# ВЕРИФИКАЦИЯ ЛОГИЧЕСКИХ ОПИСАНИЙ С ФУНКЦИОНАЛЬНОЙ НЕОПРЕДЕЛЕННОСТЬЮ

**Людмила Черемисинова, Дмитрий Новиков**

***Резюме**. Задача верификации, заключающаяся в доказательстве поведенческой эквивалентности двух описаний одного и того же устройства, рассматривается для случая, когда одно из них поведенчески не полностью определено. В такой постановке задача верификации сводится к проверке эквивалентности логических описаний на области задания того описания, которое определено не на всей области возможных значений аргументов. Рассматриваются пути решения этой задачи для разных форм задания описаний. Предлагаемые методы основаны на быстрых булевых вычислениях над булевыми и троичными векторами большой размерности.*

***Ключевые слова:** верификация, булевы вычисления, автоматизация проектирования, моделирование.*

## Введение

Задача верификации [1] заключается в доказательстве поведенческой эквивалентности двух описаний одного и того же устройства, представляющих разные проектные решения, полученные в ходе логического проектирования. Эта задача в литературе традиционно рассматриваются для случая, когда оба описания поведенчески полностью определены и представляют структурные реализации одного и того же устройства. Соответственно задача верификации сводится к проверке эквивалентности пары комбинационных схем. В настоящее время усилиями ведущих научных групп университетов и крупных фирм, занимающихся проектированием дискретных устройств, разработаны [1 – 4] и разрабатываются эффективные методы решения задачи верификации в такой постановке.

В настоящей работе задача верификации рассматривается для случая, когда одно из описаний поведенчески не полностью определено. Этот случай возникает на начальных этапах проектирования, когда проектируемое устройство частично определено, т. е. существуют наборы значений входных переменных, которые при штатном функционировании устройства никогда не появляются на входах устройства. Соответственно поведение такого устройства на этих наборах в процессе логического синтеза может быть доопределено произвольным образом. Исходное логическое описание верифицируемого комбинационного устройства с функциональной неопределенностью может быть представлено в виде системы частично определенных булевых функций.

Рассматривается случай, когда комбинационная структура, полученная в процессе декомпозиции, минимизации и/или синтеза, представляет собой многоблочную структуру, каждый блок которой описывается системой полностью определенных булевых функций. Задача проверки функциональной эквивалентности двух логических описаний трансформируется в этом случае в задачу проверки реализуемости одного логического описания с функциональной неопределенностью другим описанием, полностью определенным.

Задача верификации сводится к проверке эквивалентности логических описаний на области определения описания с неопределенностью, задающего систему частично определенных булевых функций. Рассматриваются пути решения этой задачи для разных форм задания системы частично определенных булевых функций: на наборах и интервалах значений аргументов. Предлагаются методы верификации, в

основе которых лежат быстрые булевы вычислениях над булевыми и троичными векторами большой размерности.

## Основные определения

В настоящей работе рассматривается случай, когда одно описание представляет собой систему $F = \{f_1(X), f_2(X), \ldots, f_m(X)\}$ ($X = \{x_1, x_2, \ldots, x_n\}$) частично определенных булевых функций, второе – многоуровневую многовыходную комбинационную схему $S$ в базисе элементов трех типов: инверторы, многоместные конъюнкторы и многоместные дизъюнкторы. К такому типу схем может быть преобразована многоуровневая схема из любой сложности элементов или блоков. Задача решается для двух основных видов задания системы $F$ частично определенных булевых функций:

1) на наборах значений входных переменных из $X$;

2) на интервалах в $n$-мерном булевом пространстве переменных из $X$.

