Московский государственный технический университет
имени Н.Э. Баумана

Факультет «Фундаментальные науки»
Кафедра «Математическое моделирование»

А.Н. Канатников

ДИСКРЕТНАЯ МАТЕМАТИКА

Конспект лекций

Для студентов специальности
«Прикладная математика»

Москва
2006

1. БУЛЕВЫ ФУНКЦИИ

1.1. Булевы алгебры

Булева алгебра: симметричное полукольцо с унарной операцией дополнения $x \to \overline{x}$ с аксиомами $x + \overline{x} = 1$, $x\overline{x} = 0$. Единственность дополнения. Булевы алгебры и булевы кольца. Булева алгебра как решетка. Решетка — алгебра $(L, \vee, \wedge)$ в которой операции (hешеточные объединение и пересечение) ассоциативны, коммутативны, идемпотентны и связаны тождествами поглощения $a \vee (a \wedge b) = a$, $a \wedge (a \vee b) = a$. Всякое симметричное полукольцо является решеткой, а дистрибутивная решетка (каждая операция дистрибутивна относительно другой) с нулем и единицей — симметричным полукольцом. Пример булевой алгебры — булев куб $\mathbb{B}^n$. Теорема: любая конечная булева алгебра изоморфна булеву кубу некоторой размерности.

Напомним, что по определению булева алгебра — это симметричное полукольцо, в котором введена еще одна унарная операция — дополнение. Дополнение элемента x, обозначаемое $\overline{x}$, удовлетворяет аксиомам $x + \overline{x} = 1$, $x\overline{x} = 0$. Отметим, что эти два равенства для любого x определяют элемент $\overline{x}$ однозначно, т.е. существование дополнения определяется свойствами двух бинарных операций.

Подробнее можно сказать, что булева алгебра — универсальная алгебра вида $(L,\ \{+,\ \cdot,\ ^{-}\})$, удовлетворяющая условиям:

а) каждая бинарная операция коммутативна, ассоциативна, идемпотентна и имеет нейтральный элемент;

б) каждая бинарная операция дистрибутивна относительно другой;

в) нейтральный элемент каждой бинарной операции является аннулирующим другой бинарной операции;

г) дополнение связано с бинарными операциями аксиомами $x + \overline{x} = 1$, $x\overline{x} = 0$.

Обе операции порождают два отношения порядка, являющиеся взаимно обратными. Так, отношение $x \leqslant y$, равносильное равенству $x + y = y$, также определяется равенством $xy = x$.

Если в булевой алгебре B ввести операцию $x \oplus y = x\overline{y} + \overline{x}y$, то получим алгебраическую систему $(L,\ \{\oplus,\ \cdot\})$, которая является кольцом с единицей, в котором операция умножения идемпотентна (булевым кольцом). Короче говоря, булева алгебра является булевым кольцом. Наоборот, в любом булевом кольце $(L,\ \{\oplus,\ \cdot\})$ умножение коммутативно, а сложение удовлетворяет условию $x \oplus x = 0$. Положив $x + y = x \oplus y \oplus xy$, $\overline{x} = x + 1$, получим алгебраическую систему $(L,\ \{+,\ \cdot,\ ^{-}\})$, представляющую собой булеву алгебру.

К понятию булевой алгебры можно подойти и с другой стороны. Коммутативная идемпотентная полугруппа называется ***полурешеткой***. В полурешетке (как и в полукольце) можно ввести отношение естественного порядка согласно правилу $x \leqslant y \Leftrightarrow x+y = y$. Наоборот, любое упорядоченное множество, в котором любое двухэлементное множество имеет точную верхнюю грань, можно интерпретировать как полурешетку с операцией $x + y = \sup\{x,\ y\}$. Если в упорядоченном множестве каждое двухэлементное множество имеет и точную верхнюю, и точную нижнюю грани, то на это множестве можно ввести структуру полурешетки двумя способами: с операцией $\sup\{x,\ y\}$ и с операцией $\inf\{x,\ y\}$. Первую структуру называют верхней полурешеткой, а вторую — нижней. Две решеточные операции связаны тождествами

$$\sup\{x,\ \inf\{x, y\}\} = x, \quad \inf\{x,\ \sup\{x, y\}\} = x.$$

Алгебра $(R,\ \{\vee,\ \wedge\})$, для которой $(R,\ \vee)$ и $(R,\ \wedge)$ есть полурешетки, причем операции связаны тождествами поглощения $a \vee (a \wedge b) = a$, $a \wedge (a \vee b) = a$, называется решеткой. Бинарные операции называются решеточными (решеточное объединение и решеточное пересечение). Решеточные операции входят в структуру решетки симметрично. Поэтому любую из них можно назвать объединением, тогда втораябудет пересечением. Структуру решетки имеет любое упорядоченное множество, в котором для любого двухэлементного множества существует точная верхняя и точная нижняя грани. В этом случае решеточными операциями будут $a \vee b = \sup\{a,\ b\}$ и $a \wedge b = \inf\{a,\ b\}$.

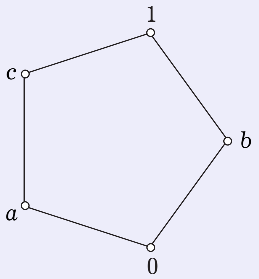

Рис. 1.1

Структуру решетки имеет любое симметричное полукольцо. Решетка в общем случае может и не быть симметричным полукольцом. Например, ***пентагон*** — упорядоченное множество $\{0,\ a,\ b,\ c,\ 1\}$ с отношением порядка, заданным диаграммой Хассе[*] на рис. 1.1, — является решеткой, но в этой решетке операции не являются дистрибутивными друг относительно друга, как должно быть в симметричном полукольце. Однако всякая дистрибутивная решетка (т.е. решетка, в которой каждая операция дистрибутивна относительно другой) с нулем и единицей (нейтральными элементами объединения и пересечения) является симметричным полукольцом.

Простейший пример булевой алгебры — *двухэлементная булева алгебра*

$$\mathbb{B} = (\{0,\ 1\},\ \{+,\ \cdot,\ {}^{-}\})$$

с операциями $x+y = \max\{x,\ y\}$, $xy = \min\{x,\ y\}$, $\overline{x} = x+1$. Обобщением этого примера является булева алгебра $\mathbb{B}^n$ с покомпонентным выполнением операций алгебры $\mathbb{B}$ на кортежах. Алгебру $\mathbb{B}^n$ называют n-й декартовой степенью $\mathbb{B}$ или ***n-мерным булевым кубом***. Оказывается, что любая конечная булева алгебра изоморфна некоторому булеву кубу. Попросту говоря, других примеров конечных алгебр, отличных от булева куба, нет.

Другой важный пример булевой алгебры — алгебра $\mathcal{S}_A = \left(2^A,\ \{\cup,\ \cap,\ {}^{-}\}\right)$, т.е. множество всех подмножеств множества A с теоретико-множественными операциями объединения, пересечения и дополнения. Отметим, что все практически значимые примеры булевых алгебр можно интерпретировать в рамках алгебры $\mathcal{S}_A$ или какой-либо ее подалгебры. Например, элементы булевой алгебры $\mathbb{B}^n$ (n-мерные булевы векторы) можно рассматривать как запись характеристической функции подмножества в множестве из n элементов. А тогда операции в $\mathbb{B}^n$ будут соответствовать теоретико-множественным операциям. Симметричное полукольцо $([a,\ b],\ \{\max,\ \min\})$ можно преобразовать в алгебру множеств, поставив в соответствие каждому $x \in [a,\ b]$ отрезок $[a,\ x]$. Множество таких отрезков замкнуто относительно операций объединения и пересечения, т.е. образует подалгебру в $2^{[a,b]}$. Правда это полукольцо не является булевой алгеброй, так как в нем нельзя ввести операцию дополнения.

1.2. Булевы функции

Булева функция — отображение $f\colon B^n \to B$, где B — некоторая булева алгебра. Наибольший интерес — двухэлементная алгебра $\mathbb{B}$. Табличный способ задания булевой функции. Примеры – функции одной и двух переменных. Вектор значений булевой функции. Аналитический способ. Язык булевой алгебры: тройка (P, C, F) (переменные, константы, функциональные символы). Интерпретация языка: сопоставление каждому функциональному символу арности n конкретной булевой функции от n аргументов. Синтаксическое дерево. Подформулы и значение формулы. Неоднозначность соответствия формул и функций. Эквивалентность функций и эквивалентность формул. Теорема о замене подформулы эквивалентной и теорема о замене переменной в эквивалентных формулах. Замыкание множества булевых функций. Замкнутые и полные множества функций.

