^{n-1} 2^{n-i} L(\Sigma'_i) = 2^{n-1}2 + \sum_{i=2}^{n-1} 2^{n-i} 5(i-1) = \\ &= 2^n + 5 \cdot 2^{n-1} \sum_{i=2}^{n-1} \frac{i-1}{2^{i-1}} \le 2^n + 5 \cdot 2^{n-1} \sum_{j=1}^{n-2} \frac{j}{2^j} \le \\ &\le 2^n + 5 \cdot 2^{n-1} \sum_{j=1}^{\infty} \sum_{k=j}^{\infty} \frac{1}{2^k} \le 6 \cdot 2^n. \end{aligned}$$

Аналогичным образом для глубины C_{2^n} при $n \ge 2$ справедливы неравенства

$$\begin{aligned} D(\mathsf{C}_{2^n}) &= \sum_{i=1}^{n-1} D(\Sigma'_i) = 1 + \sum_{i=2}^{n-1} (2i-3) = \\ &= 1 + (n+1)(n-2) - 3(n-2) = 1 + (n-2)^2. \end{aligned}$$

Лемма доказана.

2. Известно, что любая схема, складывающая два n-разрядных двоичных числа, состоит не менее чем из $5n-3$ элементов. Поэтому построенные выше схемы Σ_n являются минимальными по сложности при всех n. В тоже время для больших n глубины этих схем далеки от минимально возможных. Покажем, что существуют сумматоры глубины которых пропорциональны логарифму числа разрядов складываемых чисел.

Докажем вспомогательное утверждение.

Лемма 5.1.3. *Пусть зависящие от переменных $b_1, a_2, b_2, \ldots, a_{2^k}, b_{2^k}$ функции $y_2, y_3, \ldots, y_{2^k+1}$ такие, что*

$$y_2 = b_1, \quad y_{j+1} = b_j \oplus a_j y_j \quad \text{при } j = 2,\ldots,2^k.$$

Тогда существует вычисляющая функции $y_2,\ldots,y_{2^k+1}$ схема P_k, для сложности и глубины которой справедливы соотношения

$$L(\mathsf{P}_k) \le 4 \cdot 2^k, \quad D(\mathsf{P}_k) \le 4k - 2.$$

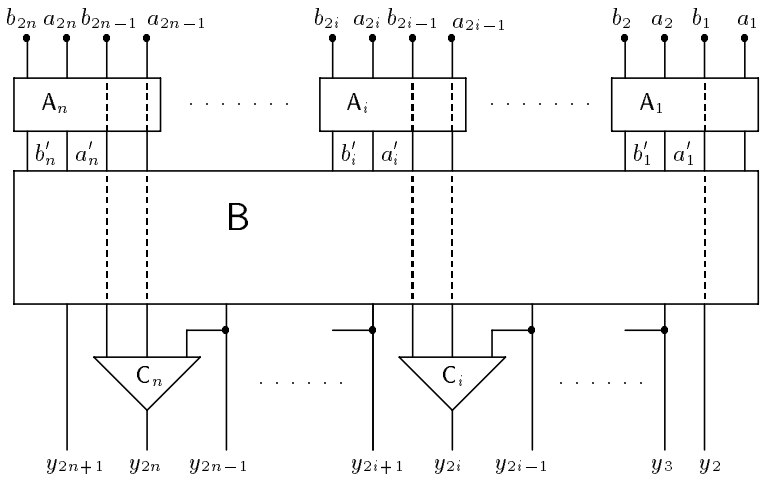


Рис. 5.1.3

Доказательство. Лемму докажем индукцией по k. При $k = 1$ нужно вычислить только y_3. Это можно сделать схемой P_1, состоящей из одного элемента сложения и одного элемента умножения. Очевидно, что $L(\mathsf{P}_1) = 2$ и $D(\mathsf{P}_1) = 2$. Предположим, что при некотором $k \geq 1$ требуемая схема P_k существует. Используя эту схему, построим схему P_{k+1}.

Прежде всего заметим, что при каждом j, $2 \leq j \leq 2^k$, справедливо равенство

$$y_{2j+1} = b_{2j} \oplus a_{2j} y_{2j} = b_{2j} \oplus a_{2j}(b_{2j-1} \oplus a_{2j-1}) y_{2j-1}.$$

Для всех $j \in \{2, \ldots, 2^k\}$ введем новые функции

$$y'_{j+1} = y_{2j+1}, \quad b'_j = b_{2j}, \quad a'_j = a_{2j}(b_{2j-1} \oplus a_{2j-1}).$$

Пусть, кроме того, $y'_2 = b'_1 = b_2 \oplus a_2 b_1$. Тогда новые функции y'_j и новые переменные a'_j и b'_j связаны следующими равенствами:

$$y'_2 = b'_1, \quad y'_{j+1} = b'_j \oplus a'_j y'_j, \quad \text{при } 2 \leq j \leq 2^k. \tag{5.1.5}$$

Воспользуемся этими равенствами для вычисления функций y_j. Сделаем это в три этапа. Сначала вычислим все новые переменные a'_j и b'_j. Затем вычислим что все функции y_j с нечетными индексами. Из равенств (5.1.5), условий леммы и предположения индукции следует, что это можно сделать при помощи схемы P_k, подключив ее входы к вычисленным ранее переменным a'_j и b'_j. Наконец, каждую функцию y_{2j} с четным индексом вычислим по формуле $y_{2j} = b_{2j-1} \oplus a_{2j-1} y_{2j-1}$, используя вычисленную ранее функцию y_{2j-1}.

Выполняющая указанные вычисления схема P_{k+1} изображена на рисунке 5.1.3, где $n = 2^k$. Эта схема состоит из 2^k подсхем A_j, $1 \leq j \leq 2^k$, подсхемы B и $2^k - 1$ подсхем C_j, $2 \leq j \leq 2^k$. Кратко рассмотрим подсхемы P_{k+1} и оценим их сложности и глубины.

1. Подсхема A_1 вычисляет b'_1. Очевидно, что $L(\mathsf{A}_1) = 2$ и $D(\mathsf{A}_1) = 2$. При $j \geq 2$ подсхема A_j вычисляет функцию a'_j. Легко видеть, что $L(\mathsf{A}_j) = 2$ и $D(\mathsf{A}_j) = 2$.

2. Подсхема B является экземпляром схемы P_k. По предположению индукции $L(\mathsf{B}) \leq 4 \cdot 2^k$ и $D(\mathsf{B}) \leq 4k - 2$.

3. Подсхема C_j вычисляет функцию y_{2j} в соответствии с формулой $y_{2j} = b_{2j-1} \oplus a_{2j-1} y_{2j-1}$. Легко видеть, что $L(\mathsf{C}_j) = 2$ и $D(\mathsf{C}_j) = 2$.

Из конструкции схемы P_{k+1}, пп. 1–3 и предположения индукции легко получаем, что

$$L(\mathsf{P}_{k+1}) \leq L(\mathsf{P}_k) + 4 \cdot 2^k - 2 \leq 4 \cdot 2^k + 4 \cdot 2^k - 2 = 4 \cdot 2^{k+1},$$
$$D(\mathsf{P}_{k+1}) \leq D(\mathsf{P}_k) + 4 \leq 4k - 2 + 4 = 4(k+1) - 2.$$

Лемма доказана.

Теорема 5.1.1. *Существует n-разрядный сумматор Σ_n^*, для сложности и глубины которого справедливы неравенства*

$$L(\Sigma_n^*) \le 11n, \quad D(\Sigma_n^*) \le 4\lceil \log_2 n \rceil.$$

Доказательство. Рассмотрим сложение двух целых n-разрядных чисел $\boldsymbol{x}$ и $\boldsymbol{y}$. Для каждого $j \in \{1, \ldots, n\}$ определим функции

$$b_j = x_j y_j, \quad a_j = x_j \oplus y_j.$$

Тогда (см. (5.1.2) на стр. 94) для переноса q_{j+1} в $(j+1)$-й разряд суммы $\boldsymbol{x}+\boldsymbol{y}$ справедлива формула

$$q_{j+1} = x_j y_j \oplus (x_{j-1} \oplus y_{j-1}) q_j = b_j \oplus a_j q_j.$$

Вычислив величины b_j и a_j, для вычисления переносов q_{j+1} воспользуемся схемой $\mathsf{P}_{\lceil \log_2 n \rceil}$ из леммы 5.1.3. Легко видеть, для сложности и глубины схемы Q_n, производящей вычисления всех a_j, b_j и q_{j+1}, справедливы соотношения

$$L(\mathsf{Q}_n) \le 2n + 4 \cdot 2^{\lceil \log_2 n \rceil} \le 10n, \quad D(\mathsf{Q}_n) \le 4\lceil \log_2 n \rceil - 1.$$

Теперь для вычисления суммы $\boldsymbol{x}$ и $\boldsymbol{y}$ достаточно попарно сложить вычисленные схемой Q_n переносы q_j и суммы $x_j \oplus y_j$. Теорема доказана.

3. Разностью двух n-разрядных целых положительных чисел $\boldsymbol{x}$ и $\boldsymbol{y}$, представленных в двоичной системе счисления, назовем такой $(n+1)$-разрядный вектор $\boldsymbol{r}$, что его первые n разрядов образуют число $\boldsymbol{r}_1$, равное модулю разности $\boldsymbol{x}$ и $\boldsymbol{y}$,

$$\boldsymbol{r}_1 = (r_1, \ldots, r_n) = \sum_{i=1}^{n} r_i 2^{i-1} = |\boldsymbol{x} - \boldsymbol{y}|,$$

а его $(n+1)$-й разряд r_{n+1} равен знаку этой разности,

$$r_{n+1} = \begin{cases} 1, & \text{если } \boldsymbol{x} < \boldsymbol{y}, \\ 0, & \text{если } \boldsymbol{x} \ge \boldsymbol{y}. \end{cases}$$

Булев оператор $R_n : \{0,1\}^{2n} \to \{0,1\}^{n+1}$, вычисляющий разность двух n-разрядных целых положительных чисел, назовем оператором *вычитания*.

Опишем простой способ нахождения разности двух произвольных чисел $\boldsymbol{x}$ и $\boldsymbol{y}$. Вместе с числом $\boldsymbol{x} = \sum_{i=1}^{n} x_i 2^{i-1}$ рассмотрим его дополнение $\overline{\boldsymbol{x}} = \sum_{i=1}^{n} \overline{x}_i 2^{i-1}$. Очевидно, что $\boldsymbol{x} + \overline{\boldsymbol{x}} = 2^n - 1$. Следовательно, для оператора суммирования S_n и любых целых n-разрядных чисел $\boldsymbol{x}$ и $\boldsymbol{y}$ выполняется равенство

$$S_n(\overline{\boldsymbol{x}}, \boldsymbol{y}) = 2^n - 1 - \boldsymbol{x} + \boldsymbol{y}.$$

Далее через s_2 будем обозначать $(n+1)$-й разряд числа $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y})$, а через s_1 — число, составленное из младших n разрядов $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y})$, т. е. $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y}) = s_2 2^n + \mathsf{s}_1$. Покажем, что

$$(\mathsf{s}_1 + s_2)^{(s_2)} = |\boldsymbol{x} - \boldsymbol{y}|.$$

Для этого рассмотрим два случая: $s_2 = 1$ и $s_2 = 0$. Если $s_2 = 1$, то $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y}) \ge 2^n$, и, следовательно, $\boldsymbol{x} < \boldsymbol{y}$. В этом случае $\mathsf{s}_1 = -1 - \boldsymbol{x} + \boldsymbol{y}$. Тогда

$$(\mathsf{s}_1 + s_2)^{(s_2)} = \mathsf{s}_1 + s_2 = \mathsf{s}_1 + 1 = -\boldsymbol{x} + \boldsymbol{y} = |\boldsymbol{x} - \boldsymbol{y}|.$$

Если $s_2 = 0$, то $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y}) < 2^n$, и, следовательно, $\boldsymbol{x} \ge \boldsymbol{y}$. В этом случае $\mathsf{s}_1 = 2^n - 1 - \boldsymbol{x} + \boldsymbol{y}$. Тогда

$$(\mathsf{s}_1 + s_2)^{(s_2)} = \overline{\mathsf{s}_1} = 2^n - 1 - (2^n - 1 - \boldsymbol{x} + \boldsymbol{y}) = |\boldsymbol{x} - \boldsymbol{y}|.$$

Таким образом, $(\mathsf{s}_1 + s_2)^{(s_2)} = |\boldsymbol{x} - \boldsymbol{y}|$ и число s_2 определяет знак разности $\boldsymbol{x} - \boldsymbol{y}$: разность отрицательна, если $s_2 = 1$, и неотрицательна, если $s_2 = 0$. Следовательно, пара (s_1, s_2) позволяет легко определить разностью $\boldsymbol{x}$ и $\boldsymbol{y}$.

Теорема 5.1.2. *Существует схема $\overline{\Sigma}_n$, вычисляющая модуль и знак разности двух n-разрядных чисел, для сложности и глубины которой справедливы равенства*

$$L(\overline{\Sigma}_n) = 8n - 3, \quad L(\overline{\Sigma}_n) = 3n - 1.$$

Доказательство. Для доказательства теоремы достаточно построить схему, вычисляющую оператор вычитания R_n. Выше было показано, что при любых $\boldsymbol{x}$ и $\boldsymbol{y}$ значение оператора $R_n(\boldsymbol{x}, \boldsymbol{y})$ легко выражается через разряды числа $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y}) = s_2 2^n + \mathsf{s}_1$ по следующему правилу: к n-разрядному числу s_1 надо прибавить одноразрядное число s_2 и каждый разряд результата сложить с $\overline{s}_2$ по модулю 2. В соответствии с этим правилом построим схему $\overline{\Sigma}_n$. Составим ее из двух последовательно соединенных подсхем A и B. Подсхема A будет вычислять $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y})$, а подсхема B — прибавлять s_2 и складывать каждый разряд результата с $\overline{s}_2$ по модулю 2.

В разделе 4.3 было показано, что если схема в базисе из всех двухместных функций вычисляет функцию не являющуюся отрицанием переменной и содержит инвертор, то этот инвертор может быть удален из схемы так, что вычисляемая схемой функция не изменится. Аналогичное свойство справедливо и для схем вычисляющих системы функций: Если система булевых функций F не содержит отрицаний переменных, то из любой схемы, вычисляющей эту систему в базисе $P_2(2)$, можно удалить все элементы отрицания. Очевидно, что оператор сложения S_n именно такой. Поэтому для вычисления $S_n(\overline{\boldsymbol{x}}, \boldsymbol{y})$ используем преобразованный подходящим образом сумматор Σ_n. Легко видеть, что в этом случае сложность подсхемы A не превосходит $5n - 3$, а глубина — $2n - 1$.

Для прибавления s_2 используем второй сумматор Σ_n. Так как s_2 одноразрядное число, то из каждой подсхемы S_j, $j > 1$, сумматора Σ_n (см. рисунки 5.1.1 и 5.1.2) можно удалить по три элемента. Наконец поразрядное прибавление $\overline{s}_2$ требует не более n элементов. Поэтому, $L(\mathsf{B}) = 3n$ и $D(\mathsf{B}) = n$. Теорема доказана.

Если заранее известно, что разность неотрицательна, то для ее вычисления можно использовать любой сумматор. Достаточно обратить разряды уменьшаемого числа и разряды результата. В частности существует схема $\overline{\Sigma}^*_n$ вычисляющая разность двух целых n-разрядных чисел $\boldsymbol{x}$ и $\boldsymbol{y}$, $\boldsymbol{x} \geq \boldsymbol{y}$, для сложности и глубины которой справедливы соотношения

$$L(\overline{\Sigma}^*_n) \leq 11n, \quad D(\overline{\Sigma}^*_n) \leq 4\lceil \log_2 n \rceil. \tag{5.1.6}$$

Задачи

5.1.1. Построить счетчик C_8.

5.1.2. Построить сумматор Σ^*_8.

5.1.3. Построить схему $\mathsf{S}_{n,m}$, вычисляющую сумму n-разрядного и m-разрядного целых чисел.

5.1.4. Построить схему, увеличивающую n-разрядное число на единицу.

5.1.5. Построить схему, вычисляющую сумму двух целых чисел по модулю 2^n.

5.1.6. Построить схему, вычисляющую сумму двух n-разрядных чисел, каждое из которых задано модулем и знаком.

5.1.7. Построить схему S_n, вычисляющую сумму двух n-разрядных чисел, для которой $L(\mathsf{S}_n) = \mathcal{O}(n)$ и $D(\mathsf{S}_n) \sim 2\log_2 n$.

5.1.8. Построить схему S_n, вычисляющую сумму двух n-разрядных чисел, для которой $L(\mathsf{S}_n) = \mathcal{O}(n)$ и $D(\mathsf{S}_n) \sim \log_2 n$.

5.2. Вычисление суммы нескольких целых чисел

1. Ниже рассматриваются простые неглубокие схемы, вычисляющие сумму большого числа целых положительных чисел, заданных своими двоичными разложениями.

Лемма 5.2.1. *Пусть $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ — произвольные n-разрядные числа, $\boldsymbol{c}$ и $\boldsymbol{r}$ — такие $(n+1)$- и n-разрядное целые, что $\boldsymbol{c} - \boldsymbol{r} = \boldsymbol{x} + \boldsymbol{z} - \boldsymbol{y}$ и, более того,*

$$c_1 = 0, \quad 2c_{i+1} - r_i = x_i + z_i - y_i$$

для каждого $i \in \{1, \ldots, n\}$. Тогда существует вычисляющая $\boldsymbol{c}$ и $\boldsymbol{r}$ схема $\widetilde{\Sigma}_n$ для сложности и глубины которой справедливы равенства

$$L(\widetilde{\Sigma}_n) = 5n, \quad D(\widetilde{\Sigma}_n) = 3.$$

Доказательство. Так как $\boldsymbol{c} + \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{z} + \boldsymbol{r}$ и $c_1 = 0$ и $2c_{i+1} + y_i = x_i + z_i + r_i$ для каждого $i \in \{1, \ldots, n\}$, то легко видеть (см. (5.1.1) и (5.1.2)), что

$$c_{i+1} = x_i z_i \oplus r_i(x_i \oplus z_i), \quad y_i = x_i \oplus z_i \oplus r_i.$$

Из второго равенства находим $r_i = x_i \oplus y_i \oplus z_i$. Подставим r_i в первое равенство:

$$\begin{aligned} c_{i+1} &= x_i z_i \oplus r_i(x_i \oplus z_i) = x_i z_i \oplus (x_i \oplus y_i \oplus z_i)(x_i \oplus z_i) = \\ &= x_i z_i \oplus x_i \oplus z_i \oplus y_i(x_i \oplus z_i) = (x_i \vee z_i) \oplus y_i(x_i \oplus z_i). \end{aligned}$$

Тогда в качестве схемы $\widetilde{\Sigma}_n$ можно взять схему, изображенную на рисунке 5.2.1. Эта схема

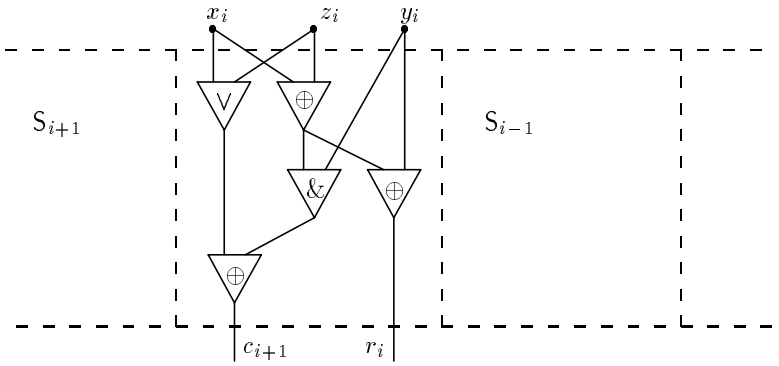


Рис. 5.2.1

состоит из n независимо работающих одинаковых подсхем S_i. Подсхема S_i имеет три входа и два выхода (входы и выходы S_i нумеруются слева направо). Входы S_i подключены к i-м разрядам чисел $\boldsymbol{x}$, $\boldsymbol{z}$ и $\boldsymbol{y}$. На первом выходе S_i вычисляется $(i+1)$-й разряд числа $\boldsymbol{c}$, на втором выходе — i-й разряд числа $\boldsymbol{r}$. Очевидно, что схема на рисунке 5.2.1 состоит из $5n$ элементов, а ее глубина равна трем. Лемма доказана.

Далее выходы $\widetilde{\Sigma}_n$, на которых вычисляются разряды числа $\boldsymbol{c}$, будем называть положительными, а выходы, на которых вычисляются разряды числа $\boldsymbol{r}$, — отрицательными.

Замечание 5.2.1. Схема $\widetilde{\Sigma}_n$ обладает одним замечательным свойством — в этой схеме отсутствуют переносы и при вычислении каждого разряда чисел $\boldsymbol{c}$ и $\boldsymbol{r}$ используется только по одному разряду $\boldsymbol{x}$, $\boldsymbol{y}$ и $\boldsymbol{z}$. Часто это свойство позволяет строить более простые схемы если о складываемых числах доступна какая-либо априорная информация. Рассмотрим, например, $4n$-разрядные целые числа $\boldsymbol{x}$, $\boldsymbol{y}$ и $\boldsymbol{z}$ такие, что $\boldsymbol{x} = 2^{2n}\boldsymbol{x}'$, $\boldsymbol{y} = 2^n\boldsymbol{y}'$ и $\boldsymbol{z} = \boldsymbol{z}'$, где каждое из чисел $\boldsymbol{x}'$, $\boldsymbol{y}'$, $\boldsymbol{z}'$ содержит по $2n$ разрядов. Условно эти числа изображены на рисунке 5.2.2 прямоугольниками. Заштрихованные части прямоугольников соответствуют тем разрядам $\boldsymbol{x}$, $\boldsymbol{y}$ и $\boldsymbol{z}$ которые могут быть отличны от нуля. Все разряды соответствующие не штрихованным областям равны нулю. Легко видеть, что при нахождении суммы $\boldsymbol{x} + \boldsymbol{z} - \boldsymbol{y}$ вычисления достаточно проводить только для средних $2n$ разрядов складываемых чисел (на рисунке эти разряды расположены между двумя вертикальными штриховыми линиями). Это связано с тем, что вычисленные схемой $\widetilde{\Sigma}_{4n}$ старшие n разрядов чисел $\boldsymbol{c}$ и $\boldsymbol{r}$ равны старшим n разрядам числа $\boldsymbol{x}$, младшие n разрядов числа $\boldsymbol{r}$ равны младшим n

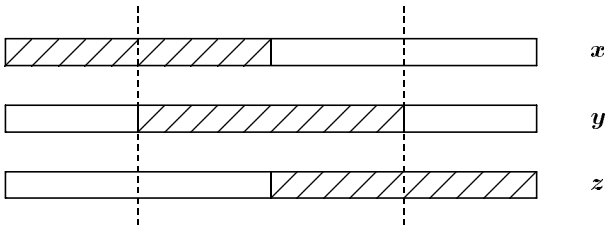


Рис. 5.2.2

разрядам числа $\boldsymbol{z}$, а все младшие n разрядов $\boldsymbol{c}$ нулевые. Поэтому для рассматриваемых $\boldsymbol{x}$, $\boldsymbol{y}$ и $\boldsymbol{z}$ сумму $\boldsymbol{x}+\boldsymbol{z}-\boldsymbol{y}$ можно вычислить схемой $\widetilde{\Sigma}_{2n}$ входы которой подключены к средним разрядам складываемых чисел, т. е. сложность вычисления $\boldsymbol{x}+\boldsymbol{z}-\boldsymbol{y}$ для чисел рассматриваемого вида равна $10n$, а не $20n$ как в общем случае.

Пусть $\boldsymbol{x}_1, \boldsymbol{x}_2$ — n-разрядные двоичные числа. Пару $(\boldsymbol{x}_1, \boldsymbol{x}_2)$ назовем n-разрядным *двойным числом* $\boldsymbol{x}$, значением $\boldsymbol{x}$ назовем разность $\boldsymbol{x}_1 - \boldsymbol{x}_2$. Число $\boldsymbol{x}_1$ называется положительной компонентой $\boldsymbol{x}$, а $\boldsymbol{x}_2$ — отрицательной компонентой $\boldsymbol{x}$. На множестве двойных чисел естественным образом определяются обратное число и сумма двух чисел:

$$-(\boldsymbol{x}_1, \boldsymbol{x}_2) = (\boldsymbol{x}_2, \boldsymbol{x}_1),$$
$$(\boldsymbol{x}_1, \boldsymbol{x}_2) + (\boldsymbol{y}_1, \boldsymbol{y}_2) = (\boldsymbol{p}, \boldsymbol{q}),$$

где $\boldsymbol{p}$ и $\boldsymbol{q}$ такие, что $\boldsymbol{p} - \boldsymbol{q} = (\boldsymbol{x}_1 + \boldsymbol{y}_1) - (\boldsymbol{x}_2 + \boldsymbol{y}_2)$. Заметим, что хотя сумма двух двойных чисел определена не единственным образом (например $(0,0)+(1,0) = ((01),(00)) = ((10),(01))$), значение суммы всегда определено однозначно. Далее, говоря о сумме любого количества двойных чисел будем иметь ввиду не конкретную пару $(\boldsymbol{p}, \boldsymbol{q})$, а целый класс чисел имеющих одно и тоже значение. В частности под вычислением суммы двойных чисел понимается нахождение любого двойного числа значение которого равно сумме значений суммируемых чисел.

Прежде чем рассматривать вопросы о сложности и глубине сложения двойных чисел приведем одно полезное утверждение, являющееся тривиальным следствием доказательства леммы 5.2.1.

Лемма 5.2.2. *Пусть $\boldsymbol{x}, \boldsymbol{y}$ — произвольные n-разрядные числа, $\boldsymbol{c}$ и $\boldsymbol{r}$ — такие $(n+1)$- и n-разрядное целые, что $\boldsymbol{c} - \boldsymbol{r} = \boldsymbol{x} + \boldsymbol{y}$. Существует схема $\Sigma'_{2,n}$ вычисляющая $\boldsymbol{c}$ и $\boldsymbol{r}$ для сложности и глубины которой справедливы равенства*

$$L(\Sigma'_{2,n}) = 2n, \quad D(\Sigma'_{2,n}) = 1.$$

Лемму 5.2.2 можно рассматривать как утверждение о сложности и глубине сложения двух n-разрядных чисел, при условии, что результатом такого сложения будет двойное число.

Лемма 5.2.3. *Для любого $n \geq 1$ существует схема $\Sigma_{2,n}$, вычисляющая сумму двух n-разрядных двойных чисел, для сложности и глубины которой справедливы равенства*

$$L(\Sigma_{2,n}) = 10n - 3, \quad D(\Sigma_{2,n}) = 5.$$

Доказательство. Пусть $(\boldsymbol{x}, \boldsymbol{y})$, $(\boldsymbol{z}, \boldsymbol{w})$ — произвольные n-разрядные двойные числа. Положим $(\boldsymbol{x}, \boldsymbol{y}) + (\boldsymbol{z}, \boldsymbol{w}) = (\boldsymbol{p}, \boldsymbol{q})$. Пусть целые $\boldsymbol{c}$ и $\boldsymbol{r}$ вычислены схемой Σ_n из леммы 5.2.1 при условии, что на ее входы поданы числа $\boldsymbol{x}$, $\boldsymbol{z}$ и $\boldsymbol{y}$. Тогда $\boldsymbol{x} + \boldsymbol{z} - \boldsymbol{y} = \boldsymbol{c} - \boldsymbol{r}$ и

$$\begin{aligned}(\boldsymbol{x}, \boldsymbol{y}) + (\boldsymbol{z}, \boldsymbol{w}) &= (\boldsymbol{x} + \boldsymbol{z} - \boldsymbol{y}) - \boldsymbol{w} = \\ &= (\boldsymbol{c} - \boldsymbol{r}) - \boldsymbol{w} = -(\boldsymbol{r} + \boldsymbol{w} - \boldsymbol{c}).\end{aligned} \tag{5.2.1}$$

Из последнего равенства видно, что сумму $(\boldsymbol{x}, \boldsymbol{y}) + (\boldsymbol{z}, \boldsymbol{w})$ можно вычислить при помощи двух схем $\widetilde{\Sigma}_n$ и $\widetilde{\Sigma}_{n+1}$. Сначала схема $\widetilde{\Sigma}_n$ применяется к числам $\boldsymbol{x}$, $\boldsymbol{z}$ и $\boldsymbol{y}$. В результате

получим $(n+1)$-разрядное число $\boldsymbol{c}$ и n-разрядное число $\boldsymbol{r}$. Затем к числам $\boldsymbol{r}$, $\boldsymbol{w}$ и $\boldsymbol{c}$ применяется схема $\widetilde{\Sigma}_{n+1}$. Причем первые два входа j-й подсхемы схемы $\widetilde{\Sigma}_{n+1}$ подключаются к j-м разрядам чисел $\boldsymbol{r}$ и $\boldsymbol{w}$, а третий вход — к j-му разряду числа $\boldsymbol{c}$. В соответствии с (5.2.1), схема $\widetilde{\Sigma}_{n+1}$ вычислит число $\boldsymbol{r}+\boldsymbol{w}-\boldsymbol{c}=-(\boldsymbol{p},\boldsymbol{q})$. Так как $-(\boldsymbol{p},\boldsymbol{q})=(\boldsymbol{q},\boldsymbol{p})$, то очевидно, что сумма $(\boldsymbol{x},\boldsymbol{y})+(\boldsymbol{z},\boldsymbol{w})$ вычислена: на отрицательных выходах $\widetilde{\Sigma}_{n+1}$ вычисляются разряды числа $\boldsymbol{p}$, на положительных — разряды числа $\boldsymbol{q}$. Схема $\Sigma_{2,n}$ построена.