В первом случае каждую частично определенную булеву функцию $f_i(X)$ можно задать парой множеств $M_{fi}^1$ и $M_{fi}^0$ наборов значений переменных, на которых она принимает значения 1 и 0 соответственно, предполагая, что на остальных наборах ее значение не определено. Система $F$ булевых функций представляется в этом случае парой матриц: булевой $\boldsymbol{B}$, задающей все наборы из $M = \bigcup_{i=1}^{m} (M_{fi}^1 \cup M_{fi}^0)$, и троичной $\boldsymbol{T}$, задающей значения функций на них – “1”, “0”, если они определены, и “–”, если не определены. Матрицы $\boldsymbol{B}$ и $\boldsymbol{T}$ имеют по $l$ строк (где $l$ – число наборов в множестве $M$) и по $n$ и $m$ столбцов соответственно.

Во втором случае каждая частично определенная булева функция $f_i(X)$ представляется парой множеств $U_{fi}^1$ и $U_{fi}^0$, но не наборов, а интервалов значений переменных, каждый из которых задает в общем случае более чем один набор. Точнее интервал ранга $k$ включает в себя $2^{n-k}$ наборов значений $n$ переменных, понимая под рангом интервала число его компонент, имеющих значение 0 или 1. Система $F$ представляется в этом случае парой троичных матриц: $\boldsymbol{U}$, задающей интервалы из $U = \bigcup_{i=1}^{m} (U_{fi}^1 \cup U_{fi}^0)$, и $\boldsymbol{T}$, задающей значения функций на них – “1”, “0”, если они определены, или “–”, в противном случае.

Задание функции на интервалах в отличие от задания на наборах имеет следующие особенности. Интервалы $u_i, u_j \in U$ могут пересекаться (наборы – нет). Значение “–” элемента $t_i^j$ матрицы $\boldsymbol{T}$ может означать, что 1) значение функции $f_j$ не определено на всем интервале $u_i$ (на всех входящих в него наборах); 2) функция $f_j$ не принимает одно и то же значение на всем интервале $u_i$: в этом интервале существует, как минимум, два набора, на которых она имеет разные значения из множества {1, 0, –}. Таким образом, значение “–” компоненты $t_i^j$ матрицы $\boldsymbol{T}$ не всегда означает, что значение функции $f_j$ не определено (в том смысле, что оно может быть при реализации доопределено произвольным образом) на всем интервале $u_i$, как при первом типе задания функций. Значение “–” говорит лишь о том, что функция $f_j$ не принимает одно и то же значение на всем интервале $u_i$ .

## Преобразование представления многоблочной структуры

Каждый блок многоблочной структуры $S$ является многовыходным и задается системой дизъюнктивных нормальных форм (ДНФ), представляемой парой матриц: троичной, строки которой задают элементарные конъюнкции, и булевой, в которой единичная компонента на пересечении $i$-й строки и $j$-го столбца равна 1, если $i$-я конъюнкция входит в $j$-ю ДНФ. Множество входных переменных структуры в целом совпадает с

множеством $X$, а $m$ реализуемых выходных функций $y_i(X)$, являются полностью определенными. Условие реализуемости системы $F$ структурой $S$ заключается в том, что для всех $f_i(X) \in F$ и соответствующих им функций $y_i(X)$ должно выполняться: $M_{fi}^1 \subseteq M_{yi}^1$ и $M_{fi}^0 \subseteq M_{yi}^0$, т. е. на области определения каждой функции $f_i(X)$ значения функций $f_i$ и $y_i$ должны совпадать.

Каждый блок структуры $S$ можно рассматривать как трехуровневую многовыходную логическую схему, первый уровень которой составляют инверторы, второй – многоместные конъюнкторы, а третий уровень – многоместные дизъюнкторы. Перенумеруем полюсы многоблочной структуры, начиная с ее входных полюсов, и далее выходных полюсов инверторов, конъюнкторов и дизъюнкторов всех блоков. Припишем им переменные $z_i$, выделив входные и выходные полюсы многоблочной структуры. В результате многоблочная структура может рассматриваться как многовыходная логическая схема $C$ из инверторов, конъюнкторов и дизъюнкторов, на выходных полюсах которой должны быть реализованы функции исходной системы частично определенных булевых функций и которая состоит из инверторов, дизъюнкторов и конъюнкторов. Предлагаемый ниже метод годится также и для сетей из любых других элементов, реализующих симметрические логические операции $\varphi(z_1, z_2, \ldots, z_k)$.

