

ISSN 1816-0301 (Print)
ISSN 2617-6963 (Online)

ИНФОРМАТИКА

INFORMATICS

TOM 18
VOL. 18

2 | 2021

ОТ РЕДАКЦИИ

В журнале «Информатика» публикуются оригинальные и обзорные статьи, описывающие результаты фундаментальных и прикладных исследований специалистов академического и вузовского профиля в области информатики и информационных технологий.

Основной целью журнала является публикация наиболее значимых новых результатов в указанной области. Приветствуются статьи, описывающие заключительные результаты научных проектов и диссертационных исследований, открывающие новые направления исследований, которые находятся на стыке информатики и других наук.

Журнал рассчитан на широкий круг специалистов в области информатики и информационных технологий.

Основные разделы журнала:

- биоинформатика;
- математическое моделирование;
- защита информации и надежность систем;
- информационные технологии;
- логическое проектирование;
- обработка сигналов, изображений, речи, текста и распознавание образов;
- автоматизация проектирования;
- интеллектуальные системы.

Префикс DOI: 10.37661

Условия распространения материалов:

контент доступен под лицензией Creative Commons Attribution 4.0 License

Индексирование:

Высшей аттестационной комиссией Республики Беларусь журнал «Информатика» был включен в список научных изданий для опубликования результатов диссертационных исследований.

В декабре 2017 г. включен в базу данных Российского индекса научного цитирования (РИНЦ). С помощью инструментов и сервисов, доступных на платформе eLIBRARY (раздел «Личный кабинет»), можно самостоятельно корректировать список своих публикаций и цитирований в РИНЦ.

В июле 2017 г. включен в базу журналов открытого доступа Directory of Open Access Journals (DOAJ).

С помощью поисковых систем Google Scholar, WorldCat, Соционет можно получить свободный доступ к полному тексту научных публикаций журнала.

Адрес редакции:

ул. Сурганова, 6, к. 305, г. Минск, 220012, Беларусь
Тел. +375 (017) 351 26 22

Editorial address:

Surganova str., 6, of. 305, Minsk, 220012, Belarus
Phone +375 (017) 351 26 22

E-mail: rio@newman.bas-net.by

<https://inf.grid.by/jour>

THE EDITOR'S NOTE

The journal «Informatics» is a scientific publication in computer sciences and information technologies which reviews the results in basic and applied research of scientists from the universities and scientific centers.

The journal focuses on the most significant and modern papers of research projects results and PhD/DSc thesis in computer sciences.

The journal is edited for the specialists in IT and computer sciences research and application.

The main sections of the journal:

- bioinformatics;
- mathematical modeling;
- information protection and system reliability;
- information technology;
- logical design;
- signal, image, speech, text processing and pattern recognition;
- computer-aided design;
- artificial intelligence methods.

DOI Prefix: 10.37661

Distribution:

content is distributed under Creative Commons Attribution 4.0 License

Indexation:

the journal «Informatics» is in the list of scientific publications recommended by the Higher Attestation Commission of the Republic of Belarus for scientists to publish the results of PhD/DSc research.

In December 2017 the journal was included in the database of the Russian Science Citation Index (RISC) and provides free access to reviewed electronic scientific paper, improving scientific information traffic and also raising quotation of works of the authors (please use <https://elibrary.ru> or section https://elibrary.ru_author_tools).

In July 2017 included in the database of open access journals Directory of Open Access Journals (DOAJ).

Using the Google Scholar, WorldCat, Соционет search engine, you can get free access to full text of scientific publications of magazine.

ОБЪЕДИНЕННЫЙ ИНСТИТУТ ПРОБЛЕМ ИНФОРМАТИКИ
НАЦИОНАЛЬНОЙ АКАДЕМИИ НАУК БЕЛАРУСИ

ИНФОРМАТИКА

Том 18, № 2, апрель-июнь 2021

Ежеквартальный научный журнал

Издается с января 2004 г.

Учредитель и издатель – Объединенный институт проблем информатики
Национальной академии наук Беларуси (ОИПИ НАН Беларуси)

Г л а в н ы й р е д а к т о р

Тузиков Александр Васильевич, д-р физ.-мат. наук, проф., член-корр. НАН Беларуси,
генеральный директор ОИПИ НАН Беларуси (Минск, Беларусь)

З а м е с т и т е л ь г л а в н о г о р е д а к т о р а

Ковалев Михаил Яковлевич, д-р физ.-мат. наук, проф., член-корр. НАН Беларуси,
ОИПИ НАН Беларуси (Минск, Беларусь)

Р е д а к ц и о н н а я к о л л е г и я

Абламейко Сергей Владимирович, д-р техн. наук, проф., академик НАН Беларуси, БГУ (Минск, Беларусь)

Анищенко Владимир Викторович, канд. техн. наук, доцент, ООО «СофтКлуб» (Минск, Беларусь)

Бибило Петр Николаевич, д-р техн. наук, проф., ОИПИ НАН Беларуси (Минск, Беларусь)

Бобов Михаил Никитич, д-р техн. наук, проф., ОАО «АГАТ – системы управления» – управляющая
компания холдинга «Геоинформационные системы управления» (Минск, Беларусь)

Долгий Александр Борисович, д-р техн. наук, проф., Высшая инженерная школа Бретани (Нант, Франция)

Дудин Александр Николаевич, д-р физ.-мат. наук, проф., БГУ (Минск, Беларусь)

Карпов Алексей Анатольевич, д-р техн. наук, доцент, СПИИРАН (Санкт-Петербург, Россия)

Килин Сергей Яковлевич, д-р физ.-мат. наук, проф., академик НАН Беларуси, Президиум НАН
Беларуси (Минск, Беларусь)

Краснопрошин Виктор Владимирович, д-р техн. наук, проф., БГУ (Минск, Беларусь)

Крот Александр Михайлович, д-р техн. наук, проф., ОИПИ НАН Беларуси (Минск, Беларусь)

Кругликов Сергей Владимирович, д-р воен. наук, канд. техн. наук, доцент, ОАО «Гипросвязь»
(Минск, Беларусь)

Кундас Семен Петрович, д-р техн. наук, проф., БНТУ (Минск, Беларусь)

Лиходед Николай Александрович, д-р физ.-мат. наук, проф., БГУ (Минск, Беларусь)

Матус Петр Павлович, д-р физ.-мат. наук, проф., Институт математики НАН Беларуси (Минск, Беларусь)

Скляр Валерий Анатольевич, д-р техн. наук, проф., Университет Авейру (Авейру, Португалия)

Сотсков Юрий Назарович, д-р физ.-мат. наук, проф., ОИПИ НАН Беларуси (Минск, Беларусь)

Стемпковский Александр Леонидович, д-р техн. наук, проф., академик РАН, ИПИМ РАН (Москва, Россия)

Харин Юрий Семенович, д-р физ.-мат. наук, проф., член-корр. НАН Беларуси, НИИ ППМИ БГУ
(Минск, Беларусь)

Чернявский Александр Федорович, д-р техн. наук, проф., академик НАН Беларуси, НИИ ПФП
им. А. Н. Севченко БГУ (Минск, Беларусь)

Ярмолик Вячеслав Николаевич, д-р техн. наук, проф., БГУИР (Минск, Беларусь)

Редакционный совет

Ефанов Дмитрий Викторович, Российский университет транспорта (Московский институт инженеров транспорта) (Москва, Россия)

Кумари Мадху, Университетский центр исследований и разработок, Университет Чандигарха (Мохали, Пенджаб, Индия)

Лазарев Александр Алексеевич, Институт проблем управления им. В. А. Трапезникова РАН (Москва, Россия)

Лай Цунг-Чьян, Азиатский университет в Тайчжуне (Китайская Народная Республика, Тайвань)

Марина Нинослав, Университет информационных наук и технологий им. Св. апостола Павла (Охрид, Македония)

Меликян Вазген Шаваршович, Национальный политехнический университет Армении (Ереван, Армения)

Пеш Эрвин, Зигенский университет (Зиген, Германия)

Сингх Таджиндер, Институт инженерии и технологий Сант Лонговал (Лонговал, Пенджаб, Индия)

Ходаченко Максим Леонидович, Институт космических исследований Австрийской академии наук (Грац, Австрия)

Чиулла Карло, Университет Эпока (Тирана, Албания)

Штейнберг Борис Яковлевич, Институт математики, механики и компьютерных наук Южного федерального университета (Ростов-на-Дону, Россия)

ИНФОРМАТИКА

Том 18, № 2, апрель-июнь 2021

Ответственный за выпуск *Мойсейчик Светлана Сергеевна*

Редактор *Гончаренко Галина Борисовна*

Корректор *Михайлова Анна Антоновна*

Компьютерная верстка *Бутевич Ольга Борисовна*

Сдано в набор 24.05.2021. Подписано в печать 21.06.2021. Формат 60×84 1/8. Бумага офсетная. Гарнитура Таймс. Ризография. Усл. печ. л. 11,8. Уч.-изд. л. 11,6. Тираж 50 экз. Заказ 5.

Государственное научное учреждение «Объединенный институт проблем информатики
Национальной академии наук Беларуси».

Свидетельство о государственной регистрации издателя, изготовителя, распространителя печатных изданий № 1/274 от 04.04.2014. ЛП № 02330/444 от 18.12.13. Ул. Сурганова, 6, 220012, Минск, Беларусь.

ISSN 1816-0301 (Print)
ISSN 2617-6963 (Online)

THE UNITED INSTITUTE OF INFORMATICS PROBLEMS
OF THE NATIONAL ACADEMY OF SCIENCES OF BELARUS

INFORMATICS

Vol. 18, no. 2, April-June 2021

Published quarterly

Issued since January 2004

Founder and publisher – the United Institute of Informatics Problems
of the National Academy of Sciences of Belarus (UIIP NASB)

Editor-in-Chief

Alexander V. Tuzikov, Dr. Sci. (Phys.-Math.), Prof., Corr. Member of NASB, General Director
of the UIIP NASB (Minsk, Belarus)

Deputy Editor-in-Chief

Mikhail Y. Kovalyov, Dr. Sci. (Phys.-Math.), Prof., Corr. Member of NASB, UIIP NASB (Minsk, Belarus)

Editorial Board

Sergey V. Ablameyko, Dr. Sci. (Eng.), Prof., Academician of NASB, BSU (Minsk, Belarus)

Uladimir V. Anishchanka, Cand. Sci. (Eng.), Associate Professor, SoftClub Ltd. (Minsk, Belarus)

Petr N. Bibilo, Dr. Sci. (Eng.), Prof., UIIP NASB (Minsk, Belarus)

Mikhail N. Bobov, Dr. Sci. (Eng.), Prof., Open Joint-Stock Company "AGAT – Control Systems –
Managing Company of Geoinformation Control Systems Holding" (Minsk, Belarus)

Alexandre B. Dolgui, Dr. Sci. (Eng.), Prof., IMT Atlantique (Nantes, France)

Alexander N. Dudin, Dr. Sci. (Phys.-Math.), Prof., BSU (Minsk, Belarus)

Alexey A. Karpov, Dr. Sci. (Eng.), Associate Professor, SPII RAS (Saint Petersburg, Russia)

Sergey Ya. Kilin, Dr. Sci. (Phys.-Math.), Prof., Academician of NASB, Presidium of NASB (Minsk, Belarus)

Viktor V. Krasnoproshin, Dr. Sci. (Eng.), Prof., BSU (Minsk, Belarus)

Alexander M. Krot, Dr. Sci. (Eng.), Prof., UIIP NASB (Minsk, Belarus)

Sergey V. Kruglikov, Dr. Sci. (Milit.), Cand. Sci. (Eng.), Associate Professor, JSV "Giprosvjaz" (Minsk, Belarus)

Semen P. Kundas, Dr. Sci. (Eng.), Prof., BNTU (Minsk, Belarus)

Nikolai A. Likhoded, Dr. Sci. (Phys.-Math.), Prof., BSU (Minsk, Belarus)

Petr P. Matus, Dr. Sci. (Phys.-Math.), Prof., Institute of Mathematics of NASB (Minsk, Belarus)

Valery A. Sklyarov, Dr. Sci. (Eng.), Professor, University of Aveiro (Aveiro, Portugal)

Yuri N. Sotskov, Dr. Sci. (Phys.-Math.), Prof., UIIP NASB (Minsk, Belarus)

Alexander L. Stempkovsky, Dr. Sci. (Eng.), Prof., Academician of RAS, IPPM RAS (Moscow, Russia)

Yuriy S. Kharin, Dr. Sci. (Phys.-Math.), Prof., Corr. Member of NASB, RI APMI BSU (Minsk, Belarus)

Alexander F. Cherniavsky, Dr. Sci. (Eng.), Prof., Academician of NASB, A. N. Sevchenko IAPP BSU (Minsk, Belarus)

Vyacheslav N. Yarmolik, Dr. Sci. (Eng.), Prof., BSUIR (Minsk, Belarus)

Editorial Council

Dmitry V. Efanov, Russian University of Transport (Moscow Institute of Transport Engineers) (Moscow, Russia)

Madhu Kumari, University Center for Research & Development, Chandigarh University (Mohali, Punjab, India)

Alexander A. Lazarev, V. A. Trapeznikov Institute of Control Sciences of the RAS (Moscow, Russia)

Tsung-Chyan Lai, Asia University at Taichung (The People's Republic of China, Taiwan)

Ninoslav Marina, St. Paul the Apostle University of Information Sciences and Technology (Ohrid, Macedonia)

Vazgen Sh. Melikyan, National Polytechnic University of Armenia (Yerevan, Armenia)

Erwin Pesch, University of Siegen (Siegen, Germany)

Tajinder Singh, Sant Longowal Institute of Engineering & Technology (Longowal, Punjab, India)

Maxim L. Khodachenko, Space Research Institute, Austrian Academy of Sciences (Graz, Austria)

Carlo Ciulla, Epoka University (Tirana, Albania)

Boris Steinberg, Institute of Mathematics, Mechanics and Computer Science Southern Federal University (Rostov-on-Don, Russia)

INFORMATICS

Vol. 18, no. 2, April-June 2021

Issue Head *Sviatlana S. Maiseichyk*

Editor *Halina B. Hancharenka*

Corrector *Hanna A. Mikhailava*

Computer Imposition *Volha B. Butsevich*

Sent for press 24.05.2021. Output 21.06.2021. Format 60×84 1/8. Offset paper. Headset Times. Riesography. Printed sheets 11,8. Publisher's signatures 11,6. Circulation 50 copies. Order 5.

State Scientific Institution "The United Institute of Informatics Problems of the National Academy of Sciences of Belarus".

Certificate on the state registration of the publisher, manufacturer, distributor of printing editions

no. 1/274 dated 04.04.2014. License for the press no. 02330/444 dated 18.12.13.

6, Surganov Str., 220012, Minsk, Belarus.

ISSN 1816-0301 (Print)
ISSN 2617-6963 (Online)

СОДЕРЖАНИЕ

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

- Бибило П. Н., Романов В. И.** Минимизация многоуровневых представлений систем полностью определенных булевых функций с использованием разложений Шеннона и алгебраических представлений кофакторов..... 7
- Поттосин Ю. В.** Минимизация булевых функций в классе ортогональных дизъюнктивных нормальных форм 33

ЗАЩИТА ИНФОРМАЦИИ И НАДЕЖНОСТЬ СИСТЕМ

- Мартинovich Т. С., Парамонов Н. Н., Рымарчук А. Г., Чиж О. П.** Исследование надежности аппаратной части офисного кластера..... 48

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

- Савенко А. Г.** Интеллектуальный анализ качества учебного контента по статистике успеваемости студентов в системе управления обучением «Скорина» 58
- Тимофеев А. М.** Оценка влияния интенсивности оптического сигнала на вероятность ошибочной регистрации данных в однофотонном канале связи.....72

ОБРАБОТКА СИГНАЛОВ, ИЗОБРАЖЕНИЙ, РЕЧИ, ТЕКСТА И РАСПОЗНАВАНИЕ ОБРАЗОВ

- Жук Р. С.** Автоматическое обнаружение и отслеживание движущихся объектов, наблюдаемых видеокамерой беспилотного летательного аппарата 83

УЧЕНЫЕ БЕЛАРУСИ

- Киркоров Николай Иванович – талантливый разработчик, изобретатель и руководитель** 98
- О научной деятельности Николая Романовича Торопова** 100

ISSN 1816-0301 (Print)
ISSN 2617-6963 (Online)

CONTENTS

LOGICAL DESIGN

- Bibilo P. N., Romanov V. I.** Minimization of binary decision diagrams for systems of completely defined Boolean functions using Shannon expansions and algebraic representations of cofactors..... 7
- Pottosin Yu. V.** Minimization of Boolean functions in the class of orthogonal disjunctive normal forms 33

INFORMATION PROTECTION AND SYSTEM RELIABILITY

- Martinovich T. S., Paramonov N. N., Rymarchuk A. G., Tchij O. P.** The study of the reliability of the hardware part of the office cluster..... 48

INFORMATION TECHNOLOGIES

- Savenko A. G.** Intelligent analysis of the quality of educational content on statistics of student performance in the learning management system "Scorina" 58
- Timofeev A. M.** Assessment of the influence of intensity of optical signal on the probability of erroneous data registration in a single-photon communication channel..... 72

SIGNAL, IMAGE, SPEECH, TEXT PROCESSING AND PATTERN RECOGNITION

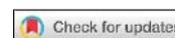
- Zhuk R. S.** Automatic detection and tracking the moving objects observed by an unmanned aerial vehicles video camera83

SCIENTISTS OF BELARUS

- Kirkorov Nikolai Ivanovich – talented developer, inventor and leader**..... 98
- About scientific activity of Nikolai Romanovich Toropov** 100

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

LOGICAL DESIGN



УДК 519.714
<https://doi.org/10.37661/1816-0301-2021-18-2-7-32>

Оригинальная статья
Original Paper

Минимизация многоуровневых представлений систем полностью определенных булевых функций с использованием разложений Шеннона и алгебраических представлений кофакторов

П. Н. Бибило[✉], В. И. Романов

*Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
[✉]E-mail: bibilo@newman.bas-net.by*

Аннотация. В системах проектирования цифровых СБИС (сверхбольших интегральных схем) графовый аппарат BDD (Binary Decision Diagram – бинарная диаграмма решений) применяется при верификации СБИС, а также при технологически независимой оптимизации, выполняемой как первый этап синтеза логических схем в различных технологических базах. BDD представляет собой ациклический граф, задающий булеву функцию либо систему булевых функций. Каждой вершине этого графа соответствует полная или редуцированная формула разложения Шеннона. После получения BDD-представлений систем булевых функций предлагается выполнять дополнительные логические оптимизации на основе описываемого в статье метода поиска алгебраических представлений кофакторов (подфункций разложения Шеннона) одного уровня BDD в виде дизъюнкции, конъюнкции либо суммы по модулю два подфункций того же уровня либо нижних уровней BDD. Ориентированный граф BDD для системы функций строится на основе разложений Шеннона всех компонентных функций системы по одной и той же переменной. Метод позволяет уменьшать число литералов путем замены формул разложений Шеннона более простыми логическими формулами и сокращать число литералов в описании системы булевых функций. Число литералов в алгебраических многоуровневых представлениях систем полностью определенных булевых функций является основным критерием логической оптимизации при синтезе комбинационных схем из библиотечных логических элементов.

Ключевые слова: система булевых функций, дизъюнктивная нормальная форма, Binary Decision Diagram, разложение Шеннона, синтез логической схемы, СБИС

Для цитирования. Бибило, П. Н. Минимизация многоуровневых представлений систем полностью определенных булевых функций с использованием разложений Шеннона и алгебраических представлений кофакторов / П. Н. Бибило, В. И. Романов // Информатика. – 2021. – Т. 18, № 2. – С. 7–32. <https://doi.org/10.37661/1816-0301-2021-18-2-7-32>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 25.01.2021
Подписана в печать | Accepted 21.04.2021
Опубликована | Published 26.06.2021

Minimization of binary decision diagrams for systems of completely defined Boolean functions using Shannon expansions and algebraic representations of cofactors

Petr N. Bibilo[✉], Vladimir I. Romanov

*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus
✉E-mail: bibilo@newman.bas-net.by*

Abstract. In the systems of digital VLSI design (Very Large Integrated Circuits), the BDD (Binary Decision Diagram) is used for VLSI verification, as well as for technologically independent optimization as the first stage in the synthesis of logic circuits in various technological bases. The BDD is an acyclic graph defining a Boolean function or a system of Boolean functions. Each vertex of this graph corresponds to the complete or reduced Shannon expansion formula. When BDD representation for systems of Boolean functions is constructed, it is possible to perform additional logical optimization based on the proposed method of searching for algebraic representations of cofactors (subfunctions) of the same BDD level in the form of a disjunction, conjunction either exclusive-or of cofactors of the same level or lower levels of BDD. A directed BDD graph for a system of functions is constructed on the basis of Shannon expansion of all component functions of the system by the same permutation of variables. The method allows to reduce the number of literals by replacing the Shannon expansion formulas with simpler formulas that are disjunctions or conjunctions of cofactors, and to reduce the number of literals in specifying a system of Boolean functions. The number of literals in algebraic multilevel representations of systems of fully defined Boolean functions is the main optimization criterion in the synthesis of combinational circuits from librarian logic elements.

Keywords: system of Boolean functions, disjunctive normal form, Binary Decision Diagram, Shannon expansion, digital logic synthesis, VLSI

For citation. Bibilo P. N., Romanov V. I. Minimization of binary decision diagrams for systems of completely defined Boolean functions using Shannon expansions and algebraic representations of cofactors. *Informatika*, 2021, vol. 18, no. 2, pp. 7–32 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-7-32>

Conflict of interest. The authors declare of no conflict of interest.

Введение. Минимизация различных форм представления систем булевых функций выполняется в качестве начального этапа синтеза многовыходных комбинационных схем. Оптимизация двухуровневых (И-ИЛИ) представлений систем булевых функций сводится к различным способам совместной либо раздельной минимизации функций в классе ДНФ (дизъюнктивных нормальных форм) [1] и эффективно применяется при синтезе схем программируемых логических матриц [2]. Минимизация многоуровневых представлений обычно выполняется как этап технологически независимой оптимизации при синтезе схем из библиотечных логических элементов [3] и структур FPGA (Field-Programmable Gate Array – программируемая пользователем вентильная матрица). При синтезе логических схем в составе FPGA также используются библиотеки параметризованных логических элементов, состоящие из сотен логических ячеек с различным числом входов, триггеров, мультиплексоров и универсальных логических элементов, настраиваемых на реализацию булевых функций от ограниченного числа $k = 4, 6$ переменных [4]. Развитие методов многоуровневой оптимизации началось с алгебраической факторизации – выделения общих частей в алгебраических представлениях функций [5–9]. Такие представления часто задавались в виде булевых сетей. В булевых сетях (графах) используются функциональные описания вершин (узлов), узлы при покрытии (технологическом отображении – втором этапе синтеза логической схемы) объединяются в подсхемы, а каждая подсхема реализуется одним библиотечным логическим элементом. Методы факторизации были реализованы в промышленных программах [5, 7] автоматизации проектирования логических схем в различных библиотеках. Нахождение многоуровневых представлений осуществляется на основе ре-

шения различных задач декомпозиции (разложения) систем функций [10–14]. Среди многочисленных видов функциональных разложений и соответствующих им форм представлений систем функций, пожалуй, наиболее эффективными для технологически независимой оптимизации оказались многоуровневые представления систем функций на основе разложения Шеннона, называемые в литературе BDD-представлениями [15–21]. Широкое применение BDD – ориентированных графов, задающих формулы разложений Шеннона, основывается на компактности графов и эффективности вычислительных процедур над такими графами [20]. Эффективность вычислений обусловлена тем обстоятельством, что BDD-представления булевых функций соответствуют ортогонализированным формам задания и исходных функций, и всех подфункций (кофакторов) разложений Шеннона.

В настоящей статье предлагается использовать алгебраические представления кофакторов BDD в виде дизъюнкции, конъюнкции либо исключающего ИЛИ (суммы по модулю два) других кофакторов той же BDD. При этом применяется инверсирование кофакторов и учитывается сложность самих кофакторов, участвующих в алгебраических представлениях. Предложенный в статье метод основывается на вычислениях и сравнениях функций кофакторов и позволяет улучшать результаты логической минимизации по числу литералов булевых переменных – основному критерию технологически независимой оптимизации [3] при синтезе комбинационных схем из библиотечных логических элементов. Уменьшение сложности многоуровневых представлений систем функций приводит к меньшим по площади кристаллам заказных цифровых СБИС и цифровых логических блоков, размещаемых в составе FPGA.

Основные определения. Булевыми называются двоичные $(0, 1)$ функции $f(\mathbf{x}) = f(x_1, x_2, \dots, x_n)$ двоичных (булевых) переменных x_1, x_2, \dots, x_n , литералом булевой переменной – переменная x_i либо ее инверсия \bar{x}_i . Пусть $V^{\mathbf{x}}$ – булево пространство, построенное над переменными булева вектора $\mathbf{x} = (x_1, x_2, \dots, x_n)$. Элементами этого пространства являются n -компонентные наборы (векторы) \mathbf{x}^* нулей и единиц. Булева функция, значения 0, 1 которой определены на всех элементах $\mathbf{x}^* \in V^{\mathbf{x}}$, называется *полностью определенной булевой функцией*. Полностью определенная булева функция (далее – булева функция) принимает единичное значение на элементах \mathbf{x}^* подмножества M_f^1 булева пространства $V^{\mathbf{x}}$ и нулевое значение на элементах подмножества M_f^0 . Очевидно, что $M_f^1 \cap M_f^0 = \emptyset$, $V^{\mathbf{x}} = M_f^0 \cup M_f^1$. Булевы функции $f_1(\mathbf{x})$ и $f_2(\mathbf{x})$ равны, если и только если $M_{f_1}^1 = M_{f_2}^1$, $M_{f_1}^0 = M_{f_2}^0$. Будем называть булевы функции $f_1(\mathbf{x})$ и $f_2(\mathbf{x})$ *взаимно инверсными* ($f_2 = \bar{f}_1$, $f_1 = \bar{f}_2$), если и только если $M_{f_1}^1 = M_{f_2}^0$, $M_{f_1}^0 = M_{f_2}^1$. Будем говорить, что рассматривается *инверсная форма* булевой функции f , если рассматривается инверсия \bar{f} этой функции, и *прямая форма* функции f , если речь идет о f . Под *векторной булевой функцией* $\mathbf{f}(\mathbf{x})$ будем понимать упорядоченную систему булевых функций $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$, значениями векторных функций на элементах \mathbf{x}^* булева пространства являются m -компонентные двоичные векторы $\mathbf{f}(\mathbf{x}^*)$.

Разложением Шеннона полностью определенной булевой функции $f = f(\mathbf{x})$, $\mathbf{x} = (x_1, x_2, \dots, x_n)$, по переменной x_i называется представление

$$f = f(\mathbf{x}) = \bar{x}_i f_0(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \vee x_i f_1(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n). \quad (1)$$

Функции $f_0 = f_0(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$, $f_1 = f_1(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$ называются *кофакторами* (cofactors) разложения по переменной x_i либо *подфункциями*. Они получаются из функции $f = f(x_1, \dots, x_n)$ подстановкой вместо переменной x_i констант 0 и 1. Каждая из под-

функций f_0 и f_1 может быть разложена по одной из переменных из множества $x_1, \dots, \{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n\}$. Процесс разложения подфункций заканчивается, когда все n переменных будут использованы для разложения либо когда все подфункции вырождаются до констант 0, 1. На каждом шаге разложения выполняется сравнение на равенство полученных подфункций и оставляется одна из нескольких попарно равных подфункций.

Под *BDD-представлением* полностью определенной векторной булевой функции $f(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$ понимается ациклический ориентированный граф, задающий последовательные разложения Шеннона всех компонентных функций $f_i(\mathbf{x})$, $i = 1, \dots, m$, по всем ее переменным x_1, x_2, \dots, x_n при одном и том же заданном порядке (перестановке) переменных, по которым проводятся разложения [21]. Для упрощения графа листовые вершины 0, 1 обычно дублируются, а ориентация дуг на изображениях графов BDD не показывается, так как всегда принимается, что дуги ориентированы сверху вниз.

Наиболее близкими к графам BDD для векторных функций являются широко известные в литературе сокращенные упорядоченные BDD (англ. Reduced Ordered BDD, ROBDD) для одной полностью определенной булевой функции, в которых каждой функциональной вершине соответствует одна функция (подфункция разложения Шеннона), при этом функциональные вершины лишь подразумеваются (отождествляются с вершинами-переменными). Подробное описание OBDD (упорядоченных BDD) дано в работах [18, 19], ROBDD – в [20]. Далее под BDD будут пониматься ROBDD для систем функций (векторных функций) [21, с. 40]. BDD векторных функций иногда называются в литературе совместными BDD. Будем изображать на рисунках графы BDD, которые содержат функциональные вершины, соответствующие разлагаемым функциям, вершины-переменные и листовые вершины, соответствующие константам 0, 1 [21, с. 16]. BDD-представлению соответствуют формулы разложения Шеннона, каждой функциональной вершине – своя полная либо сокращенная формула разложения Шеннона. Функциональные вершины обычно показываются на рисунках для удобства изложения методов минимизации BDD и иллюстрации зависимостей переменных в формулах разложений Шеннона. Рисунки классических BDD не содержат функциональных вершин. BDD нашли очень широкое применение не только при проектировании логических схем, но и в информатике в целом [20].

Пример BDD-представления векторной булевой функции. Рассмотрим векторную булеву функцию, задаваемую системой ДНФ (табл. 1). Эта функция может быть представлена BDD (рис. 1), построенной по перестановке переменных $\langle x_1, x_2, x_5, x_3, x_4 \rangle$. Функциональным вершинам такой BDD соответствуют следующие формулы разложения Шеннона:

$$\begin{aligned} f_1 &= \bar{x}_1 h_1 \vee x_1 h_2; & f_2 &= \bar{x}_1 h_2 \vee x_1 h_3; & f_3 &= \bar{x}_1 h_4 \vee x_1 h_5; & f_4 &= \bar{x}_1 h_4 \vee x_1 h_6; & f_5 &= \bar{x}_1 h_6 \vee x_1 h_7; \\ h_1 &= \bar{x}_2 t_1 \vee x_2 t_3; & h_2 &= \bar{x}_2 t_2 \vee x_2 t_8; & h_3 &= \bar{x}_2 t_5 \vee x_2 t_6; & h_4 &= \bar{x}_2 t_6 \vee x_2 t_7; & h_5 &= \bar{x}_2 t_6 \vee x_2 t_{10}; \\ h_6 &= \bar{x}_2 t_4 \vee x_2 t_9; & h_7 &= \bar{x}_2 t_{11} \vee x_2 t_{10}; & t_1 &= \bar{x}_5 t_{15} \vee x_5 t_{16}; & t_2 &= \bar{x}_5 \bar{x}_4 \vee x_5 t_{12}; & t_3 &= \bar{x}_5 t_{13} \vee x_5 t_{14}; \\ t_4 &= \bar{x}_5 t_{12} \vee x_5 \bar{x}_4; & t_5 &= \bar{x}_5 t_{13} \vee x_5 \bar{x}_4; & t_6 &= \bar{x}_5 \bar{x}_3 \vee x_5 t_{14}; & t_7 &= \bar{x}_5 x_4 \vee x_5 \bar{x}_4; & t_8 &= \bar{x}_5 \bar{x}_4 \vee x_5 \bar{x}_3; \\ t_9 &= \bar{x}_5 t_{13}; & t_{10} &= \bar{x}_5 t_{17} \vee x_5 t_{15}; & t_{11} &= \bar{x}_5 x_4 \vee x_5 x_3; & t_{12} &= \bar{x}_3 x_4; & t_{13} &= \bar{x}_3 \vee x_3 x_4; \\ t_{14} &= \bar{x}_3 \bar{x}_4 \vee x_3; & t_{15} &= \bar{x}_3 x_4 \vee x_3; & t_{16} &= \bar{x}_3 \vee x_3 \bar{x}_4; & t_{17} &= x_3 \bar{x}_4. \end{aligned} \quad (2)$$

Корневыми вершинами BDD (рис. 1) являются f_1, \dots, f_5 , листовыми – 0, 1. На первом уровне BDD располагаются два неконстантных кофактора x_4, \bar{x}_4 и два кофактора, равные константам 0, 1; на втором уровне – кофакторы $t_{12}, \dots, t_{17}, x_3, \bar{x}_3$ и константа ноль; на третьем уровне – кофакторы t_1, \dots, t_{11} ; на четвертом – кофакторы h_1, \dots, h_7 . Многоуровневое описа-

ние (2) содержит 106 литералов булевых переменных. Уравнения кофакторов могут содержать различное число литералов. В частности, кофактор t_1 содержит четыре литерала, t_{15} – три, t_{17} – два литерала. Если перейти к алгебраическому символьному заданию системы ДНФ, то можно получить число (162) литералов в системе ДНФ, заданной в табл. 1. При этом каждый троичный вектор в левой части таблицы задает элементарную конъюнкцию переменных x_1, \dots, x_5 , а единицы в правой части табл. 1 отмечают вхождения элементарных конъюнкций в ДНФ компонентных функций f_1, \dots, f_5 . Например, последнему троичному вектору (1110–) соответствует элементарная конъюнкция $x_1x_2x_3\bar{x}_4$, входящая в ДНФ функций f_3 и f_5 . Эта конъюнкция внесет восемь литералов в их общее число при оценке сложности системы ДНФ по числу литералов.

Таблица 1
Матричное представление системы ДНФ векторной функции

$x_1 x_2 x_3 x_4 x_5$	$f_1 f_2 f_3 f_4 f_5$
1--00	10000
110-1	10000
-0011	10000
010-0	10001
0111-	10000
011-1	10000
0001-	10000
000-1	10000
0010-	10000
001-0	10000
-100-	10000
0--00	01000
010-1	01000
0-011	01000
1--01	01000
110-0	01010
111-1	01000
1000-	01100
1-1-1	00101
-1-10	00010
-0-01	00010
-0010	00111
0000-	00110
001-1	00110
0--01	00110
01-10	00101
00-01	00001
10-10	01000
1011-	00001
11-11	00101
1110-	00101

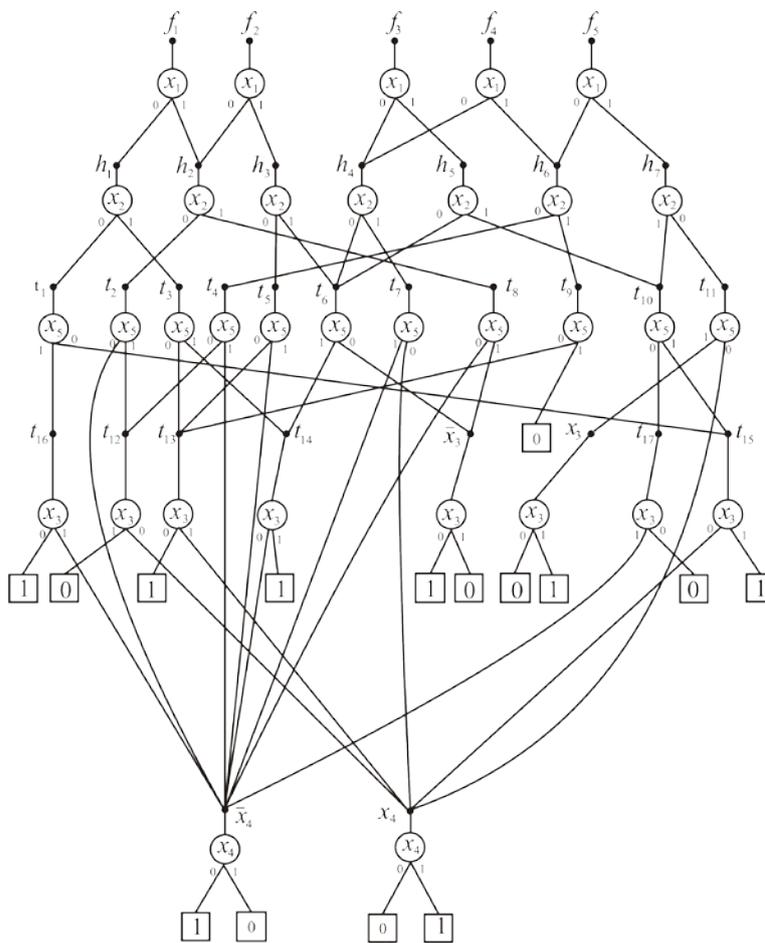


Рис. 1. BDD для векторной булевой функции (табл. 1)

Метод минимизации многоуровневого представления векторной булевой функции. Исходными данными для предлагаемого метода являются граф BDD, представляющий исходную векторную полностью определенную булеву функцию, и логические уравнения (формулы разложения Шеннона), соответствующие функциональным вершинам BDD и задающие многоуровневое описание компонентных функций. Данные уравнения легко могут быть записаны по графу BDD.

Предлагаемый метод включает этапы, выполняемые для кофакторов каждого уровня BDD, за исключением корневого, листового и первого. На первом уровне BDD располагаются кофакторы, получаемые в результате разложения по последней переменной в заданной перестав-

новке переменных, по которым ведется разложение Шеннона. Кофакторы первого уровня BDD зависят от одной переменной, кофакторы второго уровня – не более чем от двух переменных и т. д.

В методе уровни BDD рассматриваются сверху вниз, т. е. начиная от кофакторов, получаемых при разложении по первой переменной, и заканчивая кофакторами второго уровня BDD. Метод ориентирован на замену наибольшего числа формул разложения Шеннона формулами дизъюнкции, конъюнкции либо суммы по модулю два, что позволяет сокращать общее число литералов в многоуровневом задании векторной булевой функции. Для рассматриваемого уровня BDD решается задача нахождения наибольшего числа кофакторов, которые представимы в алгебраическом виде с использованием других кофакторов данного уровня. Для этого осуществляется переход на функциональный уровень представления кофакторов. С целью сокращения общего числа литералов в искомом многоуровневом представлении исходной системы булевых функций выбирается форма (прямая либо инверсная) реализации кофакторов и новая перестановка переменных, по которой строится граф BDD для кофакторов, не представленных в виде двухоперандных логических выражений (уравнений). Для рассматриваемого уровня BDD выполняются следующие восемь этапов.

Этап 1. Получение функций кофакторов, зависящих от входных переменных. Нахождение взаимно инверсных кофакторов. Выбирается форма представления кофакторов рассматриваемого уровня BDD (таблица истинности; пара ДНФ, задающая области определения кофактора) и находятся кофакторы как в прямой, так и инверсной форме. Находятся пары взаимно инверсных кофакторов.

Этап 2. Выбор формы реализации кофакторов. Для каждого кофактора осуществляется выбор, в какой форме (прямой либо инверсной) его реализовать в BDD (*задача 1*). Строится новый подграф нижних уровней BDD для системы кофакторов по новой перестановке переменных, если найдены пары взаимно инверсных кофакторов.

Этап 3. Нахождение всех вариантов алгебраических представлений кофакторов с использованием их инверсий. Находится множество уравнений, задающих всевозможные алгебраические представления кофакторов, т. е. кофакторы данного уровня, содержащие более двух литералов, выражаются в виде дизъюнкции, конъюнкции либо суммы по модулю два других кофакторов этого же уровня BDD. Кофакторы, которые могут быть выражены в виде уравнения, называются *представимыми*. При составлении уравнений для представимых кофакторов используются прямые и инверсные формы кофакторов, участвующих в уравнениях.

Этап 4. Нахождение множества реализуемых кофакторов. Из множества представимых кофакторов выделяется максимальное по мощности множество *реализуемых кофакторов*. Для этого из множества всех уравнений алгебраического представления одного и того же кофактора выбирается одно уравнение либо оно вообще не выбирается. Чтобы решить проблему выбора, строится ориентированный граф G зависимости кофакторов и выполняется редукция циклического графа G к ациклическому подграфу, если граф G не является ациклическим (*задача 2*). Решение задачи 2 позволяет выбрать из множества всех уравнений реализуемые уравнения, которые будут записаны позже в результирующее оптимизированное многоуровневое представление исходной векторной функции. По оптимизированному представлению в дальнейшем проектировании осуществляется синтез логической схемы в той или иной технологической (целевой) библиотеке.

Этап 5. Построение BDD для кофакторов, которые не являются реализуемыми. Для нереализуемых кофакторов строится подграф BDD по новой перестановке переменных рассматриваемого уровня BDD. Новая перестановка переменных ориентирована на уменьшение числа вершин BDD и, следовательно, числа формул разложения Шеннона, соответствующих нереализуемым кофакторам.

Этап 6. Составление уравнений для нереализуемых кофакторов с использованием кофакторов нижнего уровня BDD. Данный этап выполняется для BDD, построенной на этапе 5, с целью увеличения числа реализуемых кофакторов, т. е. числа алгебраических уравнений, полученных на этапе 4. Составляются уравнения для нереализованных кофакторов рассматриваемого

мого уровня BDD с использованием кофакторов нижнего уровня. Для этого множество переменных каждого кофактора нижнего уровня расширяется – добавляется несущественная переменная, по которой проводилось разложение Шеннона на рассматриваемом уровне BDD.

Этап 7. Расширение ациклического подграфа по новым уравнениям. Если на этапе 6 получено большое число (более 100) уравнений, то выполняется последовательная проверка на отсутствие циклов поочередно добавляемых в граф вершины и пары дуг, соответствующих одному уравнению. Если цикл не образуется, то это означает, что кофактор, находящийся в левой части добавляемого уравнения, является реализуемым. Если на этапе 6 получено небольшое число (менее 100) уравнений, то по ним в ациклический подграф, полученный на этапе 4, добавляются все соответствующие вершины и дуги и решается задача 2, т. е. осуществляется редукция расширенного подграфа к результирующему ациклическому подграфу, по которому определяется результирующее множество реализуемых кофакторов.

Этап 8. Корректировка многоуровневого BDD-представления векторной функции. Исходное множество формул разложений Шеннона, задающих BDD-представление (т. е. векторную булеву функцию), корректируется: формулы разложения Шеннона реализуемых кофакторов заменяются новыми формулами дизъюнкции, конъюнкции либо суммы по модулю два. Используются операции инверсии для записи формул одного из кофакторов, составляющих пару взаимно инверсных кофакторов. Здесь следует отметить, что алгоритмы, реализующие предлагаемый метод, могут начинать работу с произвольно выбранного уровня BDD.

Проиллюстрируем на примере BDD (см. рис. 1) предложенный метод, начиная с третьего уровня BDD, кофакторы t_1, \dots, t_{11} которого зависят от трех переменных x_3, x_4, x_5 .

Нахождение алгебраических представлений кофакторов на третьем уровне BDD осуществляется в восемь этапов.

Этап 1. Получение прямой и инверсной формы представления кофакторов. Нахождение взаимно инверсных кофакторов. Из BDD можно получить области определения как исходных функций, так и любого из кофакторов в виде ортогонализированных ДНФ (см. подробное описание в работе [21]). В ортогонализированной ДНФ каждая пара элементарных конъюнкций является ортогональной (их произведение равно нулю). Каждому пути из корневой вершины BDD, помеченной функцией f_j , к листовой вершине 1 соответствует элементарная конъюнкция, включающая дуги (литералы) x_i, \bar{x}_i на данном пути. При этом дуге, помеченной символом 0, соответствует отрицательный литерал \bar{x}_i ; дуге, помеченной символом 1, – положительный литерал x_i , а всем путям между указанными вершинами – дизъюнкция получаемых элементарных конъюнкций, образующая ортогонализированную ДНФ $D_{f_j}^1$. Пути из корневой вершины f_j к листовой вершине 0 задают ортогонализированную ДНФ $D_{f_j}^0$. При построении ДНФ, описывающих области определения кофактора, вместо корневой вершины BDD рассматривается функциональная вершина данного кофактора. Например, области нулевых и единичных значений кофакторов t_8, t_{11} , полученные из графа BDD на рис. 1, задаются ортогонализированными ДНФ $D_{t_8}^0 = x_4 \bar{x}_5 \vee x_3 x_5$, $D_{t_8}^1 = \bar{x}_4 \bar{x}_5 \vee \bar{x}_3 x_5$, $D_{t_8}^0 = D_{t_{11}}^1$, $D_{t_8}^1 = D_{t_{11}}^0$. Матричные формы данных ДНФ заданы в табл. 2. Видно, что кофакторы являются взаимно инверсными: $t_8 = \bar{t}_{11}$, $t_{11} = \bar{t}_8$.