Пусть задана булева алгебра $(B,\ \{\vee,\ \wedge,\ {}^{-}\})$. Булева функция — это отображение $f\colon B^n \to$ $\to B$, т.е. функция от n переменных, область изменения каждой из которых есть сама алгебра, причем значениями функции также являются элементы булевой алгебры. Мы остановимся на

[*] Диаграмма Хассе — изображение в виде неориентированного графа конечного упорядоченного множества. При этом ребрами соединяются вершины, связанные отношением доминирования, причем вершина, изображенная ниже, предшествует расположенной выше.

простейшем случае, имеющем наибольший интерес, когда B — это двухэлементная алгебра $\mathbb{B}$. В этом случае каждый аргумент функции принимает два возможных значения 0 и 1 и сама функция принимает лишь два тех же значения. Подобная функция представляет собой отображение $f\colon \mathbb{B}^n \to \mathbb{B}$. Комбинируя такие функции, можно получить любое отображение вида $F\colon \mathbb{B}^n \to \mathbb{B}^m$, а в конечном счете булевы функции нескольких переменных для любой конечной булевой алгебры.

Нетрудно подсчитать количество возможных булевых функций вида $f\colon \mathbb{B}^n \to \mathbb{B}$: это количество равно 2^{2^n} (по формуле Y^X, где X количество элементов области определения, а Y — количество элементов области значений).

Поскольку и область определения, и область значений конечны, булевы функции удобно задавать с помощью таблиц. Элементы области определения — n-мерные векторы, которые можно рассматривать как двоичные записи целых неотрицательных чисел от 0 до 2^n-1. Таблица, определяющая булеву функцию, может выглядеть следующим образом (табл. 1.1).

Таблица 1.1

$x_1,\ x_2,\ \ldots,\ x_n$	$f(x_1, x_2, \ldots, x_n)$
$0,\ 0,\ \ldots,\ 0$	$f(0, 0, \ldots, 0)$
. . . .	. . . .
$\alpha_{k1},\ \alpha_{k2},\ \ldots,\ \alpha_{kn}$	$f(\alpha_{k1}, \alpha_{k2}, \ldots, \alpha_{kn})$
. . . .	. . . .
$1,\ 1,\ \ldots,\ 1$	$f(1, 1, \ldots, 1)$

В качестве примера приведем таблицы булевых функций одного переменного (табл. 1.2) и семи наиболее важных функций двух переменных (табл. 1.3).

Таблица 1.2

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	0	0	1	1
1	1	0	1	0

Таблица 1.3

$x_1,\ x_2$	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$	$f_5(x)$	$f_6(x)$	$f_7(x)$
00	0	0	0	1	1	1	1
01	1	0	1	1	0	1	0
10	1	0	1	0	0	1	0
11	1	1	0	1	1	0	0

Отметим, что булева функция n переменных представляет собой n-арную операцию на булевой алгебре $\mathbb{B}$. В частности, функции двух переменных — это бинарные операции, для которых обычно используют инфиксную форму записи. Так, для функций приведенных в табл. 1.3, используют следующие названия и символы операций:

$f_1(x_1, x_2) = x_1 \vee x_2 = x_1 + x_2$ — дизъюнкция;
$f_2(x_1, x_2) = x_1 \wedge x_2 = x_1 \cdot x_2$ — конъюнкция;
$f_3(x_1, x_2) = x_1 \oplus x_2$ — сложение по модулю 2;
$f_4(x_1, x_2) = x_1 \to x_2$ — импликация;
$f_5(x_1, x_2) = x_1 \sim x_2$ — эквивалентность;
$f_6(x_1, x_2) = x_1 \mid x_2$ — штрих Шеффера („не и“);
$f_7(x_1, x_2) = x_1 \downarrow x_2$ — стрелка Пирсона („не или“).

Функции f_1 и f_2 — базовые операции булевой алгебры $\mathbb{B}$, т.е. сложение и умножение (или, по-другому, решеточное объединение и пересечение), но в теории булевых функции традиционно предпочитают названия, указывающие на связь с математической логикой. Сложение по модулю 2 — альтернатива дизъюнкции, превращающая полукольцо $\mathbb{B}$ в поле $\mathbb{Z}_2$. Штрих Шеффера и стрелка Пирсона — это отрицания соответственно конъюнкции и дизъюнкции, т.е.

$$x_1 \mid x_2 = \overline{x_1 \wedge x_2}, \qquad x_1 \downarrow x_2 = \overline{x_1 \vee x_2}.$$

Инфиксная форма записи и представление штриха Шеффера и стрелки Пирсона с помощью операций булевой алгебры указывает еще на один способ представления булевых функций — аналитический. Вообще говоря, любая алгебраическая система содержательна, если для нее удается создать такие формы записи операций и их последовательностей, для которых возникают многообразные способы преобразования выражений. Речь идет о понятии „формула“.

Формула интуитивно понимается как запись с помощью определенного языка некоторой последовательности действий над объектами алгебраической системы.

В самом общем смысле под языком понимают следующее. Рассмотрим некоторое непустое конечное множество V, элементы которого будем называть ***символами***. Произвольные последовательности символов называются ***словами*** или ***цепочками***. Слова могут быть любой длины. Вводится специальное слово нулевой длины. Множество всех слов обозначим V^*. Это множество можно представить как объединение всех декартовых степеней множества V (нулевой, содержащей слово нулевой длины, первой со словами длины 1, второй со словами длины 2 и т.д.). ***Язык*** — это некоторое подмножество в V^*, т.е. некоторый фиксированный набор слов, составленных с помощью элементов множества V, которое называют ***алфавитом языка***.

Конкретный язык формируется правилами, по которым из всех слов, составленных с помощью данного алфавита, выбираются „правильные“ или допустимые слова. В нашем случае словами должны быть правильно записанные формулы булевой алгебры, а символами — символы переменных, знаков операций и элементов алгебры. Язык формул можно построить в рамках данного выше понятия формального языка. Однако, упрощая изложение, мы снимем ограничение конечности алфавита языка*, полагая, что алфавит может быть счетным множеством. Строго формальный математический язык можно ввести следующим образом.

Пусть дана тройка объектов $(P,\ C,\ F)$, где P — счетное множество, элементы которого мы будем называть ***булевыми переменными***; C — счетное множество символов, используемых для обозначения элементов алгебраической системы (в данном случае двух элементов 0 и 1 алгебры $\mathbb{B}$), которые мы будем называть ***константами***; F — счетное множество ***функциональных символов***, каждый из которых имеет натуральную характеристику — ***арность***.

Понятие формулы в рассматриваемом языке вводится индуктивно, т.е. указываются конкретные способы построения правильных формул на основе уже построенных формул. Базис индукции — изначальный набор ***элементарных формул***, под которыми мы будем понимать элементы множества $P \cup C$. Шаг индукции описывает, как из уже построенных формул получаются новые: если $X_1,\ X_2,\ \ldots,\ X_n$ — формулы и $f \in F$ — функциональный символ арности n, то $f(X_1, X_2, \ldots, X_n)$ — формула. Кратко сказанное можно сформулировать так:

1) элементы множества $P \cup C$ — формулы;

2) если $X_1,\ X_2,\ \ldots,\ X_n$ — формулы и $f \in F$ — функциональный символ арности n, то $f(X_1, X_2, \ldots, X_n)$ — формула

Первый пункт определения — базис индукции. Второй (и последующие, если они есть) определяет правила построения новых формул из уже построенных. Это шаг индукции.

В данном случае под алфавитом следует понимать объединение $P \cup C \cup F$, к которому добавлены еще три служебных символа: две круглые скобки и запятая. Но заметим, что при таком подходе нарушается требование конечности алфавита. Мы сознательно идем на такое нарушение, чтобы не усложнять язык формул. ***Язык булевой алгебры*** — это множество всех правильно составленных формул.

Хотя мы выделили константы, т.е. символы, обозначающие объекты рассматриваемой алгебраической системы, в самостоятельный класс, часто удобно рассматривать их как специальный вид функциональных символов арности 0. Таким символам соответствуют постоянные булевы функции.

Индуктивное определение формулы позволяет корректно составлять сами формулы, как последовательности символов, но не придает им какого-либо смысла. Чтобы наполнить формулы смыслом, необходимо каждому функциональному символу сопоставить конкретную операцию

*Ситуацию здесь можно пояснить следующим образом. При записи математических формул используют числовые константы, которые фигурируют в десятичной форме записи. Такая запись включает десять цифр знак числа и десятичную запятую. правильных записей чисел существует счетное число. Но рассматривая сложные конструкции формул, можно каждую такую запись считать самостоятельным символом, не раскрывая способ формирования записи.