Ранжируем (пронумеруем) элементы схемы $C$ таким образом, чтобы любая межэлементная связь соединяла выходной полюс элемента с меньшим номером с входным полюсом элемента с большим номером. Необходимым и достаточным условием ранжируемости схемы является отсутствие в ней контуров. Это условие заведомо выполняется для структуры каждого блока, предполагается, что многоблочная структура, представляемая схемой $C$, в целом этому условию также удовлетворяет.

## Задача верификации для случая задания системы на наборах значений переменных

Идея предлагаемого метода проверки реализуемости системы частично определенных булевых функций многоблочной структурой $S$ состоит в моделировании поведения многовыходной логической схемы на области задания системы $F$. Условие реализуемости сводится к проверке для каждого набора значений переменных $\boldsymbol{b}_i \in \boldsymbol{B}$ условия неортогональности булева вектора значений $y_i(\boldsymbol{b}_i)$ выходных функций структуры $S$ троичному вектору $\boldsymbol{t}_i \in \boldsymbol{T}$.

Будем использовать идею двоичного параллельного моделирования [5], проводя моделирование многовыходной логической схемы $C$ сразу на всех входных наборах из множества $M$. При параллельном моделировании схемы на $l$ наборах состояние каждого полюса (включая входные и выходные) схемы представляется булевым вектором размерности $l$. Таким образом, каждый, вектор представляет состояния одного полюса для всех $l$ входных состояний схемы, а совокупность одноименных компонент всех векторов соответствует состоянию всех полюсов схемы для одного и того же входного набора.

В начале моделирования имеется упорядоченное множество $n$ булевых векторов размерности $l$, представляющих состояния $n$ входных полюсов во всех $l$ наборах и задаваемых столбцами матрицы $\boldsymbol{B}$. Затем последовательно просматриваются элементы (в порядке возрастания их номеров) предварительно ранжированной схемы $C$, реализующие функции $\varphi_i(z_{1i}, z_{2i}, \ldots, z_{ki})$, и выполняется функция $\varphi_i$ над ее аргументами $z_{1i}, z_{2i}, \ldots, z_{ki}$. Так как каждому из аргументов $z_{ji}$ функции соответствует булев вектор $\boldsymbol{z}_{ji}$, то операция сводится к выполнению покомпонентной операции $\varphi_i$ над булевыми векторами $\boldsymbol{z}_{1i}, \boldsymbol{z}_{2i}, \ldots, \boldsymbol{z}_{ki}$. Результатом операции является новый вектор $\boldsymbol{z}_i$, также размерности $l$.

После просмотра последнего элемента схемы будут найдены реакции схемы на все наборы значений входных переменных, входящие в область $M$ определения системы $F$. При этом каждая выходная функция $y_i$ схемы имеет определенное значение (0 или 1) на всех наборах значений входных переменных, в частности и на тех, на которых определенное значение имеет и соответствующая функция $f_i \in F$.

Остается только сравнить значения функций значения $y_i$ и $f_i$ на области $M_{fi}^1 \cup M_{fi}^0$ на ортогональность, это сводится к проверке, не ортогональны ли следующие пары векторов: троичный вектор $\boldsymbol{t}_i$, соответствующий *i*-му столбцу матрицы **T**, и булев вектор $\boldsymbol{z}_p$, соответствующий *i*-му выходному полюсу схемы. Реализуемость системы *F* многоблочной структурой *S* имеет место, если все эти пары векторов не ортогональны. В случае ортогональности некоторой пары можно путем обратного прослеживания логической схемы *C* найти причину, ответственную за нарушение условия реализуемости: определить блок структуры *S*, ее выход или ДНФ системы задающей, задающей его функциональное описание, и наконец, конъюнкции этой ДНФ.

Таким образом задача проверки реализуемости системы частично определенных булевых функций многоблочной структурой сводится к выполнению булевых вычислений над векторами (последовательностями бит) одной и той же (но произвольной) размерности.