В общем случае проверка равенства $t_i = t_j$ сводится к проверке одновременного выполнения двух условий: $D_{t_i}^0 \bar{D}_{t_j}^0 = 0$, $D_{t_i}^1 \bar{D}_{t_j}^1 = 0$. Алгоритмы перемножения и инверсирования матричных форм ДНФ описаны в работе [22 с. 34], структуры данных – в [23]. В работе [24] приведены алгоритмы и примеры перехода от ДНФ к полиномам Жегалкина, для которых легко выполняются логические операции инверсирования и перемножения булевых функций. Проверка условий $D_{t_i}^0 \bar{D}_{t_j}^0 = 0$, $D_{t_i}^1 \bar{D}_{t_j}^1 = 0$ может быть также сведена к задаче «выполнимость» [25, 26], если

перевести ДНФ в конъюнктивную нормальную форму. Переход от задания булевой функции t_i парой ДНФ $D_{t_i}^0, D_{t_i}^1$ к заданию функции в виде таблицы истинности не вызывает затруднений [21].

Таблица 2

Задание областей значений кофакторов t_8, t_{11} в виде ДНФ

Кофактор	$x_3 x_4 x_5$	Значение	ДНФ	Область значений
t_8	-10 1-1	0 0	$D_{t_8}^0$	$M_{t_8}^0$
	-00 0-1	1 1	$D_{t_8}^1$	$M_{t_8}^1$
t_{11}	-00 0-1	0 0	$D_{t_{11}}^0$	$M_{t_{11}}^0$
	-10 1-1	1 1	$D_{t_{11}}^1$	$M_{t_{11}}^1$

Этап 2. Выбор формы реализации кофакторов. Введем в рассмотрение k -компонентный булев вектор $\alpha = (\alpha_1, \dots, \alpha_k)$, называемый далее вектором поляризации для кофакторов (функций) t_1, \dots, t_k . Обозначим $\alpha_i = 1$, если рассматривается кофактор t_i , и $\alpha_i = 0$, если берется инверсия \bar{t}_i кофактора t_i .

Задача 1. Для кофакторов t_1, \dots, t_k требуется найти вектор поляризации $\alpha = (\alpha_1, \dots, \alpha_k)$, для которого значение оценки сложности BDD, построенной по общей для всех кофакторов перестановке переменных, будет минимально.

В качестве оценки сложности BDD в большинстве научных работ рассматривается число вершин графа BDD. Однако уравнения разложения Шеннона, соответствующие вершинам BDD, могут содержать различное число литералов. Поэтому в настоящем исследовании целесообразно в качестве оценок сложности BDD (и других алгебраических представлений) использовать критерий «число литералов» в уравнениях, соответствующих BDD. Решение задачи 1 представляет особый интерес и в данной работе не рассматривается. Далее для демонстрации предлагаемого метода некоторые кофакторы при реализации выбраны в инверсной форме, что привело к получению новой перестановки $\langle x_1, x_2, x_4, x_5, x_3 \rangle$ переменных (рис. 2) при реализации кофакторов третьего уровня BDD и некоторому сокращению числа литералов. Многоуровневое представление (3) содержит 98 литералов булевых переменных:

$$\begin{aligned}
 f_1 &= \bar{x}_1 h_1 \vee x_1 h_2; f_2 = \bar{x}_1 h_2 \vee x_1 h_3; f_3 = \bar{x}_1 h_4 \vee x_1 h_5; f_4 = \bar{x}_1 h_4 \vee x_1 h_6; f_5 = \bar{x}_1 h_6 \vee x_1 h_7; h_1 = \bar{x}_2 \bar{g}_1 \vee x_2 \bar{g}_3; \\
 h_2 &= \bar{x}_2 g_2 \vee x_2 \bar{g}_8; h_3 = \bar{x}_2 \bar{g}_5 \vee x_2 g_6; h_4 = \bar{x}_2 g_6 \vee x_2 \bar{g}_7; h_5 = \bar{x}_2 g_6 \vee x_2 g_{10}; h_6 = \bar{x}_2 g_4 \vee x_2 g_9; \\
 h_7 &= \bar{x}_2 g_8 \vee x_2 g_{10}; g_1 = \bar{x}_4 s_2 \vee x_4 s_3; g_2 = \bar{x}_4 \bar{x}_5 \vee x_4 s_1; g_3 = \bar{x}_4 s_0 \vee x_4 s_1; g_4 = \bar{x}_4 x_5 \vee x_4 s_2; \\
 g_5 &= \bar{x}_4 s_0 \vee x_4 x_5; g_6 = \bar{x}_4 s_5 \vee x_4 s_6; g_7 = \bar{x}_4 \bar{x}_5 \vee x_4 x_5; g_8 = \bar{x}_4 s_3 \vee x_4 s_4; g_9 = \bar{x}_4 s_2 \vee x_4 \bar{x}_5; \\
 g_{10} &= \bar{x}_4 x_3 \vee x_4 x_5; s_0 = \bar{x}_5 x_3; s_1 = x_5 \bar{x}_3; s_2 = \bar{x}_5 \bar{x}_3; s_3 = x_5 x_3; s_4 = \bar{x}_5 \vee x_5 x_3; s_5 = \bar{x}_5 \bar{x}_3 \vee x_5; \\
 s_6 &= \bar{x}_5 \bar{x}_3 \vee x_5 x_3.
 \end{aligned} \tag{3}$$

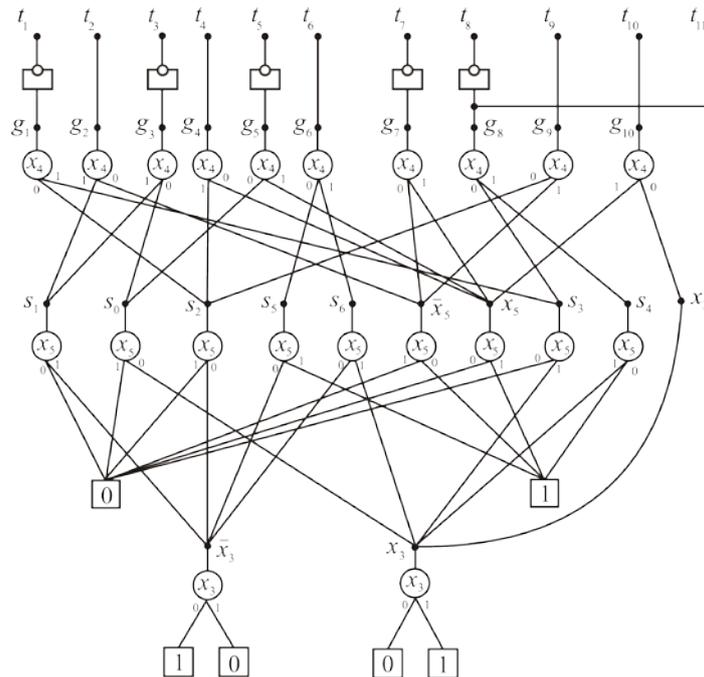


Рис. 2. Результат выполнения этапа 2 – реализация инверсных форм кофакторов t_1, t_3, t_5, t_7, t_8

Этап 3. Нахождение всех вариантов алгебраических представлений кофакторов с использованием их инверсий. Для получения уравнений, задающих алгебраические представления кофакторов, будем использовать обобщенно ортогонализированные формы векторных булевых функций. Нахождение обобщенно ортогонализированной формы для векторной булевой функции $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$ сводится к задаче нахождения ее минимального дизъюнктивного базиса. Зададим каждую компонентную булеву функцию $f_i(\mathbf{x})$ двумя полностью определенными булевыми функциями $\lambda_i^0(\mathbf{x}), \lambda_i^1(\mathbf{x})$, такими, что функция $\lambda_i^0(\mathbf{x})$ имеет характеристическое множество $M_{f_i}^0$, а функция $\lambda_i^1(\mathbf{x})$ – характеристическое множество $M_{f_i}^1$. Минимальным дизъюнктивным базисом для $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$ называется минимальная по мощности система попарно ортогональных полностью определенных булевых функций $\Pi = \{\Pi_1(\mathbf{x}), \dots, \Pi_k(\mathbf{x})\}$, такая, что каждая полностью определенная функция $\lambda_i^1(\mathbf{x}), \lambda_i^0(\mathbf{x})$ равна дизъюнкции некоторого подмножества функций системы Π . Полностью определенная булева функция $f_i(\mathbf{x})$ задает двухблочное разбиение $R_i = \{M_{f_i}^0, M_{f_i}^1\}$ булева пространства V^x на непересекающиеся подмножества. Нахождение для $\mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$ минимального дизъюнктивного базиса (в теоретико-множественной интерпретации) может быть сведено к нахождению произведения разбиений $R_i, i = 1, \dots, m$, и составлению из непустых блоков полученного произведения разбиений каждого из исходных блоков $M_{f_i}^0, M_{f_i}^1$ [27, с. 12]. Пусть каждая компонентная функция f_i задана парой попарно ортогональных ДНФ $D_{f_i}^0, D_{f_i}^1$, данные ДНФ представляют функции $\lambda_i^0(\mathbf{x}), \lambda_i^1(\mathbf{x})$ соответственно. В этом случае нахождение минимального дизъюнктивного базиса Π может быть сведено к выполнению операций перемножения, инверсирования и сравнения на равенство ДНФ (алгоритм приведен в работе [28, с. 124]).

Опишем алгоритм получения уравнений (алгебраических представлений) для кофакторов одного уровня BDD. Рассмотрим кофакторы g_1, \dots, g_{10} третьего уровня BDD (рис. 2). Для каждой неупорядоченной пары g_i, g_j , где $i, j = 1, \dots, 10, i \neq j$, построим обобщенно

ортогонализированную форму, затем для каждой из функций g_p , $p = 1, \dots, 10$, $p \neq i$, $p \neq j$, будет проверяться выполнение равенств

$$g_p = g_i \vee g_j, \quad (4)$$

$$g_p = g_i \& g_j, \quad (5)$$

$$g_p = g_i \oplus g_j. \quad (6)$$

При этом каждый из кофакторов рассматривается как в прямой, так и инверсной форме. Например, для дизъюнкции проверяется выполнение равенств

$$g_p = g_i \vee g_j, \quad g_p = \bar{g}_i \vee g_j, \quad g_p = g_i \vee \bar{g}_j, \quad g_p = \bar{g}_i \vee \bar{g}_j, \\ \bar{g}_p = g_i \vee g_j, \quad \bar{g}_p = \bar{g}_i \vee g_j, \quad \bar{g}_p = g_i \vee \bar{g}_j, \quad \bar{g}_p = \bar{g}_i \vee \bar{g}_j.$$

Подобные равенства проверяются для конъюнкции и суммы по модулю два. Если равенство выполняется, то оно записывается в виде уравнения. Легко понять, что согласно законам булевой алгебры будут появляться двойственные уравнения. Например, если выполняется равенство $g_p = \bar{g}_i \vee \bar{g}_j$, то обязательно будет записано двойственное равенство $\bar{g}_p = g_i \& g_j$ и т. д. Если выполняется равенство $g_p = g_i \oplus g_j$, то при таких проверках будут получены еще три двойственных равенства: $g_p = \bar{g}_i \oplus \bar{g}_j$, $\bar{g}_p = \bar{g}_i \oplus g_j$, $\bar{g}_p = g_i \oplus \bar{g}_j$. Для проведения проверок выполнения всех перечисленных равенств по обобщенно ортогонализированной форме векторной функции $g_{i,j} = (g_i, g_j)$ строится обобщенно ортогонализированная форма векторной функции $g = (g_p, g_i, g_j)$, по которой легко проверяется выполнение данных равенств. Приведем пример: по BDD на рис. 2 выпишем элементарные конъюнкции, которые задают в виде ДНФ области нулевых и единичных значений кофакторов g_1 , g_7 , g_6 , g_8 . Соответствующие ДНФ приведены в табл. 3.

Таблица 3

Задание областей значений кофакторов g_1 , g_7 , g_6 , g_8 в виде ДНФ

Компонентная функция	$x_3 x_4 x_5$	Значение	ДНФ	Область значений	Функция λ
g_1	-01 -10 011 100	0 0 0 0	$D_{g_1}^0$	$M_{g_1}^0$	λ_1^0
	000 111	1 1	$D_{g_1}^1$	$M_{g_1}^1$	λ_1^1
g_7	-01 -10	0 0	$D_{g_7}^0$	$M_{g_7}^0$	λ_7^0
	-00 -11	1 1	$D_{g_7}^1$	$M_{g_7}^1$	λ_7^1
g_6	100 011 110	0 0 0	$D_{g_6}^0$	$M_{g_6}^0$	λ_6^0
	000 -01 111 010	1 1 1 1	$D_{g_6}^1$	$M_{g_6}^1$	λ_6^1
	-00 010 011	0 0 0	$D_{g_8}^0$	$M_{g_8}^0$	λ_8^0
	101 -10 111	1 1 1	$D_{g_8}^1$	$M_{g_8}^1$	λ_8^1

ДНФ Π_i минимального дизъюнктивного базиса для векторной функции $\mathbf{g}_{7,6} = (g_7, g_6)$ приведены в табл. 4. Представим ДНФ, задающие области нулевых и единичных значений кофакторов g_7, g_6 , в виде дизъюнкций ДНФ минимального дизъюнктивного базиса:

$$D_{g_7}^0 = \Pi_2 \vee \Pi_3, D_{g_7}^1 = \Pi_1 \vee \Pi_4, D_{g_6}^0 = \Pi_3 \vee \Pi_4, D_{g_6}^1 = \Pi_1 \vee \Pi_2.$$

Обобщенно ортогонализированные формы векторных функций $\mathbf{g}_{7,6} = (g_7, g_6)$ и $\mathbf{g} = (g_1, g_7, g_6)$ даны в табл. 5 и 6 соответственно. Кофактор g_1 может быть выражен на тех же четырех ДНФ Π_i : $D_{g_1}^0 = \Pi_2 \vee \Pi_3 \vee \Pi_4$; $D_{g_1}^1 = \Pi_1$. Функция g_1 может быть выражена как конъюнкция функций g_7, g_6 , если рассматривать логическую операцию $\&$ над соответствующими значениями функций g_7, g_6 из табл. 5, т. е. $g_1 = g_7 \& g_6$. По табл. 6 легко проверить, что $g_8 \neq g_7 \vee g_6, g_8 \neq g_7 \oplus g_6$.

Таблица 4

Минимальный дизъюнктивный базис для векторной функции $\mathbf{g} = (g_7, g_6)$

Π_i	$x_3 \ x_4 \ x_5$
Π_1	0 0 0 1 1 1
Π_2	0 0 1 0 1 0 1 0 1
Π_3	1 1 0
Π_4	0 1 1 1 0 0

Таблица 5

Обобщенно ортогонализированная форма векторной функции $\mathbf{g} = (g_7, g_6)$

Π_i	$g_7 \ g_6$
Π_1	1 1
Π_2	0 1
Π_3	0 0
Π_4	1 0

Таблица 6

Обобщенно ортогонализированная форма векторной функции $\mathbf{g} = (g_1, g_7, g_6)$

Π_i	$g_1 \ g_7 \ g_6$
Π_1	1 1 1
Π_2	0 0 1
Π_3	0 0 0
Π_4	0 1 0

Если же рассматривать кофактор g_8 , то для проверки равенств

$$g_8 = g_7 \vee g_6, \tag{7}$$

$$g_8 = g_7 \& g_6, \tag{8}$$

$$g_8 = g_7 \oplus g_6 \tag{9}$$

построим обобщенно ортогонализированную форму векторной функции $\mathbf{g} = (g_8, g_7, g_6)$, перемножая ДНФ Π^i (табл. 4) с ДНФ $D_{g_8}^0, D_{g_8}^1$ (табл. 3). Тогда в результате построения минимального дизъюнктивного базиса для $\mathbf{g} = (g_8, g_7, g_6)$ выяснится, что $\Pi_1 = P_1 \vee P_2, \Pi_2 = P_3 \vee P_4, \Pi_3 = P_5, \Pi_4 = P_6$, где P_i даны в табл. 7. В результате получается обобщенно ортогонализированная форма векторной функции $\mathbf{g} = (g_8, g_7, g_6)$ (табл. 8), которой соответствуют формулы $D_{g_7}^0 = P_3 \vee P_4 \vee P_5, D_{g_7}^1 = P_1 \vee P_2 \vee P_6, D_{g_6}^0 = P_5 \vee P_6, D_{g_6}^1 = P_1 \vee P_2 \vee P_3 \vee P_4, D_{g_8}^0 = P_1 \vee P_3 \vee P_6, D_{g_8}^1 = P_2 \vee P_4 \vee P_5$. По табл. 8 легко проверить, что кофактор g_8 не может быть представлен формулами (7)–(9), т. е. $g_8 \neq g_7 \vee g_6, g_8 \neq g_7 \& g_6, g_8 \neq g_7 \oplus g_6$. Использование инверсий кофакторов g_8, g_7, g_6 также не приводит к получению алгебраических представлений кофактора g_8 . Такой вывод можно сделать, если представить каждую ДНФ $P_i, i = 1, \dots, 6$, минимального дизъюнктивного базиса в виде совершенной ДНФ, а затем перейти к таблицам истинности кофакторов g_8, g_7, g_6 .

Таблица 7

Минимальный дизъюнктивный
базис для векторной функции

$$g = (g_8, g_7, g_6)$$

P_i	$x_3 \ x_4 \ x_5$
P_1	0 0 0
P_2	1 1 1
P_3	0 0 1
P_4	0 1 0 1 0 1
P_5	1 1 0
P_6	0 1 1 1 0 0

Таблица 8

Обобщенно ортогонализованная
форма векторной функции

$$g = (g_8, g_7, g_6)$$

P_i	$g_8 \ g_7 \ g_6$
P_1	0 1 1
P_2	1 1 1
P_3	0 0 1
P_4	1 0 1
P_5	1 0 0
P_6	0 1 0

Обобщенно ортогонализованная форма – это компактная форма задания таблицы истинности полностью определенной векторной булевой функции. Компактность может быть обеспечена за счет представления ДНФ минимального дизъюнктивного базиса минимизированными формами, например кратчайшими ДНФ [2].

Утверждение. Проверка условий существования функциональных представлений

$$\varphi(x) = f_1(x) \vee f_2(x), \quad \varphi(x) = f_1(x) \& f_2(x), \quad \varphi(x) = f_1(x) \oplus f_2(x)$$

может выполняться как по таблице истинности векторной функции $f = (\varphi, f_1, f_2)$, так и по таблице задания обобщенно ортогонализованной формы этой векторной функции.

В рассматриваемых примерах функций от трех переменных обобщенно ортогонализированные формы (табл. 5, 6, 8) содержат четыре либо шесть строк, что мало отличается от таблиц истинности, содержащих для данных примеров по восемь строк. В ситуациях, когда решаются практические задачи большой размерности (десятки переменных), обобщенно ортогонализированные формы могут быть гораздо более компактнее таблиц истинности.

Зададим все кофакторы таблицей истинности (табл. 9), что позволит проверять правильность алгебраического представления кофакторов третьего уровня BDD. В результате выполнения этапа 3 будет получено 48 уравнений, не считая 48 двойственных. Приведем их в двух таблицах: уравнения без использования инверсий – в табл. 10, с использованием инверсий – в табл. 11.

Таблица 9

Кофакторы третьего уровня BDD (см. рис. 2)

$x_3 \ x_4 \ x_5$	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	g_{10}
0 0 0	1	1	0	0	0	1	1	0	1	0
0 0 1	0	0	0	1	0	1	0	0	0	0
0 1 0	0	0	0	1	0	1	0	1	1	0
0 1 1	0	1	1	0	1	0	1	0	0	1
1 0 0	0	1	1	0	1	0	1	0	0	1
1 0 1	0	0	0	1	0	1	0	1	0	1
1 1 0	0	0	0	0	0	0	0	1	1	0
1 1 1	1	0	0	0	1	1	1	1	0	1

Для каждого из уравнений с операторами \vee , $\&$ в табл. 10 и 11 имеется двойственное уравнение, а для уравнений с операторами \oplus из табл. 11 – два двойственных.

Таблица 10

Варианты представления кофакторов третьего уровня BDD без использования инверсий операндов

Номер уравнения	Уравнение
1	$g_1 = g_2 \oplus g_5$
2	$g_1 = g_6 \oplus g_4$
3	$g_1 = g_7 \oplus g_3$
4	$g_1 = g_7 \& g_6$
5	$g_2 = g_5 \oplus g_1$
6	$g_3 = g_2 \& g_{10}$
7	$g_3 = g_2 \& g_5$
8	$g_3 = g_7 \oplus g_1$
9	$g_4 = g_6 \oplus g_1$
10	$g_5 = g_2 \oplus g_1$
11	$g_5 = g_7 \& g_{10}$
12	$g_6 = g_1 \oplus g_4$
13	$g_6 = g_1 \vee g_4$
14	$g_7 = g_2 \vee g_1$
15	$g_7 = g_2 \vee g_5$
16	$g_7 = g_3 \oplus g_1$
17	$g_7 = g_3 \vee g_1$
18	$g_7 = g_5 \vee g_1$

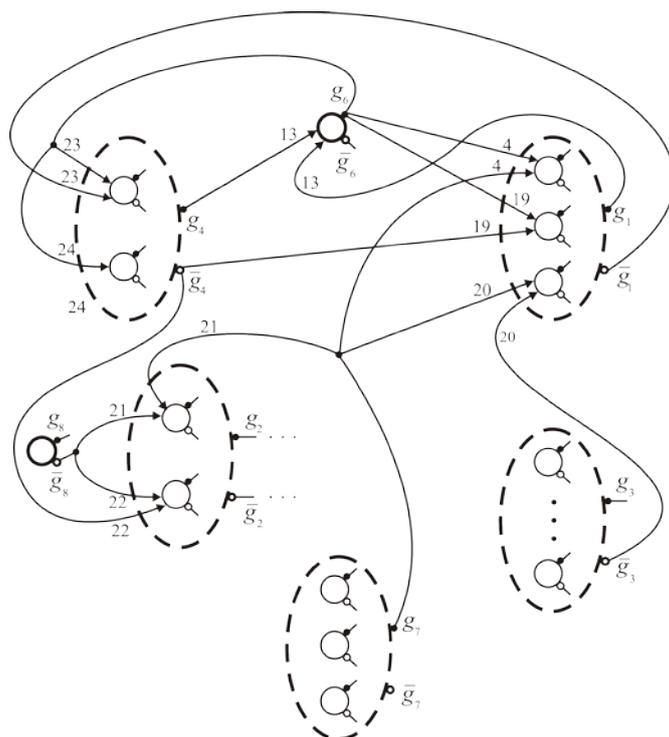
Таблица 11

Варианты представления кофакторов третьего уровня BDD с использованием инверсий операндов

Номер уравнения	Уравнение	Номер уравнения	Уравнение
19	$g_1 = g_6 \& \bar{g}_4$	34	$g_3 = \bar{g}_8 \& g_5$
20	$g_1 = g_7 \& \bar{g}_3$	35	$g_3 = \bar{g}_8 \& \bar{g}_6$
21	$g_2 = \bar{g}_8 \& g_7$	36	$g_3 = \bar{g}_9 \& \bar{g}_6$
22	$g_2 = \bar{g}_8 \& \bar{g}_4$	37	$g_5 = g_{10} \& \bar{g}_4$
23	$g_4 = g_6 \& \bar{g}_1$	38	$g_5 = g_7 \& \bar{g}_9$
24	$g_4 = g_6 \vee \bar{g}_7$	39	$g_5 = \bar{g}_9 \& \bar{g}_4$
25	$g_3 = g_{10} \& \bar{g}_6$	40	$g_1 = \bar{g}_2 \oplus \bar{g}_5$
26	$g_3 = g_2 \& \bar{g}_1$	41	$g_1 = \bar{g}_6 \oplus \bar{g}_4$
27	$g_3 = g_2 \& \bar{g}_6$	42	$g_1 = \bar{g}_7 \oplus \bar{g}_3$
28	$g_3 = g_2 \& \bar{g}_9$	43	$g_2 = \bar{g}_5 \oplus \bar{g}_1$
29	$g_3 = g_5 \& \bar{g}_1$	44	$g_3 = \bar{g}_7 \oplus \bar{g}_1$
30	$g_3 = g_5 \& \bar{g}_6$	45	$g_4 = \bar{g}_6 \oplus \bar{g}_1$
31	$g_3 = g_7 \& \bar{g}_1$	46	$g_5 = \bar{g}_2 \oplus \bar{g}_1$
32	$g_3 = g_7 \& \bar{g}_6$	47	$g_6 = \bar{g}_1 \oplus \bar{g}_4$
33	$g_3 = \bar{g}_8 \& g_{10}$	48	$g_7 = \bar{g}_3 \oplus \bar{g}_1$

Этап 4. Нахождение множества реализуемых кофакторов. Построим ориентированный граф G , задающий формулы алгебраических представлений кофакторов (будем называть их логическими уравнениями либо просто уравнениями). Вершинам графа G соответствуют кофакторы в прямой форме, которые упоминаются в уравнениях. Обозначим это множество кофакторов K_Y , а множество кофакторов, не участвующих в уравнениях, K_Z . Дуги (ориентированные ребра), соответствующие операндам алгебраически представленного кофактора (результату логической операции), помечены одним и тем же числом – номером уравнения. Каждое уравнение задается подграфом с тремя вершинами и двумя помеченными дугами: вершина, соответствующая представимому кофактору, имеет две заходящие дуги (с одной и той же меткой – номером уравнения). Исходящие из вершины представимого кофактора дуги (рис. 3) могут соответствовать прямой форме кофактора (черный кружок) либо инверсной форме кофактора (светлый кружок).

Вершины, соответствующие операндам уравнения, инцидентны исходящим дугам. Например, уравнению 13 $g_6 = g_1 \vee g_4$ соответствует подграф с тремя вершинами g_1, g_4, g_6 : в вершину g_6 заходят из вершин g_1, g_4 дуги, помеченные числом 13 (номером уравнения). Если кофактор в уравнении разложения Шеннона содержит один либо два литерала, то для него на этапе 2 не строятся алгебраические представления, поэтому соответствующая данному кофактору вершина в графе G не имеет заходящих дуг. Введем понятие кластера (подмножества вершин графа G): в кластер входят вершины, соответствующие уравнениям с одним и тем же кофактором (возможно, в инверсном виде) в левой части. Именем одинаковых кофакторов и обозначается кластер. В рассматриваемом примере граф G будет содержать 10 вершин, которые соединяются между собой 48 парами дуг. Фрагмент графа показан на рис. 3. Заметим, что каждой из вершин графа G соответствует кофактор, уравнение которого содержит четыре литерала (см. выражение (3) для кофакторов третьего уровня BDD).

Рис. 3. Фрагмент графа G

С использованием графа G задача нахождения множества реализуемых кофакторов сводится к следующей задаче.

Задача 2. В графе G требуется оставить в каждом кластере только одну вершину (условие A) и удалить из графа G такое подмножество пар одинаково помеченных дуг, чтобы граф G стал ациклическим (условие B) и содержал наибольшее число неизолированных вершин, в каждую из которых заходит только одна пара одинаково помеченных дуг (условие C).

Смысл оставления в кластере одной вершины (условие A) очевиден – кофактор при схемной реализации достаточно представить только одним уравнением. Требование ациклическости (условие B) продемонстрируем на примере двух уравнений ($g_6 = g_1 \vee g_4$, $g_1 = g_6 \& g_7$) из рассматриваемого примера. Данные уравнения в графе G создают цикл, т. е. логическое противоречие в требовании алгебраического (и схемного) представления кофакторов, заключающееся в следующем. Чтобы выразить g_6 в виде конъюнкции $g_6 = g_1 \vee g_4$ (уравнение 13), требуется схема, реализующая кофактор g_1 . Однако, чтобы получить g_1 в виде $g_1 = g_7 \& g_6$ (см. уравнение (4)), требуется схема, реализующая g_6 . Условие C требует, чтобы как можно больше кофакторов было представлено в виде конъюнкции, дизъюнкции либо суммы по модулю два.

После решения задачи 2 множество K_Y вершин графа G разбивается на три попарно непересекающихся подмножества. Вершины, не имеющие входящих дуг, но имеющие исходящую дугу, соответствуют нереализуемым кофакторам. Вершины, имеющие одну пару входящих дуг (с одной и той же пометкой), соответствуют реализуемым кофакторам. Вершины, которые оказались изолированными, например, при удалении дуг для удовлетворения условия B , не будут участвовать в уравнениях и зачисляются в множество K_Z также нереализуемых кофакторов. Реализуемый кофактор будет записан в виде уравнения в результирующее многоуровневое представление системы булевых функций.

Для решения задачи 2 предлагается использовать следующий *эвристический алгоритм*, основанный на последовательном рассмотрении уравнений, по которым строятся соответствующие подграфы графа G :

Шаг 1. Построить подграф G_{NOTI} графа G , образованный вершинами и дугами уравнений без использования инверсий.

Шаг 2. Удалить из каждого кластера подграфа G_{NOTI} вершины (и заходящие в них дуги), соответствующие уравнениям, которые содержат операцию \oplus в тех случаях, если имеются вершины, соответствующие уравнениям, которые содержат операцию \vee либо $\&$.

Шаг 3. Сократить число вершин в кластерах подграфа G_{SNOTI} и оставить в каждом кластере только одну вершину.

Шаг 3.1. Упорядочить кластеры по возрастанию (неубыванию) числа содержащихся в них вершин и рассматривать кластеры в этом порядке.

Шаг 3.2. Удалить вершины в кластере.

Положить пустым текущее множество T вершин: $T = \emptyset$. Каждая оставленная вершина g_r в кластере, в которую заходят одноименно помеченные дуги, инцидентные вершинам g_i, g_j , пополняет множество T элементами g_i, g_j, g_r .

Для выбора оставляемой в первом кластере вершины применяется

Эвристика 1. Каждая вершина g_r кластера, имеющая заходящие дуги из вершин g_i, g_j , оценивается суммарным числом W_{qr} исходящих дуг из вершин g_i, g_j . Оставляется в кластере та вершина g_s , которой соответствует максимальное значение W_{qr} .

Вершины, которые не остались в кластере, удаляются из графа вместе с заходящими в них дугами. В множество T заносятся три вершины: вершина g_s и две вершины, инцидентные заходящим в вершину g_s дугам.

Для следующих рассматриваемых кластеров используется

Эвристика 2. В кластере оставляется та вершина, которая имеет две заходящие дуги из вершин множества T . Если такой вершины нет, то оставляется та вершина, которая имеет одну заходящую дугу из вершин множества T . Если нет вершин, для которых данная эвристика выполняется, то для оставления вершины в кластере используется эвристика 1. Вершины, которые не остались в кластере, удаляются из графа вместе с заходящими в них дугами.

Шаг 3 считается выполненным, когда в каждом кластере останется по одной вершине.

Шаг 4. Удалить пары одинаково помеченных дуг для получения ациклического подграфа G_{SNOTI} . (Выполняется, если граф G_{SNOTI} содержит циклы, т. е. не является ациклическим.)

Для графа G_{SNOTI} , содержащего циклы, выполняются шаги 4.1–4.3 до тех пор, пока граф не станет ациклическим.

Шаг 4.1. Найти все циклы графа G_{SNOTI} .

Шаг 4.2. Каждую пару дуг с одинаковой пометкой (номером) p оценить числом C_p циклов, которые могут быть разорваны при удалении из графа данной пары дуг.

Шаг 4.3. Удалить пару дуг с пометкой p , которой соответствует максимальное значение числа C_p . Переход на шаг 4.1.

Шаг 5. Из множества уравнений с инверсиями кофакторов удалить уравнения для реализованных кофакторов. Поочередно добавлять в полученный на шаге 4 ациклический граф дуги и вершины, соответствующие нереализованным кофакторам, и проверять получаемый граф на ациклическость. Дуги и вершины из множества уравнений для кофакторов с инверсиями не добавляются в граф, если они вносят цикл. Если рассмотрены все уравнения из множества оставшихся уравнений с инверсиями, то выполнить переход на шаг 6.

Шаг 6. Конец.

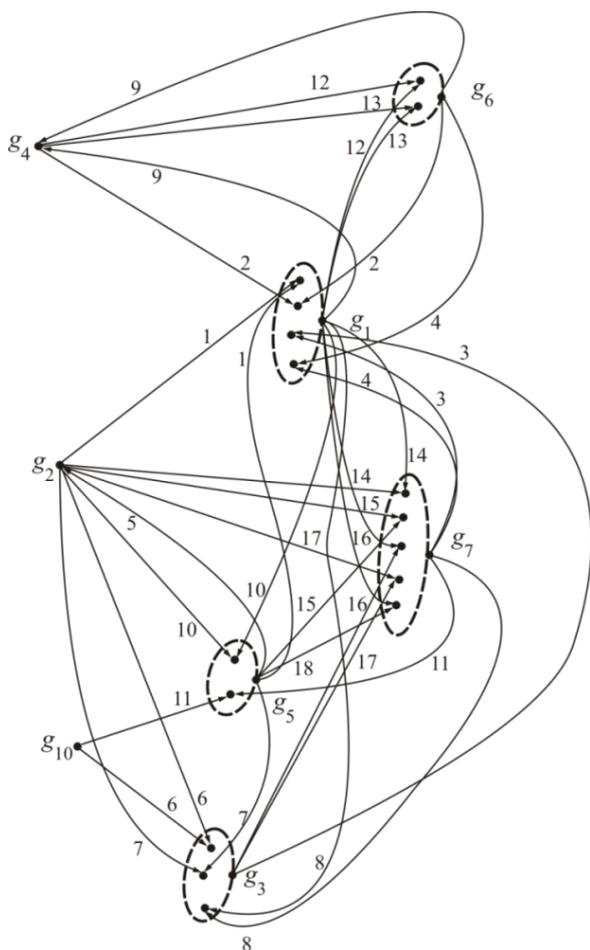


Рис. 4. Подграф G_{NOTI} для 18 уравнений из табл. 10

Рассмотрим алгоритм на примере графа G_{SNOTI} (рис. 4).

Шаг 1. Подграф G_{NOTI} изображен на рис. 4, содержит восемь вершин, из них вершины g_2, g_4, g_{10} не являются вершинами-кластерами, вершин g_8, g_9 нет в подграфе G_{NOTI} , так как кофакторы g_8, g_9 не входят в уравнения из табл. 10.

Замечание: если граф G построен по уравнениям, в которых нет операций инверсий операндов, то вершины графа представляются на рис. 4 точками (черными кружками, а светлые кружки отсутствуют).

Шаг 2. Уравнения, по которым строится подграф G_{SNOTI} графа G_{NOTI} , заданы в табл. 12, полученный подграф G_{SNOTI} показан на рис. 5.

Таблица 12

Варианты алгебраического представления кофакторов после шага 2 алгоритма решения задачи 2

Номер уравнения	Уравнение	Номер уравнения	Уравнение
4	$g_1 = g_7 \& g_6$	13	$g_6 = g_1 \vee g_4$
7	$g_3 = g_2 \& g_5$	17	$g_7 = g_3 \vee g_1$
6	$g_3 = g_2 \& g_{10}$	18	$g_7 = g_5 \vee g_1$
9	$g_4 = g_6 \oplus g_1$	14	$g_7 = g_2 \vee g_1$
11	$g_5 = g_7 \& g_{10}$	15	$g_7 = g_2 \vee g_5$

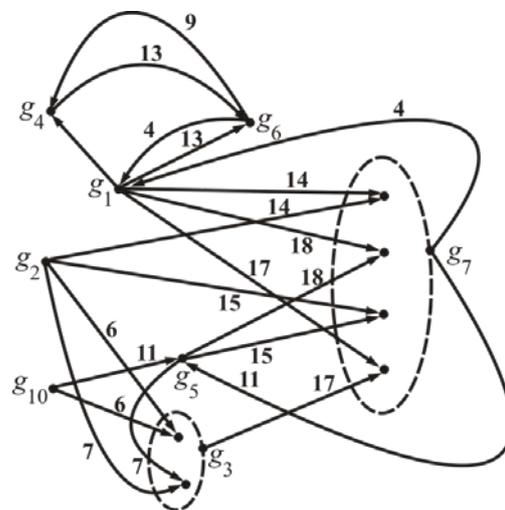


Рис. 5. Подграф G_{SNOTI}

Шаг 3. Сокращение числа вершин в кластерах подграфа G_{SNOTI} .

Шаг 3.1. Порядок рассмотрения кластеров $\langle g_3, g_7 \rangle$.

Шаг 3.2. По эвристике 1 на этом шаге в кластере g_3 оставляем вершину с заходящими дугами g_2, g_5 , так как суммарное число W_{g_3} исходящих дуг для нее равно семи. Полагаем $T = \{g_2, g_5, g_3\}$. Для кластера g_7 выбираем вершину с заходящими дугами g_2, g_5 (эвристика 2), так как в этом случае в множестве $T = \{g_2, g_5, g_3\}$ имеются две общие вершины g_2, g_5 . После этого граф G принимает вид, показанный на рис. 6, а.

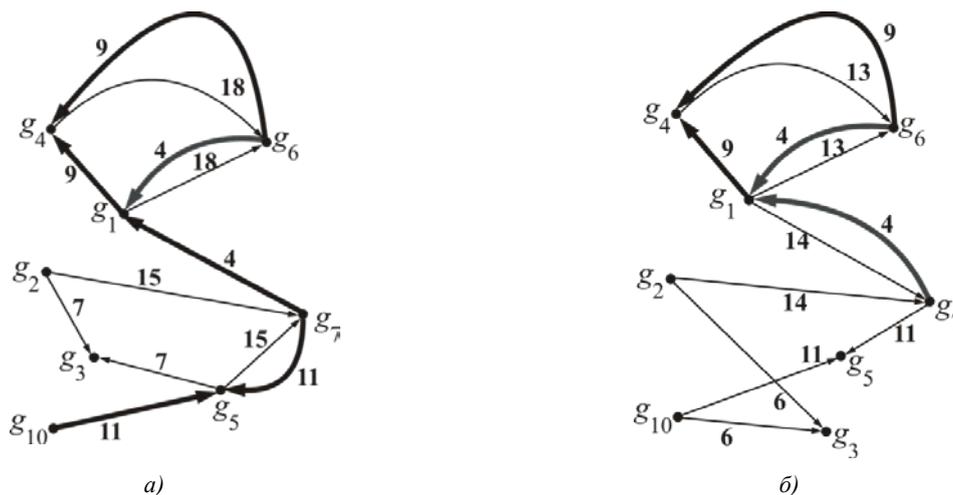


Рис. 6. Подграф G_{SNOTI} после выполнения шага 3 алгоритма решения задачи 2:
а) по эвристике 1; б) другой вариант

Шаг 4. Чтобы разорвать циклы (первый цикл – $\{g_1, g_6\}$, второй – $\{g_5, g_7\}$, третий – $\{g_4, g_6\}$, четвертый – $\{g_1, g_4, g_6\}$), удаляем пары дуг с пометками 4, 11, 9; результирующий граф, в вершины g_3, g_7, g_6 которого заходят попарно дуги с одинаковыми пометками, соответствует трем выбранным (реализуемым) уравнениям: $g_3 = g_2 \& g_5, g_7 = g_2 \vee g_5, g_6 = g_1 \vee g_4$. Полученное решение означает, что три представимых кофактора (g_3, g_7, g_6) могут быть представлены в виде дизъюнкции либо конъюнкции четырех кофакторов (g_2, g_5, g_1, g_4).

Применяемые эвристики позволяют получать приближенное решение задачи 2. Если на шаге 3 в первом рассматриваемом кластере g_3 будет оставлена вершина, имеющая заходящие дуги из вершин g_2, g_{10} , то можно получить четыре алгебраически представимых кофактора. Покажем это, еще раз выполнив шаги 3 и 4.

Шаг 3 (повторно). Порядок рассмотрения кластеров $\langle g_3, g_7 \rangle$. Оставляем вершину кластера g_3 , имеющую заходящие дуги из вершин g_2, g_{10} , полагаем $T = \{g_2, g_{10}, g_3\}$. Для второго кластера g_7 выбираем вершину с дугами, имеющими пометку 14 и инцидентными вершинам g_1, g_2 , так как $g_2 \in T$, где $T = \{g_2, g_{10}, g_3\}$. По эвристике 1 имеется одна вершина, входящая в множество T . В результате выполнения первого шага граф G принимает вид, представленный на рис. 6, б.

Шаг 4 (повторно). В графе (рис. 6, б) имеются четыре цикла: $\{g_1, g_6\}, \{g_1, g_7\}, \{g_4, g_6\}, \{g_1, g_4, g_6\}$. Удаление пары дуг с пометками 4, 9 разрывает оба цикла. Результирующий ациклический граф, полученный в результате шага 4 эвристического алгоритма, показан на рис. 7. Данному графу соответствуют четыре уравнения: $g_3 = g_2 \& g_{10}, g_5 = g_7 \& g_{10}, g_6 = g_1 \vee g_4,$

$g_7 = g_1 \vee g_2$. Четыре кофактора (g_2, g_{10}, g_1, g_4) позволяют получить алгебраические представления других четырех кофакторов (g_3, g_5, g_6, g_7). Эти представления кофакторов будем использовать далее.

Шаг 5. Из табл. 11 уравнений с инверсиями удалим уравнения для четырех реализуемых кофакторов (g_3, g_5, g_6, g_7), оставшиеся уравнения заданы в табл. 13. В ациклический граф (рис. 7, а) будем добавлять поочередно вершины и дуги, соответствующие уравнениям из табл. 13 (всего 11 уравнений). Каждое из уравнений вносит цикл в граф, кроме уравнения 22 ($g_2 = \bar{g}_8 \& \bar{g}_4$) (рис. 7, б). Таким образом, использование инверсий позволяет найти выражение $g_2 = \bar{g}_8 \& \bar{g}_4$, свидетельствующее о том, что кофактор g_2 стал реализуемым. Реализуемые кофакторы третьего уровня с использованием инверсий: $g_3 = g_7 \& g_{10}$; $g_5 = g_7 \& g_{10}$; $g_6 = g_1 \vee g_4$; $g_7 = g_1 \vee g_2$; $g_2 = \bar{g}_8 \& \bar{g}_4$.

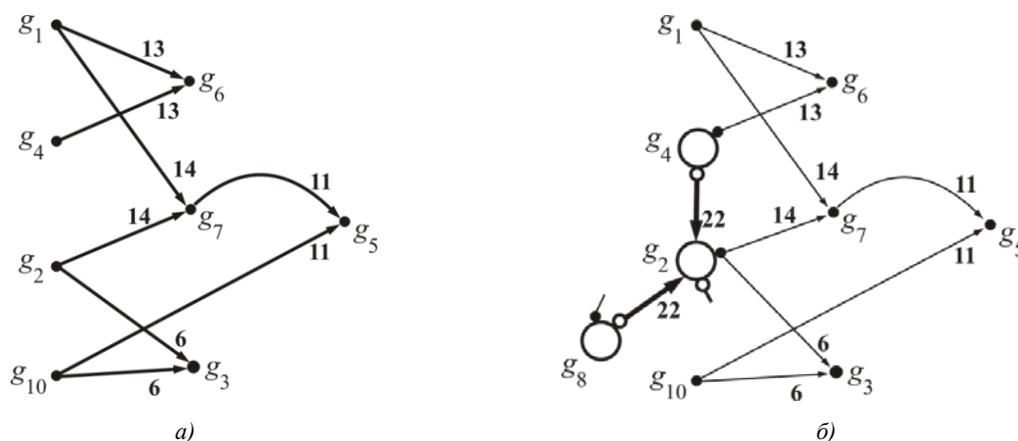


Рис. 7. Ациклический граф G после выполнения: а) шага 4 эвристического алгоритма решения задачи 2; б) эвристического алгоритма решения задачи 2

Таблица 13

Варианты алгебраического представления нереализованных кофакторов

Номер уравнения	Уравнение	Номер уравнения	Уравнение
19	$g_1 = g_6 \& \bar{g}_4$	40	$g_1 = \bar{g}_2 \oplus \bar{g}_5$
20	$g_1 = g_7 \& \bar{g}_3$	41	$g_1 = \bar{g}_6 \oplus \bar{g}_4$
21	$g_2 = \bar{g}_8 \& g_7$	42	$g_1 = \bar{g}_7 \oplus \bar{g}_3$
22	$g_2 = \bar{g}_8 \& \bar{g}_4$	43	$g_2 = \bar{g}_5 \oplus \bar{g}_1$
23	$g_4 = g_6 \& \bar{g}_1$	45	$g_4 = \bar{g}_6 \oplus \bar{g}_1$
24	$g_4 = g_6 \vee \bar{g}_7$		

Этап 5. Построение BDD для кофакторов, которые не являются реализуемыми. Нереализуемыми кофакторами являются $g_1, g_4, g_8, g_9, g_{10}$. Построение BDD для них приводит к перестановке $\langle x_4, x_5, x_3 \rangle$ переменных и следующим формулам разложения Шеннона:

$$g_1 = \bar{x}_4 w_0 \vee x_4 w_2; g_4 = \bar{x}_4 x_5 \vee x_4 w_0; g_8 = \bar{x}_4 w_2 \vee x_4 w_3; g_9 = \bar{x}_4 w_0 \vee x_4 w_5; \quad (10)$$

$$g_{10} = \bar{x}_4 x_3 \vee x_4 x_5; w_0 = \bar{x}_5 \bar{x}_3; w_1 = x_5; w_2 = x_5 x_3; w_3 = \bar{x}_3 \bar{x}_5 \vee x_3; w_4 = x_3; w_5 = \bar{x}_5$$

(сначала по переменной x_4 , затем по переменной x_5). На первом уровне – разложение по переменной x_3 . Кофакторы второго уровня BDD, зависящие от переменных x_5, x_3 , заданы в табл. 14.

