

ТЕХНИЧЕСКИЕ НАУКИ

Ефанова Надежда Викторовна

студентка

Муромцев Виктор Владимирович

канд. техн. наук, доцент, заведующий кафедрой

ФГАОУ ВПО «Белгородский государственный
национальный исследовательский университет»

г. Белгород, Белгородская область

ИСПОЛЬЗОВАНИЕ ТАБЛИЦ РЕШЕНИЙ ДЛЯ ПРОЕКТИРОВАНИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Аннотация: в данной статье авторами рассматриваются вопросы простоты и удобства модификации программного обеспечения в использовании таблиц решений для построения бинарных диаграмм. Для изучения характеристик бинарных диаграмм использовался детальный анализ таблиц решений, способов их построения и функции, позволяющие реализовать построение бинарных диаграмм.

Ключевые слова: таблица решений, бинарная диаграмма, булевы функции, граф, ветвление.

Одной из важнейших характеристик программного обеспечения (ПО) является простота его модификации. Для создания легко модифицируемых программ существуют различные методы и приемы. Одним из таких приемов является использование таблиц решений на начальных этапах проектирования. Такие таблицы позволяют в компактной форме установить связь между условиями и действиями.

Кроме таблиц решений для определения связи между условиями и действиями также часто используют бинарные диаграммы решений (БДР) или соответ-

ствующие им блок-схемы, диаграммы деятельности, конечные автоматы. Преимущества БДР в том, что по ним легко синтезировать программы. Однако таблицы решений обладают рядом преимуществ перед БДР. Так при описании сложных условий таблица решений компактнее и проще для понимания.

Целью исследования являются способы, позволяющие осуществлять построение бинарных диаграмм для программирования таблиц решений. В связи с этим возникли следующие задачи:

- осуществить анализ предметной области;
- выяснить, что представляет собой бинарная диаграмма;
- изучить и освоить метод построения таблиц решений;
- узнать какие логические функции используются в методах построения.

Бинарная диаграмма решений (БДР) или же, так называемая программа с ветвлением, является формой представления булевой функции $f(x_1, x_2, \dots, x_n)$ от n переменных в виде направленного ациклического графа, состоящего из внутренних узлов решений, которые принимают значение x_i , каждое из которых имеет по два потомка, а также два терминальных узла, помеченных как 0 и 1. При этом каждый из которых соответствует одному из двух значений булевой функции.

В зарубежной литературе бинарные диаграммы решений, и соответственно программы с ветвлением называют binary decision diagram или сокращенно BDD и branching programs – BP соответственно. Данную функцию $f(x_1, x_2, \dots, x_n)$ можно представить в виде графа, состоящего из нескольких внутренних узлов решений и двух терминальных узлов, называемых 0-терминальный узел и 1-терминальный узел. Каждый внутренний узел решений на уровне i помечен булевой переменной x_i и имеет по два потомка, называемых младшим потомком и старшим потомком.

Во время перехода от внутреннего узла x_i к младшему или старшему потомку выполняется в зависимости от значения переменной x_i , то есть 0 или 1 соответственно. Для заданных значений $x_1, x_2, \dots, x_n$ путь от корневого узла до 1-терминального узла соответствуют тому, что на этих входных значениях булева функция $f(x_1, x_2, \dots, x_n)$ принимает значение 1.

БДР называется *упорядоченной*, если различные переменные появляются в том же порядке на всем пути от корневого узла графа.

БДР называется *сокращенной*, если для графа применены следующие два правила сокращения:

- слияние любых изоморфных подграфов;
- удаление всех узлов с изоморфными потомками.

В большинстве своем под БДР понимают именно сокращенную упорядоченную бинарную диаграмму решений. Главное преимущество такой СУБДР в том, что она каноническая, точнее она является уникальной для конкретной функции и заданного порядка переменных. Это свойство делает СУБДР полезной для проверки функциональной эквивалентности.

На рисунке 1 изображено бинарное дерево принятия решений, но без применения правил сокращения, соответствующее приведенной на этом же рисунке таблице истинности для булевой функции $f(x_1, x_2, x_3)$. Для заданных входных значений x_1, x_2, x_3 можно определить значение булевой функции, если двигаться по дереву от корневого узла к терминальным узлам дерева, выбирая направление перехода из узла x_i , которое зависит от входного x_i .