## Задача верификации для случая задания системы на интервалах значений переменных

Возможны два пути решения задачи верификации системы частично определенных булевых функций, заданных на интервалах значений переменных: 1) сведение ее к рассмотренному выше случаю верификации системы частично определенных булевых функций путем перехода от задания функций на интервалах к заданию на наборах; 2) решение этой задачи на области, заданной множеством интервалов.

Первый путь решения целесообразно использовать в том случае, когда мало число интервалов множества *M*, имеющих ранг меньший, чем *n*, и эти ранги близки к *n*. В этом случае можно надеяться, что область задания системы частично определенных булевых функций не возрастет резко по размеру. Второй путь решения целесообразно использовать в том случае, когда число интервалов ранга, меньшего, чем *n*, велико и эти интервалы имеют ранг, значительно меньший, чем *n*. В этом случае при переходе к заданию функций на наборах значений переменных размер области задания системы может возрасти настолько, что задача верификации может стать практически не решаемой. В работе рассматривается решение задачи верификации системы частично определенных булевых функций, заданных на интервалах значений переменных.

Как и в случае задания системы функций на наборах значений аргументов будем проводить параллельное моделирование многовыходной логической схемы, но не наборах значений переменных, а на интервалах из множества *U*. При таком моделировании состояние каждого полюса схемы представляется не булевым, а в общем случае троичным вектором. Таким образом, каждый, вектор представляет состояния одного полюса для всех *l* входных состояний схемы, а совокупность одноименных компонент всех векторов соответствует состоянию всех полюсов схемы для одного и того же интервала значений входных переменных. При этом значение “–” *i*-й компоненты этого вектора говорит в общем случае лишь о том, что функция, реализуемая полюсом, имеет разные значения на разных наборах *i*-го интервала.

В начале моделирования имеется упорядоченное множество *n* интервалов *l*-мерного булева пространства, задаваемых столбцами матрицы $\boldsymbol{U}$. Последовательно просматриваются элементы предварительно ранжированной схемы *C*, реализующие функции $\varphi_i(z_{1i}, z_{2i}, \ldots, z_{ki})$, и выполняется функция $\varphi_i$ над ее аргументами $z_{1i}, z_{2i}, \ldots, z_{ki}$. Так как каждому из аргументов $z_{ji}$ функции соответствует интервал $\boldsymbol{z}_{ji}$, то операция сводится к выполнению покомпонентной операции $\varphi_i$ над троичными векторами $\boldsymbol{z}_{1i}, \boldsymbol{z}_{2i}, \ldots, \boldsymbol{z}_{ki}$. Результатом операции является новый троичный же вектор $\boldsymbol{z}_i$. Ниже приводится

определение основных операций над троичными переменными для случая принятой выше интерпретации неопределенного значения "–":

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| $a$: | 0 | 0 | 0 | – | – | – | 1 | 1 | 1 |
| $b$: | 0 | – | 1 | 0 | – | 1 | 0 | – | 1 |
| | | | | ---------- | | | | | |
| $\bar{a}$: | 1 | 1 | 1 | – | – | – | 0 | 0 | 0 |
| $a \vee b$: | 0 | – | 1 | – | – | 1 | 1 | 1 | 1 |
| $a \wedge b$: | 0 | 0 | 0 | 0 | – | – | 0 | – | 1 |

После окончания процесса моделирования проверяется, реализуется ли система $F$ функций многоблочной структурой $S$. При моделировании схемы $C$ на интервалах значений переменных, эта проверка может потребовать значительных затрат по времени. Сначала троичный вектор $t_i$ значений каждой функции $f_i \in F$ на всех интервалах задания системы $F$ сравнивается с троичным вектором $z_p$, соответствующим $i$-му выходному полюсу схемы. Возможны три случая.

1. Векторы $t_i$ и $z_p$ ортогональны по $j$-й компоненте. Делается вывод, что схема $C$ не реализует функцию $f_i$.