соответствующей арности (конкретную булеву функции с соответствующим количеством аргументов). Другими словами, необходимо задать отображение множества F в множество всех булевых функций, при котором арность функционального символа совпадает с арностью операции. Кроме того, необходимо фиксировать смысл символов, предназначенных для обозначения констант, т.е. задать отображение множества C в алгебру $\mathbb{B}$. Два указанных отображения составляют ***интерпретацию языка***.

Следует учесть еще одно обстоятельство. Общепринятым является запись бинарных операций не в виде $f(x_1, x_2)$, а в виде $x_1\ f\ x_2$ (как правило, со специальным символом операции). Поэтому в наш язык следует ввести символы инфиксных операций, а в качестве шага индукции ввести дополнительное правило:

3) если X_1 и X_2 — формулы, то и слова $(X_1 + X_2)$, $(X_1 \cdot X_2)$, $(X_1 \oplus X_2)$, $(X_1 \to X_2)$, $(X_1 \sim X_2)$, $(X_1 \mid X_2)$, $(X_1 \downarrow X_2)$ являются формулами.

При использовании инфиксной формы записи бинарных операций в формулах оказывается большое количество пар скобок, что затрудняет восприятие таких формул. Скобки нужны для того, чтобы однозначно определить последовательность шагов, которые приводят от элементарных формул к конечной формуле. Чтобы сократить количество скобок и сделать формулы более читаемыми, используют правило приоритетов, согласно которому каждой бинарной операции с инфиксной формой записи присваивается приоритет — некоторое натуральное число. Сперва выполняются операции с большим приоритетом, затем с меньшим. Среди операций с одинаковым приоритетом в первую очередь выполняется та, знак которой находится левее. Мы будем использовать инфиксную форму на практике, как более привычную и удобную, но в теоретических рассуждениях с целью упрощения ограничимся лишь префиксной формой записи операций.

Рассматривая различные множества F функциональных символов и различные их интерпретации, мы можем строить различные схемы построения формул, которые различаются набором базовых функций. Например, в качестве базовых могут рассматриваться три базовые операции булевой алгебры: сложение, умножение и отрицание, но могут быть и другие варианты (например, сложение по модулю 2 и умножение — базовые операции в $\mathbb{Z}_2$).

Любая корректная формула либо элементарная, либо получена в результате шага индукции, т.е. с помощью функционального символа f и некоторого набора формул $X_1, \ldots, X_n$. Каждая из формул X_i в свою очередь либо элементарная формула, либо получена с помощью своего функционального символа f_i и некоторого набора формул $X_{I1}, \ldots, X_{in_i}$ и т.д. Двигаясь от конечной формулы, мы за конечное число шагов придем к элементарным формулам — переменным и константам. Представленный анализ формулы называется синтаксическим. Его результаты можно представить в виде ориентированного дерева. Корень дерева — сам формула, вершины глубины 1, формулы X_i и т.д. Каждому листу дерева соответствует переменная или константа. Вершинам, не являющимся корнем соответствуют формулы, которые входят как составная часть в анализируемую формулу. Они называются ***подформулами***.

Пример 1.1. Рассмотрим формулу $f_1(f_2(x_1, x_2), f_3(x_2), f_4(x_1, f_2(x_3), x_4))$. Эта формула получена с помощью функционального символа f_1 арности 3. Подформулами глубины 1 являются $f_2(x_2)$, $f_3(x_2)$ и $f_4(x_1, f_2(x_3), x_4)$. Первая из них имеет подформулу — переменную x_2, вторая имеет также одну подформулу — переменную x_2, а третья — уже три подформулы: первая и третья — переменные x_1 и x_4, вторая подформула — сложная подформула $f_2(x_3)$. В результате получаем дерево синтаксического анализа, представленное на рис. 1.2. Высоту дерева (в данном случае 3) называют ***сложностью формулы***.

Основная задача в связи с аналитическим способом задания булевых функций — описание класса функций, представимых формулами. В дальнейшем мы для упрощения изложения будем отождествлять множество функциональных символов языка с соответствующим множеством булевых функций. Можно считать, что каждой булевой функции соответствует свой уникальный символ и F состоит как раз из таких символов, так что, задав F, мы тем самым задаем и интерпретацию формул.

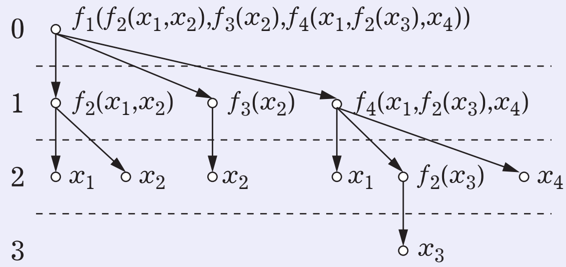

Рис. 1.2

Один из способов получения новых формул — замена одной из подформул другой. Такое преобразование называется ***подстановкой***. Замене может подлежать как сложная формула, так и любая элементарная.

В формулу, как правило, входят переменные (булевы переменные). Но есть формулы, в которых элементарные составляющие — все константы. Такие формулы имеют определенное значение, соответствующее конкретному объекту алгебраической системы (т.е. в нашем случае 0 и 1). Если в формулу входят переменные, то ее значение не определено. Однако если в формуле заменить каждую переменную константой (т.е. дать переменной конкретное значение), получим новую формулу, имеющую значение. Таким образом, каждому набору значений переменных, входящих в формулу, соответствует конкретное значение формулы.

Это обстоятельство позволяет рассматривать формулу как форму записи функции. Однако надо иметь в виду, что с помощью одной и той же формулы можно задавать различные функции. Например, формула $x-t$ определяет функции $f(x,t=x-t$ и $f(t,x)=x-t$, а также, например, функцию $f(x,y,t)=x-t$.

Из этого примера можно сделать вывод: чтобы с помощью формулы задать функцию, необходимо установить соответствие между аргументами функции и переменными, входящими в формулу. Каждой переменной входящей в формулу, должен соответствовать один аргумент функции, некоторые из аргументов могут быть ене связаны с какими-либо переменными. Если такое соответствие установить, то набор значений переменных, входящих в формулу, можно записать как булев вектор. При этом каждому булевому вектору будет соответствовать значение формулы, и мы получаем корректно заданную булеву функцию.

Рассматривая формулы как способ описания функций, естественно не различать те формулы, которые порождают одинаковые функции при фиксированном соответствии между переменными и аргументами функции.

На практике мы часто пользуемся эквивалентными преобразованиями. Например, если f_1 — булева функция из табл. 1.3, то $f_1(x_1,x_2)=f_1(x_2,x_1)$. Знак равенства означает, что формулы задают одну и ту же функцию, т.е. эквивалентны. Вообще говоря, знак равенства в данном контексте рассматривается как знак тождества, т.е. формулы дают одно и то же значение при любых значениях входящих в равенство переменных. Если состав переменных слева и справа один и тот же, то тождество как раз и означает, что левая и правая части задают одну и ту же функцию. Однако возможно и такое равенство $x\oplus x=0$, также верное при любых значениях неизвестных. Но в левую часть входит переменная x, а в правой части вообще стоит константа. Левая часть задает функцию двух переменных, а правая часть — нульарную операцию (константу). Надо либо отказаться признать такое равенство, либо установить эквивалентность каких-либо функций. Второе более удобно.

Две формулы* $\Phi[x_1,\ldots,x_n]$ и $\Psi[y_1,\ldots,y_m]$ с наборами переменных x_1, …, x_n и y_1, …, y_m соответственно назовем ***эквивалентными***, если равенство $\Phi[x_1,\ldots,x_n]=\Psi[y_1,\ldots,y_m]$ остается верным при любых значениях всех переменных. Если наборы переменных не совпадают, то значение одной из формул или обеих не зависит от значений некоторых переменных. Такие

*Использование квадратных скобок подчеркивает, что это не обозначение функции. Здесь имеется в виду формула, обозначенная как Φ, все переменные которой входят в указанный список переменных.

переменные назовем ***фиктивными***. Переменную, не являющуюся фиктивной, будем называть ***существенной***. У двух эквивалентных формул существенные переменные должны быть общими, хотя часть общих переменных может быть и фиктивными.

Следующая теорема отражает обычные правила преобразования формул, приводящие к эквивалентным формулам.

Теорема 1.1. Если формулы $\Phi[x_1, \ldots, x_n]$ и $\Psi[y_1, \ldots, y_m]$ эквивалентны, то замена в каждой из них всех вхождений какой-либо из переменных произвольной формулой приведет к двум новым формулам, эквивалентным между собой.

