

ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

УДК 519.7

А.Д. Закревский

ПРОГРАММИРОВАНИЕ ВЫЧИСЛЕНИЙ В МНОГОМЕРНОМ БУЛЕВОМ ПРОСТРАНСТВЕ¹

Предлагается набор макроопераций POBS над булевыми 2^n -компонентными векторами, существенно облегчающий программирование вычислений в многомерном булевом пространстве. Применение набора POBS иллюстрируется на примерах анализа частичных булевых функций на монотонность и наличие функциональных закономерностей, решения задач последовательной композиции и декомпозиции. Важную роль в нем играют операции взаимодействия соседних элементов пространства.

Ключевые слова: *булево пространство, комбинаторные задачи, набор макроопераций, операции над соседними элементами.*

Булевым пространством размерности n принято называть множество, составленное из 2^n булевых n -компонентных векторов, на котором определено отношение соседства – два вектора являются соседними, если они различаются значениями ровно в одной компоненте. Его можно представить графом, вершины которого соответствуют элементам булева пространства, а ребра соединяют вершины, соответствующие соседним элементам. Такой граф принято называть n -мерным кубом, и он широко используется в учебной литературе, при описании методов минимизации булевых функций и решения других задач логического проектирования. Однако уже при $n > 5$ изображение графа становится слишком громоздким и неудобным для практического использования, в чем можно убедиться, глядя на рис. 1.

Более приемлемым с точки зрения программиста является представление булева пространства размерности n в виде булева 2^n -вектора, т. е. вектора с 2^n компонентами, соответствующими элементам пространства и пронумерованными стандартным образом, начиная с нуля: компоненте с номером k соответствует элемент пространства, представляющий собой n -компонентный булев вектор, задающий двоичный код номера k . Приписывая этим компонентам значения из множества $\{0, 1\}$, можно задать любую булеву функцию от n переменных. Например, булев вектор

$f = 10000000\ 00000000\ 00001000\ 00000001\ 10000000\ 00000000\ 00110000\ 00000001$ определяет некоторую булеву функцию f шести переменных $x_1, x_2, x_3, x_4, x_5, x_6$, принимающую значение 1 на следующих наборах их значений:

000000, 010100, 011111, 100000, 110010, 1100011, 111111.

¹ Работа была частично поддержана Белорусским республиканским фондом фундаментальных исследований (Проект Ф07МС-034).

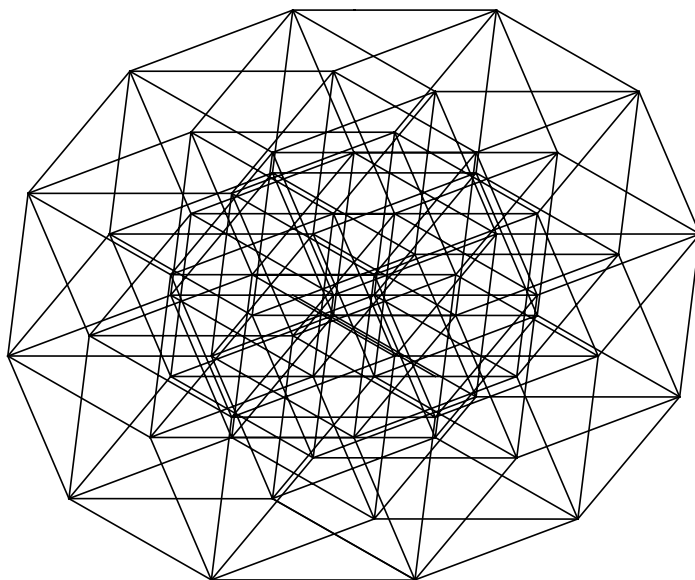


Рис. 1. Шестимерный куб

Для удобства визуального восприятия вектор f разбит на восемь фрагментов, соответствующих интервалам булева пространства с внутренними переменными x_4, x_5, x_6 и представляющих коэффициенты дизъюнктивного шенноновского разложения функции f по переменным x_1, x_2, x_3 .