Таблица 14

Кофакторы второго уровня BDD

$x_3 x_5$	w_0	$w_1 = x_5$	w_2	w_3	$w_4 = x_3$	$w_5 = \bar{x}_5$
0 0	1	0	0	1	0	1
0 1	0	1	0	0	0	0
1 0	0	0	0	1	1	1
1 1	0	1	1	1	1	0

Этап 6. Составление уравнений для нереализуемых кофакторов с использованием кофакторов нижнего уровня BDD. Составляются уравнения для нереализуемых кофакторов $g_1, g_4, g_8, g_9, g_{10}$ с использованием реализуемых кофакторов и кофакторов второго уровня, заданных в табл. 14. Для этого табл. 14 преобразуется в табл. 15; переменная x_4 является несущественной для кофакторов, приведенных в табл. 15. Полученные уравнения заданы в табл. 16.

Таблица 15

Кофакторы второго уровня BDD с несущественной переменной x_4

$x_3 x_4 x_5$	w_0	$w_1 = x_5$	w_2	w_3	$w_4 = x_3$	$w_5 = \bar{x}_5$
0 0 0	1	0	0	1	0	1
0 0 1	0	1	0	0	0	0
0 1 0	1	0	0	1	0	1
0 1 1	0	1	0	0	0	0
1 0 0	0	0	0	1	1	1
1 0 1	0	1	1	1	1	0
1 1 0	0	0	0	1	1	1
1 1 1	0	1	1	1	1	0

Этап 7. Расширение ациклического подграфа по новым уравнениям. В полученный на этапе 4 граф (рис. 7, б) последовательно (по одному уравнению) добавляются вершины и дуги, соответствующие уравнениям из табл. 16. Каждое из уравнений, кроме уравнения 54 ($g_9 = \bar{g}_{10} \& w_5$), преобразует граф в циклический. Добавление уравнения 54 оставляет граф ациклическим (рис. 8) с шестью реализуемыми кофакторами.

Таблица 16

Варианты алгебраического представления кофакторов второго уровня BDD

Номер уравнения	Уравнение
49	$g_{10} = g_3 \oplus w_2$
50	$g_{10} = g_3 \vee w_2$
51	$g_{10} = \bar{g}_3 \oplus \bar{w}_2$
52	$g_{10} = g_5 \vee w_2$
53	$g_8 = \bar{g}_2 \& w_3$
54	$g_9 = \bar{g}_{10} \& w_5$
55	$g_9 = \bar{g}_{10} \& w_3$
56	$g_9 = g_3 \oplus w_5$
57	$g_9 = \bar{g}_5 \oplus w_5$

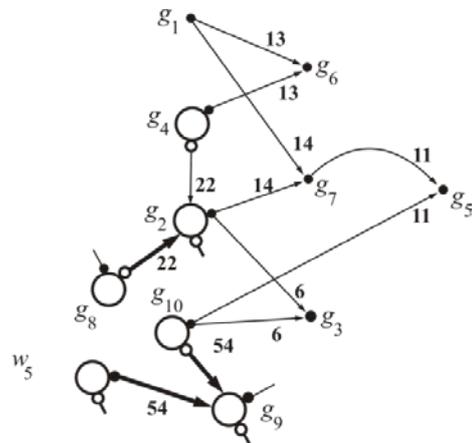


Рис. 8. Результирующий ациклический граф G

Реализуемыми являются шесть кофакторов: $g_3 = g_2 \& g_{10}$, $g_5 = g_7 \& g_{10}$, $g_6 = g_1 \vee g_4$, $g_7 = g_1 \& g_2$, $g_2 = \bar{g}_8 \& \bar{g}_4$, $g_9 = \bar{g}_{10} \& w_5$, где $w_5 = \bar{x}_5$. Уравнение $g_9 = \bar{x}_4 w_0 \vee x_4 w_5$ разложения Шеннона из списка (10) заменено алгебраическим выражением $g_9 = \bar{g}_{10} \& w_5$.

Этап 8. Корректировка многоуровневого BDD-представления векторной функции. После выполнения этапа 7 многоуровневое представление кофакторов третьего уровня BDD записывается в виде следующих уравнений, которым соответствует рис. 9:

$$\begin{aligned}
 &g_3 = g_2 \& g_{10}; \quad g_5 = g_7 \& g_{10}; \quad g_6 = g_1 \vee g_4; \quad g_7 = g_1 \& g_2; \quad c_2 = g_4 \vee g_8; \quad g_2 = \bar{c}_2; \\
 &g_{10} = \bar{g}_{10}; \quad g_9 = w_5 \& c_{10}; \quad g_1 = \bar{x}_4 w_0 \vee x_4 w_2; \quad g_8 = \bar{x}_4 w_2 \vee x_4 w_3; \quad g_{10} = \bar{x}_4 x_3 \vee x_4 x_5; \\
 &g_4 = \bar{x}_4 x_5 \vee x_4 w_0; \quad w_0 = \bar{x}_5 \bar{x}_3; \quad w_2 = x_5 x_3; \quad w_3 = \bar{x}_3 \bar{x}_5 \vee x_3; \quad w_5 = \bar{x}_5.
 \end{aligned}
 \tag{11}$$

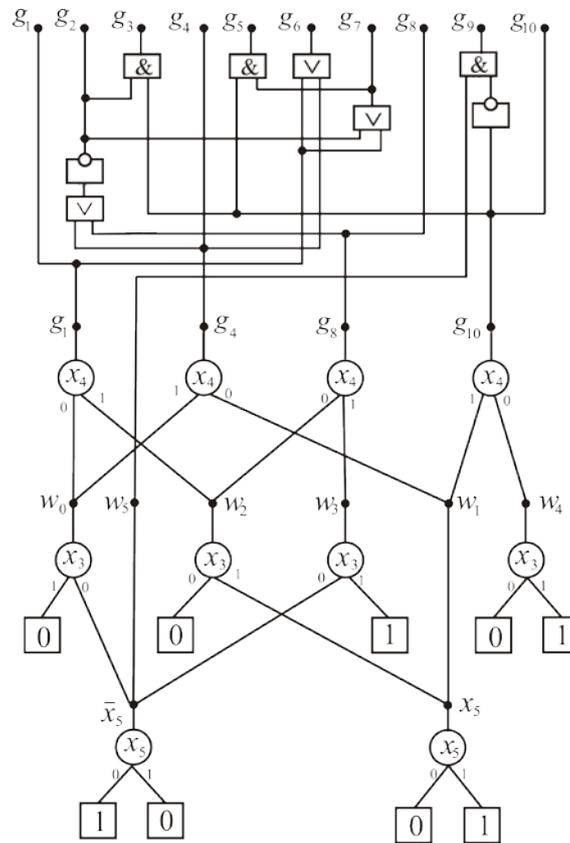


Рис. 9. Реализация подфункций g_1, \dots, g_{10} в виде BDD и алгебраических разложений

Нахождение алгебраических представлений кофакторов на втором уровне BDD. Для тривиальных кофакторов второго уровня, равных переменным либо их инверсиям, алгебраические представления не ищутся. Тривиальными кофакторами второго уровня BDD (см. табл. 14) являются $w_1 = x_5$, $w_4 = x_3$, $w_5 = \bar{x}_5$, нетривиальными – w_0, w_2 . Уравнения для w_0, w_2 , использующие кофакторы второго уровня BDD, заданы в табл. 17. Заметим, что кофактор w_3 не выражается через другие кофакторы, а w_0 представляет собой элементарную конъюнкцию: $w_0 = \bar{x}_3 \bar{x}_5$. Реализация в виде BDD кофакторов второго уровня не может быть упрощена за счет их алгебраических представлений.

Результирующее многоуровневое представление векторной булевой функции (см. табл. 1) содержит 86 литералов:

$$\begin{aligned}
 f_1 &= \bar{x}_1 h_1 \vee x_1 h_2; & f_2 &= \bar{x}_1 h_2 \vee x_1 h_3; & f_3 &= \bar{x}_1 h_4 \vee x_1 h_5; & f_4 &= \bar{x}_1 h_4 \vee x_1 h_6; & f_5 &= \bar{x}_1 h_6 \vee x_1 h_7; \\
 h_1 &= \bar{x}_2 \bar{g}_1 \vee x_2 \bar{g}_3; & h_2 &= \bar{x}_2 g_2 \vee x_2 \bar{g}_8; & h_3 &= \bar{x}_2 \bar{g}_5 \vee x_2 g_6; & h_4 &= \bar{x}_2 g_6 \vee x_2 \bar{g}_7; & h_5 &= \bar{x}_2 g_6 \vee x_2 g_{10}; \\
 h_6 &= \bar{x}_2 g_4 \vee x_2 g_9; & h_7 &= \bar{x}_2 g_8 \vee x_2 g_{10}; & g_3 &= g_2 \& g_{10}; & g_5 &= g_7 \& g_{10}; & g_6 &= g_1 \vee g_4; \\
 g_7 &= g_1 \vee g_2; & c_2 &= g_4 \vee g_8; & g_2 &= \bar{c}_2; & c_{10} &= \bar{g}_{10}; & g_9 &= w_5 \& c_{10}; & g_1 &= \bar{x}_4 w_0 \vee x_4 w_2; \\
 g_8 &= \bar{x}_4 w_2 \vee x_4 w_3; & g_{10} &= \bar{x}_4 x_3 \vee x_4 x_5; & g_4 &= \bar{x}_4 x_5 \vee x_4 w_0; & w_0 &= \bar{x}_5 \bar{x}_3; & w_2 &= x_5 x_3; & w_3 &= \bar{x}_3 \bar{x}_5 \vee x_3; & w_5 &= \bar{x}_5.
 \end{aligned}
 \tag{12}$$

В исходный граф BDD (см. рис. 1) внесены графические изображения логических уравнений реализуемых кофакторов и операторов инверсии, в результате чего получен рис. 10. При составлении результирующих уравнений следует иметь в виду тот факт, что два последовательно идущих инвертора можно удалить из схемы реализации кофакторов.

Таблица 17
Варианты алгебраического представления нетривиальных кофакторов второго уровня BDD

Номер уравнения	Уравнение
58	$w_0 = w_3 \oplus w_4$
59	$w_0 = w_3 \& \bar{w}_4$
60	$w_0 = \bar{w}_3 \oplus \bar{w}_4$
61	$w_0 = w_5 \& \bar{w}_4$
62	$w_2 = w_1 \oplus \bar{x}_3$
63	$w_2 = w_3 \oplus w_5$
64	$w_2 = w_3 \& w_1$
65	$w_2 = w_4 \& w_1$

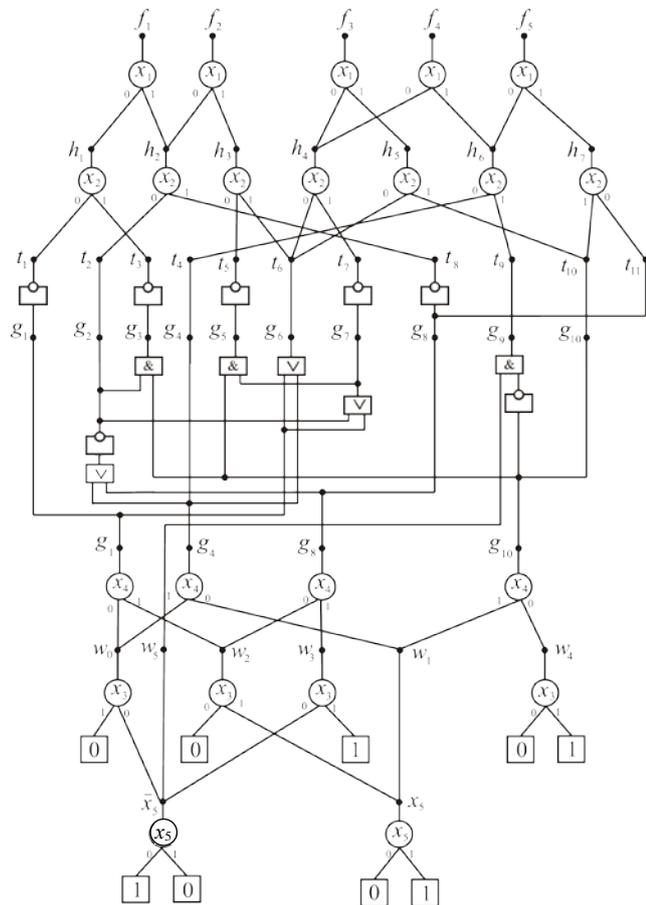


Рис. 10. Результирующий граф BDD

Предложенный метод оптимизации многоуровневых представлений векторных булевых функций может быть обобщен по нескольким направлениям.

Во-первых, может быть расширен список алгебраических выходных функций φ , по которым осуществляется разложение $g_p = \varphi(g_i, g_j)$ кофакторов. К выходным функциям φ , определяющим логические операции $\vee, \&, \oplus$, могут быть добавлены другие булевы функции, существенно зависящие от двух либо большего числа переменных.

Во-вторых, на этапах 6 и 7 метода уравнения могут составляться с применением не одного, а нескольких нижних уровней BDD.

В-третьих, желательно проверять, позволяет ли использование операции \oplus в качестве выходной функции алгебраического представления кофактора g_p сократить сложность BDD нижнего уровня (за счет исключения реализации на BDD кофактора g_p) на пять и более литералов, так как будем считать, что сложность реализации операции $g_p = g_i \oplus g_j = \bar{g}_i g_j \vee g_i \bar{g}_j$ составляет четыре литерала.

Поиск алгебраических представлений кофакторов может осуществляться в процессе построения BDD для векторной булевой функции, которая задана в виде системы ДНФ компонентных функций. (Методы, алгоритмы и программы построения BDD по системе ДНФ булевых функций описаны в работе [21].) В этом случае приходится проверять возможность алгебраического представления кофакторов, заданных произвольными (неортогонализированными) ДНФ, элементарные конъюнкции которых находятся в различных отношениях, таких как поглощение, склеивание, ортогональность, неортогональность [2]. При использовании метода, когда нижние уровни BDD еще не построены, этапы 6 и 7 метода будут отсутствовать. Метод станет более быстродействующим, однако качество решений может ухудшиться.

Оценки сложности функциональных описаний и логических схем для различных способов оптимизации. В табл. 18 для различных способов логической оптимизации одной и той же векторной булевой функции (см. табл. 1) даны оценки сложности функциональных и схемных реализаций (логических схем). Для получения минимизированной системы ДНФ (см. строку 1 табл. 18) использовалась программа [29], минимизация BDD-представлений осуществлялась во всех случаях программой из работы [30].

Таблица 18

Сравнение способов логической оптимизации

Функциональное описание	Сложность функционального описания				Сложность логической схемы из библиотечных элементов		
	α	d	k	P	Площадь	Число логических элементов	Задержка схемы, нс
1. Минимизированная система ДНФ (табл. 1), программа [29]	0	38	119	162	17 443	47	3,96
2. Булева сеть, полученная программой BoolNetOpt [31]	60	25	40	130	16 556	46	3,29
3. BDD по перестановке $\langle x_1, x_2, x_4, x_3, x_5 \rangle$, полученная программой [30]	26	28	55	114	14 793	41	3,62
4. BDD с нахождением инверсий кофакторов [24] по перестановке $\langle x_1, x_2, x_5, x_3, x_4 \rangle$, формулы (13)	21	23	47	96	15 797	44	2,88
5. Исходная BDD по перестановке $\langle x_1, x_2, x_5, x_3, x_4 \rangle$, формулы (2)	24	26	51	106	13 827	38	3,36
6. Алгебраическое представление кофакторов третьего уровня BDD без инверсий кофакторов и без операции \oplus , формулы (14)	21	21	42	89	14 486	42	3,66
7. Предложенный метод, формулы (12)	23	20	38	86	13 492	35	4,49

В строке 2 табл. 18 применялась предварительная логическая минимизация на основе аппарата булевых сетей [31]; в строке 3 – оптимизация BDD по перестановке $\langle x_1, x_2, x_4, x_3, x_5 \rangle$ с использованием программы [24], в строке 4 – оптимизация с использованием программы BDD-Builder, которая при построении BDD находит взаимно инверсные кофакторы на каждом уровне BDD и получает следующие формулы:

$$\begin{aligned}
 f_1 &= \bar{x}_1 s_6 \vee x_1 s_2; f_2 = \bar{x}_1 s_2 \vee x_1 s_3; f_3 = \bar{x}_1 s_0 \vee x_1 s_5; f_4 = \bar{x}_1 s_0 \vee x_1 s_1; f_5 = \bar{x}_1 s_1 \vee x_1 s_9; \\
 s_0 &= \bar{x}_2 s_{11} \vee x_2 s_{23}; s_1 = \bar{x}_2 s_{18} \vee x_2 s_{19}; s_2 = \bar{x}_2 s_{12} \vee x_2 s_{13}; s_3 = \bar{x}_2 s_{10} \vee x_2 s_{11}; \\
 s_5 &= \bar{x}_2 s_{11} \vee x_2 s_{15}; s_6 = \bar{x}_2 s_{20} \vee x_2 s_{21}; s_9 = \bar{x}_2 s_{13} \vee x_2 s_{15}; s_{10} = \bar{x}_5 s_{30} \vee x_5 \bar{x}_4; \\
 s_{11} &= \bar{x}_5 \bar{x}_3 \vee x_5 \bar{s}_{27}; s_{12} = \bar{x}_5 \bar{x}_4 \vee x_5 s_{27}; s_{13} = \bar{x}_5 \bar{x}_4 \vee x_5 \bar{x}_3; s_{15} = \bar{x}_5 s_{30} \vee x_5 s_{31}; \\
 s_{18} &= \bar{x}_5 s_{27} \vee x_5 \bar{x}_4; s_{19} = \bar{x}_5 s_{30}; s_{20} = \bar{x}_5 s_{31} \vee x_5 s_{42}; s_{21} = \bar{x}_5 s_{30} \vee x_5 s_{27}; \\
 s_{23} &= \bar{x}_5 x_4 \vee x_5 \bar{x}_4; s_{27} = \bar{x}_3 x_4; s_{30} = x_3 \bar{x}_4; s_{31} = \bar{x}_3 x_4 \vee x_3; s_{42} = \bar{x}_3 \vee x_3 \bar{x}_4.
 \end{aligned} \tag{13}$$

Строка 5 в табл. 18 соответствует оценке сложности функционального описания согласно формулам (2), строка 6 – формулам (14), полученным в результате применения предложенного в статье метода без поиска инверсий кофакторов и без использования операции \oplus :

$$\begin{aligned}
 f_1 &= \bar{x}_1 h_1 \vee x_1 h_2; f_2 = \bar{x}_1 h_2 \vee x_1 h_3; f_3 = \bar{x}_1 h_4 \vee x_1 h_5; f_4 = \bar{x}_1 h_4 \vee x_1 h_6; f_5 = \bar{x}_1 h_6 \vee x_1 h_7; \\
 h_1 &= \bar{x}_2 \bar{g}_1 \vee x_2 \bar{g}_3; h_2 = \bar{x}_2 g_2 \vee x_2 \bar{g}_8; h_3 = \bar{x}_2 \bar{g}_5 \vee x_2 g_6; h_4 = \bar{x}_2 g_6 \vee x_2 \bar{g}_7; h_5 = \bar{x}_2 g_6 \vee x_2 g_{10}; \\
 h_6 &= \bar{x}_2 g_4 \vee x_2 g_9; h_7 = \bar{x}_2 g_8 \vee x_2 g_{10}; g_1 = \bar{x}_4 r_0 \vee x_4 r_2; g_2 = \bar{x}_4 \bar{x}_5 \vee x_4 r_1; g_3 = g_2 \& g_{10}; \\
 g_7 &= g_1 \vee g_2; g_6 = g_1 \vee g_4; g_5 = g_7 \& g_{10}; g_9 = \bar{x}_4 r_0 \vee x_4 \bar{x}_5; g_8 = \bar{x}_4 r_2 \vee x_4 r_3; \\
 g_{10} &= \bar{x}_4 x_3 \vee x_4 x_5; g_4 = \bar{x}_4 x_5 \vee x_4 r_0; r_0 = \bar{x}_3 \bar{x}_5; r_1 = \bar{x}_3 x_5; r_2 = x_3 x_5; r_3 = \bar{x}_3 \bar{x}_5 \vee x_3.
 \end{aligned} \tag{14}$$

Все полученные представления векторной функции были записаны на языке VHDL [32], затем по данным пяти VHDL-описаний были синтезированы логические схемы в одной и той же библиотеке логических элементов и одинаковых установках опций, управляющих процессом синтеза в синтезаторе LeonardoSpectrum [33].

В табл. 18 используются следующие обозначения: α – число внутренних переменных в функциональном описании, d – число двухоперандных операций дизъюнкции, k – число двухоперандных операций конъюнкции, P – число литералов в функциональном описании. Площадь схемы (суммарная площадь всех элементов) подсчитывалась в условных единицах. Лучшие решения отмечены жирным шрифтом.

Для рассмотренного примера векторной булевой функции применение предложенного в статье метода позволяет получать как более простое функциональное описание, так и менее сложную (по площади) логическую схему, синтезируемую по этому описанию. Рост числа уровней в минимизированном функциональном описании может привести в результате синтеза схемы к увеличению ее задержки, что соответствует общей закономерности, связывающей площади логических схем и их задержки при синтезе многоуровневых логических схем. Сравним (табл. 18) лучшие решения по задержке (строка 4) и площади (строка 7): схема с площадью 13 492 выигрывает 17 % по площади у схемы с площадью 15 797, имеющей задержку 2,88 нс, и проигрывает 36 % по быстродействию, так как задержка схемы с меньшей площадью составляет 4,49 нс.

Заключение. Предложенный в статье метод позволяет дополнительно упрощать многоуровневые BDD-представления систем полностью определенных булевых функций, используя алгебраические разложения подфункций, находящихся на одном уровне BDD, в виде двухоперандных дизъюнкций, конъюнкций и сумм по модулю два. Метод легко обобщается на случай исходного задания функций в виде систем ДНФ и на случай других видов $g_p = \varphi(g_i, g_j)$ разложений подфункций g_p , когда в качестве выходных используются булевы функции φ , существенно зависящие от двух и большего числа переменных. Дополнительная оптимизация по числу литералов в логических уравнениях, соответствующих многоуровневым представлениям систем булевых функций, приводит к более простым функциональным описаниям, по которым осуществляется синтез комбинационных логических схем в различных технологических базисах. Однако уменьшение числа литералов в функциональных описаниях, позволяющее со-

кращать площадь схем, может приводить к увеличению числа логических уровней в минимизированных описаниях и увеличению временных задержек схем.

Вклад авторов. Бибило П. Н. – научное руководство исследованием, написание текста, утверждение окончательного варианта статьи для публикации, согласие нести ответственность за все аспекты работы; Романов В. И. – программная реализация алгоритмов, проведение вычислительных экспериментов, интерпретация полученных результатов.

Список использованных источников

1. Logic Minimization Algorithm for VLSI Synthesis / K. R. Brayton [et al.]. – Boston : Kluwer Academic Publishers, 1984. – 193 p.
2. Закревский, А. Д. Логический синтез каскадных схем / А. Д. Закревский. – М. : Наука, 1981. – 416 с.
3. Брейтон, Р. К. Синтез многоуровневых комбинационных логических схем / Р. К. Брейтон, Г. Д. Хэчтел, А. Л. Санджованни-Винчензелли // ТИИЭР. – 1990. – Т. 78, № 2. – С. 38–83.
4. Тарасов, И. Е. ПЛИС Xilinx. Языки описания аппаратуры VHDL и Verilog, САПР, приемы проектирования / И. Е. Тарасов. – М. : Горячая линия – Телеком, 2020. – 538 с.
5. Синтез асинхронных автоматов на ЭВМ / под ред. А. Д. Закревского. – Минск : Наука и техника, 1975. – 184 с.
6. Brayton, K. R. Factoring logic functions / K. R. Brayton // IBM J. of Research and Development. – 1987. – Vol. 31, no. 2. – P. 187–198.
7. MIS: a multiple-level logic optimization systems / R. K. Brayton [et al.] // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. – 1987. – Vol. 6, iss. 6. – P. 1062–1081.
8. Boolean decomposition in multilevel logic optimization / S. Devadas [et al.] // IEEE J. of Solid-State Circuits. – 1989. – Vol. 24, no. 2. – P. 399–407.
9. Brayton, R. Boolean factoring and decomposition of logic networks / R. Brayton, A. Mishchenko, S. Chatterjee; eds.: S. P. Khatri, K. Gulati // Advanced Techniques in Logic Synthesis, Optimizations and Applications. – Springer, 2011. – P. 47–66.
10. Curtis, H. A. A New Approach to the Design of Switching Circuit / H. A. Curtis. – Princeton, Van Nostrand, 1962. – 635 p.
11. Scholl, C. Functional Decomposition with Applications to FPGA Synthesis / C. Scholl. – Kluwer Academic Publishers, 2001. – 288 p.
12. Поттосин, Ю. В. Табличные методы декомпозиции систем полностью определенных булевых функций / Ю. В. Поттосин, Е. А. Шестаков. – Минск : Беларус. навука, 2006. – 327 с.
13. Бибило, П. Н. Декомпозиция булевых функций на основе решения логических уравнений / П. Н. Бибило. – Минск : Беларус. навука, 2009. – 211 с.
14. Lin, H.-P. Ashenurst decomposition using SAT and interpolation / H.-P. Lin, J.-H. R. Jiang, R.-R. Lee; eds.: S. P. Khatri, K. Gulati // Advanced Techniques in Logic Synthesis, Optimizations and Applications. – Springer, 2011. – P. 67–86.
15. Bryant, R. E. Graph-based algorithms for Boolean function manipulation / R. E. Bryant // IEEE Transactions on Computers. – 1986. – Vol. 35, no. 8. – P. 677–691.
16. Drechsler, R. Binary Decision Diagrams: Theory and Implementation / R. Drechsler, B. Becker. – Springer, 1998. – 210 p.
17. Ebdndt, R. Advanced BDD Optimization / R. Ebdndt, G. Fey, R. Drechsler. – Springer, 2005. – 222 p.
18. Bryant, R. E. Ordered binary decision diagrams / R. E. Bryant, C. Meinel // Logic Synthesis and Verification / eds.: S. Hassoun, T. Sasao, R. K. Brayton. – Kluwer Academic Publishers, 2002. – P. 285–307.
19. Meinel, C. Algorithms and Data Structures in VLSI Design: OBDD – Foundations and Applications / C. Meinel, T. Theobald. – Berlin, Heidelberg : Springer-Verlag, 1998. – 267 p.
20. Кнут, Д. Э. Искусство программирования. Т. 4, А. Комбинаторные алгоритмы. Ч. 1 : пер. с англ. / Д. Э. Кнут. – М. : Вильямс, 2013. – 960 с.
21. Бибило, П. Н. Применение диаграмм двоичного выбора при синтезе логических схем / П. Н. Бибило. – Минск : Беларус. навука, 2014. – 231 с.
22. Уткин, А. А. Анализ логических сетей и техника булевых вычислений / А. А. Уткин. – Минск : Наука и техника, 1979. – 152 с.
23. Романов, В. И. Программные средства для решения логико-комбинаторных задач / В. И. Романов // Информатика. – 2005. – № 4(8). – С. 114–123.

24. Бибило, П. Н. Использование полиномов Жегалкина при минимизации многоуровневых представлений систем булевых функций на основе разложения Шеннона / П. Н. Бибило, Ю. Ю. Ланкевич // Программная инженерия. – 2017. – № 8. – С. 369–384.
25. Goldberg, E. BerkMin: a fast and robust SAT-solver / E. Goldberg, Y. Novikov // Discrete Applied Mathematics. – 2007. – Vol. 155, no. 12. – P. 1549–1561.
26. Handbook of Satisfiability / eds.: A. Biere, M. Heule, H. van Maaren, T. Walsh. – IOS Press, 2009. – 980 p.
27. Романовский, И. В. Дискретный анализ : учебное пособие / И. В. Романовский. – 4-е изд., испр. и доп. – СПб. : Невский Диалект, БХВ-Петербург, 2008. – 336 с.
28. Бибило, П. Н. Синтез комбинационных схем методами функциональной декомпозиции / П. Н. Бибило, С. В. Енин. – Минск : Наука и техника, 1987. – 189 с.
29. 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.
30. Бибило, П. Н. Минимизация многоуровневых представлений систем булевых функций на основе разложения Шеннона / П. Н. Бибило, Ю. Ю. Ланкевич // Информатика. – 2017. – № 2(54). – С. 45–57.
31. Бибило, П. Н. Логическая минимизация булевых сетей с использованием разложения Шеннона / П. Н. Бибило, Ю. Ю. Ланкевич // Информатика. – 2019. – Т. 16, № 2. – С. 73–89.
32. Ashenden, P. J. VHDL–2008. Just the New Stuff / P. J. Ashenden, J. Lewis. – Burlington, MA, USA : Morgan Kaufman Publishers, 2008. – 909 p.
33. Бибило, П. Н. Системы проектирования интегральных схем на основе языка VHDL. StateCAD, ModelSim, LeonardoSpectrum / П. Н. Бибило. – М. : СОЛОН-Пресс, 2005. – 384 с.

References

1. Brayton K. R., Hachtel G. D., McMullen C., Sangiovanni-Vincentelli A. L. *Logic Minimization Algorithm for VLSI Synthesis*. Boston, Kluwer Academic Publishers, 1984, 193 p.
2. Zakrevskij A. D. Logicheskij sintez kaskadnyh skhem. *Logical Synthesis of Cascading Circuit*. Moscow, Nauka, 1981, 416 p. (In Russ.).
3. Brayton R. K., Hachtel G. D., Sangiovanni-Vincentelli A. L. *Synthesis of multi-level combinational logic circuits*. Trudy Institute inzhenerov po jelektronike i radiotekhnike [Proceedings of the Institute of Electronics and Radio Engineering], 1990, vol. 78, no. 2, pp. 38–83 (In Russ.).
4. Tarasov I. E. PLIS Xilinx. Yazyki opisaniya apparatury VHDL i Verilog, SAPR, priemy proektirovaniya. *XILINX FPGA. Hardware Description Languages VHDL and Verilog, CAD, Design Techniques*. Moscow, Goryachaya liniya – Telekom, 2020, 538 p.
5. Sintez asinhronnyh avtomatov na EHVM. *Synthesis of Asynchronous Automata on a Computer*. In Zakrevskogo A. D. (ed.). Minsk, Nauka i tekhnika, 1975, 184 p.
6. Brayton K. R. Factoring logic functions. *IBM Journal of Research and Development*, 1987, vol. 31, no. 2, pp. 187–198.
7. Brayton R. K., Rudell R., Sangiovanni-Vincentelli A. L., Wang A. R. MIS: a multiple-level logic optimization systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 1987, vol. 6, iss. 6, pp. 1062–1081.
8. Devadas S., Wang A. R., Newton A. R., Sangiovanni-Vincentelli A. Boolean decomposition in multilevel logic optimization. *IEEE Journal of Solid-State Circuits*, 1989, vol. 24, no. 2, pp. 399–407.
9. Brayton R., Mishchenko A., Chatterjee S. Boolean factoring and decomposition of logic networks. In Khatri S. P., Gulati K. (eds.). *Advanced Techniques in Logic Synthesis, Optimizations and Applications*. Springer, 2011, pp. 47–66.
10. Curtis H. A. *A New Approach to the Design of Switching Circuit*. Princeton, Van Nostrand, 1962, 635 p.
11. Scholl C. *Functional Decomposition with Applications to FPGA Synthesis*. Kluwer Academic Publishers, 2001, 288 p.
12. Pottosin Yu. V., Shestakov E. A. Tablichnye metody dekompozicii sistem polnost'yu opredelennyh bulevykh funkciy. *Tabular methods for decomposition of systems of completely defined Boolean functions*. Minsk, Belaruskaja navuka, 2006, 327 p. (In Russ.).
13. Bibilo P. N. Dekompoziciya bulevykh funkciy na osnove resheniya logicheskikh uravnenij. *Decomposition of Boolean Functions Based on the Solution of Logical Equations*. Minsk, Belaruskaja navuka, 2009, 211 p. (In Russ.).

14. Lin H.-P., Jiang J.-H. R., Lee R.-R. Ashenurst decomposition using SAT and interpolation. In Khatri S. P., Gulati K. (eds.). *Advanced Techniques in Logic Synthesis, Optimizations and Applications*. Springer, 2011, pp. 67–86.
15. Bryant R. E. Graph-based algorithms for Boolean function manipulation. *IEEE Transactions on Computers*, 1986, vol. 35, no. 8, pp. 677–691.
16. Drechsler R., Becker B. *Binary Decision Diagrams: Theory and Implementation*. Springer, 1998, 210 p.
17. Ebdndt R., Fey G., Drechsler R. *Advanced BDD Optimization*. Springer, 2005, 222 p.
18. Bryant R. E., Meinel C. Ordered binary decision diagrams. In S. Hassoun, T. Sasao, R. K. Brayton (eds.). *Logic Synthesis and Verification*. Kluwer Academic Publishers, 2002, pp. 285–307.
19. Meinel C., Theobald T. *Algorithms and Data Structures in VLSI Design: OBDD – Foundations and Applications*. Berlin, Heidelberg, Springer-Verlag, 1998, 267 p.
20. Knuth D. E. *The Art of Computer Programming, Volume 4A: Combinatorial Algorithms, Part 1*. Addison-Wesley Professional, 2011, 912 p.
21. Bibilo P. N. Primenenie diagram dvoichnogo vybora pri sinteze logicheskikh shem. *Application of Binary Selection Diagrams in the Synthesis of Logic Circuits*. Minsk, Belaruskaja navuka, 2014, 231 p. (In Russ.).
22. Utkin A. A. Analiz logicheskikh setej i tekhnika bulevykh vychislenij. *The analysis of logical networks and the technique of Boolean calculations*. Minsk, Nauka i tekhnika, 1979, 152 p. (In Russ.).
23. Romanov V. I. *Software tools for the solution of the logical-combinatorial task*. Informatika [Informatics], 2005, no. 4(8), pp. 114–123 (In Russ.).
24. Bibilo P. N., Lankevich Yu. Yu. *The use of Zhegalkin polynomials in minimizing multilevel representations of systems of Boolean functions based on the Shannon expansion*. Programmnyaya inzheneriya [Software Engineering], 2017, no. 8, pp. 369–384 (In Russ.).
25. Goldberg E., Novikov Y. BerkMin: a fast and robust SAT-solver. *Discrete Applied Mathematics*, 2007, vol. 155, no. 12, pp. 1549–1561.
26. *Handbook of Satisfiability*. In Biere A., Heule M., Van Maaren H., Valsh T. (eds.). IOS Press, 2009, 980 p.
27. Romanovskij I. V. Diskretnyj analiz. *Discrete Analysis*. Saint Petersburg, Nevskij Dialekt, BHV-Peterburg, 2008, 336 p. (In Russ.).
28. Bibilo P. N., Enin S. V. Sintez kombinacionnykh skhem metodami funkcional'noj dekompozicii. *Synthesis of Combination Circuits by Methods of Functional Decomposition*. Minsk, Nauka i tekhnika, 1987, 189 p. (In Russ.).
29. Toropov N. R. *Minimization of systems of Boolean functions in the class DNF*. Logicheskoe proektirovanie [Logical Design]. Minsk, Institut tehnikeskoy kibernetiki Nacional'noj akademii nauk Belarusi, 1999, iss. 4, pp. 4–19 (In Russ.).
30. Bibilo P. N., Lankevich Yu. Yu. *Minimizing multilevel representations of systems of Boolean functions based on Shannon expansion*. Informatika [Informatics], 2017, no. 2(54), pp. 45–57 (In Russ.).
31. Bibilo P. N., Lankevich Yu. Yu. *Logical optimization of Boolean nets using Shannon expansion*. Informatika [Informatics], 2019, vol. 16, no. 2, pp. 73–89 (In Russ.).
32. Ashenden P. J., Lewis J. *VHDL–2008. Just the New Stuff*. Burlington, MA, USA, Morgan Kaufman Publishers, 2008, 909 p.
33. Bibilo P. N. Cistemy proektirovaniya integral'nykh skhem na osnove yazyka VHDL. StateCAD, ModelSim, LeonardoSpectrum. *Integrated Circuit Design Systems Based on the VHDL Language*. StateCAD, ModelSim, LeonardoSpectrum. Moscow, SOLON-Press, 2005, 384 p. (In Russ.).

Информация об авторах

Бибилу Петр Николаевич, доктор технических наук, профессор, Объединенный институт проблем информатики Национальной академии наук Беларуси. E-mail: bibilo@newman.bas-net.by

Романов Владимир Ильич, кандидат технических наук, доцент, Объединенный институт проблем информатики Национальной академии наук Беларуси. E-mail: rom@newman.bas-net.by

Information about the authors

Petr N. Bibilo, Dr. Sci. (Eng.), Professor, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus. E-mail: bibilo@newman.bas-net.by

Vladimir I. Romanov, Cand. Sci. (Eng.), The United Institute of Informatics Problems of the National Academy of Sciences of Belarus. E-mail: rom@newman.bas-net.by



УДК 519.711
<https://doi.org/10.37661/1816-0301-2021-18-2-33-47>

Оригинальная статья
Original Paper

Минимизация булевых функций в классе ортогональных дизъюнктивных нормальных форм

Ю. В. Поттосин

*Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
E-mail: pott@newman.bas-net.by*

Аннотация. Ортогональные дизъюнктивные нормальные формы (ДНФ) булевых функций имеют широкое применение в логическом проектировании дискретных устройств. Задача ортогонализации ДНФ состоит в том, чтобы для заданной функции получить ДНФ, любые две элементарные конъюнкции которой ортогональны, т. е. их конъюнкция тождественно равна нулю. Предлагается подход к решению данной задачи с помощью средств теории графов. Подход рассчитан на представление функции в виде совершенной ДНФ. Предполагается получение всех интервалов булева пространства, на которых заданная функция имеет значение 1, и рассматривается граф пересечения этих интервалов. Рассматриваются два метода получения минимальной ортогональной ДНФ. Один из них сводит данную задачу к получению наименьшего доминирующего множества в графе путем покрытия его вершин их замкнутыми окрестностями, другой – к получению максимального независимого множества с помощью лексикографического перебора. Показывается, как предлагаемый подход распространяется на не полностью определенные булевы функции.

Ключевые слова: булева функция, дизъюнктивная нормальная форма, ортогональность элементарных конъюнкций, задача о кратчайшем покрытии, граф пересечения, доминирующее множество, независимое множество

Для цитирования. Поттосин, Ю. В. Минимизация булевых функций в классе ортогональных дизъюнктивных нормальных форм / Ю. В. Поттосин // Информатика. – 2021. – Т. 18, № 2. – С. 33–47. <https://doi.org/10.37661/1816-0301-2021-18-2-33-47>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 03.03.2021
Подписана в печать | Accepted 15.04.2021
Опубликована | Published 26.06.2021

Minimization of Boolean functions in the class of orthogonal disjunctive normal forms

Yuri V. Pottosin

*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus
E-mail: pott@newman.bas-net.by*

Abstract. The orthogonal disjunctive normal forms (DNFs) of Boolean functions have wide applications in the logical design of discrete devices. The problem of DNF orthogonalization is to get for a given function such a DNF that any two its terms would be orthogonal, i. e. the conjunction of them would be equal identically to zero. An approach to solve the problem using the means of graph theory is suggested. The approach is proposed by representation of the function as perfect DNF. Obtaining all the intervals of the Boolean space where the given function has value 1 is supposed, and the intersection graph of those intervals is considered. Two methods to obtain a minimum orthogonal DNF are considered. One of them reduces the problem toward finding out the smallest dominating set in the graph by covering its vertices with their closed neighborhoods, the other – to obtain the maximum independent set by lexicographic enumeration. It is shown how the suggested approach can be extended on incompletely specified Boolean functions.

Keywords: Boolean function, disjunctive normal form, orthogonal terms, short cover problem, intersection graph, dominating set, independent set

For citation. Pottosin Yu. V. Minimization of Boolean functions in the class of orthogonal disjunctive normal forms. *Informatics*, 2021, vol. 18, no. 2, pp. 33–47 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-33-47>

Conflict of interest. The author declare of no conflict of interest.

Введение. Под ортогональной ДНФ булевой функции понимается такая ДНФ, в которой все ее члены – элементарные конъюнкции – попарно ортогональны, т. е. конъюнкция двух любых ее членов тождественно равна нулю. Более общей задачей является ортогонализация системы булевых функций, которая формулируется следующим образом. Пусть задана некоторая система полностью определенных булевых функций $F = \{f_1, f_2, \dots, f_m\}$ и требуется найти совокупность взаимно ортогональных булевых функций $\varphi_1, \varphi_2, \dots, \varphi_N$, такую, что любую $f_i \in F$ можно было бы выразить дизъюнкцией некоторых из φ_j ($j = 1, 2, \dots, N$), причем общее число N этих функций было бы минимальным. Под ортогональностью здесь также понимается справедливость выражения $\varphi_j \wedge \varphi_k = 0$ при любых значениях аргументов. В таком виде эта задача была поставлена в статье [1]. Ортогонализацию ДНФ можно считать частным случаем ортогонализации системы булевых функций, где каждая функция из заданной системы представляется одной элементарной конъюнкцией.

Задача ортогонализации системы булевых функций имеет ряд приложений. Одним из ее приложений является минимизация числа столбцов таблицы переходов автомата при получении ее по заданному микропрограммному автомату [2] или по автомату с абстрактным состоянием [3]. Метод минимизации числа столбцов таблицы переходов автомата описан в работе [4]. Ортогональные ДНФ используются при синтезе схем из программируемых логических матриц [5]. Для вычисления вероятности сложного события удобно использовать ортогональную ДНФ алгебры событий [6, 7]. Как показано в статье [8], применение ортогональных ДНФ при синтезе логических схем позволяет улучшить их диагностические свойства и обеспечивает построение самопроверяемых схем.

В работе [1] представлены два метода ортогонализации системы булевых функций. Один из них заключается в нахождении всех 2^m конъюнкций вида $f_1^{\sigma_1} f_1^{\sigma_2} \dots f_1^{\sigma_m}$, где $\sigma_1, \sigma_2, \dots, \sigma_m$ – константы 0 или 1 и $f_i^{\sigma_i} = \bar{f}_i$, если $\sigma_i = 0$, и $f_i^{\sigma_i} = f_i$, если $\sigma_i = 1$. Ненулевые конъюнкции составят решение. Другой метод рассчитан на задание каждой функции множеством наборов значений аргументов, на которых она принимает значение 1, и заключается в получении всех нену-

стных пересечений этих множеств и их дополнений. Там же доказано, что число получаемых взаимно ортогональных функций минимально.