◄ Объединим списки переменных у формул и напишем равенство

$$\Phi[z_1, \ldots, z_k] = \Psi[z_1, \ldots, z_k].$$

Предположим, что все вхождения переменной z_k заменены формулой $\Delta[u_1, \ldots, u_l]$. Получим формулы $\widetilde{\Phi}[z_1, \ldots, z_{k-1}, u_1, \ldots, u_l]$ и $\widetilde{\Psi}[z_1, \ldots, u_1, \ldots, u_l]$. Обозначения исходят из того, что переменные формулы Δ не входят в формулы Φ и Ψ. На самом деле это допущение несущественно и служит лишь упрощению рассуждений.

Задав произвольные значения переменным z_1, ..., z_{k-1}, u_1, ..., u_l, мы получим для переменной z_k значение $\Delta(u_1, \ldots, u_l)$. Проверка равенства значений двух формул равносильна проверке исходного равенства, в котором в качестве значения переменной z_k взято $\Delta(u_1, \ldots, u_l)$. Поскольку исходное равенство верно, то и после подстановки равенство останется верным. ►

Теорема 1.2. Если в формуле заменить одну из подформул эквивалентной, то новая формула будет эквивалентна исходной.

◄ Пусть дана формула $\Phi[x_1, \ldots, x_n]$, в которую входит подформула $\Psi[x_1, \ldots, x_n]$ (мы можем считать, что подформула включает все переменные исходной формулы, рассматривая недостающие как фиктивные). Заменим подформулу Ψ новой переменной z, которая не входит в список x_1, ..., x_n. Получим новую формулу $\Gamma[x_1, \ldots, x_n, z]$, связь которой с исходной формулой можно записать в виде

$$\Phi[x_1, \ldots, x_n] = \Gamma[x_1, \ldots, x_n, z]\Big|_{z=\Psi[x_1,\ldots,x_n]}.$$

Замена формулы $\Psi[x_1, \ldots, x_n]$ эквивалентной формулой $\widetilde{\Psi}[x_1, \ldots, x_n]$ приведет к новой формуле

$$\Gamma[x_1, \ldots, x_n, z]\Big|_{z=\widetilde{\Psi}[x_1,\ldots,x_n]}.$$

Надо показать, что при любых значениях неизвестных верно равенство

$$\Gamma[x_1, \ldots, x_n, z]\Big|_{z=\Psi[x_1,\ldots,x_n]} = \Gamma[x_1, \ldots, x_n, z]\Big|_{z=\widetilde{\Psi}[x_1,\ldots,x_n]}.$$

Но нетрудно заметить, что, задавая значения неизвестных, мы в силу эквивалентности формул Ψ и $\widetilde{\Psi}$ получим один и тот же набор значений переменных в формуле Γ. Значит, в обеих частях равенства мы получим одно и то же значение. ►

В функции $f(x_1, \ldots, x_n)$ назовем i-й аргумент ***фиктивным аргументом***, если значение функции не зависит от значения этого аргумента, т.е. функции

$$f(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \quad \text{и} \quad f(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

совпадают. Вместо двух этих равных друг другу функций можно рассмотреть функцию

$$\tilde{f}(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) = f(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n),$$

у которой количество аргументов на единицу меньше. Такое преобразование назовем ***удалением фиктивного аргумента***.

Возможно и противоположное преобразование — ***введение фиктивных переменных***, а именно:

$$\hat{g}(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n) = g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n).$$

Две функции назовем ***эквивалентными***, если после удаления в каждой из них всех фиктивных аргументов, мы получим равные функции.

Нетрудно проверить, что введенное отношение на множестве булевых функций действительно является отношением эквивалентности. Точно так же на множестве все булевых формул эквивалентность (в смысле равенства значений при любых значениях переменных) — это отношение эквивалентности.

Введение этих отношений преследует единственную цель: обозначить степень неоднозначности, с которой функции записываются формулами. Можно утверждать, что при фиксированном порядке переменных каждому классу эквивалентных формул соответствует ровно один класс эквивалентных функций. Вопрос: является ли это соответствие биекцией? Можно ли утверждать, что каждый класс эквивалентных функций определяется некоторым классом эквивалентных формул? Ответ зависит от выбора набора базовых функций, или ***базиса***, т.е. множества F.

Пусть F — некоторое множество функций. Его ***замыканием*** назовем множество $[F]$ всех булевых функций, которые представимы формулами с базисом F (***формулами над множеством*** $\boldsymbol{F}$). Если замыкание множества F совпадает с F, то это множество называется ***замкнутым***. Множество F называется ***полным***, если его замыкание совпадает с множеством всех булевых функций.

Термин „полный базис“ как раз и означает, что любую булеву функцию можно записать аналитически, используя в качестве исходных функции базиса. Замыкание множества — совокупность всех функций, которые могут быть представлены формулами над этим множеств. Замкнутое множество F — такое множество, которое не может быть расширено добавлением функций, представимых формулами над F.

Замыкание может рассматриваться как операция на совокупности всевозможных множеств булевых функций. Такая операция обладает следующими свойствами:

1) $[\varnothing] = \varnothing$;
2) $[[X]] = [X]$;
3) $X \subset [X]$;
4) $[X] \cup [Y] \subset [X \cup Y]$.

Первое свойство носит формальный характер. Второе вытекает из конечности процедуры построения любой формулы: достаточно, следуя по дереву синтаксического анализа, последовательно заменять функции из $[X]$ их формулами над X. Третье и четвертое свойства очевидны.

Из четвертого свойства вытекает, что если $X \subset Y$, то $[X] \subset [Y]$. Действительно, включение $X \subset Y$ равносильно равенству $X \cup Y = Y$. Если $X \subset Y$, то в силу свойства 4 заключаем, что $[X] \cup [Y] \subset [X \subset Y] = [Y]$. Следовательно, $[X] \subset [Y]$.

Из указанных свойств замыкания вытекает следующее утверждение.

Теорема 1.3. Если F — полное множество булевых функций, каждая из которых представима формулой над множеством G, то и G — полное множество.

◄ Так как каждая функция из F есть формула над G, то $F \subset [G]$. Из свойств замыкания вытекает, что

$$[F] \subset [[G]] = [G].$$

Но поскольку $[F]$ совпадает с множеством всех булевых функций, то и $[G]$ совпадает с множеством всех булевых функций, т.е. G — полный базис. ►

1.3. ДНФ и КНФ

Стандартный базис. Элементарные формулы — литералы. Элементарная конъюнкция (дизъюнкция). Дизъюнктивная (конъюнктивная) нормальная форма и совершенная форма. Теорема: любая булева функция, отличная от 0 (от 1) представима в виде СДНФ (СКНФ). Полнота стандартного базиса. Примеры полных базисов: базис Жегалкина, штрих Шеффера, стрелка Пирса.

Стандартный базис — это набор из трех исходных операций булевой алгебры: сложения (объединения), умножения (пересечения) и отрицания.

Здесь мы будем называть ***литералом*** переменную x или ее отрицание $\overline{x}$ и обозначать $\hat{x}$. Булево пересечение нескольких литералов, определяемых различными переменными, т.е. выражение вида $X = \hat{x}_1\hat{x}_2\ldots\hat{x}_k$, называется ***элементарной конъюнкцией***. Требование, чтобы все переменные были различны обусловливается следующим. Если в конъюнкцию входит несколько одинаковых литералов, то в силу коммутативности, ассоциативности и идемпотентности конъюнкции можно, переходя к эквивалентной формуле, оставить лишь один литерал (например, $\overline{x}_1\overline{x}_1 = \overline{x}_1$). Если в конъюнкцию входит переменная и ее отрицание, то формула эквивалентна константе 0, поскольку $x\,\overline{x} = 0$ и для любой формулы Y имеем $Y\,x\,\overline{x} = 0$.

Дизъюнкция нескольких элементарных конъюнкций называется ***дизъюнктивной нормальной формой***, или ***ДНФ***. Например,

$$x_1\,x_3 + \overline{x}_2\,x_3\overline{x}_4 + \overline{x}_1\,x_2\,\overline{x}_3\,x_5.$$

Если состав переменных в каждой элементарной конъюнкции данной ДНФ один и тот же, то ДНФ называется ***совершенной***. Приведенный пример — это ДНФ, не являющаяся совершенной. Напротив, формула

$$x_1\,x_2\,x_3\,\overline{x}_4 + \overline{x}_1\,\overline{x}_2\,x_3\overline{x}_4 + \overline{x}_1\,x_2\,\overline{x}_3\,x_4$$

есть совершенная форма.