Булевы 2^n -векторы служат основными объектами преобразований, проводимых при решении разнообразных логико-комбинаторных задач, которые возникают при проектировании дискретных устройств и разработке систем искусственного интеллекта [1]. С целью повышения эффективности их программирования в данной статье предлагается базисный набор макроопераций над такими векторами, названный POBS (Parallel Operations in Boolean Space).

Как показывает опыт, этот набор оказывается весьма эффективным, позволяя быстро оперировать на современной персоналке с длинными булевыми векторами, представляющими произвольные булевы функции многих переменных, до 27 переменных включительно. Заметим, что для однострочной распечатки такого вектора (содержащего $2^{27} = 134\,217\,728$ символов) потребовалась бы бумажная лента длиной свыше 250 километров.

1. Покомпонентные операции над булевыми векторами

Простейшими операциями набора POBS служат *покомпонентные операции*: операция инвертирования над одним вектором и произвольные двухместные булевы операции над парой одномерных векторов. Проиллюстрируем их следующими примерами.

Обозначим через g и h булевы векторы, представляющие булевы функции $g(x)$ и $h(x)$, где $x = (x_1, x_2, x_3, x_4, x_5)$. Пусть

$$g = 10001100\ 00100010\ 11100001\ 00101010,$$

$$h = 00110010\ 10001111\ 01100011\ 01100011.$$

Тогда

$$\begin{aligned}\bar{g} &= 01110011\ 11011101\ 00011110\ 11010101, \\ g \vee h &= 10111110\ 10101111\ 11100011\ 01101011, \\ g \wedge h &= 00000000\ 00000010\ 01100001\ 00100010, \\ g \oplus h &= 10111110\ 10101101\ 10000010\ 01001001 \quad \text{и т. д.}\end{aligned}$$

Более сложной является операция *перестановки аргументов* булевой функции, представленной булевым 2^n -вектором. Она определяется перестановкой на множестве номеров переменных и приводит к соответствующей перестановке компонент вектора.

Например, в результате перестановки номеров (4, 2, 1, 3) переменные множества $x = (x_1, x_2, x_3, x_4)$ перегруппируются в новую последовательность (x_4, x_2, x_1, x_3) . Это приводит к соответствующей перестановке компонент вектора f , представляющего булеву функцию $f(x)$. Компонента f_k помещается на месте с номером i , если двоичный код k номера k представляет собой результат произведения матрицы перестановок P на двоичный код i номера i . Другими словами, вектор k равен покомпонентной дизъюнкции столбцов матрицы P , отмеченных единицами в векторе i . Проиллюстрируем это на примере замещения компоненты f_{13} компонентой f_{14} :

	P	i	k
1	0 0 0 1	1	1
2	0 1 0 0	1	1
3	1 0 0 0	0	1
4	0 0 1 0	1	0
	1 2 3 4	13	14

В результате перестановка (4, 2, 1, 3) на множестве компонент вектора x приводит к следующей перестановке на множестве компонент вектора f :

$$(0, 8, 1, 9, 4, 12, 5, 13, 2, 10, 3, 11, 6, 14, 7, 15).$$

Пусть $f = 0111\ 1010\ 0100\ 1001$. Тогда рассматриваемая перестановка на множестве компонент этого вектора приводит к его значению f^* , соответствующему новому порядку (x_4, x_2, x_1, x_3) перечисления аргументов:

$$f^* = 0100\ 1110\ 1110\ 0001.$$

2. Операции над соседними элементами в булевом пространстве

В набор POBS входят также операции взаимодействия различных компонент в пределах одного булева 2^n -вектора f , задающего функцию $f(x) = f(x_1, x_2, \dots, x_n)$. Главную роль играют при этом *операции над соседними элементами булева пространства*. Соседними называются булевы n -векторы, различающиеся значениями ровно в одной компоненте. При представлении булева пространства в виде многомерного булева куба они отображаются вершинами, соединенными ребрами.