Ясно, что задача минимизации булевых функций в классе ортогональных ДНФ имеет неполиномиальную сложность. Были предложены эвристические методы ортогонализации системы функций и ДНФ [6, 7, 9–12]. Рассматриваемый в работах [6, 7, 9, 12] метод основан на идее дизъюнктивного разложения элементарной конъюнкции на серию других конъюнкций, каждая из которых либо ортогональна всем конъюнкциям из некоторой совокупности, либо поглощается одной из них. Ряд методов основан на дизъюнктивном разложении Шеннона, они описаны в работах [6, 7, 10–12]. В статье [13] представлен метод, применяющий аппарат, который связан с понятием покрытия троичной матрицы, используемого при декомпозиции булевых функций. Для не полностью определенных функций данная задача рассматривалась в докладе [14]. Предложенный в нем метод ортогонализации ДНФ не гарантирует минимальности получаемой ДНФ. В работе [15] задача ортогонализации системы не полностью определенных булевых функций, как она поставлена для полностью определенных функций в статье [1], сведена к раскраске графа.

Далее представлен подход к решению задачи минимизации ортогональной ДНФ, обеспечивающий точное решение с помощью средств теории графов.

Описание подхода. Предусматривается задание булевой функции в виде совершенной ДНФ, т. е. в виде двоичной матрицы, где строки представляют наборы значений аргументов, на которых она принимает значение 1. С помощью операции простого склеивания получим все интервалы пространства аргументов, где заданная функция имеет значение 1. Понятие интервала связано с отношением «больше» или «меньше» на множестве булевых векторов [7]. Считается, что $1 > 0$, и полагается, что между булевыми векторами $\mathbf{a} = (a_1, a_2, \dots, a_n)$ и $\mathbf{b} = (b_1, b_2, \dots, b_n)$ имеет место соотношение $\mathbf{a} \geq \mathbf{b}$, если $a_i > b_i$ или $a_i = b_i$ для любой пары одноименных компонент векторов. Интервал – это множество векторов, где есть минимальный вектор, максимальный вектор и все векторы, меньшие максимального и большие минимального. Процесс получения всех интервалов, на которых заданная функция принимает значение 1, присутствует также в классическом методе Квайна – МакКласки минимизации ДНФ. Интервал соответствует элементарной конъюнкции переменных с отрицаниями или без них. Если элементарные конъюнкции ортогональны, то соответствующие интервалы не пересекаются и, наоборот, пересекающиеся интервалы соответствуют неортогональным элементарным конъюнкциям. В методе Квайна – МакКласки рассматриваются только максимальные интервалы, т. е. те, которые не являются собственными подмножествами других интервалов. Задача минимизации ДНФ сводится к получению кратчайшего покрытия элементов булева пространства, где заданная функция имеет значение 1, максимальными интервалами. В случае минимизации ортогональной ДНФ надо рассматривать все интервалы и находить такое покрытие, элементы которого не пересекаются.

В терминах теории графов задачу можно сформулировать следующим образом. Отношение пересечения на множестве всех интервалов, куда включаются и одноэлементные интервалы, т. е. отдельные векторы-строки заданной матрицы, представим в виде неориентированного графа, вершинам которого соответствуют интервалы. Две его вершины связаны ребром, если и только если соответствующие интервалы находятся в отношении пересечения. Назовем такой граф *графом пересечения интервалов*.

Независимое доминирующее множество в полученном графе соответствует ортогональной ДНФ. Независимым множеством графа называется множество вершин, никакие две из которых не связаны ребром [16]. Доминирующим множеством графа является такое множество вершин, что если какая-то вершина не содержится в нем, то она смежна с некоторой вершиной из этого множества [16]. Следовательно, независимым доминирующим множеством графа является доминирующее множество, обладающее свойством независимости. Минимальная ортогональная ДНФ соответствует *наименьшему независимому доминирующему множеству* графа пересечения интервалов.

Любому независимому множеству в графе пересечения интервалов соответствует ортогональная ДНФ. Независимое множество максимальное в графе, если оно не является собственным подмножеством другого независимого множества. Минимальная ортогональная ДНФ со-

ответствует независимому множеству графа пересечения интервалов, обладающему минимальной мощностью. Такое множество назовем *наименьшим максимальным независимым множеством*. Следует заметить, что всякое максимальное независимое множество является доминирующим. Действительно, если предположить, что какая-то вершина в заданном графе не вошла в некоторое максимальное независимое множество S , то она смежна по крайней мере с одной вершиной из S . Таким образом, наименьшее независимое доминирующее множество является также наименьшим максимальным независимым множеством. В решении задачи минимизации ортогональной ДНФ можно использовать методы нахождения как доминирующих, так и независимых множеств.

Нахождение наименьшего независимого доминирующего множества в графе. Доминирующее множество графа $G = (V, E)$ с множеством вершин $V = \{v_1, v_2, \dots, v_n\}$ и множеством ребер E находится с помощью решения задачи о кратчайшем покрытии [7]. Множество вершин V покрывается его подмножествами вида $\{v_i\} \cup N(v_i)$, где $N(v_i)$ – окрестность вершины v_i , т. е. множество вершин, смежных с v_i . Множество $\{v_i\} \cup N(v_i)$ называют замкнутой окрестностью вершины v_i и обозначают $N[v_i]$. Для получения независимого доминирующего множества к такому покрытию добавляется еще требование, чтобы принадлежащие ему подмножества не пересекались.

Удобно рассматривать матричную формулировку задачи о кратчайшем покрытии, при которой в заданной двоичной матрице надо выделить минимальное количество строк так, чтобы в каждом столбце имелась единица хотя бы в одной из выделенных строк. При нахождении наименьшего доминирующего множества графа такой матрицей является его матрица смежности, нулевые элементы главной диагонали которой заменены единицами (i -я строка и i -й столбец матрицы представляют множество $N[v_i]$).

Точный метод решения задачи о кратчайшем покрытии представляет собой обход дерева поиска [7]. Вершины дерева поиска сопоставляются с ситуациями, которые можно достичь в данном процессе, а ребра – варианты очередного шага выполняемого процесса. Текущая ситуация, соответствующая некоторой вершине дерева поиска, представляется переменной матрицей X , содержащей столбцы заданной матрицы, которые еще не покрыты, и строки, которые можно использовать для их покрытия. В этой ситуации выбирается первый из столбцов с минимальным числом единиц, так минимизируется число вариантов продолжения поиска. Очередной шаг процесса состоит в выборе покрывающей строки для этого столбца и включении ее в получаемое решение. Таким образом, вершины дерева поиска соответствуют некоторым столбцам исходной матрицы, а ребра – выбираемым для их покрытия строкам. Включаемая в решение строка удаляется из матрицы X вместе с покрываемыми ею столбцами. Удаляются также строки, смежные со строкой, которая включается в решение. Под *смежными строками* будем понимать строки, соответствующие смежным вершинам заданного графа.

В работе [7] сформулированы два правила редукции, которые следует выполнять для сокращения размерности задачи. Одно из них позволяет исключать из рассмотрения столбцы, если в матрице X имеются другие столбцы, покрываемые ими. Это правило можно применять и при поиске независимого доминирующего множества. В исходном значении матрицы X каждый i -й столбец представляет множество $N[v_i]$. В графе пересечений интервалов $G = (V, E)$ для двух вершин v_i и v_j справедливо $N[v_i] \subseteq N[v_j]$, если вершина v_i соответствует элементу булева пространства, а v_j – интервалу, содержащему этот элемент. Следовательно, в исходной матрице X можно оставить для покрытия только те столбцы, которые соответствуют отдельным элементам булева пространства.

Второе правило, по которому из рассмотрения исключаются строки, покрываемые другими строками, неприменимо в рассматриваемой задаче из-за требования, чтобы элементы покрытия не пересекались.

Нахождение наименьшего максимального независимого множества в графе. В работе [7] описан способ перебора максимальных независимых множеств в лексикографическом порядке. Предлагаемый метод получения наименьшего максимального независимого множества заключается в выборе его в процессе данного перебора.

Пусть $V = \{v_1, v_2, \dots, v_n\}$ – множество вершин графа G . Весь процесс нахождения максимальных независимых множеств можно разбить на n этапов, каждый из которых связан с определенной вершиной $v_i \in V$. На i -м этапе находятся максимальные независимые множества, содержащие вершину v_i и не содержащие вершин с меньшими номерами, т. е. таких v_j , для которых $j < i$. В качестве решения сохраняется множество, которое обладает наименьшей мощностью.

Пусть S_i – одно из независимых множеств графа G , формируемых на i -м этапе. За начальное значение множества S_i принимается множество, состоящее из единственной вершины v_i . Множество S_i расширяется за счет поочередного включения в него элементов $v_j \in V$, удовлетворяющих условиям

$$i < j \leq n, v_j \in \bigcap_{v \in S_i} \{V \setminus N(v)\},$$

где $N(v)$ – окрестность вершины v .

Каждый раз при соблюдении данных условий выбирается v_j с минимальным j . Это расширение множества S_i продолжается до тех пор, пока выполняется одно из условий:

- 1) $\bigcap_{v \in S_i} \{V \setminus N(v)\} \neq \emptyset$;

- 2) мощность множества S_i достигает значения, на единицу меньше мощности хранимого множества, а проверка на максимальность показывает, что S_i не является максимальным.

Формируемое множество S_i проверяется на максимальность согласно следующему свойству: независимое множество S является максимальным тогда и только тогда, когда $S \cup N(S) = V$, где $N(S) = \bigcup_{v \in S} N(v)$. Множество S_i в этом случае сохраняется как промежуточное решение.

Заметим, что наибольшее независимое множество (независимое множество наибольшей мощности) графа пересечения интервалов является единственным и состоит из вершин, соответствующих одноэлементным интервалам. Поэтому для получения наименьшего максимального независимого множества указанный перебор надо начинать с вершин, соответствующих наибольшим интервалам. Искомое множество будет получено быстрее, если вершины графа пересечения интервалов упорядочить по невозрастанию мощностей соответствующих им интервалов, а внутри подмножеств вершин с одним и тем же таким показателем – по неубыванию степеней вершин.

Для проверки второго из указанных выше условий можно руководствоваться следующими соображениями. Есть возможность оценить окончательную мощность множества S_i перед тем, как вводить в него вершину v_j , соответствующую одноэлементному интервалу. Пусть m – число элементов булева пространства, на которых заданная функция имеет значение 1; k – число вершин, принадлежащих S_i ; l – число элементов, принадлежащих интервалам, соответствующим вершинам из S_i . Тогда мощность множества S_i может достигнуть величины $k + m - l$.

Чтобы построить следующее по порядку независимое множество, из полученного S_i (сохраненного или не сохраненного в качестве решения) удаляется вершина v_p , присоединенная к S_i последней, и выполняется та же процедура с вершинами v_q , где $q > p$.

На k -м этапе проверяется условие $A_k \cup B_k = V$, где $A_k = \{v_k, v_{k+1}, \dots, v_n\}$ и $B_k = \bigcup_{j=k}^n N(v_j)$. Если

оно не выполняется, то данный процесс надо прекратить, так как никакое подмножество множества A_k не составит максимального независимого множества.

Примеры получения минимальной ортогональной ДНФ. Рассмотрим булеву функцию $f(x_1, x_2, x_3, x_4, x_5)$, заданную следующим множеством векторов (одноэлементных интервалов), на которых она имеет значение 1:

$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$
1 00101	4 01100	7 01011	10 10110	13 10111
2 00110	5 10010	8 01101	11 11010	14 11011
3 01010	6 00111	9 10011	12 11100	15 11110

С помощью простого склеивания получим все интервалы, представляемые следующими троичными векторами:

$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$
16 001-1	21 -1010	26 1-010	31 1011-	36 -011-
17 0-101	22 0110-	27 -0111	32 1-110	37 -101-
18 0011-	23 -1100	28 -1011	33 1101-	38 10-1-
19 -0110	24 1001-	29 10-11	34 11-10	39 1-01-
20 0101-	25 10-10	30 1-011	35 111-0	40 1--10

Граф пересечений, содержащий 40 вершин, удобно задать перечислением списков окрестностей вершин, где вершины обозначены номерами соответствующих интервалов:

1: 16,17	21: 3,11,20,26,33,34,37,39,40
2: 18,19,36	22: 4,8,17,23
3: 20,21,37	23: 4,12,22,35
4: 22,23	24: 5,9,25,26,29,30,38,39,40
5: 24,25,26,38,39,40	25: 5,10,19,24,26,31,32,36,38,39,40
6: 16,18,27,36	26: 5,11,21,24,25,33,34,37,38,39,40
7: 20,28,37	27: 6,13,16,18,29,31,36,38
8: 17,22	28: 7,14,20,30,33,37,39
9: 24,29,30,38,39	29: 9,13,24,27,30,31,36,38,39
10: 19,25,31,32,36,38,40	30: 9,14,24,28,29,33,37,38,39
11: 21,26,33,34,37,39,40	31: 10,13,19,25,27,29,32,36,38,40
12: 23,35	32: 10,15,19,25,31,34,35,36,38,40
13: 27,29,31,36,38	33: 11,14,21,26,28,30,34,37,39,40
14: 28,30,33,37,39	34: 11,15,21,26,32,33,35,37,39,40
15: 32,34,35,40	35: 12,15,32,34,40
16: 1,6,17,18,27,36	36: 2,6,10,13,16,18,19,25,27,29,31,32,38,40
17: 1,8,16,22	37: 3,7,11,14,20,21,26,28,30,33,34,39,40
18: 2,6,16,19,27,36	38: 5,9,10,13,19,24,25,26,27,29,30,31,32,36,39,40
19: 2,10,25,31,32,36,38,40	39: 5,9,11,14,21,24,25,26,28,29,30,33,34,37,38,40
20: 3,7,21,28,37	40: 5,10,11,15,19,21,24,25,26,31,32,33,34,35,36,37,38,39

Матрицу X зададим в виде табл. 1, где по правилу редукции оставлены только те столбцы, которые соответствуют одиночным векторам, а для удобства чтения нули представлены пустыми клетками.

Одним из столбцов, обладающих минимальным числом единиц, является столбец 1, который покрывают строки 1, 16 и 17. Эти варианты выбора строки для покрытия столбца 1 определяют три ветви дерева поиска с начальными вершинами, обозначенными номерами строк. Из покрывающих строк какого-либо столбца рекомендуется выбирать в первую очередь ту, которая содержит максимальное число единиц. Выбрав строку 16 и включив ее в текущее решение, приходим к ситуации, представленной в табл. 2. Она получена из табл. 1 удалением покрываемых ею столбцов 1 и 6 и строк, соответствующих интервалам, с которыми пересекается интервал, соответствующий строке 16.

Таблица 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1														
2		1													
3			1												
4				1											
5					1										
6						1									
7							1								
8								1							
9									1						
10										1					
11											1				
12												1			
13													1		
14														1	
15															1
16	1					1									
17	1						1								
18		1				1									
19		1							1						
20			1				1								
21			1							1					
22				1			1								
23				1								1			
24					1			1							
25					1				1						
26					1					1					
27						1						1			
28							1						1		
29								1				1			
30									1					1	
31										1			1		
32										1					1
33											1			1	
34											1				1
35												1			1
36		1				1			1				1		
37			1				1			1				1	
38				1				1	1			1			
39					1				1	1				1	
40						1				1	1				1

В ситуации, представленной в табл. 2, рассмотрим два варианта покрытия столбца 2 – строку 2 и строку 19. Выбираем строку 19 и, двигаясь дальше по дереву поиска, проходим его вершины в последовательности 16, 19, 22, 35, 29, 37, 5 до получения покрытия, которое соответствует следующей ортогональной ДНФ, представленной в виде троичной матрицы (нумерация интервалов сохранена):

$$\begin{array}{ccccc|c}
 x_1 & x_2 & x_3 & x_4 & x_5 & \\
 \hline
 1 & 0 & 0 & 1 & 0 & 5 \\
 0 & 0 & 1 & - & 1 & 16 \\
 - & 0 & 1 & 1 & 0 & 19 \\
 0 & 1 & 1 & 0 & - & 22 \\
 1 & 0 & - & 1 & 1 & 29 \\
 1 & 1 & 1 & - & 0 & 35 \\
 - & 1 & 0 & 1 & - & 37
 \end{array}$$

Ветвь дерева поиска с начальной вершиной 16 изображена на рис. 1. Движение вниз по ветвям прекращается, когда видно, что длина получаемого покрытия не меньше длины уже полученного. Например, пройдя последовательность 16, 19, 22, 35, 13, видим, что в соответствующей таблице нет строки, покрывающей все столбцы. Следовательно, длина получаемого нового покрытия не может быть меньше семи. Эта ситуация представлена в табл. 3.

Таблица 2

	3	4	5	7	8	9	11	12	13	14	15
3	1										
4		1									
5			1								
7				1							
8					1						
9						1					
11							1				
12								1			
13									1		
14										1	
15											1
19											
20	1			1							
21	1						1				
22		1			1						
23		1						1			
24			1			1					
26			1				1				
28				1						1	
29						1			1		
30							1			1	
33								1			1
34								1			1
35									1		1
37	1			1			1			1	
39			1			1	1			1	

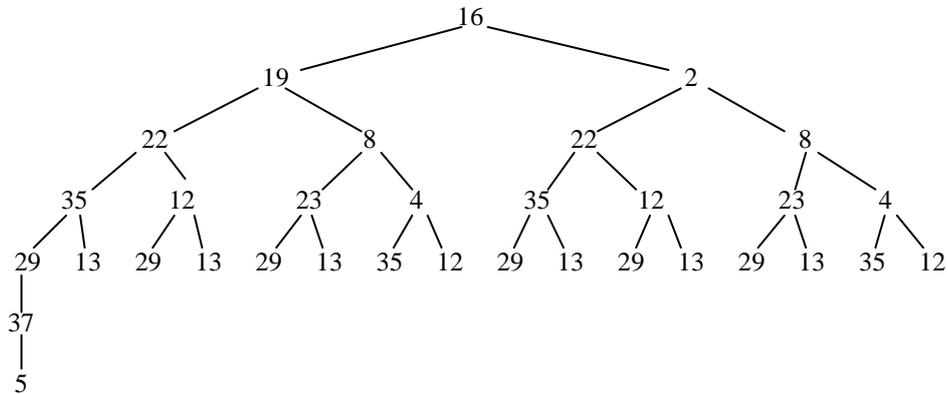


Рис. 1. Ветвь дерева поиска с начальной вершиной 16

Дальнейший обход ветви дерева поиска с начальной вершиной 16 не приводит к меньшему покрытию. Перейдя к ветви с началом в вершине 17, действуем аналогичным образом. Эта ветвь изображена на рис. 2. Двигаясь в последовательности 17, 23, 36, 34, видим, что не получим покрытия с числом строк, меньшим семи. Кратчайшее покрытие дает последовательность 17, 23, 36, 15, 37, 24. Дальнейший обход ветви с началом в вершине 17 так же, как и обход ветви с началом в вершине 1, не дает покрытия с меньшим числом строк. Минимальная ортогональная ДНФ заданной функции представлена матрицей

$$\begin{array}{ccccc|l}
 x_1 & x_2 & x_3 & x_4 & x_5 & \\
 \hline
 1 & 1 & 1 & 1 & 0 & 15 \\
 0 & - & 1 & 0 & 1 & 17 \\
 - & 1 & 1 & 0 & 0 & 23 \\
 1 & 0 & 0 & 1 & - & 24 \\
 - & 0 & 1 & 1 & - & 36 \\
 - & 1 & 0 & 1 & - & 37
 \end{array}$$

Таблица 3

	3	5	7	9	11	13	14
3	1						
5		1					
7			1				
9				1			
11					1		
13						1	
14							1
20	1		1				
21	1				1		
24		1		1			
26		1			1		
28			1				1
29				1		1	
30				1			1
33					1		1
37	1		1		1		1
39		1		1	1		1

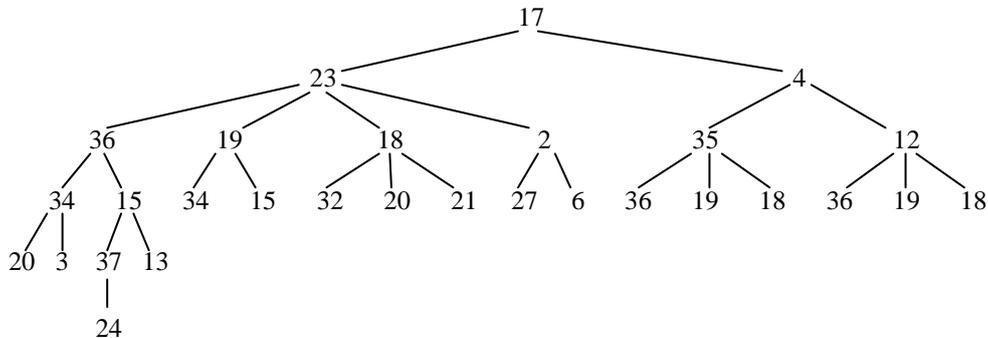


Рис. 2. Ветвь дерева поиска с начальной вершиной 17

Рассмотрим приведенный выше пример булевой функции, применив к нему метод нахождения наименьшего максимального независимого множества. Как было указано ранее, для применения лексикографического перебора необходимо переупорядочить интервалы и, соответственно, вершины графа. Интервалы выстроим в следующем порядке:

$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$	$x_1x_2x_3x_4x_5$
1 - 1 0 1 -	9 0 1 0 1 -	17 1 0 0 1 -	25 1 - 0 1 0	33 0 0 1 1 1
2 - 0 1 1 -	10 1 1 1 - 0	18 1 0 - 1 1	26 0 0 1 0 1	34 1 1 1 1 0
3 1 0 - 1 -	11 0 0 1 - 1	19 1 - 0 1 1	27 0 1 1 0 0	35 1 1 0 1 1
4 1 - 0 1 -	12 0 0 1 1 -	20 1 0 1 1 -	28 0 1 1 0 1	36 1 0 0 1 1
5 1 - - 1 0	13 - 1 0 1 1	21 1 - 1 1 0	29 1 1 1 0 0	37 1 0 1 1 1
6 0 - 1 0 1	14 - 0 1 1 1	22 1 1 0 1 -	30 0 1 0 1 1	38 1 0 0 1 0
7 0 1 1 0 -	15 0 1 1 0 -	23 1 1 - 1 0	31 0 0 1 1 0	39 1 0 1 1 0
8 - 1 1 0 0	16 - 1 0 1 0	24 1 0 - 1 0	32 0 1 0 1 0	40 1 1 0 1 0

Задание графа пересечения интервалов списком окрестностей вершин представлено ниже:

- | | |
|--|----------------------|
| 1: 4,5,9,13,16,19,22,23,25,30,32,35,40 | 7: 6,8,27,28 |
| 2: 3,5,11,12,14,15,18,20,21,24,31,33,37,39 | 8: 7,10,27,29 |
| 3: 2,4,5,14,15,17,18,19,20,21,24,25,36,37,38,39 | 9: 1,13,16,30,32 |
| 4: 1,3,5,13,16,17,18,19,22,23,24,25,35,36,38,40 | 10: 5,8,23,29,34 |
| 5: 1,2,3,4,10,15,16,17,20,21,22,23,24,25,34,38,39,40 | 11: 2,6,12,14,26,33 |
| 6: 7,11,26,28 | 12: 2,11,14,15,31,33 |

13:	1,4,9,19,22,30,35	27:	7,8
14:	2,3,11,12,18,20,33,37	28:	6,7
15:	2,3,5,12,20,21,24,31,39	29:	8,10
16:	1,4,5,9,22,23,25,32,40	30:	1,9,13
17:	3,4,5,18,19,24,25,36,38	31:	2,12,15
18:	2,3,4,14,17,19,20,36,37	32:	1,9,16
19:	1,3,4,13,17,18,22,35,36	33:	2,11,12,14
20:	2,3,5,14,15,18,21,24,37,39	34:	5,10,21,23
21:	2,3,5,10,15,20,23,24,34,39	35:	1,4,13,19,22
22:	1,4,5,13,16,19,23,25,35,40	36:	3,4,17,18,19
23:	1,4,5,10,16,21,22,25,34,40	37:	2,3,14,18,20
24:	2,3,4,5,15,17,20,21,25,38,39	38:	3,4,5,17,24,25
25:	1,3,4,5,16,17,22,23,24,38,40	39:	2,3,5,15,20,21,24
26:	6,11	40:	1,4,5,16,22,23,25

При таком упорядочении вершин уже первым в лексикографическом порядке максимальным независимым множеством является $\{1, 2, 6, 8, 17, 34\}$. Продолжение процесса не даст максимального независимого множества меньшей мощности. Соответствующая ортогональная ДНФ представлена матрицей

$$\begin{array}{ccccc|c} x_1 & x_2 & x_3 & x_4 & x_5 & \\ \hline - & 1 & 0 & 1 & - & 1 \\ - & 0 & 1 & 1 & - & 2 \\ 0 & - & 1 & 0 & 1 & 6 \\ - & 1 & 1 & 0 & 0 & 8 \\ 1 & 0 & 0 & 1 & - & 17 \\ 1 & 1 & 1 & 1 & 0 & 34 \end{array}$$

Распространение подхода на не полностью определенные функции. Задача минимизации не полностью определенной (частичной) булевой функции ставится как задача получения минимальной ДНФ, реализующей заданную функцию, т. е. принимающей значение, которое совпадает со значением этой функции везде, где оно определено. Обычно частичная булева функция задается двумя множествами: областью M^1 булева пространства аргументов, где функция имеет значение 1, и областью M^0 , где функция имеет значение 0. Область, где значение функции неопределено (остальная часть булева пространства), обозначается символом M^- . Классический метод минимизации частичной функции предполагает нахождение всех максимальных интервалов в множестве $M^1 \cup M^-$ и покрытие ими элементов множества M^1 . Для получения ортогональной ДНФ кроме максимальных надо рассматривать и все немаксимальные интервалы, а также учитывать их пересечение. Метод оказывается малоэффективным при слабой определенности заданной функции, поскольку многие интервалы являются бесполезными, находясь целиком в области M^- . В связи с этим выделен класс слабо определенных функций и для них разработан метод минимизации [3]. Данный класс определяется соотношением мощностей множеств $|M^1 \cup M^0| \ll |M^-|$. Метод использует понятие *интервально покрываемого множества*, которое представляет собой подмножество множества M^1 , такое, что существует интервал, содержащий все его элементы и не пересекающийся с множеством M^0 . Максимальное интервально покрываемое множество представляет собой интервально покрываемое множество, которое не является собственным подмножеством другого такого множества. Метод из работы [3] сводит задачу минимизации слабо определенной булевой функции к получению кратчайшего покрытия элементов множества M^1 максимальными интервально покрываемыми множествами.

Предлагаемый подход можно применять и для минимизации частичных, в том числе слабо определенных, функций в классе ортогональных ДНФ. Рассматриваются интервалы, связанные с интервально покрываемыми множествами, и граф пересечения этих интервалов. Получение

минимальной ортогональной ДНФ сводится к нахождению в данном графе наименьшего доминирующего независимого множества, или наименьшего максимального независимого множества. При этом если интервально покрываемое множество M_j^1 является собственным подмножеством интервально покрываемого множества M_i^1 , а соответствующие им интервалы совпадают, то M_j^1 исключается из рассмотрения. Далее рассматриваются только интервалы, связанные с интервально покрываемыми множествами. Пусть частичная булева функция задана следующими матрицами, которые обозначены символами соответствующих множеств:

$$M^1 = \begin{matrix} & x_1 & x_2 & x_3 & x_4 & x_5 & \\ \begin{matrix} 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{matrix} & \begin{matrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{matrix} & \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{matrix} & \end{matrix}, \quad M^0 = \begin{matrix} & x_1 & x_2 & x_3 & x_4 & x_5 & \\ \begin{matrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{matrix} & \begin{matrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix} & \end{matrix}.$$

Ниже представлены интервально покрываемые множества, полученные путем лексикографического перебора, и связанные с ними интервалы в виде троичных векторов. Некоторые множества в этом списке отсутствуют. Например, отсутствует множество {2, 9}, поскольку оно является подмножеством множества {2, 8, 9}, а обоим этим множествам соответствует один и тот же интервал, представляемый троичным вектором (- 1 1 - 0):

		$x_1x_2x_3x_4x_5$			$x_1x_2x_3x_4x_5$			$x_1x_2x_3x_4x_5$
1	{1}	0 0 1 0 1	8	{3,6}	1 0 - 1 0	15	{6,9}	1 - 1 1 0
2	{1,4}	0 0 1 - 1	9	{3,6,7,8,9}	1 - - - 0	16	{7}	1 1 0 1 0
3	{2}	0 1 1 0 0	10	{3,6,7,9}	1 - - 1 0	17	{7,8,9}	1 1 - - 0
4	{2,8}	- 1 1 0 0	11	{4}	0 0 1 1 1	18	{7,9}	1 1 - 1 0
5	{2,8,9}	- 1 1 - 0	12	{5}	1 0 0 1 1	19	{8}	1 1 1 0 0
6	{3}	1 0 0 1 0	13	{6}	1 0 1 1 0	20	{8,9}	1 1 1 - 0
7	{3,5}	1 0 0 1 -	14	{6,8,9}	1 - 1 - 0	21	{9}	1 1 1 1 0

Граф пересечений интервалов представлен списками окрестностей вершин:

1:	2	12:	7
2:	1,11	13:	8,9,10,14,15
3:	4,5	14:	4,5,8,9,10,13,15,17,18,19,20,21
4:	3,5,9,14,17,19,20	15:	5,8,9,10,13,14,17,18,20,21
5:	3,4,9,10,14,15,17,18,19,20,21	16:	9,10,17,18
6:	7,8,9,10	17:	4,5,9,10,14,15,16,18,19,20,21
7:	6,8,9,10,12	18:	5,9,10,14,15,16,17,20,21
8:	6,7,9,10,13,14,15	19:	4,5,9,14,17,20
9:	4,5,6,7,8,10,13,14,15,16,17,18,19,20,21	20:	4,5,9,10,14,15,17,18,19,21
10:	5,6,7,8,9,13,14,15,16,17,18,20,21	21:	5,9,10,14,15,17,18,20
11:	2		

Начальная ситуация в поиске решения представлена в табл. 4. Дальнейшие действия выполняются так же, как в случае полностью определенных функций. Дерево поиска изображено на рис. 3. Первое независимое доминирующее множество, полученное в процессе поиска, составляют вершины 2, 4, 7, 15 и 16. Затем получено наименьшее независимое доминирующее множество {2, 5, 8, 12}, которому соответствует следующая матрица:

$$\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \\ \left[\begin{array}{cccc} 0 & 0 & 1 & -1 \\ -1 & 1 & -0 & 0 \\ 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & 1 \end{array} \right] & \begin{array}{l} 2 \\ 5 \\ 8 \\ 12 \end{array} \end{array}$$

Таблица 4

	1	3	6	11	12	13	16	19	21
1	1								
2	1			1					
3		1							
4		1						1	
5		1						1	1
6			1						
7			1		1				
8			1			1			
9			1				1	1	1
10			1			1	1		1
11				1					
12					1				
13						1			
14						1		1	1
15						1			1
16							1		
17							1	1	1
18							1		1
19								1	
20								1	1
21									1

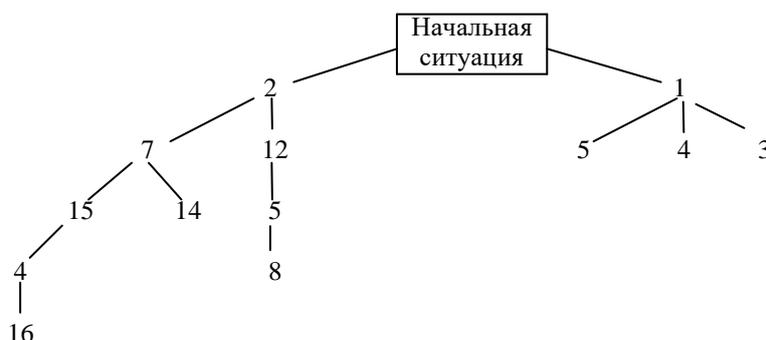


Рис. 3. Дерево поиска минимальной ортогональной ДНФ

Далее необходимо выполнить упрощение ДНФ, если это возможно, путем расширения интервалов. Естественно, что каждый интервал не должен пересекаться с множеством M^0 и ни с одним из интервалов, вошедших в решение. При таком расширении переменная x_4 оказалась несущественным аргументом:

$$\begin{array}{ccccc} x_1 & x_2 & x_3 & x_4 & x_5 \\ \left[\begin{array}{cccc} 0 & 0 & 1 & -1 \\ -1 & 1 & -0 & 0 \\ 1 & 0 & - & 0 \\ -0 & 0 & -1 & \end{array} \right] & \begin{array}{l} 2 \\ 5 \\ 8 \\ 12 \end{array} \end{array} = \begin{array}{ccccc} x_1 & x_2 & x_3 & x_5 \\ \left[\begin{array}{cccc} 0 & 0 & 1 & 1 \\ -1 & 1 & 0 & \\ 1 & 0 & -1 & \\ -0 & 0 & 0 & \end{array} \right] \end{array}.$$

На рассмотренном выше примере продемонстрируем действие метода, использующего нахождение наименьшего максимального независимого множества. Для лексикографического перебора упорядочим интервально покрываемые множества и соответствующие им интервалы следующим образом:

	$x_1x_2x_3x_4x_5$		$x_1x_2x_3x_4x_5$		$x_1x_2x_3x_4x_5$
1	{3,6,7,8,9}	1	---0	8	{3,6}
2	{2,8,9}	2	-11-0	9	{2,8}
3	{7,8,9}	3	11--0	10	{7,9}
4	{6,8,9}	4	1-1-0	11	{8,9}
5	{3,6,7,9}	5	1--10	12	{6,9}
6	{1,4}	6	001-1	13	{1}
7	{3,5}	7	1001-	14	{4}
				15	{5}
				16	{2}
				17	{3}
				18	{7}
				19	{6}
				20	{8}
				21	{9}

Нумерацию вершин графа пересечения интервалов приведем в соответствие с нумерацией интервалов:

1:	2,3,4,5,7,8,9,10,11,12,17,18,19,20,21	12:	1,2,3,4,5,8,10,11,19,21
2:	1,3,4,5,9,10,11,12,16,20,21	13:	6
3:	1,2,4,5,9,10,11,12,18,20,21	14:	6
4:	1,2,3,5,8,9,10,11,12,19,20,21	15:	7
5:	1,2,3,4,7,8,10,11,12,17,18,19,21	16:	2,9
6:	13,14	17:	1,5,7,8
7:	1,5,8,15,17	18:	1,3,5,10
8:	1,4,5,7,12,17,19	19:	1,4,5,8,12
9:	1,2,3,4,11,16,20	20:	1,2,3,4,9,11
10:	1,2,3,4,5,11,12,18,21	21:	1,2,3,4,5,10,11,12
11:	1,2,3,4,5,9,10,12,20,21		

Первым независимым множеством в процессе перебора оказалось {1, 6, 15, 16}. Дальнейший перебор не приводит к множеству с меньшей мощностью. Можно заметить, что поиск решения должен быть прекращен на этапе 7, так как $A_7 = \{7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21\}$, $B_7 = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 15, 16, 17, 18, 19, 20, 21\}$ и $A_7 \cup B_7 \neq V$. Ортогональная ДНФ, построенная в соответствии с полученным наименьшим независимым множеством, и результат расширения интервалов имеют следующий вид:

$$\begin{array}{l}
 \begin{array}{c} x_1 \ x_2 \ x_3 \ x_4 \ x_5 \\
 \left[\begin{array}{ccccc} 1 & - & - & - & 0 \\ 0 & 0 & 1 & - & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{array} \right] \begin{array}{l} 2 \\ 5 \\ 8 \\ 12 \end{array}
 \end{array}
 \quad
 \begin{array}{c} x_1 \ x_2 \ x_3 \ x_4 \ x_5 \\
 \left[\begin{array}{ccccc} 1 & - & - & - & 0 \\ 0 & 0 & 1 & - & 1 \\ - & 0 & 0 & - & 1 \\ 0 & - & - & 0 & 0 \end{array} \right]
 \end{array}
 \end{array}$$

Заключение. Рассмотренные методы ортогонализации ДНФ гарантируют получение минимального числа элементарных конъюнкций в ортогональных ДНФ. В работе показано, как для решения данной задачи можно применить аппарат теории графов. В отличие от многих методов ортогонализации ДНФ, предполагающих задание функции в произвольной ДНФ, предлагаемый подход рассчитан на представление функции в виде совершенной ДНФ. Данный подход распространяется также на не полностью определенные булевы функции. Задача минимизации ортогональной ДНФ имеет неполиномиальную сложность [1], поэтому практическое применение любого метода, получающего точное решение этой задачи, весьма ограничено. Полезность точного метода, в частности, заключается в том, что с его помощью можно оценивать качество эвристических методов, а также определять направление процесса решения задачи

при создании эвристического метода. В качестве эвристического метода ортогонализации ДНФ можно предложить метод, который использует упорядочение вершин графа пересечения интервалов и получение в этом графе первого в лексикографическом порядке максимального независимого множества.

Список использованных источников

1. Кузнецов, О. П. Ортогональные системы булевых функций и их применение к анализу и синтезу логических сетей / О. П. Кузнецов // Автоматика и телемеханика. – 1970. – № 10. – С. 117–128.
2. Баранов, С. И. Синтез микропрограммных автоматов / С. И. Баранов. – Л. : Энергия, 1979. – 232 с.
3. Закревский, А. Д. Логический синтез каскадных схем / А. Д. Закревский. – М. : Наука, 1981. – 416 с.
4. Синтез асинхронных автоматов на ЭВМ / А. Д. Закревский [и др.]. – Минск : Наука и техника, 1975. – 184 с.
5. Бибило, П. Н. Синтез комбинационных ПЛМ-структур для СБИС / П. Н. Бибило. – Минск : Навука і тэхніка, 1992. – 323 с.
6. Закревский, А. Д. Расчет надежности технической системы при заданных критических наборах событий / А. Д. Закревский, Ю. В. Поттосин // Логическое проектирование. – Минск : Ин-т техн. кибернетики АН Беларуси, 1996. – С. 123–131.
7. Закревский, А. Д. Логические основы проектирования дискретных устройств / А. Д. Закревский, Ю. В. Поттосин, Л. Д. Черемисинова. – М. : Физматлит, 2007. – 592 с.
8. Матросова, А. Ю. О вероятностном моделировании дискретных устройств / А. Ю. Матросова // Автоматика и телемеханика. – 1995. – № 3. – С. 156–164.
9. Гаврилов, М. А. Логическое проектирование дискретных автоматов (языки, методы, алгоритмы) / М. А. Гаврилов, В. В. Девятков, Е. И. Пупырев. – М. : Наука, 1977. – 352 с.
10. Бибило, П. Н. Синтез комбинационных схем методом функциональной декомпозиции / П. Н. Бибило, С. В. Енин. – Минск : Наука и техника, 1987. – 189 с.
11. Кардаш, С. Н. Ортогонализация системы ДНФ булевых функций / С. Н. Кардаш // Информационные технологии и системы 2020 (ИТС 2020) : материалы Междунар. науч. конф., Минск, Беларусь, 18 нояб. 2020 г. – Минск : БГУИР, 2020. – С. 41–42.
12. Поттосин, Ю. В. Методы дискретной математики в проектировании цифровых устройств / Ю. В. Поттосин. – Saarbrücken : LAP LAMBERT Academic Publishing, 2017. – 176 с.
13. 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. – С. 107–115.
14. Паршина, Н. А. Минимизация частичных булевых функций в классе ортогональных ДНФ / Н. А. Паршина // Новые информационные технологии в исследовании дискретных структур : докл. Второй Всерос. конф. – Екатеринбург : УрО РАН, 1998. – С. 181–184.
15. Поттосин, Ю. В. Ортогонализация системы не полностью определенных булевых функций / Ю. В. Поттосин // Вопросы синтеза логики ЦВМ : в 3 ч. – Каунас : КПИ, 1976. – Ч. III. – С. 39–42.
16. Лекции по теории графов / В. А. Емеличев [и др.]. – М. : Наука, 1990. – 384 с.

References

1. Kuznetsov O. P. *Orthogonal systems of Boolean functions and their applications in analysis and synthesis of logical networks*. Avtomatika i telemekhanika [Automatics and Remote Control], 1970, no. 10, pp. 117–128 (In Russ.).
2. Baranov S. I. *Sintez mikroprogrammnyh avtomanov. Synthesis of Microprogrammed Automata*. Leningrad, Energia, 1979, 232 p. (In Russ.).
3. Zakrevskij A. D. *Logicheskij sintez kaskadnyh shem. Logical Synthesis of Cascaded Circuits*. Moscow, Nauka, 1981, 416 p. (In Russ.).
4. Zakrevskij A. D., Balaklej L. I., Eliseeva N. A., Oranov A. M., Pottosin Yu. V., ..., Jankovskaja A. E. *Sintez asinhronnyh avtomatov na EVM. Synthesis of Asynchronous Automata in a Computer*. Minsk, Nauka i tehnika, 1975, 184 p. (In Russ.).
5. Bibilo P. N. *Sintez kombinatsionnyh PLM-struktur dl'a SBIS. Synthesis of Combinational PLA-structures for VLSI*. Minsk, Navuka i tehnika, 1992, 323 p. (In Russ.).

6. Zakrevskij A. D., Pottosin Yu. V. *Calculation of engineering system reliability when the critical set of events is given*. Logicheskoe proektirovanie [Logical Design]. Minsk, Institut tehnicheckoj kibernetiki Akademii nauk Belarusi, 1996, pp. 123–131 (In Russ.).
7. Zakrevskij A. D., Pottosin Yu. V., Cheremisinova L. D. Logicheskie osnovy proektirovanija diskretnyh ustrojstv. *Logical Fundamentals of Discrete Devices Design*. Moscow, Fizmatlit, 2007, 592 p. (In Russ.).
8. Matrosova A. Yu. *On probabilistic simulation of discrete devices*. Avtomatika i telemekhanika [Automatics and Remote Control], 1995, no. 3, pp. 156–164 (In Russ.).
9. Gavrilov M. A., Devjatkov V. V., Pupyrev E. I. Logicheskoe proektirovanie diskretnyh avtomatov (jazyki, metody, algoritmy). *Logical Design of Discrete Automata (Languages, Methods, Algorithms)*. Moscow, Nauka, 1977, 352 p. (In Russ.).
10. Bibilo P. N., Enin S. V. Sintez kombinatsionnyh shem metodom funkcional'noj dekompozicii. *Synthesis of Combinational Circuits by the Method of Functional Decomposition*. Minsk, Nauka i tehnika, 1987, 189 p. (In Russ.).
11. Kardash S. N. *Orthogonalization of a DNF system of Boolean functions*. Informacionnye tehnologii i sistemy 2020 (ITS 2020): materialy Mezhdunarodnoj nauchnoj konferencii, Minsk, Belarus', 18 nojabrja 2020 g. [Informational Technologies and Systems 2020 (ITS 2020): Proceedings of International Scientific Conference, Minsk, Belarus, 18 November 2020], Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2020, pp. 41–42 (In Russ.).
12. Pottosin Yu. V. Metody diskretnoj matematiki v proektirovanii cifrovyh ustrojstv. *Discrete Mathematics Methods in Digital Devices Design*. Saarbrücken, LAP LAMBERT Academic Publishing, 2017, 176 p. (In Russ.).
13. Pottosin Yu. V., Shestakov E. A. *Orthogonalization of a system of completely specified Boolean functions*. Logicheskoe proektirovanie [Logical Design]. Minsk, Institut tehnicheckoj kibernetiki Nacional'noj akademii nauk Belarusi, 2000, pp. 107–115 (In Russ.).
14. Parshina N. A. *Minimization of partial Boolean functions in the class of orthogonal DNFs*. Novye informacionnye tehnologii v issledovanii diskretnyh struktur: doklady Vtoroj Vserossijskoj konferencii [Novel Information Technologies in the Research of Discrete Structures: Proceedings of the Second All-Russian Conference]. Ekaterinburg, Ural'skoe otdelenie Rossijskoj akademii nauk, 1998, pp. 181–184 (In Russ.).
15. Pottosin Yu. V. *Orthogonalization of a system of incompletely specified Boolean functions*. Voprosy sinteza logiki CVM [The Problems of Digital Computer Logic Synthesis]. Kaunas, KPI, 1976, part III, pp. 39–42 (In Russ.).
16. Emelichev V. A., Mel'nikov O. I., Sarvanov V. I., Tyshkevich R. I. Lekcii po teorii grafov. *Lectures on Graph Theory*. Moscow, Nauka, 1990, 384 p. (In Russ.).

