ИНФОРМАТИКА

2008 апрель-июнь № 2

УДК 519.714.7

Ю.В. Поттосин, Н.Р. Торопов, Е.А. Шестаков

МЕТОД МИНИМИЗАЦИИ СИСТЕМЫ ПОЛНОСТЬЮ ОПРЕДЕЛЕННЫХ БУЛЕВЫХ ФУНКЦИЙ

Рассматривается задача минимизации системы дизъюнктивных нормальных форм (ДНФ) произвольного вида, заданной в матричной форме. Критерием минимизации является общее число различных элементарных конъюнкций в получаемой системе ДНФ. Предлагается оригинальный способ сведения данной задачи к задаче о кратчайшем покрытии, основанный на простой операции пересечения множеств. Приводятся результаты испытаний компьютерной программы.

Введение

Классическая задача минимизации булевых функций является важной задачей в логическом проектировании дискретных устройств, привлекающей внимание специалистов соответствующей области. Об этом говорит обильный поток публикаций на данную тему, не ослабевающий за полвека существования задачи. Следует отличать минимизацию одной булевой функции от совместной минимизации системы булевых функций, целью которой является получение системы ДНФ с минимумом общего числа различных элементарных конъюнкций. На многих примерах можно убедиться, что совместная минимизация часто дает значительный эффект по сравнению с раздельной минимизацией каждой функции. Особенно данная задача важна при проектировании схем в базисе программируемых логических матриц, когда при ее решении достигается минимум площади кристалла, на котором размещается матрица.

Предлагаемый метод можно рассматривать как дальнейшее развитие метода Блейка – Порецкого [1] и распространение его на случай системы булевых функций. Он рассчитан на системы полностью определенных булевых функций, представляемых в ДНФ произвольного вида. При этом используется матричный способ задания системы ДНФ, когда исходная система задается в виде пары матриц ($\boldsymbol{U}$, $\boldsymbol{V}$), где $\boldsymbol{U}$ – троичная матрица, представляющая своими строками элементарные конъюнкции, а $\boldsymbol{V}$ – булева матрица, столбцы которой своими единицами показывают, какие конъюнкции каким ДНФ принадлежат [2].

Рассматриваемая задача, как и в известных методах минимизации булевых функций, сводится в настоящей работе к задаче покрытия булевой матрицы, однако для такого сведения применяется оригинальный способ, основанный на использовании совокупности подмножеств строк троичной матрицы, которая близка к понятию покрытия троичной матрицы. Это понятие было первоначально введено при исследовании задачи декомпозиции булевых функций [3]. Оно подобно понятию «blanket», которое упоминалось в работе [4]. Подход, основанный на понятии покрытия троичной матрицы, успешно применялся при решении таких задач, как декомпозиция булевых функций [5], выявление существенных аргументов [6] и ортогонализация булевых функций [7].

1. Получение сокращенной системы ДНФ

Пусть задана система булевых функций $F = \{f_1, f_2, \dots, f_m\}$ от общего множества аргументов $X = \{x_1, x_2, \dots, x_n\}$, образующих булево пространство M. Множества $M_i^1 \subseteq M$ и $M_i^0 \subseteq M$ представляют соответственно области единичных и нулевых значений функции f_i ($i = 1, 2, \dots, m$).

Рассмотрим пару множеств (I, G), где I – интервал булева пространства M, а G – множество тех функций $f_i \in F$, для которых $I \subseteq M_i^1$. Назовем пару (I, G) *g-интервалом*. *Максимальным* g-интервалом является такой g-интервал (I, G), что $I \cap M_j^0 \neq \varnothing$ для любой функции $f_j \notin G$ из F, а любое расширение интервала I нарушает условие $I \subseteq M_i^1$ хотя бы для одной из функций $f_i \in G$. Как показано в работе [8], интервал I из пары (I, G) соответствует простой импликанте функции, выражаемой конъюнкцией $\bigwedge\limits_{f_i \in G} f_i$.

Задача ставится следующим образом: для системы булевых функций, заданных в виде ДНФ, найти минимальную совокупность максимальных g-интервалов, покрывающую все пары вида (m_k, f_l), где $f_l \in F$ и $m_k \in M_l^1$.

Пусть система булевых функций F задана парой матриц $(\boldsymbol{U}, \boldsymbol{V})$, любая пара одноименных строк $(\boldsymbol{u}, \boldsymbol{v})$ которой представляет некоторый g-интервал (I, G) так, что троичный вектор u представляет интервал I, а булев вектор v – множество функций G. Другой интерпретацией пары матриц $(\boldsymbol{U}, \boldsymbol{V})$ является система ДНФ, где строка $\boldsymbol{u}$ матрицы $\boldsymbol{U}$ представляет элементарную конъюнкцию, а строка v матрицы $\boldsymbol{V}$ – множество ДНФ, содержащих данную конъюнкцию.