Использование операций над соседними элементами позволяет параллелить логические вычисления и тем самым ускорять их. Эта идея не нова. Так, в 1961 – 1962 гг. в Сибирском физико-техническом институте разрабатывалась специальная приставка к универсальной вычислительной машине, существенно ускоряющая процесс решения задач логического проектирования и названная L-машиной

[2, 3]. В ее основу положена идея распределенного в пространстве выполнения логических операций на серии информационных полей, играющих роль своеобразных регистров. Они структурно подобны десятимерному булевому кубу и позволяют непосредственно представлять булевы функции десяти переменных и выполнять покомпонентные операции над ними. Одно из полей называется основным, и в нем могут совершаться преобразования в пределах одной функции. Схемная реализация основного поля обеспечивает одновременное выполнение любой двухместной булевой операции в пределах каждой из 512 пар элементов поля, соседних по некоторой выбранной переменной.

Аналогичная идея была положена в основу коммутационной машины, предложенной У.Д. Хиллисом в 1978 г. и разработанной компанией Thinking Machine Corporation в 1985 г. Эта машина обеспечивает информационную связь в многопроцессорном компьютере, компоненты которого непосредственно связаны между собой аналогично вершинам n -мерного булева куба. Передача информации между любыми процессорами в таком компьютере производится не более чем за n тактов. На базе этой разработки была создана серия коммутационных машин, которые используются исследователями, работающими в области искусственного интеллекта, для решения задач логического вывода [4].

При значениях n , превышающем 5, вектор f удобно задавать булевой матрицей размером $2^5 \times 2^{n-5}$, представляя ее 32-элементные строки словами в памяти машины (что естественно для большинства современных компьютеров). В этом случае любые два соседние по переменной x_k элемента пространства M находятся в пределах одного слова, если $k < 6$, и принадлежат различным словам в противном случае, что должно учитываться при программировании. Заметим, что при иллюстрации приводимых ниже примеров используются более компактные матрицы размером $2^4 \times 2^{n-4}$.

Введем следующие простейшие операции преобразования булевой функции $f(x) = f(x_1, x_2, \dots, x_n)$, заданной вектором f , путем взаимодействия соседних элементов:

- $f - k$ – присвоение значения 0 аргументу x_k (получение функции $f(x_k = 0)$);
- $f + k$ – присвоение значения 1 аргументу x_k (получение функции $f(x_k = 1)$).

Продemonстрируем их выполнение на следующих матрицах, в которых показано разбиение множества элементов на две части, соответствующие различным значениям выбранной переменной x_k и называемые ниже условно левой (отмеченной полужирным шрифтом) и правой:

f	$f-4$	$f+4$	
-----	-----	-----	
-----	-----	-----	
-----	-----	-----	
-----	-----	-----	
0110110101011110	0110011001010101	1101110111101110	
0010010000010110	0010001000010001	0100010001100110	
1100101001110001	1100110001110111	1010101000010001	
0100010111010011	0100010011011101	0101010100110011	
0110110101011110	0110110101011110	1100101001110001	
0010010000010110	0010010000010110	0100010111010011	
1100101001110001	0110110101011110	1100101001110001	
0100010111010011	0010010000010110	0100010111010011	
1 2 f	$f-1$	$f+1$	3 4 5 6

3. Операции преобразования размерности булевых векторов

Такие операции позволяют реализовать взаимодействие булевых векторов различной размерности.

Рассмотрим два булевых вектора:

n -вектор $\mathbf{u}$ с k единицами, отмечающими некоторые k переменных из множества $\mathbf{x} = (x_1, x_2, \dots, x_n)$,

2^k -вектор $\mathbf{h}$, задающий булеву функцию h отмеченных переменных.