Из леммы 5.2.1 легко следует, что $\Sigma_{2,n}$ содержит не более $10n+5$ элементов, а ее глубина не превосходит шести. В действительности рассматриваемая схема несколько проще. Заметим, что в схеме $\widetilde{\Sigma}_{n+1}$ в каждой подсхеме S_j глубина третьего входа равна

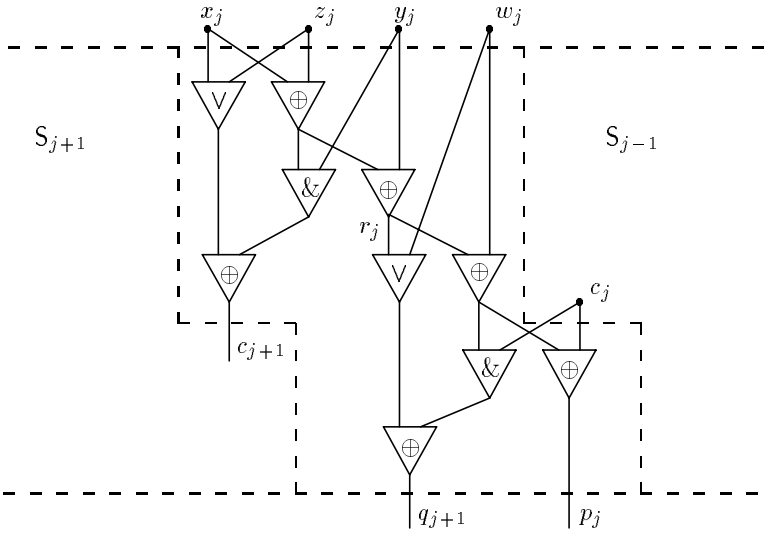


Рис. 5.2.3

двум, а в подсхеме S_{n+1} только один существенный вход. Поэтому схему $\Sigma_{2,n}$ можно представить в виде объединения n одинаковых подсхем S_j, каждая из которых содержит по десять элементов. Конструкция подсхемы S_j изображена на рисунке 5.2.3. Подсхема S_j имеет пять входов и три выхода. Входы S_j подключены к j-м разрядам чисел $\boldsymbol{x}$, $\boldsymbol{z}$, $\boldsymbol{y}$, $\boldsymbol{w}$, и к первому выходу подсхемы S_{j-1}. Второй и третий выходы S_j являются выходами схемы $\Sigma_{2,n}$. На втором выходе S_j вычисляется $(j+1)$-й разряд числа $\boldsymbol{q}$, на третьем выходе — j-й разряд числа $\boldsymbol{p}$. Первый выход подсхемы S_n также является выходом $\Sigma_{2,n}$ и на нем вычисляется $(n+1)$-й разряд числа $\boldsymbol{q}$.

При $j\in\{2,\ldots,n\}$ сложность каждой подсхемы S_j равна десяти, а глубина — пяти. Так как на пятый вход подсхемы S_1 подается тождественный нуль ($c_1\equiv 0$), то легко видеть, что в S_1 три последних элемента можно удалить. Поэтому, сложность схемы $\Sigma_{2,n}$ равна $10n-3$, а глубина такая же как и у подсхем S_j, т. е. пять. Лемма доказана.

Лемма 5.2.4. *Для любого $n\geq 1$ существует схема $\Sigma_{3,n}$, вычисляющая сумму трех n-разрядных двойных чисел, для сложности и глубины которой справедливы равенства*

$$L(\Sigma_{3,n})=20n+7,\quad D(\Sigma_{3,n})=8.$$

Доказательство. Пусть $(\boldsymbol{x}_1,\boldsymbol{x}_2)$, $(\boldsymbol{y}_1,\boldsymbol{y}_2)$ и $(\boldsymbol{z}_1,\boldsymbol{z}_2)$ — произвольные n-разрядные двойные числа. Схему $\Sigma_{3,n}$ составим из двух схем $\widetilde{\Sigma}_n$ и одной схемы $\Sigma_{2,n+1}$. Сначала при помощи схем $\widetilde{\Sigma}_n$ вычислим такие целые $\boldsymbol{p}_1,\boldsymbol{p}_2$ и $\boldsymbol{q}_1,\boldsymbol{q}_2$, что

$$\boldsymbol{p}_1-\boldsymbol{p}_2=\boldsymbol{x}_1+\boldsymbol{y}_1-\boldsymbol{z}_2,\quad \boldsymbol{q}_1-\boldsymbol{q}_2=\boldsymbol{x}_2+\boldsymbol{y}_2-\boldsymbol{z}_1$$

Затем используя схему $\Sigma_{3,n+1}$ вычислим двойное число $\boldsymbol{r}=(\boldsymbol{r}_1,\boldsymbol{r}_2)$, равное сумме двух $(n+1)$-разрядных двойных чисел $(\boldsymbol{p}_1,\boldsymbol{p}_2)$ и $(\boldsymbol{q}_2,\boldsymbol{q}_1)$. Очевидно, что

$$L(\Sigma_{3,n})=2L(\widetilde{\Sigma}_n)+L(\Sigma_{2,n+1})=20n+7,$$
$$D(\Sigma_{3,n})=D(\widetilde{\Sigma}_n)+D(\Sigma_{2,n+1})=8.$$

Лемма доказана.

Теорема 5.2.1. *Для любых $N, n \geq 1$ существует схема $\Sigma_{N,n}$, вычисляющая сумму N n-разрядных двойных чисел, для сложности и глубины которой при $N \to \infty$ справедливы неравенства*

$$L(\Sigma_{N,n}) \lesssim 10N(n+1), \quad D(\Sigma_{N,n}) \leq 5\lceil \log_2 N \rceil.$$

Доказательство. Пусть $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ — произвольные n-разрядные двойные числа. Схему $\Sigma_{N,n}$ построим в соответствии со следующим алгоритмом. Числа $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ разобьем на пары и для каждой пары вычислим ее сумму используя построенные в лемме 5.2.3 схемы $\Sigma_{2,n}$. В результате получится примерно $\frac{1}{2}N$ $(n+1)$-разрядных двойных чисел. Новые числа снова разобьем на пары и для каждой пары вычислим ее сумму и т. д. Будем выполнять итерации до тех пор, пока не останется всего одно число.

Оценим глубину и сложность схемы $\Sigma_{N,n}$. Из теоремы 6.1.2 легко следует, что число итераций не больше $\lceil \log_2 N \rceil$, а так как каждая итерация выполняется схемой глубины пять, то

$$D(\Sigma_{N,n}) \leq 5\lceil \log_2 N \rceil.$$

Теперь оценим сложность схемы $\Sigma_{N,n}$. Положим $R = \lceil \log_2 N \rceil$. Через N_i обозначим количество чисел, остающихся после i-й итерации. Легко видеть, что

$$N_i \leq \frac{1}{2}\Big(N_{i-1} + 1\Big) < \left(\frac{1}{2}\right)^i N + 1.$$

На i-й итерации используется не более N_i схем $\Sigma_{2,n+i-1}$, поэтому

$$\begin{aligned} L(\Sigma_{N,n}) &\leq \sum_{i=1}^{R} 10(n+i-1)N_i \leq 10\sum_{i=1}^{R}(n+i-1)\left(\left(\frac{1}{2}\right)^i N + 1\right) \leq \\ &\leq 10\sum_{i=1}^{R}\left\{\left(\frac{1}{2}\right)^i N(n-1) + \left(\frac{1}{2}\right)^i Ni + (n+i-1)\right\} \leq \\ &\leq 10N(n-1) + 20N + 5R(2n+R). \end{aligned}$$

Следовательно, при $N \to \infty$, для сложности схемы $\Sigma_{N,n}$ справедливо неравенство

$$L(\Sigma_{N,n}) \leq \ 10N(n+1)(1+o(1)).$$

Теорема доказана.

Задачи

5.2.1. Построить схему, преобразующую три целых n-разрядных числа в два числа с такой же суммой.

5.2.2. Построить схему, преобразующую четыре целых n-разрядных числа в два числа с такой же суммой.

5.2.3. Построить схему, вычисляющую сумму трех n-разрядных двойных чисел, глубина которой равна семи.

5.2.4. Показать, что при $n \to \infty$ существует схема, вычисляющая сумму n n-разрядных чисел, глубина которой асимптотически не больше $6,2\log_2 n$.

5.3. Умножение целых чисел

Рассмотрим несколько простых схем умножающих целые двоичные n-разрядные числа.

1. Самые простые по структуре схемы получаются при использовании умножения "в столбик". Сначала первый сомножитель умножается на каждый разряд второго сомножителя. Затем, при помощи сумматоров Σ_k, вычисляется сумма n получившихся чисел.

Сумма вычисляется за $\lceil \log_2 n \rceil$ шагов. На первом шаге вычисляется не более $\frac{n}{2}$ попарных сумм n-разрядных чисел, на втором шаге — не более $\frac{n}{4}$ попарных сумм $(n+2)$-разрядных чисел, и т. д. Поэтому для вычисления суммы потребуется схема состоящая не более чем из

$$\sum_{k=1}^{\lceil \log_2 n \rceil} 5(n+2^{k-1})\frac{n}{2^k} < 5n^2 \sum_{k=1}^{\infty} \frac{1}{2^k} + 5n \sum_{k=1}^{\lceil \log_2 n \rceil} \frac{1}{2} \le 5n^2 + 5n \log_2 n$$

элементов, при этом глубина схемы не превосходит $2n\lceil \log_2 n \rceil$. Таким образом, имеет место следующий результат.

Теорема 5.3.1. *Существует схема M_n, вычисляющая произведение двух n-разрядных чисел, для сложности и глубины которой при $n \to \infty$ справедливы неравенства*

$$L(\mathsf{M}_n) \lesssim 6n^2, \quad D(\mathsf{M}_n) \lesssim 2n \log_2 n.$$

В дальнейшем нам потребуется схема M_4. Поэтому рассмотрим ее более подробно. Пусть M_4 умножает $\boldsymbol{x} = (x_1, x_2, x_3, x_4)$ и $\boldsymbol{y} = (y_1, y_2, y_3, y_4)$. В M_4 для умножения $\boldsymbol{x}$ на разряды y_i достаточно использовать 16 элементов конъюнкции. Затем вычисляются суммы $z_1 = \boldsymbol{x}y_1 + 2\boldsymbol{x}y_2$ и $z_2 = \boldsymbol{x}y_3 + 2\boldsymbol{x}y_4$. Вычисление каждой суммы можно рассматривать как сложение одного 4-разрядного и одного 3-разрядного чисел. Поэтому каждая сумма может быть вычислена схемой из 16 элементов глубины 6. Наконец вычисляется сумма $4z_2 + z_1$. Ее можно рассматривать как сумму 5-разрядного и 3-разрядного чисел. Поэтому она может быть вычислена схемой из 18 элементов глубины 7. Таким образом, для сложности и глубины M_4 выполняются равенства

$$L(\mathsf{M}_4) = 66, \quad D(\mathsf{M}_4) = 14.$$

Теперь рассмотрим схемы, умножающие двойные числа. Аналог теоремы 5.3.1 для двойных чисел легко следует из теоремы 5.2.1. Справедливо следующее утверждение.

Теорема 5.3.2. *Существует схема M'_n, вычисляющая произведение двух n-разрядных двойных чисел, для сложности и глубины которой при $n \to \infty$ справедливы неравенства*

$$L(\mathsf{M}'_n) \lesssim 44n^2, \quad D(\mathsf{M}'_n) \lesssim 5 \log_2 n.$$

Так же без доказательства приведем простое утверждение о сложности умножения обычных целых чисел.

Теорема 5.3.3. *Существует схема $\widetilde{\mathsf{M}}_n$, вычисляющая произведение двух n-разрядных целых чисел, для сложности и глубины которой при $n \to \infty$ справедливы неравенства*

$$L(\widetilde{\mathsf{M}}_n) \lesssim 7n^2, \quad D(\widetilde{\mathsf{M}}_n) \lesssim 9 \log_2 n.$$

Как и предыдущие схемы, схема $\widetilde{\mathsf{M}}_n$ основана на умножении "в столбик" и легко строится из схем $\Sigma'_{2,n}$, $\Sigma_{2,k}$ и Σ^*_{2n}.

2. Сложность всех рассмотренных выше схем умножения n-разрядных чисел пропорциональна n^2. В настоящее время разработаны различные алгоритмы умножения целых чисел, позволяющие строить значительно более экономные схемы. Наиболее простые из этих схем состоят из $\mathcal{O}(n \log_2 n \cdot \log_2 \log_2 n)$ элементов, а их глубина пропорциональна $\log_2 n$. Вместе с тем, конструкции таких схем достаточно сложны. Поэтому ниже рассмотрим самую простую конструкцию, позволяющую строить относительно неглубокие схемы, состоящие менее чем из n^2 элементов. Покажем, что имеет место следующая теорема.

Теорема 5.3.4. *Пусть $n = 2^k + 2$. Тогда при $n \to \infty$ существует схема $\mathsf{M}^\star_n$, вычисляющая произведение двух n-разрядных целых чисел, для сложности и глубины которой справедливы неравенства*

$$L(\mathsf{M}^\star_n) \lesssim 95 n^{\log_2 3}, \quad D(\mathsf{M}^\star_n) \lesssim 17 \log_2 n.$$

Сформулированная теорема является простым следствием доказываемой ниже леммы 5.3.1 об умножении двойных чисел. Построение схем, умножающих двойные числа, начнем с простого частного случая — построим схему M_4^*, умножающую два 4-разрядных двойные числа $\boldsymbol{x}$ и $\boldsymbol{y}$.

Пусть $\boldsymbol{x}_1$ и $\boldsymbol{y}_1$ положительные, а $\boldsymbol{x}_2$ и $\boldsymbol{y}_2$ отрицательные компоненты сомножителей. Произведение $\boldsymbol{xy}$ вычислим следующим образом.

1. При помощи двух схем $\overline{\Sigma}_4$ (стр. 99) вычислим модули разностей $\boldsymbol{r}_x = |\boldsymbol{x}_1 - \boldsymbol{x}_2|$ и $\boldsymbol{r}_y = |\boldsymbol{y}_1 - \boldsymbol{y}_2|$, и их знаки s_x и s_y.

2. При помощи схемы M_4 найдем произведение $\boldsymbol{p} = \boldsymbol{r}_x\boldsymbol{r}_y$ модулей разностей.

3. Положительную $\boldsymbol{p}_1$ и отрицательную $\boldsymbol{p}_2$ компоненты произведения $\boldsymbol{xy}$ вычислим по формулам:

$$\boldsymbol{p}_1 = (s_x \sim s_y) \cdot \boldsymbol{p}, \quad \boldsymbol{p}_2 = (s_x \oplus s_y) \cdot \boldsymbol{p}.$$

Легко видеть, что

$$\begin{aligned} L(\mathsf{M}_4^*) &\le 2L(\overline{\Sigma}_4) + L(\mathsf{M}_4) + 17 = 141, \\ D(\mathsf{M}_4^*) &\le D(\overline{\Sigma}_4) + D(\mathsf{M}_4) + 2 = 27. \end{aligned} \tag{5.3.1}$$

Лемма 5.3.1. *Пусть $n = 2^k + 2$. Тогда при любом $k \ge 1$ существует схема M_n^*, вычисляющая произведение двух n-разрядных двойных чисел, для сложности и глубины которой справедливы неравенства*

$$\begin{aligned} L(\mathsf{M}_{2^k+2}^*) &\le 95 \cdot 3^k - 90 \cdot 2^k + 41, \\ D(\mathsf{M}_{2^k+2}^*) &\le 13k + 14. \end{aligned} \tag{5.3.2}$$

Доказательство. Лемму докажем индукцией по k. В основание индукции положим построенную выше схему M_4^*, умножающую 4-разрядные двойные числа. Легко видеть, что величины из (5.3.1) удовлетворяют неравенствам (5.3.2) для $k = 1$. Покажем, что из справедливости (5.3.2) при некотором $k \ge 1$ следует их справедливость при $k + 1$.

Пусть $n = 2^k + 2$, $\boldsymbol{x}$ и $\boldsymbol{y}$— произвольные $(2n-2)$-разрядные двойные числа. Представим их в виде

$$\boldsymbol{x} = \boldsymbol{x}_2 2^{n-1} + \boldsymbol{x}_1, \quad \boldsymbol{y} = \boldsymbol{y}_2 2^{n-1} + \boldsymbol{y}_1,$$

где каждое из чисел $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, $\boldsymbol{y}_1$, $\boldsymbol{y}_2$ состоит не более чем из $n-1$ разрядов. Тогда

$$\boldsymbol{xy} = \boldsymbol{x}_2\boldsymbol{y}_2 2^{2n-2} + (\boldsymbol{x}_2\boldsymbol{y}_1 + \boldsymbol{x}_1\boldsymbol{y}_2)2^{n-1} + \boldsymbol{x}_1\boldsymbol{y}_1.$$

Откуда после несложных преобразований для произведения $\boldsymbol{xy}$ получаем равенство

$$\boldsymbol{xy} = \boldsymbol{x}_2\boldsymbol{y}_2 2^{2n-2} + (\boldsymbol{x}_2\boldsymbol{y}_2 + \boldsymbol{x}_1\boldsymbol{y}_1)2^{n-1} - (\boldsymbol{x}_2 - \boldsymbol{x}_1)(\boldsymbol{y}_2 - \boldsymbol{y}_1)2^{n-1} + \boldsymbol{x}_1\boldsymbol{y}_1. \tag{5.3.3}$$

Следовательно, умножение двух $(2n-2)$-разрядных чисел сводится к двум умножениям $(n-1)$-разрядных чисел, одному умножению n-разрядных чисел и нескольким сложениям.

Рекурсивная конструкция схемы M_{2n-2}^* показана на рисунке 5.3.1. Полагаем, что в этой схеме подсхемы перенумерованы целыми числами начиная с единицы так, что первая подсхема находится в левом верхнем углу и нумерация продолжается слева направо – сверху вниз.

1. Подсхема S_1 является экземпляром схемы M_n^* и вычисляет произведение $\boldsymbol{z}_1 = \boldsymbol{x}_2\boldsymbol{y}_2$ двух $(n-1)$-разрядных чисел. Следовательно,

$$L(\mathsf{S}_1) = L(\mathsf{M}_n^*), \qquad D(\mathsf{S}_1) = D(\mathsf{M}_n^*).$$

2. Подсхема S_2 является экземпляром схемы M_n^* и вычисляет два $(n-1)$-разрядных числа $\boldsymbol{p}_2$ и $\boldsymbol{p}_1$ таких, что $\boldsymbol{z}_2 = \boldsymbol{p}_2 2^{n-1} + \boldsymbol{p}_1 = \boldsymbol{x}_1\boldsymbol{y}_1$. Следовательно,

$$L(\mathsf{S}_2) = L(\mathsf{M}_n^*), \quad D(\mathsf{S}_2) = D(\mathsf{M}_n^*).$$

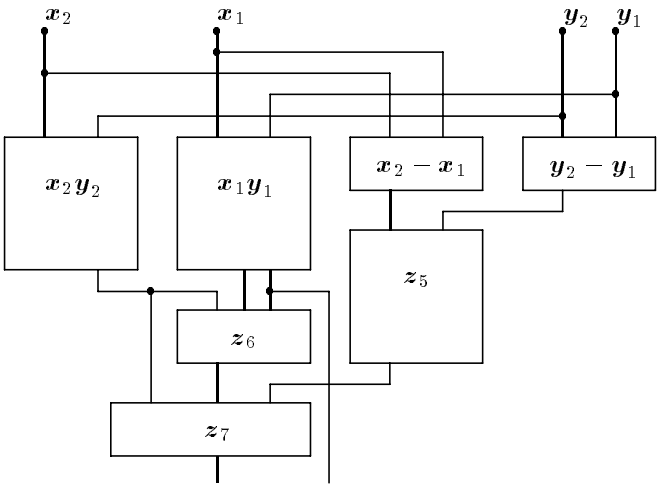


Рис. 5.3.1

3. Подсхема S_3 является экземпляром схемы $\Sigma_{2,n-1}$ и вычисляет разность $\boldsymbol{z}_3 = \boldsymbol{x}_2 - \boldsymbol{x}_1$ двух $(n-1)$-разрядных чисел. Следовательно,

$$L(\mathsf{S}_3) \le 10n - 13, \quad D(\mathsf{S}_3) = 5.$$

4. Подсхема S_4 является экземпляром схемы $\Sigma_{2,n-1}$ и вычисляет разность $\boldsymbol{z}_4 = \boldsymbol{y}_2 - \boldsymbol{y}_1$ двух $(n-1)$-разрядных чисел. Следовательно,

$$L(\mathsf{S}_4) \le 10n - 13, \quad D(\mathsf{S}_4) = 5.$$

5. Подсхема S_5 является экземпляром схемы M^*_n и вычисляет произведение $\boldsymbol{z}_5 = \boldsymbol{z}_3\boldsymbol{z}_4$ двух n-разрядных чисел. Следовательно,

$$L(\mathsf{S}_5) = L(\mathsf{M}^*_n), \quad D(\mathsf{S}_5) = D(\mathsf{M}^*_n).$$

6. Подсхема S_6 является экземпляром схемы $\Sigma_{2,2n-2}$ и вычисляет сумму $\boldsymbol{z}_6 = \boldsymbol{z}_1 + \boldsymbol{z}_2$ двух $(2n-2)$-разрядных чисел. Следовательно,

$$L(\mathsf{S}_6) \le 20n - 23, \quad D(\mathsf{S}_6) = 5.$$

7. Подсхема S_7 является экземпляром схемы $\Sigma_{3,3n-3}$ и вычисляет сумму $\boldsymbol{z}_7$ трех чисел: $(3n-3)$-разрядного числа $(\boldsymbol{z}_1 2^{n-1} + \boldsymbol{p}_2)$, получающегося "бесплатно" из чисел $\boldsymbol{z}_1 2^{n-1}$ и $\boldsymbol{p}_2$ ненулевые разряды которых не пересекаются; $2n$-разрядного числа $-\boldsymbol{z}_5$; $(2n-1)$-разрядного числа $\boldsymbol{z}_6$. Учитывая замечание 5.2.1 на странице 100 легко видеть, что

$$L(\mathsf{S}_7) \le 50n - 33, \quad D(\mathsf{S}_7) = 8.$$

Суммируя сложности всех подсхем, видим, что имеет место рекуррентное неравенство

$$L(\mathsf{M}^*_{2n-2}) = \sum_{i=1}^{7} L(\mathsf{S}_i) \le 3L(\mathsf{M}^*_n) + 90n - 82.$$

Из этого неравенства и предположения индукции получаем

$$\begin{aligned} L(\mathsf{M}^*_{2^{k+1}+2}) &\le 3\left(95 \cdot 3^k - 90 \cdot 2^k + 41\right) + 90 \cdot 2^k - 82 = \\ &= 95 \cdot 3^{k+1} - 90 \cdot 2^{k+1} + 41. \end{aligned}$$

Для глубины схемы $\mathsf{M}^*_{2^{k+1}+2}$ из ее конструкции и предположения индукции легко следует, что

$$D(\mathsf{M}^*_{2^{k+1}+2}) \le D(\mathsf{M}^*_{2^k+2}) + 13,$$

Откуда после простых преобразований получаем второе неравенство леммы. Лемма доказана.

Задачи

5.3.1. Построить схему умножения 6-разрядных двоичных чисел с как можно меньшей глубиной.

5.3.2. Построить схему $\mathsf{M}_{n,m}$, вычисляющую произведение n-разрядного и m-разрядного целых чисел.

5.3.3. Построить схему, вычисляющую произведение двух целых чисел по модулю 2^n.

5.3.4. Построить схему, вычисляющую произведение двух целых чисел по модулю $2^n - 1$.

5.3.5. Построить схему, вычисляющую квадрат целого n-разрядного числа.

5.3.6. Показать, что существует схема умножения двух 4-разрядных чисел глубина которой не превосходит 9.

5.3.7. Показать, что при $n \to \infty$ существует схема, вычисляющая произведение двух n-разрядных целых чисел, сложность которой не превосходит $120n^{\log_2 3}$.

5.4. Сортировка

Пусть $\boldsymbol{x} = (x_1, \ldots, x_n)$ — набор действительных чисел. *Сортировкой* набора $\boldsymbol{x}$ называется перестановка его разрядов в порядке невозрастания их величин. Сортировка встречается в качестве составной части большого числа разнообразных алгоритмов и является одной из наиболее важных комбинаторных задач. В этом параграфе будут построены схемы, сортирующие булевы наборы и имеющие небольшие сложности и глубины. Затем будет показано, что построенные схемы могут быть использованы для сортировки не только булевых наборов, но и наборов действительных чисел.

1. Набор $\boldsymbol{\alpha}$ из $\mathbb{B}^n$ называется *упорядоченным* если $\alpha_i \le \alpha_j$ для всех $1 \le i < j \le n$. Схема в базисе $\{\vee, \&\}$ с n входами и n выходами называется схемой *двоичной сортировки* или *сортирующей схемой*, если она преобразует произвольный набор в упорядоченный набор такого же веса. В следующей теореме приводится конструкция эффективных сортирующих схем. Эти схемы строятся из двухэлементных подсхем с двумя входами и двумя выходами. На первом выходе каждой подсхемы вычисляется конъюнкция ее входов, а на втором выходе — дизъюнкция входов. Такие двухэлементные схемы (см. рисунок 5.4.1) будем называть *булевыми компараторами*. Легко видеть, что компаратор является схемой, сортирующей наборы длины два.

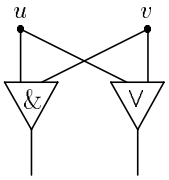


Рис. 5.4.1

Теорема 5.4.1. *Существует схема S_{2^k}, сортирующая наборы длины 2^k, для сложности и глубины которой справедливы равенства*

$$L(\mathsf{S}_{2^k}) = k(k-1)2^{k-2} + 2^k - 1, \qquad D(\mathsf{S}_{2^k}) = \frac{1}{2}k(k+1).$$

Доказательство. Сначала построим схему $\mathsf{S}_{2n,2n}$, объединяющую два упорядоченных набора $(u_1, \ldots, u_{2n})$ и $(v_1, \ldots, v_{2n})$ в один упорядоченный набор $(w_1, \ldots, w_{4n})$. Схема $\mathsf{S}_{2n,2n}$ называется $(2n, 2n)$-схемой нечетно-четного слияния и строится индуктивно. В основании индукции лежит схема $\mathsf{S}_{1,1}$, упорядочивающая два одноэлементных набора и состоящая из единственного компаратора. Очевидно, что

$$L(\mathsf{S}_{1,1}) = 2, \qquad D(\mathsf{S}_{1,1}) = 1. \tag{5.4.1}$$

Предположим, что схема $\mathsf{S}_{n,n}$ построена. Тогда схема $\mathsf{S}_{2n,2n}$, конструкция которой представлена на рисунке 5.4.2, строится следующим образом.

1. Из элементов с нечетными номерами составляются два упорядоченных набора $(u_1, u_3, \ldots, u_{2n-1})$ и $(v_1, v_3, \ldots, v_{2n-1})$, которые сливаются схемой $\mathsf{S}_{n,n}$ в упорядоченный набор $(p_1, \ldots, p_{2n})$.

2. Из элементов с четными номерами составляются два упорядоченных набора $(u_2, u_4, \ldots, u_{2n})$ и $(v_2, v_4, \ldots, v_{2n})$, которые сливаются в упорядоченный набор $(q_1, \ldots, q_{2n})$ схемой $\mathsf{S}_{n,n}$.

3. Наборы $(p_1, \ldots, p_{2n})$ и $(q_1, \ldots, q_{2n})$ преобразуются в упорядоченный набор $(w_1, \ldots, w_{4n})$ по формулам $w_1 = p_1$, $w_{2i} = p_{i+1} \& q_i$, $w_{2i+1} = p_{i+1} \vee q_i$ для $i = 1, 2, \ldots, 2n-1$, и $w_{4m} = q_{2m}$.

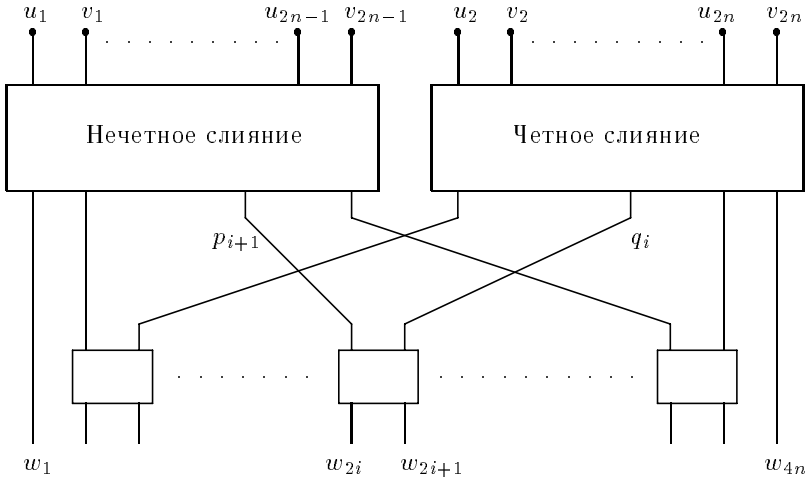


Рис. 5.4.2

Покажем, что схема $\mathsf{S}_{2n,2n}$ действительно преобразует два упорядоченных n-элементных набора в упорядоченный набор. Допустим, что набор $\boldsymbol{u} = (u_1, \ldots, u_{2n})$ состоит из k нулей и $2n-k$ единиц, а набор $\boldsymbol{v} = (v_1, \ldots, v_{2n})$ — из l нулей и $2n-l$ единиц. Тогда в наборе $(p_1, \ldots, p_{2n})$, получившемся после слияния нечетных элементов наборов $\boldsymbol{u}$ и $\boldsymbol{v}$, будет $t = \lceil k/2 \rceil + \lceil l/2 \rceil$ нулей и $2n - t$ единиц. Так же легко видеть, что набор $(q_1, \ldots, q_{2n})$, получившейся после слияния четных элементов наборов $\boldsymbol{u}$ и $\boldsymbol{v}$, будет состоять из $s = \lfloor k/2 \rfloor + \lfloor l/2 \rfloor$ нулей и $2n - s$ единиц. Так как для любого x разность $\lceil x \rceil - \lfloor x \rfloor$ может быть равна только нулю или единице, то величина

$$R = (\lceil k/2 \rceil + \lceil l/2 \rceil) - (\lfloor k/2 \rfloor + \lfloor l/2 \rfloor)$$

может принимать только три значения: 0, 1 и 2.