Первым этапом минимизации ДНФ по рассматриваемому методу является получение всех максимальных g-интервалов для заданной системы булевых функций. Для решения этой задачи матрицы $\boldsymbol{U}$ и $\boldsymbol{V}$ преобразуются с помощью следующих операций над их строками. Пусть имеется две пары строк матриц $\boldsymbol{U}$ и $\boldsymbol{V}$ – $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ и $(\boldsymbol{u}_j, \boldsymbol{v}_j)$.

Обобщенное склеивание. Матрица $\boldsymbol{U}$ дополняется продуктом обобщенного склеивания троичных векторов $\boldsymbol{u}_i$ и $\boldsymbol{u}_j$, а матрица $\boldsymbol{V}$ – результатом выполнения поразрядной конъюнкции над булевыми векторами $\boldsymbol{v}_i$ и $\boldsymbol{v}_j$. Операция обобщенного склеивания выполняется только тогда, когда векторы $\boldsymbol{u}_i$ и $\boldsymbol{u}_j$ смежны [2] (т. е. ортогональны только по одной компоненте), а результат поразрядной конъюнкции векторов $\boldsymbol{v}_i$ и $\boldsymbol{v}_j$ отличен от нулевого вектора. Назовем такие пары $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ и $(\boldsymbol{u}_j, \boldsymbol{v}_j)$ *смежными*.

Пересечение. Матрица $\boldsymbol{U}$ дополняется результатом пересечения троичных векторов $\boldsymbol{u}_i$ и $\boldsymbol{u}_j$, а матрица $\boldsymbol{V}$ – результатом выполнения поразрядной дизъюнкции над булевыми векторами $\boldsymbol{v}_i$ и $\boldsymbol{v}_j$. Операция пересечения выполняется только при неортогональности векторов $\boldsymbol{u}_i$ и $\boldsymbol{u}_j$. Такие пары $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ и $(\boldsymbol{u}_j, \boldsymbol{v}_j)$ назовем *пересекающимися*.

Поглощение. Если вектор $\boldsymbol{u}_i$ поглощает вектор $\boldsymbol{u}_j$ [2], а вектор $\boldsymbol{v}_i$ имеет единицы везде, где имеет единицы вектор $\boldsymbol{v}_j$, то строки $\boldsymbol{u}_j$ и $\boldsymbol{v}_j$ удаляются из матриц $\boldsymbol{U}$ и $\boldsymbol{V}$. Будем говорить в этом случае, что пара $(\boldsymbol{u}_i, \boldsymbol{v}_i)$ *поглощает* пару $(\boldsymbol{u}_j, \boldsymbol{v}_j)$. Проверка условия, при котором выполняется эта операция, производится всякий раз, когда получена новая пара строк в результате обобщенного склеивания или пересечения. Если условие выполняется, то выполняется данная операция.

Рекомендуется выполнять описанные операции в следующем порядке. Строки матриц просматриваются сверху вниз, и для очередной рассматриваемой пары строк отыскиваются смежные и пересекающиеся пары, расположенные только сверху от нее. Полученный продукт обобщенного склеивания или пересечения, если он не поглощается другой парой строк, приписывается к матрицам снизу. Если он сам поглощает какую-либо пару строк, то эта пара удаляется. Процесс заканчивается, когда, дойдя до последней строки (включая вновь появившиеся), обнаружим, что новые строки в матрицах не появляются.

В результате описанных преобразований получаем пару матриц $(\boldsymbol{U}, \boldsymbol{V})$, представляющую все максимальные g-интервалы. Можно рассматривать ее как представление системы ДНФ, которую, следуя традиционной терминологии в теории булевых функций, назовем *сокращенной системой ДНФ*.

В качестве примера для демонстрации описанных преобразований возьмем систему из трех ДНФ [9], каждая из которых является минимальной (результат раздельной минимизации). Представим эту систему в виде пары матриц $(\boldsymbol{U}, \boldsymbol{V})$:

$$\boldsymbol{U} = \begin{array}{c} \begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\ \begin{bmatrix} - & 0 & 0 \\ 1 & - & 1 \\ 0 & - & 1 \\ - & 1 & 0 \\ - & 0 & 1 \\ 1 & 1 & - \\ 0 & - & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}, \qquad \boldsymbol{V} = \begin{array}{c} \begin{array}{ccc} f_1 & f_2 & f_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}.$$