Введем в набор POBS операцию $\mathbf{h} \times \mathbf{u}$ переноса функции h в фрагмент пространства n переменных, соответствующий конъюнкции инверсий переменных, не отмеченных в $\mathbf{u}$. Все элементы остальных фрагментов получают при этом значение 0.

Пусть, например, $n = 5$, $\mathbf{u} = 01101$ и $\mathbf{h} = 10010011$.

Тогда строится 2^5 -вектор, в котором выделяется фрагмент, соответствующий конъюнкции $\bar{x}_1 \bar{x}_4$:

00000000 00000000 00000000 00000000,

и в последний вписывается вектор $\mathbf{h}$. В результате получаем:

$\mathbf{h} \times \mathbf{u} = 10000100 00001100 00000000 00000000.$

По аналогии вводится операция $\mathbf{f} : \mathbf{u}$, осуществляющая обратный перенос информации из упомянутого фрагмента в вектор $\mathbf{h}$, задающий получаемую при этом булеву функцию h от k переменных. Так, если $\mathbf{u} = 01010$ и

$\mathbf{f} = 00110010 11100000 11100110 00011101,$

то находится фрагмент, соответствующий конъюнкции $\bar{x}_1 \bar{x}_3 \bar{x}_5$:

00110010 11100000 11100110 00011101,

и содержащаяся в нем информации используется для построения искомого вектора:

$\mathbf{h} = \mathbf{f} : \mathbf{u} = 0111.$

Если известно, что представленная вектором $\mathbf{f}$ функция f зависит лишь от переменных множества $\mathbf{u}$ (остальные оказываются фиктивными), то посредством операции $\mathbf{f} : \mathbf{u}$ они удаляются из функции и результат представляется вектором $\mathbf{h}$.

4. Операции над частичными булевыми функциями

Перейдем к рассмотрению частичных булевых функций, широко используемых при решении задач логического проектирования.

Произвольную частичную булеву функцию f от n переменных обычно представляют соответствующим 2^n -компонентным троичным вектором $\mathbf{f}^-$, однако при разработке программ удобнее задавать ее парой булевых векторов $\mathbf{f}^1$ и $\mathbf{f}^0$, также 2^n -компонентных. При этом единицами в векторе $\mathbf{f}^1$ отмечаются компоненты, на которых функция f принимает значение 1, а в векторе $\mathbf{f}^0$ – компоненты, на которых функция равна нулю.

Пусть, например,

$\mathbf{f}^- = 1-001010 011--01-1.$

Тогда

$$\begin{aligned} f^1 &= 10001010 \ 011000101, \\ f^0 &= 00110101 \ 100001000. \end{aligned}$$

В этом случае операция присвоения значения 0 аргументу x_k будет представляться парой операций $f^1 - k$, $f^0 - k$, а значения 1 – парой $f^1 + k$, $f^0 + k$.

Подобно вектору f , используемому во введенных выше операциях, вектор u также может оказаться троичным, например, представляя некоторую элементарную конъюнкцию, и тогда его также целесообразно заменить на пару булевых векторов u^1 и u^0 , в данном случае n -компонентных. Например, в операции дизъюнктивного разложения частичной булевой функции f по всем переменным объединенного множества $u^1 \cup u^0$ получение коэффициента при представленной таким образом конъюнкции осуществляется серией операций

$$f^1 - u^0, f^0 - u^0, f^1 + u^1, f^0 + u^1.$$

5. Программирование в базисе POBS

Продemonстрируем технику программирования в базисе POBS на примерах решения некоторых задач теории булевых функций.

1. Проверка частичной булевой функции на монотонность

Рассмотрим частичную булеву функцию $f(x) = (x_1, x_2, \dots, x_n)$, заданную двумя множествами наборов значений аргументов: множеством M^1 , на котором она принимает значение 1, и множеством M^0 , где она принимает значение 0. Назовем ее *монотонной*, а точнее, *положительной* функцией, если для любой пары наборов $p \in M^1$ и $q \in M^0$ выполняется условие $p > q$ (вектор p больше вектора q), т.е. для любой пары (p_i, q_i) одноименных компонент $p_i \geq q_i$ и хотя бы для одной пары $p_i > q_i$.