Если $R = 0$ или $R = 1$, то набор $(p_1, q_1, \ldots, p_{2n}, q_{2n})$ будет упорядоченным. Если $R = 2$, то в этом наборе на первых $k + l - 1$ местах будут стоять нули, на $(k+l)$-м месте будет стоять единица, на $(k+l+1)$-м месте будет стоять последний нуль, и на оставшихся местах — единицы. Из равенства $R = 2$ легко следует, что числа k и l нечетные, и, следовательно, $k + l = 2h$ — четное число. Тогда,

$$w_{2h} = p_{h+1} \& q_h = q_h = 0, \qquad w_{2h+1} = p_{h+1} \vee q_h = p_{h+1} = 1,$$

т.е. один из компараторов, находящихся в последнем ряду схемы, поменяет местами последний нуль и первую единицу набора $(p_1, q_1, \ldots, p_{2n}, q_{2n})$. Легко видеть, что после этого набор станет упорядоченным. Следовательно, схема $\mathsf{S}_{2n,2n}$ действительно объединяет два упорядоченных набора в упорядоченный набор.

Теперь оценим сложность и глубину этой схемы. Из конструкции схемы имеем

$$L(\mathsf{S}_{2n,2n}) = 2L(\mathsf{S}_{n,n}) + (4n-2), \tag{5.4.2}$$

$$D(\mathsf{S}_{2n,2n}) = D(\mathsf{S}_{n,n}) + 1. \tag{5.4.3}$$

Индукцией по k покажем, что при $k \geq 0$ для сложности схемы $\mathsf{S}_{2^k,2^k}$ справедливо равенство

$$L(\mathsf{S}_{2^k,2^k}) = k2^{k+1} + 2. \tag{5.4.4}$$

Действительно, при $k = 0$ равенство (5.4.4) следует из (5.4.1). Допустим, что (5.4.4) верно при всех $k \le m-1$. Тогда из этого предположения и равенства (5.4.2) имеем

$$\begin{aligned} L(\mathrm{S}_{2^m,2^m}) &= 2L(\mathrm{S}_{2^{m-1},2^{m-1}}) + 2\cdot 2^m - 2 = \\ &= 2((m-1)2^m + 2) + 2\cdot 2^m - 2 = m2^{m+1} + 2. \end{aligned}$$

Следовательно, (5.4.4) справедливо при всех целых $k \ge 0$.

Аналогичным образом, из (5.4.3) и (5.4.1) при всех целых $k \ge 0$ для глубины $\mathrm{S}_{2^k,2^k}$ имеем

$$D(\mathrm{S}_{2^k,2^k}) = k + 1. \tag{5.4.5}$$

Теперь, также индуктивно, построим схему S_{4n}, сортирующую наборы из $4n$ элементов. В основание индукции положим схему S_2, сортирующую двухэлементные наборы и состоящую из одного компаратора. Очевидно, что

$$L(\mathrm{S}_2) = 2, \qquad D(\mathrm{S}_2) = 1. \tag{5.4.6}$$

Допустим, что схема S_{2n} построена. Тогда схему S_{4n} составим из двух схем, сортирующих $2n$-элементные наборы, и одной схемы нечетно-четного слияния двух $2n$-элементных наборов. Конструкция схемы представлена на рисунке 5.4.3.

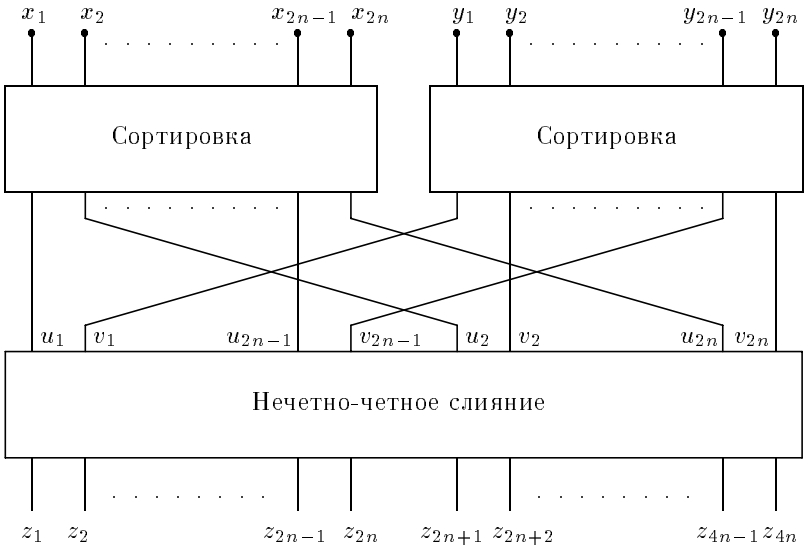


Рис. 5.4.3

Оценим сложность и глубину этой схемы. Из конструкции схемы имеем

$$L(\mathrm{S}_{4n}) = 2L(\mathrm{S}_{2n}) + L(\mathrm{S}_{2n,2n}), \tag{5.4.7}$$

$$D(\mathrm{S}_{4n}) = D(\mathrm{S}_{2n}) + D(\mathrm{S}_{2n,2n}). \tag{5.4.8}$$

Индукцией по k покажем, что для сложности схемы S_{2^k} при всех $k \ge 1$ справедливо равенство

$$L(\mathrm{S}_{2^k}) = k(k-1)2^{k-1} + 2^{k+1} - 2. \tag{5.4.9}$$

При $k = 1$ равенство (5.4.9) следует из (5.4.6). Допустим, что оно верно при всех $k \le m-1$. Тогда из предположения индукции, равенства (5.4.7) и равенства (5.4.4) имеем

$$\begin{aligned} L(\mathrm{S}_{2^m}) &= 2L(\mathrm{S}_{2^{m-1}}) + L(\mathrm{S}_{2^{m-1},2^{m-1}}) = \\ &= 2((m-1)(m-2)2^{m-2} + 2^m - 2) + (m-1)2^m + 2 = \\ &= m(m-1)2^{m-1} + 2^{m+1} - 2. \end{aligned}$$

Следовательно, (5.4.9) справедливо при всех целых $k \geq 1$. Из (5.4.5) и (5.4.8) для глубины $\mathsf{S}_{2^k,2^k}$ имеем

$$D(\mathsf{S}_{2^k}) = D(\mathsf{S}_{2^{k-1}}) + k = \sum_{j=1}^{k} j = \frac{1}{2}k(k+1).$$

Теорема доказана.

Построенные в доказательстве теоремы 5.4.1 сортирующие схемы были предложены в 1968 году Бэтчером, и называются теперь *схемами Бэтчера*.

2. Построенные в предыдущей теореме схемы можно использовать для сортировки наборов действительных чисел. Для этого в схеме S_{2^k} элементы дизъюнкции надо заменить элементами вычисления максимума, а элементы конъюнкции — элементами вычисления минимума. То, что преобразованная схема будет сортировать наборы действительных чисел вытекает из следующей теоремы.

Теорема 5.4.2. *Схема* S *с* n *входами не является сортирующей схемой только в том случае, когда существует булев набор длины* n*, который не может быть отсортирован этой схемой.*

Доказательство. Пусть схема S преобразует последовательность $u_1, \ldots, u_n$ в последовательность $v_1, \ldots, v_n$. Пусть $f : \mathbb{R} \to \mathbb{R}$ — произвольная монотонная функция. Индукцией по числу компараторов легко показать, что схема S преобразует последовательность $f(u_1), \ldots, f(u_n)$ в последовательность $f(v_1), \ldots, f(v_n)$. Предположим, что последовательность $v_1, \ldots, v_n$ не является упорядоченной. Тогда найдется такое i, что $v_{i+1} < v_i$. Функцию f определим следующим образом:

$$f(x) = \begin{cases} 0, & \text{если } x \leq v_{i+1}, \\ 1, & \text{если } x > v_{i+1}. \end{cases}$$

Легко видеть, что в этом случае двоичная последовательность $f(u_1), \ldots, f(u_n)$ будет преобразована схемой S в неупорядоченную последовательность $f(v_1), \ldots, f(v_n)$. Теорема доказана.

Задачи

5.4.1. Построить схему Бэтчера для сортировки наборов длины восемь.

5.4.2. Построить схему, сортирующую наборы длины n, если:
a) $n = 5$; b) $n = 6$; c) $n = 7$.

5.5. Сложность вычисление коэффициентов алгебраической нормальной формы

1. Напомним, в параграфе 3.2 было показано, что вычисление коэффициентов АНФ произвольной булевой функции, зависящей от n переменных, сводится к умножению матрицы $\mathbf{P}_n$ на вектор значений этой функции. При этом матрицы $\mathbf{P}_n$ удовлетворяют соотношениям: справедливо представление

$$\mathbf{P}_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, \quad \mathbf{P}_n = \begin{pmatrix} \mathbf{P}_{n-1} & \mathbf{0} \\ \mathbf{P}_{n-1} & \mathbf{P}_{n-1} \end{pmatrix}, \quad n \geq 2. \tag{5.5.1}$$

Отметим, что обратное преобразование выполняется также при помощи матрицы $\mathbf{P}_n$: умножение этой матрицы на вектор коэффициентов АНФ функции f дает вектор значений f. Таким образом, задача вычисления коэффициентов АНФ сводится к построению схемы, вычисляющей произведение матрицы $\mathbf{P}_n$ и вектора значений функции.

Теорема 5.5.1. *Существует схема* S_n, *состоящая из элементов умножения и сложения по модулю два, умножающая матрицу* $\mathbf{P}_n$ *на произвольный булев вектор длины* 2^n, *сложность и глубина которой удовлетворяют равенствам*:

$$L(\mathsf{S}_n) = n2^{n-1}, \qquad D(\mathsf{S}_n) = n.$$

Доказательство. Схемы S_n, сложности и глубины которых удовлетворяют условиям теоремы построим индукцией по n. Схема S_1 строится тривиально, она состоит из одного элемента сложения. Предположим, что схема S_{n-1} построена. Воспользуемся этой схемой для построения схемы S_n, умножающей матрицу $\mathbf{P}_n$ на вектор $\boldsymbol{x} = (x_0, \ldots, x_{2^n-1})$. Положим $\boldsymbol{x}_1 = (x_0, \ldots, x_{2^{n-1}-1})$ и $\boldsymbol{x}_2 = (x_{2^{n-1}}, \ldots, x_{2^n-1})$. Тогда из (5.5.1) видим, что

$$\begin{pmatrix} \mathbf{P}_{n-1} & \mathbf{0} \\ \mathbf{P}_{n-1} & \mathbf{P}_{n-1} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{P}_{n-1}\boldsymbol{x}_1 \\ \mathbf{P}_{n-1}\boldsymbol{x}_1 \oplus \mathbf{P}_{n-1}\boldsymbol{x}_2 \end{pmatrix},$$

т.е. вычисление произведения $\mathbf{P}_n\boldsymbol{x}$ сводится к вычислению двух произведений $\mathbf{P}_{n-1}\boldsymbol{x}_1$ и $\mathbf{P}_{n-1}\boldsymbol{x}_2$ и к последующему сложению двух наборов длины 2^{n-1}. Поэтому схему S_n построим так, как это показано на рисунке 5.5.1. Схема S_n состоит из двух экземпляров схемы

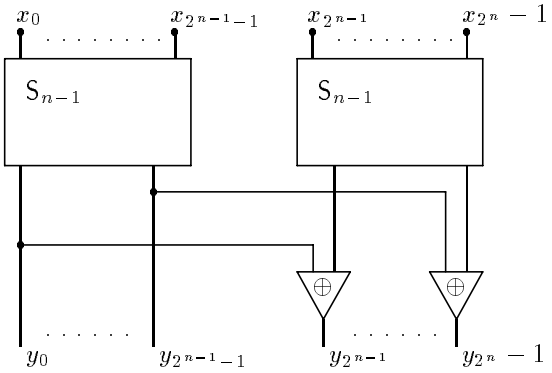


Рис. 5.5.1

S_{n-1} и 2^{n-1} элементов сложения. Легко видеть, что для сложности схемы S_n справедливы равенства

$$\begin{aligned} L(\mathsf{S}_n) &= 2L(\mathsf{S}_{n-1}) + 2^{n-1} = 2\left(2L(\mathsf{S}_{n-2}) + 2^{n-2}\right) + 2^{n-1} = \\ &= 4L(\mathsf{S}_{n-2}) + 2 \cdot 2^{n-1} = \ldots = 2^{n-1}L(\mathsf{S}_1) + (n-1)2^{n-1} = n2^{n-1}. \end{aligned}$$

Для глубины S_n из рис. 5.5.1 видим, что

$$D(\mathsf{S}_n) = D(\mathsf{S}_{n-1}) + 1 = n.$$

Теорема доказана.

Задачи

5.5.1. Построить схему, вычисляющую коэффициенты АНФ булевой функции трех переменных по ее значениям.

5.5.2. Пусть f и g — булевы функции n переменных. Оценить сложность вычисления коэффициентов АНФ функции fg, если известны коэффициенты АНФ функций f и g.

5.6. Вычисление преобразования Фурье

Напомним, что преобразованием Фурье 1-го типа булевой функции $f(\boldsymbol{x})$ называется действительнозначная функция $F(\boldsymbol{u})$, вычисляемая по формуле

$$F(\boldsymbol{u}) = \sum_{\boldsymbol{x} \in \mathbb{B}^n} (-1)^{(\boldsymbol{u},\boldsymbol{x})} f(\boldsymbol{x}), \qquad \boldsymbol{u} \in \mathbb{B}^n. \tag{5.6.1}$$

Непосредственно из (5.6.1) следует, что преобразование Фурье 1-го типа булевой функции $f(\boldsymbol{x})$ можно найти вычислив произведение $\mathbf{H}_n\boldsymbol{f}$ вектора значений $\boldsymbol{f}$ функции f и матрицы Адамара $\mathbf{H}_n$, задаваемой рекуррентными равенствами

$$\mathbf{H}_1 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad \mathbf{H}_n = \begin{pmatrix} \mathbf{H}_{n-1} & \mathbf{H}_{n-1} \\ \mathbf{H}_{n-1} & -\mathbf{H}_{n-1} \end{pmatrix}.$$

Рассмотрим сложность вычисления преобразования Фурье 1-го типа булевой функции. Преобразование Фурье будем вычислять при помощи схем, элементы которых реализуют действительные операции сложения и вычитания. Строгое определение таких схем приводить не будем, так как оно почти дословно совпадает с определением схем из функциональных элементов, данным в четвертой главе. Единственное отличие состоит в том, что базис схемы состоит из действительных, а не булевых функций.

Лемма 5.6.1. *Существует состоящая из элементов действительного сложения и вычитания схема* S_n*, которая по вектору значений* n*-местной булевой функции вычисляет ее преобразование Фурье* 1*-го типа, и для сложности и глубины этой схемы справедливы равенства*:

$$L(\mathsf{S}_n) = n2^n, \qquad D(\mathsf{S}_n) = n.$$

Доказательство. Схемы S_n, сложности и глубины которых удовлетворяют условиям леммы построим индукцией по n. Схема S_1 строится тривиально, она состоит из одного элемента сложения и одного элемента вычитания. Предположим, что схема S_{n-1} построена. Воспользуемся этой схемой для построения схемы S_n, умножающей матрицу $\mathbf{H}_n$ на вектор значений $(f_0, \ldots, f_{2^n-1})$ функции f. Результатом умножения будет вектор значений преобразования Фурье булевой функции f. На рис. 5.6.1. представлена индуктивная процедура построения схемы S_n. Схема S_n состоит из двух экземпляров схемы S_{n-1}, 2^{n-1}

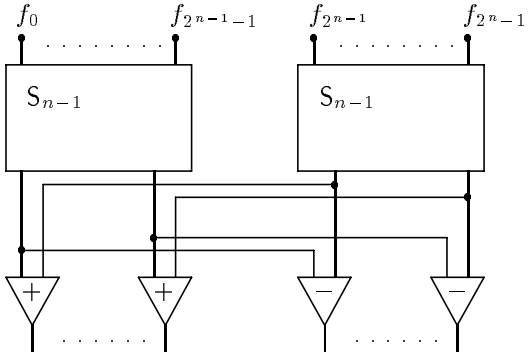


Рис. 5.6.1

элементов сложения и 2^{n-1} элементов вычитания. Легко видеть, что для сложности схемы S_n справедливы равенства

$$\begin{aligned} L(\mathsf{S}_n) &= 2L(\mathsf{S}_{n-1}) + 2 \cdot 2^{n-1} = 2\left(2L(\mathsf{S}_{n-2}) + 2^{n-1}\right) + 2^n = \\ &= 4L(\mathsf{S}_{n-2}) + 2 \cdot 2^n = \ldots = 2^{n-1}L(\mathsf{S}_1) + (n-1)2^n = n2^n. \end{aligned}$$

Для глубины S_n из рис. 5.5.1 видим, что

$$D(\mathsf{S}_n) = D(\mathsf{S}_{n-1}) + 1 = n.$$

Лемма доказана.

Теперь покажем, что сложность построенных в доказательстве леммы 5.6.1 схем не более чем в два раза превосходит минимально возможную сложность. Для этого рассмотрим схемы, которые состоят из элементов, реализующих сложение и вычитание n-мерных

действительных векторов. Каждая из рассматриваемых схем имеет n входов, и ее i-й вход подключен к единичному вектору $\boldsymbol{e}_i$ из стандартного в $\mathbb{R}^n$ базиса $E_n = \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$. При помощи таких схем будем порождать системы $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$ целочисленных векторов пространства $\mathbb{R}^n$ исходя из векторов базиса E_n. В данном случае схемы удобно представлять в виде последовательно выполняемых операций сложения и вычитания n-мерных векторов. Минимальное число таких операций, достаточное для порождения системы Z, называется сложностью этой системы и обозначается через $L_{\{+,-\}}(Z)$. Например, система $Z_2 = \{(1,1), (1,-1)\}$ порождается одним сложением $\boldsymbol{z}_1 = \boldsymbol{e}_1 + \boldsymbol{e}_2$ и одним вычитанием $\boldsymbol{z}_2 = \boldsymbol{e}_1 - \boldsymbol{e}_2$. Так как для порождения каждого вектора, отличного от вектора базиса, необходима, по крайней мере, одна операция, то, очевидно, что сложность системы Z_2 равна двум. Для сложности порождения произвольной системы Z справедливо следующее утверждение.

Лемма 5.6.2. *Для сложности $L_{\{+,-\}}(Z)$ порождения системы целочисленных векторов $Z = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n\}$ при помощи сложений и вычитаний справедливо неравенство*

$$L_{\{+,-\}}(Z) \geq \log_2 |\det(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)|.$$

Доказательство. Допустим, что существует последовательность вычислений, порождающая систему векторов Z за t шагов, каждый из которых состоит в вычислении суммы или разности двух векторов. Положим $\boldsymbol{y}_1 = \boldsymbol{e}_1, \ldots, \boldsymbol{y}_n = \boldsymbol{e}_n$ и для каждого $j \in \{1, \ldots, t\}$ через $\boldsymbol{y}_{n+j}$ обозначим вектор, вычисляемый на j-м шаге. Через r_k обозначим максимальное значение модуля определителя, вычисленного на векторах системы $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n+k}\}$, т. е.

$$r_k = \max_{1 \leq i_1 \leq \cdots \leq i_n \leq n+k} |\det(\boldsymbol{y}_{i_1}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})|.$$

Очевидно, что $r_0 = |\det(\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n)| = 1$. Оценим сверху величину r_k для $k > 0$. Из линейности определителя по каждому аргументу легко следует, что

$$\begin{aligned} r_k &= |\det(\boldsymbol{y}_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})| = |\det(\boldsymbol{y}'_{n+k} + \boldsymbol{y}''_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})| \leq \\ &= |\det(\boldsymbol{y}'_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n}) \ + \ \det(\boldsymbol{y}''_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})| \leq \\ &\leq |\det(\boldsymbol{y}'_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})| + |\det(\boldsymbol{y}''_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n})|. \end{aligned}$$

Так как векторы $\boldsymbol{y}'_{n+k}, \boldsymbol{y}''_{n+k}, \boldsymbol{y}_{i_2}, \ldots, \boldsymbol{y}_{i_n}$ принадлежат системе $\{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{n+k-1}\}$, то, очевидно, что каждый из определителей в правой части последнего неравенства не превосходит r_{k-1}. Поэтому $r_k \leq 2r_{k-1} \leq 2^k$. Логарифмируя полученное неравенство, видим, что двоичный логарифм модуля определителя любой системы векторов, порожденной за k шагов, не превосходит k. Следовательно, для системы векторов Z справедливо неравенство

$$L_{\{+,-\}}(Z) = t \geq |\det(\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n)|.$$

Лемма доказана.

Теперь заметим, что схема, порождающая систему векторов $\{(z_{i1}, \ldots, z_{in})\}_{i=1}^n$, будет вычислять систему линейных функций $\{z_{i1}x_1 + \cdots + z_{in}x_n\}_{i=1}^n$, если i-й вход схемы подключить к переменной x_i, а элементы, складывающие и вычитающие векторы, заменить элементами, вычисляющими суммы и разности действительных чисел. Поэтому для сложности любой схемы S, вычисляющей преобразование Фурье 1-го типа n-местной булевой функции, справедливо неравенство

$$L(\mathsf{S}) \geq \log_2 |\det \mathbf{H}_n|.$$

Так как матрица $\mathbf{H}_n$ ортогональна и симметрична, и скалярный квадрат любой ее строки равен 2^n, то легко видеть, что $\mathbf{H}_n\mathbf{H}_n = (2^n)^{2^n}\mathbf{E}_{2^n}$. Поэтому $\det \mathbf{H}_n = \sqrt{2^{n2^n}}$, и, следовательно,

$$L(\mathsf{S}) \geq \log_2 \left|\sqrt{2^{n2^n}}\right| = n2^{n-1}.$$

Таким образом, из полученного неравенства и доказанной выше леммы 5.6.1, получаем следующий результат.

Теорема 5.6.1. *Для сложности вычисления преобразования Фурье* 1*-го типа* n*-местной булевой функции в базисе из сложений и вычитаний справедливы неравенства*

$$n2^{n-1} \leq L_{\{+,-\}}(F_n) \leq n2^n.$$

Задачи

5.6.1. Построить схему, вычисляющую преобразование Фурье 1-го типа булевой функции трех переменных по ее значениям.

5.6.2. Оценить сложность вычисления преобразования Фурье 2-го типа n-местной булевой функции.

5.6.3. Построить схему, вычисляющую преобразование Фурье 2-го типа булевой функции трех переменных по ее значениям.

Глава 6.

Асимптотические методы построения схем

В двух предыдущих главах изучалась сложность конкретных булевых функций и операторов — бралась вполне определенная функция (или оператор) и для этой функции (оператора) строилась вычисляющая ее схема. В настоящей главе применяется иной подход. Рассматривается достаточно большое множество булевых функций или операторов, например $P_2(n)$, и для этого множества разрабатывается единый метод построения схем, позволяющий для любой функции из рассматриваемого множества построить вычисляющую ее схему. Для многих естественных множеств булевых функций и операторов, например для $P_2(n)$, этот подход оказывается очень эффективным, позволяя для почти всех функций и операторов из рассматриваемых множеств строить схемы, сложность которых при $n \to \infty$ стремится к сложности минимальных схем. Такие схемы будем называть асимптотически минимальными.

6.1. Вычисление дизъюнкции n функций

Ниже рассматривается задача определения глубины дизъюнкции n функций $f_1, \ldots, f_n$ при условии, что известны глубины функций f_i. Схему D_n, состоящую из дизъюнкторов и вычисляющую дизъюнкцию n функций, назовем *n-местным дизъюнктором*. Длину самой длинной цепи, связывающей i-й вход дизъюнктора с его выходом назовем высотой i-го входа. При n большем трех существуют различные дизъюнкторы, имеющие входы разной высоты. На рисунке 6.1.1 изображены два различных четырехместных дизъюнктора D_4. В левом дизъюнкторе высоты входов составляют набор $(3,3,2,1)$, в правом — $(2,2,2,2)$. Дизъюнктор назовем *равномерным*, если высоты его входов различаются не более чем на единицу. Правый дизъюнктор на рисунке 6.1.1 является равномерным, а левый нет.

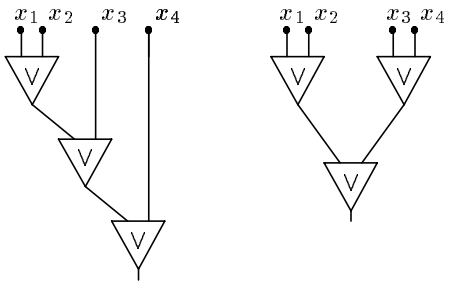


Рис. 6.1.1

Теорема 6.1.1. *Если для набора целых положительных чисел $(l_1, \ldots, l_n)$ справедливо неравенство*

$$\sum_{i=1}^{n} \frac{1}{2^{l_i}} = 1,$$

то существует такой n-местный дизъюнктор D_n, что высоты его входов образуют набор $(l_1, \ldots, l_n)$.

Доказательство. Теорема легко доказывается индукцией по числу входов дизъюнктора. При $n = 2$ утверждение теоремы очевидно — существует единственный дизъюнктор

D_2, состоящий из одного элемента, и высота каждого его входа равна единице. Предположим, что теорема справедлива при $n = k - 1$.

Пусть $(l_1, \ldots, l_k)$ — произвольный набор целых положительных чисел, удовлетворяющих равенству $\sum_{i=1}^{k} 2^{-l_i} = 1$. Покажем, что существует k-местный дизъюнктор D_k, у которого высота i-го входа равна i_i для каждого i. Без ограничения общности будем полагать, что l_i упорядочены в порядке убывания индексов, т. е. $l_1 \geq \cdots \geq l_k$. Нетрудно показать, что в рассматриваемом наборе два последних числа равны: $l_{k-1} = l_k$. Набор $(l_1, \ldots, l_k)$ преобразуем в новый набор $(l'_1, \ldots, l'_{k-1})$ в котором первые $k-2$ числа такие же, как и в исходном наборе, а последнее l'_{k-1} равно $l_k + 1$. Легко видеть, что

$$\sum_{i=1}^{k-1} \frac{1}{2^{l'_i}} = \sum_{i=1}^{k} \frac{1}{2^{l_i}} = 1. \tag{6.1.1}$$

Из (6.1.1) и предположения индукции следует, что существует дизъюнктор D_{k-1}, высоты входов которого составляют набор $(l'_1, \ldots, l'_{k-1})$. Подключим $(k-1)$-й вход дизъюнктора D_{k-1} к двуместному дизъюнктору D_2. Легко видеть, что построенная схема является искомым дизъюнктором. Следовательно, глубина схемы S не превосходит d. Теорема доказана.

Без доказательства приведем простое следствие теоремы 6.1.1.

Следствие 6.1.1. *Если для набора целых положительных чисел $(l_1, \ldots, l_n)$ справедливо неравенство*

$$\sum_{i=1}^{n} \frac{1}{2^{l_i}} \leq 1,$$

то существует n-местный дизъюнктор D_n высоты входов которого образуют такой набор $(l'_1, \ldots, l'_n)$, что $l'_i \leq l_i$ при всех $i \in \{1, \ldots, n\}$.

Теорема 6.1.2. *Пусть $f_1, \ldots, f_k$ — булевы функции, $f = f_1 \vee \cdots \vee f_k$. Тогда*

$$D(f) \leq \left\lceil \log_2 \sum_{i=1}^{k} 2^{D(f_i)} \right\rceil.$$

Доказательство. Положим $d = \left\lceil \log_2 \sum_{i=1}^{k} 2^{d_i} \right\rceil$, $d_j = D(f_j)$, $l_j = d - d_j$, где $j = 1, 2, \ldots, k$. Тогда

$$2^d = 2^{\left\lceil \log_2 \sum_{i=1}^{k} 2^{d_i} \right\rceil} \geq 2^{\log_2 \sum_{i=1}^{k} 2^{d_i}} = \sum_{i=1}^{k} 2^{d_i}.$$

Следовательно,

$$\sum_{i=1}^{k} 2^{-l_i} = \sum_{i=1}^{k} 2^{d_i - d} = 2^{-d} \sum_{i=1}^{k} 2^{d_i} \leq 1.$$

Из полученного неравенства и следствия 6.1.1 следует, что существует такой k-местный дизъюнктор D_k, что при каждом $j \in \{1, \ldots, k\}$ ее j-й вход имеет высоту $l'_j \leq l_j = d - d_j$. Пусть схема S_j вычисляет функцию f_j и глубина этой схемы равна d_j. Схему S, вычисляющую функцию f, построим следующим образом: при каждом $j \in \{1, \ldots, k\}$ j-й вход дизъюнктора D_k подключим к выходу схемы S_j. Легко видеть, что длина любой ориентированной цепи, связывающей произвольный вход схемы S_j с выходом D_k не превосходит величины

$$l'_j + d_j \leq l_j + d_j = d.$$

Следовательно, глубина схемы S не превосходит d. Теорема доказана.

Задачи

6.1.1. Показать, что глубина любого n-местного равномерного дизъюнктора не превосходит $\lceil \log_2 n \rceil$.

6.1.2. Показать, что набор высот входов любого n-местного дизъюнктора удовлетворяет неравенству $\sum_{i=1}^{n} 2^{-l_i} \leq 1$.

6.1.3. Доказать следствие 6.1.1.

6.1.4. Показать, что для любой элементарной конъюнкции $x_1^{(\sigma_1)} \cdot \ldots \cdot x_n^{(\sigma_n)}$ найдется:

а) вычисляющая ее в базисе $\{\&, \vee, \neg\}$ схема S_1 для сложности и глубины которой справедливы неравенства $L(\mathsf{S}_1) \leq 2n$ и $D(\mathsf{S}_1) \leq \lceil \log_2 n \rceil + 1$;

b) вычисляющая ее в базисе $\{\&, \vee, \neg\}$ схема S_2 для сложности и глубины которой справедливы неравенства $L(\mathsf{S}_2) \leq n$ и $D(\mathsf{S}_2) \leq \lceil \log_2 n \rceil + 2$.