В результате обобщенного склеивания смежных пар ($\boldsymbol{u}_1, \boldsymbol{v}_1$) и ($\boldsymbol{u}_2, \boldsymbol{v}_2$) получим пару векторов (1 0 –, 1 0 0), которая добавляется в матрицы $\boldsymbol{U}$ и $\boldsymbol{V}$ в качестве новых строк. Затем добавляется пара (0 1 –, 0 1 0), которая является продуктом обобщенного склеивания смежных пар ($\boldsymbol{u}_3, \boldsymbol{v}_3$) и ($\boldsymbol{u}_4, \boldsymbol{v}_4$). Пара ($\boldsymbol{u}_5, \boldsymbol{v}_5$) пересекается с парами ($\boldsymbol{u}_2, \boldsymbol{v}_2$) и ($\boldsymbol{u}_3, \boldsymbol{v}_3$). Выполнив над ними операцию пересечения, получим пары (1 0 1, 1 0 1) и (0 0 1, 0 1 1), которые также добавляются в матрицы $\boldsymbol{U}$ и $\boldsymbol{V}$. Продолжая этот процесс с удалением поглощаемых пар строк, окончательно получим

$$
\begin{array}{c}
\begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\
\begin{bmatrix}
- & 0 & 0 \\
0 & - & 1 \\
- & 0 & 1 \\
1 & 1 & - \\
0 & - & 0 \\
1 & 0 & - \\
0 & 1 & - \\
0 & 0 & 1 \\
0 & 0 & 0 \\
0 & 0 & - \\
1 & - & 1 \\
- & 1 & 0
\end{bmatrix}
\end{array}
\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{array}
,\quad
\begin{array}{c}
\begin{array}{ccc} f_1 & f_2 & f_3 \end{array} \\
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 1 \\
0 & 0 & 1 \\
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 1 & 1 \\
1 & 0 & 1 \\
0 & 0 & 1 \\
1 & 0 & 1 \\
0 & 1 & 1
\end{bmatrix}
\end{array}
\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{array}
. \qquad (1)
$$

Матрицы (1) представляют сокращенную систему ДНФ, т. е. их строки представляют все максимальные g-интервалы, из которых надо выбрать минимальную совокупность, покрывающую все пары вида (m_k, f_l), где в данном случае $f_l \in \{f_1, f_2, f_3\}$ и $m_k \in M_l^1$.

2. Сведéние к задаче о кратчайшем покрытии

Следующим этапом процесса минимизации является построение матрицы покрытия, строкам которой соответствуют максимальные g-интервалы, а столбцам – пары вида (m_k, f_l), и нахождение минимального количества строк, покрывающих все столбцы. Можно разбить полученные интервалы на отдельные элементы булева пространства M, тогда получим пары (m_k, f_l) в явном виде. В этом случае возникает сомнение в целесообразности применения описанного метода получения максимальных g-интервалов. Достаточно разбить интервалы из исходного задания и затем применить известный метод Квайна – МакКласки, использующий операцию простого склеивания. Здесь предлагается другой способ выполнения рассматриваемого этапа, который не требует разбиения интервалов на отдельные элементы пространства M.

Пусть $\boldsymbol{U}$ – троичная матрица, представляющая множество интервалов, которое составляет область M^1 некоторой булевой функции f. Рассмотрим совокупность $P(\boldsymbol{U})$ непустых подмножеств множества номеров строк матрицы $\boldsymbol{U}$, такую, что для каждого элемента множества M^1 имеется подмножество в совокупности $P(\boldsymbol{U})$, состоящее из номеров всех строк матрицы $\boldsymbol{U}$, которые представляют содержащие данный элемент интервалы, и других подмножеств в $P(\boldsymbol{U})$ нет. Например, если матрицей $\boldsymbol{U}$ является исходная матрица приведенного примера, то для вектора (0 0 0) таким подмножеством является {1, 7}.

Введем обозначение: $t(\boldsymbol{m}, \boldsymbol{U})$ – множество номеров тех строк матрицы $\boldsymbol{U}$, которые представляют интервалы, содержащие булев вектор $\boldsymbol{m}$. Например, для той же исходной матрицы и $\boldsymbol{m}$ = (0 0 0) имеем $t(\boldsymbol{m}, \boldsymbol{U}) = \{1, 7\}$. Ясно, что $t(\boldsymbol{m}, \boldsymbol{U})$ является элементом множества $P(\boldsymbol{U})$.

Пусть $\boldsymbol{U}$ – матрица, задающая совокупность интервалов, покрывающую область булева пространства M^1. Задачу кратчайшего покрытия области M^1 интервалами из заданного множества можно сформулировать следующим образом: для матрицы $\boldsymbol{U}$ построить минимальную совокупность ее строк, такую, чтобы любой элемент множества $P(\boldsymbol{U})$ содержал номер хотя бы одной строки из данной совокупности.