Информация об авторе

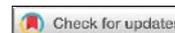
Поттосин Юрий Васильевич, кандидат физико-математических наук, ведущий научный сотрудник, Объединенный институт проблем информатики Национальной академии наук Беларуси, Минск, Беларусь.
E-mail: pott@newman.bas-net.by

Information about the author

Yuri V. Pottosin, Cand. Sci. (Phys.-Math.), Leading Researcher, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, Belarus.
E-mail: pott@newman.bas-net.by

ЗАЩИТА ИНФОРМАЦИИ И НАДЕЖНОСТЬ СИСТЕМ

INFORMATION PROTECTION AND SYSTEM RELIABILITY



УДК 004.3; 519.718.2; 519.873; 629.78
<https://doi.org/10.37661/1816-0301-2021-18-2-48-57>

Оригинальная статья
Original Paper

Исследование надежности аппаратной части офисного кластера

Т. С. Мартинович[✉], Н. Н. Парамонов, А. Г. Рымарчук, О. П. Чиж

Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
[✉]E-mail: martintan@newman.bas-net.by

Аннотация. Исследование показателей надежности аппаратной части офисного суперкомпьютера проводилось на примере кластера СКИФ-ГЕО-Офис РБ, разработанного в рамках научно-технической программы Союзного государства «СКИФ-НЕДРА» (2015–2018 гг.). Компоненты кластера расположены в малогабаритной стойке на базе корпуса типа Full Tower «Aerocool Expredator Black».

Приводятся реализованные в кластере базовые архитектурные принципы, его состав и структурно-функциональная схема. Предлагается методическое обеспечение расчета надежности кластера с учетом предыдущих исследований авторов, обосновывается его структурная схема надежности. Даются формулы расчета основных показателей надежности ядра и совокупности вычислительных средств (СВС) кластера, анализируются последствия отказов его комплектующих.

Предлагается математическая модель надежности (граф состояний) СВС кластера, позволяющая получать формулы для вычисления средней наработки на отказ и на сбой СВС. Оценивается надежность кластера в целом по справочным данным о надежности его комплектующих и по данным эксплуатации суперкомпьютеров семейства СКИФ, вычисляются показатели надежности кластера.

Ключевые слова: офисный кластер, показатели надежности, расчет надежности, структурная схема надежности, семейство СКИФ, кластер СКИФ-ГЕО-Офис РБ

Для цитирования. Исследование надежности аппаратной части офисного кластера / Т. С. Мартинович [и др.] // Информатика. – 2021. – Т. 18, № 2. – С. 48–57. <https://doi.org/10.37661/1816-0301-2021-18-2-48-57>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 29.07.2020
Подписана в печать | Accepted 21.12.2020
Опубликована | Published 26.03.2021

The study of the reliability of the hardware part of the office cluster

T. S. Martinovich[✉], N. N. Paramonov, A. G. Rymarchuk, O. P. Tchij

*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus
✉E-mail: martintan@newman.bas-net.by*

Abstract. The study of measures of reliability of the hardware part of the office cluster was carried out on the example of the cluster SKIF-GEO-Office RB (further as “cluster”) developed within the framework of scientific and technical program "SKIF-NEDRA" (2015–2018, Program of the Union State of Russia and Belarus). The cluster components are located in a small rack on the basis of full Tower "Aerocool Expredator Black" type case.

The basic architectural principles implemented in the cluster, the composition, structural and functional scheme of the cluster are given. The methodological support for calculating the reliability of the cluster, based on previous studies of the authors, and its structural scheme of reliability is justified. The choice of the main measures of reliability of the cluster core and the set of computing facilities is justified and formulas of calculation of these measures are given. The analysis of the consequences of failures of component parts of the cluster is carried out.

A mathematical model of reliability (state graph) of the set of computing facilities of cluster is proposed, which allows to derive formulas for calculating the average value of the time-to-failure and time-to-interruption of cluster. The estimation of the reliability of the cluster as a whole, based on the calculation of measures of reliability on the reference data on the reliability of components as well as on the operation of supercomputers of the family SKIF. The measures of reliability of the cluster are calculated.

Keywords: office cluster, measure of reliability, reliability calculation, structural scheme of reliability, family SKIF, cluster SKIF-GEO-Office RB

For citation. Martinovich T. S., Paramonov N. N., Rymarchuk A. G., Tchij O. P. The study of the reliability of the hardware part of the office cluster. *Informatics*, 2021, vol. 18, no. 2, pp. 48–57 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-48-57>

Conflict of interest. The author declare of no conflict of interest.

Введение. В рамках выполнения научно-технической программы Союзного государства «СКИФ-Недра» (2015–2018) [1] проводились исследования по разработке различных кластерных суперкомпьютерных конфигураций, в том числе кластера СКИФ-ГЕО-Офис РБ, предназначенного для установки в отраслевых структурных подразделениях и обладающего небольшими габаритами, весом, энергопотреблением и уровнем акустических шумов. Разработанный в результате кластер может применяться как для самостоятельного выполнения офисных научных расчетов, автоматизированного проектирования, обработки массивов данных, так и для разработки программ, подготовки данных и проведения предварительных расчетов в качестве дополнения к мощным высокопроизводительным кластерным системам.

При создании кластера были использованы следующие базовые принципы [2]:

- классическая кластерная архитектура с распределенной памятью;
- работа под управлением операционной системы (ОС) типа Linux;
- организация единой сети кластера Gigabit Ethernet (GbE), реализующей функции системной и вспомогательной сетей для MPI-обменов, файловых операций ввода-вывода и оптимизации параллельного выполнения программ;
- реализация гибридных вычислительных узлов (ВУ) на базе чипов, в которых интегрированы ядра, и традиционных (классических) процессоров архитектуры x86-64 и GPU – гибридных процессоров APU (Accelerated Processing Unit);
- реализация ВУ кластера на базе компактных компьютерных комплектов типа NUC;

– применение в структуре кластера одного выделенного управляющего узла, выполняющего функции управления вычислениями и подключения пользователей из локальной сети структурного подразделения (офиса), а также функции файл-сервера;

– применение в системе охлаждения кластера отечественных тепловых трубок и радиаторов для отвода тепла от процессоров.

В настоящей статье рассматривается задача исследования и расчета показателей надежности аппаратной части офисного суперкомпьютера на примере кластера СКИФ-ГЕО-Офис РБ.

Компоненты кластера СКИФ-ГЕО-Офис РБ. Составляющие кластера расположены в малогабаритной стойке на базе корпуса типа Full Tower «Aerocool Expredator Black», габариты изделия 620×232×605 мм, масса 36 кг. Интегральная пиковая производительность кластера при выполнении операций с плавающей точкой с двойной точностью составляет 18 Тфлопс [3].

Работа кластера организована под управлением ОС Linux на основе последних версий дистрибутива Fedora с обновлениями для повышения производительности. Средством разработки параллельных программ является связка из стандартных компиляторов дистрибутива Linux (C/C++, Fortran) и библиотеки MPI. В качестве программных средств пакетной обработки заданий используется свободно распространяемая версия системы пакетной обработки PBS Torque. Планировщик задач – свободно распространяемый программный пакет Maui.

В состав кластера СКИФ-ГЕО-Офис РБ входят:

– девять ВУ Intel NUC7i7BNH, Intel Core i7-7567U Processor (3.5/4.0 GHz, Dual Core/HT с расширением набора команд SSE 4.1/4.2, AVX 2.0, 4 MB Cache, 28 Вт), iGPU Intel Iris Graphics 540 (300–950 MHz), 32 GB (2×16) DDR4, 512 GB Samsung SSD, интерфейсы Gigabit Ethernet, M.2, USB 3.0, SATA;

– управляющий узел Intel NUC7i7BNH, Intel Core i7-7567U Processor (3.5/4.0 GHz, Dual Core/HT с расширением набора команд SSE 4.1/4.2, AVX 2.0, 4 MB Cache, 28 Вт), iGPU Intel Iris Graphics 540 (300–950 MHz), 32 GB (2×16) DDR4, 1.2 TB Intel SSD, интерфейсы Gigabit Ethernet, M.2, USB 3.0, SATA;

– коммутатор Gigabit Ethernet Zyxel GS-1100-16;

– корпус Full Tower «Aerocool Expredator Black»;

– 10 блоков питания ВУ и управляющего узла по 65 Вт каждый;

– блок питания вентиляторов и индикации PD-45A;

– 12 патч-кордов 5Е;

– 10 радиаторов с запрессованными тепловыми трубками.

Структурно-функциональная схема кластера показана на рис. 1, где УУ – управляющий узел, БП – блок питания вычислительного (управляющего) узла, АД – адаптер сетевой.

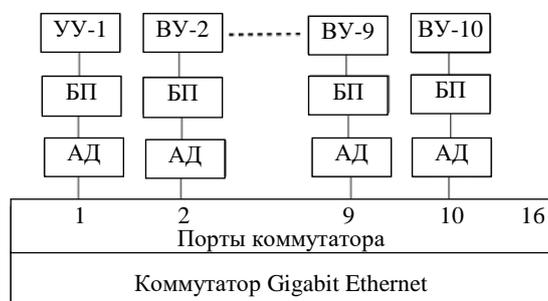


Рис. 1. Схема кластера СКИФ-ГЕО-Офис РБ [1]

Методическое обеспечение расчетов надежности кластера. Расчет показателей надежности кластера СКИФ-ГЕО-Офис РБ выполнен по методике оценки надежности кластерных вычислительных систем, предложенной канд. техн. наук, доцентом Л. И. Кульбаком [4]. Согласно

этой методике схема надежности кластера состоит из двух структурных частей – ядра кластера и СВС (узлов). Отказы составных частей СВС приводят к снижению производительности кластера, а отказы частей ядра – к снижению до нуля.

Согласно [4] в качестве основных и дополнительных показателей надежности выберем показатели, приведенные в табл. 1.

Таблица 1

Показатели надежности кластера СКИФ-ГЕО-Офис РБ и его составных частей

Показатели надежности	Кластер в целом	Ядро кластера	СВС
Основной	Коэффициент сохранения эффективности, $K_{\text{эф.кластера}}$	Коэффициент готовности, $K_{\text{г.я}}$	Коэффициент сохранения эффективности (потенциальной производительности), $K_{\text{эф.СВС}}$
Дополнительный 1	Средняя наработка на отказ, $T_{\text{о.кластера}}$	Средняя наработка на отказ, $T_{\text{о.я}}$	Средняя наработка на отказ, $T_{\text{о.СВС}}$
Дополнительный 2	Средняя наработка на сбой, $T_{\text{сб.кластера}}$	Среднее время восстановления, $T_{\text{в.я}}$	Средняя наработка на сбой, $T_{\text{сб.СВС}}$

Отличительной характеристикой кластера является параллельное участие в вычислительном процессе большого количества ВУ. Исключение из вычислительного процесса одного или нескольких узлов не препятствует продолжению использования кластера по назначению, а лишь снижает его производительность.

Под потенциальной (пиковой) производительностью кластера будем понимать производительность, которая достигается при работе всех узлов кластера. По мере уменьшения количества работоспособных ВУ коэффициент сохранения эффективности кластера падает.

Пороговое (допустимое) уменьшение количества работоспособных ВУ в основном зависит от специфики конкретной прикладной задачи, выполняемой пользователем. В частности, если для решения прикладной задачи требуется определенное количество ВУ, то выход из строя нескольких ВУ кластера, не участвующих в вычислительном процессе, не оказывает влияния на эффективность ее выполнения. Если произошел отказ ВУ, задействованного в вычислительном процессе, то, как правило, прекращается расчет задачи. В результате определенных действий оператора происходит реконфигурация системы и выполнение задачи пользователя начинается заново. Такое событие в соответствии с ГОСТ 27.002–2015 «Надежность в технике. Термины и определения» называется сбоем. Критерием сбоя кластера является такое событие, когда получен неверный результат в задаче пользователя (теста) или произошло незапланированное прерывание вычислительного процесса и для устранения последствий достаточно провести повторные действия оператора, не требующие замены технических средств или настройки регулировок. При отказе кластера для восстановления его работоспособности необходимо осуществить ремонт или замену отказавшего оборудования.

С учетом вышеизложенного получим равенство

$$K_{\text{эф.кластера}} = \begin{cases} K_{\text{эф.СВС}} & \text{при работоспособном ядре кластера,} \\ 0 & \text{при неработоспособном ядре кластера.} \end{cases} \quad (1)$$

Вероятность пребывания ядра кластера в работоспособном состоянии в произвольный момент времени представляет собой коэффициент готовности ядра $K_{\text{г.я}}$. Следовательно, выражение (1) можно записать в виде

$$K_{\text{эф.кластера}} = K_{\text{г.я}} \cdot K_{\text{эф.СВС}} \quad (2)$$

Средняя наработка на сбой кластера будет равна средней наработке на сбой СВС:

$$T_{\text{ср.сб.кластера}} = T_{\text{ср.сб.СВС}} \quad (3)$$

Оценку показателей надежности кластера следует начинать с оценки влияния отказов его составных частей на коэффициент сохранения эффективности $K_{\text{эф.кластера}}$.

В табл. 2 приведены составные части кластера, влияние их отказов на потенциальную производительность и способы восстановления работоспособности кластера.

Таблица 2

Способы восстановления работоспособности кластера при отказах его составных частей

Составные части кластера	Ущерб от отказа	Способ восстановления
ВУ	Сбой кластера	Реконфигурация
Управляющий узел	Отказ кластера	Замена на другой управляющий узел
Коммутатор сети (общая часть всех портов коммутатора)		Замена на другой коммутатор сети
Узел связи с ВУ (порт коммутатора, адаптер сети с кабелем)	Сбой кластера	Реконфигурация
Вентилятор	Повреждение	Текущее восстановление
Блок питания вентиляторов и индикации		
Радиаторы с запрессованными тепловыми трубками		
Корпус Full Tower «Aerocool Expredator Black»		
Патч-корд 5Е		

В результате анализа влияния отказов составных частей кластера на его показатели надежности определена структурная схема надежности (ССН) ядра кластера (рис. 2).



Рис. 2. ССН ядра кластера

Таким образом, верно равенство

$$\Lambda_{\text{я}} = \Lambda_{\text{уу}} + \Lambda_{\text{к.оч}},$$

где $\Lambda_{\text{я}}$, $\Lambda_{\text{уу}}$, $\Lambda_{\text{к.оч}}$ – интенсивности отказов ядра, управляющего узла и общей части (без учета портов) коммутатора кластера соответственно.

Среднюю наработку на отказ ($T_{\text{о.я}}$) и коэффициент готовности ($K_{\text{г.я}}$) ядра кластера представим отношениями

$$T_{\text{о.я}} = 1 / \Lambda_{\text{я}}, \quad K_{\text{г.я}} = T_{\text{о.я}} / (T_{\text{о.я}} + T_{\text{в.я}}),$$

где $T_{\text{в.я}}$ – среднее время восстановления ядра кластера.

Управляющий узел кластера, по сути, представляет собой ВУ кластера, предназначенный для выполнения дополнительных функций. С учетом этого рассмотрим ССН ВУ на рис. 3 и применим результаты расчетов его надежности к управляющему узлу.



Рис. 3. ССН ВУ кластера

Согласно ССН ВУ интенсивность отказов вычислительного (управляющего) узла кластера ($\Lambda_{\text{ВУ}}$) определена выражением

$$\Lambda_{\text{ВУ}} = \Lambda_{\text{пр}} + \Lambda_{\text{б.п}} + \Lambda_{\text{с.а}} + \Lambda_{\text{п.к}},$$

где $\Lambda_{\text{пр}}$, $\Lambda_{\text{б.п}}$, $\Lambda_{\text{с.а}}$ и $\Lambda_{\text{п.к}}$ – интенсивности отказов процессора, блока питания ВУ кластера, сетевого адаптера и порта коммутатора с кабелем соответственно.

Среднюю наработку на отказ ВУ кластера получаем по формуле

$$T_{\text{о.ВУ}} = 1 / \Lambda_{\text{ВУ}}.$$

Определяем среднее время восстановления ядра кластера:

$$T_{\text{в.я}} = T_{\text{о.я}} (\Lambda_{\text{уу}} T_{\text{в.уу}} + \Lambda_{\text{к.оч}} T_{\text{в.к.оч}}),$$

где $T_{\text{в.уу}}$ и $T_{\text{в.к.оч}}$ – среднее время восстановления управляющего узла кластера и общей части коммутатора соответственно.

Математическая модель надежности СВС кластера. Для оценки показателей надежности СВС предлагается использовать ее модель состояний в процессе эксплуатации. Модель состояний СВС графически выглядит в виде ориентированного размеченного графа, узлами которого являются состояния объекта в процессе его эксплуатации, ориентированные ребра изображают пути перехода из одного состояния в другое, а разметка ребер показывает интенсивности переходов. Допустим, что процесс изменения состояний СВС при ее эксплуатации является дискретным марковским процессом с конечным числом состояний и непрерывным временем, а потоки перехода СВС в другие состояния являются простейшими, т. е. имеют показательное распределение.

Обозначим множество состояний СВС W и разделим его на два непересекающихся подмножества: $W_{\text{РС}}$ – состояния СВС с принятой в качестве допустимой производительностью СВС, $W_{\text{НРС}}$ – состояния СВС с принятой в качестве недопустимой производительностью СВС. Выделим из множества $W_{\text{РС}}$ подмножество состояний, из которых возможен непосредственный переход в множество $W_{\text{НРС}}$, и обозначим его $W_{\text{РС-НРС}}$.

Примем стратегию восстановления СВС с обеспечением наибольшего коэффициента сохранения эффективности, реализуемую введением экстренного восстановления ВУ. Обычное восстановление ВУ начинается сразу после его отказа и может осуществляться продолжительное время (сутки, недели, месяцы) в процессе использования кластера по назначению с уменьшенной производительностью. Восстановление работоспособности узлов производится в порядке очередности их отказов. После восстановления ВУ немедленно вводится в конфигурацию кластера и становится доступным для ПО кластера. В случае накопления в составе СВС количества неработоспособных узлов более установленного числа n производится их экстренное восстановление.

Отметим, что состояние СВС меняется при каждом отказе и восстановлении ВУ. Формально граф состояний СВС должен иметь $(N_0 + 1)$ состояние, где N_0 – исходное число узлов в составе СВС. На практике время восстановления ВУ выбирается таким, чтобы в СВС накапливалось незначительное число (единицы) неработоспособных узлов. Примем предельное число неработоспособных ВУ в составе СВС, после превышения которого вводится экстренное восстановление, $n = 2$. Вероятность накопления в системе более двух неработоспособных узлов по данным расчетов надежности кластерных конфигураций семейства СКИФ очень мала (примерно 0,005).

Среднее время восстановления отказавшего узла в обычном случае принято $T_{\text{в.ВУ.о}} = 168$ ч (одна неделя), а в экстренном случае – $T_{\text{в.ВУ.э}} = 8$ ч. Среднее время восстановления ядра кластера $T_{\text{в.я}} = 1,0$ ч.

Граф состояний СВС для принятой стратегии восстановления ВУ изображен на рис. 4, где S_i – состояние СВС, когда в нем находятся i накопившихся неработоспособных ВУ ($i = 0, 1, 2, 3$); S_0 – исходное состояние СВС, когда все ВУ в СВС работоспособны и доступны для ПО кластера; Λ_i , $i = 0, 1, 2$, – интенсивность перехода СВС из состояния S_i в состояние S_{i+1} , включая

состояние S_3 , в котором вводится режим восстановления в экстренном случае; V_0 – интенсивность восстановления ВУ в обычном случае; V_3 – интенсивность восстановления ВУ в экстренном случае.

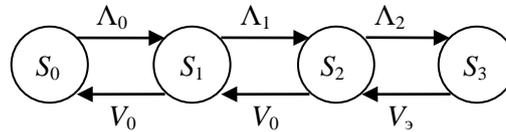


Рис. 4. Граф состояний СВС

По размеченному графу состояний (рис. 4) можно составить систему уравнений [5] с учетом замены одного уравнения на уравнение нормировки:

$$\Lambda_0 P_0 = V_0 P_1, (V_0 + \Lambda_1) P_1 = \Lambda_0 P_0 + V_0 P_2, (V_0 + \Lambda_2) P_2 = \Lambda_1 P_1 + V_3 P_3, \sum_0^3 P_i = 1, \quad (4)$$

где P_i – вероятность пребывания СВС в состоянии S_i .

Рассмотрим значения переменной Λ_i , используемой в графе состояний СВС. Для кластера справедливо соотношение

$$\Lambda_i = (N_0 - i) \Lambda_{\text{ВУ}}, \quad i = 0, 1, 2, \quad (5)$$

где $\Lambda_{\text{ВУ}}$ – интенсивность отказов ВУ, N_0 – количество ВУ в исходном состоянии СВС, i – количество накопившихся неработоспособных ВУ в СВС.

Корнями системы уравнений (4) с учетом (5) являются равенства

$$P_1 = B_1 P_0, \quad B_1 = N_0 \Lambda_{\text{ВУ}} / V_0; \quad (6)$$

$$P_2 = B_2 P_0, \quad B_2 = \{N_0 \Lambda_{\text{ВУ}} [V_0 + (N_0 - 1)] \Lambda_{\text{ВУ}}\} / V_0^2; \quad (7)$$

$$P_3 = B_3 P_0, \quad B_3 = \{N_0 \Lambda_{\text{ВУ}} [V_0 + (N_0 - 1) \Lambda_{\text{ВУ}}] [V_0 + (N_0 - 2) \Lambda_{\text{ВУ}} - V_0 N] [(N_0 - 1) \Lambda_{\text{ВУ}}^2]\} / V_0^2 V_3; \quad (8)$$

$$P_0 = [1 + B_1 + B_2 + B_3]^{-1}. \quad (9)$$

Средний коэффициент сохранения эффективности для режима восстановления в обычном случае рассчитывается согласно выражению

$$K_{\text{ср.эф}} = \sum_{i=0}^2 K_{\text{эф.}i} P_i, \quad (10)$$

где $K_{\text{эф.}i}$ – коэффициент сохранения эффективности в i -м состоянии СВС,

$$K_{\text{эф.}i} = (N_0 - i) / N_0 \quad (11)$$

при $i = 0, 1, 2$.

Вычисляем наработку на сбой СВС для режима восстановления в обычном случае:

$$T_{\text{сб.СВС}} = \frac{1}{(P_0 + P_1 + P_2) \Lambda_{\text{ВУ}}} \sum_0^2 \frac{1}{N_0 - i}. \quad (12)$$

Расчет надежности кластера. В табл. 3 приведены показатели безотказности компонентов ССН согласно статистическим данным эксплуатации кластерных конфигураций семейства СКИФ в Республиканском суперкомпьютерном центре коллективного пользования (РСКЦ КП) ОИПИ НАН Беларуси.

Таблица 3

Показатели безотказности составных частей ССН кластера [1]

Составная часть ССН	Интенсивность отказов, 10^{-6} 1/ч	Средняя наработка на отказ, ч
Чип ВУ	31,125	32 128
Блок питания ВУ	0,500	2 000 000
Сетевой адаптер	0,213	46 948 356
Порт коммутатора с кабелем	0,166	60 240 963
Коммутатор	6,943	144 030

В соответствии с ССН на рис. 3 и данными табл. 3 производим расчет надежности ВУ кластера. Получаем интенсивность отказов $\Lambda_{ВУ} = 32,004 \cdot 10^{-6}$ 1/ч, среднюю наработку на отказ $T_{о.ВУ} = 31\,246$ ч и коэффициент готовности $K_{г.ВУ} = 0,999\,968$ при среднем времени восстановления $T_{в.ВУ} = 8,0$ ч.

Согласно ССН на рис. 2 и данным табл. 3 при расчете ядра кластера находим интенсивность отказов $\Lambda_{я} = 38,947 \cdot 10^{-6}$ 1/ч и среднюю наработку на отказ $T_{о.я} = 25\,676$ ч. Коэффициент готовности $K_{г.я} = 0,999\,961$ при среднем времени восстановления $T_{в.я} = 1,0$ ч.

На основании статистических данных эксплуатации кластерных конфигураций семейства СКИФ в РСКЦ КП ОИПИ НАН Беларуси установлены следующие значения переменных для эксплуатации кластера: исходное количество ВУ в СВС $N_0 = 9$; допустимое число накопленных отказов ВУ $n = 2$; среднее время восстановления отказавшего ВУ в обычном случае $T_{в.ВУ.о} = 168$ ч (одна неделя); интенсивность восстановления ВУ в обычном случае $V_0 = 1/T_{в.ВУ.о} = 5,95 \cdot 10^{-3}$ 1/ч; среднее время восстановления отказавшего ВУ в экстренном случае $T_{в.ВУ.э} = 8$ ч; интенсивность восстановления ВУ в экстренном случае $V_э = 1/T_{в.ВУ.э} = 0,125$ 1/ч.

С учетом принятых данных и выражений (6)–(12) получим равенства

$$B_1 = 0,047\,08; B_2 = 0,0049; B_3 < 10^{-4};$$

$$P_1 = 0,044\,752; P_2 = 0,004\,658; P_0 = 0,950\,554;$$

$$K_{ср.эф.СВС} = 0,993\,957; T_{сб.СВС} = 12\,112 \text{ ч} = 1,38 \text{ года.}$$

В соответствии с формулами (2), (3) находим значения показателей надежности кластера в целом:

$$K_{ср.эф.кластера} = K_{г.я} \cdot K_{ср.эф.СВС} = 0,999\,961 \cdot 0,993\,957 = 0,993\,918;$$

$$T_{о.кластера} = T_{о.я} = 25\,676 \text{ ч} = 2,93 \text{ года}; T_{сб.кластера} = T_{сб.СВС} = 12\,112 \text{ ч} = 1,38 \text{ года.}$$

Заключение. Согласно проведенной оценке надежности кластера в целом, основанной на вычислении показателей надежности по справочным данным о надежности компонентов и комплектующих элементов, сведениям об эксплуатации суперкомпьютеров семейства СКИФ и полученным расчетным соотношениям были определены показатели надежности кластера СКИФ-ГЕО-Офис РБ. Вычисления выполнялись с учетом значений показателей надежности составных частей ССН кластера и принятых исходных данных: номинального количества ВУ

в СВС $N_0 = 9$, предельного числа неработоспособных ВУ в составе СВС до введения экстренного восстановления $n = 2$, среднего времени восстановления ВУ в обычном случае 168 ч (неделя) и экстренном случае 8 ч, среднего времени восстановления ядра кластера 1 ч. В результате получены: средний коэффициент сохранения эффективности $K_{\text{ср.эф.кластера}} = 0,993\ 918$, средняя наработка на отказ кластера $T_{\text{о.кластера}} = 2,93$ года, наработка на сбой кластера $T_{\text{сб.кластера}} = 1,38$ года.

Приведенный расчет показателей надежности разработанного офисного кластера СКИФ-ГЕО-Офис РБ позволяет определить наиболее эффективные стратегии его эксплуатации, а также подходы к проектированию и применению архитектурных и структурных решений кластерных суперкомпьютерных конфигураций.

К основным направлениям обеспечения необходимых показателей надежности можно отнести:

- корректный расчет состава и обеспечение комплектности восстанавливаемого набора запасных частей;
- минимизацию количества составных частей ядра кластера, например, за счет использования в структуре кластера двух выделенных управляющих узлов (основного и резервного) для управления вычислениями и подключения пользователей из внешней локальной сети, организации системной сети для MPI-обменов и командной (вспомогательной) сети для оптимизации параллельного выполнения программ.

Предложенная методика расчета надежности аппаратной части офисного кластера была использована при оценке показателей надежности созданного в 2020 г. в ОИПИ НАН Беларуси опытного образца офисного кластера СКИФ-ГРИД-СНГ-Офис (12 ВУ, включая управляющий узел; интегральная пиковая производительность при выполнении операций с плавающей точкой с двойной точностью равна 22 Тфлопс) [6].

Вклад авторов. Мартинович Т. С. – проведение расчета надежности офисного кластера по статистическим данным эксплуатации кластерных конфигураций семейства СКИФ в РСКЦ КП ОИПИ НАН Беларуси; Парамонов Н. Н. – научное руководство исследованием, методическое обеспечение расчетов надежности кластера, проверка полученных результатов; Рымарчук А. Г. – изучение и решение вопросов, связанных с достоверностью данных о составе технических средств и данных по эксплуатации кластерных конфигураций; Чиж О. П. – разработка концепции исследования, критический пересмотр содержания и утверждение окончательного варианта статьи для публикации.

Список использованных источников

1. Белорусские суперкомпьютеры семейства СКИФ / Н. Н. Парамонов [и др.] ; Объединенный институт проблем информатики Национальной академии наук Беларуси. – Гомель : Вечерний Гомель-Медиа, 2020. – 268 с.
2. Белорусские кластеры семейства «СКИФ-ГЕО» / М. Е. Кулешова [и др.] // Седьмой Нац. суперкомпьютерный форум : сб. докл., Переславль-Залесский, 27–30 нояб. 2018 г. / Институт программных систем РАН. – Режим доступа: http://2018.nskf.ru/TesisAll/00_Plenar/051_RymarchukAG.pdf/. – Дата доступа: 20.06.2020.
3. Малогабаритный офисный кластер семейства СКИФ-ГЕО-Офис РБ [Электронный ресурс] / М. Е. Кулешова [и др.] // Шестой Нац. суперкомпьютерный форум : сб. докл., Переславль-Залесский, 28 нояб. – 01 дек. 2017 г. / Институт программных систем РАН. – Режим доступа: <http://2017.nscf.ru/nauchno-prakticheskaya-konferenciya/tezisy-dokladov/>. – Дата доступа: 20.06.2020.
4. Анищенко, В. В. Модели надежности кластерных вычислительных систем / В. В. Анищенко, Л. И. Кульбак, Т. С. Мартинович // Вес. Нац. акад. наук Беларуси. Сер. физ.-техн. наук. – 2008. – № 1. – С. 89–99.
5. Викторова, В. С. Модели и методы расчета надежности технических систем / В. С. Викторова, А. С. Степеняц. – Изд. 2-е, испр. – М. : Ленанд, 2016. – 256 с.
6. Компактный вычислительный кластер : пат. РБ на полезную модель № 12417, МПК 606F / А. Г. Рымарчук, А. Н. Евдокимчиков, В. В. Мазюк, С. В. Кругликов, Н. Н. Парамонов, О. П. Чиж, В. Е. Маршалович, В. В. Мурашко, Е. И. Печковский. – Оубл. 30.10.2020.

References

1. Paramonov N. N., Tchij O. P., Rymarchuk A. G., Ablamejko S. V., Anishchenko V. V., Kruglikov S. V., Tuzikov A. V. Belorusskie superkomp'yutery semejstva SKIF. *Belarusian Supercomputers of the SKIF Family*, Gomel, Vechernij Gomel'-Media, 2020, 268 p. (In Russ.).
2. Kuleshova M. E., Paramonov N. N., Rymarchuk A. G., Tchij O. P. *Belarusian clusters of the SKIF-GEO family*. Sed'moj Nacional'nyj superkomp'juternyj forum: sbornik dokladov, Pereslavl'-Zalesskij, 27–30 nojabrja 2018 g. Institut programmyh sistem Rossijskoj akademii nauk [7th National Supercomputer Forum: Collection of Reports, Pereslavl'-Zalessky, 27 November – 30 November 2018. Program Systems Institute of the Russian Academy of Sciences] (In Russ.). Available at: http://2018.nskf.ru/TesisAll/00_Plenar/051_RymarchukAG.pdf/ (accessed 20.06.2020).
3. Kuleshova M. E., Murashko N. N., Paramonov N. N., Rymarchuk A. G., Tchij O. P. *Small office cluster of the Belarusian SKIF family-GEO-Office*. Shestoj Nacional'nyj superkomp'juternyj forum: sbornik dokladov, Pereslavl'-Zalesskij, 28 nojabrja – 01 dekabrja 2017 g. Institut programmyh sistem Rossijskoj akademii nauk [6th National Supercomputer Forum: Collection of Reports, Pereslavl'-Zalessky, 28 November – 01 December 2017. Program Systems Institute of the Russian Academy of Sciences] (In Russ.). Available at: <http://2017.nskf.ru/nauchno-prakticheskaya-konferenciya/tezisy-dokladov/> (accessed 20.06.2020).
4. Anishchenko V. V., Kulbak L. I., Martinovich T. S. *Reliability models of cluster computing systems*. Vestsi Natsyianal'nai akademii navuk Belarusi. Seryia fizika-technichnykh navuk [Proceedings of the National Academy of Sciences of Belarus. Physical-technical series], 2008, no. 1, pp. 89–99 (In Russ.).
5. Viktorova V. S., Stepenyanc A. S. *Modeli i metody rascheta nadezhnosti tekhnicheskikh sistem. Models and Methods for Calculating the Reliability of Technical Systems*. Moscow, Lenand, 2016, 256 p. (In Russ.).
6. Rymarchuk A. G., Evdokimchikov A. N., Mazjuk V. V., Kruglikov S. V., Paramonov N. N., ..., Pechkovskij E. I. *Kompaktnyj vychislitel'nyj klaster: patent Respubliki Belarus' na poleznuju model' № 12417, MPK 606F. Compact Computing Cluster: patent of the Republic of Belarus for Utility Model no. 12417, MPK 606F*. Publ. date 30.10.2020 (In Russ.).

Информация об авторах

Мартинович Татьяна Сергеевна, научный сотрудник, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: martintan@newman.bas-net.by

Парамонов Николай Николаевич, кандидат технических наук, доцент, ведущий научный сотрудник, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: nick@newman.bas-net.by

Рымарчук Александр Григорьевич, главный конструктор проекта, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: rymarchuk@newman.bas-net.by

Чиж Олег Петрович, кандидат физико-математических наук, заведующий лабораторией высокопроизводительных систем, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: otchij@newman.bas-net.by

Information about the authors

Tatyana S. Martinovich, Researcher, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: martintan@newman.bas-net.by

Nikolaj N. Paramonov, Cand. Sci. (Eng.), Associate Professor, Leading Researcher, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: nick@newman.bas-net.by

Aleksandr G. Rymarchuk, Chief Designer of the Project, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: rymarchuk@newman.bas-net.by

Oleg P. Tchij, Cand. Sci. (Phys.-Math.), Head of the Laboratory of High-Performance Systems, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: otchij@newman.bas-net.by

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

INFORMATION TECHNOLOGIES



УДК 004.89, 004.421, 519.688
<https://doi.org/10.37661/1816-0301-2021-18-2-58-71>

Оригинальная статья
Original Paper

Интеллектуальный анализ качества учебного контента по статистике успеваемости студентов в системе управления обучением «Скорина»

А. Г. Савенко[✉]

*Институт информационных технологий
Белорусского государственного университета
информатики и радиоэлектроники,
ул. Козлова, 28, Минск, 220037, Беларусь
✉E-mail: savenko@bsuir.by*

Аннотация. Анализируются достоинства и недостатки дистанционной формы получения образования, специфики организации учебного процесса и использования учебного контента для такой формы обучения. Указываются проблемы оценки качества образовательного контента для дистанционного образовательного процесса и его эффективности. Описываются структура разрабатываемой системы управления обучением «Скорина» и механизмы создания учебного контента, процесса обучения, аттестации обучаемых и сбора статистики успеваемости. Приводятся математическая модель образовательного процесса системы управления обучением «Скорина» и подход к оценке знаний обучаемых.

Предлагается реализация функциональной схемы и алгоритма работы модуля интеллектуального анализа качества учебного контента системы управления обучением «Скорина». Модуль предназначен для выявления недостатков учебного контента и совершенствования материалов, объективно вызывающих трудности у студентов при изучении отдельных вопросов или тем изучаемых дисциплин на основании статистических данных успеваемости студентов. Связь конкретных частей учебного контента с определенными вопросами контрольных тестов реализуется посредством специальных меток, создаваемых при разработке содержания изучаемых дисциплин. Алгоритм интеллектуального анализа позволяет определять и исключать из статистической выборки студентов, характеристики успеваемости которых объективно не связаны с качеством учебного контента.

Ключевые слова: интеллектуальный анализ данных, академическая успеваемость, учебный контент, оценка качества процесса обучения, статистика успеваемости студентов, система управления обучением, графовая модель образовательного процесса

Для цитирования. Савенко, А. Г. Интеллектуальный анализ качества учебного контента по статистике успеваемости студентов в системе управления обучением «Скорина» / А. Г. Савенко // Информатика. – 2021. – Т. 18, № 2. – С. 58–71. <https://doi.org/10.37661/1816-0301-2021-18-2-58-71>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 30.12.2020
Подписана в печать | Accepted 13.01.2021
Опубликована | Published 26.06.2021

Intelligent analysis of the quality of educational content on statistics of student performance in the learning management system "Scorina"

Andrei G. Savenko✉

*Institute of Information Technologies of the Belarusian State
University of Informatics and Radioelectronics
st. Kozłova, 28, Minsk, 220037, Belarus
✉E-mail: savenko@bsuir.by*

Abstract. The article provides the analysis of the advantages and disadvantages of distance education, the specifics of educational process and the use of educational content for this form of education. The problems of assessing the quality of educational content for the distance educational process and its effectiveness are determined. The structure of the developed learning management system "Scorina" and the mechanisms for creating educational content, the learning process, attestation of students and the collection of progress statistics are described. A mathematical model of educational process of learning management system "Scorina", and approach to assessing the knowledge of students are presented.

The implementation of functional diagram and the algorithm of the module for the intelligent analysis of the quality of educational content of learning management system "Scorina" is proposed. The module is designed to identify the shortcomings in educational content and to improve the materials that could cause the difficulties for students when studying certain issues or topics of the disciplines are studied based on statistical data of student performance. The connection of specific parts of educational content with specific questions of control tests is implemented through special labels created during the development of the content of the studied disciplines. The mining algorithm allows to identify and exclude from the statistical sample the information of the students whose academic performance characteristics are not objectively related to the quality of educational content.

Keywords: data mining, academic performance, educational content, quality assessment of the learning process, student performance statistics, learning management system, graph model of the educational process

For citation. Savenko A. G. Intelligent analysis of the quality of educational content on statistics of student performance in the learning management system "Scorina". *Informatics*, 2021, vol. 18, no. 2, pp. 58–71 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-58-71>

Conflict of interest. The author declare of no conflict of interest.

Введение. В настоящее время наиболее перспективной и быстроразвивающейся формой получения образования является дистанционная. Это обусловлено как внешними по отношению к системе образования факторами (например, пандемией вируса Covid-19), так и внутренними (общей цифровизацией всех сфер общественной жизни, необходимостью наращивания экспорта образовательных услуг и др.). В качестве отдельных факторов развития дистанционной формы получения образования можно выделить ее преимущества в сравнении с традиционными формами: возможность внедрения и использования в учебном процессе виртуальной и дополненной реальностей [1, 2], инклюзивность образовательного процесса [3], экономическую эффективность дистанционной формы образования [4], использование инновационных информационно-коммуникационных технологий в образовании [5], реализацию игрового подхода в обучении [6], возрастающую конкуренцию на рынке образовательных услуг и т. д.

За последние годы для нужд образования были разработаны средства тестирования обучаемых, электронные учебные пособия, справочники и программы-тренажеры. Интенсивное развитие сетевых технологий, средств передачи информации в сети Интернет и их широкое распространение в самых разных сферах человеческой деятельности существенно расширили возможности учебных заведений и преподавателей при организации образовательного процесса.

Во время разработки системы управления обучением «Скорина» были проанализированы существующие формально-алгоритмическое обеспечение и программные системы для органи-

зации современного образовательного процесса [7, 8]. Были выявлены достоинства и недостатки используемых методов и средств для передачи учебно-методического материала, взаимодействия преподавателей и обучающихся в системе образования. Одним из основных недостатков существующих решений в области организации процесса обучения является недостаточно эффективное использование системного подхода, которое позволяет обеспечить процесс взаимосвязи и взаимодействия основных компонентов системы образовательного процесса: математического, алгоритмического и программного обеспечения.

Исследование проводилось с целью органически увязать все изложенные факты в единую систему образования, разработать модули системы управления обучением для усовершенствования существующего образовательного процесса (например, интеллектуального анализа качества учебного контента). Использование такого подхода может повысить эффективность организации современного образовательного процесса путем создания единого информационно-образовательного пространства для подготовки высококвалифицированных специалистов.

Проблемы оценки качества образовательного контента для дистанционного образовательного процесса и его эффективности. Наряду с наличием неоспоримых преимуществ дистанционная форма получения образования отличается, однако, недостаточной готовностью учреждений образования к качественной подготовке специалистов, обусловленной необходимостью изменения подхода к образовательному процессу, трансформацией роли преподавателя и изменением формы и содержания образовательного контента.

Рассмотрим проблему качества и приемлемости образовательного контента для дистанционной формы получения образования и подходы к решению данной проблемы. Традиционные учебные материалы (для очной и заочной форм получения образования) по своей форме и содержанию не в полной мере соответствуют возможностям современных систем управления обучением, используемым при дистанционном обучении, а также запросам со стороны потребителей образовательных услуг. Учебный контент для дистанционной формы обучения должен быть интерактивным, наглядным и соответствующим образом структурированным [9].

Как правило, студенты справляются с изучением учебных дисциплин по-разному. Это обусловлено различной базовой подготовкой, индивидуальными способностями к изучению материала, уровнем ответственности, прилагаемыми усилиями и индивидуальным состоянием обучающегося. Может сложиться ситуация, когда студент-отличник плохо справляется с изучением конкретного модуля, темы или дисциплины, о чем будет свидетельствовать статистика прохождения соответствующей промежуточной или текущей аттестации. Безусловно, данный факт еще не является показателем плохого качества учебного материала. Если же доля таких студентов будет более высокой (например, выше средних статистических показателей для соответствующих модулей у студентов очной формы получения образования) или большинство студентов будут испытывать затруднения при ответе на один или одни и те же вопросы, то это может свидетельствовать о недостатках методики преподавания и изложения материала [10]. Следовательно, существует необходимость улучшения определенной части учебного контента. Исходными данными для выявления таких причинно-следственных связей может выступать статистика успеваемости обучаемых по изучаемым дисциплинам (модулям дисциплины или конкретным разделам) [11]. Применительно к электронным средствам обучения (ЭСО), в частности, для дистанционной формы получения образования получение данной статистики автоматизировано и не представляет особых трудностей для ее использования и интеллектуального анализа.

Математическая модель образовательного процесса в системе управления обучением «Скорина». На рис. 1 показана упрощенная схема образовательного процесса в системе управления обучением «Скорина», где обозначено множество обучаемых $S = \{s_i, i = \overline{1, n}\}$, $|S| = n$, n – количество всех обучаемых; множество всех изучаемых дисциплин $D = \{d_j, j = \overline{1, m}\}$, $|D| = m$, m – количество всех дисциплин; подмножества обучаемых, изучающих дисциплину d_j , $Sd_j = \{sd_i, i = \overline{1, q_j}, q_j \leq n \mid sd_i \in S\}$, $Sd_j \subseteq S$, $|Sd_j| = q_j$, q_j – количество студентов, изучающих дисциплину d_j . При этом данные подмножества могут быть пересекающимися ($Sd_1 \cap Sd_2, \dots, Sd_1 \cap Sd_{m-1}; Sd_{m-1} \cap Sd_{m-2}, \dots$).

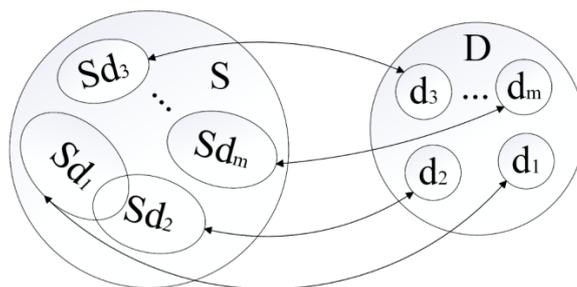


Рис. 1. Упрощенная схема процесса обучения в системе управления обучением «Скорина»

Представим данную схему в виде графа $G(V, E)$; $S, D \subset V$, где подмножеству вершин s_i инцидентны ребра, входящие в вершины d_j подмножества D (рис. 2). Тогда $e(sd_j, d_j) \in E$.