6.2. Вычисление систем дизъюнкций. Широкие системы

При построении схем, вычисляющих системы дизъюнкций, удобно различать "широкие" системы, в которых переменных больше чем дизъюнкций, и "узкие" системы, в которых больше дизъюнкций. Сначала рассмотрим сложность вычисления "широких" систем. "Узкие" системы будут рассмотрены в следующем параграфе.

1. Пусть $f_{ij} \in \mathbb{B}$ для всех $1 \leq i \leq m$ и $1 \leq j \leq n$. Рассмотрим систему F, состоящую из m дизъюнкций

$$f_i = f_{i1}x_1 \vee \cdots \vee f_{ij}x_j \vee \cdots \vee f_{in}x_n, \quad i = 1, \ldots, m. \tag{6.2.1}$$

Величины f_{ij}, $j = 1, 2, \ldots, n$, называются *коэффициентами* дизъюнкции f_i. Множество всех систем дизъюнкций вида (6.2.1) обозначим через $\mathcal{D}(m, n)$. Булеву (m, n)-матрицу $\mathbf{A} = (a_{ij})$ назовем *матрицей системы дизъюнкций* (6.2.1), если $a_{ij} = f_{ij}$ для всех $1 \leq i \leq m$ и $1 \leq j \leq n$. Например, матрицей системы $F = \{x_1 \vee x_2 \vee x_3, x_2 \vee x_3\}$ будет матрица

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

Характеристическим вектором переменной x_j системы (6.2.1) называется j-й столбец матрицы этой системы.

2. Опишем простой и эффективный метод, позволяющий сводить вычисление "широких" систем дизъюнкций, симметричных относительно части своих переменных, к вычислению более простых систем, зависящих от меньшего числа переменных.

Прежде всего заметим, что если некоторая система дизъюнкций F симметрична относительно переменных x_i и x_j, то в каждую дизъюнкцию этой системы переменные x_i и x_j одновременно либо входят, либо не входят. Определить переменные, относительно которых система дизъюнкций симметрична, проще всего по ее матрице. Легко видеть, что если система дизъюнкций $F(x_1, \ldots, x_n)$ симметрична относительно переменных $x_{i_1}, \ldots, x_{i_k}$, то в матрице $\mathbf{F}$ этой системы столбцы с номерами $i_1, \ldots, i_k$ будут одинаковыми.

Перейдем к изложению метода. Сделаем это сначала на простом примере. Рассмотрим состоящую из дизъюнкций f_1 и f_2 систему F и ее матрицу $\mathbf{F}$, такие, что

$$f_1 = x_1 \vee x_2 \vee x_3 \vee x_4, \qquad \mathbf{F} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$
$$f_2 = x_3 \vee x_4 \vee x_5 \vee x_6.$$

Легко видеть, что в каждую дизъюнкцию системы F пары переменных (x_1, x_2), (x_3, x_4) и (x_5, x_6) одновременно либо входят, либо не входят. Поступим следующим образом. Преобразуем систему F в новую систему дизъюнкций $G(y_1, y_2, y_3)$ выполнив замену переменных

$$y_1 = x_1 \vee x_2, \quad y_2 = x_3 \vee x_4, \quad y_3 = x_5 \vee x_6.$$

Очевидно, что G состоит из двух дизъюнкций $y_1 \vee y_2$ и $y_2 \vee y_3$, и связана с системой F следующим равенством

$$F(x_1, x_2, x_3, x_4, x_5, x_6) = G(x_1 \vee x_2, x_3 \vee x_4, x_5 \vee x_6).$$

Воспользуемся им для построения схемы, вычисляющей систему F. Сначала построим схему S_1 вычисляющую систему дизъюнкций G и схему S_2 вычисляющую функции y_1, y_2, y_3. Затем i-й вход схемы S_2 подключим к i-му выходу схемы S_1. Легко видеть, что получившаяся схема состоит из пяти дизъюнкторов и вычисляет систему F.

Рассмотренный пример легко обобщается на случай, когда система дизъюнкций симметрична относительно нескольких подмножеств переменных. Действительно, пусть $M_1, \ldots, M_k$ такие непересекающиеся подмножества множества переменных $\{x_1, ..., x_n\}$, что переменные одного подмножества одновременно либо входят, либо не входят в каждую дизъюнкцию системы F, и при этом каждая переменная x_i принадлежит одному из этих множеств. Переменные $x_1, \ldots, x_n$ перенумеруем двумя индексами так, чтобы переменные $x_{j1}, \ldots, x_{ji_j}$ составляли множество M_j. После замены переменных $y_j = x_{j1} \vee \cdots \vee x_{ji} \vee \cdots \vee x_{ji_j}$, где $j = 1, 2, \ldots, k$, из системы F получим новую систему дизъюнкций $G(y_1, \ldots, y_k)$ такую, что

$$F(x_1, \ldots, x_n) = G(x_{11} \vee \cdots \vee x_{1i_1}, \ldots, x_{k1} \vee \cdots \vee x_{ki_k}).$$

Схему, вычисляющую систему F, как и в рассмотренном примере, составим из двух подсхем. Первая из этих подсхем содержит n входов, подключенных к переменным $x_1, \ldots, x_n$, и k выходов. На ее j-м выходе вычисляется дизъюнкция

переменных, входящих в подмножество M_j. Вторая подсхема вычисляет систему G и ее i-й вход подключен к i-му выходу первой подсхемы.

Воспользуемся представленным методом для доказательства следующей леммы.

Лемма 6.2.1. *Пусть $m \le \log_2 n - 2\log_2\log_2 n$. Тогда при $n \to \infty$ произвольная система дизъюнкций $F \in \mathcal{D}(m, n)$ может быть вычислена такой схемой S, что*

$$L(\mathsf{S}) = n + \mathcal{O}\left(\frac{n}{\log_2 n}\right), \quad D(\mathsf{S}) \le \lceil \log_2 n \rceil + 1.$$

Доказательство. Пусть $\mathbf{F}$ — матрица системы F. Как уже отмечалось, если система F симметрична относительно переменных $x_{i_1}, \ldots, x_{i_k}$, то столбцы матрицы $\mathbf{F}$ с номерами $i_1, \ldots, i_k$ будут одинаковыми. Параметры леммы m и n таковы $(2^m \ll n)$, что число различных видов столбцов высоты m много меньше числа столбцов $\mathbf{F}$. Поэтому в матрице $\mathbf{F}$ будет много одинаковых столбцов, и, следовательно, система F симметрична относительно некоторых подмножеств своих переменных. Используем это свойство рассматриваемой системы.

Для каждого $j \in \{1, 2, \ldots, 2^m - 1\}$ сформируем множество переменных M_j такое, что переменная x_i принадлежит M_j тогда и только тогда, когда i-й столбец матрицы $\mathbf{F}$ совпадает с двоичным представлением числа j. Далее для каждого непустого множества M_j определим дизъюнкцию входящих в это множество переменных:

$$y_j = \bigvee_{x_i \in M_j} x_i. \tag{6.2.2}$$

Легко видеть, что для k-й функции f_k, $1 \le k \le m$, системы F справедливо равенство

$$f_k = \bigvee y_j, \tag{6.2.3}$$

в котором дизъюнкция берется по всем тем целым j, $1 \le j \le 2^m - 1$, в двоичном представлении которых коэффициент при $(m-k)$-й степени двойки равен единице.

Каждую функцию y_j вычислим отдельной схемой S_j, которая является экземпляром равномерного дизъюнктора $\mathsf{D}_{|M_j|}$. Глубина каждой схемы S_j удовлетворяет неравенству

$$D(\mathsf{S}_j) \le \lceil \log_2 |M_j| \rceil. \tag{6.2.4}$$

Так как каждая функция x_i входит не более чем в одну дизъюнкцию y_j, то очевидно, что

$$\sum_{j=1}^{2^m-1} L(\mathsf{S}_j) < n. \tag{6.2.5}$$

Каждую функцию f_k вычислим отдельной схемой S'_k. Схема S'_k вычисляет f_k в соответствии с формулой (6.2.3) и является экземпляром дизъюнктора, входы которого подключены к выходам соответствующих схем S_j. Из (6.2.3) немедленно следует, что

$$\sum_{k=1}^{m} L(\mathsf{S}'_k) \le m \cdot 2^m \le \frac{n}{\log_2 n}. \tag{6.2.6}$$

Таким образом, из (6.2.5) и (6.2.6) следует, что

$$L(\mathsf{S}) \le n + \frac{n}{\log_2 n}.$$

Теперь оценим общую глубину схемы, вычисляющей функцию f_k. Из теоремы 6.1.2 и неравенства (6.2.4) следует, что для каждого k

$$\begin{aligned} D(f_k) &\le \left\lceil \log_2 \sum_{j=1}^{2^m-1} 2^{D(\mathsf{S}_j)} \right\rceil \le \left\lceil \log_2 \sum_{j=1}^{2^m-1} 2^{\lceil \log_2 |M_j| \rceil} \right\rceil \le \\ &\le \left\lceil \log_2 \sum_{j=1}^{2^m-1} 2^{1+\log_2 |M_j|} \right\rceil = \left\lceil 1 + \log_2 \sum_{j=1}^{2^m-1} |M_j| \right\rceil \le \lceil \log_2 n \rceil + 1. \end{aligned}$$

Лемма доказана.

Пример 6.2.1. Воспользуемся методом, изложенным в доказательстве леммы 6.2.1, для построения схемы S, вычисляющей систему дизъюнкций F с матрицей

$$\mathbf{F} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$

Сначала, следуя доказательству этой леммы, сформируем множества M_j. Легко видеть, что

$$\begin{aligned} &M_1 = \varnothing, && M_3 = \{4, 8\}, && M_5 = \{1\}, && M_7 = \{3, 6, 9\}, \\ &M_2 = \varnothing, && M_4 = \{2, 7\}, && M_6 = \{5\}. \end{aligned}$$

Так же легко (см. (6.2.2)) получим функции y_j, соответствующие непустым множествам M_j:

$$\begin{aligned} &y_3 = x_4 \vee x_8, \quad y_5 = x_1, \quad y_7 = x_3 \vee x_6 \vee x_9, \\ &y_4 = x_2 \vee x_7, \quad y_6 = x_5. \end{aligned} \tag{6.2.7}$$

Наконец, выразим функции системы F через функции y_j. В соответствии с равенством (6.2.3) имеем

$$f_1 = y_4 \vee y_5 \vee y_6 \vee y_7, \quad f_2 = y_3 \vee y_6 \vee y_7, \quad f_3 = y_3 \vee y_5 \vee y_7. \tag{6.2.8}$$

Теперь, используя формулы (6.2.7) и (6.2.8), можно легко построить требуемую схему S. Сначала четырьмя дизъюнкторами вычислим функций y_3, y_4 и y_7 (функции y_5 и y_6 получаются даром). Затем, заметив, что в равенствах (6.2.8) дизъюнкция $y_5 \vee y_7$ входит в две функции f_1 и f_3, шестью дизъюнкторами вычислим функции f_1, f_2 и f_3. Таким образом, схема S состоит из десяти элементов. $\square$

Использованный в предыдущей лемме метод применим только к очень широким системам. Если двоичный логарифм числа переменных системы дизъюнкций меньше числа ее дизъюнкций, то в матрице такой системы может не найтись одинаковых столбцов, и в этом случае напрямую воспользоваться разработанной выше техникой построения схем не удастся. Расширить границы применимости этой техники можно при помощи очень простого приема — разделения системы дизъюнкций на несколько подсистем, размеры каждой из которых удовлетворяют условиям леммы 6.2.1. Именно такой способ использован в доказательстве следующей леммы.

Лемма 6.2.2. *При $n \to \infty$ произвольная система дизъюнкций $F \in \mathcal{D}(m,n)$ может быть вычислена такой схемой* S*, что*

$$L(\mathsf{S}) \le \frac{n(m+\log_2 n)}{\log_2 n}\left(1+\mathcal{O}\left(\frac{\log_2\log_2 n}{\log_2 n}\right)\right),$$
$$D(\mathsf{S}) \le \lceil \log_2 n \rceil + 1.$$

Доказательство. Положим $q = \lfloor \log_2 n - 2\log_2\log_2 n \rfloor$, $k = \lceil m/q \rceil$. Без ограничения общности полагаем, что $m \ge q$, так как при $m < q$ настоящая лемма следует из леммы 6.2.1. Из дизъюнкций системы F сформируем k новых систем $F_1, \ldots, F_k$ так, что

$$F_j = \{f_{(j-1)q+1}, \ldots, f_{jq}\} \text{ при } j = 1,2,\ldots,k-1,$$
$$F_k = \{f_{(k-1)q+1}, \ldots, f_m\}.$$

Каждую из систем F_i вычислим собственной схемой S_i, сложность и глубина которой в силу леммы 6.2.1 удовлетворяют соотношениям

$$L(\mathsf{S}_i) \le n + \frac{n}{\log_2 n}, \quad D(\mathsf{S}_i) \le \lceil \log_2 n \rceil + 1.$$

Очевидно, что схема $\mathsf{S} = \cup_{i=1}^{k} \mathsf{S}_i$ вычисляет систему дизъюнкций $F = \cup_{i=1}^{k} F_i$ и для ее глубины и сложности справедливы соотношения

$$D(\mathsf{S}) \le \max_{1\le i\le k} D(\mathsf{S}_i),$$
$$L(\mathsf{S}) = \sum_{i=1}^{k} L(\mathsf{S}_i) \le \left(n+\frac{n}{\log_2 n}\right)\left\lceil \frac{m}{\log_2 n - \log_2\log_2 n}\right\rceil \le$$
$$\le \frac{n(m+\log_2 n)}{\log_2 n}\left(1+\mathcal{O}\left(\frac{\log_2\log_2 n}{\log_2 n}\right)\right).$$

Лемма доказана.

3. Систему дизъюнкций F из $\mathcal{D}(n, 2^n-1)$ назовем *универсальной* и обозначим через U_n, если j-й столбец матрицы $\mathbf{U}_n$ этой системы совпадает с двоичным представлением числа j. Например, универсальной системой U_3 с матрицей $\mathbf{U}_3$ будет система трех дизъюнкций $\{f_1, f_2, f_3\}$, если

$$f_1 = x_4 \vee x_5 \vee x_6 \vee x_7,$$
$$f_2 = x_2 \vee x_3 \vee x_6 \vee x_7,$$
$$f_3 = x_1 \vee x_3 \vee x_5 \vee x_7.$$

$$\mathbf{U}_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Универсальные системы интересны тем, что любая система F, состоящая из m дизъюнкций, может быть получена из системы U_m удалением одних переменных и дублирование других.

Теорема 6.2.1. *Справедливы равенства*

$$L(U_n) = 2(2^n - n - 1), \qquad D(U_n) = n - 1.$$

Доказательство. Верхние оценки. Для доказательства теоремы построим схему U_n, вычисляющую универсальную систему дизъюнкций U_n, состоящую из $2(2^n-n-1)$ дизъюнкторов и имеющую глубину $n-1$. Схему U_n построим индукцией по n. На рисунке 6.2.1. представлена индуктивная процедура построения схемы U_n: в левой части изображен базис индукции, в правой — индуктивный переход. Индуктивный переход основан

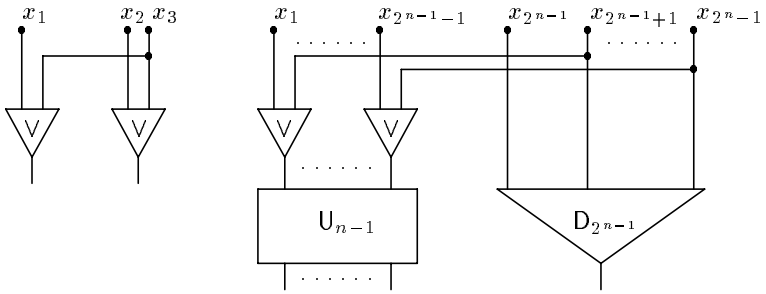


Рис. 6.2.1

на следующем простом свойстве матрицы $\mathbf{U}_n$:

$$u_{ij} = u_{ij+2^{n-1}} \quad \text{при всех } 2 \le i \le n \text{ и } 1 \le j \le 2^{n-1}-1.$$

Найдем сложность $L(\mathsf{U}_n)$ и глубину $D(\mathsf{U}_n)$ схемы U_n. Очевидно, что $L(\mathsf{U}_2)=2$ и $D(\mathsf{U}_2)=1$. Из конструкции схемы U_n видим, что

$$L(\mathsf{U}_n) = L(\mathsf{U}_{n-1}) + L(\mathsf{D}_{2^{n-1}}) + 2^{n-1} - 1 = L(\mathsf{U}_{n-1}) + 2(2^{n-1}-1),$$
$$D(\mathsf{U}_n) = \max(1 + D(\mathsf{U}_{n-1}), D(\mathsf{D}_{2^{n-1}})) = \max(1 + D(\mathsf{U}_{n-1}), n-1).$$

По предположению индукции

$$L(\mathsf{U}_{n-1}) = 2(2^{n-1} - (n-1) - 1), \quad D(\mathsf{U}_{n-1}) = n-2.$$

Следовательно,

$$L(\mathsf{U}_{n-1}) = 2(2^{n-1} - (n-1) - 1) + 2(2^{n-1}-1) = 2(2^n - n - 1),$$
$$D(\mathsf{U}_{n-1}) = \max(1 + (n-2), n-1) = n-1.$$

Верхние оценки сложности и глубины доказаны.

Прежде чем доказывать нижние оценки сложности системы U_n, дадим важное определение и докажем две необходимых леммы, имеющих так же самостоятельный интерес.

Пусть S — произвольная схема. Вершину v схемы S назовем *вершиной первого ветвления* входа x_i, если v и вход x_i связаны цепью, проходящей через вершины $\mathsf{v}_0 = x_i, \boldsymbol{v}_1, \ldots, \mathsf{v}_{k-1}, \mathsf{v}_k = \mathsf{v}$, в которой каждая из вершин v_j, где $j = 0, 1, \ldots, k-1$, имеет ровно одного потомка v_{j+1}, а вершина v — не менее двух потомков. Для примера, рассмотрим первые две схемы, изображенные слева на рисунке 4.4.3 на странице 87. В первой схеме входы x и y не имеют вершин ветвления, а вход z сам является своей вершиной первого ветвления. На второй схеме входы x и y имеют общую вершину первого ветвления — элемент, помеченный единицей, а у входа z нет вершины первого ветвления.

Лемма 6.2.3. *В любой схеме S, вычисляющей систему дизъюнкций F, два входа имеют различные вершины первого ветвления, если характеристические векторы соответствующих этим входам переменных различны и каждый из этих векторов содержит не менее двух единиц.*

Доказательство. Пусть схема S вычисляет систему дизъюнкций $F(x_1, \ldots, x_n)$, и пусть i-й и j-й входы S удовлетворяют условиям леммы. Допустим, что у i-го и j-го входов общая вершина первого ветвления v. В схеме S вместо всех переменных, исключая x_i и x_j, подставим тождественный нуль. После такой подстановки новая схема S' будет вычислять систему дизъюнкций, зависящих только от двух переменных x_i и x_j. Матрица этой

системы будет состоять из двух столбцов — характеристических векторов переменных x_i и x_j. Далее рассмотрим два случая: первый, когда в новой матрице нет строки с двумя единицами; второй, когда такая строка есть.

В первом случае среди дизъюнкций обязательно найдется хотя бы одна, существенно зависящая только от одной из переменных, например от x_i, и одна, существенно зависящая только от другой переменной. Во втором случае найдется дизъюнкция, существенно зависящая только от одной переменной, например от x_i, и дизъюнкция, существенно зависящая от x_i и x_j. В обоих случаях первую дизъюнкцию обозначим через $f_1(x_i, x_j)$, вторую — через $f_2(x_i, x_j)$. Легко видеть, что

$$f_1(0,1) \neq f_1(1,0), \qquad f_2(0,0) \neq f_2(0,1), \quad f_1(0,1) \neq f_2(0,1). \tag{6.2.9}$$

Теперь рассмотрим вершину v. После подстановки нулей в этой вершине будет вычисляться некоторая функция $h(x_i, x_j)$, зависящая только от x_i и x_j. Так как любая цепь, связывающая каждый из входов x_i или x_j с любым выходом схемы S', проходит через вершину v, то каждую функцию, вычисляемую схемой S', будем рассматривать как функцию, зависящую только от $h(x_i, x_j)$. Таким образом, $f_1(x_i, x_j) = g_1(h(x_i, x_j)$ и $f_2(x_i, x_j) = g_2(h(x_i, x_j)$. Из первых двух неравенств (6.2.9) видно, что ни g_1, ни g_2, не являются тождественными постоянными. Из третьего неравенства следует, что $g_1 \neq g_2$, т.е. одна из этих функций будет отрицанием, а вторая — тождественной функцией. Без ограничения общности будем полагать, что отрицанием будет g_1. Тогда

$$0 = f_1(0,0) = \overline{h}(0,0) \neq h(0,0) = f_2(0,0) = 0.$$

Противоречие. Лемма доказана.

Лемма 6.2.4. *Пусть система дизъюнкций F такая, что среди столбцов ее (m,n)-матрицы* $\mathbf{F}$: *(i) нет ни одного нулевого столбца; (ii) есть k различных столбцов, каждый из которых содержит не менее двух единиц. Тогда*

$$L(F) \geq n + k - m.$$

Доказательство. Пусть S — минимальная схема, вычисляющая систему F. Из леммы 6.2.3 легко следует, что в S найдется не менее k входов, каждый из которых имеет собственную вершину первого ветвления, и, следовательно, в схеме S есть k вершин, каждая из которых имеет не менее двух потомков. Число элементов схемы S обозначим через L, число ребер — через N. Оценим N, подсчитывая ребра, выходящие из вершин схемы S.

1. Из каждого входа схемы обязательно выходит одно ребро — всего n ребер.

2. Из всех элементов схемы, не являющихся ее выходами, обязательно выходит по одному ребру — всего $L - m$ ребер.

3. Кроме этого в S есть не менее k вершин первого ветвления и из каждой выходит не менее двух ребер — всего $2k$ ребер. Из этих ребер k учтено в 1 и 2 пунктах. Следовательно, неучтенных ребер k.

Общее число выходящих ребер — $n + k + L - m$. Следовательно, $N \geq n + k + L - m$.

Теперь оценим N, подсчитывая ребра, входящие в вершины схемы S. Ребра входят только в элементы S, причем в каждый элемент входит не более двух ребер. Поэтому $N \leq 2L$. Следовательно,

$$n + k + L \leq N \leq 2L.$$

Откуда немедленно получаем

$$L \geq n + k - m.$$

Лемма доказана.

Заметим, что из доказанной леммы легко следует минимальность схемы, построенной в примере 6.2.1.

Доказательство нижней оценки теоремы 6.2.1. Так как все 2^n-1 столбцов матрицы $\mathbf{U}_n$ различные, ненулевые и среди них есть 2^n-n-1 столбцов содержащих не менее чем по две единицы, то в силу леммы 6.2.4

$$L(U_n) \geq (2^n-1)+(2^n-n-1)-n = 2(2^n-n-1).$$

Теорема доказана.

Задачи

6.2.1. Показать, что для любой системы дизъюнкций f из $\mathcal{D}(m,2^m)$ справедливо неравенство $L_\vee(f) \leq 2^{m+1}-2m-1$.

6.2.2. Пусть $\mathcal{L}(m,n)$ — множество линейных (m,n)-операторов. Показать, что для любого оператора f из $\mathcal{L}(m,2^m)$ справедливо неравенство $L_\oplus(f) \leq 2^{m+1}-2m-1$.

6.2.3. Показать, что для любой системы дизъюнкций f из $\mathcal{D}(m,n)$ справедливо неравенство $L_\vee(f) \leq n+2^m-2m-1$.

6.2.4. Показать, что при $n \geq 2^m-1$ в $\mathcal{D}(m,n)$ найдется такая система дизъюнкций f, для которой справедливо неравенство $L_\vee(f) \geq n+2^m-2m-1$.

6.2.5. Показать, что для любого оператора f из $\mathcal{L}(m,n)$ справедливо неравенство $L_\oplus(f) \leq n+2^m-2m-1$.

6.2.6. Пусть $\mathbf{N}_m=(n_{i,j})$ — такая булева (m,m) матрица, что $(n_{i,j})=1 \iff i \neq j$, N_m — линейный (m,m)-оператор с матрицей коэффициентов $\mathbf{N}_m$. Показать, что
a) $L_\oplus(N_3)=3$; b) $L_\oplus(N_4)=6$; c) $L_\oplus(N_n)=2n-2$.

6.2.7. Доказать аналог леммы 6.2.4 для схем, вычисляющих линейные операторы.

6.2.8. Найти сложность системы дизъюнкций f, если ее матрица $\mathbf{F}$ имеет следующий вид:

$$\mathbf{F}=\begin{pmatrix}1&1&1&0&1&1&1&0&1&0\\1&1&1&1&1&1&0&1&1&1\\1&0&1&1&0&1&0&1&1&1\end{pmatrix}.$$

6.2.9. Найти сложность линейного оператора f, если его матрица $\mathbf{F}$ имеет следующий вид:

$$\mathbf{F}=\begin{pmatrix}1&1&1&0&1&1&1&1&0&0\\0&0&1&0&1&1&0&1&1&1\\1&0&1&1&0&1&1&1&0&0\end{pmatrix}.$$

6.3. Вычисление систем дизъюнкций. Узкие системы

Теперь рассмотрим системы дизъюнкций, в которых дизъюнкций больше чем переменных. Сначала докажем аналог теоремы 6.2.1 — найдем сложность одновременного вычисления всех монотонных дизъюнкций n переменных. Затем оценим сложность вычисления произвольной "узкой" системы.

Лемма 6.3.1. *Существует схема $\mathsf{U}_n^\star$, вычисляющая все монотонные дизъюнкции переменных $x_1,\ldots,x_n$ сложность и глубина которой равны*

$$L(\mathsf{U}_n^\star)=2^n-n-1, \qquad D(\mathsf{U}_n^\star)=\lceil \log_2 n \rceil.$$

Доказательство. Лемму докажем индукцией по n. В основание индукции положим схему $\mathsf{U}_1^\star$, вычисляющую единственную монотонную дизъюнкцию одной переменной. Эта схема не содержит ни одного элемента и ее вход является одновременно ее выходом.

Положим $m=\lceil k/2 \rceil$, $l=\lfloor k/2 \rfloor$. По предположению индукции существуют такие схемы $\mathsf{U}_m^\star$ и $\mathsf{U}_l^\star$, что

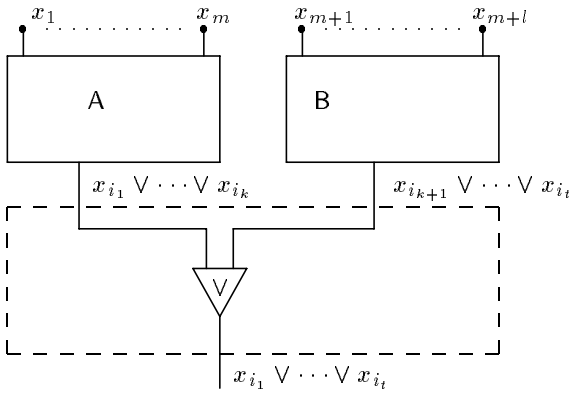


Рис. 6.3.1

(*i*) схема $\mathsf{U}^\star_m$ вычисляет все $2^m - 1$ монотонные дизъюнкции переменных $x_1, \ldots, x_m$ и для ее сложности и глубины справедливы равенства

$$L(\mathsf{U}^\star_m) = 2^m - m - 1, \qquad D(\mathsf{U}^\star_m) = \lceil \log_2 m \rceil;$$

(*ii*) схема $\mathsf{U}^\star_l$ вычисляет все $2^l - 1$ монотонные дизъюнкции переменных $x_{m+1}, \ldots, x_{m+l}$ и для ее сложности и глубины справедливы равенства

$$L(\mathsf{U}^\star_m) = 2^l - l - 1, \qquad D(\mathsf{U}^\star_m) = \lceil \log_2 l \rceil.$$

Воспользуемся этими схемами для построения требуемой схемы $\mathsf{U}^\star_n$.

Опишем конструкцию схемы $\mathsf{U}^\star_n$. Эта схема состоит из трех подсхем A, B и C. Подсхема A является экземпляром схемы $\mathsf{U}^\star_m$ входы которой подключены к переменным $x_1, \ldots, x_m$. Подсхема B является экземпляром схемы $\mathsf{U}^\star_l$ входы которой подключены к переменным $x_{m+1}, \ldots, x_{m+l}$. Подсхема C состоит из $(2^m - 1)(2^l - 1) = 2^{m+l} - 2^m - 2^l + 1$ дизъюнкторов. Каждый из этих дизъюнкторов вычисляет дизъюнкцию одного выхода подсхемы A с одним выходом подсхемы B. Выходами схемы $\mathsf{U}^\star_n$ являются выходы подсхем A и B и все элементы подсхемы C. Конструкция схемы $\mathsf{U}^\star_n$ условно изображена на рис. 6.3.1.

Оценим сложность и глубину схемы $\mathsf{U}^\star_n$. Из конструкции схемы и предположения индукции легко видеть, что

$$\begin{aligned} L(\mathsf{U}^\star_n) = L(\mathsf{A}) + L(\mathsf{B}) + L(\mathsf{C}) &= (2^m - m - 1) + (2^l - l - 1) + \\ &+ (2^{m+l} - 2^m - 2^l + 1) = 2^{m+l} - 1. \end{aligned}$$

Так как для любого нечетного n большего единицы $\lceil \log_2 n \rceil = \lceil \log_2 (n+1) \rceil$ а для любого четного — $2\lceil n/2 \rceil = n$, то

$$\begin{aligned} D(\mathsf{U}^\star_n) &= \max(D(\mathsf{A}), D(\mathsf{B})) + 1 = D(\mathsf{U}^\star_m) + 1 = \\ &= \lceil \log_2 m \rceil + 1 = \left\lceil \log_2 \left(2 \left\lceil \frac{n}{2} \right\rceil \right) \right\rceil = \lceil \log_2 n \rceil \end{aligned}$$

Лемма доказана.