Очевиден простой метод проверки функции на монотонность, основанный на переборе всех пар (p, q) и проверке для каждой из них условия $p > q$. Однако с увеличением числа переменных n и соответственным ростом мощностей множеств M^1 и M^0 такой метод оказывается слишком трудоемким. Более эффективен предлагаемый ниже метод, использующий операции из набора POBS.

Зададим функцию $f(x)$ двумя булевыми 2^n -векторами: вектором f^1 и f^0 . Элементы множества M^1 представлены единицами в векторе f^1 . Обозначим через M^* множество таких элементов булева пространства, каждый из которых больше некоторого элемента из множества M^0 , и представим это множество вектором f^* .

Утверждение 1. Функция $f(x)$ монотонна, если и только если $f^* f^0 = 0$.

Вектор f^* можно найти с помощью введенных выше операций набора POBS, последовательностью n шагов. Сначала получим вектор $f^2 = (f^1 - 1) \vee f^1$, который представит множество M^1 , дополненное элементами булева пространства, соседними «сверху» к некоторым элементам из M^1 по переменной x_1 . Затем полученное множество расширим аналогичным образом по следующей переменной x_2 : $f^3 = (f^2 - 1) \vee f^2$. Затем, повторив эту операцию по всем остальным переменным, получим искомым вектор $f^{n+1} = f^*$.

Пусть, например, $n = 5$:

$$\begin{aligned} f^1 &= 00010000 \ 00100000 \ 00000001 \ 00001010, \\ f^0 &= 11000010 \ 00000100 \ 10100000 \ 10000000. \end{aligned}$$

В этом случае процесс последовательного расширения множества M^1 , приводящий к получению вектора f^* , представляющего множество M^* , отобразится следующей последовательностью векторов, получаемых на очередных шагах

$$\begin{aligned} f^2 &= 00010000\ 00100000\ 00010001\ 00101010, \\ f^3 &= 00010000\ 00110000\ 00010001\ 00111011, \\ f^4 &= 00010001\ 00110011\ 00010001\ 00111011, \\ f^5 &= 00010001\ 00110011\ 00010001\ 00111011, \\ f^6 &= 00010001\ 00110011\ 00010001\ 00111111 = f^*. \end{aligned}$$

Покомпонентная конъюнкция полученного вектора с вектором

$$11000010\ 00000100\ 10100000\ 10000000 \quad f^0$$

равна нулю ($f^* \wedge f^0 = \mathbf{0}$), следовательно, рассматриваемая булева функция монотонна.

2. Поиск функциональных закономерностей

В современных информационных технологиях важную роль играют процедуры извлечения знаний из потока данных, т. е. поиска закономерностей, позволяющих находить правильные решения при решении интеллектуальных задач [5]. Ниже рассматривается частный, но важный случай закономерностей, а именно функциональные закономерности, хорошо зарекомендовавшие себя в естествознании.

В работе [6] была решена следующая задача. Допустим, что задано множество объектов, представленных различными комбинациями значений n двоичных признаков (признак есть – признака нет). Спрашивается, можно ли всегда однозначно определить значение некоторого выделенного признака, если известны значения остальных? А если можно, то как это сделать?

Представим исходную информацию совокупностью R некоторых элементов в n -мерном булевом пространстве $M = \{0, 1\}^n$ над множеством признаков. Эти элементы задают известные объекты, и их можно рассматривать как корни некоторого булева уравнения $F = 1$ с переменными из множества $x = (x_1, x_2, \dots, x_n)$.