Пусть система булевых функций $F = \{f_1, f_2, \ldots, f_m\}$ задана парой матриц $(\boldsymbol{U}, \boldsymbol{V})$, представляющей сокращенную систему ДНФ. Выделим строчные миноры $\boldsymbol{U}^1, \boldsymbol{U}^2, \ldots, \boldsymbol{U}^m$ в матрице $\boldsymbol{U}$ с сохранением нумерации строк, где $\boldsymbol{U}^i$ представляет ДНФ соответствующей функции f_i ($i = 1, 2, \ldots, m$) и определяется единицами i-го столбца матрицы $\boldsymbol{V}$. Для каждого минора $\boldsymbol{U}^i$ получим $P(\boldsymbol{U}^i)$. Теперь для нахождения кратчайшего покрытия множества всех пар вида (m_k, f_l) максимальными g-интервалами надо построить минимальную совокупность строк матрицы $\boldsymbol{U}$, такую, чтобы любой элемент множества $P(\boldsymbol{U}^1) \cup P(\boldsymbol{U}^2) \cup \ldots \cup P(\boldsymbol{U}^m)$ содержал номер хотя бы одной строки из данной совокупности.

Сформулируем правила редукции, которые можно применять при решении задачи покрытия и которые согласуются с правилами, сформулированными в работе [2]:

1. Если $A \in P(\boldsymbol{U})$, $B \in P(\boldsymbol{U})$ и $A \subseteq B$, то B удаляется из $P(\boldsymbol{U})$.

2. Если номер i присутствует только в тех элементах множества $P(\boldsymbol{U})$, где присутствует номер k, то i удаляется отовсюду.

После преобразования множества $P(\boldsymbol{U})$ согласно приведенным правилам редукции удобно представить это множество в виде булевой матрицы покрытия $\boldsymbol{B}$, строки которой соответствуют строкам матрицы $\boldsymbol{U}$, а столбцы – элементам множества $P(\boldsymbol{U})$. Каждый столбец является векторным представлением соответствующего элемента множества $P(\boldsymbol{U})$. Задача теперь заключается в том, чтобы в матрице $\boldsymbol{B}$ выделить минимум строк так, чтобы каждый столбец имел единицу хотя бы в одной из выделенных строк.

Остановимся на способе получения множества $P(\boldsymbol{U})$. Пусть для двух троичных матриц $\boldsymbol{U}'$ и $\boldsymbol{U}''$ с одинаковым числом строк получены множества $P(\boldsymbol{U}')$ и $P(\boldsymbol{U}'')$.

У т в е р ж д е н и е. *Если из матриц $\boldsymbol{U}'$ и $\boldsymbol{U}''$ построить матрицу $\boldsymbol{U}$, приписав столбцы матрицы $\boldsymbol{U}''$ к столбцам $\boldsymbol{U}'$, то множество $P(\boldsymbol{U})$ можно получить, взяв за его элементы всевозможные непустые парные пересечения элементов $P(\boldsymbol{U}')$ с элементами $P(\boldsymbol{U}'')$.*

Например, преобразуем матрицы $\boldsymbol{U}'$ и $\boldsymbol{U}''$ следующим образом. К матрице $\boldsymbol{U}'$ припишем справа столько столбцов, сколько их имеет матрица $\boldsymbol{U}''$. Всем элементам данных столбцов придадим значение «–». Такие же столбцы, число которых равно числу столбцов исходной матрицы $\boldsymbol{U}'$, припишем слева к матрице $\boldsymbol{U}''$. Очевидно, что $P(\boldsymbol{U}')$ и $P(\boldsymbol{U}'')$ при этом не изменятся. Произвольно выберем булев вектор $\boldsymbol{m}$, принадлежащий какому-нибудь интервалу, представляемому строкой матрицы $\boldsymbol{U}$. По построению матрицы $\boldsymbol{U}$ имеем $t(\boldsymbol{m}, \boldsymbol{U}) = t(\boldsymbol{m}, \boldsymbol{U}') \cap t(\boldsymbol{m}, \boldsymbol{U}'')$, т. е. каждый элемент множества $P(\boldsymbol{U})$ является пересечением двух множеств, одно из которых – элемент множества $P(\boldsymbol{U}')$, а другое – элемент множества $P(\boldsymbol{U}'')$. С другой стороны, для каждой пары пересекающихся элементов, один из которых взят из $P(\boldsymbol{U}')$, а другой – из $P(\boldsymbol{U}'')$, найдется вектор $\boldsymbol{u}$, такой, что $t(\boldsymbol{m}, \boldsymbol{U}')$ и $t(\boldsymbol{m}, \boldsymbol{U}'')$ являются теми самыми элементами. Множество $t(\boldsymbol{m}, \boldsymbol{U}') \cap t(\boldsymbol{m}, \boldsymbol{U}'')$ состоит из номеров всех строк матриц $\boldsymbol{U}'$ и $\boldsymbol{U}''$, каждая из которых представляет интервал, содержащий $\boldsymbol{m}$. Пересечение интервалов, представляемых строками матриц $\boldsymbol{U}'$ и $\boldsymbol{U}''$ с общим номером i, равно интервалу, представляемому i-й строкой матрицы $\boldsymbol{U}$. Следовательно, для любой пары $t(\boldsymbol{m}, \boldsymbol{U}')$, $t(\boldsymbol{m}, \boldsymbol{U}'')$ найдется множество $t(\boldsymbol{m}, \boldsymbol{U})$, такое, что $t(\boldsymbol{m}, \boldsymbol{U}) = t(\boldsymbol{m}, \boldsymbol{U}') \cap t(\boldsymbol{m}, \boldsymbol{U}'')$.