Заметим, что построенная в лемме 6.3.1 схема $\mathsf{U}^\star_n$ является минимальной схемой для системы всех монотонных дизъюнкций n переменных как по сложности, так и по глубине.

Лемма 6.3.2. *При $n \to \infty$ произвольная система дизъюнкций $F \in \mathcal{D}(m, n)$ может быть вычислена такой схемой S, что*

$$\begin{aligned} L(\mathsf{S}) &\le \frac{mn}{\log_2 m} \left(1 + \mathcal{O}\left(\frac{\log_2 \log_2 m}{\log_2 m} \right) \right), \\ D(\mathsf{S}) &\le \lceil \log_2 n \rceil + 1. \end{aligned}$$

Доказательство. Положим $q = \lfloor \log_2 m - \log_2 \log_2 m \rfloor$, $k = \lceil n/q \rceil$. Без ограничения общности полагаем, что $n \geq q$. Действительно, при $n < q$ число различных монотонных дизъюнкций n переменных меньше числа дизъюнкций системы F. Следовательно, F можно вычислить схемой $\mathsf{U}_n^\star$, состоящей не более чем из $2^n < \frac{m}{\log_2 m}$ дизъюнкторов.

Множество переменных $\{x_1, \ldots, x_n\}$ разобьем на k подмножеств I_j так, что

$$I_j = \{x_{(j-1)q+1}, \ldots, x_{jq}\}, \text{ при } j = 1, 2, \ldots, k-1,$$
$$I_k = \{x_{(k-1)q+1}, \ldots, x_n\}.$$

Каждую дизъюнкцию f_i из системы F разобьем на k дизъюнкций f_{ij} так, что f_{ij} является дизъюнкцией всех переменных входящих в f_i и одновременно принадлежащих множеству I_j, т. е.

$$f_i(x_1, \ldots, x_n) = \left(\bigvee_{j=1}^{k-1} f_{ij}(x_{(j-1)q+1}, \ldots, x_{jq}) \right) \vee f_{ik}(x_{(k-1)q+1}, \ldots, x_n).$$

Из дизъюнкций f_{ij} сформируем k новых систем $F_1, \ldots, F_k$ так, что для каждого j

$$F_j = \{f_{1j}, \ldots, f_{mj}\}.$$

Каждую систему F_j вычислим собственной схемой S_j. В качестве схемы S_j, $j = 1, 2, \ldots, k-1$, возьмем экземпляр построенной в лемме 6.3.1 схемы $\mathsf{U}_q^\star$, входы которой подключены к переменным $x_{(j-1)q+1}, \ldots, x_{jq}$. Последней схемой S_k будет схема $\mathsf{U}_{n-(k-1)q}^\star$, входы которой подключены к переменным $x_{(k-1)q+1}, \ldots, x_n$. Легко видеть, что

$$L(\mathsf{S}_j) \leq 2^q \leq \frac{m}{\log_2 m}, \quad D(\mathsf{S}_j) \leq \lceil \log_2 q \rceil. \tag{6.3.1}$$

Поэтому общая сложность всех схем S_j не больше чем

$$\sum_{j=1}^{k} \mathsf{S}_j \leq \frac{m}{\log_2 m} \left\lceil \frac{n}{\log_2 m - \log_2 \log_2 m} \right\rceil \leq \frac{mn}{\log_2^2 m} \left(1 + \mathcal{O}\left(\frac{\log_2 \log_2 m}{\log_2 m} \right) \right). \tag{6.3.2}$$

Теперь при любом i дизъюнкция f_i легко вычисляется равномерным дизъюнктором с не более чем k входами: j-й вход этого дизъюнктора подключен к выходу схемы S_j, вычисляющему дизъюнкцию f_{ij}. Сложность такого дизъюнктора не превосходит $k-1$, а глубина $\lceil \log_2 k \rceil$. Следовательно, сложность схемы, состоящей из всех этих дизъюнкторов, не больше чем

$$m(k-1) \leq m \left\lceil \frac{n}{\log_2 m - \log_2 \log_2 m} \right\rceil \leq \frac{mn}{\log_2 m} \left(1 + \mathcal{O}\left(\frac{\log_2 \log_2 m}{\log_2 m} \right) \right), \tag{6.3.3}$$

а ее глубина не превосходит

$$\lceil \log_k \rceil = \lceil \log_2 n/q \rceil. \tag{6.3.4}$$

Из (6.3.3) и (6.3.2) следует, что

$$L(\mathsf{S}) \leq \frac{mn}{\log_2 m} \left(1 + \mathcal{O}\left(\frac{\log_2 \log_2 m}{\log_2 m} \right) \right).$$

Теперь оценим глубину схемы S. Величины n и q представим в виде произведений $\alpha 2^s$ и $\beta 2^t$, где s и t — целые, а для постоянных α и β справедливы неравенства $\frac{1}{2} < \alpha, \beta \leq 1$. Тогда $\lceil n/q \rceil = \frac{\alpha}{\beta} 2^{s-t}$ и из второго неравенства (6.3.1) и (6.3.4) легко следует, что

$$D(\mathsf{S}) \leq \lceil \log_2 q \rceil + \lceil \log_2 \lceil n/q \rceil \rceil \leq$$
$$\leq t + s - t + \lceil \log_2 \alpha/\beta \rceil \leq s + 1 = \lceil \log_2 n \rceil + 1.$$

Лемма доказана.

Пример 6.3.1. Используя метод, примененный в доказательстве леммы 6.3.2, покажем, что произвольная система F, состоящая из десяти дизъюнкций, зависящих от десяти переменных $x_1, \ldots, x_{10}$, может быть вычислена схемой, которая состоит не более чем из 40 дизъюнкторов.

Каждую дизъюнкцию f_i из вычисляемой системы F разобьем на четыре дизъюнкции f_{i1}, f_{i2}, f_{i3} и f_{i4} так, что первая дизъюнкция будет зависет (не обязательно существенно) от переменных x_1, x_2 и x_3, вторая — от переменных x_4, x_5 и x_6, третья — от x_7 и x_8, четвертая — от x_9 и x_10. Затем вычислим все монотонные дизъюнкции переменных x_1, x_2 и x_3, все монотонные дизъюнкции переменных x_4, x_5 и x_6, все монотонные дизъюнкции переменных x_7 и x_8, и все монотонных дизъюнкции переменных x_9 и x_{10}. Из леммы 6.3.1 следует, что для вычисления первого и второго множеств дизъюнкций достаточно использовать по четыре дизъюнктора, а для вычисления двух последних — по одному дизъюнктору. Всего потребуется десять элементов. Легко видеть, что среди вычисленных дизъюнкций обязательно найдутся все возможные дизъюнкции f_{ij}. Поэтому теперь каждую дизъюнкцию f_i можно вычислить при помощи не более чем трех элементов, вычисляя дизъюнкцию $f_{i1} \vee f_{i2} \vee f_{i3} \vee f_{i4}$. Следовательно, для окончательного вычисления всех десяти дизъюнкций потребуется еще не более 30 элементов, которые вместе с использованными ранее десятью элементами и составят схему требуемой сложности. □

Задачи

6.3.1. Найти сложность и глубину системы одночленов P_n. Показать, что для этой системы существует схема, являющаяся одновременно минимальной по сложности и глубине.

6.3.2. Показать, что для системы дизъюнкций $\{x_1 \vee x_2 \vee x_3 \vee x_4, x_2 \vee x_3 \vee x_4\}$ не существует схемы, которая одновременно минимальна по сложности и глубине.

6.3.3. Оценить сверху сложность произвольной системы дизъюнкций из $\mathcal{D}(m,n)$, если:
a) $m = 20, n = 10$, b) $m = 20, n = 16$, c) $m = 20, n = 20$, d) $m = 24, n = 16$.

6.3.4. Оценить сверху сложность произвольного линейного оператора из $\mathcal{L}(m,n)$, если:
a) $m = 10, n = 20$, b) $m = 16, n = 16$, c) $m = 16, n = 24$, d) $m = 18, n = 32$.

6.3.5. Показать, что для каждого линейного булева (n,n)-оператора f при $n \to \infty$:
a) $L(f) \lesssim \frac{n^2}{\log_2 n}$; b) $L(f) \lesssim \frac{n^2}{2\log_2 n}$.

6.3.6. Показать, что для каждого булева линейного (n,n)-оператора f с верхнетреугольной матрицей при $n \to \infty$:
a) $L(f) \lesssim \frac{n^2}{2\log_2 n}$; b) $L(f) \lesssim \frac{n^2}{4\log_2 n}$.

6.3.7. Показать, что при $n \to \infty$ найдется такой линейный булев (n,n)-оператор f, что $L(f) \gtrsim \frac{n^2}{2\log_2 n}$.

6.3.8. Пусть $B \subseteq P_2(2)$ — полный базис в P_2. Показать, что $L_B(P_2(n)) = 2^{2^n} - n$.

6.4. Одновременное вычисление всех элементарных конъюнкций

Оценим сложность и глубину множества K_n, состоящего из всех элементарных конъюнкций вида $x_1^{\sigma_1} \& \ldots \& x_n^{\sigma_n}$.

Лемма 6.4.1. *Существует схема K_n, вычисляющая все элементарные конъюнкции переменных $x_1, \ldots, x_n$ сложность и глубина которой равны*

$$L(\mathsf{K}_n) = 2^n + \mathcal{O}\left(2^{n/2}\right), \quad D(\mathsf{K}_n) = \lceil \log_2 n \rceil.$$

Доказательство. Схемы K_n, удовлетворяющие равенствам леммы определим индуктивно. В основание индукции положим схемы K_2 и K_3, вычисляющие все элементарные

конъюнкции двух и трех переменных. Схема K_2 состоит из четырех элементов, вычисляющих функции: $x\&y$, $\overline{x}\&y$, $x\&\overline{y}$, $\overline{x}\&\overline{y}$. Очевидно, что ее сложность равна четырем, а глубина единице. Схема K_3 состоит из подсхемы A, являющейся экземпляром схемы K_2, четырех элементов, каждый из которых реализует функцию $x\&y$, и четырех элементов, каждый из которых реализует функцию $x\&\overline{y}$. В схеме K_3 входы подсхемы A подключены к переменным x_1 и x_2, каждый выход этой подсхемы умножается на переменную x_3 при помощи элементов $x\&y$ и на ее отрицание при помощи элементов $x\&\overline{y}$. Легко видеть, что сложность схемы K_3 равна 12, а глубина двум.

Положим $m = \lceil n/2 \rceil$, $l = \lfloor n/2 \rfloor$. Опишем конструкцию схемы K_n. Эта схема состоит из трех подсхем A, B и C. Подсхема A является экземпляром схемы K_m, входы которой подключены к переменным $x_1, \ldots, x_m$. Подсхема B является экземпляром схемы K_l, входы которой подключены к переменным $x_{m+1}, \ldots, x_{m+l}$. Подсхема C состоит из $2^m \cdot 2^l = 2^n$ конъюнкторов. Каждый из этих конъюнкторов умножает один выход подсхемы A на один выход подсхемы B. Выходами схемы K_n являются все элементы подсхемы C.

Из конструкции схемы легко видеть, что

$$L(\mathsf{K}_n) = L(\mathsf{A}) + L(\mathsf{B}) + L(\mathsf{C}) \leq L(\mathsf{A}) + L(\mathsf{B}) + 2^n. \tag{6.4.1}$$

Индукцией по n покажем, что $L(\mathsf{K}_n) \leq \frac{3}{2}2^n$. Очевидно, что для схем K_2 и K_3 это неравенство справедливо. Далее полагая, что $n \geq 4$, $L(\mathsf{K}_m) \leq \frac{3}{2}2^m$ и $L(\mathsf{K}_l) \leq \frac{3}{2}2^l$, из (6.4.1) немедленно получаем

$$L(\mathsf{K}_n) \leq 2^n + \frac{3}{2}2^m + \frac{3}{2}2^l \leq \frac{3}{2}2^n.$$

Таким образом, для любого n справедливо неравенство $L(\mathsf{K}_n) \leq \frac{3}{2}2^n$. Подставляя это неравенство в (6.4.1) имеем

$$L(\mathsf{K}_n) \leq 2^n + \frac{3}{2}2^m + \frac{3}{2}2^l \leq 2^n + 3 \cdot 2^{n/2}.$$

Первое равенство леммы доказано. Второе равенство доказывается также как и в лемме 6.3.1. Лемма доказана.

Очевидно, что схемы K_n асимптотически минимальны по сложность и минимальны по глубине. Следовательно, для сложности и глубины системы K_n справедливы равенства

$$L(K_n) \sim 2^n, \qquad D(K_n) = \lceil \log_2 n \rceil.$$

Задачи

6.4.1. Найти $L(K_8))$ и $D(K_8)$.

6.4.2. Оценить сверху $L(K_{2^n})$ если: a) $n = 5$; b) $n = 10$; c) $n = 25$.

6.4.3. Показать, что $L_{\{\vee,\neg\}}(K_n) \sim 2^n$.

6.4.4. Показать, что $L_B(K_n) \sim 2^n$ для любого полного базиса B.

6.4.5. Функция $f(x_1, \ldots, x_n, y_0, \ldots, y_{2^n-1})$ называется функцией выбора, если

$$f(\alpha_1, \ldots, \alpha_n, y_0, \ldots, y_{2^n-1}) = y_{|(\alpha_1,\ldots,\alpha_n)|}.$$

Оценить сложность и глубину функции выбора.

6.4.6. Оператор $f(x_1, \ldots, x_n, y_0, \ldots, y_{2^n-1}) = (f_0, \ldots, f_{2^n-1})$ называется оператором сдвига, если

$$f(\alpha_1, \ldots, \alpha_n, y_0, \ldots, y_{2^n-1}) = (y_k, y_{k+1}, \ldots, y_{2^n-1}, 0, \ldots, 0),$$

где $k = |(\alpha_1, \ldots, \alpha_n)|$. Оценить сложность и глубину оператора сдвига.

6.5. Асимптотически минимальный метод вычисления булевых функций

В начале четвертой главы в теореме 4.1.3 было установлено, что сложность почти каждой n-местной булевой функции асимптотически не меньше чем $\frac{2^n}{n}$. Ниже доказывается, что при $n \to \infty$ каждая n-местная булева функция может быть вычислена схемой, сложность которой асимптотически не превосходит $\frac{2^n}{n}$, т. е. для почти каждой n-местной булевой функции устанавливается асимптотически точное значение ее сложности.

Теорема 6.5.1. *Пусть $n \to \infty$. Тогда:*

(i) для почти каждой булевой функции f, зависящей от n переменных

$$L(f) \geq \frac{2^n}{n}, \quad D(f) \geq n - \log_2 n;$$

(ii) для каждой булевой функции f, зависящей от n переменных

$$L(f) \leq \frac{2^n}{n}\left(1 + \mathcal{O}\left(\frac{\log_2 n}{n}\right)\right), \quad D(f) \leq n + \log_2 \log_2 n + 4.$$

Доказательство. *(i)* Первое неравенство утверждения *(i)* является простым следствием теоремы 4.1.2. Второе неравенство является непосредственным следствием первого и задачи 4.1.1.

(ii) Для каждого $i = 0, 1, \ldots, 2^k - 1$ положим

$$f_i(x_{k+1}, \ldots, x_n) = f(\sigma_1, \ldots, \sigma_k, x_{k+1}, \ldots, x_n),$$

где $i = \sum_{t=1}^{k} \sigma_t 2^{k-t}$. Функцию f разложим по первым k переменным:

$$f(x_1, \ldots, x_n) = \bigvee_{\sigma_1 \ldots \sigma_k} f_i(x_{k+1}, \ldots, x_n) \cdot x_1^{(\sigma_1)} \ldots x_k^{(\sigma_k)}. \tag{6.5.1}$$

Каждую функцию f_i из (6.5.1) в свою очередь разложим по оставшимся $n-k$ переменным. Тогда

$$f_i(x_{k+1}, \ldots, x_n) = \bigvee_{\sigma_{k+1} \ldots \sigma_n} f_i(\sigma_{k+1}, \ldots, \sigma_n) \cdot x_{k+1}^{(\sigma_{k+1})} \ldots x_n^{(\sigma_n)},$$

или, учитывая определение функции f_i,

$$f_i(x_{k+1}, \ldots, x_n) = \bigvee_{\sigma_{k+1} \ldots \sigma_n} f(\sigma_1, \ldots, \sigma_n) \cdot x_{k+1}^{(\sigma_{k+1})} \ldots x_n^{(\sigma_n)}, \tag{6.5.2}$$

где как и ранее $i = \sum_{t=1}^{k} \sigma_t 2^{n-t}$. Для всех целых i, j таких, что $0 \leq i \leq 2^k - 1$ и $0 \leq j \leq 2^{n-k} - 1$, положим

$$y_j = x_{k+1}^{(\sigma_{k+1})} \cdot \ldots \cdot x_n^{(\sigma_n)}, \qquad f_{i,j} = f_i(\sigma_1, \ldots, \sigma_k),$$

где $j = \sum_{t=k+1}^{n} \sigma_t 2^{n-t}$.

Теперь рассмотрим систему $\{h_0, \ldots, h_{2^k-1}\}$, состоящую из дизъюнкций

$$h_i = f_{i,0} y_0 \vee \ldots \vee f_{i,j} y_j \vee \ldots \vee f_{i,2^{n-k}-1} y_{2^{n-k}-1}. \tag{6.5.3}$$

Заметим, что коэффициент $f_{i,j}$ этой системы равен значению функции f на таком наборе $(\sigma_1 \ldots \sigma_n)$, что $i = \sum_{t=1}^{k} \sigma_t 2^{k-t}$ и $j = \sum_{t=k+1}^{n} \sigma_t 2^{n-t}$. Поэтому подставляя функции y_j и постоянные $f_{i,j}$ в (6.5.2) видим, что для каждой функции f_i из разложения (6.5.1) справедливо представление

$$f_i(x_{k+1}, \ldots, x_n) = \bigvee_{j=0}^{2^{n-k}-1} f_{i,j} y_j = h_i(y_0, \ldots, y_{2^{n-k}-1}).$$

Следовательно,

$$f(x_1,\ldots,x_n)=\bigvee_{i=0}^{2^k-1} h_i\cdot x_1^{(\sigma_1)}\ldots x_k^{(\sigma_k)}.$$

Воспользуемся последним равенством для построения вычисляющей функцию f схемы S. Эта схема состоит из пяти подсхем (соединение подсхем изображено на рисунке 6.5.1) устроенных следующим образом.

1. Подсхема S_1 является экземпляром схемы K_k, построенной в лемме 6.4.1, и вычисляет все элементарные конъюнкции вида $x_1^{(\sigma_1)}\ldots x_k^{(\sigma_k)}$. Очевидно, что

$$L(\mathsf{S}_1)\le 2^k+\mathcal{O}\left(2^{k/2}\right),\quad D(\mathsf{S}_1)\le\lceil\log_2 k\rceil.$$

2. Подсхема S_2 является экземпляром схемы K_{n-k}, построенной в лемме 6.4.1, и вычисляет все элементарные конъюнкции вида $x_{k+1}^{(\sigma_{k+1})}\ldots x_n^{(\sigma_n)}$. Очевидно, что

$$L(\mathsf{S}_2)\le 2^{n-k}+\mathcal{O}\left(2^{(n-k)/2}\right),\quad D(\mathsf{S}_2)\le\lceil\log_2(n-k)\rceil.$$

3. Подсхема S_3 вычисляет систему дизъюнкций $\{h_0,\ldots,h_{2^k-1}\}$, используя функции y_j, вычисленные подсхемой S_2. Конструкция S_3 аналогична конструкции схемы из леммы 6.3.2[1)]. Из этой леммы следует, что сложность и глубина схемы S_3 удовлетворяют соотношениям

$$L(\mathsf{S}_3)\le\frac{2^n}{k}\left(1+\mathcal{O}\left(\frac{\log_2 k}{k}\right)\right),\quad D(\mathsf{S}_3)\le n-k+1.$$

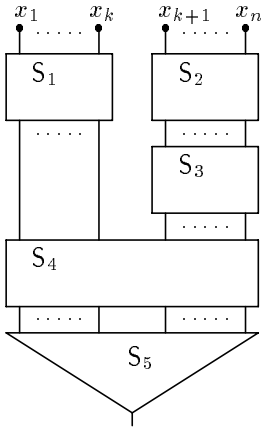


Рис. 6.5.1

4. Подсхема S_4 умножает функции, вычисленные подсхемой S_3, на элементарные конъюнкции, вычисленные подсхемой S_1. Очевидно, что

$$L(\mathsf{S}_4)\le 2^k,\quad D(\mathsf{S}_4)=1.$$

5. Подсхема S_5 вычисляет дизъюнкцию произведений, вычисленных подсхемой S_4. Легко видеть, что

$$L(\mathsf{S}_5)=2^k-1,\quad D(\mathsf{S}_5)=k.$$

Положим $k=\lfloor n-2\log_2 n\rfloor$. При выбранном значении параметра k оценим глубину и сложность схемы S. Из конструкции схемы S легко видеть, что

$$\begin{aligned}D(\mathsf{S})&\le\max\Big\{D(\mathsf{S}_1),D(\mathsf{S}_2)+D(\mathsf{S}_3)\Big\}+D(\mathsf{S}_4)+D(\mathsf{S}_5)\le\\&\le D(\mathsf{S}_2)+D(\mathsf{S}_3)+D(\mathsf{S}_4)+D(\mathsf{S}_5)\le\\&\le\lceil\log_2(n-k)\rceil+(n-k+1)+1+k\le n+\log_2\log_2 n+4.\end{aligned}$$

Сложность схемы S равна сумме сложностей подсхем $\mathsf{S}_1,\ldots,\mathsf{S}_5$. Легко убедиться, что при выбранном значении параметра k и $n\to\infty$ сложности подсхем S_1, S_2, S_4 и S_5 есть $\mathcal{O}\left(\frac{2^n}{n^2}\right)$. Следовательно,

$$L(\mathsf{S})\le L(\mathsf{S}_3)+\mathcal{O}\left(\frac{2^n}{n^2}\right),$$

и после несложных преобразований имеем

$$L(\mathsf{S})\le\frac{2^n}{n}\left(1+\mathcal{O}\left(\frac{\log_2 n}{n}\right)\right).$$

Теорема доказана.

[1)] Вместо леммы 6.3.2 можно воспользоваться леммой 6.2.2, изменив соответствующим образом определяемое ниже значение параметра k.

Доказанная теорема легко обобщается на случай булевых операторов. Соответствующую теорему о сложности операторов приведем без доказательства, которое во многом аналогично доказательству теоремы 6.5.1.

Теорема 6.5.2. *Пусть $n \to \infty$. Тогда:*
(i) для почти каждого булевого (m,n)-оператора f

$$L(f) \geq \frac{2^n m}{n + \log_2 m}, \quad D(f) \geq n - \log_2 n;$$

(ii) для каждого булевого (m,n)-оператора f

$$L(f) \leq \frac{2^n m}{n + \log_2 m}\left(1 + \mathcal{O}\left(\frac{\log_2 n}{n}\right)\right), \quad D(f) \leq n + \log_2 \log_2 n + 4.$$

Также без доказательства приведем теорему о сложности почти всех частичных булевых функций. Доказательство этой теоремы значительно сложнее доказательства двух предыдущих теорем.

Теорема 6.5.3. *Пусть $n \to \infty$, $D \subseteq \mathbb{B}^n$, $|D| \geq n \log_2 n$. Тогда:*
(i) для почти каждой частичной булевой функции f, определенной на области D

$$L(f) \geq \frac{|D|}{\log_2 |D|};$$

(ii) для каждой частичной булевой функции f, определенной на области D

$$L(f) \lesssim \frac{|D|}{\log_2 |D|} + \mathcal{O}(n).$$

2. Воспользуемся теоремой 6.5.1 и результатами предыдущей главы для оценки сложности произвольной симметрической булевой функции. Имеет место следующий результат.

Теорема 6.5.4. *Пусть $n \to \infty$. Тогда для любой симметрической булевой функции, зависящей от n переменных, существует вычисляющая ее схема S_n, для сложности и глубины которой справедливы неравенства*

$$L(\mathsf{S}_n) \lesssim 6n, \quad D(\mathsf{S}_n) \lesssim 6 \log_2 n.$$

Доказательство. Пусть $f(x_1, \ldots, x_n)$ — симметрическая булева функция, $k = \lceil \log_2(n+1) \rceil$, $W(x_1, \ldots, x_n)$ — булев (k,n)-оператор подсчета. Введем булеву функцию $g(x'_1, \ldots, x'_k)$ так, что $g(\beta_1, \ldots, \beta_k) = f(\alpha_1, \ldots, \alpha_n)$ если $\sum_{i=1}^{k} 2^{i-1}\beta_i = \sum_{i=1}^{n} \alpha_i$. Тогда $f(x_1, \ldots, x_n) = g(W(x_1, \ldots, x_n))$. Поэтому схему, вычисляющую функцию $f(x_1, \ldots, x_n)$, можно составить из двух последовательно соединенных подсхем A и B. Подсхема A вычисляет вес набора $(x_1, \ldots, x_n)$. Подсхема B вычисляет функцию $g(x'_1, \ldots, x'_k)$, а ее входы подключены к выходам подсхемы A. Следовательно,

$$L(\mathsf{S}_n) \leq L(\mathsf{A}) + L(\mathsf{B}), \quad D(f) \leq D(\mathsf{A}) + D(\mathsf{B}). \tag{6.5.4}$$

В силу теоремы 6.5.1 при $n \to \infty$ для сложности и глубины подсхемы B справедливы неравенства

$$L(\mathsf{B}) \lesssim \frac{2^k}{k} = o(n), \quad D(\mathsf{B}) \sim k \sim \log_2 n. \tag{6.5.5}$$

Подсхема A вычисляет сумму одноразрядных двоичных чисел $x_1, \ldots, x_n$ и состоит из трех последовательно соединенных подсхем $\mathsf{A}_1, \mathsf{A}_2$ и A_3. Рассмотрим эти подсхемы и для каждой подсхемы оценим ее сложность и глубину.

1. Зафиксируем целое m так, чтобы $2^m = \mathcal{O}(\log_2 n)$. Числа $x_1, \ldots, x_n$ разобъем на $\left\lceil \frac{n}{2^m} \right\rceil$ подмножеств A_j, каждое из которых кроме может быть последнего, содержит по 2^m чисел. Подсхема A_1 состоит из $\left\lceil \frac{n}{2^m} \right\rceil$ независимых счетчиков C_{2^m}, вычисляющих для каждого A_j сумму входящих в него чисел. Следовательно,

$$L(\mathsf{A}_1) \le 6 \cdot 2^m \left(\frac{n}{2^m} + 1 \right) = 6n + \mathcal{O}(\log_2 n),$$
$$D(\mathsf{A}_1) \le m^2 = \mathcal{O}\left((\log_2 \log_2 n)^2 \right).$$

2. Подсхема A_2 вычисляет двойное число $\boldsymbol{z}$, равное сумме всех чисел вычисленных подсхемой A_1. Так как каждое из этих чисел является $(m+1)$-разрядным числом, а их количество не превосходит $\frac{n}{2^m} + 1$, то для сложности и глубины A_2 справедливы неравенства

$$L(\mathsf{A}_2) \le 10 \left(\frac{n}{2^m} + 1 \right)(m+1) = \mathcal{O}\left(\frac{n \log_2 \log_2 n}{\log_2 n} \right) = o(n),$$
$$D(\mathsf{A}_2) \le 5 \left(\log_2 \left(\frac{n}{2^m} \right) + 1 \right) + 1 < 5 \log_2 n.$$

3. Подсхема A_3 является экземпляром схемы $\overline{\Sigma}_k^*$ (см. стр. 99) и вычисляет значение $\boldsymbol{z}$, т.е. находит разность положительной и отрицательной компонент этого двойного числа. Из (5.1.6) имеем

$$L(\mathsf{A}_3) = L(\overline{\Sigma}_k^*) = \mathcal{O}(\log_2 k), \quad D(\mathsf{A}_3) = D(\overline{\Sigma}_k^*) = \mathcal{O}(\log_2 \log_2 n).$$

Таким образом, суммируя сложности и глубины схем, перечисленных в пп. 1-3, видим, что при $n \to \infty$ для подсхемы A справедливы неравенства

$$L(\mathsf{A}) \lesssim 6n, \quad D(\mathsf{A}) \lesssim 5 \log_2 n.$$

Отсюда и из (6.5.4) и (6.5.5) получаем требуемые оценки сложности и глубины схемы S_n. Теорема доказана.

Задачи

6.5.1. Пусть $\mathbb{V}$ подпространство в $\mathbb{B}^n$ размерности k, $F_{\mathbb{V}}$ — подмножество $P_2(n)$, состоящее из всех функций равных нулю на наборах не принадлежащих $\mathbb{V}$. Показать, что при $k \to \infty$:

a) среди функций из $F_{\mathbb{V}}$ найдется такая функция f, что $L(f) \gtrsim \frac{2^k}{k}$;

b) для каждой функции f из $F_{\mathbb{V}}$ справедливо неравенство $L(f) \lesssim \frac{2^k}{k}$.

6.5.2. Пусть $\mathbb{V}$ подпространство в $\mathbb{B}^n$ размерности k, $F_{\mathbb{V}}$ — подмножество $P_2(n)$, состоящее из всех функций, постоянных на смежных классах пространства $\mathbb{B}^n$ по $\mathbb{V}$. Оценить $\max L(f)$, где максимум берется по всем функциям из $F_{\mathbb{V}}$ при условии, что $n - k \to \infty$.