Поскольку в булевой алгебре сложение и умножение — симметричные операции и всегда можно интерпретировать сложение как умножение, а умножение как сложение, существует и двойственное понятие — ***конъюнктивная нормальная форма*** (***КНФ***), представляющая собой конъюнкцию элементарных дизъюнкций, и ***совершенная конъюнктивная форма*** (***СКНФ***). Из принципа двойственности для симметричных полуколец вытекает, что любому утверждению относительно ДНФ отвечает двойственное утверждение относительно КНФ, которое получается заменой сложения (дизъюнкции) умножением, умножения (конъюнкции) сложением, константы 0 константой 1, константы 1 константой 0, отношения порядка двойственным (обратным) порядком. Поэтому далее мы остановимся на изучении только ДНФ.

Теорема 1.4. Любая булева функция, отличная от константы 0 представима в виде СДНФ.

◀ Условимся под x^σ понимать формулу x, если $\sigma = 1$, и формулу $\overline{x}$, если $\sigma = 0$. Пусть функция $f(y_1, \ldots, y_n)$ принимает значение 1 на векторе $(t_1, \ldots, t_n)$ (такой вектор называют ***конституэнтой единицы***). Тогда элементарная конъюнкция $x_1^{t_1}x_2^{t_2}\ldots x_n^{t_n}$ также принимает значение 1 на этом наборе, но обращается в нуль на всех остальных n-мерных булевых векторах. Рассмотрим формулу

$$\Phi[x_1, x_2, \ldots, x_n] = \sum_{f(t_1,\ldots,t_n)=1} x_1^{t_1}x_2^{t_2}\ldots x_n^{t_n},$$

в которой сумма (объединение) распространяется на все те наборы $(t_1, \ldots, t_n)$ значений аргументов, на которых заданная функция принимает значение 1. Отметим, что множество таких наборов не пусто, так что в сумме есть по крайней мере одно слагаемое.

Нетрудно заметить, что формула Φ обращается в 1 при тех, и только при тех значениях переменных, при которых обращается в 1 рассматриваемая функция. Значит, формула Ψ представляет функцию f. ▶

Следствие 1.1. Стандартный базис является полным.

◀ Действительно, если функция не является константой 0, то она представима либо в виде СДНФ, которая является формулой над стандартным базисом. Константу 0 можно представить, например, формулой $f(x_1, x_2, \ldots, x_n) = x_1\overline{x}_1$. ▶

Пример 1.2. Рассмотрим функцию трех переменных $m(x_1, x_2, x_3)$ (табл. 1.4), называемую ***мажоритарной функцией***. Эта функция принимает значение 1, если больше половины ее аргументов имеют значение 1. Поэтому ее часто называют функцией голосования. Построим для нее СДНФ.

Таблица 1.4

x_1, x_2, x_3	$m(x)$
000	0
001	0
010	0
011	1
100	0
101	1
110	1
111	1

Мажоритарная функция имеет 4 конституэнты единицы, а значит, в ее СДНФ должно быть четыре слагаемых. Результат будет следующий:

$$m(x_1, x_2, x_3) = \overline{x}_1\, x_2\, x_3 + x_1\, \overline{x}_2\, x_3 + x_1\, x_2\, \overline{x}_3 + x_1\, x_2\, x_3$$

Аналогично строится СКНФ. Выбираем конституэнты нуля и для каждой составляем элементарную дизъюнкцию. Получим:

$$m(x_1, x_2, x_3) = (x_1 + x_2 + x_3)(x_1 + x_2 + \overline{x}_3)(x_1 + \overline{x}_2 + x_3)(\overline{x}_1 + x_2 + x_3). \ \#$$

Полнота стандартного базиса позволяет подбирать и другие полные системы функций. Полнота множества F может быть установлена из следующих соображений. Предположим, каждая из трех функций стандартного бузиса представима формулой над F. Тогда в силу теоремы 1.3 иножество F будет полным.

Пример 1.3. Множество из операций сложения по модулю 2, умножения и константы 1 называют ***базисом Жегалкина***. Сложение по модулю 2 и умножение — базовые операции кольца $\mathbb{Z}_2$, выражения, составленные с их помощью — это многочлены над кольцом $\mathbb{Z}_2$. Константа 1 в данном случае необходима для записи свободного члена. Поскольку $xx = x$, то все сомножители в многочлене имеют степень 1. Поэтому при записи многочлена можно обойтись без понятия степени. Примеры формул над базисом Жегалкина:

$$xy \oplus x \oplus y, \quad x \oplus 1, \quad xyz \oplus xz \oplus x \oplus y \oplus 1.$$

Любую такую формулу называют ***полиномом Жегалкина***. Фактически полином Жегалкина — это многочлен над кольцом $\mathbb{Z}_2$.

Нетрудно сконструировать формулы над базисом Жегалкина, представляющие операции сложения и отрицания стандартного базиса (умножение у двух базисов общее):

$$x + y = x \oplus y \oplus xy, \quad \overline{x} = x \oplus 1.$$

Поэтому базис Жегалкина — полное множество.

Можно показать, что для любой булевой функции полином Жегалкина определен однозначно (точнее, с точностью до порядка слагаемых). Коэффициенты полинома Жегалкина при небольшом количестве переменных можно найти методом неопределенных коэффициентов.

Пример 1.4. Рассмотрим множество из единственной функции — штриха Шеффера*. Это множество полно, что следует из следующих легко проверяемых тождеств:

$$\overline{x} = x \,|\, x, \quad xy = \overline{x \,|\, y} = (x \,|\, y) \,|\, (x \,|\, y), \quad x + y = \overline{x} \,|\, \overline{y} = (x \,|\, x) \,|\, (y \,|\, y).$$

Пример 1.5. Базис, состоящий из единственной функции — стрелки Пирса, также является полным. Проверка этого аналогична случаю штриха Шеффера. Впрочем, это заключение можно сделать и на основании принципа двойственности для симметричных полуколец.

*Штрих Шеффера — бинарная, но не ассоциативная операция. Поэтому при использовании инфиксной формы следует быть внимательным: результат зависит от порядка выполнения операций. В этом случае рекомендуется явно указывать порядок операций при помощи скобок, например писать $(x \,|\, y) \,|\, z$, а не $x \,|\, y \,|\, z$, хотя обе формы равнозначны.

1.4. Критерий Поста

Пять классов Поста T_0, T_1, S. M, L. Их замкнутость. Критерий Поста: множество полно, если оно не содержится ни в одном из классов Поста. Примеры.

Рассмотрим некоторые классы булевых функций:

• класс функций T_0, сохраняющих константу 0, т.е. функций, удовлетворящих условию $f(0,\ldots,0)=0$;

• аналогичный класс функций T_1, сохраняющих константу 1, т.е. удовлетворящих условию $f(1,\ldots,1)=1$;

• класс S самодвойственных функций, т.е. функций, удовлетворяющих условию $\forall\alpha\ \overline{f(\overline{\alpha})} = f(\alpha)$ (здесь α — n-мерный булев вектор, отрицание которого выполняется покомпонентно);

• класс M функций, монотонных относительно естественного порядка полукольца $\mathbb{B}^n$, т.е. функций, удовлетворяющих условию $\alpha\leqslant\beta\Rightarrow f(\alpha)\leqslant f(\beta)$;

• класс L линейных функций — функций, представимых полиномом Жегалкина первой степени (попросту говоря суммой по модулю 2 литералов, в число которых может входить константа 0).

Выделенные классы называют ***классами Поста***. Они интересны тем, что каждый определяется свойством, сохраняющимся при конструировании формул. Точнее говоря, если функции f, g_1, …, g_k принадлежат одному из указанных классов, то и их композиция $f(g_1,g_2,\ldots,g_k)$ принадлежит этому же классу. Это означает, что каждый из названных классов есть замкнутое множество. Отметим также, что классы Поста замкнуты относительно операций введения или удаления фиктивных аргументов.

Рассмотрение классов Поста приводит к следующему необходимому условию полноты рассматриваемого базиса: базис, целиком попадающий в один из классов Поста, не может быть полным. Действительно, если базис включен в один из классов Поста, то замыкание базиса также оказывается в этом классе. Но ни один из классов Поста не совпадает со всем множеством булевых функций. Следовательно, замыкание базиса не будет совпадать с множеством всех булевых функций. Менее очевидным является то, что это условие и достаточно.

Теорема 1.5 (критерий Поста). Множество F полно тогда и только тогда, когда оно не является подмножеством никакого из классов Поста.