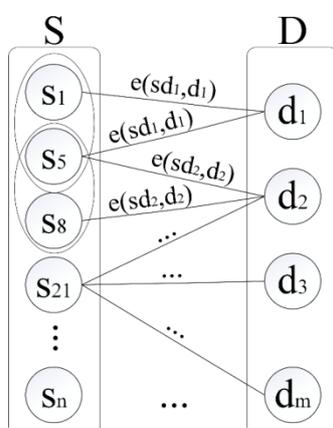


Рис. 2. Графовая модель «обучаемый – изучаемая дисциплина» в системе управления обучением «Скорина»

Система управления обучением «Скорина» имеет модульную структуру, учебный контент создается непосредственно в конструкторе дисциплин самой системы. Он может быть представлен в одном или нескольких форматах: текстово-графическом; видеоформате; в виде файлов презентаций, пособий, книг, виртуальных лабораторных и практических занятий. Структура учебной дисциплины имеет свою иерархию – три уровня вложенности (дисциплина – модуль – блок). Первый уровень иерархии – дисциплина d_j , второй – модуль дисциплины md_{jk} (конкретная глава или тема дисциплины). Количество модулей в дисциплине теоретически не ограничено и определяется необходимостью разбиения дисциплины на конкретные темы. Характеристическое свойство множества модулей Md_j дисциплины d_j имеет вид $Md_j = \{md_{ji}, i = \overline{1, k_j}, k_j \in N\}$, $|Md_j| = k_j$, k_j – количество модулей дисциплины d_j . Третий уровень – материалы по теме модуля (это блоки теоретического лекционного материала, блоки лабораторных работ по модулю, блоки практических занятий). После изучения и выполнения каждого блока материалов по модулю предусмотрено пробное и контрольное тестирование. Пробное тестирование предназначено для самоконтроля студентов и предполагает закрытое тестирование с автоматическим выставлением его результата. Контрольное тестирование может быть гибридным (открытые, закрытые вопросы, сопоставления и т. п.). Статистика прохождения контрольного тестирования сохраняется в базе данных и включает информацию по вопросам, на которые даны правильные и неправильные ответы; время прохождения тестирования; количество попыток прохождения теста; оценку за тестирование; минимальный проходной балл, установленный преподавателем [12, 13]. Обобщенная графовая модель образовательного процесса с контрольным тестированием на уровне модулей в системе управления обучением «Скорина» представлена на рис. 3.

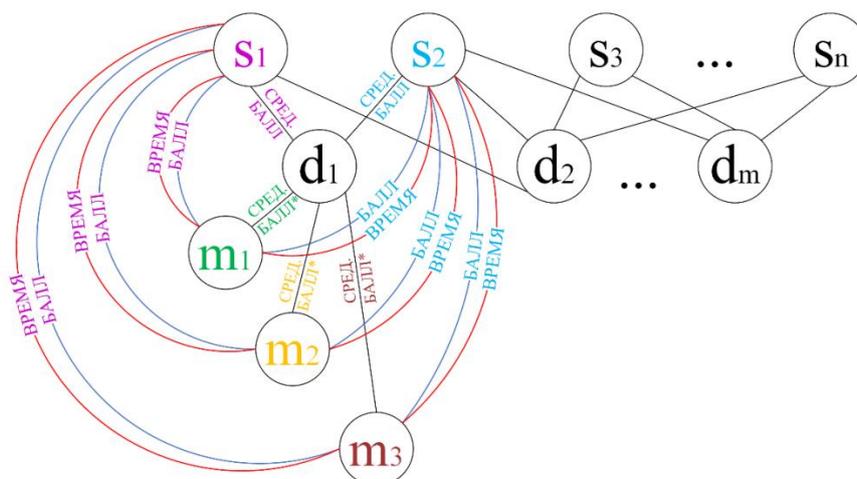


Рис. 3. Обобщенная графовая модель образовательного процесса с контрольным тестированием на уровне модулей

Как видно из рис. 3, модель представляет собой взвешенный мультиграф. Весом ребер, инцидентных вершинам s_i и m_k , являются время прохождения контрольного тестирования по модулю m_k и оценка контрольного теста по модулю m_k , полученная конкретным обучаемым (на рис. 3 разные обучаемые и вес инцидентных им ребер обозначены различным цветом). Весом ребер, инцидентных вершинам s_i и d_j , является средний балл конкретных студентов по конкретным дисциплинам (на рис. 3 обозначено как «сред. балл»). Весом ребер, инцидентных вершинам m_k и d_j , являются средние баллы прохождения контрольного тестирования по модулю всех обучаемых, изучающих данный модуль (на рис. 3 обозначено как «сред. балл*») и каждый модуль обозначен разным цветом).

При создании учебного контента и вопросов контрольного тестирования ставятся тематические метки (определенный идентификатор, например «1», «2», «3», ...). Тематическая метка идентифицирует определенную часть учебного контента (абзац или предложение текста, определенный хронометраж видеолекции, определенные слайды презентации или ее всю). При создании контрольного теста каждый его вопрос соотносится с соответствующей тематической меткой учебного контента. Таким образом, можно отслеживать тематические метки, соответствующие вопросам, на которые были даны верные и неверные ответы при контрольном тестировании. Графовая модель GM , иллюстрирующая связь определенных частей образовательного контента и вопросов контрольных тестов, показана на рис. 4.

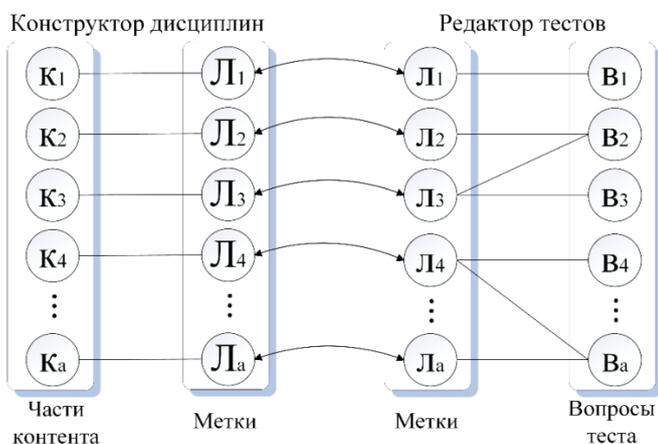


Рис. 4. Графовая модель связи определенных частей учебного контента с вопросами контрольного теста

Как видно из рис. 4, максимальный порядок графа $|V(GM)| = 4a$, где a – количество выделенных частей учебного контента при его создании в конструкторе дисциплин. Степень вершин графа $d(v) > 1$ может быть только у вершин, образующих подмножества вопросов контрольных тестов и меток этих вопросов, формируемых в редакторе тестов. Это значит, что для ответа на вопрос контрольного тестирования необходимы знания, содержащиеся в нескольких частях учебного контента, обозначенных соответствующими метками. Минимальная степень вершин графа $\delta(GM) = 1$.

Использование графовой модели образовательного процесса имеет свои преимущества. Во-первых, геометрическая интерпретация графовой модели дает возможность наглядного анализа данных и процессов. Во-вторых, матричная интерпретация обеспечивает эффективную обработку информации на ЭВМ, что позволяет автоматизировать процесс получения статистических данных успеваемости обучаемых в системе управления обучением «Скорина». Так как модель представляет собой мультиграф, количество связей между вершинами не ограничено. За счет этого в схему данных можно добавлять новые отношения, дополняя тем самым статистические данные. Кроме того, графовая схема данных имеет ряд преимуществ в сравнении с реляционной [14, 15].

Оценка знаний обучаемых и получение статистики успеваемости. При создании контрольного теста по модулю дисциплины или блоку модуля преподаватель для каждого вопроса выбирает его тип: закрытый (необходимо выбрать правильный (или правильные) ответ из предложенных), открытый (необходимо дать развернутый ответ на поставленный вопрос) или сопоставление (в соответствии с поставленным заданием необходимо соотнести определенные блоки или расположить их в установленной последовательности). На вопросы закрытого типа и сопоставления преподаватель помечает правильный вариант ответа. Также предусмотрено задание веса вопроса в формировании итоговой оценки по контрольному тесту. Оценка знаний по результатам прохождения тестов с вопросами открытого типа и сопоставления производится системой в автоматическом режиме с учетом весов вопросов (если они заданы) или пропорционально количеству вопросов (если веса не заданы). Оценка контрольных тестов, содержащих вопросы открытого типа, выставляется в личном кабинете преподавателя после проверки развернутых ответов обучаемого.

Важными статистическими данными успеваемости обучаемых в системе управления обучением «Скорина» являются также количество правильных и неправильных ответов на каждый вопрос контрольного теста всеми студентами, прошедшими данный тест, количество попыток и время прохождения контрольного теста каждым обучаемым. Количество правильных и неправильных ответов определяет трудность заданий контрольного теста. Доля правильных ответов определяется по формуле

$$r_i = \frac{Right_i}{z}, \quad (1)$$

где $Right_i$ – количество всех правильных ответов на i -й вопрос теста; z – количество обучаемых, ответивших на i -й вопрос теста; доля неправильных ответов – по формуле

$$w_i = \frac{Wrong_i}{z}, \quad (2)$$

где $Wrong_i$ – количество всех неправильных ответов на i -й вопрос теста.

Общий уровень успеваемости определяется как среднее значение балла, полученного всеми обучаемыми, изучающими конкретный блок модуля (модуль дисциплины):

$$B_{cp} = \frac{\sum_{i=1}^q B_i}{q}, \quad (3)$$

где B_i – балл каждого i -го обучаемого, изучающего блок модуля (модуль дисциплины).

Сумма квадратов отклонений значений баллов от среднего арифметического называется вариацией и определяется выражением

$$RSS_A = \sum (A_i - B_{cp})^2, \quad (4)$$

где A_i – количество правильных ответов обучаемого.

Чем меньше значение вариации, тем более похожа сложность заданий контрольного теста и все обучающиеся решают задачу одинаково (например, не допускают ошибок или все допускают максимальное количество ошибок). Проанализировав значение показателя вариации, можно сделать вывод, что задача является либо слишком простой, либо слишком сложной. Однако значение RSS_A прямо пропорционально зависит от количества обучаемых, изучающих данную дисциплину (чем больше обучаемых, тем больше значение суммы квадратов отклонений значений баллов от среднего арифметического). Более объективным показателем будет являться дисперсия, которая учитывает количество обучаемых q , изучающих определенный блок модуля (модуль дисциплины), и определяется по формуле

$$Var_A^2 = \frac{RSS_A}{q-1}. \quad (5)$$

где $q > 1$.

Маленькое значение дисперсии говорит о том, что большинство обучаемых либо верно выполняют задания контрольного теста, либо, наоборот, неверно, т. е. задания являются либо очень простыми, либо очень сложными.

Исходные данные и функциональная схема модуля интеллектуального анализа качества учебного контента. Модуль интеллектуального анализа качества учебного контента предназначен для выявления и совершенствования материалов учебного контента, объективно вызывающих трудности у студентов при изучении дисциплин.

В соответствии с общими принципами интеллектуального анализа информации [16, 17] для его проведения необходимо располагать нормализованными исходными данными. В рассматриваемом случае исходными данными (в том числе полученными на основании статистики прохождения контрольных тестов всех студентов) являются:

- рациональный процент успеваемости студентов. На первом этапе можно взять средний балл прохождения аттестации студентов по классической очной форме получения образования. Со временем он будет уточняться и оптимизироваться для студентов, обучаемых в системе управления обучением «Скорина»;
- абсолютный процент успеваемости студентов в системе управления обучением «Скорина», т. е. процент студентов, прошедших контрольное тестирование на балл не ниже среднего от общего числа студентов по определенному блоку модуля (или модулю дисциплины);
- время прохождения контрольного теста каждым студентом, изучающим блок модуля (модуль дисциплины);
- успеваемость каждого студента по другим модулям дисциплины;
- успеваемость каждого студента по другим дисциплинам;
- тематические метки вопросов, на которые были даны неверные ответы каждого изучающего блок модуля студента, связанные с соответствующими тематическими метками конкретных частей учебного контента. Например, метка «1», которой обозначен конкретный вопрос в тесте, – это та же самая метка «1», которой обозначен определенный абзац в теоретическом учебном материале. Такие метки предназначены для определения конкретной части учебного материала, при изучении которого у студентов возникают трудности;
- максимальное значение рациональной успеваемости за время выполнения алгоритма.

При выявлении определенных причинно-следственных связей между полученными статистическими данными и качеством соответствующего образовательного материала модуль визуализирует и отправляет на оценку результат анализа эксперту (преподавателю) для принятия решения и совершенствования учебного материала.

Обобщенная функциональная схема модуля интеллектуального анализа показана на рис. 5.

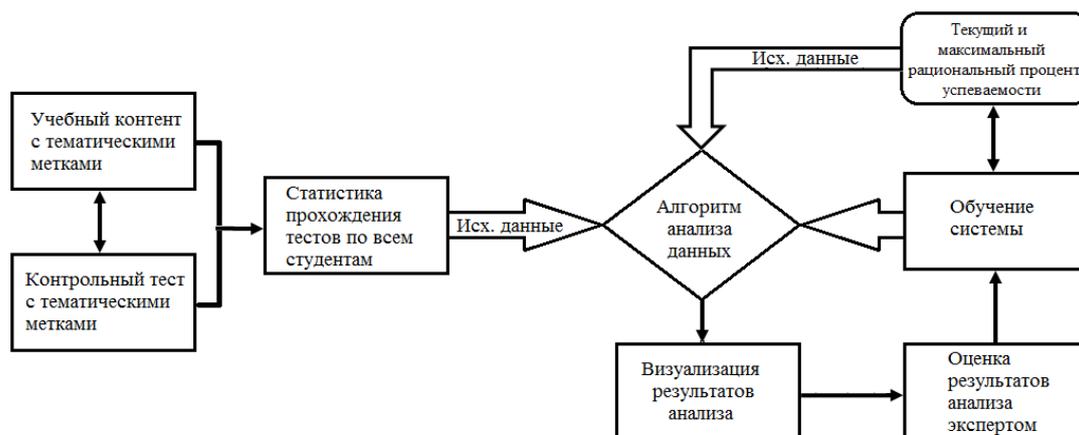


Рис. 5. Обобщенная функциональная схема модуля интеллектуального анализа

Алгоритм работы модуля интеллектуального анализа качества учебного контента. Работу модуля интеллектуального анализа можно разделить на следующие этапы:

- 1) определение наличия проблемы или причин ее отсутствия;
- 2) уточнение статистических данных;
- 3) вывод результатов анализа и действия эксперта.

На первом этапе алгоритм сравнивает значение абсолютной успеваемости всех студентов по блоку модуля с текущим значением рациональной успеваемости по данному блоку. Если абсолютная успеваемость меньше рациональной, это свидетельствует о том, что по данному блоку модуля студенты справляются хуже ожидаемого и алгоритм переходит ко второму этапу. Если же абсолютная успеваемость, наоборот, выше рациональной, значит, значение рациональной успеваемости может быть занижено и его необходимо корректировать. Для этого сравнивается текущее значение абсолютной успеваемости с максимальным значением рациональной успеваемости за время работы модуля. Если первое окажется больше, то будет предложено увеличить значение текущей рациональной успеваемости. В противном случае изменения не потребуются.

На втором этапе уточняются статистические данные всех студентов, абсолютная успеваемость которых ниже на заданное значение текущей рациональной успеваемости. С точки зрения математической модели необходимо разбить множество обучаемых S на два подмножества S' и S^* (рис. 6). Для примера, изображенного на рис. 6 (контрольное тестирование на уровне модуля дисциплины), $S' = \{s'_i, i = \overline{1, q} \mid e(sd_j, d_j) < B_{\text{рац}} \times x\}$, $S' \subseteq S$, $S^* = \{s^*_i, i = \overline{1, q} \mid e(sd_j, d_j) \geq B_{\text{рац}} \times x\}$, $S^* \subseteq S$, где $B_{\text{рац}}$ – значение текущей рациональной успеваемости, x – коэффициент, определяющий значение, на которое абсолютная успеваемость обучаемых ниже текущей рациональной успеваемости. Теоретически множество $X = \{x \in R \mid 0 < x \leq 1\}$. С практической точки зрения целесообразно использовать значения коэффициента x из диапазона от 0,7 до 0,95.

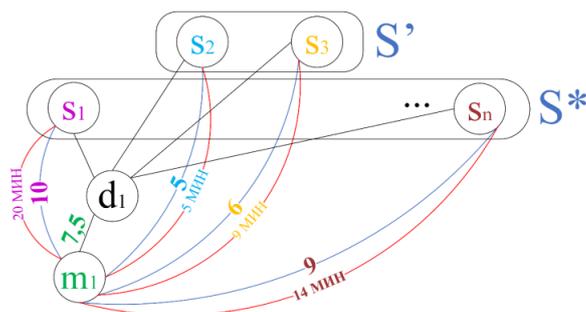


Рис. 6. Пример разбиения множества обучаемых на подмножества по признаку текущей успеваемости с контрольным тестированием на уровне модуля дисциплины

Далее анализируются такие данные, как время, затраченное на прохождение контрольного теста, успеваемость студентов по другим модулям дисциплины и успеваемость во временном срезе. Таким образом, из общей выборки исключаются данные студентов, характеристики успеваемости которых являются низкими по внутренним причинам (рис. 7) и не связаны с качеством учебного контента, а по каждому такому случаю студенту отправляется соответствующая рекомендация. Например, с точки зрения качества учебного контента о недостоверности характеристик успеваемости может свидетельствовать несопоставимо малое значение времени, затраченного на прохождение контрольного теста (например, студент прошел тест за 10 с, наугад расставив ответы). Пример исключения данных таких обучаемых из статистической выборки представлен на рис. 8. Также снижение успеваемости конкретного студента по всем дисциплинам в определенный промежуток времени может указывать на наличие внутренних причин (психологическое состояние, загруженность на работе, снижение мотивации и интереса к выбранной специальности и т. д.). Более высокие показатели по другим модулям этой же дисциплины свидетельствуют о недостатках учебного контента анализируемого блока модуля. Так происходит уточнение исходных данных для анализа и исключение из выборки данных, объективно не зависящих от качества учебного контента. Вместе с тем следует отметить, что в алгоритме не обрабатываются и не учитываются вопросы, по которым были получены верные ответы от студентов. Установить, были эти верные ответы случайными или обдуманными, не представляется возможным.

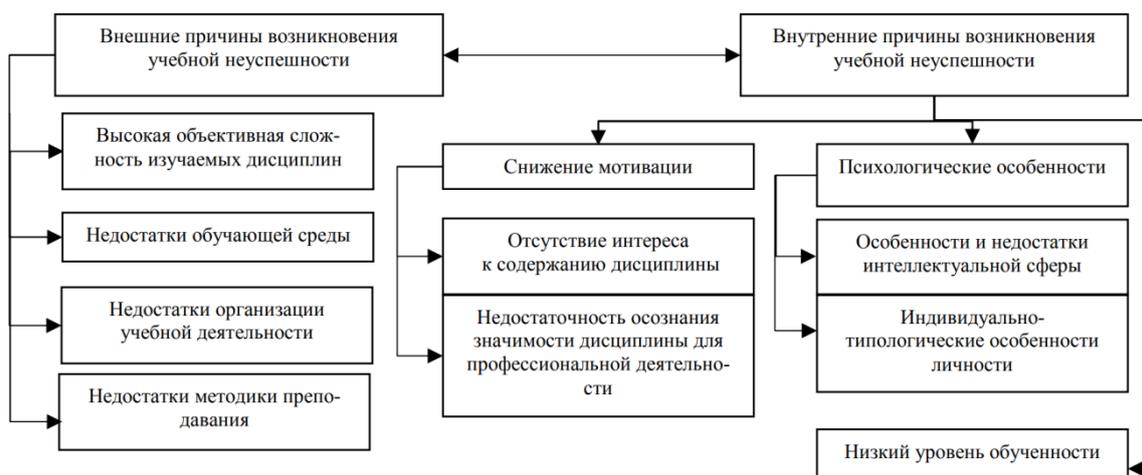


Рис. 7. Причины возникновения учебной неуспешности студентов [10]

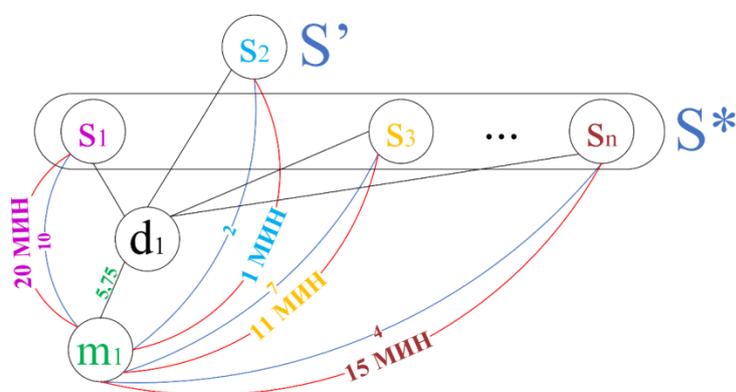


Рис. 8. Выделение из множества обучаемых тех, кто прошел контрольное тестирование на уровне модуля дисциплины за время, неадекватное заданию

После исключения из выборки статистических данных, объективно не зависящих от качества учебного контента, повторяется первый этап – определяется необходимость совершенствования учебного контента с соответствующими метками.

На третьем этапе осуществляется вывод результатов выполнения алгоритма. В случае если была установлена зависимость низкой успеваемости студентов от качества учебного контента, соответствующее сообщение отправляется эксперту (преподавателю) для принятия решения о необходимости совершенствования части учебного контента с соответствующими метками. Часть учебного контента с соответствующими метками визуализируется, и эксперт (преподаватель) либо принимает решение о его переработке, либо не вносит изменений и пишет рекомендации, которые отправляются всем студентам, испытывающим затруднения при ответе на вопросы, связанные с данной частью учебного контента.

Обобщенная блок-схема алгоритма работы модуля интеллектуального анализа изображена на рис. 9.

Таким образом, представленный алгоритм интеллектуального анализа позволяет установить причинно-следственные связи между низкой успеваемостью студентов по изучаемому блоку модуля дисциплины (модулю дисциплины) и качеством учебного контента; идентифицировать конкретные части образовательного контента, нуждающиеся в совершенствовании, и представить их эксперту для принятия решения. Используя дополнительные базы данных, по такому же алгоритму можно проводить анализ способностей конкретного студента к изучению определенных дисциплин, например, при выборе специализации по специальности.

Блок-схема этапа уточнения статистических данных алгоритма представлена на рис. 10.

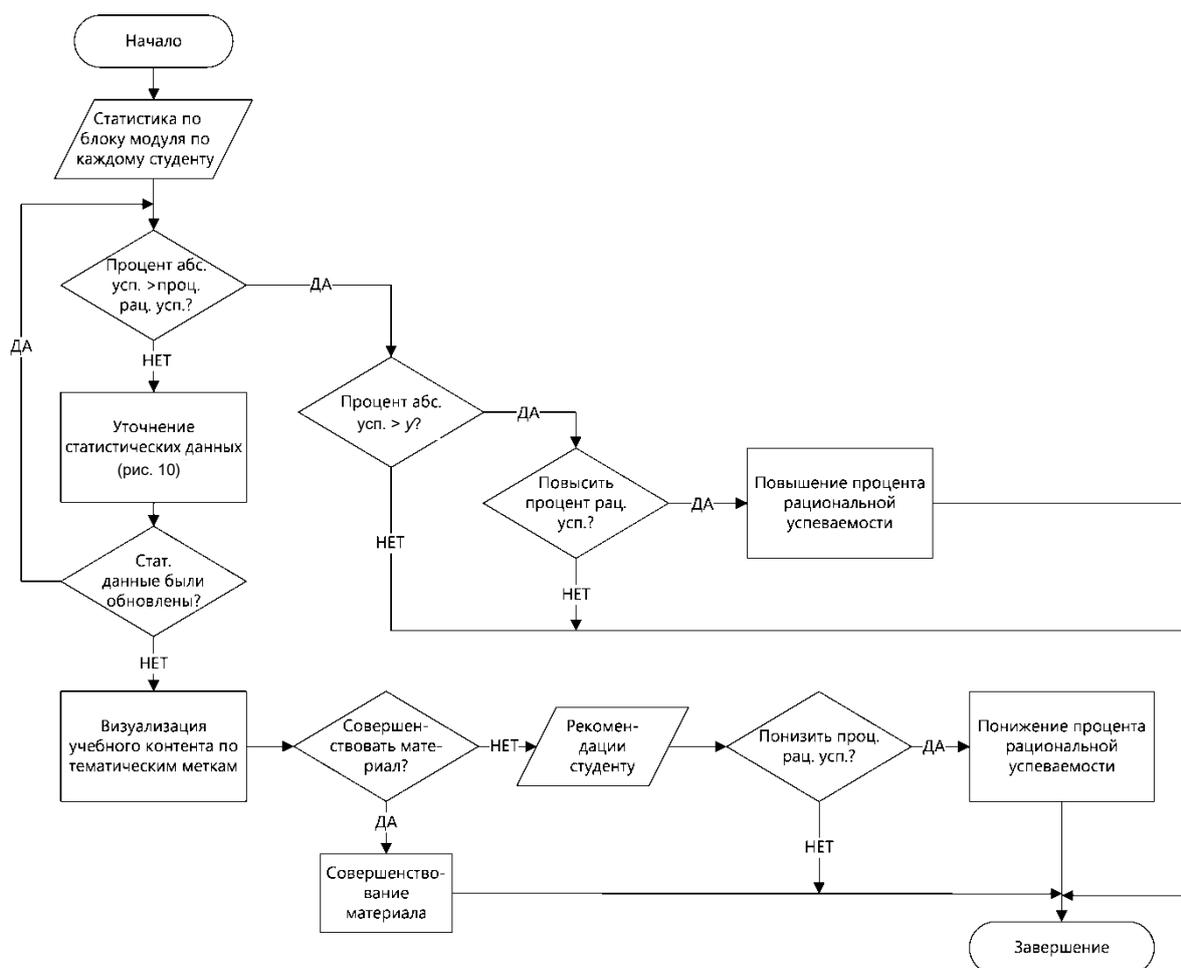


Рис. 9. Обобщенная блок-схема алгоритма работы модуля интеллектуального анализа

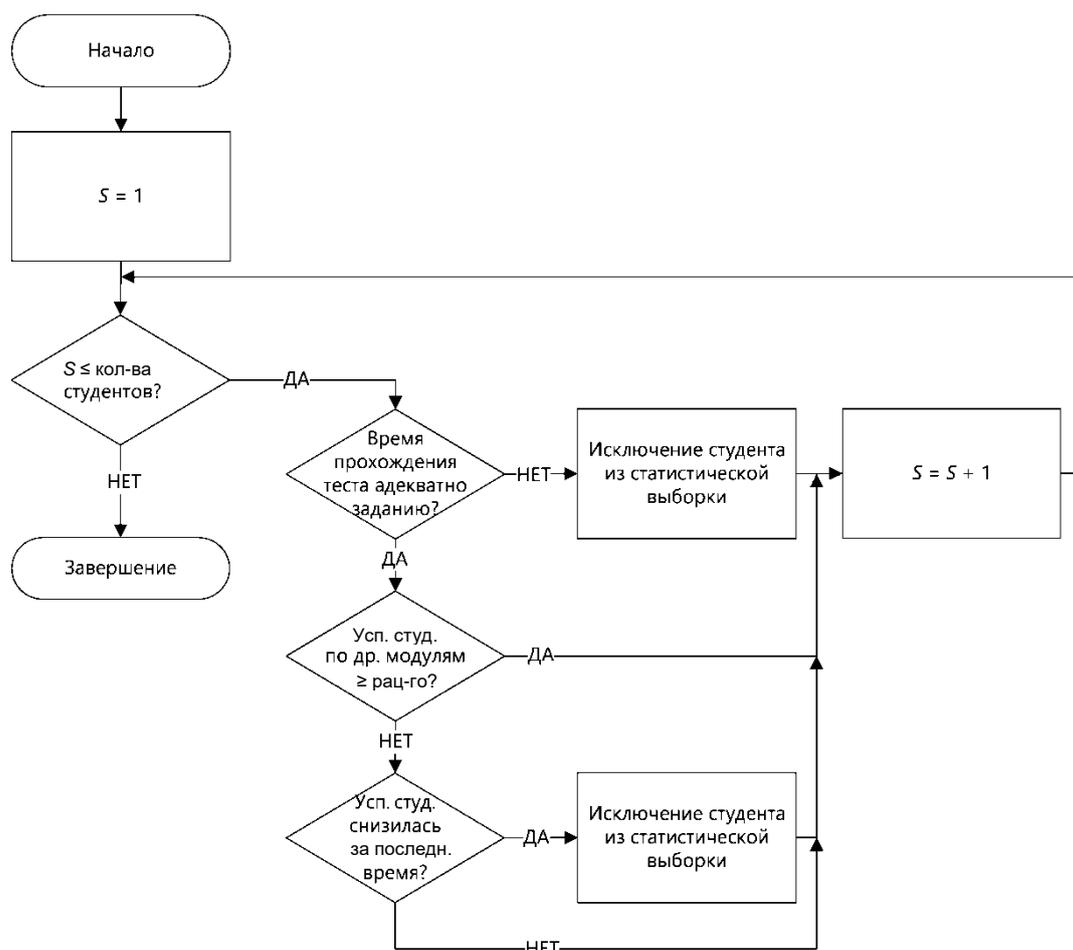


Рис. 10. Блок-схема этапа уточнения статистических данных алгоритма

Заключение. В результате проведенного научного исследования были проанализированы достоинства и недостатки дистанционной формы обучения, определены специфические особенности учебного контента для данной формы получения образования. Выявлены проблемы оценки качества образовательного контента для дистанционного образовательного процесса и его эффективности. Разработана математическая модель образовательного процесса системы управления обучением «Скорина» и ее программно-алгоритмическая реализация. Предложено математическое обеспечение оценки качества знаний обучающихся, которое в силу своей универсальности имеет широкую сферу применения в различных формах обучения. Представлена обобщенная схема модуля интеллектуального анализа качества учебного контента по статистике успеваемости студентов, отражающая его функциональные возможности в современном дистанционном и других видах образовательного процесса. Разработанный алгоритм работы модуля интеллектуального анализа системы управления обучением «Скорина» предназначен для анализа и адаптации материалов учебного контента к обучаемым с учетом их индивидуальных возможностей при освоении изучаемого материала. Все это позволяет обеспечить высокую гибкость, производительность, универсальность и адаптивность современного образовательного процесса. Представленный алгоритм при доработке может быть также использован обучаемым для выбора специализации. Следует отметить актуальность предложенных разработок, а также перспективы их дальнейшего развития в системе управления обучением «Скорина» и внедрения данной системы в образовательный процесс всех форм обучения для повышения качества подготовки специалистов.

Список использованных источников

1. Савенко, А. Г. Преимущества и перспективы использования виртуальной и дополненной реальности в дистанционном образовательном процессе / А. Г. Савенко // Дистанционное обучение – образовательная среда XXI века : материалы X Междунар. науч.-метод. конф., Минск, 7–8 дек. 2017 г. – Минск : БГУИР, 2017. – С. 119.
2. Савенко, А. Г. Виртуальная реальность как способ получения и доставки учебного контента / А. Г. Савенко, Н. А. Кукалев, А. Г. Савенко // Высшее техническое образование: проблемы и пути развития = Engineering education: challenges and developments : материалы IX Междунар. науч.-метод. конф., Минск, 1–2 нояб. 2018 г. – Минск : БГУИР, 2018. – С. 394–397.
3. Савенко, А. Г. Преимущества и реализация дистанционного образовательного процесса для лиц с особыми потребностями / А. Г. Савенко // Непрерывное профессиональное образование лиц с особыми потребностями : сб. ст. Междунар. науч.-практ. конф., Минск, 14–15 дек. 2017 г. – Минск : БГУИР, 2017. – С. 106–108.
4. Карпекин, И. А. Преимущества и эффективность внедрения дистанционной формы образования в образовательный процесс учреждений образования любого типа / И. А. Карпекин, А. Г. Савенко // Дистанционное обучение – образовательная среда XXI века : материалы XI Междунар. науч.-метод. конф., Минск, 12–13 дек. 2019 г. – Минск : БГУИР, 2019. – С. 136–137.
5. Суский, А. А. Преимущества и перспективы внедрения нейронных сетей в образовательный процесс как инструмент повышения качества подготовки специалистов / А. А. Суский, А. Г. Савенко // Высшее техническое образование: проблемы и пути развития = Engineering education: challenges and developments : материалы IX Междунар. науч.-метод. конф., Минск, 1–2 нояб. 2018 г. – Минск : БГУИР, 2018. – С. 454–456.
6. Савенко, А. Г. Игровой подход в обучении программированию детей и подростков / А. Г. Савенко // Информационные технологии в технических, политических и социально-экономических системах : материалы Междунар. науч.-техн. конф. / Белорусский национальный технический университет. – Минск : БНТУ, 2018. – С. 30.
7. Савенко, А. Г. Анализ технологий современного дистанционного образовательного процесса в Республике Беларусь и перспективы их развития / А. Г. Савенко // Актуальные вопросы профессионального образования : тез. докл. II Междунар. науч.-практ. конф., Минск, 11 апр. 2019 г. – Минск : БГУИР, 2019. – С. 227–228.
8. Савенко, А. Г. Проблемы в нормативно-правовом поле и пути их решения при внедрении эффективного дистанционного образовательного процесса в учреждениях образования Республики Беларусь / А. Г. Савенко // Дистанционное обучение – образовательная среда XXI века : материалы XI Междунар. науч.-метод. конф., Минск, 12–13 дек. 2019 г. – Минск : БГУИР, 2019. – С. 265.
9. Скудняков, Ю. А. Структурная организация процесса дистанционного обучения / Ю. А. Скудняков, А. Г. Савенко, А. В. Матвеев // Дистанционное обучение – образовательная среда XXI века : материалы X Междунар. науч.-метод. конф., Минск, 7–8 дек. 2017 г. – Минск : БГУИР, 2017. – С. 82.
10. Останкина, Е. Н. О факторах учебной неуспешности студентов / Е. Н. Останкина // Вестник Череповецкого гос. ун-та. – 2013. – Т. 1, № 1(54). – С. 127–131.
11. Анализ статистики успеваемости студентов как средство повышения качества образования / Е. А. Ерохина [и др.] // Инженерное образование. – 2014. – № 15. – С. 200–205.
12. Савенко, А. Г. Ротационно-гибридная модель современного образовательного процесса и ее программно-алгоритмическая реализация / А. Г. Савенко, Ю. А. Скудняков // Информационные системы и технологии – 2019 : сб. материалов XXV Междунар. науч.-техн. конф., Нижний Новгород, 19 апр. 2019 г. / НГТУ им. Р. Е. Алексеева. – Н. Новгород, 2019. – С. 451–458.
13. Савенко, А. Г. Один из подходов к организации современного образовательного процесса / А. Г. Савенко, Ю. А. Скудняков // Информатика. – 2021. – Т. 18, № 1. – С. 96–104. <https://doi.org/10.37661/1816-0301-2021-18-1-96-104>
14. Курочкин, А. В. Агрегация и индексирование данных нескольких источников на основе графовой модели в базах данных медицинских экспертных систем / А. В. Курочкин, В. С. Садов // Информатика. – 2020. – Т. 17, № 3. – С. 25–35. <https://doi.org/10.37661/1816-0301-2020-17-3-25-35>
15. Пилецкий, И. И. Граф знаний и машинное обучение как базис методологии искусственного интеллекта в обучении / И. И. Пилецкий, М. П. Батура, Н. А. Волорова // BIG DATA and Advanced Analytics = BIG DATA и анализ высокого уровня : сб. науч. ст. VII Междунар. науч.-практ. конф., Минск, 19–20 мая 2021 г. / редкол.: В. А. Богуш [и др.]. – Минск : Бестпринт, 2021. – С. 198–209.

16. Tatur, M. M. Intelligent data analysis: from theory to practice / M. M. Tatur, N. A. Iskra // Открытые семантические технологии проектирования интеллектуальных систем (OSTIS–2018) : материалы Международ. науч.-техн. конф. – Минск : БГУИР, 2018. – С. 171–174.

17. Интеллектуальный анализ данных и облачные вычисления / Татур М. М. [и др.] // Доклады БГУИР. – 2019. – № 6(124). – С. 62–71. <https://doi.org/10.35596/1729-7648-2019-124-6-62-71>

References

1. Savenko A. G. *Advantages and prospects of using virtual and augmented reality in the distance educational process*. Distantionnoe obuchenie – obrazovatel'naya sreda XXI veka: materialy X Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 7–8 dekabrya 2017 g. [*Distance Learning – Educational Environment of the XXI Century: Materials of the X International Scientific and Methodological Conference, Minsk, 7–8 December 2017*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2017, p. 119 (In Russ.).

2. Savenko A. G., Kukalev N. A., Savenko A. G. *Virtual reality as a way of receiving and delivering educational content*. Vysshee tehicheskoe obrazovanie: problem i puti razvitiya: materialy IX Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 1–2 nojabrya 2018 g. [*Engineering Education: Challenges and Developments: Materials of the IX International Scientific and Methodological Conference, Minsk, 1–2 November 2018*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2018, pp. 394–397 (In Russ.).

3. Savenko A. G. *Advantages and implementation of distance learning process for people with special needs*. Nepreryvnoe professional'noe obrazovanie lits s osobymi potrebnyami: sbornik statej Mezhdunarodnoj nauchno-prakticheskoy konferentsii, Minsk, 14–15 dekabrya 2017 g. [*Continuing Professional Education of Persons with Special Needs: Collection of Articles of the International Scientific and Practical Conference, Minsk, 14–15 December 2017*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2017, pp. 106–108 (In Russ.).

4. Karpekin I. A., Savenko A. G. *Advantages and effectiveness of introducing distance education into the educational process of educational institutions of any type*. Distantionnoe obuchenie – obrazovatel'naya sreda XXI veka: materialy XI Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 12–13 dekabrya 2019 g. [*Distance Learning – Educational Environment of the XXI Century: Materials of the XI International Scientific and Methodological Conference, Minsk, 12–13 December 2019*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2019, pp. 136–137 (In Russ.).

5. Suskij A. A., Savenko A. G. *Advantages and prospects of introducing neural networks into the educational process as a tool for improving the quality of training of specialists*. Vysshee tehicheskoe obrazovanie: problemy i puti razvitiya: materialy IX Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 1–2 nojabrya 2018 g. [*Engineering Education: Challenges and Developments: Materials of the IX International Scientific and Methodological Conference, Minsk, 1–2 November 2018*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2018, pp. 454–456 (In Russ.).

6. Savenko A. G. *A playful approach to teaching programming to children and adolescents*. Informatsionnye tehnologii v tehicheskikh, politicheskikh i sotsial'no-ekonomicheskikh sistemah: materialy Mezhdunarodnoj nauchno-tehicheskoy konferentsii [*Information Technologies in Technical and Socio-Economic Systems: Materials of the Scientific and Technical Conference*]. Minsk, Belorusskij nacional'nyj tehicheskij universitet, 2018, p. 30 (In Russ.).

7. Savenko A. G. *Analysis of technologies of modern distance educational process in the Republic of Belarus and prospects for their development*. Aktualnye voprosy professional'nogo obrazovaniya: tezisy dokladov II Mezhdunarodnoj nauchno-prakticheskoy konferentsii, Minsk, 11 aprelja 2019 g. [*Topical Issues of Vocational Education: Abstracts of the II International Scientific and Practical Conference, Minsk, 11 April 2019*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2019, pp. 227–228 (In Russ.).

8. Savenko A. G. *Problems in the regulatory and legal field and ways to solve them in the implementation of an effective distance educational process in educational institutions of the Republic of Belarus*. Distantionnoe obuchenie – obrazovatel'naya sreda XXI veka: materialy XI Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 12–13 dekabrya 2019 g. [*Distance Learning – Educational Environment of the XXI Century: Materials of the XI International Scientific and Methodological Conference, Minsk, 12–13 December 2019*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2019, p. 265 (In Russ.).

9. Skudnyakov Y. A., Savenko A. G., Matveev A. V. *Structural organization of the distance learning process*. Distantionnoe obuchenie – obrazovatel'naya sreda XXI veka: materialy X Mezhdunarodnoj nauchno-metodicheskoy konferentsii, Minsk, 7–8 dekabrya 2017 g. [*Distance Learning – Educational Environment of*

the XXI Century: Materials of the X International Scientific and Methodological Conference, Minsk, 7–8 December 2017]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2017, p. 82 (In Russ.).

10. Ostantkina E. N. *About the factors of academic failure of students*. Vestnik Cherepovetskogo gosudarstvennogo universiteta [*Bulletin of the Cherepovets State University*], 2013, vol. 1, no. 1(54), pp. 127–131 (In Russ.).

11. Erohina E. A., Hruslova D. V., Zhurin U. V., Klyshinskij E. S. *Analysis of student performance statistics as a means of improving the quality of education*. Inzhenernoe obrazovanie [*Engineering Education*], 2014, no. 15, pp. 200–205 (In Russ.).

12. Savenko A. G., Skudnyakov Y. A. *Rotational-hybrid model of the modern educational process and its program-algorithmic realization*. Informatsionnye sistemy i tehnologii – 2019: sbornik materialov XXV Mezhdunarodnoj nauchno-tehnicheskoy konferentsii, Nizhnij Novgorod, 19 aprelja 2019 g. [*Information Systems and Technologies – 2019: Collection of Materials of the XXV International Scientific and Technical Conference, Nizhnij Novgorod, 19 April 2019*]. Nizhnij Novgorod, Nizhegorodskij gosudarstvennyj tehničeskij universitet im. R. E. Alekseeva, 2019, pp. 451–458 (In Russ.).

13. Savenko A. G., Skudnyakov Yu. A. One of the approaches to the organization of the modern educational process. *Informatika [Informatics]*, 2021, vol. 18, no. 1, pp. 96–104 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-1-96-104>

14. Kurochkin A. V., Sadov V. S. *Aggregation and indexing of data from several sources based on a graph model in databases of medical expert systems*. *Informatika [Informatics]*, 2020, vol. 17, no. 3, pp. 25–35 (In Russ.). <https://doi.org/10.37661/1816-0301-2020-17-3-25-35>

15. Pileckij I. I., Batura M. P., Volorova N. A. *Knowledge graph and machine learning as the basis of artificial intelligence methodology in teaching*. BIG DATA i analiz vysokogo urovnja: sbornik nauchnyh statej VII Mezhdunarodnoj nauchno-praktičeskoy konferencii, Minsk, 19–20 maja 2021 g. [*BIG DATA and Advanced Analytics : Collection of Scientific Articles of the VII International Scientific and Practical Conference, Minsk, 19–20 May 2021*]. Minsk, Bestprint, 2021, pp. 198–209 (In Russ.).

16. Tatur M. M., Iskra N. A. *Intelligent data analysis: from theory to practice*. Otkrytye semanticheskie tehnologii proektirovaniya intellektual'nyh sistem (OSTIS-2018): materialy Mezhdunarodnoj nauchno-tehnicheskoy konferentsii [*Open Semantic Technologies for the Design of Intelligent Systems (OSTIS-2018): Materials of the International Scientific and Technical Conference*]. Minsk, Belorusskij gosudarstvennyj universitet informatiki i radioelektroniki, 2018, pp. 171–174.

17. Tatur M. M., Lukashevich M. M., Pertsev D. Y., Iskra N. A. *Intelligent data analysis and cloud computing*. Doklady Belorusskogo gosudarstvennogo universiteta informatiki i radioelektroniki [*Reports of the Belarusian State University of Informatics and Radioelectronics*], 2019, vol. 6(124), pp. 62–71 (In Russ.). <https://doi.org/10.35596/1729-7648-2019-124-6-62-71>

Информация об авторе

Савенко Андрей Геннадьевич, аспирант, старший преподаватель кафедры информационных систем и технологий, факультет компьютерных технологий, Институт информационных технологий Белорусского государственного университета информатики и радиоэлектроники.
E-mail: savenko@bsuir.by

Information about the author

Andrei G. Savenko, Postgraduate Student, Senior Lecturer, Department of Information Systems and Technologies, Faculty of Computer Technologies, Institute of Information Technologies of the Belarusian State University of Informatics and Radioelectronics.
E-mail: savenko@bsuir.by



УДК 621.383
<https://doi.org/10.37661/1816-0301-2021-18-2-72-82>

Оригинальная статья
Original Paper

Оценка влияния интенсивности оптического сигнала на вероятность ошибочной регистрации данных в однофотонном канале связи

А. М. Тимофеев

Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
✉E-mail: tamvks@mail.ru

Аннотация. При измерении маломощных оптических сигналов в однофотонных каналах связи приемные модули должны обеспечивать минимальную вероятность ошибочной регистрации данных. В этой связи целесообразно использовать счетчики фотонов, которые являются высокочувствительными, однако характеризуются наличием ошибок при регистрации данных. Поэтому цель работы – оценка влияния интенсивности оптического сигнала на вероятность ошибочной регистрации данных в однофотонном канале связи с приемным модулем на основе счетчика фотонов.