6.5.3. Пусть $F(n, N)$ — подмножество $P_2(n)$, состоящее из всех функций равных нулю на наборах с номерами большими N, т.е. $f(\boldsymbol{\alpha}) = 0$ для каждой $f \in F(n, N)$ при всех $|\boldsymbol{\alpha}| > N$. Показать, что:

a) среди функций из $F(n, N)$ найдется такая функция f, что $L(f) \gtrsim \frac{N}{\log_2 N}$;

b) для каждой функции f из $F(n, N)$ справедливо неравенство $L(f) \lesssim \frac{N}{\log_2 N}$.

6.5.4. Пусть $F(n)$ — подмножество $P_2(n)$, состоящее из всех функций равных нулю на наборах четного веса. Показать, что:

a) среди функций из $F(n)$ найдется такая функция f, что $L(f) \ge \frac{2^{n-1}}{n}$;

b) для каждой функции f из $F(n)$ справедливо неравенство $L(f) \lesssim \frac{2^{n-1}}{n}$.

6.5.5. Пусть $G(n, m)$ — подмножество $P_2(n)$, состоящее из всех таких функций f, что $f(\boldsymbol{\alpha}) = 0$, если $|\boldsymbol{\alpha}| \not\equiv 0 \pmod m$. Оценить $\max L(f)$, где максимум берется по всем функциям из $G(n, m)$ при условии, что $m = o(n)$.

6.5.6. Показать, что при $n \to \infty$ для каждой функции f из $P_2(n)$ справедливо неравенство:

a) $L_{\{\vee,\neg\}}(f) \lesssim \frac{2^n}{n}$; b) $L_{\{\&,\neg\}}(f) \lesssim \frac{2^n}{n}$; c) $L_{\{\&,\oplus,1\}}(f) \lesssim \frac{2^n}{n}$;
d) $L_{|}(f) \lesssim \frac{2^n}{n}$; e) $L_{\{\downarrow\}}(f) \lesssim \frac{2^n}{n}$.

6.5.7. Пусть $B \subseteq P_2(2)$ и $[B] = P_2$. Показать, что при $n \to \infty$ для каждой функции f из $P_2(n)$ справедливо неравенство $L_{\{B\}}(f) \lesssim \frac{2^n}{n}$.

6.5.8. Оценить $\max L(f)$, где максимум берется по всем самодвойственным функциям n переменных.

6.5.9. Оценить $\max L(f)$, где максимум берется по всем булевым операторам $f : \mathbb{B}^n \to \mathbb{B}^n$ таким, что $\|f(x)\| = 1$ для всех $x \in \mathbb{B}^n$.

6.5.10. Показать, что для каждой функции $f \in P_2(n)$ веса N справедливо неравенство $L(f) \lesssim \frac{Nn}{\log_2 N}$.

6.5.11. Доказать теорему 6.5.2.

6.5.12. Доказать нижнюю оценку теоремы 6.5.3.

6.5.13. Функция $f(x_1, \ldots, x_{2n})$ симметрична относительно первых n переменных. Оценить $L(f)$.

6.5.14. Оценить сверху сложность произвольной симметрической булевой функции, зависящей от n переменных, если: a) $n = 3$, b) $n = 4$, c) $n = 8$, d) $n = 16$.

Глава 7.

Средняя сложность булевых функций

Важная особенность рассматриваемых в предыдущих главах схем из функциональных элементов заключается в том, что схемы выполняют одно и тоже число шагов при различных величинах подаваемых на их входы. По этой причине схемы являются хорошей моделью для изучения сложности вычислений в "худшем случае". Однако не всегда число шагов в худшем случае будет хорошо описывать ту или иную реальную ситуацию. Так, например, простая и естественная задача определения максимального значения в большом массиве булевых величин в худшем случае требует линейного по числу элементов массива числа шагов. Хотя средняя по всем возможным массивам сложность не зависит от размера массива и является постоянной величиной.

Изучению среднего случая при вычислении булевых функций посвящена настоящая глава. Булевы функции будем вычислять при помощи неветвящихся программ с условной остановкой (называемых далее просто неветвящимися программами). Эти программы являются естественным обобщением понятия схемы. Как и схемы, неветвящиеся программы не содержат операций условного перехода и косвенной адресации, но в отличие от схем, в программах есть возможность досрочного прекращения работы при выполнении определенного условия. Такие вычисления можно представить следующим образом. Вычисления выполняет процессор, снабженный памятью, состоящей из отдельных ячеек. Процессор способен вычислять некоторое количество двухместных элементарных функций; множество этих функций назовем базисом вычисления. Каждая ячейка памяти в любой момент времени доступна процессору как для чтения, так и для записи информации. Процессор работает под управление программы, являющейся последовательностью элементарных операторов двух видов. Каждый оператор первого вида вычисляется значение некоторой базисной функции, аргументами которой является содержимое определенных ячеек памяти. Вычисленный результат также помещается в одну из ячеек памяти. Оператор второго вида может прекратить выполнение программы. Каждый такой оператор имеет единственный аргумент — содержимое некоторой ячейки памяти. Если значение аргумента равно определенному фиксированному числу, например единице, то процессор прекращает работу. Если значение аргумента иное, то выполняется следующий оператор программы. В памяти выделяется множество особых ячеек содержимое которых после прекращения работы объявляется результатом работы программы. Естественной мерой сложности таких программ является среднее по всем возможным аргументам время работы.

Неветвящиеся программы обладают достаточно мощными вычислительными способностями, характерными для вычислительных систем высокого уровня. Поэтому их можно рассматривать в качестве математической модели высокоуровневых вычислений, выполняемых на универсальных компьютерах.

7.1. Неветвящиеся программы

1. Пусть $X = \{x_1, \ldots, x_n\}$ — множество независимых булевых переменных. Введем множество переменных $Y = \{\mathtt{y}_1, \ldots, \mathtt{y}_l\}$ и множество переменных $Z = \{\mathtt{z}_1, \ldots, \mathtt{z}_m\}$. Переменные из множества Y назовем внутренними, а переменные из множества Z — выходны-

ми переменными. Пусть, далее, $\mathtt{a} \in Y \cup Z$, $\mathtt{b}, \mathtt{c} \in X \cup Y \cup Z$, f — булева функция, зависящая не более чем двух переменных. *Функциональным оператором* $\mathtt{p}$ назовем выражение

$$\mathtt{p}: \quad \mathtt{a} = f(\mathtt{b}, \mathtt{c}).$$

Переменную $\mathtt{a}$ назовем *выходом* функционального оператора $\mathtt{p}$, а переменные $\mathtt{b}$, $\mathtt{c}$ — *входами* этого оператора. Пусть теперь $\mathtt{a} \in X \cup Y \cup Z$. *Оператором остановки* $\mathtt{p}$ назовем выражение

$$\mathtt{p}: \quad \mathtt{Stop}(\mathtt{a}).$$

Переменную $\mathtt{a}$ назовем входом оператора остановки $\mathtt{p}$.

Последовательность $\mathsf{P} = \mathtt{p}_1 \ldots \mathtt{p}_i \ldots \mathtt{p}_L$, состоящая из функциональных операторов и операторов остановки, называется *неветвящейся программой с условной остановкой*, если при любом $j \in \{1, 2, \ldots, L\}$ каждый вход оператора $\mathtt{p}_j$ есть либо независимая переменная, либо выход некоторого функционального оператора $\mathtt{p}_i$, где $i < j$.

Неветвящаяся программа работает в дискретные моменты времени $t = 0, 1, 2, \ldots$, не изменяет значения независимых переменных и изменяет значения внутренних и выходных переменных. Значения $\mathtt{y}_i(\boldsymbol{x}; t)$ внутренних переменных $\mathtt{y}_i$ и значения $\mathtt{z}_j(\boldsymbol{x}; t)$ выходных переменных $\mathtt{z}_j$ программы P в произвольный момент времени t на наборе независимых переменных $\boldsymbol{x} = (x_1, \ldots, x_n)$ определим индуктивно:

• В начальный момент времени $t = 0$ значения всех внутренних и выходных переменных считаем неопределенными;

• Если внутренняя переменная $\mathtt{y}_i$ (выходная переменная $\mathtt{z}_j$) не является выходом оператора $\mathtt{p}_t$, то положим

$$\mathtt{y}_i(\boldsymbol{x}; t) = \mathtt{y}_i(\boldsymbol{x}; t-1), \qquad \mathtt{z}_j(\boldsymbol{x}; t) = \mathtt{z}_j(\boldsymbol{x}; t-1);$$

• Если внутренняя переменная $\mathtt{y}_i$ (выходная переменная $\mathtt{z}_j$) является выходом оператора $\mathtt{p}_t$, а $\mathtt{b}(\boldsymbol{x}; t-1)$ и $\mathtt{c}(\boldsymbol{x}; t-1)$ — значения входов оператора $\mathtt{p}_t$ в момент времени $t-1$, то положим

$$\mathtt{y}_i(\boldsymbol{x}; t) = f_t(\mathtt{b}(\boldsymbol{x}; t-1), \mathtt{c}(\boldsymbol{x}; t-1)),$$
$$\mathtt{z}_j(\boldsymbol{x}; t) = f_t(\mathtt{b}(\boldsymbol{x}; t-1), \mathtt{c}(\boldsymbol{x}; t-1)).$$

Значением оператора $\mathtt{p}_t$ программы P на наборе независимых переменных $\boldsymbol{x} = (x_1, \ldots, x_n)$ назовем значение его выхода в момент времени t и обозначим через $\mathtt{p}_t(\boldsymbol{x})$.

Через $n(\mathtt{p})$ обозначим номер оператора $\mathtt{p}$ в программе P, т. е. $n(\mathtt{p}_i) = i$. Пусть $\mathtt{p}_{t_1}, \ldots, \mathtt{p}_{t_r}$ — все операторы остановки из P, причем $t_1 < \cdots < t_r$. Тогда через $\mathtt{s}_j$ будем обозначать j-й оператор остановки программы P, т. е. $\mathtt{s}_j \equiv \mathtt{p}_{t_j}$.

Функциональный оператор $\mathtt{p}_i$ (переменную x_l) назовем *нулевым аргументом* оператора остановки $\mathtt{s}_j$, $n(\mathtt{s}_j) = r$, и обозначим через $\mathtt{q}_j$, если:

(i) выход оператора $\mathtt{p}_i$ (переменная x_l) является входом оператора $\mathtt{s}_j$.

(ii) среди операторов $\mathtt{p}_t$, $i < t < r$, нет оператора, выход которого совпадает с выходом оператора $\mathtt{p}_i$.

Будем говорить, что k-й оператор остановки $\mathtt{s}_k$ останавливает вычисления программы P на наборе $\boldsymbol{x}$, если

$$\mathtt{q}_1(\boldsymbol{x}) = \cdots = \mathtt{q}_{k-1}(\boldsymbol{x}) = 0, \quad \mathtt{q}_k(\boldsymbol{x}) = 1.$$

Результат действия программы P на наборе $\boldsymbol{x}$ обозначим через $\mathsf{P}(\boldsymbol{x})$ и его l-ю компоненту $\mathsf{P}_l(\boldsymbol{x})$ определим следующим образом:

$$\mathsf{P}_l(\boldsymbol{x}) = \begin{cases} \mathtt{z}_l(\boldsymbol{x}; t_k), & \text{если } \mathtt{q}_1(\boldsymbol{x}) = \cdots = \mathtt{q}_{k-1}(\boldsymbol{x}) = 0, \ \ \mathtt{q}_k(\boldsymbol{x}) = 1, \\ \mathtt{z}_l(\boldsymbol{x}; L), & \text{если } \mathtt{q}_1(\boldsymbol{x}) = \cdots = \mathtt{q}_k(\boldsymbol{x}) = 0, \end{cases}$$

т.е. $\mathsf{P}_l(\boldsymbol{x})$ равно значению l-й выходной переменной $\mathtt{z}_l$ в момент остановки программы. Легко видеть, что

$$\begin{aligned}\mathsf{P}_l(\boldsymbol{x}) &= \mathsf{q}_1(\boldsymbol{x})\mathtt{z}_l(\boldsymbol{x};t_1) \vee \overline{\mathsf{q}}_1(\boldsymbol{x})\mathsf{q}_2(\boldsymbol{x})\mathtt{z}_l(\boldsymbol{x};t_2) \vee \ldots \\ &\ldots \vee \overline{\mathsf{q}}_1(\boldsymbol{x})\overline{\mathsf{q}}_2(\boldsymbol{x}) \cdots \overline{\mathsf{q}}_{k-1}(\boldsymbol{x})\mathsf{q}_k(\boldsymbol{x})\mathtt{z}_l(\boldsymbol{x};t_k) \vee \ldots \\ &\ldots \vee \overline{\mathsf{q}}_1(\boldsymbol{x})\overline{\mathsf{q}}_2(\boldsymbol{x}) \cdots \overline{\mathsf{q}}_{r-1}(\boldsymbol{x})\mathsf{q}_r(\boldsymbol{x})\mathtt{z}_l(\boldsymbol{x};t_r) \vee \overline{\mathsf{q}}_1(\boldsymbol{x})\overline{\mathsf{q}}_2(\boldsymbol{x}) \cdots \overline{\mathsf{q}}_r(\boldsymbol{x})\mathtt{z}_l(\boldsymbol{x};L).\end{aligned} \tag{7.1.1}$$

Пример 7.1.1. Рассмотрим три разных программы, вычисляющих дизъюнкцию четырех переменных. Операторы каждой из этих программ расположены в одном из следующих вертикальных столбцов:

p_1 :	$\mathtt{z} = 1$	$\mathtt{z} = x_1 \vee x_2$	$\mathtt{y}_1 = x_1 \vee x_2$
p_2 :	$\mathtt{Stop}(x_1)$	$\mathtt{Stop}(\mathtt{z})$	$\mathtt{y}_2 = x_3 \vee x_4$
p_3 :	$\mathtt{Stop}(x_2)$	$\mathtt{z} = x_3 \vee x_4$	$\mathtt{z} = \mathtt{y}_1 \vee \mathtt{y}_2$
p_4 :	$\mathtt{Stop}(x_3)$		
p_5 :	$\mathtt{Stop}(x_4)$		
p_6 :	$\mathtt{z} = 0$		

Первая программа состоит из шести операторов и работает следующим образом. Сначала выходной переменной присваивается значение единица. Затем последовательно проверяются условия равенства единице переменных x_i. Если первая переменная равна единице, то первый оператор остановки прекращает работу программы. Если $x_1 = 0$, то начинает работу второй оператор остановки, который в свою очередь прекращает работу программы, если $x_2 = 1$. Если $x_2 = 0$, то аналогичным образом работает третий оператор остановки, а затем, если $x_3 = 0$, — четвертый. Если ни один из операторов остановки не прекратил работу программы, т.е. если все переменные равны нулю, то выполняется последний оператор программы, который присваивает выходной переменной нулевое значение. Используя (7.1.1) убеждаемся, что первая программа действительно вычисляет дизъюнкцию четырех переменных:

$$\begin{aligned}\mathsf{P}(\boldsymbol{x}) &= x_1 \cdot 1 \vee \overline{x}_1 x_2 \cdot 1 \vee \overline{x}_1\overline{x}_2 x_3 \cdot 1 \vee \overline{x}_1\overline{x}_2\overline{x}_3 x_4 \cdot 1 \vee \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_4 \cdot 0 = \\ &= x_1 \vee \overline{x}_1 x_2 \vee \overline{x}_1\overline{x}_2 x_3 \vee \overline{x}_1\overline{x}_2\overline{x}_3 x_4 = x_1 \vee \overline{x}_1 x_2 \vee \overline{x}_1\overline{x}_2(x_3 \vee \overline{x}_3 x_4) = \\ &= x_1 \vee \overline{x}_1 x_2 \vee \overline{x}_1\overline{x}_2(x_3 \vee x_4) = x_1 \vee \overline{x}_1(x_2 \vee \overline{x}_2(x_3 \vee x_4)) = \\ &= x_1 \vee \overline{x}_1(x_2 \vee x_3 \vee x_4) = x_1 \vee x_2 \vee x_3 \vee x_4.\end{aligned}$$

Вторая программа состоит из трех операторов. Аналогичным образом для ее значения имеем

$$\mathsf{P}(\boldsymbol{x}) = (x_1 \vee x_2)(x_1 \vee x_2) \vee \overline{(x_1 \vee x_2)}(x_3 \vee x_4) = x_1 \vee x_2 \vee x_3 \vee x_4.$$

Третья программа состоит только из функциональных операторов и, поэтому, по существу является схемой из функциональных элементов. Очевидно, что она так же вычисляет дизъюнкцию переменных x_1, x_2, x_3 и x_4. □

2. *Сложностью* $C(\mathsf{P})$ программы P назовем число операторов этой программы. *Временем работы* $T_{\mathsf{P}}(\boldsymbol{x})$ программы P на наборе переменных $\boldsymbol{x}$ назовем минимальное $n(\mathsf{s}_j)$ такое, что $\mathsf{q}_j(\boldsymbol{x}) = 1$, т.е. это число операторов, выполненных до остановки программы. Если все $\mathsf{q}_j(\boldsymbol{x}) = 0$, то выполняются все операторы программы и в этом случае $T_{\mathsf{P}}(\boldsymbol{x}) = C(\mathsf{P})$. Величину

$$T(\mathsf{P}) = 2^{-n}\sum T_{\mathsf{P}}(\boldsymbol{x}),$$

где суммирование производится по всем двоичным наборам длины n, назовем *средним временем работы* программы P. Если для некоторого булевого оператора f и любого двоичного набора x справедливо равенство $f(\boldsymbol{x}) = \mathsf{P}(\boldsymbol{x})$, то будем говорить, что программа P вычисляет оператор f. Величину

$$T(f) = \min T(\mathsf{P}),$$

где минимум берется по всем программам, вычисляющим f, назовем *средним временем вычисления* (*средней сложностью*) оператора f. Программу P, вычисляющую оператор f, для которой справедливо равенство $T(\mathsf{P}) = T(f)$, назовем *минимальной* программой. Величину

$$C(f) = \min C(\mathsf{P}),$$

где минимум берется по всем программам, вычисляющим f, назовем *программной сложностью* оператора f. Величина $C(f)$ характеризует время, необходимое для вычисления f в худшем случае, поэтому $C(f)$ так же будем называть сложностью в худшем случае. Отметим, что неветвящаяся программа, не содержащая операторы остановки и вычисляющая функцию, отличную от независимой переменной, является обычной схемой из функциональных элементов, базис которой состоит из всех не более чем двухместных булевых функций. Поэтому средняя сложность любой булевой функции $f(x_1, \ldots, x_n)$, существенно зависящей не менее чем от двух переменных, не меньше ее схемной сложности, т. е.

$$T(f(x_1, \ldots, x_n)) \le L(f(x_1, \ldots, x_n)).$$

Пример 7.1.2. Рассмотрим две программы P_1 и P_2, вычисляющие систему из двух функций — дизъюнкции и конъюнкции четырех переменных. В этих программах

p_1 : $\mathtt{z}_1 = x_1 \oplus x_2$	$\mathtt{z}_1 = x_1 \vee x_2$
p_2 : $\mathtt{z}_2 = 0$	$\mathtt{z}_1 = \mathtt{z}_1 \vee x_3$
p_3 : $\mathtt{Stop}(\mathtt{z}_1)$	$\mathtt{z}_1 = \mathtt{z}_1 \vee x_4$
p_4 : $\mathtt{z}_1 = x_3 \oplus x_4$	$\mathtt{z}_2 = x_1 \,\&\, x_2$
p_5 : $\mathtt{Stop}(\mathtt{z}_1)$	$\mathtt{z}_2 = \mathtt{z}_2 \,\&\, x_3$
p_6 : $\mathtt{z}_1 = x_1 \vee x_3$	$\mathtt{z}_2 = \mathtt{z}_2 \,\&\, x_4$
p_7 : $\mathtt{z}_2 = x_1 \,\&\, x_3$	

дизъюнкция вычисляется переменной $\mathtt{z}_1$, а конъюнкция — переменной $\mathtt{z}_2$. Легко видеть, что сложности первой и второй программ равны, соответственно, семи и шести. Найдем их средние времена работы. В первой программе первый оператор остановки прекращает ее работу на восьми наборах: (0100), (0101), (0110), (0111), (1000), (1001), (1010), (1011); второй оператор остановки — на четырех наборах: (0001), (0001), (1101), (1111); наконец, на оставшихся четырех наборах (0000), (0011), (1100) и (1111) выполняются все операторы программы. Поэтому для среднего времени работы программы P_1 имеем

$$T(\mathsf{P}_1) = \frac{1}{16}\Big(3 \cdot 8 + 5 \cdot 4 + 7 \cdot 5\Big) = \frac{9}{2}.$$

Вторая программа состоит только из функциональных операторов, и, следовательно, среднее время работы этой программы совпадает с ее сложностью, т. е. $T(\mathsf{P}_2) = 6$. $\square$

3. Пусть переменная $\mathtt{a}$ является входом оператора p_i программы P, а переменная $\mathtt{b}$ — выходом функционального оператора p_j этой программы. Будем говорить, что на входе оператора p_i вычисляется булева функция $f(\boldsymbol{x})$, если

$$\mathtt{a}(\boldsymbol{x}; i-1) = f(\boldsymbol{x})$$

при всех значениях независимых переменных, при которых значение переменной $\mathtt{a}$ определено. Аналогичным образом скажем, что на выходе оператора p_j вычисляется булева функция $h(\boldsymbol{x})$, если

$$\mathtt{b}(\boldsymbol{x}; j) = h(\boldsymbol{x})$$

при всех значениях независимых переменных, при которых значение переменной $\mathtt{b}$ определено.

Будем говорить, что два оператора p_i и p_j одной программы имеют общий вход, если на их входах вычисляются одинаковые функции.

Среди всех программ выделим множество приведенных программы и покажем, что любая программа без увеличения сложности и среднего времени работы может быть преобразована в приведенную программу.

Программу P назовем *приведенной*, если:

- вход каждого оператора программы P не является тождественной постоянной;
- никакие два оператора остановки программы P не имеют общего входа.

Лемма 7.1.1. *Произвольная программа* P *может быть преобразована в приведенную программу* P' *так, что*:

(i) $T(\mathsf{P}') \le T(\mathsf{P}), \quad C(\mathsf{P}') \le C(\mathsf{P})$;

(ii) $\mathsf{P}'(x_1, \ldots, x_n) = \mathsf{P}(x_1, \ldots, x_n)$ *при всех* $(x_1, \ldots, x_n)$.

Доказательство. Пусть P — произвольная программа, $\boldsymbol{x} = (x_1, \ldots, x_n)$ — набор независимых переменных, $\mathsf{p}_t : \mathsf{a} = g(\mathsf{b}, \mathsf{c})$ — оператор, на первом входе которого вычисляется постоянная функция, т. е. $\mathsf{b}(\boldsymbol{x}; t-1) = \mathrm{const}$. В этом случае $\mathsf{a}(\boldsymbol{x}; t)$ зависит только от $\mathsf{c}(\boldsymbol{x}; t-1)$, и, следовательно, существует функция h такая, что $h(\mathsf{c}(\boldsymbol{x}; t-1)) = g(\mathsf{b}(\boldsymbol{x}; t-1), \mathsf{c}(\boldsymbol{x}; t-1))$. Заменяя в P оператор p_t оператором $p'_t : \mathsf{a} = h(\mathsf{c})$, получаем программу, удовлетворяющую условиям (i) и (ii).

Допустим теперь, что в программе P найдутся два оператора остановки p_i и p_j, $i < j$, с одним и тем же входом a. Если $\mathsf{a}(\boldsymbol{x}; i-1) = \mathsf{a}(\boldsymbol{x}; j-1) = 1$, то оператор p_i останавливает выполнение программы, и оператор p_j не выполняется. Если $\mathsf{a}(\boldsymbol{x}; j-1) = 0$, то оператор p_j не останавливает выполнение программы. Следовательно, оператор p_j может быть удален из программы P, и это ни как не скажется на ее работе. Лемма доказана.

Легко видеть, что все программы, рассмотренные в примерах 7.1.1 и 7.1.2, являются приведенными. Далее будем рассматривать только приведенные программы.

Задачи

7.1.1. Найти среднее время работы каждой программы из примера 7.1.1.

7.1.2. Доказать, что программа P_1 из примера 7.1.2 вычисляет дизъюнкцию и конъюнкцию четырех переменных.

7.1.3. Показать, что средняя сложность системы, состоящей из n-местных дизъюнкции и конъюнкции, зависящих от одних и тех же переменных, не превосходит пяти.

7.1.4. Показать, что в любой приведенной программе с n входами число операторов остановки не превосходит $\frac{1}{2}(L+n)$, где L — сложность программы.

7.2. Функции трех переменных

Эффекты, связанные с возможностью досрочного прекращения вычислений, начинают проявляться уже при вычислении булевых функций трех переменных. Ранее (параграф 4.2, стр. 77) было показано, что схемная сложность каждой булевой функции трех переменных не превосходит четырех, причем существуют функции, например функция голосования $\tau_2(x_1, x_2, x_3)$ (см. параграф 4.4, стр. 86), сложность которых равна четырем. Далее покажем, что средняя сложность любой булевой функции трех переменных не превосходит $2\frac{1}{2}$.

Пусть f — произвольная булева функция трех переменных. Разложим f по первой переменной

$$f(x_1, x_2, x_3) = x_1 f_1(x_2, x_3) \vee \overline{x}_1 f_2(x_2, x_3).$$

Легко видеть, что программа P

$$\begin{aligned} &\mathsf{p}_1 : \mathsf{z} = f_1(x_2, x_3) \\ &\mathsf{p}_2 : \mathtt{Stop}(x_1) \\ &\mathsf{p}_3 : \mathsf{z} = f_2(x_2, x_3) \end{aligned}$$

вычисляет функцию f. Действительно, в соответствии с определением функции, вычисляемой неветвящейся программой с условной остановкой, для $\mathsf{P}(\boldsymbol{x})$ справедливы равенства

$$\mathsf{P}(\boldsymbol{x}) = \mathsf{q}_1(\boldsymbol{x})\mathsf{z}(\boldsymbol{x};1) \vee \overline{\mathsf{q}}_1(\boldsymbol{x})\mathsf{z}(\boldsymbol{x};3) = x_1 f_1(x_2,x_3) \vee \overline{x}_1 f_2(x_2,x_3).$$

На четырех наборах — (100), (101), (110), (111) — программа P выполняет два действия, на остальных четырех наборах — три. Поэтому

$$T(f) \le T(\mathsf{P}) = \frac{1}{8}\,(2\cdot 4 + 3\cdot 4) = 2\frac{1}{2}.$$

Теперь покажем, что средняя сложность функции голосования $\tau_2(x_1,x_2,x_3)$ равна $2\frac{1}{2}$. Пусть P — программа, вычисляющая $\tau_2(x_1,x_2,x_3)$ с минимальным средним временем. Если первый оператор остановки s_1 является третьим оператором P, то очевидно, что $T(\mathsf{P}) \ge 3$. Так как в любой программе на первом месте должен стоять функциональный оператор, то далее полагаем, что s_1 является вторым оператором P. Теперь для доказательства неравенства $T(\tau_2(x_1,x_2,x_3)) \ge 2\frac{1}{2}$ достаточно показать, что в P оператор остановки прекращает вычисления не более чем на четырех наборах из восьми, так как в этом случае $T(\mathsf{P}) \ge \frac{1}{8}\,(4\cdot 2 + 4\cdot 3) = 2\frac{1}{2}$.

Легко видеть, что нулевым аргументом оператора остановки может быть либо выходная, либо независимая переменная, т. е. начало программы P имеет следующий вид:

$$\begin{array}{ll} \mathsf{p}_1 : \ \mathtt{z} = f(x_1,x_2) & \mathtt{z} = f(x_1,x_2) \\ \mathsf{p}_2 : \ \mathtt{Stop(z)} & \mathtt{Stop}(x_i) \end{array}$$

где $i \in \{1,2,3\}$. В первом случае если $T_{\mathsf{P}}(\boldsymbol{x}) = 2$, то $\mathsf{P}(\boldsymbol{x}) = 1$. Следовательно, оператор p_2 прекращает вычисления только на двух наборах: (110) и (111). Во втором случае очевидно, что p_2 прекращает вычисления ровно на четырех наборах.

Задачи

7.2.1. Найти все функции трех переменных средняя сложность которых равна $2\frac{1}{2}$.

7.2.2. Найти все возможные значения, которые может принимать средняя сложность функций трех переменных.

7.2.3. Найти все такие функции $f(x,y,z)$, существенно зависящие от всех своих переменных, для которых $T(f) = L(f)$.

7.2.4. Найти $T(x_1 \vee x_2 \vee x_3 \vee x_4)$.

7.2.5. Найти $T(x_1 \& x_2 \& x_3 \& x_4)$.

7.2.6. Найти $T(x_1 \oplus x_2 \oplus x_3 \oplus x_4)$.

7.3. Симметрические функции

В предыдущей главе в разделе 6.5 было показано, что с точностью до постоянного множителя сложность любой симметрической булевой функции пропорциональна числу ее существенных аргументов. В случае средней сложности ситуация иная — средняя сложность зависит не от числа аргументов функции, а с точностью до постоянного множителя совпадает с величиной $n - \mu(f) + 2$, где через $\mu(f)$ обозначается максимальное число последовательных слоев, на которых функция f принимает одинаковые значения. Например, для линейной функции $x_1 \oplus \cdots \oplus x_n$ значение μ равно единице, для функции голосования n переменных — $\lceil (n+1)/2 \rceil$, а для n-местных дизъюнкции и конъюнкции — n.

Теорема 7.3.1. *Для любой симметрической булевой функции $f(x_1,\ldots,x_n)$ при $n \to \infty$ справедливо равенство*

$$T(f) \asymp n - \mu(f) + 2.$$

Доказательство. Нижняя оценка. Пусть f — произвольная симметрическая функция, зависящая от n аргументов. Рассмотрим два случая: $\mu(f) > n-2$ и $\mu(f) \le n-2$.