◄ Доказательство достаточности критерия состоит в построении из заданных функций формул, представляющих функции стандартного базиса. Отметим, что в стандартном базисе одна из бинарных операций выражается через отрицание и другую бинарную операцию, например

$$x+y=\overline{\overline{x}\,\overline{y}}.$$

Поэтому достаточно построить формулы, определяющие отрицание и умножение.

Пусть F не является подмножеством ни одного из классов Поста. Для каждой функции $f\in F$ рассмотрим формулу $g(x)=f(x,x,\ldots,x)$. Поскольку F не содержится в T_0 и в T_1, в F, во-первых, есть непостоянные функции, а во-вторых есть хотя бы одна функция f_1, для которой $g_1(0)=1$, и есть хотя бы одна функция f_2, для которой $g_2(1)=0$. Это возможно, если одна из функций $g_1(x)$ и $g_2(x)$ есть отрицание, либо обе функции постоянны и представляют константы0 и 1. Рассмотрим оба случая.

Пусть функции $g_1(x)$ и $g_2(x)$ являются константы 0 и 1. Тогда для любой функции $f\in F$ формула $f(\alpha_1,\alpha_2,\ldots,\alpha_{i-1},x,\alpha_{i+1},\ldots,\alpha_n)$, в которой α_j — константы, есть формула над F. Выберем функцию f, не являющуюся монотонной. Тогда существуют два булевых вектора p и q, удовлетворяющие условиям $p<q$ и $f(p)=1$, $f(q)=0$. Векторы p и q можно соединить цепочкой $p=p_0,p_1,\ldots,p_k=q$ непосредственно предшествующих друг другу векторов (соседних). В этой цепочке найдется два соседних вектора p_{j-1} и p_j, которые отличаются только одной компонентой с некоторым номером i и на которых $f(p_{j-1})=1$, $f(p_j)=0$. Пусть α_j, $j\neq i$, одинаковые

компоненты этих векторов. Тогда формула $f(\alpha_1, \alpha_2, \ldots, \alpha_{i-1}, x, \alpha_{i+1}, \ldots, \alpha_n)$ представляет собой операцию отрицания.

Предположим, например, что функция $g_1(x)$ является отрицанием. Тогда мы можем составлять формулы вида $f(x \oplus \sigma^1, \ldots, x \oplus \sigma_n)$, где $x \oplus \sigma$ есть переменная x при $\sigma = 0$ и ее отрицание при $\sigma = 1$. Выберем функцию $f \in F$, не являющуюся самодвойственной. Тогда можно указать такой булев вектор $p \in \mathbb{B}^n$, что $f(\overline{p}) \neq \overline{f(p)}$, откуда $f(\overline{p}) = f(p)$. Пусть σ_1, ..., σ_n — компоненты вектора p. Рассмотрим функцию $g(x) = f(x \oplus \sigma^1, \ldots, x \oplus \sigma_n)$, определяемую выбранной несамодвойственной функцией f и вектором p. Так как $0 \oplus \sigma_i = \sigma_i$, $1 \oplus \sigma_i = \overline{\sigma}_i$, то $g(0) = f(p) = f(\overline{p}) = g(\overline{x})$. Следовательно, функция $g(x)$ является константой. Пусть, например, $g(x) = 1$. Тогда $\overline{g}(x) = 0$ — другая константа.

Итак, мы показали, что множество F, не являющееся подмножеством классов T_0, T_1, S, M, позволяет получить в виде формул отрицание и обе константы. Для построения формулы для произведения выберем функцию $f \in F$, не являющуюся линейной. Составляя формулы вида $f(X_1, X_2, \ldots, X_n)$, где X_i — это либо переменная x, либо переменная y, мы получим функции двух аргументов. Покажем, что среди таких функций есть нелинейные. В полиноме Жегалкина, представляющем функцию f, выберем нелинейное (степени два или выше) слагаемое наименьшей степени. Пусть это слагаемое имеет вид $x_{i_1} x_{i_2} \ldots x_{i_k}$. В формуле $f(X_1, X_2, \ldots, X_n)$ положим $X_{i_1} = x$, $X_{i_j} = y$, $j = \overline{2, k}$, а для остальных переменных, не вошедших в выбранное слагаемое выберем значение 0. Тогда выбранное слагаемое преобразуется в xy, остальные нелинейные слагаемые обнулятся, и мы получим функцию вида $g(x, y) = xy \oplus \alpha x \oplus \beta y \oplus \gamma$.

Так как

$$g(x, y) = xy \oplus \alpha x \oplus \beta y \oplus \gamma = (x \oplus \beta)(y \oplus \alpha) \oplus \alpha\beta \oplus \gamma = (x \oplus \beta)(y \oplus \alpha) \oplus \gamma',$$

заключаем, что $g(x \oplus \beta, y \oplus \alpha) \oplus \gamma' = xy$. Но $x \oplus \sigma$ — это переменная x при $\sigma = 0$ и ее отрицание $\overline{x}$ при $\sigma = 1$. Поэтому, если $g(x, y)$ принадлежит замыканию F, то и функция $g(x \oplus \beta, y \oplus \alpha) \oplus \gamma' = xy$, т.е. конъюнкция, принадлежит замыканию F.

Итак, мы показали, что если множество F не является подмножеством никакого класса Поста, то формулами над F можно представить отрицание и конъюнкцию, а значит, и дизъюнкцию. В этом случае, согласно теореме 1.3, множество F полное. ►

1.5. Минимизация ДНФ

Минимальная ДНФ — наименьшее число литералов (вхождений переменных). Кратчайшая ДНФ — наименьшее число элементарных конъюнкций. Геометрия булева куба. Импликанты и простые импликанты. Сокращенная ДНФ. Избыточные импликанты и тупиковые ДНФ. Методы построения сокращенной ДНФ. Алгоритм Квайна — Мак-Клоски. Склейка. Таблицы Квайна и простые импликанты. Ядровые импликанты. Тупиковые ДНФ и функция Патрика. Выбор кратчайших и минимальных ДНФ.

Каждая функция может быть представлена дизъюнктивной нормальной формой различными способами. например, изменить ДНФ можно, если в одно из ее слагаемых ввести фиктивную переменную с помощью сомножителя $x + \overline{x}$. Возникает задача из всех ДНФ найти наиболее простую в том или ином смысле.

Возможны по крайней мере два подхода к оценке сложности ДНФ: по количеству элементарных конъюнкций в ней (это количество называют ***длиной ДНФ***) и по общему количеству литералов, т.е. по сумме длин элементарных конъюнкций, входящих в ДНФ (эту сумму называют ***сложностью ДНФ***). ДНФ называют ***минимальной***, если она имеет наименьшую сложность, и ***кратчайшей***, если она имеет наименьшую длину.

В приведенной терминологии задача состоит в выборе среди всех ДНФ, представляющих данную функцию, наименьших и кратчайших. Решение такой задачи сводится к отбору некоторого ограниченного круга ДНФ, „подозрительных на минимум“, и последующему перебору отобранных ДНФ. Здесь возможны различные приемы. Чтобы описать и объяснить эти приемы, перейдем к геометрической интерпретации области определения булевой функции — булева куба.

Отдельные элементы, составляющие булев куб $\mathbb{B}^n$, т.е. булевы векторы принято называть ***вершинами куба***. Множество булевых векторов, у которых компоненты заданного набора, например с номерами $i_1, \ldots, i_k$, имеют определенные значения образуют подалгебру булевой алгебры, называемую ***гранью булева куба***. Грань представляет собой булев куб, но меньшей размерности, равной $n-k$, где n — размерность булева куба, k — количество фиксированных номеров. Грань размерности 1 называется ***ребром булева куба***. Кортеж номеров фиксированных компонент, определяющий грань куба называют ***направлением грани***. Грани, имеющие одинаковое направление, не пересекаются. Они называются ***параллельными***. Среди паралельных граней выделим ***соседние***, у которых совпадают значения всех фиксированных компонент, кроме одной.

Грани булева куба удобно обозначать, как слово из трех символов 0, 1 и $*$, где 0 и 1 обозначают фиксированные значения компонент, а $*$ указывает меняющиеся компоненты. Например, слово $00*1$, обозначает ребро четырехмерного булева куба, определяемое условиями $x_1 = 0$, $x_2 = 0$, $x_4 = 0$. Третья компонента, отмеченная звездочкой, варьируется. При таком способе записи размерность грани совпадает с количеством звездочек, грани параллельны, если у них совпадает положение звездочек (например $10*0$ и $01*1$). Грани соседние, если их записи различаются только в одной позиции, в которой для одной грани указано значение 0, а для другой 1 (например $01*0*1$ и $11*0*1$).