Если матрица $\boldsymbol{U}$ состоит из одного столбца и произвольного числа одноэлементных строк, то $P(\boldsymbol{U})$ состоит не более чем из двух элементов. Один из них представляет множество, состоящее из номеров всех строк, имеющих нули и «–», второй – множество, состоящее из номеров всех строк, имеющих единицы и «–». Если все элементы одностолбцовой матрицы $\boldsymbol{U}$ имеют одно и то же значение, то $P(\boldsymbol{U})$ является одноэлементным множеством.

Пусть матрица $\boldsymbol{U}$ имеет n столбцов. Для минора $\boldsymbol{U}_1$ матрицы $\boldsymbol{U}$, состоящего из ее единственного первого столбца, получим $P(\boldsymbol{U}_1)$, как указано выше. Множество $P(\boldsymbol{U}_2)$ для минора, состоящего из двух первых столбцов матрицы $\boldsymbol{U}$, получим путем всевозможных пересечений элементов множества $P(\boldsymbol{U}_1)$ с элементами подобного множества для второго столбца. Для минора $\boldsymbol{U}_3$, состоящего из трех первых столбцов матрицы $\boldsymbol{U}$, получим $P(\boldsymbol{U}_3)$ аналогично путем всевозможных пересечений элементов множества $P(\boldsymbol{U}_2)$ с элементами подобного множества для третьего столбца. Продолжая таким образом выполнять операцию пересечения множеств, получим, наконец, $P(\boldsymbol{U}_n) = P(\boldsymbol{U})$. Если в результате вычисления некоторого $P(\boldsymbol{U}_i)$ $(1 \le i \le n)$ появилось одноэлементное множество, то строка матрицы $\boldsymbol{U}$ с соответствующим номером включается в решение как принадлежащая ядру [2].

Вернемся к рассматриваемому примеру. ДНФ функций f_1, f_2 и f_3 представляются соответственно следующими минорами левой из пары матриц ($\boldsymbol{U}$, $\boldsymbol{V}$), представляющей полученную ранее сокращенную систему ДНФ (1):

$$\boldsymbol{U}^1 = \begin{array}{c} \begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\ \begin{bmatrix} - & 0 & 0 \\ 1 & 0 & - \\ 0 & 0 & 0 \\ 1 & - & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 6 \\ 9 \\ 11 \end{array}, \quad \boldsymbol{U}^2 = \begin{array}{c} \begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\ \begin{bmatrix} 0 & - & 1 \\ 0 & 1 & - \\ 0 & 0 & 1 \\ - & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 2 \\ 7 \\ 8 \\ 12 \end{array}, \quad \boldsymbol{U}^3 = \begin{array}{c} \begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\ \begin{bmatrix} - & 0 & 1 \\ 1 & 1 & - \\ 0 & - & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & - \\ 1 & - & 1 \\ - & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 3 \\ 4 \\ 5 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{array}.$$

Найдем множества $P(\boldsymbol{U}^1)$, $P(\boldsymbol{U}^2)$ и $P(\boldsymbol{U}^3)$, используя описанный выше способ:

$P(\boldsymbol{U}^1{}_1) = \{(1,6,11), (1,9)\}$; $P(\boldsymbol{U}^1{}_2) = \{(1,6,11), (1,9), (11)\}$;
$P(\boldsymbol{U}^1{}_3) = P(\boldsymbol{U}^1) = \{(1,6), (1,9), (6,11), (11)\}$.