(1) В первом случае $1 \le n - \mu(f) + 2 < 4$. Поэтому нижняя оценка теоремы следует из очевидного неравенства $T(f) \ge 1$, справедливого для любой булевой функции.

(2) Во втором случае $n-\mu(f)-1 \ge \frac{1}{4}(n-\mu(f)+2)$. Поэтому для доказательства нижней оценки теоремы достаточно показать, что $T(f) \ge n - \mu(f) - 1$.

Пусть P — минимальная программа, вычисляющая f, s_1 — первый оператор остановки этой программы, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ — набор, на котором оператор s_1 останавливает вычисления. Если оператор s_1 является k-м оператором P и $k < n - \mu(f) - 1$, то система функций $\{\mathsf{q}_1(\boldsymbol{x}), \mathsf{z}(\boldsymbol{x}; k-1)\}$, вычисляемая первыми k операторами P, существенно зависит не более чем от $m = n - \mu(f) - 1$ переменных. Так как f симметрическая функция, то без ограничения общности полагаем, что этими переменными являются $x_1, \ldots, x_m$. В этом случае из минимальности программы P легко следует существование таких постоянных $\alpha_1, \ldots, \alpha_m$, что оператор s_1 останавливает работу P вне зависимости от значений переменных $x_{m+1}, \ldots, x_n$. Следовательно, для любых значений $\sigma_{m+1}, \ldots, \sigma_n$ этих переменных выполняется равенство

$$f(\alpha_1, \ldots \alpha_m, \sigma_{m+1}, \ldots, \sigma_n) = \sigma,$$

где σ — булева константа. В тоже время, найдется такой набор $\beta_{m+1}, \ldots, \beta_n$, что

$$\mathsf{P}(\alpha_1, \ldots \alpha_m, 0, \ldots, 0) \ne \mathsf{P}(\alpha_1, \ldots \alpha_m, \beta_{m+1}, \ldots, \beta_n).$$

Если такой набор не существует, то f принимает одинаковое значение σ на наборах из $n - m = \mu(f) + 1$ последовательных слоев, что противоречит определению величины $\mu(f)$. Пришли к противоречию. Следовательно, $T(f) \ge k \ge n - \mu(f) - 1$.

Верхняя оценка. Теперь покажем, что для любой симметрической булевой функции f, зависящей от n переменных, ее средняя сложность $T(f)$ есть $\mathcal{O}(n - \mu(f) + 2)$. Рассмотрим два случая: $\mu(f) \le \frac{n}{2}$ и $\mu(f) > \frac{n}{2}$.

(1) В первом случае $n - \mu(f) \ge \frac{n}{2}$, и для вычисления f достаточно использовать обычную схему из функциональных элементов. Очевидно, что

$$T(f) \le L(f) = \mathcal{O}(n) = \mathcal{O}(n - \mu(f) + 2).$$

(2) Рассмотрим второй случай. Допустим, что максимальная последовательность слоев, на которых достигается величина $\mu(f)$, начинается с h-го слоя, и значение f на наборах из этих слоев равно σ. Положим $m = n - \mu(f) + 1$. Тогда $f(\alpha_1, \ldots, \alpha_n) = \sigma$ на любом наборе $(\alpha_1, \ldots, \alpha_n)$, содержащем не менее h единиц и не менее $n - (h + \mu(f) - 1) = m - h$ нулей. Воспользуемся этим свойством функции f для ее вычисления. Опишем программу P, вычисляющую функцию f. Положим $n = (2m-1)t + k$, где $0 \le k < 2m - 1$. Программу P представим в виде $t+1$ последовательных подпрограмм $\mathsf{P} = \mathsf{P}_1 \ldots \mathsf{P}_j \ldots \mathsf{P}_t \mathsf{P}_{t+1}$, работающих следующим образом.

• Подпрограмма P_1 присваивает выходной переменной значение σ, вычисляет сумму первых $(2m-1)$ переменных и останавливает вычисления если эта сумма не меньше h и не больше $2m - 1 - (m - h) = m + h - 1$.

• При каждом $j \in \{2, 3, \ldots, t\}$ подпрограмма P_j вычисляет сумму

$$S_j = \sum_{i=(2m-1)(j-1)+1}^{(2m-1)j} x_i$$

и останавливает вычисления если $h \le S_j \le m + h - 1$.

• Последняя подпрограма P_{t+1} вычисляет $f(x_1, \ldots, x_n)$ и присваивает это значение выходной переменной.

Легко видеть, что при каждом $j \in \{1, \ldots, t\}$ подпрограмма P_j состоит из $\mathcal{O}(m)$ операторов, а подпрограмма P_{t+1} — из $\mathcal{O}(n) = \mathcal{O}(mt)$ операторов. Таким образом

$$C(\mathsf{P}_j) = \mathcal{O}(m),\ j \in \{1, \ldots, t\}; \quad C(\mathsf{P}_{t+1}) = \mathcal{O}(mt); \quad C(\mathsf{P}) = \mathcal{O}(mt). \tag{7.3.1}$$

Оценим среднее время работы программы P. Нетрудно убедиться в том, что подпрограмма P_1 останавливает вычисления на

$$A_1 = 2^{n-(2m-1)} \sum_{i=h}^{m+h-1} \binom{2m-1}{i} \geq 2^{n-1} \tag{7.3.2}$$

наборах, а каждая подпрограмма P_j, при всех j больших единицы и не превосходящих t, — на

$$A_j = \left(2^n - \sum_{i=1}^{j-1} A_i\right) 2^{-(2m-1)} \left(\sum_{i=h}^{m+h-1} \binom{2m-1}{i}\right) \geq \frac{1}{2}\left(2^n - \sum_{i=1}^{j-1} A_i\right) \tag{7.3.3}$$

наборах. Индукцией по j покажем, что

$$\sum_{i=1}^{j} A_i \geq 2^n - 2^{n-j} \tag{7.3.4}$$

для всякого $j \in \{1, 2, \ldots, t\}$. В основание индукции ($j = 1$) положим неравенство (7.3.2). Далее предположим, что $\sum_{i=1}^{s} A_i \geq 2^n - 2^{n-s}$ для любого $s \in \{1, \ldots, j-1\}$. Тогда из (7.3.3) и предположения индукции имеем

$$\begin{aligned}\sum_{i=1}^{j} A_j &= \sum_{i=1}^{j-1} A_i + A_j \geq \sum_{i=1}^{j-1} A_i + \frac{1}{2}\left(2^n - \sum_{i=1}^{j-1} A_i\right) = \\ &= \frac{1}{2}\left(2^n + \sum_{i=1}^{j-1} A_i\right) \geq \frac{1}{2}\left(2^n + 2^n - 2^{n-j+1}\right) = 2^n - 2^{n-j}.\end{aligned}$$

Так как $\sum_{i=1}^{j} A_i \leq 2^n$, то из (7.3.4) имеем

$$\sum_{i=s}^{j} A_i = \sum_{i=1}^{j} A_i - \sum_{i=1}^{s-1} A_i \leq 2^n - (2^n - 2^{n-s+1}) = 2^{n-s+1}. \tag{7.3.5}$$

Также из (7.3.4) легко следует, что вместе подпрограммы $\mathsf{P}_1, \ldots, \mathsf{P}_t$ прекращают вычисления не менее чем на $2^n(1 - 2^{-t})$ наборах, и поэтому подпрограмма P_{t+1} работает не более чем на 2^{n-t} наборах. Следовательно, учитывая неравенство (7.3.5) и равенство (7.3.1),

$$\begin{aligned}T(P) &= \frac{1}{2^n}\left(\sum_{j=1}^{t} A_j \sum_{i=1}^{j} C(P_i) + 2^{n-t} C(P)\right) = \\ &= \frac{\mathcal{O}(m)}{2^n}\left(\sum_{j=1}^{t} A_j j + t \cdot 2^{n-t}\right) = \frac{\mathcal{O}(m)}{2^n}\left(\sum_{j=1}^{t}\sum_{i=j}^{t} A_i + 2^n\right) = \\ &= \frac{\mathcal{O}(m)}{2^n}\left(\sum_{j=1}^{t} 2^{n-j+1} + 2^n\right) = \mathcal{O}(m)\left(\sum_{j=1}^{\infty} 2^{1-j} + 1\right) = \mathcal{O}(m).\end{aligned}$$

Теорема доказана.

Подробнее рассмотрим простейшую симметрическую пороговую функцию — дизъюнкцию растущего числа аргументов. Для этой функции приведем вычисляющую ее программу $\mathsf{P}_\vee$. Приводимая ниже программа является конкретизацией алгоритма, использованного в предыдущей теореме для вычисления произвольной симметрической булевой функции.

Без ограничения общности полагаем, что n — четное:

$$\begin{aligned}
&\mathsf{p}_1: && \mathtt{z} = x_1 \vee x_2\\
&\mathsf{p}_2: && \mathtt{Stop(z)}\\
&\ldots && \ldots\ldots\ldots\ldots\\
&\mathsf{p}_j: && \mathtt{z} = x_j \vee x_{j+1}\\
&\mathsf{p}_{j+1}: && \mathtt{Stop(z)}\\
&\ldots && \ldots\ldots\ldots\ldots\\
&\mathsf{p}_{n-3}: && \mathtt{z} = x_{n-3} \vee x_{n-2}\\
&\mathsf{p}_{n-2}: && \mathtt{Stop(z)}\\
&\mathsf{p}_{n-1}: && \mathtt{z} = x_{n-1} \vee x_n
\end{aligned}$$

Легко видеть, что

$$T(\mathsf{P}_\vee) \le \frac{1}{2^n}\left(\sum_{j=1}^{n/2} 2j\cdot 3\cdot 2^{n-2j}\right) < 6\left(\sum_{j=1}^{\infty}\frac{j}{4^j}\right) =$$
$$= 6\left(\sum_{j=1}^{\infty}\sum_{i=j}^{\infty}\frac{1}{4^i}\right) = 6\left(\sum_{j=1}^{\infty}\frac{1}{4^j}\frac{1}{1-\frac{1}{4}}\right) = 6\cdot\frac{4}{3}\cdot\frac{1}{4}\cdot\frac{4}{3} = \frac{8}{3}.$$

Покажем, что приведенная программа асимптотически минимальна. Более точно, справедлива следующая теорема.

Теорема 7.3.2. *При $n\to\infty$ справедливо неравенство*

$$T(x_1\vee\ldots\vee x_n)\sim\frac{8}{3}.$$

Доказательство. Так как $T(\mathsf{P}_\vee)\le\frac{8}{3}$, то для доказательства теоремы достаточно показать, что $T(x_1\vee\ldots\vee x_n)\gtrsim\frac{8}{3}$. Для этого индукцией по числу переменных функции $D_n = x_1\vee\cdots\vee x_n$ покажем, что при $n\ge 2$ для любой минимальной программы P_n, вычисляющей функцию D_n, имеет место неравенство

$$T(\mathsf{P}_n)\ge\frac{8}{3}-\frac{1}{2^{n-4}}. \tag{7.3.6}$$

При $n = 2,3$ неравенство (7.3.6) справедливо. Допустим так же, что оно верно и для некоторого $n\ge 3$.

Напомним, что первым оператором любой программы, вычисляющей полностью определенную булеву функцию, не может быть оператор остановки; перед оператором остановки необходим хотя бы один функциональный оператор, выходом которого является выходная переменная. Если первый оператор остановки программы P_n является третьим оператором P_n, то очевидно, что $T(\mathsf{P}_n) > 3$. Поэтому далее достаточно рассмотреть случай, когда в программе P_n первый оператор остановки стоит на втором месте. С точностью до переименования переменных возможны три принципиально различных случая для выбора первых двух операторов:

$$\begin{array}{lccc}
\mathsf{p}_1: & \mathtt{z} = \varphi(x_1,x_2) & \mathtt{z} = \varphi(x_1,x_2) & \mathtt{z} = \varphi(x_1,x_2)\\
\mathsf{p}_2: & \mathtt{Stop}(x_1) & \mathtt{Stop}(x_3) & \mathtt{Stop(z)}
\end{array}$$

где φ — некоторая двуместная булева функция. Последовательно рассмотрим эти случаи.

1. Программу P_n преобразуем в новую программу P_{n-1}, подставив вместо переменной x_1 нуль и удалив первый оператор остановки, который после подстановки $x_1 = 0$ никогда не будет останавливать вычисления. Поэтому

$$\sum T_{\mathsf{P}_n}(0,x_2,\ldots,x_n) = \sum\left(T_{\mathsf{P}_{n-1}}(x_2,\ldots,x_n)+1\right).$$

Здесь и далее суммирование ведется по всем возможным значениям переменных x_i. Очевидно, что программа P_{n-1} вычисляет дизъюнкцию $(n-1)$ переменных, и для этой программы справедливо неравенство

$$\frac{1}{2^{n-1}}\sum T_{\mathsf{P}_{n-1}}(x_2,\ldots,x_n)\geq T(D_{n-1}(x_2,\ldots,x_n)).$$

Поэтому, при $n\geq 3$ из предыдущих соотношений и предположения индукции имеем

$$\begin{aligned}T(D_n)&=\frac{1}{2^n}\left(\sum T_{\mathsf{P}_n}(1,x_2,\ldots,x_n)+\sum T_{\mathsf{P}_n}(0,x_2,\ldots,x_n)\right)\geq\\&\geq\frac{1}{2^n}\left(2^{n-1}\cdot 2+2^{n-1}\bigl(1+T(D_{n-1}(x_2,\ldots,x_n))\bigr)\right)\geq\\&\geq\frac{3}{2}+\frac{1}{2}T(D_{n-1})\geq\frac{3}{2}+\frac{1}{2}\left(\frac{8}{3}-\frac{1}{2^{n-5}}\right)=\\&=\left(\frac{8}{3}-\frac{1}{2^{n-4}}\right)+\frac{1}{6}>\frac{8}{3}-\frac{1}{2^{n-4}}.\end{aligned}$$

В первом случае неравенство теоремы доказано.

2. Если $\varphi\equiv 1$, то рассматриваемый случай сводится к предыдущему. Поэтому будем полагать, что φ не равна тождественной единице. Следовательно, найдутся такие α и β, что $\varphi(\alpha,\beta)=0$. Тогда,

$$\mathsf{P}_n(\alpha,\beta,1,x_4,\ldots,x_n)=0,$$

т.е. программа P_n не может вычислять дизъюнкцию. Следовательно, второй случай невозможен.

3. Прежде всего покажем, что никакая минимальная программа, вычисляющая дизъюнкцию $n>2$ переменных, не содержит функциональный оператор, вычисляющий тождественный нуль и выполняемый программой раньше, чем первый оператор остановки.

Действительно, предположим, что в некоторой минимальной программе P имеется функциональный оператор p_t, вычисляющий тождественный нуль и выполняемый программой раньше, чем ее первый оператор остановки $\mathsf{p}_j:\mathtt{Stop(a)}$, здесь $t<j$. Из леммы 7.1.1 следует, что выходом такого оператора не может быть внутренняя переменная. Если выходом оператора является выходная переменная $\mathtt{z}$[1)], то легко видеть, что вход $\mathtt{a}(\boldsymbol{x};j-1)$ первого оператора остановки должен существенно зависеть от всех n переменных. В противном случае значение дизъюнкции n переменных будет равно нулю при единичном значении переменной, не являющейся существенным аргументом функции, вычисляемой на входе оператора остановки. Следовательно, $T(\mathsf{P})\geq n$. Противоречие с верхней оценкой.

Очевидно, что в рассматриваемом случае $\varphi(0,0)=0$, так как иначе $\mathsf{P}_n(0,\ldots,0)=1$. Преобразуем программу P_n в новую программу P_{n-1}, подставив вместо переменных x_1 и x_2 нули, удалив из P_n ставший ненужным первый оператор остановки, и преобразовав получившуюся программу к приведенному виду. Из доказанного выше свойства минимальных программ вычисляющих дизъюнкцию следует, что программа P_{n-2} содержит по крайней мере на два оператора (p_1 и p_2) меньше чем программа P_n и, поэтому,

$$\sum T_{\mathsf{P}_n}(0,0,x_3,\ldots,x_n)\geq\sum\left(T_{\mathsf{P}_{n-2}}(x_3,\ldots,x_n)+2\right).$$

Как и в первом случае, легко видеть, что новая программа P_{n-2} вычисляет дизъюнкцию $(n-2)$ переменных, и для этой программы справедливо неравенство

$$\frac{1}{2^{n-2}}\sum T_{\mathsf{P}_{n-2}}(x_3,\ldots,x_n)\geq T(D_{n-2}(x_1,\ldots,x_n)).$$

[1)] Здесь без ограничения общности полагаем, что между рассматриваемым функциональным оператором и первым оператором остановки нет ни одного функционального оператора, выходом которого является переменная $\mathtt{z}$.

Поэтому, при $n \geq 4$

$$T(D_n) = \frac{1}{2^n}\left(\sum_{\alpha_1 \vee \alpha_2 = 1} T_{\mathsf{P}_n}(\alpha_1, \alpha_2, x_3, ..., x_n) + \sum T_{\mathsf{P}_n}(0,0,x_3,...,x_n)\right)$$
$$\geq \frac{1}{2^n}\left(3 \cdot 2^{n-2} \cdot 2 + 2^{n-2}\left(2 + T(D_{n-2}(x_3, \ldots, x_n))\right)\right) \geq$$
$$\geq 2 + \frac{1}{4}T(D_{n-2}) \geq 2 + \frac{1}{4}\left(\frac{8}{3} - \frac{1}{2^{n-6}}\right) = \frac{8}{3} - \frac{1}{2^{n-4}}.$$

В третьем случае неравенство (7.3.6) доказано. Следовательно, при $n \to \infty$ средняя сложность дизъюнкции n переменных асимптотически не меньше чем $\frac{8}{3}$. Теорема доказана.

Можно показать, что дизъюнкция является самой простой "в среднем" симметрической булевой функцией. Для другой естественной симметрической функции — линейной функции n аргументов средняя сложность совпадает с обычной сложностью, т.е.

$$L(x_1 \oplus x_2 \oplus \cdots \oplus x_n) = T(x_1 \oplus x_2 \oplus \cdots \oplus x_n) = n - 1.$$

Задачи

7.3.1. Пусть $f \in P_2(n)$, $n \geq 3$ и f существенно зависит от всех своих аргументов. Показать, что $T(f) > 2$.

7.3.2. Пусть $n \to \infty$. Показать, что для всех $\sigma_1, \ldots, \sigma_n$ справедливы асимптотические равенства:
а) $T(x_1^{(\sigma_1)} \vee \ldots \vee x_n^{(\sigma_n)}) \sim \frac{8}{3}$, b) $T(x_1^{(\sigma_1)} \& \ldots \& x_n^{(\sigma_n)}) \sim \frac{11}{3}$.

7.3.3. Указать функцию f_n, существенно зависящую от n аргументов, для которой $T(f_n) < T(D_n)$.

7.3.4. Показать, что $T(x_1 \oplus x_2 \oplus \cdots \oplus x_n) = n - 1$.

7.4. Асимптотические методы построения программ

1. Рассмотрим задачу об определении средней сложности "почти всех" булевых функций n переменных. Покажем, что средняя сложность почти каждой булевой функции с точностью до постоянного множителя совпадает с ее обычной сложностью.

Теорема 7.4.1. *Пусть $n \to \infty$. Тогда: (i) для почти каждой булевой функции f, зависящей от n переменных*

$$T(f) \gtrsim \frac{2^{n-4}}{n};$$

(ii) для каждой булевой функции f, зависящей от n переменных

$$T(f) \lesssim \frac{2^{n-1}}{n}.$$

Доказательство. *(i)* Пусть f — булева функция n переменных, P — программа, вычисляющая f. Каждому двоичному набору $\boldsymbol{x}$ длины n, рассматриваемому как двоичная запись натурального числа, поставим в соответствие его номер $N_{\mathsf{P}}(\boldsymbol{x})$ такой, что $1 \leq N_{\mathsf{P}}(\boldsymbol{x}) \leq 2^n$; $N_{\mathsf{P}}(\boldsymbol{x}) < N_{\mathsf{P}}(\boldsymbol{y})$, если $T_{\mathsf{P}}(\boldsymbol{x}) < T_{\mathsf{P}}(\boldsymbol{y})$; $N_{\mathsf{P}}(\boldsymbol{x}) < N_{\mathsf{P}}(\boldsymbol{y})$, если $T_{\mathsf{P}}(\boldsymbol{x}) = T_{\mathsf{P}}(\boldsymbol{y})$ и $\boldsymbol{x} < \boldsymbol{y}$.

Оценим число булевых функций средняя сложность каждой из которых не превосходит величины $\frac{2^{n-4}}{n}$. Пусть f — одна из таких функций, P — минимальная программа, вычисляющая f. Рассмотрим набор $\boldsymbol{x}_0$ такой, что $N_{\mathsf{P}}(\boldsymbol{x}_0) = 2^{n-1}$. Тогда из определения средней сложности следует, что

$$T(\mathsf{P}) = 2^{-n}\sum_{\boldsymbol{y}} T_{\mathsf{P}}(\boldsymbol{y}) > 2^{-n} \sum_{\boldsymbol{y} \mid N(\boldsymbol{y}) > N(\boldsymbol{x}_0)} T_{\mathsf{P}}(\boldsymbol{y}) \geq \frac{1}{2}T_{\mathsf{P}}(\boldsymbol{x}_0). \qquad (7.4.1)$$

Поэтому, $T_{\mathsf{P}}(\boldsymbol{x}_0) < 2T(f)$. Так как $T(f) \le \frac{2^{n-4}}{n}$, то легко видеть, что

$$T_{\mathsf{P}}(\boldsymbol{x}_0) < 2\frac{2^{n-4}}{n} = \frac{2^{n-3}}{n}. \tag{7.4.2}$$

Каждая функция однозначно определяется первыми $T_{\mathsf{P}}(\boldsymbol{x}_0)$ операторами своей минимальной программы P и двоичным вектором длины не более чем 2^{n-1}, состоящим из значений функции f на тех аргументах, время работы P на которых больше времени работы этой программы на $\boldsymbol{x}_0$. Обозначим через N_0 число различных программ, состоящих не более чем из $T_{\mathsf{P}}(\boldsymbol{x}_0)$ операторов. Тогда число функций, средняя сложность которых не превосходит $\frac{2^{n-4}}{n}$, ограничена сверху величиной $N_0 2^{2^{n-1}}$. Оценим N_0.

Любая программа P определяется списком своих операторов p_i, каждый из которых однозначно задается следующими данными:

- типом оператора — возможны всего два варианта, оператор может быть либо функциональным, либо оператором остановки;
- двуместной булевой функцией f_i, вычисляемой функциональным оператором (для оператора остановки эта информация опускается) — существует всего 16 различных двуместных булевых функций;
- номером переменной, выходной или внутренней, являющейся выходом функционального оператора (для оператора остановки эта информация опускается) — если программа P состоит из L операторов, то общее число внутренних и выходной переменных не превосходит L и без ограничения общности полагаем, что внутренние переменные нумеруются числами от 1 до $L-1$, а выходной переменной присваивается номер L;
- номерами переменных, независимых или внутренних, являющихся входами оператора — полагаем, что независимые переменные нумеруются числами от $L+1$ до $L+n$, поэтому общее число пар номеров не превосходит $(L+n)^2$.

Таким образом для числа N, равного числу различных программ, состоящих из L операторов, справедливо неравенство

$$N \le \left(2 \cdot 16 \cdot L \cdot (L+n)^2\right)^L \le (4(L+n))^{3L}. \tag{7.4.3}$$

Подставляя в (7.4.3) вместо L величину $T_{\mathsf{P}}(\boldsymbol{x}_0)$ и учитывая неравенство (7.4.2), получаем, что при $n \ge 5$ имеет место неравенство

$$N_0 \le \left(4\left(T_{\mathsf{P}}(\boldsymbol{x}_0)+n\right)\right)^{3T_{\mathsf{P}}(\boldsymbol{x}_0)} \le \left(4\left(\frac{2^{n-3}}{n}+n\right)\right)^{3\cdot 2^{n-3}/n} \le 2^{3\cdot 2^{n-3}}.$$

Следовательно, число функций, средняя сложность которых не превосходит $\frac{2^{n-4}}{n}$, не больше чем

$$2^{3\cdot 2^{n-3}} 2^{2^{n-1}} = 2^{\frac{7}{8}\cdot 2^n} = o\left(2^{2^n}\right).$$

Таким образом, средняя сложность почти каждой булевой функции, зависящей от n переменных, не меньше чем $\frac{2^{n-4}}{n}$. Первое неравенство теоремы доказано.

(ii) Напомним, что каждому двоичному набору $(\sigma_1 \ldots \sigma_k)$ соответствует его номер $|(\sigma_1 \ldots \sigma_k)| = \sum_{i=1}^{k} \sigma_i 2^{k-i}$. Положим $s = \lfloor n - \log_2 n \rfloor$. Функцию f разложим по первым $n-s$ переменным:

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

Программу, вычисляющую функцию f, представим в следующем виде

$$\mathsf{P} = \mathsf{P}_0 \ldots \mathsf{P}_j \ldots \mathsf{P}_{2^{n-s}-1},$$

где $j = |(\sigma_1 \ldots \sigma_{n-s})|$, P_j — программа, вычисляющая функцию

$$f_j(x_{n-s+1}, \ldots, x_n) = f(\sigma_1, \ldots, \sigma_{n-s}, x_{n-s+1}, \ldots, x_n)$$

и прекращающая работу программы P, если $x_1^{\sigma_1}\& \ldots \& x_{n-s}^{\sigma_{n-s}} = 1$. Так как схемная сложность произвольной булевой функции, зависящей от s переменных, асимптотически не превосходит $\frac{2^s}{s}$, то $C(\mathsf{P}_j) \lesssim \frac{2^s}{s}$. Поэтому

$$T(\mathsf{P}) \sim \frac{1}{2^n} \sum_{j=0}^{2^{n-s}-1} \left(2^s \sum_{i=1}^{j} C(\mathsf{P}_i) \right) = \frac{1}{2^n} 2^s \sum_{j=0}^{2^{n-s}-1} \sum_{i=1}^{j} C(\mathsf{P}_i) \lesssim$$

$$\lesssim \frac{1}{2^n} 2^s \frac{2^s}{s} \sum_{j=0}^{2^{n-s}-1} j \lesssim \frac{1}{2^n} \frac{2^{2s}}{s} \frac{2^{2(n-s)}}{2} \sim \frac{2^{n-1}}{s} \sim \frac{2^{n-1}}{n}.$$

Теорема доказана.

Без доказательства приведем аналог теоремы 7.4.1 для булевых операторов.

Теорема 7.4.2. *Пусть $n, m \to \infty$, $m = n^{\mathcal{O}(1)}$. Тогда*: *(i) для почти каждого булевого (m,n)-оператора f*

$$T(f) \gtrsim \frac{2^{n-2}m}{n};$$

(ii) для каждого булевого (m,n)-оператора f

$$T(f) \lesssim \frac{2^{n-2}m}{n}.$$

2. Будем говорить, что программа P использует память объема d, если число внутренних и выходных переменных этой программы равно d. Среднюю сложность функции f при вычислении ее программами, объем памяти которых не превосходит d, обозначим через $T_d(f)$. Одну и ту же функцию можно вычислить разными программами, использующими разный объем памяти. При этом среднее время работы программы может существенно зависеть от объема памяти — чем больше память, тем меньше среднее время.

Пример 7.4.1. Рассмотрим две программы, вычисляющие конъюнкцию шести переменных:

p_1 : $\mathtt{z} = 0$	$\mathtt{z} = x_1 \& x_2$
p_2 : $\mathtt{y} = \overline{x_1 \& x_2}$	$\mathtt{z} = \mathtt{z} \ \& \ x_3$
p_3 : $\mathtt{Stop(y)}$	$\mathtt{z} = \mathtt{z} \ \& \ x_4$
p_4 : $\mathtt{y} = \overline{x_3 \& x_4}$	$\mathtt{z} = \mathtt{z} \ \& \ x_5$
p_5 : $\mathtt{Stop(y)}$	$\mathtt{z} = \mathtt{z} \ \& \ x_6$
p_6 : $\mathtt{z} = x_5 \& x_6$	

В первой программе объем используемой памяти равен двум, во второй — единице. Среднее время работы первой программы равно $3\frac{9}{16}$, а второй — 5. Нетрудно показать, что среднее время любой программы, вычисляющей конъюнкцию шести переменных, не меньше пяти, если объем используемой программой памяти равен единице. □

В общем случае имеет место следующий результат.

Теорема 7.4.3. *Пусть $n \to \infty$, $d \geq n$. Тогда*: *(i) для почти каждой булевой функции f, зависящей от n переменных*

$$T_d(f) \gtrsim \frac{2^{n-4}}{\log_2 d};$$

(ii) для каждой булевой функции f, зависящей от n переменных

$$T_d(f) \lesssim \frac{2^{n-1}}{\log_2 d}.$$

Доказательство этой теоремы почти дословно повторяет доказательство теоремы 7.4.1. В доказательстве нижней оценки теоремы 7.4.3 неравенство (7.4.2) из доказательства теоремы 7.4.1 естественным образом превращается в неравенство $T_{\mathsf{P}}(\boldsymbol{x}_0) < \frac{2^{n-3}}{\log_2 d}$, а неравенство (7.4.3) надо заменить оценкой числа программ, состоящих из L операторов и использующих память объема d:

$$N_d \le \left(2 \cdot 16 \cdot d \cdot (d+n)^2\right)^L \le (4(d+n))^{3L} \le (8d)^{3L}.$$

В доказательстве верхней оценки единственное отличие состоит в выборе параметра s — в рассматриваемом случае надо положить $s = \lfloor \log_2 d - \log_2 \log_2 d \rfloor$.

Задачи

7.4.1. Показать, что для каждой функции f из $P_2(n)$, принимающей равные значения на противоположных наборах, при $n \to \infty$ справедливо неравенство $T(f) \lesssim \frac{2^{n-2}}{n}$.

7.4.2. Показать, что для каждой функции f из $P_2(n)$, равной нулю на наборах четного веса, при $n \to \infty$ справедливо неравенство $T(f) \lesssim \frac{2^{n-3}}{n}$.