2. Вектор $t_i$ поглощает вектор $z_p$, т.е. все компоненты вектора $t_i$, значения которых отличны от "–", совпадают по значению с соответствующими компонентами вектора $z_p$. Делается вывод, что схема $C$ реализует функцию $f_i$.

3. Значение $j$-й компоненты вектора $z_p$ равно "–", тогда как значение этой компоненты в векторе $t_i$ равно 1 или 0. Невозможно дать однозначный ответ на вопрос реализует ли схема $C$ функцию $f_i$.

В третьем случае необходим дополнительный анализ, позволяющий найти причину несоответствия откликов выхода $y_p$ схемы и значения функции $f_p$ на наборах значений входных переменных из интервала $\boldsymbol{t}_i$. Самый простой метод заключается в повторном моделировании схемы $C$, но уже на наборах значений переменных из этого интервала. Более изощренный метод заключается в оперативном анализе результатов моделирования, проводимом в его процессе. Если в результате вычислений на некотором шаге получается неопределенное значение на некотором полюсе $z_q$ схемы $C$, то соответствующий интервал значений входных переменных дробится на подинтервалы, на которых $z_q$ принимает определенные значения. Это делается для того, чтобы предотвратить дальнейшее распространение неопределенности при моделировании. Например, если $z_q = x_1 x_3 x_4 x_6$, а вычисления ведутся на интервале $u_i$ = 1 0 – 1 – –, то $z_q(u_i)$ = –, то расщепляем интервал на три следующие: $u_i^1$ = 1 0 1 1 – 1, : $u_i^2$ = 1 0 0 1 – –, $u_i^3$ = 1 0 – 1 – 0, на элементах которого $z_q$ принимает одно и то же значение: $z_q(u_i^1) = 1$, $z_q(u_i^2) = z_q(u_i^3) = 0$.

Для того, чтобы избежать излишних расщеплений интервалов при моделировании исходную систему частично определенных булевых функций имеет смысл ортогонализовать. После ортогонализации каждый из интервалов задания функций системы будет обладать тем свойством, что все функции будут принимать одно и то же значение (1, 0, –) на всех его элементах. При этом значение "–" некоторой функции будет означать, что значение этой функции не определено на всем интервале.

## Ортогонализация системы частично определенных булевых функций

В работах [6, 7] задача ортогонализации ставится для системы полностью определенных булевых функций $F = \{f_1(X), f_2(X), \ldots, f_m(X)\}$, заданных на интервалах значений переменных из $X$ парой матриц: троичной, представляющей интервалы значений переменных и булевой, отмечающей единицами те интервалы, на которых функции принимают значение 1. Задача заключается в нахождении совокупности взаимно ортогональных полностью определенных булевых функций $\varphi_1(X)$, $\varphi_2(X)$, …, $\varphi_t(X)$ такой, что

любую функцию $f_i \in F$ можно выразить в виде дизъюнкции некоторых из ортогональных функций $\varphi_j$ ($j$ = 1, 2, …, $r$), причем общее число $r$ этих функций должно быть минимальным. Под ортогональностью функций $\varphi_j$ и $\varphi_k$ понимается выполнение условия $\varphi_j \wedge \varphi_k = 0$ при любых значениях аргументов из $X$. При этом, если функции $\varphi_j$ взаимно ортогональны, то интервалы, на которых определены разные функции не пересекаются, но интервалы, на которых определена одна и та же функция могут и пересекаться.

Аналогично системам полностью определенных булевых функций задача ортогонализации может быть поставлена и для системы частично определенных булевых функций $F = \{f_1(X), f_2(X), \ldots, f_m(X)\}$, если задавать каждую функцию $f_i \in F$ двумя функциями $f_i^1$ и $f_i^0$. В область единичных значений функции $f_i^1$ включаются те интервалы, на которых функция $f_i$ принимает значение 1, а в область единичных значений функции $f_i^0$ включаются те интервалы, на которых функция $f_i$ принимает значение 0. На остальной области булева пространства обе эти функции принимают значения 0. Таким образом от системы частично определенных булевых функций $F$ мы переходим к системе $F'$ полностью определенных функций, содержащей удвоенное число функций. Решаем задачу ортогонализации для системы $F'$ одним из известных методов [6, 7], а затем совершаем обратный переход к системе $F$ , но уже ортогонализованной.

