

# ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ ПРОГРАММ МИНИМИЗАЦИИ, ВЫПОЛНЯЕМЫХ ПРИ СИНТЕЗЕ КОМБИНАЦИОННЫХ СХЕМ

Бибило П. Н., Кириенко Н. А., Романов В. И.

Объединённый институт проблем информатики Национальной академии наук Беларуси

Минск, Республика Беларусь

E-mail: {bibilo, kir, rom}@newman.bas-net.by

*Приведены результаты экспериментальных исследований эффективности программ минимизации многоуровневых алгебраических представлений систем булевых функций, выполняемых при синтезе комбинационных схем. Исследуется три подхода: 1) совместная минимизация систем булевых функций; 2) раздельная минимизация; 3) выделение из исходной системы «связанных» подсистем, каждая из которых минимизируется отдельно. После минимизации выполнялся синтез логических схем в библиотеке проектирования заказных цифровых КМОП СВИС. Полученные схемы сравнивались по площади кристалла и по быстродействию. Эксперименты показали, что для некоторых систем функций эффективнее при синтезе является совместная минимизация, для других – раздельная минимизация, среднюю позицию занимает оптимизационная процедура выделения «связанных» подсистем для совместной минимизации.*

## ВВЕДЕНИЕ

Минимизация двухуровневых либо многоуровневых представлений систем булевых функций является первым и важнейшим этапом, выполняемым при синтезе комбинационных схем. Двухуровневыми (И-ИЛИ) представлениями называют представления функций в виде дизъюнктивных нормальных форм (ДНФ), многоуровневыми – различные формы функциональных разложений. В качестве основных методов технологически независимой оптимизации применяются методы минимизации графовых BDD-представлений [1] и их модификаций, при этом внимание чаще уделяется совместной минимизации [1, 2] и декомпозиции [3]. Одинаковость частей в таблицах истинности булевых функций привела к идее использования связанности (общности) областей определений булевых функций при синтезе многовыходных комбинационных схем. Выделение связанных подсистем функций по таблицам истинности и ДНФ функций рассматривалось в [4], где были предложены различные меры связанности функций. В данной работе предложена новая мера связанности подсистем функций и алгоритм последовательного формирования связанных подсистем как для BDD-представлений, так и для булевых сетей.

### I. ВЫДЕЛЕНИЕ ПОДСИСТЕМ С ЗАДАННОЙ МЕРОЙ СВЯЗАННОСТИ

Предлагаемый эвристический алгоритм является модификацией предложенного в [4] алгоритма и состоит в последовательном формировании (на каждой итерации  $i$ ) по текущей (остаточной) системе функций очередной подсистемы  $P_i$  функций (подсистема  $P_i$  характеризуется мерой компонентной связанности, не меньшей  $q$ ). На первой итерации ( $i=1$ ) текущую систему функций образуют функции исходной системы. На каждой итерации требуется выполнить шаги 1-3.

Шаг 1. Рассмотреть неупорядоченную пару функций  $\{f_i, f_j\}$ ,  $i, j = 1, \dots, m, i \neq j$ , текущей системы и найти такую пару функций  $L$ , которая имеет максимальное значение меры связанности, и не меньше параметра  $q$ . Если указанной пары функций нет, то переход на шаг 4.

Шаг 2. Составить из функций найденной на первом шаге пары  $L$  подсистему  $P_i$ , и добавлять в нее поочередно те функции  $f_r$ , которые находятся с помощью следующей эвристики: из множества функций текущей системы выбирается та функция  $f_r$ , которая обеспечивает наибольшее возможное значение меры компонентной связанности для подсистемы  $P_i \cup \{f_r\}$ .

Шаг 3. Если нет ни одной функции  $f_r$  такой, что подсистема  $P_i \cup \{f_r\}$  имеет меру компонентной связанности не меньшую  $q$ , то закончить формирование подсистемы  $P_i$  и объявить не входящие в нее функции текущей системой. Переход на шаг 1 для формирования подсистемы на итерации  $i+1$ .

Шаг 4. Закончить формирование подсистем, когда все функции текущей системы будут включены в формируемые подсистемы, либо когда в текущей системе нельзя будет найти ни одной пары функций, характеризующих мерой связанности, не меньшей  $q$ , либо в текущей системе имеется только одна функция. Конец алгоритма.

Достоинством данного алгоритма является его высокое быстродействие.