Говорят, что уравнение $F = 1$ разрешимо относительно некоторой переменной, если ее можно представить булевой функцией от остальных переменных уравнения, определенной на множестве R [5]. Рассмотрим задачу выявления таких переменных в уравнении $F = 1$ и построения соответствующих функций.

Утверждение 2. Необходимым и достаточным условием разрешимости уравнения $F = 1$ относительно переменной x_i является отсутствие в множестве R пар наборов, соседних по переменной x_i .

Доказательство от противного (по правилу *modus tollens*): если найдется такая пара, то переменная x_i принимает в ней различные значения на одинаковых наборах значений остальных переменных, что противоречит определению функционального отношения.

Обозначим через $f(x)$ характеристическую булеву функцию множества R , положив, что $f(q_j) = 1$, если $q_j \in R$ и $f(q_j) = 0$, если $q_j \notin R$. Через $f(x_i = 0)$ и $f(x_i = 1)$ обозначим результат замены в функции $f(x)$ переменной x_i на константу 0 или 1 соответственно.

Утверждение 3. Уравнение $F = 1$ разрешимо относительно переменной x_i , если и только если $f(x_i = 0) \wedge f(x_i = 1) = 0$.

Это утверждение позволяет применить для проверки разрешимости уравнения по переменной x_i введенные выше векторные операции $f - i$ и $f + i$. Переформулируя утверждение в терминах этих операций, можно утверждать, что необходимым и достаточным условием разрешимости уравнения $F = 1$ относительно переменной x_i является выполнение отношения

$$(f - i) \wedge (f + i) = 0.$$

При выполнении этого условия возникает задача нахождения соответствующей булевой функции, которая в общем случае оказывается частичной, и ее оптимального доопределения. Оптимизация может заключаться как в минимизации числа аргументов функции, так и в упрощении ее алгебраического представления, например в дизъюнктивной нормальной форме (ДНФ).

Рассмотрим первую из этих задач. Она аналогична задаче минимизации безусловного диагностического теста и может решаться тем же методом. Некоторый аргумент x_k может быть определен как фиктивный, если после его удаления уравнение остается разрешимым по переменной x_i . Операцию удаления аргумента x_k можно представить как расширение множества R по этой переменной, т. е. как следующее преобразование его характеристической функции f

$$f := f(x_k = 0) \vee f(x_k = 1).$$

В терминах введенных выше векторных операций оно определяется как $Sf \vee k$, откуда следует

Утверждение 4. Аргумент x_k можно удалить из числа аргументов переменной x_i как функции остальных переменных, если и только если

$$((Sf \vee k) - i) \wedge ((Sf \vee k) + i) = 0.$$

3. Последовательная композиция булевых функций

Рассмотрим следующую задачу. На множестве аргументов $x = (x_1, x_2, \dots, x_n)$ булевыми n -векторами u, w и v задано разбиение на три непересекающиеся подмножества u, w и v : $x = u \cup w \cup v$. Заданы также две булевы функции $h(u, w)$ и $g(x, w, v)$, представленные соответствующими булевыми векторами h и g . Требуется вычислить их композицию при условии $x = h(u, w)$ и представить полученную при этом булеву функцию $f(x)$ 2^n -вектором f .

Такая композиция, называемая нераздельной последовательной двухблочной, иллюстрируется примером на рис. 2, где $n = 6$ и множества $u = (x_1, x_2)$, $w = (x_3, x_4)$ и $v = (x_5, x_6)$ заданы шестимерными булевыми векторами $u = 110000$, $w = 001100$ и $v = 000011$.

Положим, что функции $h(u, w)$ и $g(x, w, v)$ заданы соответствующими векторами:

$$\begin{array}{l} h = 1101001001101100, \\ g = 0011010011001001 \quad 1010010110101011. \\ \quad \quad \quad g_0 \quad \quad \quad g_1 \end{array}$$

Для удобства последующих рассуждений вектор g разбит на две половинки, задающие значения функции $g(x, w, v)$ при значениях 0 и 1 двоичной переменной x .