Непрерывными линиями на рисунке изображены переходы к старшему потомку, а пунктирными линиями изображены переходы к младшему потомку. Например, если заданы входные значения $(x_1=0, x_2=1, x_3=1)$, то из корневого узла x_1 необходимо перейти влево по пунктирной линии, так как значение x_1 равно 0, далее необходимо перейти вправо по непрерывным линиям, так как значения x_2 и x_3 равны 1. В результате мы окажемся в 1-терминальном узле, то есть значение $f(x_1=0, x_2=1, x_3=1)$ равно 1.

Бинарное дерево принятия решений на рисунке 1 слева можно преобразовать в бинарную диаграмму решений, путем применения двух правил сокращения. Результирующая БДР изображена на рисунке 2.

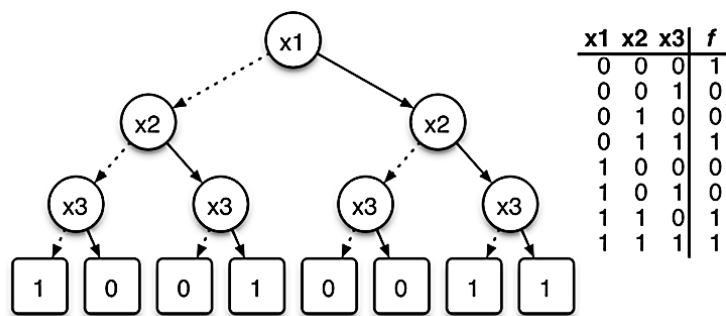


Рис. 1. Бинарное дерево принятия решений, построенное по таблице истинности для функции $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2\bar{x}_3 + x_1x_2 + x_2x_3$

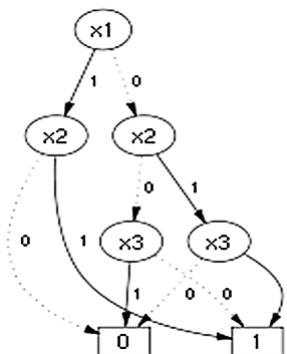


Рис. 2. Сокращенная БДР для функции f

Размер БДР определяется как выбором порядка входных переменных, так и булевой функцией. При составлении графа для булевой функции $f(x_1, \dots, x_n)$ количество узлов в лучшем случае может линейно зависеть от n , а в худшем случае зависимость может быть экспоненциальной при неудачном выборе порядка входных переменных.

Например, дана булева функция $f(x_1, \dots, x_n) = x_1x_2 + x_3x_4 + \dots + x_{2n-1}x_{2n}$. Если использовать порядок переменных $x_1 < x_3 < \dots < x_{2n-1} < x_2 < x_4 < \dots < x_{2n}$, то понадобится $2n+1$ узлов для представления функции в виде БДР.

Основной идеей для создания такой структуры данных послужило разложение Шеннона. Так как любую булеву функцию по одной из входных переменных можно разделить на две подфункции, называемых положительным и отрицательным дополнением, из которых по принципу *if-then-else* выбирается только одна подфункция в зависимости от значения входной переменной, по которой Шеннон выполнял свое разложение.

Представляя каждую такую подфункцию в виде поддерева и продолжая разложение по оставшимся входным переменным можно получить дерево принятия решений, сокращение которого даст *бинарную диаграмму решений*.

Данный способ позволяет быстро и легко написать программное обеспечение на языке высокого уровня, благодаря тому, что булеву функцию можно разложить на простые подфункции, одна из которых будет удовлетворять решению, так как подфункции находятся в зависимости от значения входных переменных.

Список литературы

1. Andersen H.R. «An Introduction to Binary Decision Diagrams», Lecture Notes, 1999, IT University of Copenhagen.
2. Binary Decision Diagrams: Theory and Implementation. – Springer, 1998.
3. Ch. Meinel, T. Theobald, «Algorithms and Data Structures in VLSI-Design: OBDD – Foundations and Applications», [Текст] Springer-Verlag, Berlin, Heidelberg, New York, 1998.
4. Knuth D. Fun With Binary Decision Diagrams (BDDs): видео лекция [Электронный ресурс]. – Режим доступа: <http://myvideos.stanford.edu/player/slplayer.aspx?coll=ea60314a-53b3-4be2-8552-dcf190caocob&co=18bcd3a8-965a-4a63-a516-a1ad74af1119&0=true>
5. Хамби Э. Программирование таблиц решений. – М.: Мир, 1976.
6. Бинарная диаграмма решений [Электронный ресурс]. – Режим доступа: <http://dic.academic.ru/dic.nsf/ruwiki/1696033>