$P(\boldsymbol{U}^2{}_1) = \{(2,7,8,12), (12)\}$; $P(\boldsymbol{U}^2{}_2) = \{(2,7,12), (12), (2,8)\}$;
$P(\boldsymbol{U}^2{}_3) = P(\boldsymbol{U}^2) = \{(2,7), (2,8), (7,12), (12)\}$.

$P(\boldsymbol{U}^3{}_1) = \{(3,4,11,12), (3,5,8,9,10,12)\}$; $P(\boldsymbol{U}^3{}_2) = \{(3,11), (3,5,8,9,10), (4,11,12), (5,12)\}$;
$P(\boldsymbol{U}^3{}_3) = P(\boldsymbol{U}^3) = \{(3,11), (3,8,10), (4,11), (5,9,10), (4,12), (5,12)\}$.

Объединение полученных множеств, редуцированное по сформулированным выше правилам редукции, имеет вид {(1), (2), (10), (11), (12)}, что представляет собой искомое решение задачи о покрытии.

Окончательно получим следующую минимальную систему ДНФ:

$$\boldsymbol{U} = \begin{array}{c} \begin{array}{ccc} x_1 & x_2 & x_3 \end{array} \\ \begin{bmatrix} - & 0 & 0 \\ 0 & - & 1 \\ 0 & 0 & - \\ 1 & - & 1 \\ - & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 10 \\ 11 \\ 12 \end{array}, \qquad \boldsymbol{V} = \begin{array}{c} \begin{array}{ccc} f_1 & f_2 & f_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ 1 \\ 2 \\ 10 \\ 11 \\ 12 \end{array}.$$

3. Программная реализация алгоритма

Описанный выше метод программно реализован на C++ и экспериментально испытан на персональном компьютере Intel Celeron 952 MHz, 384 MB RAM на известных типовых примерах Benchmark [10] и на потоке псевдослучайных систем булевых функций.

Слабым звеном всех алгоритмов строгой минимизации является их большая трудоемкость. Поэтому при реализации предлагаемого метода были использованы специальные приемы ускорения отдельных его этапов.

Применение в программном модуле получения сокращенной системы ДНФ ($\boldsymbol{U}$, $\boldsymbol{V}$) механизма автоматической настройки на классы логических объектов [11] в зависимости от размерности строк матриц ($\boldsymbol{U}$, $\boldsymbol{V}$) позволило в несколько раз повысить быстродействие модуля при $n \le 32$ и $m \le 32$, где n – число переменных, m – число функций, по сравнению с аналогичным модулем, работающим с объектами произвольной размерности.

Всюду, где это возможно, широко применяются транспонированные матрицы, чтобы дорогостоящие операции над их столбцами заменить более дешевыми операциями над строками.

При поиске кратчайшего покрытия используются приемы упреждающего отсечения ветвей в дереве перебора вариантов путем проверки наличия в них однострочных или двухстрочных покрытий [2].

Результаты экспериментальных испытаний алгоритма приведены в табл. 1–5, которые дают представление о трудоемкости отдельных этапов. В таблицах используются следующие обозначения:

n – число переменных;
m – число функций;
p – число конъюнкций в исходной системе ДНФ;
nr – средний ранг конъюнкций в исходной псевдослучайной системе ДНФ;
mr – среднее число ДНФ, в которые входит конъюнкция исходной системы ДНФ;
imp – число импликант (g-интервалов) в сокращенной системе ДНФ;
nuc – число импликант, составляющих ядро;
col – число столбцов в матрице покрытия $\boldsymbol{B}$;
q – число g-интервалов в результирующей системе ДНФ;
t_i – время построения сокращенной системы ДНФ с учетом ее размерности;
t_{ia} – время построения сокращенной системы ДНФ без учета ее размерности;
t_b – время построения матрицы покрытия $\boldsymbol{B}$;
t_c – время поиска кратчайшего покрытия.

Заметим, что время в таблицах указывается в секундах. В табл. 1 приведены результаты минимизации систем булевых функций из некоторых известных примеров серии Benchmark [10] с помощью предлагаемого алгоритма.