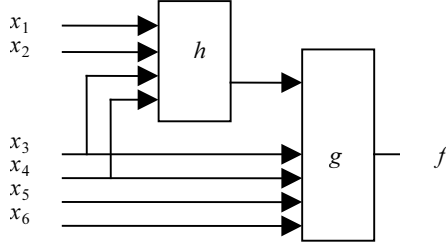
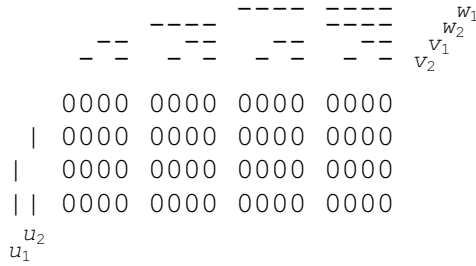


Рис. 2. Пример неразделимой последовательной двухблочной композиции

Представим булево пространство переменных $x = (u, w, v)$ в следующем виде:



Затем последовательно отобразим в это пространство функции g, h_0 и h_1 , вводя при этом дополнительные фиктивные переменные из множеств v и u и представляя результаты 2^n -векторами a, b и c :

$h \times (u, w) - v = a$							
1000	1000	0000	1000	1111	1111	0000	1111
0000	0000	1000	0000	0000	0000	1111	0000
0000	1000	1000	0000	0000	1111	1111	0000
1000	1000	0000	0000	1111	1111	0000	0000
$g_0 \times (w, v) - u = b$							
0011	0100	1100	1001	0011	0100	1100	1001
0000	0000	0000	0000	0011	0100	1100	1001
0000	0000	0000	0000	0011	0100	1100	1001
0000	0000	0000	0000	0011	0100	1100	1001
$g_1 \times (w, v) - u = c$							
1010	0101	1010	1011	1010	0101	1010	1011
0000	0000	0000	0000	1010	0101	1010	1011
0000	0000	0000	0000	1010	0101	1010	1011
0000	0000	0000	0000	1010	0101	1010	1011

В заключение находим вектор f , представляющий искомую композицию функций $h(u, w)$ и $g(x, w, v)$:

$a b \vee \bar{a} c = f$							
0011	0100	0000	1001	0000	0000	1010	0000
0000	0000	1100	0000	1010	0101	0000	1011
0000	0100	1100	0000	1010	0000	0000	1011
0011	0100	0000	0000	0000	0000	1010	1011

4. Проверка частичной булевой функции на декомпозируемость по заданному разбиению на множестве аргументов

Допустим, что известна частичная булева функция $f(x)$ от n переменных, заданная троичным вектором f^- . Требуется проверить ее на разделимость по нестрогому разбиению u/v на множестве x , т.е. узнать, существуют ли такие функции $h(u, w)$ и $g(x, w, v)$ от меньшего числа переменных, что $f(x) = g(h(u, w), w, v)$, где $w = x \setminus (u \cup v)$.

Заметим, что при положительном ответе на этот вопрос можно упростить логическую схему, реализующую функцию $f(x)$ (например, при логическом синтезе в базисе элементов LUT (look up tables), реализующих функции ограниченного числа переменных).

Известно необходимое и достаточное условие декомпозируемости полностью определенной функции $f(x)$ по разбиению u/v , которое должно выполняться для каждого коэффициента $f_i(u, v)$ дизъюнктивного разложения Шеннона функции $f(x)$ по переменным множества w : коэффициенты дизъюнктивного разложения этих коэффициентов по переменным множества u должны принимать не более двух различных значений.