7.4.3. Доказать утверждение (ii) теоремы 7.4.2.

7.4.4. Показать, что $T(U_n) \asymp n$.

7.4.5. Показать, что $T_1(x_1 \& \ldots \& x_n) = n - 1$.

7.4.6. Для произвольной n-местной булевой функции f оценить:
a) $T_1(f)$; b) $T_2(f)$; c) $T_3(f)$.

7.5. Сложность и средняя сложность функций

1. Определим насколько сильно могут различаться среднее время вычисления конкретной булевой функции и ее время вычисления в худшем случае. Положим

$$\mu(f) = \frac{C(f)}{T(f)}, \qquad \mu(n) = \max \mu(f),$$

где максимум берется по всем булевым функциям, зависящим от n переменных. Из теоремы 7.4.1 предыдущего параграфа и теоремы 6.5.1 следует, что средняя сложность и сложность в худшем случае для почти всех булевых функций различаются не более чем в постоянное число раз, т.е. $\mu(f) = \text{const}$ для почти каждой булевой функции. В тоже время пример симметрических пороговых функций показывает, что отношение сложности в худшем случае к средней сложности может расти вместе с ростом числа аргументов у функций. Покажем, что для некоторых функций это отношение может быть экспоненциально большим.

Теорема 7.5.1. *Существуют такие постоянные c_1 и c_2, что*

$$c_1 \left(\frac{2^n}{n}\right)^{1/2} \le \mu(n) \le c_2 \left(\frac{2^n}{n}\right)^{1/2}.$$

Доказательство. Пусть $k = \lceil (n + \log n)/2 \rceil$ и g — любая из "почти всех" самых сложных булевых функций от k переменных, т.е. схемная сложность функции g с точностью до постоянного множителя равна $\sqrt{\frac{2^n}{n}}$. Рассмотрим функцию

$$f(x_1, \ldots, x_n) = \overline{x}_{k+1} \& \cdots \& \overline{x}_n \& g(x_1, \ldots, x_k).$$

и программу $\mathsf{P}(g)$, вычисляющую функцию g. Нетрудно видеть, что следующая программа P вычисляет f:

$$\mathsf{p}_1 : \qquad \mathsf{z} = 0$$

p_2 : $\mathtt{Stop}(x_{k+1})$
p_3 : $\mathtt{Stop}(x_{k+2})$
. . . .
p_{n-k+1} : $\mathtt{Stop}(x_n)$
$\mathsf{P}(g)$.

После несложных вычислений для средней сложности программы P находим, что

$$T(\mathsf{P}) \le \frac{1}{2^n}\left(\sum_{j=1}^{n-k}(j+1)2^{n-j} + (n-k+1+C(g))2^k\right) = \mathcal{O}(1).$$

Так как $C(f) \sim C(g) \ge \frac{2^{k-1}}{k}$, то, по порядку величины отношение сложности функции f к ее средней сложности не меньше чем $\sqrt{\frac{2^n}{n}}$. Следовательно,

$$\mu(n) \ge c_1\left(\frac{2^n}{n}\right)^{1/2},$$

где c_1 — некоторая постоянная.

Теперь покажем, что найдется постоянная c_2 для которой при достаточно больших n выполняется неравенство

$$\mu(n) \le c_2\left(\frac{2^n}{n}\right)^{1/2}, \tag{7.5.1}$$

Пусть f — произвольная булева функция от n переменных, P — программа, которая вычисляет f, и среднее время ее работы минимально. Положим $k = \lfloor (n + \log n)/2 \rfloor$. Рассмотрим набор $\boldsymbol{x}$ такой, что $N_{\mathsf{P}}(\boldsymbol{x}) = 2^n - 2^k$. Здесь номер $N_{\mathsf{P}}(\boldsymbol{x})$ набора $\boldsymbol{x}$ определяется так же, как и в доказательстве теоремы 7.4.1. Так как

$$T(\mathsf{P}) = 2^{-n}\sum_{\boldsymbol{y}} T_{\mathsf{P}}(\boldsymbol{y}) > 2^{-n}\sum_{\boldsymbol{y}\ |\ N(\boldsymbol{y})>N(\boldsymbol{x})} T_{\mathsf{P}}(\boldsymbol{y}) \ge 2^{-n}2^k T_{\mathsf{P}}(\boldsymbol{x}),$$

то легко видеть, что

$$2^{k-n}T_{\mathsf{P}}(\boldsymbol{x}) < T(f). \tag{7.5.2}$$

Далее, пусть $\tilde{f}$ — частичная булева функция, определенная на всех таких наборах $\boldsymbol{y}_i$, что $N_{\mathsf{P}}(\boldsymbol{y}_i) > N_{\mathsf{P}}(\boldsymbol{x})$, и совпадающая на этих наборах с f. Так как $2^k \asymp \sqrt{2^n n}$, то из теоремы 6.5.3 следует существование программы $\mathsf{P}_{\tilde{f}}$, вычисляющей $\tilde{f}$ и такой, что

$$\mathrm{C}(\mathsf{P}_{\tilde{f}}) = \mathcal{O}\left(\frac{2^n}{n}\right)^{1/2}. \tag{7.5.3}$$

Теперь опишем программу P', вычисляющую функцию f. Сначала воспользуемся программой P, которая за минимальное среднее время вычисляет f. С ее помощью будем вычислять значения функции f на наборах $\boldsymbol{y}$ таких, что $N_{\mathsf{P}}(\boldsymbol{y}) \le N_{\mathsf{P}}(\boldsymbol{x})$. Так как $2^{n-k} \asymp \sqrt{\frac{2^n}{n}}$, то из (7.5.2) следует, что для вычисления функции f на этих наборах потребуется привлекать не более H_1 операторов, где

$$H_1 = \mathcal{O}\left(\left(\frac{2^n}{n}\right)^{1/2} T(f)\right). \tag{7.5.4}$$

Для вычисления функции f на оставшихся наборах воспользуемся программой $\mathsf{P}_{\tilde{f}}$, вычисляющей функцию $\tilde{f}$.

Таким образом, из (7.5.4) и (7.5.3) следует, что сложность $C(\mathsf{P}')$ программы P' по порядку не превосходит величины

$$H_1 + C(\mathsf{P}_f) = \left(\frac{2^n}{n}\right)^{1/2} T(f) + \left(\frac{2^n}{n}\right)^{1/2} \leq 2\left(\frac{2^n}{n}\right)^{1/2} T(f).$$

Так как $C(f) \leq C(\mathsf{P}')$, то найдется константа c_2 удовлетворяющая неравенству (7.5.1). Теорема доказана.

2. Хотя средняя сложность рассмотренной в теореме 7.5.1 функции f значительно меньше ее обычной сложности, в области определения f есть достаточно большая подобласть (состоящая из наборов, удовлетворяющих равенству $\overline{x}_{k+1}\& \cdots \& \overline{x}_n = 1$), в которой f является типичным представителем "почти всех" функций. Поэтому в силу теоремы 7.4.1 для почти каждой функции f, определенной описанным выше способом, ее средняя по этой подобласти сложность будет отличаться от $C(f)$ только постоянным множителем. Покажем, что этот эффект связан не со способом определения f, а является отражением общей ситуации: для любой булевой функции зависящей от n переменных существует подобласть в которой ее средняя сложность отличается от ее сложности в худшем случае по порядку величины не более чем в n раз.

Далее потребуются два определения. Пусть $f \in P_2(n)$, P — программа, $D \subseteq \mathbb{B}^n$. *Средним временем работы программы* P *на области* D называется величина

$$T_D(\mathsf{P}) = \frac{1}{|D|} \sum_{x \in D} T_{\mathsf{P}}(\boldsymbol{x}).$$

Средней сложностью функции f на области D называется величина

$$T_D(f) = \min T_D(\mathsf{P}),$$

где минимум берется по всем программам, вычисляющим f на области D.

Теорема 7.5.2. *Для любой функции $f(x_1, \ldots, x_n)$, существенно зависящей от всех своих переменных, найдется такая область $D \subseteq \mathbb{B}^n$, что*

$$T_D(f) \geq \frac{1}{14n} C(f).$$

Перед доказательством теоремы дадим одно определение и докажем необходимую лемму.

Пусть P — произвольная программа. Каждому двоичному набору $\boldsymbol{x} \in D$, рассматриваемому как двоичная запись натурального числа, поставим в соответствие его номер $N_{\mathsf{P},D}(\boldsymbol{x})$ такой, что $1 \leq N_{\mathsf{P},D}(\boldsymbol{x}) \leq |D|$, и при любом $y \in D$ справедливо неравенство $N_{\mathsf{P},D}(\boldsymbol{x}) < N_{\mathsf{P},D}(\boldsymbol{y})$, если $T_{\mathsf{P}}(\boldsymbol{x}) < T_{\mathsf{P}}(\boldsymbol{y})$, или если $T_{\mathsf{P}}(\boldsymbol{x}) = T_{\mathsf{P}}(\boldsymbol{y})$ и $\boldsymbol{x} < \boldsymbol{y}$.

Характеристическую функцию области $D \subseteq \mathbb{B}^n$ будем обозначать символом χ_D.

Лемма 7.5.1. *Пусть $D \subseteq \mathbb{B}^n$. Для любой $f : D \to \mathbb{B}$ существуют область $D' \subseteq D$ и функция h такие, что:*

(a) $f_{D \setminus D'} = h_{D \setminus D'}$,

(b) $C(h, \chi_{D \setminus D'}) \leq 10 T_D(f)$,

(c) $|D'| \leq \frac{1}{2}|D| + 1$.

Доказательство. Пусть P — вычисляющая f программа на которой достигается минимальное среднее время, и пусть набор $\boldsymbol{x}_0$ такой, что $N_{\mathsf{P},D}(\boldsymbol{x}_0) = |D| - \lfloor \frac{1}{2}|D| \rfloor$. Тогда

$$T_D(f) \geq \frac{1}{|D|}\left(\sum_{x \mid N_{\mathsf{P}}(\boldsymbol{x}) \geq N_{\mathsf{P}}(\boldsymbol{x}_0)} T_{\mathsf{P}}(\boldsymbol{x})\right) \geq \frac{1}{|D|}\left(\left\lfloor \frac{|D|}{2} \right\rfloor + 1\right) T_{\mathsf{P}}(\boldsymbol{x}_0) \geq \frac{T_{\mathsf{P}}(\boldsymbol{x}_0)}{2}.$$

Следовательно,

$$T_{\mathsf{P}}(\boldsymbol{x}_0) \le 2T_D(f).$$

Пусть $\mathsf{q}_1, \ldots, \mathsf{q}_k$ — операторы, являющиеся нулевыми аргументами операторов остановки программы P, последний из которых останавливает работу этой программы на наборе $\boldsymbol{x}_0$. Положим $D' = \{\boldsymbol{x} \mid T_{\mathsf{P}}(\boldsymbol{x}) > T_{\mathsf{P}}(\boldsymbol{x}_0)\}$. Тогда $|D'| \le \frac{1}{2}|D|$, $\chi_{D'} = \bigwedge_{i=1}^{k} \overline{\mathsf{q}}_i$ и значения функции

$$h(\boldsymbol{x}) = \mathsf{q}_1(\boldsymbol{x})\mathsf{z}(\boldsymbol{x}; t_1) \vee \overline{\mathsf{q}}_1(\boldsymbol{x})(\mathsf{q}_2(\boldsymbol{x})\mathsf{z}(\boldsymbol{x}; t_2) \vee \ldots$$
$$\vee\, \overline{\mathsf{q}}_{k-2}(\boldsymbol{x})(\mathsf{q}_{k-1}(\boldsymbol{x})\mathsf{z}(\boldsymbol{x}; t_{k-1}) \vee \overline{\mathsf{q}}_{k-1}(\boldsymbol{x})\mathsf{q}_k(\boldsymbol{x})\mathsf{z}(\boldsymbol{x}; t_k)) \ldots),$$

совпадают на $D \setminus D'$ с соответствующими значениями f и равны нулю вне этой области. Очевидно, что

$$C(h, \chi_{D'}) \le k + 3k + T_{\mathsf{P}}(\boldsymbol{x}_0) \le 5T_{\mathsf{P}}(\boldsymbol{x}_0) \le 10T_D(f).$$

Лемма доказана.

Доказательство теоремы 7.5.2. Докажем теорему методом от противного. Положим

$$T = \frac{1}{14n}C(f), \quad D_0 = \mathbb{B}^n.$$

Предположим, что для любой области $D' \subseteq D_0$ справедливо неравенство

$$T(f_{D'}) < T.$$

Воспользуемся леммой 7.5.1. В силу этой леммы существуют область $D_1 \subseteq D_0$ и функция h^1, такие что

$$f_{D_0 \setminus D_1} = h^1_{D_0 \setminus D_1}, \quad C(h^1, \chi_{D_1}) \le 10T, \quad |D_1| \le \frac{1}{2}|D_0| + 1.$$

Снова используем лемму 7.5.1, применив ее к функции f_{D_1}. В силу этой леммы существуют область $D_2 \subseteq D_1$ и функция h^2 такие, что

$$f_{D_1 \setminus D_2} = h^2_{D_1 \setminus D_2}, \quad C(h^2, \chi_{D_2}) \le 10T, \quad |D_2| \le \frac{1}{2}|D_1| + 1.$$

Повторим подобную процедуру еще $n-2$ раза. В результате для каждого i, $0 \le i \le n-1$ получим области D_{i+1}, $D_{i+1} \subseteq D_i$ и функции h^{i+1} такие, что

$$f_{D_i \setminus D_{i+1}} = h^{i+1}_{D_i \setminus D_{i+1}}, \tag{7.5.5}$$
$$C(h^{i+1}, \chi_{D_{i+1}}) \le 10T, \tag{7.5.6}$$
$$|D_{i+1}| \le \frac{1}{2}|D_i| + 1. \tag{7.5.7}$$

Из (7.5.5) следует, что

$$f_{D_i} = h^{i+1}_{D_i}\overline{\chi}_{D_{i+1}} \vee f_{D_{i+1}}\chi_{D_{i+1}}.$$

Поэтому

$$f = h^1\overline{\chi}_{D_1} \vee \chi_{D_1}(h^2\overline{\chi}_{D_2} \vee \ldots (h^n\overline{\chi}_{D_n} \vee \chi_{D_n} f_{D_n}) \ldots).$$

Следовательно, в силу (7.5.6) и последней формулы, имеет место неравенство

$$C(f) \le 13nT + C(f_{D_n}). \tag{7.5.8}$$

Оценим мощность множества D_n. В силу неравенства (7.5.7) имеем

$$|D_n| \le \frac{1}{2}|D_{n-1}| + 1 \le \frac{1}{2}\left(\frac{1}{2}|D_{n-2}| + 1\right) + 1 \le \ldots$$
$$\le \frac{1}{2}\left(\frac{1}{2}\left(\ldots\left(\frac{1}{2}|D_0| + 1\right)\ldots\right)\right) + 1 < \frac{1}{2^n}2^n + 2 \le 2.$$

Очевидно, что найдется доопределение частичной функции f_{D_n}, которое существенно зависит не более чем от одной переменной, и поэтому $C(f_{D_n}) \le 1$. Подставляя это неравенство в (7.5.8), получаем

$$C(f) < 13nT + 1 \le \frac{13n}{14n} C(f) + 1 < C(f).$$

Пришли к противоречию. Таким образом, сделанное предположение не верно. Теорема доказана.

3. Установим два утверждения, связывающих среднюю сложность и сложность в худшем случае. Сначала покажем, что для каждой булевой функции f существует программа, сложность и среднее время работы которой одновременно близки соответственно к сложности в худшем случае и к средней сложности этой функции. Затем для произвольного булева оператора опишем алгоритм, преобразовывающий вычисляющую его неветвящуюся программу в схему из функциональных элементов.

Теорема 7.5.3. *Для любой булевой функции f найдется такая вычисляющая ее программа* P*, что*

$$T(\mathsf{P}) \le 2T(f), \quad C(\mathsf{P}) \le 4C(f).$$

Доказательство. Пусть P — программа, вычисляющая функцию f за минимальное среднее время. Пусть $\boldsymbol{x}$ — набор с минимальным номером такой, что $T_{\mathsf{P}}(\boldsymbol{x}) \ge 2C(f)$ (если такого набора нет, то утверждение теоремы тривиально, так как тогда $T(f) = T(\mathsf{P}) \le C(\mathsf{P}) \le 2C(f)$). Ясно, что $T_{\mathsf{P}}(\boldsymbol{x}) \le 3C(f)$, поскольку невыполнение этого неравенства влечет существование в программе P расположенных друг за другом $C(f)+1$ операторов первого типа, что противоречит минимальности программы P. Среднее время работы программы P можно представить следующим образом:

$$\begin{aligned} T(\mathsf{P}) &= 2^{-n}\Big(\sum_{N_{\mathsf{P}}(\boldsymbol{y}) < N_{\mathsf{P}}(\boldsymbol{x})} T_{\mathsf{P}}(\boldsymbol{y}) + \sum_{N_{\mathsf{P}}(\boldsymbol{y}) \ge N_{\mathsf{P}}(\boldsymbol{x})} T_{\mathsf{P}}(\boldsymbol{y})\Big) = \\ &= T_1 + T_2 \ge T_1 + 2C(f)(2^n - N_{\mathsf{P}}(\boldsymbol{x}) + 1)2^{-n}. \end{aligned}$$

Преобразуем программу P, заменив операторы с номерами большими $T_{\mathsf{P}}(\boldsymbol{x})$ программой без операторов остановки — минимальной схемой из функциональных элементов вычисляющей f. Легко видеть, для сложности новой программы P' справедливо неравенство

$$C(\mathsf{P}') \le T_{\mathsf{P}}(\boldsymbol{x}) + C(f) \le 4C(f),$$

а для среднего времени работы этой программы — неравенство:

$$\begin{aligned} T(\mathsf{P}') &\le 2^{-n}\Big(\sum_{N_{\mathsf{P}}(\boldsymbol{y}) < N_{\mathsf{P}}(\boldsymbol{x})} T_{\mathsf{P}'}(\boldsymbol{y}) + \sum_{N_{\mathsf{P}}(\boldsymbol{y}) \ge N_{\mathsf{P}}(\boldsymbol{x})} T_{\mathsf{P}'}(\boldsymbol{y})\Big) = \\ &= T_1' + T_2' \le T_1' + 4C(f)(2^n - N_{\mathsf{P}}(\boldsymbol{x}) + 1)2^{-n} \le \\ &\le 2(T_1' + 2C(f)(2^n - N_{\mathsf{P}}(\boldsymbol{x}) + 1)2^{-n}). \end{aligned}$$

Так как $T_1' = T_1$, то $T(\mathsf{P}') \le 2T(\mathsf{P})$. Теорема доказана.

Наиболее очевидный способ преобразования программы P, вычисляющей булев (m,n)-оператор, в схему, вычисляющую такой же оператор, состоит в удалении из программы операторов остановки и добавлении новых функциональных операторов, вычисляющих компоненты P_l в соответствии с формулой (7.1.1). Если операторы остановки составляют лишь небольшую часть программы, или если между любыми соседними операторами остановки каждая из выходных переменных вычисляется заново, т.е. для каждого $l \in \{1,\dots,m\}$ между операторами s_i и s_{i+1} найдется оператор вида $p:\ \mathsf{z}_l = f(\mathsf{a},\mathsf{b})$, то сложность построенной схемы по порядку будет совпадать со сложностью исходной программы. Однако возможна ситуация, когда значительная доля операторов P является операторами остановки и между большинством соседних операторов остановки выходные переменные

вообще не вычисляются, а происходит только определение новых условий остановки. В этом случае сложность построенной схемы по порядку будет в m раз больше сложности исходной программы. В следующей теореме описывается способ преобразования произвольной программы в схему при котором сложность схемы не более чем в четыре раза превосходит сложность программы.

Теорема 7.5.4. *Произвольная программа* P, *вычисляющая булев* (m,n)-*оператор, может быть преобразована в схему из функциональных элементов* S *так, что*:

(*i*) $L(\mathsf{S}) \le 4C(\mathsf{P})$;

(*ii*) $\mathsf{S}(x_1,\dots,x_n) = \mathsf{P}(x_1,\dots,x_n)$ *при всех* $(x_1,\dots,x_n)$.

Доказательство. Пусть $\mathsf{P} = \mathsf{p}_1 \dots \mathsf{p}_L$ — произвольная программа, $\mathsf{s}_1,\dots,\mathsf{s}_r$ — все ее операторы остановки, $\mathsf{q}_1,\dots,\mathsf{q}_r$ — нулевые аргументы операторов остановки. Как и в (7.1.1) полагаем, что i-й оператор остановки s_i программмы P является ее t_i-м оператором. Предположим, что l-я выходная переменная программы P вычисляется только перед операторами остановки индексы которых принадлежат множеству $\{i_1,\dots,i_k\}$, т.е. $\mathsf{z}_l(\boldsymbol{x};t_i) = \mathsf{z}_l(\boldsymbol{x};t_{i_s})$ при всех $i \in \{i_s,\dots,i_{s+1}-1\}$. Положим

$$h'_0(\boldsymbol{x}) \equiv 1, \qquad h'_k(\boldsymbol{x}) = \bigwedge_{i=1}^{l} \overline{\mathsf{q}}_i(\boldsymbol{x}), \qquad \text{при } k \in \{1,2,\dots,r\},$$
$$h_{r+1}(\boldsymbol{x}) = h'_r(\boldsymbol{x}), \quad h_k(\boldsymbol{x}) = h'_{l-1}(\boldsymbol{x})\mathsf{q}_j(\boldsymbol{x}), \text{ при } k \in \{1,2,\dots,r\}.$$

Используя введенные функции h_i, преобразуем (7.1.1):

$$\mathsf{P}_l(\boldsymbol{x}) = h_1(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_1) \vee \dots \vee h_r(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_r) \vee h_{r+1}(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};L).$$

Теперь при всех i,j таких, что $1 \le i < j \le r+1$, определим функции

$$h_{i,j}(\boldsymbol{x}) = \bigvee_{k=i}^{j} h_k(\boldsymbol{x}).$$

Так как $\mathsf{z}_l(\boldsymbol{x};t_i) = \mathsf{z}_l(\boldsymbol{x};t_{i_s})$ при всех $i \in \{i_s,\dots,i_{s+1}-1\}$, то предыдущее равенство после несложных преобразований приводится к виду

$$\begin{aligned}\mathsf{P}_l(\boldsymbol{x}) =& h_{1,i_1-1}(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_1) \vee h_{i_1,i_2-1}(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_{i_1}) \vee \dots \\ &\dots \vee h_{i_{k-1},i_k-1}(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_{i_{k-1}}) \vee h_{i_k,r+1}(\boldsymbol{x})\mathsf{z}_l(\boldsymbol{x};t_{i_k}).\end{aligned} \tag{7.5.9}$$

Функции h_i и h'_j определены так, что

$$h_i(\boldsymbol{x})h'_j(\boldsymbol{x}) = \begin{cases} 0, & \text{при } i \le j, \\ h_i(\boldsymbol{x}), & \text{при } i > j. \end{cases}$$

Поэтому при $i < j$

$$h_{1,j}(\boldsymbol{x})h'_i(\boldsymbol{x}) = h_{i+1}(\boldsymbol{x}) \vee \dots \vee h_j(\boldsymbol{x}) = h_{i+1,j}(\boldsymbol{x}),$$

т.е. при вычисленных функциях $h_{1,i}$ и h'_i для вычисления каждой функции $h_{i,j}$, встречающейся в (7.5.9), достаточно одного функционального оператора.

Допустим, что программа P содержит L_1 функциональных операторов. Тогда преобразование P в схему S состоит в следующем:

(a) Вычисляем все функции h_i, h'_i и $h_{1,i}$. Для этого потребуется $3r-2$ операторов.

(b) Вычисляем все необходимые функции $h_{i,j}$. Для этого потребуется столько операторов, сколько раз в программе P вычисляются выходные переменные. Так как каждый раз каждая выходная переменная вычисляется собственным функциональным оператором, то потребуется не более L_1 операторов.

(c) В соответствии с равенством (7.5.9) вычисляем все компоненты P_l. Для этого потребуется не более $2L_1$ операторов.

(d) Удаляем все операторы остановки.

Легко видеть, что общее число дополнительных операторов не превосходит $3L$. Следовательно, сложность схемы S не превосходит $4L$. Теорема доказана.

Задачи

7.5.1. Оценить сверху $\max C(f)/T(f)$, где максимум берется по всем функциям n переменных одинаковой сложности L.

7.5.2. Оценить $\max C(f)/T_d(f)$ для произвольного d, бóльшего n.

7.5.3. Будем говорить, что функция $f : \mathbb{B}^n \to \mathbb{B}$ полиномиально сводится к функции $g : \mathbb{B}^n \to \mathbb{B}$, если найдется такая функция $h : \mathbb{B}^{n+1} \to \mathbb{B}$, что

$$f(x_1, \ldots, x_n) = h(x_1, \ldots, x_n, g(x_1, \ldots, x_n)), \quad C(g) = \mathcal{O}(n^c),$$

где c — константа. Показать, что найдутся булева функция n переменных g и полиномиально сводящаяся к ней булева функция n переменных f такие, что

$$C(g) \asymp C(f) \asymp \mathcal{O}\bigl(\sqrt{2^n/n}\bigr), \quad T(g)/T(f) = \mathcal{O}\bigl(\sqrt{2^n/n}\bigr).$$

7.5.4. Показать, что программа P, вычисляющая булеву функцию f, может быть преобразована в схему из функциональных элементов S, вычисляющую f, так, что $L(\mathsf{S}) \leq 2C(\mathsf{P})$.

Литература

[1] *Алексеев В. Б., Ложкин С. А.* Элементы теории графов, схем и автоматов. — М.: ВМК МГУ, 2000.

[2] *Ансель Ж.* О числе монотонных булевых функций n переменных. — В кн.: Кибернетический сборник. Новая серия. Вып.5. — М.: Мир, 1968, с. 53–63.

[3] *Ахо А., Хопкрофт Дж., Ульман Дж.* Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979.

[4] *Гаврилов Г. П., Сапоженко А. А.* Задачи и упражнения по курсу дискретной математики. — 2-е изд. М.: Наука, 1992.

[5] *Гельфанд И. М.* Лекции по линейной алгебре. — 3-е изд. М.: Наука, 1966.

[6] *Гринчук М. И.* О монотонной сложности пороговых функций. — В сб.: Дискретный анализ. Вып. 52. — Новосибирск, 1992, с. 31–110.

[7] *Ежов И. И., Скороход А. В., Ядренко М. И.* Элементы комбинаторики. — М.: Наука, 1977.

[8] *Кнут Д.Э.* Искусство программирования для ЭВМ. Т. 2: Получисленные алгоритмы. — М.: Мир, 1977.

[9] *Кнут Д.Э.* Искусство программирования для ЭВМ. Т. 3: Сортировка и поиск. — М.: Мир, 1978.

[10] *Кострикин А. И.* Введение в алгебру. — М.: Наука, 1977.

[11] *Ложкин С. А., Семенов А. А.* Об одном методе сжатия информации и о сложности реализации монотонных симметрических функций. — Известия ВУЗ, Математика, 1988, № 7, с 11–19.

[12] *Лупанов О. Б.* О синтезе некоторых классов управляющих систем. — В кн.: Проблемы кибернетики. Вып. 10. — М.: Физматгиз, 1963, с. 63–97.

[13] *Лупанов О. Б.* Об одном подходе к синтезу управляющих систем — принципе локального кодирования. — В кн.: Проблемы кибернетики. Вып. 14. — М.: Физматгиз, 1965, с. 31–110.

[14] *Лупанов О. Б.* Асимптотические оценки сложности управляющих систем. — М.: МГУ, 1984.

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

[16] *Питерсон У., Уэлдон Э.* Коды, исправляющие ошибки. — М.: Мир, 1976.

[17] *Редькин Н. П.* Доказательство минимальности некоторых схем из функциональных элементоа. — В кн.: Проблемы кибернетики. Вып. 23. — М.: Наука, 1970, с. 83–101.

[18] *Редькин Н. П.* О минимальной реализации двоичного сумматора. — В кн.: Проблемы кибернетики. Вып. 38. — М.: Наука, 1981, с. 181–216.

[19] *Сэвидж Дж. Э.* Сложность вычислений. — М.: Факториал, 1998.

[20] *Чашкин А. В.* О сложности булевых матриц, графов и соответствующих им булевых функций. — Дискретная математика, 1994, №2, с. 43–73.

[21] *Чашкин А. В.* О среднем времени вычисления значений булевых функций. — Дискретный анализ и исследование операций, 1997, №1, с. 60–78.

[22] *Чашкин А. В.* О среднем времени вычисления булевых операторов. — Дискретный анализ и исследование операций, 1998, №1, с. 88–103.

[23] *Чашкин А. В.* Среднее время вычисления значений элементарных булевых функций. — Дискретная математика, 2000, №4, с. 109–120.

[24] *Чашкин А. В.* Средняя сложность булевых функций. — В кн.: Дискретная математика и ее приложения: сборник лекций. — М. Мех.-Мат. МГУ 2001, с. 145–170.

[25] *Шоломов Л. А.* О реализации недоопределенных булевых функций схемами из функциональных элементов. — В кн.: Проблемы кибернетики. Вып. 21. — М.: Наука, 1969, с. 215–226.

[26] *Яблонский С. В.* Введение в теорию функций k-значной логики. — В кн.: Дискретная математика и математические вопросы кибернетики, т. 1, под ред. С. В. Яблонского и О. Б. Лупанова. — М.: Наука, 1974, с. 9–66.

[27] *Яблонский С. В.* Введение в дискретную математику. — 3-е изд. М.: Высш. школа, 2001.

[28] *Akl S. J.* Parallel Computation: Models and Methods. — Prentice-Hall, 1997.

[29] *Ajtai M., Komlos Ja., Szemeredi E.* Sorting in $O(n \log n)$ parallel steps. — Combinatorica, 1983, v. 3. № 1, pp. 1-19.

Предметный указатель