Отличительной чертой ортогонализованной системы частично определенных булевых функций является то, что в ней интервалы, на которых хотя бы одна из функций системы принимает разные значения, не пересекаются. Это означает, что значение “–” функции на некотором интервале говорит о том, что эта функция не определена на всех наборах значений аргументов, входящих в интервал, и ее значение на этом интервале может быть при реализации доопределено произвольным образом.

## Заключение

Предложенные методы ориентированы на верификацию систем слабо определенных булевых функций – систем, для которых число наборов области их определения $M$ (на наборах которой значение хотя бы одной функции системы определено) существенно меньше числа наборов области их неопределенности, где значения всех функций не определены. Описанные методы годятся для верификации логических описаний большой размерности.

Задача проверки реализуемости системы частично определенных булевых функций многоблочной структурой сведена к булевым вычислениям над троичными и булевыми векторами произвольной размерности. Методы допускают эффективную программную реализацию. Для удобства программирования операций над булевыми и троичными векторами произвольной размерности в языке C++ ранее были разработаны классы CBV [8] и CTM [9].

Сложность $C(M)$ программной реализации описанных методов верификации линейно зависит от общего числа полюсов моделируемой схемы (суммарного числа входных полюсов всех элементов схемы $C$) и от числа байтов (или 32-разрядных слов), используемых для представления $l$-разрядного вектора (здесь $l$ – число интервалов или наборов значений аргументов области задания системы частично определенных булевых функций) [10].

## Библиография

Drechsler R. and others. Advanced Formal Verification. – Kluwer Academic Publishers, 2005. – 249 .

Mishchenko A., Chatterjee S., Brayton R., Een N. Improvements to Combinational Equivalence Checking // Proc. ICCAD’06, Nov. 5–9, 2006. – San Jose, CA, 2006.

Ganai M.K., Zhang L., Ashar P., Gupta A., Malik S. Combining strengths of circuit-based and CNF-based algorithms for a high-performance SAT solver // Proc. ACM/IEEE Design Automation Conference, 2002 – P. 747–750.

Goldberg E., Novikov Y. BerkMin: A fast and robust SAT-Solver // Proc. European Design and Test Conference, 2002. – P. 142–149.

Закревский А.Д., Поттосин Ю.В., Черемисинова Л.Д. Основы логического проектирования. Кн. 3. Проектирование устройств логического управления. – Мн.: ОИПИ НАН Беларуси, 2006. – 252 с.

Кузнецов О.П. Ортогональные системы булевых функций и их применение к анализу и синтезу логических сетей // Автоматика и телемеханика, 1970. – № 10. – С. 117–128.

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

Василькова И.В., Романов В.И. Булевы векторы и матрицы в C++ // Логическое проектирование. – Мн.: Ин-т техн. кибернетики НАН Беларуси, 1997. – С. 150–158.

Черемисинов Д.И., Черемисинова Л.Д. Троичные векторы и матрицы в C++ // Логическое проектирование. – Мн.: Ин-т техн. кибернетики НАН Беларуси, 1998. – Вып. 3. – С. 146–156.

Романов В.И. Оптимизация булевых вычислений на программном уровне // Танаевские чтения. Доклады Второй научной конференции (28 марта 2005 г., Минск). – Мн.: ОИПИ НАН Беларуси. 2005. – С. 91–93.

## Информация об авторах

**Людмила Дмитриевна Черемисинова** – д.т.н., главный научный сотрудник Объединенного института проблем информатики Национальной академии наук Беларуси, ул. Сурганова, 6, Минск, 220012, Беларусь, e-mail: cld@newman.bas-net.by

**Дмитрий Яковлевич Новиков** – аспирант Объединенного института проблем информатики Национальной академии наук Беларуси, ул. Сурганова, 6, Минск, 220012, Беларусь, e-mail: yakov_nov@tut.by