Будем говорить, что булева функция f, определенная на $\mathbb{B}^n$, подчиняется булевой функции g, определенной на том же кубе (или f меньше g), если $f(x) \leqslant g(x)$ для любого $x \in \mathbb{B}^n$. В этом случае будем писать $f \leqslant g$. Множество вершин куба, в которых функция f принимает значение 1, будем называть ***уровнем единицы функции***, сами эти вершины называют ***конституентами единицы***. Если $f \leqslant g$, то уровень единицы функции f является подмножеством уровня единицы функции g.

Элементарная конъюнкция принимает значение 1 в том и только в том случае, когда каждый литерал в ней принимает значение 1. Часть переменных может не входить в конъюнкцию, т.е. эти переменные будут фиктивными. Варьируя значения фиктивных переменных, получим грань n-мерного булева куба. Таким образом, уровень единицы любой элементарной конъюнкции есть некоторая грань булева куба.

Любую ДНФ данной функции f можно интерпретировать как набор граней булева куба, объединение которых совпадает с уровнем единицы функции f. При этом каждая элементарная конъюнкция рассматриваемой ДНФ будет подчинена функции f. Любую элементарную конъюнкцию, подчиненную функции f, будем называть ***импликантой***.

Замечание. Далее будем предполагать, что исследуемая функция от n аргументов не меняется, так что можно заранее каждый аргумент функции связать с некоторой переменной, а точнее, с i-м аргументом связывается переменная x_i. Это позволяет не различать булевы функции и формулы, составленные из переменных $x_1, \ldots, x_n$. Именно в этом контексте элементарная конъюнкция трактуется как функция от n аргументов.

Чтобы получить кратчайшую ДНФ заданной функции, надо выбрать минимальное количество импликант (граней булева куба, содержащихся в уровне единицы функции), в совокупности накрывающих уровень 1. Для минимальной ДНФ минимума достигает сумма длин импликант, а для этого надо стремиться увеличить сумму размерностей граней куба, соответствующих элементарным конъюнкциям ДНФ. В самом деле, пусть d_i и n_i, $i=\overline{1,\,k}$, — длины конъюнкций, составляющих ДНФ, и размерности соответствующих граней булева куба; s — сложность ДНФ; r — сумма размерностей граней, соответствующих конъюнкциям ДНФ. Тогда

$$s = \sum_{i=1}^{k} d_i = \sum_{i=1}^{k} (n - n_i) = kn - r,$$

и мы видим, что увеличение суммы размерностей граней ведет к уменьшению сложности ДНФ.

При небольших размерностях булева куба (до 4) его геометрию можно представить на плоскости с помощью так называемых ***карт Карно***. Представим куб $\mathbb{B}^n$ как декартово произведение двух кубов $\mathbb{B}^k \times \mathbb{B}^l$, где k, $l \leqslant 2$. Такое произведение можно изобразить таблицей, в которой каждая строка соответствует вершине куба $\mathbb{B}^k$, а каждый столбец — вершине куба $\mathbb{B}^l$. Существенным моментом является такой порядок строк и соответственно столбцов, при котором соседним строкам и столбцам отвечают соседние вершины соответствующего булева куба. А именно, при k, $l = 1$, берем естественный порядок 0, 1. при k, $l = 2$ используем порядок 00, 01, 11, 10 (табл. 1.5).

Таблица 1.5

	00	01	11	10
00				
01				
11				
10				

В такой таблице соседние клетки соответствуют соседним вершинам четырехмерного булева куба. Если представить, что эта „шахматная доска“ склеена по горизонтали и вертикали, т.е. соседними, например, являются (00, 11) и (10, 11), а также (01, 00) и (01, 10), то соседние на доске будет в точности соответствовать соседним на кубе.

На карте Карно ребро — это пара соседних (в том числе и „через край“) клеток. Легко изобразить на карте Карно и двумерные грани, состоящие из 4-х вершин. это либо квадрат из двух примыкающих пар клеток, либо одна строка, либо один столбец. Трехмерная грань — это две соседние строки или два соседних столбца.

Таблица 1.6

	000	001	011	010	110	111	101	100
00								
01								
11								
10								

К сожалению, при увеличении размерности такая простая интерпретация исчезает. Например, для пятимерного куба (табл. 1.6), соседние клетки (включая „соседство через край“) будут соответствовать соседним вершинам. Но, кроме того, и другие пары клеток, например (01, 001) и (01, 101), не будучи соседними в таблице, соответствуют соседним вершинам. Причина в том, что на карте Карно у каждой клетки четыре соседних, в то время как в n-мерном кубе каждая вершина имеет n соседних вершин, которые получаются, если в булевом векторе изменить одну компоненту из n возможных.

Рассмотрим два приема, с помощью которых из имеющейся ДНФ можно получить более простую ДНФ (и в смысле длины, и в смысле сложности).

Склейка. Дизъюнктивная нормальная форма данной функции может содержать пару импликант, соответствующих соседним граням булева куба. Но две соседние грани булева куба вместе составляют грань большей размерности. Например, две двумерные грани 01∗∗10 и 01∗∗11 пятимерного куба в совокупности составляют трехмерную грань 01∗∗1∗. Замена в ДНФ двух таких импликант одной более короткой приводит к уменьшению и длины, и сложности ДНФ. Такую замену называют ***склейкой***. Импликанты, соответствующие соседним граням, имеют один и тот же набор переменных, причем они различаются только по одной переменной: в одной импликанте стоит сама переменная, а в другой — ее отрицание. Например, грани 01∗∗10 и 01∗∗11 соответствуют элементарным конъюнкциям $\overline{x}_1 x_2 x_4 \overline{x}_5$ и $\overline{x}_1 x_2 x_4 x_5$. Изменение ДНФ выполняется в данном случае в соответствии с тождеством $\overline{x}_1 x_2 x_4 \overline{x}_5 + \overline{x}_1 x_2 x_4 x_5 = \overline{x}_1 x_2 x_4$. Примеры возможных склеек показаны с помощью карты Карно на рис. 1.3.

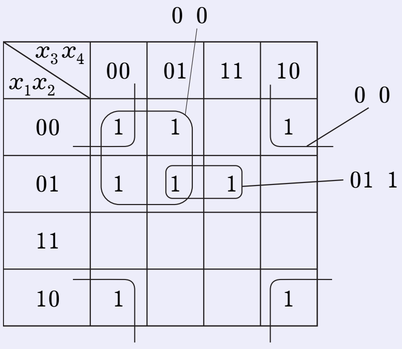

Рис. 1.3

Отталкиваясь от совершенной ДНФ, мы можем сперва склеивать различные пары соседних нульмерных граней (вершин) в ребра, затем в соседние ребра в двумерные грани и т.д. Процесс подобной склейки приведет к выявлению граней максимальной размерности, содержащихся в уровне единицы функции. Этим граням соответствуют максимальные импликанты функции (максимальные относительно введенного порядка $\leqslant$ на импликантах функции). Такие импликанты называются ***простыми импликантами***. ДНФ данной функции, в которую входят все простые импликанты, называют ***сокращенной ДНФ***.

Отметим, что если исходная ДНФ не является совершенной, то склейка может и не привести к сокращенной ДНФ. Так, если на рис. 1.3 выбрать грани $*0*0$, $01*1$, 0001, 0100, получим покрытие уровня единицы функции. В соответствующей ДНФ $\overline{x}_2\overline{x}_4+\overline{x}_1x_2x_4+\overline{x}_1\overline{x}_2\overline{x}_3x_4+\overline{x}_1x_2\overline{x}_3\overline{x}_4$ нельзя склеить какие-либо грани, но эта ДНФ не является сокращенной, так как два последних слагаемых не являются простыми импликантами.

Избыточные импликанты. В сокращенной ДНФ одна из импликант может накрываться остальными. Такую импликанту можно удалить из формулы, уменьшая и длину, и сложность ДНФ. Рассмотрим, например, функцию, представленную картой Карно на рис. 1.4. У этой функции максимальными являются шесть импликант $1*0$, $10*$, $*01$, $0*1$, $01*$, $*10$. Их сумма даст сокращенную ДНФ. Но из рисунка видно, что, например, импликанта $10*$ накрывается двумя импликантами $1*0$ и $*01$. Значит, эту импликанту можно из ДНФ удалить.