По результатам экспериментальных исследований получены зависимости вероятности ошибочной регистрации двоичных символов «0» $P_{\text{ош0}}$ от интенсивности оптического сигнала J_0 , используемого для передачи этих символов при различных напряжениях питания лавинного фотоприемника $U_{\text{пит}}$. Установлено, что с увеличением интенсивности оптического сигнала зависимости $P_{\text{ош0}}(J_0)$ сначала спадают, достигая своего наименьшего значения, а затем растут. Выполненная оценка показала, что наименьшая вероятность ошибочной регистрации двоичных символов «0» для исследованного канала связи равна $17,56 \cdot 10^{-2}$ и соответствует $J_0 = 52,41 \cdot 10^{-2}$ отн. ед. и $U_{\text{пит}} = 52,54$ В.

Ключевые слова: канал связи, однофотонная передача информации, вероятность ошибочной регистрации данных, счетчик фотонов, мертвое время

Для цитирования. Тимофеев, А. М. Оценка влияния интенсивности оптического сигнала на вероятность ошибочной регистрации данных в однофотонном канале связи / А. М. Тимофеев // Информатика. – 2021. – Т. 18, № 2. – С. 72–82. <https://doi.org/10.37661/1816-0301-2021-18-2-72-82>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 14.04.2021
Подписана в печать | Accepted 03.05.2021
Опубликована | Published 26.06.2021

Assessment of the influence of intensity of optical signal on the probability of erroneous data registration in a single-photon communication channel

Alexander M. Timofeev

*Belarusian State University of Informatics and Radioelectronics,
st. Brovki, 6, Minsk, 220013, Belarus*
✉E-mail: tamvks@mail.ru

Abstract. When measuring low-power optical signals in single-photon communication channels, the receiving modules must ensure the minimum probability of erroneous data recording. In this regard, it is advisable to use photon counters. They are highly sensitive, but are characterized by data recording errors. The aim of the work was to determine the effect of intensity of optical signal on the probability of erroneous data recording in a single-photon communication channel with a receiving module based on a photon counter.

Based on the results of experimental studies, the dependences of the probability of erroneous registration of binary symbols «0» P_{err0} on the intensity of the optical signal J_0 , used to transmit these symbols for various supply voltages of the avalanche photodetector U , were obtained. It was found that with an increase of intensity of the optical signal, the dependences $P_{err0}(J_0)$ first decrease, reaching their lowest value, and then increase. The performed assessment showed that the lowest probability of erroneous registration the binary symbols «0» for the investigated communication channel is equal to $17,56 \cdot 10^{-2}$ and corresponds to $J_0 = 52,41 \cdot 10^{-2}$ rel. units and $U = 52,54$ V.

Keywords: communication channel, single photon information transfer, probability of erroneous data registration, photon counter, dead time

For citation. Timofeev A. M. Assessment of the influence of intensity of optical signal on the probability of erroneous data registration in a single-photon communication channel. *Informatics*, 2021, vol. 18, no. 2, pp. 72–82 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-72-82>

Conflict of interest. The author declare of no conflict of interest.

Введение. В настоящее время достаточно интенсивное развитие получают волоконно-оптические системы связи на базе однофотонных каналов связи [1]. Это обусловлено тем, что однофотонные каналы связи позволяют обеспечивать абсолютную скрытность и конфиденциальность передаваемых данных, например, при организации на их основе систем квантово-криптографической связи [1, 2].

Отличительным признаком однофотонных каналов связи является то, что для передачи двоичной информации применяют маломощные оптические импульсы, которые могут содержать не более 10 фотонов в расчете на каждый бит (символ) [1–4]. Одной из наиболее технически сложных и вместе с тем достаточно важных задач является регистрация таких импульсов. Для этого целесообразно использовать высокочувствительные приемные модули – счетчики фотонов, построенные на базе лавинных фотоприемников. Однако счетчики фотонов ввиду неидеальности своих характеристик могут приводить к ошибкам при регистрации данных в однофотонном канале связи. Объясняется данный факт, в частности, тем, что счетчики фотонов характеризуются ненулевым мертвым временем [1–5]. Мертвое время счетчика фотонов – это время, в течение которого счетчик фотонов нечувствителен к падающему на него оптическому излучению [2].

Известные методы оценки показателей надежности [6, 7], учитывающие ошибки при передаче информации, неприменимы для однофотонных каналов связи. В работах [8, 9] получены выражения для расчета вероятности ошибочной регистрации информации однофотонных каналов связи, которые могут быть использованы для оценки влияния интенсивности регистрируемого сигнала на вероятность ошибочной регистрации информации. Однако для достижения наименьших потерь информации в однофотонном канале связи, содержащем в качестве приемника счетчик фотонов, весьма важно подбирать не только интенсивность регистрируемого оп-

тического излучения, но и напряжение питания фотоприемника. При этом в литературных источниках отсутствует экспериментальная оценка влияния интенсивности регистрируемого оптического излучения при передаче информации в канале однофотонной связи, содержащем в качестве приемного модуля счетчик фотонов на базе лавинного фотоприемника, на вероятность ошибочной регистрации данных с учетом напряжения питания фотоприемника. Целью настоящей работы является выполнение такой экспериментальной оценки.

Объектом исследования выбран асинхронный двоичный несимметричный однородный однофотонный канал связи без памяти и со стиранием, содержащий в качестве приемного модуля счетчик фотонов на базе лавинного фотоприемника ФД-115Л с мертвым временем продлевающегося типа. Выбор в качестве объекта исследования такого канала связи обусловлен тем, что его использование не требует наличия дополнительных линий связи для передачи и приема синхрои импульсов [4]. Мертвым временем продлевающегося типа характеризуются счетчики фотонов на базе лавинных фотоприемников, включенных по схеме пассивного гашения лавины [2].

Предметом исследования является установление зависимости вероятности ошибочной регистрации двоичных символов «0» от интенсивности оптического сигнала, используемой для передачи этих символов.

Выражение для расчета вероятности ошибочной регистрации двоичных символов «0». Вначале определим математическую модель асинхронного однофотонного канала связи. Пусть передача информации осуществляется по однофотонному каналу связи посредством двоичных символов «0» и «1». Обозначим вероятности появления символов «0» на входе и выходе канала связи как $P_s(0)$ и $P'_s(0)$, а символов «1» – соответственно как $P_s(1)$ и $P'_s(1)$. Вероятность того, что при передаче двоичного символа («0» или «1») на выходе счетчика фотонов не будет зарегистрировано ни символа «0», ни символа «1», обозначим как $P'_s(-)$. Следовательно, алфавит кодовых слов на входе рассматриваемого канала связи не совпадает с алфавитом кодовых слов на его выходе, вероятность приема двоичного символа («0» или «1») либо отсутствия символа на выходе канала связи не зависит ни от того, какой символ был на входе канала, ни от ранее принятых символов. При этом при передаче двоичного символа («0» или «1») на выходе канала может быть не зарегистрировано ни символа «0», ни символа «1». Такой канал связи является дискретным двоичным несимметричным однородным без памяти и со стиранием [10, 11]. Всеми потерями информации, за исключением потерь в счетчике фотонов, пренебрегаем. Вероятность ошибочной регистрации двоичных символов «0» для рассматриваемого канала связи определяется выражением [11]

$$P_{\text{ош}0} = 1 + \sum_{N=0}^{N_1-1} P_{st0}(N) - \sum_{N=0}^{N_2} P_{st0}(N) = P(-/0) + P(1/0) = 1 - P(0/0), \quad (1)$$

где $P_{st0}(N)$ – статистическое распределение смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов при регистрации двоичных символов «0»; $P(0/0)$ и $P(1/0)$ – вероятности регистрации на выходе канала связи символов «0» и «1» при наличии на входе канала связи символа «0»; $P(-/0)$ – вероятность того, что при наличии на входе канала связи символа «0» на его выходе не будет зарегистрирован ни символ «0», ни символ «1»; N_1 и N_2 – нижний и верхний пороговые уровни регистрации соответственно.

Нижний и верхний пороговые уровни регистрации – это наименьшее и наибольшее число зарегистрированных на выходе счетчика фотонов импульсов, при котором делается вывод, что передан символ «0». При превышении зарегистрированных импульсов числа N_2 делается вывод, что передан символ «1», а при регистрации импульсов в количестве, меньшем чем N_1 , принимается решение, что символ отсутствует [11].

Темновые и сигнальные – это импульсы, которые появляются на выходе счетчика фотонов соответственно в отсутствие оптического сигнала и в результате воздействия фотонов регистрируемого излучения [2].

Вероятность ошибочной регистрации двоичных символов «0» $P_{\text{ош}0}$ имеет две составляющие [8]. Первая составляющая определяет вероятность того, что при регистрации двоичных символов «0» счетчиком фотонов будет зарегистрировано импульсов меньше, чем нижний пороговый уровень N_1 , а вторая составляющая – вероятность того, что количество таких зарегистрированных импульсов окажется больше, чем верхний пороговый уровень N_2 . Выражение (1), учитывающее вклад каждой указанной выше составляющей $P_{\text{ош}0}$, позволяет выполнять наиболее детальный анализ причин возникновения ошибок в рассматриваемом канале связи.

Описание установки и условий проведения экспериментальных исследований. Статистические распределения $P_{\text{ст}0}(N)$ могут быть получены с использованием установки, структурная схема которой изображена на рис. 1.

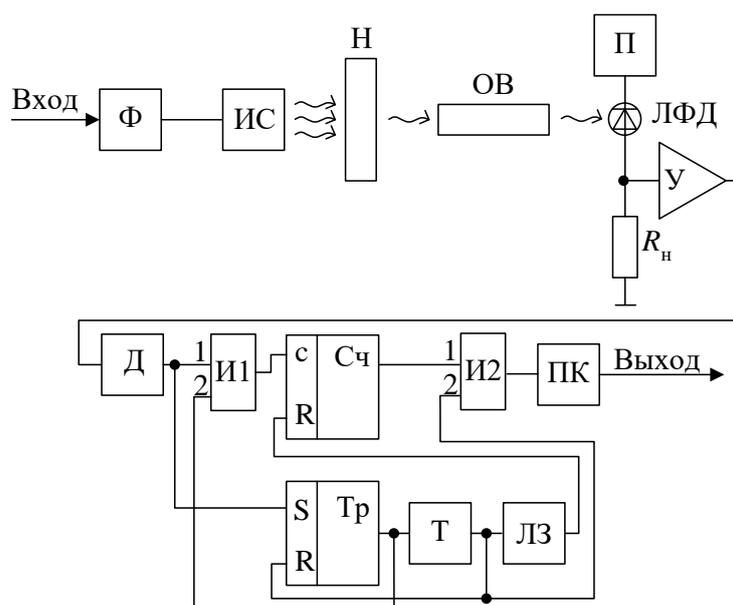


Рис. 1. Структурная схема установки для проведения экспериментальных исследований [11]:
 Ф – формирователь данных, ИС – источник оптического сигнала, Н – нейтральный светофильтр,
 ОВ – оптическое волокно, ЛФД – лавинный фотодиод, П – источник постоянного напряжения,
 R_n – нагрузочный резистор, У – усилитель, Д – амплитудный дискриминатор,
 И1 и И2 – логические элементы И, Сч – счетчик импульсов, Тр – RS-триггер,
 ПК – персональный компьютер, Т – таймер, ЛЗ – линия задержки

Установка содержит передающий модуль, оптическое волокно ОВ и приемный модуль и функционирует следующим образом [11]. Передающий модуль установки работает в режиме асинхронной передачи информации. Он включает в себя формирователь данных Ф, источник оптического сигнала ИС и нейтральный светофильтр Н. Вход передающего модуля является входом установки. Формирователь данных Ф кодирует поступающие на его вход импульсы, которые представляют собой двоичную последовательность, таким образом, что символам «0» и «1» соответствуют прямоугольные импульсы длительностью Δt и напряжением U_1 и U_2 соответственно ($U_1 < U_2$). При этом между каждой парой символов находится так называемый защитный временной интервал $t_3 = \tau_b - \Delta t$, где τ_b – средняя длительность передачи одного бита (символа). В течение времени t_3 сигнал на выходе формирователя данных Ф отсутствует.

Затем с выхода формирователя данных Ф импульсы поступают на вход источника оптического сигнала с пуассоновской статистикой фотонов ИС, в качестве которого использовался светодиод АЛ307А с длиной волны 665 нм и номинальной силой света 0,15 мкд.

На выходе источника ИС формируется оптический сигнал длительностью Δt и интенсивностью J_0 под воздействием на его вход напряжения U_0 . При появлении на входе источника ИС импульса с напряжением U_1 на его выходе генерируется оптический сигнал длительностью Δt и интенсивностью J_1 ($J_0 < J_1$).

Следует отметить, что на вход установки, показанной на рис. 1, подавались как символы «0», так и символы «1» в связи с изложенными далее фактами. Интенсивность оптического излучения, используемая для передачи заданного двоичного символа, влияет на вероятность ошибочной регистрации не только одноименного символа, но и другого двоичного символа [8, 9]. Так, интенсивность J_0 влияет на вероятность ошибочной регистрации не только символа «0», но и символа «1», а интенсивность J_1 влияет на вероятность ошибочной регистрации не только символа «1», но и символа «0». Учитывая это, а также тот факт, что пропускная способность рассматриваемого канала связи максимальна при равновероятном появлении двоичных символов «0» и «1» на входе канала связи [10], количество двоичных символов «0», подаваемых на вход установки (см. рис. 1), равняется количеству двоичных символов «1». Таким образом, выполняется условие

$$P_s(0) = P_s(1) = 0,5. \quad (2)$$

Затем оптическое излучение с выхода источника оптического сигнала ИС поступает через нейтральный светофильтр Н, который обеспечивает ослабление мощности оптического сигнала, на выход передающего модуля и далее в оптическое волокно ОВ.

Из оптического волокна ОВ излучение подается на приемный модуль установки, в состав которого входят источник постоянного напряжения П, лавинный фотодиод ЛФД, нагрузочный резистор R_n , усилитель У, амплитудный дискриминатор Д, логические элементы И1 и И2, счетчик импульсов Сч, RS-триггер Тр, таймер Т, линия задержки ЛЗ и персональный компьютер ПК. Приемный модуль установки работает в режиме асинхронного приема информации следующим образом. Из оптического волокна ОВ излучение подается на лавинный фотодиод ЛФД, в качестве которого использован лавинный фотоприемник ФД-115Л. Реализация режима счета фотонов и включение лавинного фотодиода ЛФД по схеме пассивного гашения лавинного пробоя выполнялись по методике, описанной в работе [12]. На лавинный фотодиод ЛФД от источника постоянного напряжения питания П подается напряжение обратного смещения $U_{\text{пт}}$. Величина этого напряжения выбиралась близкой или превышающей напряжение лавинного пробоя p - n -перехода лавинного фотодиода $U_{\text{пр}}$, что контролировалось вольтметром. При такой величине напряжения обратного смещения через лавинный фотодиод ЛФД протекают импульсы тока, обусловленные возникновением пробоя по локальным каналам p - n -перехода – микроплазм. Появление микроплазм может быть вызвано либо носителем заряда, образованным под действием фотона оптического излучения, либо термогенерацией. Количество данных импульсов прямо пропорционально энергии оптического импульса, которая равна произведению мощности оптического сигнала на время Δt . Количество однофотонных импульсов N также прямо пропорционально числу фотонов в импульсе.

Таким образом, на выходе лавинного фотодиода ЛФД за время передачи символа Δt формируется различное количество электрических импульсов при наличии на его входе символов «0» и «1», так как они передаются импульсами различной мощности. При этом количество электрических импульсов прямо пропорционально мощности оптического излучения, а в отсутствие потерь передаваемой информации максимальное число однофотонных импульсов лавинного фотодиода ЛФД, сформированных за время Δt при передаче символа «0», будет меньше, чем минимальное число импульсов, сформированных при передаче символа «1».

При возникновении пробоя протекающий через лавинный фотодиод ЛФД ток возрастает, что приводит к увеличению падения напряжения на нагрузочном резисторе R_n и к уменьшению напряжения на лавинном фотодиоде ЛФД. В результате этого пробой в микроплазменном канале прекращается (гасится), а на нагрузочном резисторе R_n формируется импульс напряжения.

Импульсы, выделенные на сопротивлении нагрузки R_n , поступают на вход усилителя У, обеспечивающего их усиление. Затем с выхода усилителя У импульсы подаются на вход амплитудного дискриминатора Д, порог амплитудной селекции которого установлен непосредственно над уровнем собственных шумов усилителя. В процессе проводимых измерений порог амплитудной селекции оставался неизменным.

При помощи амплитудного дискриминатора Д выполняется амплитудная селекция усиленных импульсов напряжения на фоне шумов усилителя У, а также их стандартизация по амплитуде и длительности. С выхода амплитудного дискриминатора Д импульсы поступают одновременно на S-вход RS-триггера Тр и на первый вход логического элемента И1. Первый импульс из последовательности импульсов переводит RS-триггер Тр в единичное состояние. Выход RS-триггера Тр соединен со вторым входом логического элемента И1 и управляющим входом таймера Т. Появление напряжения на втором входе логического элемента И1, соответствующего логической единице, позволяет импульсам с выхода амплитудного дискриминатора Д поступать на вход счетчика импульсов Сч, который подсчитывает число импульсов последовательности, поступающих на его вход.

В момент времени переключения RS-триггера Тр из нулевого состояния в единичное запускается таймер Т. Через промежуток времени Δt на выходе таймера Т формируется импульс, который поступает на R-вход RS-триггера Тр, переводя его в нулевое состояние. На выходе RS-триггера Тр формируется сигнал, соответствующий логическому нулю, который подается на второй вход логического элемента И1. Это останавливает поступление импульсов на вход счетчика импульсов Сч, а следовательно, и подсчет импульсов счетчиком Сч. Импульс с выхода таймера Т поступает также на вход линии задержки ЛЗ и на второй вход логического элемента И2. Выход счетчика Сч соединен с первым входом логического элемента И2. При наличии логических единиц на первом и втором входах логического элемента И2 количество импульсов, сосчитанных счетчиком Сч при передаче двоичного символа, передается в персональный компьютер ПК.

Сброс счетчика импульсов Сч происходит через некоторый интервал времени после прихода импульса от таймера Т. Длительность этого интервала задается линией задержки ЛЗ.

Следует отметить, что счетчик фотонов характеризуется мертвым временем продлевающегося типа [2], поскольку он построен на базе лавинного фотоприемника с включением по схеме пассивного гашения лавины. В этом случае длительность мертвого времени продлевающегося типа зависит от интенсивности оптического излучения, поэтому для оценки такого времени используют его среднее значение. Учитывая то, что регистрирующая аппаратура, входящая в состав счетчика фотонов, как правило, имеет мертвое время, намного меньшее, чем мертвое время лавинного фотоприемника [2], длительностью мертвого времени регистрирующей аппаратуры пренебрегаем и при дальнейших рассуждениях рассматривать ее не будем.

Персональный компьютер ПК отдельно формирует массивы данных числа импульсов при передаче символов «0» и «1», поступивших в течение интервала времени Δt . На основании полученных таким образом массивов данных рассчитываются статистические распределения смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов при регистрации двоичных символов «0» $P_{st0}(N)$, а также статистические распределения смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов при регистрации двоичных символов «1» $P_{st1}(N)$.

Затем персональный компьютер ПК выполняет сравнение количества импульсов N с двумя пороговыми уровнями N_1 и N_2 и осуществляет проверку выполнения условий [11]

$$N_1 \leq N \leq N_2, \quad (3)$$

$$N > N_2. \quad (4)$$

Если выполняется условие (3), то принимается решение, что зарегистрирован двоичный символ «0». Если выполняется условие (4), то принимается решение, что зарегистрирован двоичный символ «1». При этом ошибочная регистрация данных фиксируется в следующих случаях:

- если условия (3) или (4) не выполняются;
- если условие (3) выполняется, но при этом на входе канала связи двоичный символ «0» отсутствует;
- если условие (4) выполняется, но при этом на входе канала связи двоичный символ «1» отсутствует.

Методика достижения наименьших потерь информации. Интенсивности оптических излучений, используемых для передачи двоичных символов («0» или «1»), и пороговые уровни зарегистрированных импульсов N_1 и N_2 , обеспечивающие наименьшую вероятность ошибочной регистрации двоичных символов «0», подбирались с помощью методики, описанной в работе [11]. В качестве критерия для реализации этой методики использован параметр

$$K_p = \frac{1 - \sum_{N=N_1}^{N_2} P_{st0}(N) + \sum_{N=0}^{N_2} P_{st1}(N)}{1 + \sum_{N=N_1}^{N_2} P_{st0}(N) - \sum_{N=0}^{N_2} P_{st1}(N)}. \quad (5)$$

Методика включает пять этапов:

1. Формирование массивов данных статистических распределений смеси числа темновых и сигнальных импульсов $P_{st}(N)$ и интенсивностей оптических сигналов J . В исследуемом диапазоне интенсивностей оптических сигналов с использованием установки, показанной на рис. 1, сначала измеряют статистические распределения смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов, а затем на их основе формируют массивы данных:

$$P_{st}(N) = \{P_{st}(N)^{(1)}, P_{st}(N)^{(2)}, P_{st}(N)^{(i)}, \dots, P_{st}(N)^{(m)}\}, \quad (6)$$

$$J = \{J^{(1)}, J^{(2)}, J^{(i)}, \dots, J^{(m)}\}, \quad (7)$$

где $P_{st}(N)^{(i)}$ – i -е статистическое распределение смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов, соответствующее i -й интенсивности оптического сигнала $J^{(i)}$, $i = 2 \div m$ (m определяется объемом накопленных данных числа импульсов).

По мере реализации методики итерационно определяют величины i_0 , i_1 , N_1 , N_2 и K_p , а также i_0' , i_1' , N_1' , N_2' и K_p' следующим образом. Вначале устанавливают $i_0 = 1$, $i_1 = 2$, $N_1 = 1$, $K_p' = 10^3$, затем переходят к этапу 2.

2. Определение интенсивности оптического сигнала для передачи двоичных символов «0». Если $i_0 < m$, то устанавливают $P_{st0}(N) = P_{st}(N)^{(i_0)}$ и переходят к этапу 3; иначе завершают выполнение методики.

3. Определение интенсивности оптического сигнала для передачи двоичных символов «1». Если $i_1 \leq m$, то устанавливают $P_{st1}(N) = P_{st}(N)^{(i_1)}$ и переходят к этапу 4; иначе последовательно устанавливают $K_p' = 10^3$, увеличивают на единицу i_0 , устанавливают $i_1 = i_0 + 1$ и переходят к этапу 2.

4. Определение нижнего порогового уровня зарегистрированных на выходе счетчика фотонов импульсов N_1 , при котором делается вывод, что передан символ «0». Если $N_1 < 10$, то устанавливают $N_2 = N_1 + 1$ и переходят к этапу 5; иначе последовательно устанавливают $N_1 = 1$, увеличивают на единицу i_1 и переходят к этапу 3.

5. Определение верхнего порогового уровня зарегистрированных на выходе счетчика фотонов импульсов N_2 , при превышении которого делается вывод, что передан символ «1». Если $N_2 > 10$, то увеличивают на единицу значение N_1 и переходят к этапу 4; иначе последовательно выполняют следующие действия:

– по формуле (5) вычисляют K_p . Если $K_p < K_p'$, то последовательно устанавливают K_p' равным K_p и формируют массив данных, содержащий $i_0' = i_0$, $i_1' = i_1$, $N_1' = N_1$, $N_2' = N_2$, а также величины J_0 и J_1 , соответствующие статистическим распределениям $P_{st}(N)^{(i_0)}$ и $P_{st}(N)^{(i_1)}$;

– увеличивают на единицу верхний пороговый уровень зарегистрированных на выходе счетчика фотонов импульсов N_2 ;

– переходят к началу этапа 5.

По окончании реализации методики из массива сформированных данных i_0' , i_1' , N_1' , N_2' , J_0 и J_1 выбираются величины N_1' и N_2' , которые используются соответственно как нижний

и верхний пороговые уровни. Значения i_0' и i_1' определяют статистические распределения смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов при регистрации двоичных символов «0» $P_{st0}(N)$ и двоичных символов «1» $P_{st1}(N)$, а также интенсивности регистрируемых оптических излучений при передаче этих символов J_0 и J_1 , соответствующие $P_{st}(N)^{(i_0)}$, $J^{(i_0)}$ и $P_{st}(N)^{(i_1)}$, $J^{(i_1)}$, которые выбираются из массивов данных (6) и (7). Рассчитать вероятность ошибочной регистрации двоичных символов «0» можно путем подстановки в формулу (1) статистических распределений $P_{st0}(N)$ при выбранных пороговых уровнях зарегистрированных импульсов $N_1 = N_1'$ и $N_2 = N_2'$.

Экспериментальные результаты и их обсуждение. Выполнены исследования зависимости вероятности ошибочной регистрации двоичных символов «0» от интенсивности оптического сигнала, используемой для передачи этих символов (рис. 2).

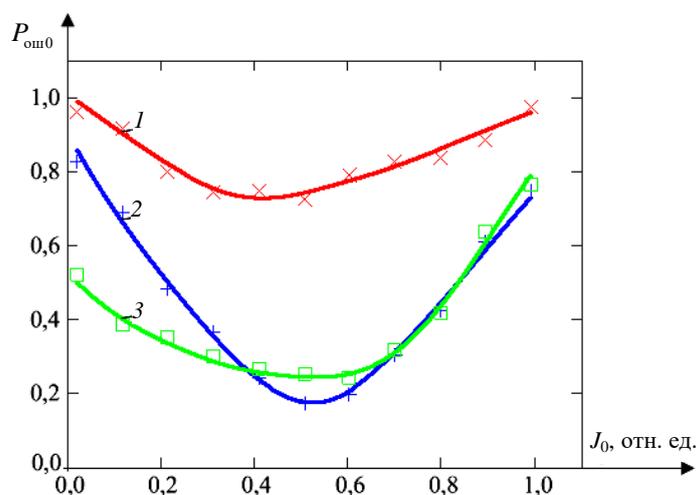


Рис. 2. Зависимость вероятности ошибочной регистрации двоичных символов «0» от интенсивности оптического сигнала, используемой для передачи этих символов
 Напряжение питания лавинного фотоприемника $U_{пит}$: 1 – 52,48 В, 2 – 52,54 В, 3 – 52,65 В

Нижний и верхний пороговые уровни зарегистрированных на выходе счетчика фотонов импульсов, а также статистические распределения смеси числа темновых и сигнальных импульсов на выходе счетчика фотонов при регистрации двоичных символов «0» $P_{st0}(N)$ определялись по методике, описанной выше. Расчет вероятности ошибочной регистрации двоичных символов «0» $P_{ош0}$ выполнен путем подстановки этих величин в формулу (1). Измерения проводились при различных значениях напряжения питания лавинного фотоприемника с использованием установки, структурная схема которой показана на рис. 1. Отметим, что для реализации режима счета фотонов напряжения питания лавинного фотоприемника выбирались вблизи напряжения его лавинного пробоя, которое определялось по методике [12] и составляло 52,71 В. Интенсивности оптических излучений J_0 , используемых для передачи двоичных символов «0», нормированы на максимальное значение из исследуемого диапазона. Все исследования проводились при температуре 293 К.

На рис. 2 видно, что каждая из зависимостей $P_{ош0}(J_0)$ имеет схожие тенденции изменения для всех исследуемых напряжений питания лавинного фотоприемника. С увеличением интенсивности оптического сигнала J_0 зависимости $P_{ош0}(J_0)$ сначала спадают, достигая своего наименьшего значения, а затем растут. Это весьма хорошо согласуется с результатами математического моделирования рассматриваемого канала связи, полученными в работе [8] при исследовании зависимостей $P_{ош0}(n_{s0})$, где n_{s0} – средняя скорость счета сигнальных импульсов на выходе счетчика фотонов при передаче символов «0».

При небольших значениях интенсивностей оптического сигнала J_0 средняя скорость счета сигнальных импульсов на выходе счетчика фотонов при передаче символов «0» n_{s0} невелика. Поэтому вероятность регистрации на выходе счетчика фотонов импульсов в количестве, большем нижнего порогового уровня регистрации N_1 , также мала. Таким образом, первая составляющая вероятности $P_{\text{ош}0}$, равная вероятности того, что на выходе счетчика фотонов не будет зарегистрировано ни символа «0», ни символа «1», в то время как на входе канала связи был сформирован символ «0» $P(-/0)$, достаточно большая. Следовательно, вероятность $P_{\text{ош}0}$ весьма высока (см. рис. 2).

С увеличением интенсивности оптического сигнала J_0 средняя скорость счета сигнальных импульсов на выходе счетчика фотонов при передаче символов «0» растет, что приводит к увеличению вероятности регистрации на выходе счетчика фотонов импульсов в диапазоне от нижнего порогового уровня регистрации N_1 до верхнего порогового уровня регистрации N_2 . При этом первая составляющая вероятности $P_{\text{ош}0}$, определяемая вероятностью $P(-/0)$, уменьшается, а вероятность $P(0/0)$ растет, поэтому зависимости $P_{\text{ош}0}(J_0)$ спадают вплоть до своего наименьшего значения.

Дальнейший рост интенсивности J_0 приводит к увеличению вероятности регистрации на выходе счетчика фотонов импульсов в количестве, превышающем верхний пороговый уровень регистрации N_2 . В результате растет вероятность регистрации на выходе канала связи символа «1» при наличии на его входе символа «0», т. е. имеет место увеличение второй составляющей вероятности $P_{\text{ош}0} - P(1/0)$. Следовательно, зависимости $P_{\text{ош}0}(J_0)$ после достижения наименьших значений начинают расти (см. рис. 2).

Отметим, что при прочих равных параметрах приема вероятность $P_{\text{ош}0}$, соответствующая наименьшей из исследуемого диапазона интенсивности оптического сигнала $J_0 = 1,79 \cdot 10^{-2}$ отн. ед., уменьшается с увеличением напряжения питания лавинного фотоприемника и равна $98,96 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,48$ В; $85,79 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,54$ В и $49,95 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,65$ В. Объясняется это ростом средней скорости счета темновых импульсов на выходе счетчика фотонов n_t при увеличении напряжения питания лавинного фотоприемника. Такой рост способствует смещению максимума статистического распределения смеси числа темновых и сигнальных импульсов $P_{st0}(N)$ при передаче символов «0» в сторону больших значений N при одинаковой интенсивности J_0 , но более высоком напряжении питания лавинного фотоприемника. Поскольку внешний вид этих распределений схож с теоретическими распределениями [9], в настоящей работе распределения $P_{st0}(N)$ не проиллюстрированы.

На рис. 2 также видно, что с увеличением напряжения питания лавинного фотоприемника минимальное значение зависимости $P_{\text{ош}0}(J_0)$ сначала снижается, а затем растет. Это обусловлено тем, что при увеличении напряжения питания лавинного фотоприемника мертвое время счетчика фотонов уменьшается, переходя в насыщение, однако квантовая эффективность регистрации сначала растет и после достижения максимальной величины уменьшается [2].

Уменьшение мертвого времени счетчика фотонов и рост его квантовой эффективности регистрации способствуют снижению количества ошибок. Следовательно, в диапазоне напряжений питания, на котором имеет место такой характер изменения мертвого времени и квантовой эффективности регистрации счетчика фотонов, минимум зависимости $P_{\text{ош}0}(J_0)$ уменьшается с ростом напряжений питания фотоприемника. Так, например, наименьшее значение зависимости $P_{\text{ош}0}(J_0)$ равно $72,82 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,48$ В и $17,56 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,54$ В.

При дальнейшем росте напряжения питания фотоприемника мертвое время счетчика фотонов практически не изменяется, однако его квантовая эффективность регистрации уменьшается. Это увеличивает количество ошибок, поэтому в таком диапазоне напряжений питания минимум зависимости $P_{\text{ош}0}(J_0)$ начинает расти (см. рис. 2). Например, наименьшее значение зависимости $P_{\text{ош}0}(J_0)$ равно $17,56 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,54$ В и $24,42 \cdot 10^{-2}$ для $U_{\text{пит}} = 52,65$ В.

Также важно отметить, что с ростом $U_{\text{пит}}$ увеличивается n_t , поэтому чем выше $U_{\text{пит}}$, тем большую скорость счета n_{s0} необходимо обеспечить для достижения наименьшей величини

ны $P_{\text{ош}0}$. Это приводит к тому, что минимальное значение зависимости $P_{\text{ош}0}(J_0)$ наблюдается при более высоких значениях J_0 с ростом напряжения питания лавинного фотоприемника: при $J_0 = 41,56 \cdot 10^{-2}$ отн. ед. для $U_{\text{пит}} = 52,48$ В; при $J_0 = 52,41 \cdot 10^{-2}$ отн. ед. для $U_{\text{пит}} = 52,54$ В; при $J_0 = 52,63 \cdot 10^{-2}$ отн. ед. для $U_{\text{пит}} = 52,65$ В.

Заключение. Применительно к асинхронному двоичному несимметричному однородному однофотонному каналу связи без памяти и со стиранием, в котором в качестве приемного модуля используется счетчик фотонов с мертвым временем продлевающегося типа, выполнена оценка влияния интенсивности оптического сигнала J_0 , используемого для передачи двоичных символов «0», на вероятность ошибочной регистрации этих символов $P_{\text{ош}0}$.

Выполненные экспериментальные исследования показали, что для достижения наименьших потерь информации при передаче двоичных символов «0» важно подбирать не только интенсивность используемого оптического излучения J_0 , но и напряжение питания лавинного фотоприемника $U_{\text{пит}}$.

Определено, что с увеличением интенсивности оптического сигнала J_0 зависимости $P_{\text{ош}0}(J_0)$ сначала спадают, достигая своего наименьшего значения, а затем растут. Аналогичный характер изменения имеет и минимум зависимости $P_{\text{ош}0}(J_0)$, что проявляется при повышении напряжения питания лавинного фотоприемника.

Установлено, что спад зависимости $P_{\text{ош}0}(J_0)$ в основном обусловлен снижением вероятности того, что на выходе канала связи не будет зарегистрирован ни символ «0», ни символ «1», в то время как на входе канала связи был сформирован символ «0» $P(-/0)$. Рост же этой зависимости происходит преимущественно за счет повышения вероятности того, что при передаче символа «0» на выходе канала связи будет зарегистрирован символ «1» $P(1/0)$.

Получено, что наименьшая вероятность ошибочной регистрации двоичных символов «0» для рассматриваемого канала связи равна $17,56 \cdot 10^{-2}$ и соответствует $J_0 = 52,41 \cdot 10^{-2}$ отн. ед. и $U_{\text{пит}} = 52,54$ В, при которых мертвое время счетчика фотонов минимально, а его квантовая эффективность регистрации максимальна.

Автору настоящей работы представляются весьма актуальными исследования, направленные на обоснование выбора лавинного фотоприемника, используемого при построении счетчика фотонов. Такие фотоприемники могут отличаться как по структуре полупроводниковых областей, так и по площади фоточувствительной поверхности. В этой связи в ходе дальнейших комплексных исследований планируется определить, как эти параметры влияют на вероятность ошибочной регистрации двоичных данных для асинхронного двоичного несимметричного однородного однофотонного канала связи без памяти и со стиранием.

Список используемых источников

1. Килин, С. Я. Квантовая криптография: идеи и практика / С. Я. Килин ; под ред. С. Я. Килина, Д. Б. Хорошко, А. П. Низовцева. – Минск : Беларус. наука, 2007. – 391 с.
2. Гулаков, И. Р. Фотоприемники квантовых систем : монография / И. Р. Гулаков, А. О. Зеневич. – Минск : Высший гос. колледж связи, 2012. – 276 с.
3. Photon counting for quantum key distribution with Peltier cooled InGaAs/InP APDs / D. Stucki [et al.] // J. of Modern Optics. – 2007. – Vol. 48, no. 13. – P. 1967–1981.
4. Тимофеев, А. М. Методика повышения достоверности принятых данных счетчика фотонов на основе анализа скорости счета импульсов при передаче двоичных символов «0» / А. М. Тимофеев // Приборы и методы измерений. – 2019. – Т. 10, № 1. – С. 80–89.
5. Single-photon detectors combining high efficiency, high detection rates, and ultra-high timing resolution / I. E. Zadeh [et al.] // APL Photonics. – 2017. – Vol. 2. – P. 111301-1–111301-7.
6. Дмитриев, С. А. Волоконно-оптическая техника: современное состояние и перспективы / С. А. Дмитриев, Н. Н. Слепов. – 2-е изд., перераб. и доп. – М. : ООО «Волоконно-оптическая техника», 2005. – 576 с.
7. Щеглов, А. Ю. Анализ и проектирование защиты информационных систем. Контроль доступа к компьютерным ресурсам: методы, модели, технические решения / А. Ю. Щеглов. – СПб. : Проф. литература, 2017. – 416 с.

8. Тимофеев, А. М. Оценка влияния продлевающегося мертвого времени счетчика фотонов на вероятность ошибочной регистрации данных квантово-криптографических каналов связи / А. М. Тимофеев // Вестник связи. – 2018. – № 1(147). – С. 56–62.
9. Тимофеев, А. М. Влияние времени однофотонной передачи информации на вероятность ошибочной регистрации данных асинхронных квантово-криптографических каналов связи / А. М. Тимофеев // Вестник ТГТУ. – 2019. – Т. 25, № 1. – С. 36–46.
10. Ключев, Л. Л. Теория электрической связи / Л. Л. Ключев. – Минск : Техноперспектива, 2008. – 423 с.
11. Тимофеев, А. М. Методика снижения потерь информации в асинхронном двоичном однофотонном канале связи с приемником на основе счетчика фотонов / А. М. Тимофеев // Приборы и методы измерений. – 2020. – Т. 11, № 1. – С. 70–81.
12. Гулаков, И. Р. Метод счета фотонов в оптико-физических измерениях / И. Р. Гулаков, С. В. Холондырев. – Минск : Университетское, 1989. – 256 с.

References

1. Kilin S. Ya. Kvantovaya kriptografiya: idei i praktika. *Quantum Cryptography: Ideas and Practices*. In Kilin S. Ya., Horoshko D. B., Nizovcev A. P. (eds.). Minsk, Belorusskaja nauka, 2007, 391 p. (In Russ.).
2. Gulakov I. R., Zenevich A. O. Fotopriemniki kvantovyih sistem. *Photodetectors of Quantum Systems*. Minsk, Vysshij gosudarstvennyj kolledzh svyazi, 2012, 276 p. (In Russ.).
3. Stucki D., Ribordy G., Stefanov A., Zbinden H., ..., Wall T. Photon counting for quantum key distribution with Peltier cooled InGaAs/InP APDs. *Journal of Modern Optics*, 2007, vol. 48, no. 13, pp. 1967–1981.
4. Timofeev A. M. *Methods of increasing the reliability of the received data of the photon counter based on the analysis of the pulse counting rate during the transmission of binary symbols "0"*. Pribory i metody izmereniy [*Devices and Methods of Measurements*], 2019, vol. 10, no. 1, pp. 80–89 (In Russ.).
5. Zadeh I. E., Los J. W., Gourgues R. B., Steinmetz V., Bulgarini G., ..., Dorenbos S. N. Single-photon detectors combining high efficiency, high detection rates, and ultra-high timing resolution. *APL Photonics*, 2017, vol. 2, pp. 111301-1–111301-7.
6. Dmitriev S. A. Volokonno-opticheskaya tehnika: sovremennoe sostoyanie i perspektivy. *Fiber-Optic Technology: Current Status and Prospects*. Moscow, ООО "Volokonno-opticheskaja tehnika", 2005, 576 p. (In Russ.).
7. Scheglov A. Yu. Analiz i proektirovanie zaschityi informatsionnyih sistem. Kontrol dostupa k kompyuternym resursam: metodyi, modeli, tehnicheckie resheniya. *Analysis and Design of Information Systems Protection. Control of Access to Computer Resources: Methods, Models, Technical Solutions*. Saint Petersburg, Professional'naja literatura, 2017, 416 p. (In Russ.).
8. Timofeev A. M. *Estimation of the photons counter lasting dead time influence on the probability of erroneous data registration of quantum-cryptographic communication channels*. Vestnik svyazi [*Communication bulletin*], 2018, № 1(147), pp. 56–62 (In Russ.).
9. Timofeev A. M. *The effect of single photon transmission time on the probability of erroneous registration of asynchronous data of quantum cryptographic communication channels*. Vestnik Tambovskogo gosudarstvennogo tehnicheckogo universiteta [*Transactions of the Tambov State Technical University*], 2019, vol. 25, no. 1, pp. 36–46 (In Russ.).
10. Klyuev L. L. Teoriya elektricheskoy svyazi. *The Theory of Electrical Communication*. Minsk, Tehnoperspektiva, 2008, 423 p. (In Russ.).
11. Timofeev A. M. *Method of achieving the least loss of information in an asynchronous binary single-photon communication channel with a receiver based on a photon counter*. Pribory i metody izmereniy [*Devices and Methods of Measurements*], 2020, vol. 11, no. 1, pp. 70–81 (In Russ.).
12. Gulakov I. R., Kholondyrev S. V. Metod scheta fotonov v optiko-fizicheskikh izmereniyah. *Photon Counting Method in Optical-Physical Measurements*. Minsk, Universitetskoe, 1989, 256 p. (In Russ.).

Информация об авторе

Тимофеев Александр Михайлович, кандидат технических наук, доцент, доцент кафедры защиты информации, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: tamvks@mail.ru

Information about the author

Alexander M. Timofeev, Cand. Sci. (Eng.), Associate Professor, Associate Professor of the Department of Information Security, Belarusian State University of Informatics and Radioelectronics.
E-mail: tamvks@mail.ru

ОБРАБОТКА СИГНАЛОВ, ИЗОБРАЖЕНИЙ, РЕЧИ, ТЕКСТА И РАСПОЗНАВАНИЕ ОБРАЗОВ

SIGNAL, IMAGE, SPEECH, TEXT PROCESSING AND PATTERN RECOGNITION



УДК 004.932
<https://doi.org/10.37661/1816-0301-2021-18-2-83-97>

Оригинальная статья
Original Paper

Автоматическое обнаружение и отслеживание движущихся объектов, наблюдаемых видеокамерой беспилотного летательного аппарата

Р. С. Жук

*Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
E-mail: ramanzhuck@gmail.com*

Аннотация. Предлагается алгоритм автоматического обнаружения и отслеживания движущихся объектов, предназначенный для использования на борту беспилотного летательного аппарата. Разработанный алгоритм основан на отслеживании на протяжении определенного времени выбранных точек изображения. Отслеживаемые точки выбираются из областей на текущем кадре, в которых интенсивность пикселей отличается от интенсивностей тех же пикселей в предыдущих кадрах, совмещенных с текущим кадром при помощи проективного преобразования. Если на нескольких соседних кадрах не фиксируется смещение отслеживаемых точек, они удаляются и на их место добавляются новые точки из областей, предположительно принадлежащих движущимся объектам на текущем кадре. На каждом кадре близкие по расположению и форме траекторий движения точки объединяются в группы, которые соответствуют движущимся объектам. Отслеживание объектов осуществляется путем сопоставления групп движущихся точек соседних кадров. Группы движущихся точек соседних кадров сопоставляются, когда они содержат большое число общих отслеживаемых точек. Алгоритм позволяет одновременно отслеживать более 20 объектов в реальном времени. Индикация объекта как движущегося происходит только в том случае, если за время его сопровождения он сместился на значительное расстояние. Рассматриваемый алгоритм имеет низкий процент ложных обнаружений объектов, хорошо обнаруживает объекты малого размера и надежно сопровождает объекты.