Таблица 1

Минимизация систем ДНФ из набора Benchmark

Пример	n	m	p	q	imp	nuc	col	t_{ia}	t_i	t_b	t_c
Max512	9	6	512	133	535	20	418	12,04	1,34	0,58	272,74
Alu1	12	8	19	19	780	19	0	40,68	3,62	2,83	0,02
mp2d	14	14	123	30	469	13	457	188,28	20,00	16,58	6,14
b12	15	9	431	41	1490	2	262	135,49	11,77	52,78	3,27
Gary	15	11	214	107	706	60	99	4,85	0,46	1,85	4,33
b2	16	17	110	104	928	54	155	7,20	0,72	5,11	27,15
in2	19	10	137	134	666	85	128	5,51	0,47	21,18	5,10
in5	24	14	62	62	1067	53	16	10,04	0,92	43,85	1,22
in7	26	10	84	54	2112	31	74	250,00	22,35	25018,74	8,64
Chkn	29	7	153	140	671	86	106	5,55	0,47	485,87	15,22

В табл. 2–5 приведены результаты экспериментальных испытаний алгоритма на потоке псевдослучайных систем булевых функций по различным осям пространства параметров. Прочерк в некоторых позициях таблиц означает, что минимизация в соответствующей точке пространства параметров не доведена до конца, так как был превышен лимит времени в 40 мин.

Таблица 2

Результаты минимизации псевдослучайных систем ДНФ при n=12, nr=6, m=20, mr=5 и различных значениях p

p	imp	nuc	col	t_{ia}	t_i	t_b	t_c
50	1983	49	4	30,60	3,12	3,43	2,46
60	3153	56	13	98,66	9,48	6,52	5,06
70	3969	61	51	182,68	17,22	9,52	8,68
80	6147	67	85	706,96	58,50	22,68	16,63
90	9404	62	366	–	207,32	65,33	33,15
100	12740	67	594	–	566,97	115,01	53,42
110	16211	67	1011	–	1166,79	200,89	101,65

Соотношение значений в столбцах t_{ia} и t_i табл. 1 и 2 показывает, что применение механизма автоматической настройки на классы логических объектов в зависимости от размерности минимизируемых систем ДНФ примерно на порядок повышает быстродействие модуля построения сокращенной системы ДНФ при $n \leq 32$ и $m \leq 32$ по сравнению с аналогичным модулем, работающим с объектами произвольной размерности.

Таблица 3

Результаты минимизации псевдослучайных систем ДНФ при p=100, n=12, nr=6, m=20 и различных значениях mr

mr	imp	nuc	col	t_i	t_b	t_c
2	4418	52	438	14,73	9,85	2,53
4	10110	47	820	258,40	78,37	42,37
6	13378	74	378	548,11	104,86	60,46
8	18181	61	1385	1999,10	338,00	229,11

Таблица 4

Результаты минимизации псевдослучайных систем ДНФ при p=100, n=12, nr=6 и различных значениях m и mr=0,2m

m	imp	nuc	col	t_i	t_b	t_c
10	5562	59	560	68,33	17,39	12,83
15	5640	69	324	33,83	19,37	10,08
20	10110	47	820	261,18	77,90	42,76
25	12446	53	1044	454,55	157,66	108,75
30	15874	51	1229	896,45	280,50	153,76

Таблица 5

Время минимизации псевдослучайных систем ДНФ при p=100, m=20, mr=5 и различных значениях n и nr

n	nr									
	n-9	n-8	n-7	n-6	n-5	n-4	n-3	n-2	n-1	n
9		8,49	8,06	3,23	–	–	67,38	33,19	36,09	0,23
10	10,34	15,81	–	–	1096,01	387,12	61,04	15,36	13,97	0,19
11	22,52	89,43	–	–	434,85	107,44	14,55	6,42	5,40	0,15
12	475,09	–	–	778,25	202,54	16,11	20,37	9,01	2,90	0,08
13	–	–	–	683,09	51,50	23,96	6,76	1,89	2,93	0,10
14	–	–	404,16	85,76	25,28	7,66	4,49	2,86	0,68	0,09
15	–	–	146,11	38,87	5,45	5,87	1,00	1,71	0,76	1,33
16	–	149,97	36,81	26,47	10,36	3,29	0,79	1,04	0,54	0,06

Заключение

Описанный подход предназначен для решения задач теории булевых функций, связанных с их заданием в виде троичных матриц, которые представляют ДНФ.

Проведенные эксперименты показали, что алгоритм, основанный на предложенном методе строгой минимизации системы булевых функций, заданных в ДНФ произвольного вида, не только представляет определенный теоретический интерес, но и обладает довольно приличной нишей в пространстве параметров (n, m, p) для практического его применения. Алгоритм можно также использовать при перепроектировании комбинационных схем, например в тех случаях, когда некоторую схему, структура которой представлена в виде некоторого множества минимальных ДНФ, надо реализовать на программируемой логической матрице. Еще одной стороной применения данного алгоритма может служить оценка качества решений, получаемых приближенными алгоритмами минимизации систем булевых функций; в этом случае его можно использовать как эталон.