### II. ОПИСАНИЕ ЭКСПЕРИМЕНТОВ

В экспериментах использовалась система FLC-2 [5]. Исходными описаниями блоков комбинационной логики являлись 39 примеров систем булевых функций на языке SF [5] в виде матричных описаний систем ДНФ – SDF описаний языка SF. Исходные описания могут быть представлены булевыми сетями (Bool-представлениями) [2]. Под Bool-представлениями в системе FLC-2 понимаются булевы сети, описываемые логически-

ми уравнениями с одним логическим оператором дизъюнкции либо конъюнкции над двумя литералами булевых переменных. Были использованы следующие программы:

BDD\_Builder – программа минимизации отдельных и совместных BDDI-представлений системы булевых функций;

BoolNetOpt2 – программа минимизации отдельных и совместных Bool-представлений системы булевых функций;

Splitter – программа выделения подсистем функций (при заданном значении  $q$  меры связанности) из совместных BDDI и из совместных Bool-представлений систем функций;

AutoSplit – программа нахождения лучшего значения  $q$  меры связанности по критерию минимальности общего числа литералов в BDDI- либо в Bool-представлениях систем функций.

Эксперимент 1. Совместная и отдельная минимизация в классе BDDI-представлений. При отдельной минимизации исходные матричные представления систем функций представлялись в виде логических сетей (формат BDDI). Для минимизации BDDI-представлений всей системы и отдельных функций использовалась программа BDD\_Builder.

Эксперимент 2. Исходными для эксперимента 2 являлись совместно минимизированные BDDI-представления, для которых выполнялась программа Splitter для значений меры связанности  $q = 10\%$ ,  $q = 20\%$ .

Эксперимент 3. Совместная и отдельная минимизация в классе булевых сетей (Bool-представлений). Базовой программой являлась программа BoolNetOpt2 [2].

Эксперимент 4. Исходными описаниями являлись совместно минимизированные Bool-представления, для которых выполнялась программа Splitter для значений меры связанности  $q = 5\%$ ,  $q = 20\%$ .

Эксперимент 5. Оценка выделения связанных подсистем. Для выделения совместно минимизируемых связанных подсистем с помощью программы AutoSplit ищутся лучшие значения параметра связанности  $q$  для BDDI-представлений путем итеративного выполнения программы Splitter для меры связанности  $q = 5\%$ ,  $10\%$ ,  $15\%$ ,  $20\%$ , ...,  $95\%$ , оценивается суммарное число литералов в каждой полученной логической сети и лучший вариант решения выбирается по минимуму найденной оценки.

Эксперимент 6. Совпадает с экспериментом 5, при этом исходными были совместно минимизированные Bool-представления, а результирующие

логические сети обрабатывались с помощью тех же стратегий, что и в эксперименте 3.

### III. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Результаты экспериментов 1 и 2 показывают, что предложенный алгоритм выделения подсистем в 13 примерах схем (из 39) позволил получить схемы меньшей площади для исходных совместных BDDI-представлений. Получение схем с меньшей задержкой чаще было обеспечено применением отдельной BDDI минимизации и алгоритмом выделения подсистем. При выделении подсистем из совместных Bool-представлений примерно в половине случаев лучшие решения были получены алгоритмом выделения подсистем. Критерий минимальности общего числа литералов в BDDI-представлениях оказался эффективным при автоматическом выборе лучшего решения для программы Splitter. Сравнение с «ручным» выбором лучшего значения меры связанности показало погрешность в 3,9% для BDDI-представлений и 6,4% для Bool-представлений.

### ЗАКЛЮЧЕНИЕ

Экспериментально установлено, что для некоторых систем функций эффективнее при синтезе логических схем является совместная минимизация, для других – отдельная минимизация, среднюю позицию занимает процедура выделения подсистем для совместной минимизации. Эффект достигается за счет того, что каждая выделенная подсистема минимизируется на основе разложения Шеннона по своей (для каждой подсистемы) перестановке переменных разложения.

1. Amaru, L. G. *New Data Structures and Algorithms for Logic Synthesis and Verification* / L. G. Amaru // Springer. – 2017. – 156 p.
2. Бибило, П. Н. Экспериментальное сравнение эффективности алгоритмов оптимизации BDD-представлений систем булевых функций / П. Н. Бибило, Ю. Ю. Ланкевич // Программные продукты и системы. – 2020. – Т. 33. – N 3. – С. 449-463.
3. Kubica, M. SMTBDD: New form of BDD for logic synthesis / M. Kubica, D. Kania // *International Journal of Electronics and Telecommunications*. – 2016. – V. 62. – № 1. – P. 33–41.
4. Бибило, П. Н. Выделение подсистем связанных функций из многоуровневого представления системы булевых функций / П. Н. Бибило, А. М. Позняк // Информатика. – 2020. – Т. 17. – N 1. – С. 63-77.
5. Бибило, П. Н. Система логической оптимизации функционально-структурных описаний цифровых устройств на основе производственно-фреймворк модели представления знаний / П. Н. Бибило, В. И. Романов // Проблемы разработки перспективных микро- и наноэлектронных систем. – 2020. – Сб. трудов / под общ. ред. акад. РАН А. Л. Стемповского. – М. : ИПИМ РАН. – 2020. – N 4. – С. 9.