Коэффициенты $f_i(u, v)$ дизъюнктивного разложения Шеннона частичной булевой функции $f(x)$ по переменным множества w будем представлять *фрагментами* T_i – троичными матрицами, строки которых соответствуют различным значениям вектора u , а столбцы – различным значениям вектора v . Элементами фрагментов служат соответствующие компоненты троичного вектора f^- . Условие разделимости функции $f(x)$ по разбиению u/v сформулируем теперь следующим образом: для каждого коэффициента $f_i(u, v)$ возможно такое доопределение соответствующей матрицы T_i (замена значений «–» на 0 или 1), при котором ее строки примут не более двух различных значений.

В работе [7] показано, что проверка этого условия сводится к проверке на бихроматичность графов ортогональности строк каждой из матриц T_i . Там же предложен эвристический алгоритм, гарантирующий получение правильного решения при условии связности рассматриваемых графов (это условие обычно выполняется). Троичный вектор f^- представляется в нем соответствующей парой булевых векторов f^1 и f^0 . Эффективно используются операции над соседями, обеспечивающие одновременную проверку всех $2^{|w|}$ фрагментов T_i .

Алгоритм старается разбить множество строк в каждом фрагменте на два класса A и B взаимно совместимых строк. Над исходными векторами f^1 и f^0 реализуется последовательность преобразований, результаты которых представляются булевыми 2^n -векторами a^1 и a^0 .

Алгоритм итеративен. Первая итерация начинается с построения класса A путем включения в него первой строки фрагмента. Эта операция сводится к последовательности подстановок значения 0 на места переменных, образующих множество u :

$$a^0 := f^0 - u,$$

$$a^1 := f^1 - u.$$

Затем в каждом фрагменте находится множество строк, ортогональных первой строке, и отмечается в булевом векторе b :

$$b := S(h^0 f^1 \vee h^1 f^0) \vee v.$$

Полученные множества образуют классы B и проверяются на совместимость:

$$a^0 := S(f^0 b) \vee u;$$

$$a^1 := S(f^1 b) \vee u.$$

Если при этом $a^0 a^1 \neq 0$, то некоторое из рассматриваемых множеств оказывается несовместимым, откуда следует, что граф ортогональности строк соответствующего фрагмента не бихроматичен и, следовательно, функция $f(x)$ не декомпозируема по разбиению u/v .

Если же $a^0 a^1 = 0$, реализуется следующая итерация. Классы A дополняются строками, ортогональными некоторым из строк классов B , и проверяются на совместимость. Затем аналогичным образом могут быть расширены классы B и т.д. Алгоритм заканчивает работу при выполнении определенного числа итераций.

ЛИТЕРАТУРА

1. Закревский А.Д. Вычисления в булевых пространствах // Логическая структура научного знания. М.: Наука, 1965. С. 292 – 310.
2. Закревский А.Д. Универсальная система для решения задач типа синтеза релейных схем // Труды Сиб. физ.-тех. ин-та. Томск, 1963. Вып. 42. С. 9 – 37.
3. Закревский А.Д. Машина для решения логических задач типа синтеза релейных схем // Синтез релейных устройств: Труды Международного симпозиума по теории релейных устройств и конечных автоматов. М.: Наука, 1965. С. 346 – 356.
4. Хиллис У. Дэниел. Коммутационная машина // В мире науки (Scientific American). 1987, август. № 8. С. 60 – 69.
5. Data mining and knowledge discovery approaches based on rule induction techniques (E. Triantaphyllou and G. Felici, Eds.). Massive Computing Series, Springer, Heidelberg, Germany, 2006.
6. Закревский А.Д. О разрешимости булевых уравнений // Доклады НАН Белоруссии. 2007. Т. 51. № 5. С. 44 – 46.
7. Закревский А.Д. Декомпозиция частичных булевых функций – проверка на разделимость по заданному разбиению // Информатика. 2007. № 1(13). С. 16 – 21.

Статья представлена кафедрой программирования факультета прикладной математики и кибернетики Томского государственного университета и оргкомитетом 7 Российской конференции с международным участием «Новые информационные технологии в исследовании сложных структур», поступила в научную редакцию 30 мая 2008 г.