В качестве иллюстрации такого применения приведем табл. 6, в которой даны результаты минимизации примеров систем из табл. 1 приближенным алгоритмом А, базирующемся на методе конкурирующих интервалов [12]. В таблице кроме введенных ранее используются еще следующие обозначения:

q_a – число конъюнкций в решении, полученном алгоритмом А;

t_e – время минимизации системы ДНФ строгим алгоритмом;

t_a – время минимизации той же системы алгоритмом А.

Таблица 6

Минимизация систем ДНФ из набора Benchmark

Пример	n	m	p	q	q_a	t_e	t_a
Max512	9	6	512	133	149	274,66	52,66
Alu1	12	8	19	19	19	6,47	0,03
mp2d	14	14	123	30	36	42,72	1,56
b12	15	9	431	41	48	67,82	17,10
Gary	15	11	214	107	107	6,64	7,00
b2	16	17	110	104	104	32,98	2,66
in2	19	10	137	134	135	26,75	5,23
in5	24	14	62	62	62	45,99	0,83
in7	26	10	84	54	55	25049,73	1,36
Chkn	29	7	153	140	145	474,56	7,61

Предлагаемый метод может иметь преимущество по сравнению с методом простых совокупностей, описанным в работах [2, 13], для троичных матриц с относительно большим числом строк и с малым числом столбцов.

Список литературы

1. Глушков, В.М. Синтез цифровых автоматов / В.М. Глушков. – М.: ГИФМЛ, 1962. – 476 с.

2. Закревский, А.Д. Логический синтез каскадных схем / А.Д. Закревский. – М.: Наука, 1981. – 416 с.

3. Шестаков, Е.А. Декомпозиция системы полностью определенных булевых функций по покрытию аргументов / Е.А. Шестаков // Автоматика и вычислительная техника. – 1994. – № 1. – С. 12–20.

4. Brzozowski, J.A. Decomposition of Boolean functions specified by cubes. Research Report CS-97-01 / J.A. Brzozowski, J.J. Lou. – Waterloo, ON, Canada: Department of Computer Science, University of Waterloo, 1997. – 36 p.

5. Поттосин, Ю.В. Декомпозиция систем полностью определенных булевых функций по их заданию в виде компактных таблиц / Ю.В. Поттосин, Е.А. Шестаков // Информатика. – 2004. – № 2. – С. 35–44.

6. Поттосин, Ю.В. Поиск существенных аргументов системы полностью определенных булевых функций / Ю.В. Поттосин, Е.А. Шестаков // Методы логического проектирования.– Минск: ОИПИ НАН Беларуси, 2003. – Вып. 2. – С. 69–78.

7. Поттосин, Ю.В. Ортогонализация системы полностью определенных булевых функций / Ю.В. Поттосин, Е.А. Шестаков // Логическое проектирование. – Минск: Ин-т техн. кибернетики НАН Беларуси, 2000. – Вып. 5.– С. 107–115.

8. Фридман, А. Теория и проектирование переключательных схем / А. Фридман, П. Менон. – М.: Мир, 1978. – 580 с.

9. Миллер, Р. Теория переключательных схем. Т. I / Р. Миллер. – М.: Наука, 1970. – 416 с.

10. Yang, S. Logic Synthesis and Optimization Benchmarks. User Guide: Version 3.0. Technical Report / S. Yang. – Microelectronics of North Carolina, 1991. – 43 p.

11. Романов, В.И. Программные средства для решения логико-комбинаторных задач / В.И. Романов // Информатика. – 2005. – № 4 (8). – С. 114–123.

12. Торопов, Н. Р. Минимизация систем булевых функций в классе ДНФ / Н. Р. Торопов // Логическое проектирование. – Минск: Ин-т техн. кибернетики НАН Беларуси, 1999. – Вып. 4. – С. 4–19.

13. Закревский, А.Д. Основы логического проектирования: в 3 кн. Кн. 2. Оптимизация в булевом пространстве / А.Д. Закревский, Ю.В. Поттосин, Л.Д. Черемисинова. – Минск: ОИПИ НАН Беларуси, 2004. – 240 с.

Поступила 06.06.07

Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: pott@newman.bas-net.by

Yu.V. Pottosin, N.R. Toropov, E.A. Shestakov

METHOD FOR MINIMIZING A SYSTEM OF COMPLETELY SPECIFIED BOOLEAN FUNCTIONS

A problem of minimization of a system of disjunctive normal forms (DNFs) that is given in a matrix form is considered. The criterion of minimization is the total number of different elementary conjunctions in the obtained DNF system. A technique to obtain the reduced DNF is described. A novel way to reduce this problem to the problem of the shortest covering is suggested that is based on a simple operation of set intersection. The results of testing the computer program are given.