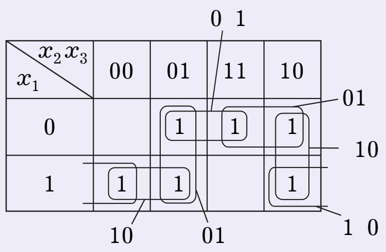

Рис. 1.4

Импликанта в ДНФ, подчиненная сумме остальных импликант этой ДНФ, называется ***избыточной***. Если из сокращенной ДНФ удалить избыточные импликанты, то получим ***тупиковую ДНФ***. По определению, тупиковая ДНФ — это ДНФ, состоящая из простых импликант, ни одна из которых в этой ДНФ не является избыточной. Например, на рис. 1.4 тупиковую ДНФ образуют импликанты $1*0$, $*01$, $01*$.

Основной целью каждого метода минимизации ДНФ является выявление всех тупиковых ДНФ. Именно среди них находится минимальная и кратчайшая формы.

Построение сокращенной ДНФ. Один из методов построения сокращенной ДНФ — это ***метод Квайна — МакКлоски***. Его суть в следующем. В качестве исходной выбирается совершенная ДНФ. На первом этапе проводится склейка импликант совершенной ДНФ в ребра, которые добавляются в ДНФ. После того как проведены все склейки вершин, из ДНФ удаляются все избыточные вершины, т.е. те, которые подчиняются добавленным ребрам. На втором этапе проводится склейка всех ребер в четырехмерные грани, которые добавляются к ДНФ. После проведения всех склеек между ребрами удаляются ребра, подчиненные двумерным граням. На третьем этапе выполняется склейка двумерных граней и удаление избыточных двумерных граней. Процесс останавливается, когда между гранями соответствующей размерности не удается провести ни одной склейки. Результатом этих преобразований и будет сокращенная ДНФ.

Пример 1.6. Рассмотрим функцию, заданную картой Карно на рис. 1.5. Первый этап склейки по методу Квайна приведет к ребрам $1*00$, $10*0$, $*010$, а также четырем ребрам двумерной грани $0**1$ и четырем ребрам двумерной грани $0*1*$. Все исходные импликанты будут удалены как избыточные.

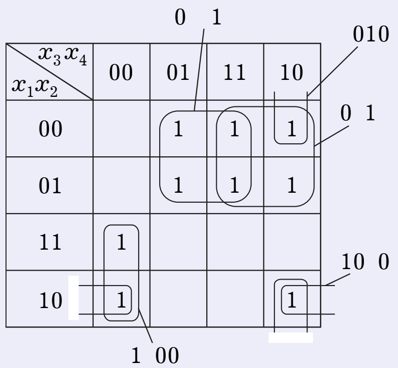

Рис. 1.5

На втором этапе мы получим две двумерные грани $0**1$ и $0*1*$. При этом будут удалены 6 ребер, входящих в эти грани. Поскольку склеить две эти грани нельзя, процесс выявления простых импликант на этом завершится. Сокращенная ДНФ будет содержать пять импликант, соответствующих граням $0**1$, $0*1*$, $1*00$, $10*0$, $*010$. #

Другой метод построения сокращенной ДНФ — ***метод Блейка***, в котором в качестве исходной можно взять любую ДНФ. В этом методе на первом этапе многократно выполняется операция обобщенного склеивания, при которой сумма

$xK_1 + \overline{x}K_2$ заменяется суммой $xK_1 + \overline{x}K_2 + K_1K_2$. Такая операция выполняется до тех пор, пока удается получать новые импликанты вида K_1K_2. После завершения первого этапа выполняется операция поглощения согласно формуле $K_1K_2 + K_2 = K_2$.

Выявление всех тупиковых ДНФ. Создать список всех тупиковых ДНФ можно, используя следующий прием. Обозначим все простые импликанты символами K_1, K_2, ..., K_m. Перенумеруем все вершины уровня единицы функции. Пусть первая вершина накрывается склейками K_{i_1}, K_{i_2}, ..., K_{i_p}. Тогда элементарная дизъюнкция $K_{i_1} + K_{i_2} + \ldots + K_{i_p}$ задает условие, что первая вершина уровня единицы функции накрывается одной из простых импликант. Составив конъюнкцию из указанных элементарных дизъюнкций по всем вершинам уровня единицы функции, мы получим КНФ, представляющее собой логическое условие того, что весь уровень единицы функции накрыт простыми импликантами. Эта КНФ называют ***функцией Патрика***. Преобразуем КНФ в ДНФ, пользуясь свойствами булевых операций (дистрибутивностью, коммутативностью, идемпотентностью). В ДНФ каждая элементарная конъюнкция будет описывать набор простых импликант, целиком покрывающих уровень единицы функции, т.е. набор, описывающий конкретную тупиковую ДНФ рассматриваемой функции. Можно показать, что таким образом можно выявить все тупиковые ДНФ.

Среди простых импликант рассматриваемой функции есть те, которые входят в любую тупиковую ДНФ. К таким относится любая импликанта, для которой среди накрываемых вершин есть хотя бы одна, не накрываемая никакой другой импликантой. Например, на рис. 1.5 вершина 1100 накрывается единственной простой импликантой 1∗00. Эта импликанта должна входить в любую тупиковую ДНФ. Импликанты, удовлетворяющие описанному условию, называются ***ядровыми импликантами***. Совокупность ядровых импликант — ***ядро*** — постоянная часть всех тупиковых ДНФ. Например, на рис. 1.5 ядро составляют импликанты 0∗∗1, 0∗1∗, 1∗00, а импликанты ∗010 и 10∗0 неядровые. При выявлении тупиковых ДНФ ядровые импликанты и накрываемые ими вершины из функции Патрика можно исключить.

Пример 1.7. У функции, представленной на рис. 1.5 диаграммой Карно, имеется пять простых импликант. Введем обозначения:

$$K_1 = 0{*}{*}1, \quad K_2 = 0{*}1{*}, \quad K_3 = 1{*}00, \quad K_4 = 10{*}0, \quad K_5 = {*}010.$$

Тогда функция Патрика для 9 вершин уровня 1 при их нумерации по строкам будет иметь следующий вид:

$$P = K_1(K_1 + K_2)(K_2 + K_5)K_1(K_1 + K_2)K_2K_3(K_3 + K_4)(K_4 + K_5).$$

Упростив согласно идемпотентности операций и тождествам поглощения ($x(x + y) = x$), получим

$$P = K_1K_2K_3(K_4 + K_5).$$

Используя дистрибутивность и идемпотентность, приходим к ДНФ:

$$P = K_1K_2K_3(K_4 + K_5) = K_1K_2K_3K_4 + K_1K_2K_3K_5.$$

Таким образом, рассматриваемая функция имеет две сокращенные ДНФ, описываемые двумя списками склеек K_1, K_2, K_3, K_4 и K_1, K_2, K_3, K_5. В результате получим

$$D_1 = \overline{x}_1x_4 + \overline{x}_1x_3 + x_1\overline{x}_3\overline{x}_4 + x_1\overline{x}_2\overline{x}_4, \qquad D_2 = \overline{x}_1x_4 + \overline{x}_1x_3 + x_1\overline{x}_3\overline{x}_4 + \overline{x}_2x_3\overline{x}_4.$$

Видно, что у обоих сокращенных ДНФ есть общая часть — первые три конъюнкции. Это ядро, входящее в каждую сокращенную ДНФ. Его можно исключить из рассмотрения, рассматривая условия накрытия только для тех вершин, которые не накрываются склейками ядра. В данном случае это единственная вершина 1010. При этом функция Патрика будет иметь вид

$\hat{P} = K_4 + K_5$. Добавив к каждому слагаемому элементарную конъюнкцию $K_1K_2K_3$, описывающую ядро, получим полную функцию Патрика. #

Ядро данной функции можно получить с помощью ***таблицы Квайна***, в которой каждая строка соответствует одной простой импликанте, а каждый столбец — одной вершине уровня единицы функции. В каждой клетке указывается 1, если импликанта накрывает вершину, и пробел в противном случае Например, составим таблицу Квайна функции, представленной картой Карно на рис. 1.3 (табл. 1.7). Чтобы определить ядро функции, необходимо выявить столбцы, в которых находится только одна единица, и пометить соответствующие строки. Импликанты, соответствующие помеченным строкам, образуют ядро. Например, в табл. 1.7 по одной единице содержится в столбцах, отвечающих вершинам 0001, 0010, 0100, 0111, 1000, 1010 (эти единицы выделены полужирным шрифтом). Поскольку в каждой строке содержится хотя бы одна выделенная единица, все простые импликанты ядровые, т.е. рассматриваемая функция имеет единственную тупиковую ДНФ.

Таблица 1.7

	0000	0001	0010	0100	0101	0111	1000	1010
$0{*}0{*}$	1	**1**		**1**	1			
${*}0{*}0$	1		**1**				**1**	**1**
$01{*}1$					1	**1**		

ОГЛАВЛЕНИЕ