Ключевые слова: обнаружение движения, сопровождение объектов, беспилотные летательные аппараты, движущаяся камера, оптический поток

Для цитирования. Жук, Р. С. Автоматическое обнаружение и отслеживание движущихся объектов, наблюдаемых видеокамерой беспилотного летательного аппарата / Р. С. Жук // Информатика. – 2021. – Т. 18, № 2. – С. 83–97. <https://doi.org/10.37661/1816-0301-2021-18-2-83-97>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 03.06.2021
Подписана в печать | Accepted 07.06.2021
Опубликована | Published 26.06.2021

Automatic detection and tracking the moving objects observed by an unmanned aerial vehicles video camera

Raman S. Zhuk

*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus
E-mail: ramanzhuck@gmail.com*

Abstract. An algorithm of automatic detection and tracking the moving objects for the use in equipment on board of unmanned aerial vehicles is considered. The developed algorithm is based on a tracking specially selected points for a certain period. Tracked points are selected from the areas on the current frame, where the pixel intensity differs from the intensities of the same pixels in previous frames, aligned with the current frame using projective transformation. If the displacement of the tracked points is not fixed on several adjacent frames, they are being deleted, and new points from the areas presumably belonging to moving objects in the current frame are added instead. On each frame the points similar by the location and shape of trajectories of movement are combined into groups that presumably correspond to moving objects. Objects are tracked by comparing the groups of moving points with the points of neighboring frames. Groups of moving points from neighboring frames are matched if they contain a large number of common tracked points. The algorithm allows simultaneous tracking of more than 20 objects in real time. The indication of objects as moving occurs only if during the time of its tracking it has shifted a considerable distance. The algorithm has a low percentage of false detections of moving objects, it detects well small objects and is capable reliably to accompany moving objects.

Keywords: motion detection, objects tracking, unmanned aerial vehicles, moving camera, optical flow

For citation. Zhuk R. S. Automatic detection and tracking the moving objects observed by an unmanned aerial vehicles video camera. *Informatics*, 2021, vol. 18, no. 2, pp. 83–97 (In Russ.). <https://doi.org/10.37661/1816-0301-2021-18-2-83-97>

Conflict of interest. The author declare of no conflict of interest.

Введение. В настоящее время в связи со стремительным развитием и широкой востребованностью систем беспилотных летательных аппаратов (БПЛА), одним из основных применений которых является визуальное наблюдение, актуальны задачи обнаружения и отслеживания объектов, наблюдаемых бортовой видеокамерой. Важным классом отслеживаемых объектов являются движущиеся объекты. В последние годы опубликовано большое число работ, посвященных алгоритмам отслеживания объектов данного класса. Очевидно, что для решения задачи отслеживания движущихся объектов, наблюдаемых видеокамерой, требуется сначала автоматически их обнаружить. Поэтому задача обнаружения движущихся объектов может рассматриваться как часть задачи отслеживания. Следовательно, задача обнаружения и отслеживания движущихся объектов, наблюдаемых бортовой видеокамерой, может быть сформулирована таким образом: имеется видеопоследовательность, получаемая бортовой камерой малого БПЛА; на кадрах этой видеопоследовательности требуется обнаружить все вновь появившиеся движущиеся объекты и отследить все движущиеся объекты, обнаруженные ранее. Решение этой задачи требует разработки новых алгоритмов, учитывающих ее специфику, которая заключается в следующем: задача должна решаться на борту малых БПЛА, оборудованных маломощными вычислителями с ARM-архитектурой (такими, как Raspberry Pi), что делает невозможным использование мощных нейросетевых подходов; малые БПЛА часто оборудуются панхроматическими камерами (без цветовых признаков); камеры перемещаются относительно земной поверхности; объекты, присутствующие на кадрах, часто бывают малоразмерными.

Описание основных подходов и обзор последних результатов обнаружения движущихся объектов представлены в работе [1]. Можно выделить три основных метода решения задачи обнаружения движения: вычитание фона, локальную разность кадров после компенсации смещения камеры (для случая движущейся камеры) и оптический поток. Недостатки этих подхо-

дов не позволяют применять их без существенных модификаций на борту БПЛА. Так, метод вычитания фона требует предварительного долгого наблюдения за одной и той же сценой перед его применением для обнаружения движения в этой сцене. Он хорошо подходит для систем наблюдения с неподвижными (либо закрепленными на неподвижных объектах) камерами. Для случая с подвижной камерой требуется построение панорамного снимка, что представляет собой трудоемкую задачу. Поскольку БПЛА может следовать на довольно высокой скорости по незнакомому маршруту, метод вычитания фона на его борту практически неприменим. Метод компенсации движения основан на локальной разности интенсивностей пикселей текущего кадра и недалеких от него по времени предыдущих кадров, предварительно преобразованных так, чтобы скомпенсировать движение точек фона. Недостатком этого метода является большое число ошибок, связанных с неточностями при компенсации движения точек фона. Чаще всего для решения задачи компенсации движения фона используется проективное преобразование (гомография), которое совмещает точки кадров в предположении, что сцена является плоской. Такой подход хорошо подходит для применения на борту БПЛА, особенно летящего на большой высоте, однако ошибки компенсации движения фона все равно будут присутствовать. Метод, основанный на оптическом потоке, находит смещение всех (либо расположенных на равномерной решетке) точек кадра при помощи одного из алгоритмов вычисления оптического потока, после чего найденное поле смещений анализируется и выделяются области, принадлежащие движущимся объектам. Движущиеся объекты на кадрах видеопоследовательностей, снятых с БПЛА, могут быть небольшого размера, поэтому необходимо искать оптический поток для большого числа точек, что требует слишком значительных вычислительных затрат.

Кроме упомянутых классических методов, в настоящее время в компьютерном зрении широко используются подходы, основанные на применении сверточных нейронных сетей (англ. convolutional neural network, CNN). Алгоритмы, основанные на CNN, показывают наилучшие результаты в задачах обнаружения и сопровождения объектов. Недостатком применения CNN является их высокая вычислительная сложность, требующая использования дорогостоящих мощных вычислителей с графическими процессорами, поддерживающими технологию параллельного программирования CUDA.

Таким образом, из всех рассмотренных методов решения поставленной задачи наиболее приемлемыми (ввиду их высокой производительности) являются подходы, основанные на компенсации движения. Вместе с тем у них есть серьезный недостаток – большое количество ложных обнаружений.

Отслеживание объектов на кадрах видеопоследовательности является еще одной хорошо изученной задачей машинного зрения. Классификация современных алгоритмов отслеживания объектов приведена в работе [2]. Обычно алгоритмы отслеживания объектов разрабатываются с целью сопровождения одного объекта и имеют высокую вычислительную сложность.

Исследование проводилось с целью разработки нового алгоритма, способного в реальном времени обнаруживать и отслеживать движущиеся объекты небольшого размера, видимые на движущейся камере, с низким процентом ложных обнаружений.

Одновременное обнаружение и отслеживание движущихся объектов. Предлагаемый в настоящей статье подход к обнаружению движения использует основные идеи, характерные для методов, основанных на компенсации движения и оптическом потоке. Вначале при помощи метода компенсации движения находятся области, которые, возможно, принадлежат движущимся объектам. После этого в найденных областях выбираются точки, которые в дальнейшем отслеживаются при помощи оптического потока. Похожий подход применялся в работе [3], где области возможного движения сначала оцениваются при помощи CNN, а затем при помощи оптического потока анализируется величина их смещения. Предлагаемый алгоритм не анализирует области возможного движения при помощи нейронной сети, поскольку вид этих областей может зависеть от внешнего вида как движущихся, так и неподвижных объектов, скорости и вида их движения, скорости и вида движения камеры, из-за чего обучение сети для универсального применения представляется большой проблемой. Вместо этого осуществляется отслеживание выбранных точек на протяжении долгого времени. Если на нескольких соседних кадрах не фиксируется смещение отслеживаемых точек, они удаляются. На каждом кадре близ-

кие по расположению и форме траекторий движения точки объединяются в группы, которые предположительно принадлежат движущимся объектам. Группы движущихся точек с соседних кадров сопоставляются в том случае, если они содержат большое число общих отслеживаемых точек, благодаря чему осуществляется отслеживание объектов, которые предположительно являются движущимися. Итоговым критерием того, что отслеживаемый объект движущийся, будет его смещение относительно фона за время отслеживания на заданную длину в пикселах, которая является параметром алгоритма. В результате такого подхода удалось получить низкий процент ложных обнаружений движущихся объектов (ниже, чем представленный в статье [2]).

Таким образом, разработанный алгоритм решает сразу несколько сложных задач компьютерного зрения. Он был протестирован на видеопоследовательностях, снятых с различных БПЛА, и показал свою применимость для решения задачи обнаружения движущихся объектов и их сопровождения.

Алгоритм обнаружения и отслеживания движущихся объектов. На кадрах видеопоследовательности, снимаемой бортовой камерой БПЛА, требуется в автоматическом режиме найти движущиеся объекты в режиме реального времени или близком к нему. В случае остановки обнаруженного движущегося объекта его отслеживание прекращается до того момента, пока он снова не начнет движение. Каждый движущийся объект выделяется ограничивающим прямоугольником и соответствующим ему номером, который свидетельствует о том, что сопровождается тот же самый объект.

В основе представленного алгоритма лежит идея отслеживания некоторых специально выбираемых точек, количество которых n строго фиксировано. На текущем кадре точки выбираются из областей, в которых интенсивность пикселей отличается от интенсивностей тех же пикселей на предыдущих кадрах, совмещенных с текущим кадром при помощи проективных преобразований. Далее выбранные точки отслеживаются на следующих кадрах с помощью алгоритма нахождения разреженного оптического потока Лукаса – Канаде [4] (известного метода компьютерного зрения, позволяющего найти соответствия между точками на двух последовательных кадрах видеопоследовательности).

Пусть $\mathbf{I} = \{I_1, \dots, I_k, \dots\}$ – последовательность кадров, на которых необходимо обнаруживать и отслеживать движущиеся объекты; $\mathbf{A} = \{P_1, \dots, P_n\}$ – массивы положений отслеживаемых точек на кадрах из \mathbf{I} , т. е. $P_i = \{p_i^{k-q_i}, \dots, p_i^{k-1}, p_i^k\}$ на кадре I_k представляет собой массив из положений отслеживаемой точки на всех кадрах с начала ее отслеживания и до текущего момента. Обозначим $\mathbf{P}^k = \{p_1^k, \dots, p_n^k\}$ положения отслеживаемых точек на кадре I_k , а $\Theta^k = \{S_1^k, \dots, S_n^k\}$ – массив траекторий отслеживаемых точек на кадре I_k , где каждая траектория $S_i^k = \{s_i^{k-q_i, k}, \dots, s_i^{k-1, k}, s_i^{k, k}\}$ отслеживаемой точки представляет собой массив точек на кадре I_k , которые соответствуют положениям P_i отслеживаемой точки на предыдущих кадрах. В S_i^k точка $s_i^{k, k} = p_i^k$, а каждая точка $s_i^{k-j, k}$ находится по формуле $s_i^{k-j, k} = H_{k-1}^k \circ H_{k-2}^{k-1} \circ \dots \circ H_{k-j}^{k-j+1}(p_i^{k-j})$, где H_{q-1}^q – проективное преобразование, которое переводит точки кадра I_{q-1} на кадр I_q в предположении, что они не движутся. На практике точка $s_i^{k, k-j} \in S_i^k$ кадра I_k всегда находится на основании точки $s_i^{k-j, k-1} \in S_i^{k-1}$ кадра I_{k-1} по формуле $s_i^{k, k-j} = H_{k-1}^k(s_i^{k-j, k-1})$. Пусть далее $\Theta^k = \{S_1^k, \dots, S_n^k\}$ – массив траекторий отслеживаемых точек на кадре I_k . На рис. 1 показаны траектории отслеживаемых точек на одном из кадров.

На каждом кадре близкие по расположению и форме траекторий отслеживаемые точки объединяются в группы, которые предположительно соответствуют движущимся объектам. Обозначим все найденные в ходе работы алгоритма движущиеся объекты как $\mathbf{O} = \{O_1, \dots, O_l, \dots\}$. Если движущийся объект O_l присутствует на текущем кадре I_k , то он характеризуется цен-

тральной точкой C_i^k , вектором скорости V_i^k , траекторией $T_i^k = \{C_i^{k-m,k}, \dots, C_i^{k-1,k}, C_i^{k,k}\}$, ограничивающим прямоугольником R_i^k , принадлежащими ему положениями отслеживаемых точек $Q_i^k \subset P^k$, а также нормированной гистограммой цветов $\mathbf{h}_i^k = \{h_1^k, \dots, h_{64}^k\}$ точек из Q_i^k .

Пусть I_k – текущий кадр видеопоследовательности \mathbf{I} . Задачей разработанного алгоритма является нахождение на нем движущихся объектов. Кроме этого, данный алгоритм устанавливает соответствие между движущимися объектами, обнаруженными на текущем кадре I_k , и объектами, которые ранее были обнаружены на предыдущих кадрах видеопотока, благодаря чему отслеживаются движущиеся объекты на кадрах видеопоследовательности.

Ниже описываются 11 шагов разработанного алгоритма, с помощью которых отслеживаются движущиеся объекты на всех кадрах видеопоследовательности, кроме двух стартовых. После этого приводятся действия, которые осуществляются на стартовых кадрах видеопоследовательности.

Первым шагом алгоритма является нахождение проективного преобразования H_{k-1}^k между I_{k-1} и I_k , которое переводит точки кадра I_{k-1} на кадр I_k в предположении, что они не движутся. Для нахождения H_{k-1}^k необходимо сначала установить соответствие между некоторыми точками (желательно, не принадлежащими движущимся объектам) кадров I_{k-1} и I_k . Для этого берутся точки кадра I_{k-1} , расположенные на равномерной сетке с шагом 50 пикселей, и ищутся точки, соответствующие им на кадре I_k , с помощью алгоритма вычисления разреженного оптического потока.

На *втором шаге* определяются \mathbf{P}^k и Θ^k – массивы положений и траекторий отслеживаемых точек на k -м кадре. Точки \mathbf{P}^k ищутся при помощи вычисления оптического потока между кадрами I_{k-1} и I_k для точек \mathbf{P}^{k-1} . После этого из массивов Λ , \mathbf{P}^k , \mathbf{P}^{k-1} и Θ^{k-1} удаляются некоторые элементы, не удовлетворяющие нескольким перечисленным ниже условиям. Если точке p_i^k выходит за пределы кадра I_k либо длина вектора смещения точки $ds_i^k = \|p_i^k - H^k(p_i^{k-1})\|$ больше заданного параметра алгоритма d_{\max} , то из массивов Λ , \mathbf{P}^k , \mathbf{P}^{k-1} и Θ^{k-1} удаляются P_i , p_i^k , p_i^{k-1} и S_i^{k-1} соответственно. Далее все неудаленные точки из \mathbf{P}^k добавляются в массивы из Λ , т. е. каждая точка $p_i^k \in \mathbf{P}^k$ добавляется в соответствующий ей массив $P_i \in \Lambda$. Чтобы найти траектории Θ^k , нужно сначала перевести все точки траекторий Θ^{k-1} на кадр I_k при помощи гомографии H_{k-1}^k и добавить к ним найденные при помощи оптического потока точки \mathbf{P}^k .

На *третьем шаге* алгоритма в Θ^k ищутся траектории, которые соответствуют движущимся на кадре I_k точкам. Вначале для каждой траектории $S_i^k \in \Theta^k$, количество точек в которой m_i , вычисляются три параметра: L_i – суммарная длина траектории S_i^k ; d_i – медиана среди 10 последних (либо всех, если $m_i < 10$) смещений точек в траектории S_i^k ; η_i – условный коэффициент, который принимает значение 1, если траектория S_i^k не содержит резких (более 90°) изменений направления, и 0 – в противном случае. После этого траектория S_i^k считается соответствующей движущейся точке при выполнении условий

$$\begin{cases} d_i \geq d_{\min}, \\ L_i > d_{\min} \cdot m_i, \\ \eta_i = 1, \\ m_i > \tau_{\min}, \end{cases}$$

где d_{\min} и τ_{\min} – параметры алгоритма, которые задают минимальное смещение и минимальное время отслеживания точек соответственно. На рис. 1 положения и траектории движущихся точек показаны голубым цветом, а неподвижных – черным.



Рис. 1. Положения и траектории движущихся и неподвижных отслеживаемых точек для одного из кадров



Рис. 2. Группы траекторий движущихся точек для одного из кадров

На четвертом шаге производится объединение положений и траекторий движущихся точек в группы. Пусть $\hat{\mathbf{P}}^k \subset \mathbf{P}^k$ и $\hat{\Theta}^k \subset \Theta^k$ – массивы положений и траекторий движущихся точек. Вначале для каждой движущейся точки с положением $p_i^k \in \hat{\mathbf{P}}^k$ и траекторией $S_i^k \in \hat{\Theta}^k$ на кадре I_k определяются два параметра: вектор скорости движущейся точки $V_i^k = (s_i^{k,k} - s_i^{k-m_i+1,k}) / m_i$, где m_i равно либо количеству точек в траектории S_i^k , если оно меньше 10, либо 10; оценка расстояния до ближайших точек группы $\rho_i^k = 4 + 2 \cdot \log_2 r_i^{\min}$, где r_i^{\min} – расстояние от точки p_i^k до ближайшей точки из $\hat{\mathbf{P}}^k$. Затем находится первая группа движущихся точек с положениями $\hat{\mathbf{Q}}_1^k \subset \hat{\mathbf{P}}^k$ и траекториями $\hat{\Omega}_1^k \subset \hat{\Theta}^k$. Для этого положение $p_i^k \in \hat{\mathbf{P}}^k$ и траектория $S_i^k \in \hat{\Theta}^k$ любой движущейся точки добавляются в пустые массивы $\hat{\mathbf{Q}}_1^k$ и $\hat{\Omega}_1^k$ и затем находятся положения и траектории движущихся точек с близкими к p_i^k и V_i^k положениями и скоростями, т. е. движущиеся точки с положениями $p_j^k \in \hat{\mathbf{P}}^k$ и траекториями $S_j^k \in \hat{\Theta}^k$, для которых выполняются условия

$$\begin{cases} r_{i,j}^k \leq \rho_i^k + \rho_j^k, \\ \|V_i^k\| < 2 \cdot \|V_j^k\|, \\ \|V_j^k\| < 2 \cdot \|V_i^k\|, \\ \alpha_{i,j}^k < \pi/3, \end{cases}$$

где $r_{i,j}^k$ – расстояние между точками p_i^k и p_j^k ; $\alpha_{i,j}^k$ – угол между векторами V_i^k и V_j^k . Далее точка с положением $p_i^k \in \hat{\mathbf{Q}}_1^k$ и траектория $S_i^k \in \hat{\Omega}_1^k$ помечаются как просмотренные, просматривается следующая движущаяся точка с положением $p_j^k \in \hat{\mathbf{Q}}_1^k$ и траекторией $S_j^k \in \hat{\Omega}_1^k$, а в массивы $\hat{\mathbf{Q}}_1^k$ и $\hat{\Omega}_1^k$ добавляются положения и траектории близких к ней движущихся точек в том случае, если они не были добавлены ранее. Процесс добавления положений и траекторий новых точек и траекторий продолжается, пока не будут просмотрены все точки и траектории в массивах $\hat{\mathbf{Q}}_1^k$ и $\hat{\Omega}_1^k$. Аналогично находится вторая группа движущихся точек с положениями $\hat{\mathbf{Q}}_2^k \subset \hat{\mathbf{P}}^k \setminus \hat{\mathbf{Q}}_1^k$ и траекториями $\hat{\Omega}_2^k \subset \hat{\Theta}^k \setminus \hat{\Omega}_1^k$, и так продолжается до тех пор, пока все движущиеся точки с положениями и траекториями из $\hat{\mathbf{P}}^k$ и $\hat{\Theta}^k$ не будут добавлены в группы.

Пусть далее $\hat{\Sigma}^k = \{\hat{Q}_1^k, \dots, \hat{Q}_j^k, \dots\}$ и $\hat{\Psi}^k = \{\hat{\Omega}_1^k, \dots, \hat{\Omega}_j^k, \dots\}$ – множества таких найденных массивов положений и траекторий движущихся точек, в которых количество положений и траекторий превосходит заданный параметр ν . На рис. 2 разными цветами показаны найденные массивы траекторий движущихся точек из $\hat{\Psi}^k$.

На *пятом шаге* алгоритма определяется, соответствуют ли найденные группы точек из $\hat{\Sigma}^k$ движущимся объектам, обнаруженным на предыдущем I_{k-1} кадре видеопоследовательности. Пусть объект $O_l \in \mathbf{O}$ был обнаружен на кадре I_{k-1} , тогда ему соответствует некоторая группа точек $Q_l^{k-1} \subset \mathbf{P}^{k-1}$. Пусть $\mathbf{O}^{k-1} \subset \mathbf{O}$ – массив объектов, обнаруженных на предыдущем кадре I_{k-1} . Будем считать, что движущаяся точка с положениями $P_i \in \Lambda$ на кадре I_{k-1} принадлежит объекту O_l , если $p_i^{k-1} \in Q_l^{k-1}$. Также будем считать, что движущаяся точка с положениями $P_i \in \Lambda$ принадлежит группе $\hat{Q}_j^k \in \hat{\Sigma}^k$ на кадре I_k , если $p_i^k \in \hat{Q}_j^k$. Тогда каждому объекту $O_l \in \mathbf{O}^{k-1}$, обнаруженному на кадре I_{k-1} , ставится в соответствие группа из $\hat{\Sigma}^k$, которой принадлежит на кадре I_k максимальное количество принадлежащих объекту O_l на кадре I_{k-1} движущихся точек. Если нет ни одной движущейся точки, одновременно принадлежащей объекту O_l на кадре I_{k-1} и какой-то группе из $\hat{\Sigma}^k$ на кадре I_k , то будем считать объект O_l потерянным на кадре I_k .

На *шестом шаге* определяются характеристики движущихся объектов, которым на предыдущем шаге удалось поставить в соответствие группы точек из $\hat{\Sigma}^k$. Пусть объекту $O_l \in \mathbf{O}^{k-1}$ соответствует группа $\hat{Q}_j^k \in \hat{\Sigma}^k$. Тогда группа точек Q_l^k либо равна \hat{Q}_j^k в том случае, если не было установлено соответствие между \hat{Q}_j^k и другими объектами из \mathbf{O}^{k-1} , либо состоит только из тех точек, принадлежащих \hat{Q}_j^k , которые являются положениями на кадре I_k движущихся точек, принадлежащих объекту O_l на кадре I_{k-1} . Далее находятся центр объекта C_l^k как среднее значение координат точек из группы Q_l^k , нормированная гистограмма цветов h_l^k точек из Q_l^k и скорость объекта V_l^k как среднее значение скоростей движущихся точек, принадлежащих Q_l^k на кадре I_k . Для нахождения траектории объекта T_l^k нужно сначала перевести все точки траекторий T_l^{k-1} на кадр I_k при помощи гомографии H_{k-1}^k и добавить к ним C_l^k .

На *седьмом шаге* алгоритма устанавливаются соответствия между группами точек из $\hat{\Sigma}^k$, которым не удалось поставить в соответствие движущиеся объекты, обнаруженные на предыдущем кадре I_{k-1} видеопоследовательности, и потерянными объектами. Пусть далее $\tilde{\mathbf{O}} \subset \mathbf{O}$ – массив потерянных объектов и для каждого объекта $O_w \in \tilde{\mathbf{O}}$ известен номер последнего кадра t_w , на котором объект O_w был обнаружен. Тогда вначале из $\hat{\Sigma}^k$ удаляются группы точек, соответствующие объектам из \mathbf{O}^{k-1} , а из массивов $\tilde{\mathbf{O}}$ и \mathbf{O} удаляются такие объекты $O_w \in \tilde{\mathbf{O}}$, для которых $t_w < k - \xi$, где ξ – заданный параметр. Пусть для каждого потерянного объекта $O_w \in \tilde{\mathbf{O}}$ известны следующие показатели: $C_w^{t_w, k-1}$ – найденное на кадре I_{k-1} положение, которое объект занимал в момент времени t_w ; $T_w^{t_w, k-1}$ – траектория объекта до момента времени t_w , найденная на кадре I_{k-1} ; $V_w^{t_w, k-1}$ – вектор скорости объекта в момент времени t_w , пересчитанный для кадра I_{k-1} . Тогда для каждого потерянного объекта $O_w \in \tilde{\mathbf{O}}$ находятся значения $C_w^{t_w, k} = H_{k-1}^k(C_w^{t_w, k-1})$ и $V_w^{t_w, k}(k) = H_{k-1}^k(C_w^{t_w, k-1} + V_w^{t_w, k-1}) - H_{k-1}^k(C_w^{t_w, k-1})$. Также для каждой группы точек $\hat{Q}_j^k \in \hat{\Sigma}^k$ вычисляются среднее значение координат точек \hat{C}_j^k , средняя скорость точек \hat{V}_j^k

и нормированная гистограмма цветов точек $\hat{\mathbf{h}}_j^k$. Затем рассчитываются расстояния от каждого потерянного объекта из $\tilde{\mathbf{O}}$ до каждой группы из $\hat{\Sigma}^k$. Расстояние $d_{w,j}$ от $O_w \in \tilde{\mathbf{O}}$ до $\hat{Q}_j^k \in \hat{\Sigma}^k$ находится по формулам

$$\begin{aligned} Z_{w,j} &= \hat{C}_j^k - C_w^{t_w \cdot k}, \\ d_{w,j}^p &= 1/(1 + \rho_{w,j}), \\ d_{w,j}^c &= \|Z_{w,j} - (k - t_w) \cdot V_w^{t_w \cdot k}\| / \|(k - t_w) \cdot V_w^{t_w \cdot k}\|, \\ d_{w,j}^a &= 2 - \cos(\alpha_{w,j}) - \cos(\beta_{w,j}), \\ d_{w,j}^h &= \|\hat{\mathbf{h}}_j^k - \mathbf{h}_w^{t_w}\|, \\ d_{w,j} &= 0,3 \cdot d_{w,j}^p + 0,1 \cdot d_{w,j}^c + 0,4 \cdot d_{w,j}^a + 0,3 \cdot d_{w,j}^h, \end{aligned}$$

где $\rho_{w,j}$ – количество движущихся точек, одновременно принадлежащих группе \hat{Q}_j^k на кадре I_k и объекту O_w на кадре I_{t_w} ; $\alpha_{w,j}$ – угол между векторами $Z_{w,j}$ и \hat{V}_j^k ; $\beta_{w,j}$ – угол между векторами $Z_{w,j}$ и $V_w^{t_w \cdot k}$. Далее последовательно устанавливаются соответствия между элементами массивов $\tilde{\mathbf{O}}$ и $\hat{\Sigma}^k$. Для этого в массивах $\tilde{\mathbf{O}}$ и $\hat{\Sigma}^k$ находятся элементы, расстояние между которыми минимально, и если это расстояние меньше единицы, то между ними устанавливается соответствие, в противном случае процесс поиска соответствий прекращается. Если процесс поиска не прекратился, то среди других элементов массивов $\tilde{\mathbf{O}}$ и $\hat{\Sigma}^k$ ищется новая пара с минимальным расстоянием и так продолжается до тех пор, пока процесс не остановится. После этого происходит вычисление характеристик объектов из $\tilde{\mathbf{O}}$, которым удалось поставить в соответствие группы из $\hat{\Sigma}^k$. Пусть объекту $O_w \in \tilde{\mathbf{O}}$ соответствует группа $\hat{Q}_j^k \in \hat{\Sigma}^k$, тогда $\mathbf{Q}_w^k = \hat{Q}_j^k$, $C_w^k = \hat{C}_j^k$, $V_w^k = \hat{V}_j^k$, $\mathbf{h}_w^k = \hat{\mathbf{h}}_j^k$. Чтобы найти траектории объекта T_w^k , нужно сначала перевести все точки траекторий $T_w^{t_w \cdot k-1}$ на кадр I_k при помощи гомографии H_{k-1}^k и добавить к ним C_w^k .

На *восьмом шаге* алгоритма в массив \mathbf{O} добавляются новые движущиеся объекты, соответствующие некоторым группам точек из $\hat{\Sigma}^k$. Пусть $\hat{Q}_j^k \in \hat{\Sigma}^k$ не соответствует движущимся объектам из $\tilde{\mathbf{O}}$ и количество точек в самой длинной траектории из \hat{Q}_j^k больше заданного параметра τ , тогда в массив \mathbf{O} длины q добавляется новый объект O_{q+1} с характеристиками $\mathbf{Q}_{q+1}^k = \hat{Q}_j^k$, $C_{q+1}^k = \hat{C}_j^k$, $T_{q+1}^k = \{C_{q+1}^k\}$, $V_{q+1}^k = \hat{V}_j^k$, $\mathbf{h}_{q+1}^k = \hat{\mathbf{h}}_j^k$.

На *девятом шаге* алгоритма производится индикация движущихся объектов. Если длина траектории T_l^k объекта $O_l \in \mathbf{O}^k$ больше заданного параметра алгоритма λ , то для данного объекта вычисляется ограничивающий прямоугольник R_l^k точек \mathbf{Q}_l^k , который рисуется на изображении. Если объекту O_l на предыдущих кадрах не был присвоен уникальный идентификатор, то он присваивается на текущем кадре. На рис. 3 показан итоговый вид индикации движущихся объектов для одного из кадров.

Хотя движущиеся объекты были найдены на предыдущем шаге, для того чтобы алгоритм мог и дальше находить движущиеся объекты на последующих кадрах, нужно совершить еще несколько дополнительных шагов.

На *десятом шаге* удаляются некоторые положения и траектории отслеживаемых точек из массивов Λ , \mathbf{P}^k и Θ^k . Если количество точек в траектории S_i^k больше τ_{\min} и либо $d_i < d_{\min}$, либо $\eta_i = 0$, то из Λ , \mathbf{P}^k и Θ^k удаляются P_i , p_i^k и S_i^k соответственно.



Рис. 3. Индикация движущихся объектов для одного из кадров

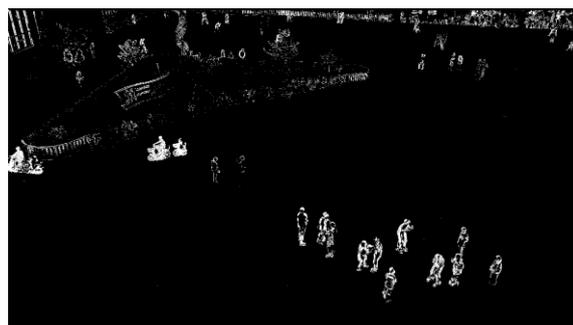


Рис. 4. Итоговый вид изображения D_k для одного из кадров

Одиннадцатым шагом алгоритма является добавление новых отслеживаемых точек. Пусть количество точек в массиве \mathbf{P}^k на данный момент равно q , где $q < n$. Обозначим число точек, которые необходимо добавить к \mathbf{P}^k , как $\mu = n - q$. Далее строится изображение D_k с такой же размерностью, как у кадра I_k , интенсивность каждого пиксела p которого равна $D_k(p) = \min\left(\left|I_k(p) - I_{k-1}(H_k^{k-3}(p))\right|, \left|I_k(p) - I_{k-1}(H_k^{k-6}(p))\right|\right)$ в том случае, если номер кадра $k > 6$, либо $D_k(p) = \left|I_k(p) - I_{k-1}(H_k^{k-1}(p))\right|$, если $k \leq 6$. В представленных формулах H_k^{k-j} – проективное преобразование, которое переводит точки кадра I_k на кадр I_{k-j} . (H_k^{k-j} можно найти как композицию преобразований между соседними кадрами, т. е. $H_k^{k-j} = H_{k-j+1}^{k-j} \circ \dots \circ H_{k-1}^{k-2} \circ H_k^{k-1}$, где, например, H_k^{k-1} – отображение, обратное к H_{k-1}^{k-1} .) Далее в D_k обнуляются значения тех пикселей, интенсивность которых меньше заданного параметра γ , и пикселей, соответствующих уже отслеживаемым точкам \mathbf{P}^k . На рис. 4 показан итоговый вид изображения D_k для одного из кадров. После нахождения изображения D_k на нем ищутся связанные области пикселей с ненулевой интенсивностью. После этого из $q/3$ самых больших найденных областей по очереди выбираются новые точки p_{q+j}^k , где $j \leq \mu$, и добавляются в \mathbf{P}^k . В массивы Λ и Θ^k также добавляются новые элементы $P_{q+j}^k = \{p_{q+j}^k\}$ и $S_{q+j}^k = \{p_{q+j}^k\}$.

Как было сказано выше, действия, которые осуществляются на стартовых кадрах I_1 и I_2 видеопотока, отличаются от описанных выше. Кадр I_1 просто сохраняется. Для кадра I_2 сначала необходимо найти проективное преобразование H_1^2 между I_1 и I_2 , как на первом шаге алгоритма, после чего добавить положения и траектории n отслеживаемых точек в пустые массивы Λ , \mathbf{P}^2 и Θ^2 , как на одиннадцатом шаге алгоритма.

Результаты тестирования алгоритма. Работа представленного алгоритма была проверена на видеопоследовательностях двух широко известных баз данных (UAV123 [5] и VisDrone [6]), снятых бортовой видеочкамерой БПЛА. Благодаря появившейся недавно базе данных PESMOD [7], содержащей размеченные видеопоследовательности с БПЛА, удалось численно сравнить результаты работы представленного алгоритма с тремя известными алгоритмами, основанными на компенсации движения: MCD [8], SCBU [9] и разработанным создателями базы данных PESMOD.

Для тестирования алгоритма из баз данных UAV123 и VisDrone были выбраны видеопоследовательности, полученные съемкой с достаточно большой высоты. Угол наклона камеры на видеопоследовательностях варьируется от 0 до 80° к вертикали. Выбранные видеопоследовательности были разбиты на две группы. В первую группу были отобраны 6 видеопоследовательностей, снятых с высоты выше 100 м, а во вторую – 16 видеопоследовательностей, снятых с высоты около 50 м. На тестовых видеопоследовательностях присутствуют движущиеся

объекты разного размера (от 40 до 10 000 пикселей), снятые при разном освещении. Количество одновременно движущихся объектов на некоторых видеопоследовательностях достигает 20. Размер кадров равен 1280×720 пикселей. Работа алгоритма тестировалась на персональном компьютере с процессором Intel(R) Core(TM) i7-4770 с тактовой частотой 3,5 ГГц. Быстродействие алгоритма на всех видеопоследовательностях составляет 25–30 кадров в секунду. На рис. 5 показаны результаты применения алгоритма на кадрах из видеопоследовательностей первой (сверху) и второй (снизу) групп соответственно.



Рис. 5. Примеры работы алгоритма на кадрах видеопоследовательностей первой (сверху) и второй (снизу) групп

При проведении анализа работы алгоритма для каждого движущегося объекта из видеопоследовательностей были записаны интервалы его движения, а также интервалы верной индикации движения, для каждого интервала индикации объекта – номер, которым идентифицировался объект. Отдельно были записаны номера и интервалы ошибочной индикации движущихся объектов. В том случае, если несколько объектов идентифицировались одним номером, считалось, что в данный момент идентифицировался только один объект (рис. 6). Если один объект идентифицировался несколькими номерами, считалось, что все индикации, кроме одной, ошибочны (рис. 7). Благодаря такому ручному описанию движений объектов и результатов работы алгоритма на видеопоследовательностях были вычислены некоторые характеристики качества работы алгоритма.



Рис. 6. Пример идентификации двух движущихся объектов одним индексом



Рис. 7. Пример идентификации одного движущегося объекта двумя индексами

Далее вначале рассмотрим характеристики качества индикации движущихся объектов, а затем характеристики качества сопровождения движущихся объектов.

С целью оценки качества индикации движущихся объектов вычислялись следующие характеристики: количество движущихся объектов на всех видеопоследовательностях, n ; количество обнаруженных хотя бы один раз движущихся объектов на всех кадрах всех видеопоследовательностей, m ; количество изображений движущихся объектов на всех кадрах всех видеопоследовательностей, N ; количество верных индикаций движущихся объектов на всех кадрах всех видеопоследовательностей, TP ; количество ошибочных индикаций движущихся объектов на всех кадрах всех видеопоследовательностей, FP ; точность (*precision*) индикации, $Pr = TP / (TP + FP)$; полнота (*recall*) индикации, $R = TP / N$.

Ниже представлены результаты тестирования для двух групп видеопоследовательностей. Видеопоследовательности первой группы, снятые с большой высоты, содержат движущиеся объекты небольшого размера (около 100 пикселей), смещение которых на соседних кадрах невелико, поэтому алгоритм запускался с $n = 2000$, $d_{\max} = 100$, $d_{\min} = 0,25$, $\tau_{\min} = 5$, $\tau = 20$, $v = 3$, $\xi = 100$, $\gamma = 20$ и $\lambda = 10$. Видеопоследовательности второй группы, снятые с небольшой высоты, содержат объекты разного размера и типа (в основном люди и автомобили), величина смещения объектов между кадрами сильно зависит от типа объекта и высоты съемки. На второй группе видеопоследовательностей алгоритм запускался с $n = 2000$, $d_{\max} = 100$, $d_{\min} = 0,5$, $\tau_{\min} = 5$, $\tau = 20$, $v = 5$, $\xi = 100$, $\gamma = 20$ и $\lambda = 40$. Оценки качества индикации движущихся объектов на кадрах видеопоследовательностей первой и второй групп приведены в табл. 1.

Таблица 1

Оценки качества индикации движущихся объектов

	Группа	
	Первая	Вторая
Количество движущихся объектов, n	48	229
Количество обнаруженных движущихся объектов, m	40	192
Количество изображений движущихся объектов на кадрах, N	10 869	59 405
Количество верных индикаций движущихся объектов, TP	5826	31 791
Количество ошибочных индикаций движущихся объектов, FP	168	1736
Точность индикации движущихся объектов, Pr	0,972	0,948
Полнота индикации движущихся объектов, R	0,536	0,535

Относительно невысокие значения полноты индикации R объясняются тем, что алгоритм перед индикацией объекта некоторое время отслеживает его до того момента, пока не будет зафиксировано значительное смещение объекта. Благодаря этому удалось получить высокую точность индикации Pr .

Чтобы увидеть зависимость вероятности верной индикации движущихся объектов от времени их движения, для каждого изображения движущегося объекта на кадрах видеопоследовательности было вычислено время движения – количество кадров, прошедшее с момента начала его движения. Оценки зависимости вероятности верной индикации движущегося объекта от времени движения для видеопоследовательностей первой группы приведены на рис. 8, а и 9, а, второй группы – на рис. 8, б и 9, б. На рис. 8 синим цветом показаны графики зависимости количества изображений движущихся объектов на кадрах видеопоследовательностей от времени движения, желтым – графики зависимости количества верных индикаций изображений движущихся объектов от времени движения. На рис. 9 изображены графики зависимости вероятности верной индикации движущихся объектов на кадрах видеопоследовательностей от времени движения.

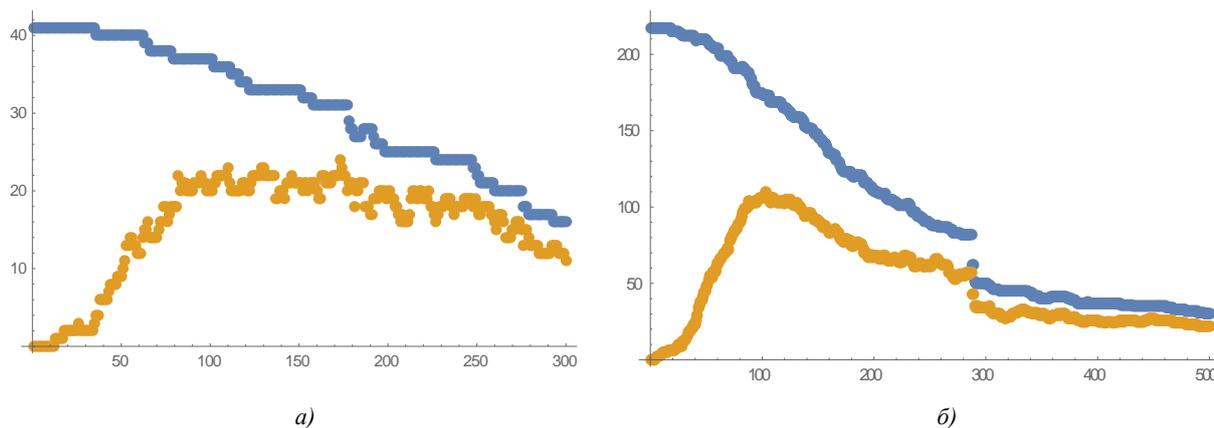


Рис. 8. Графики зависимостей количества изображений движущихся объектов и их индикаций от времени движения на кадрах видеопоследовательностей: а) первой группы; б) второй группы

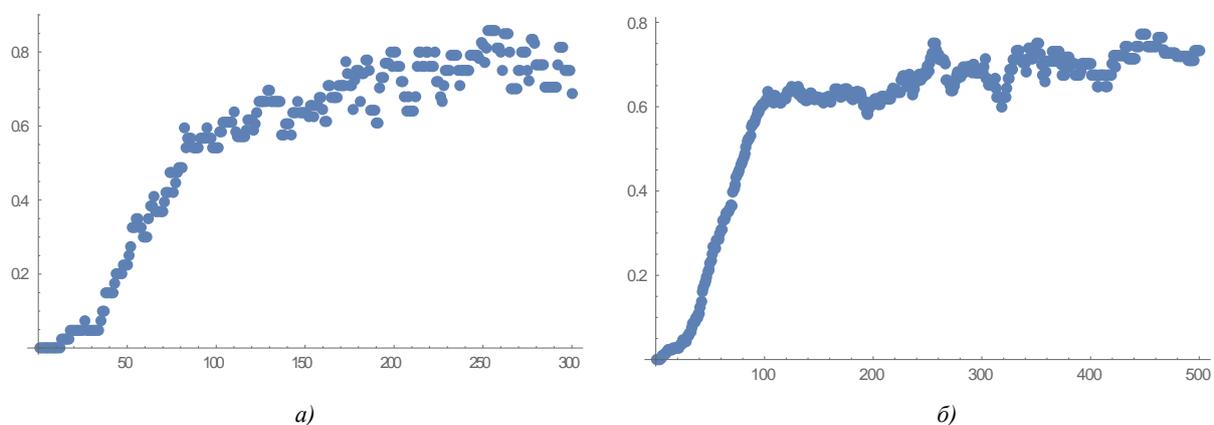


Рис. 9. Графики зависимости вероятности верной индикации движущихся объектов от времени движения для видеопоследовательностей: а) первой группы; б) второй группы

На представленных графиках видно, что алгоритм плохо обнаруживает движущиеся объекты в начале их движения, однако со временем вероятность индикации линейно растет примерно до 100-го кадра, после 100-го кадра вероятность индикации составляет примерно 0,6–0,8.

С целью оценки качества отслеживания движущихся объектов вычислялись следующие характеристики: количество обнаруженных движущихся объектов на всех видеопоследовательностях, m ; количество верно отслеженных движущихся объектов на всех видеопоследовательностях, t ; вероятность верного отслеживания объекта на видеопоследовательности, $P_i = t / m$. Обнаруженный объект считался верно отслеженным в том случае, если на протяжении всего времени движения он идентифицировался только одним числом и ни один другой объект (либо ложное обнаружение) не идентифицировался этим числом. Оценки качества отслеживания движущихся объектов на кадрах видеопоследовательностей первой и второй групп приведены в табл. 2.

Таблица 2

Оценки качества отслеживания движущихся объектов

	Группа	
	Первая	Вторая
Количество обнаруженных движущихся объектов, m	40	192
Количество верно отслеженных движущихся объектов, t	35	140
Вероятность верного отслеживания объекта, P_i	0,875	0,729

Из представленных вероятностей неясно, как долго алгоритм может отслеживать движущийся объект, поскольку все верные сопровождения могут прийти на объекты, которые двигались непродолжительное время. Для того чтобы увидеть зависимость вероятности верного отслеживания такого объекта от времени движения, для каждого идентифицированного изображения движущегося объекта на кадрах видеопоследовательности было вычислено время отслеживания – количество кадров, прошедшее с момента его первой индикации на предыдущих кадрах. Также было проверено, верно ли отслежен объект, т. е. идентифицируется ли он на всех предыдущих кадрах только одним числом. Результаты анализа зависимости вероятности верного отслеживания движущегося объекта от времени отслеживания для видеопоследовательностей первой группы приведены на рис. 10, *а* и 11, *а*, второй группы – на рис. 10, *б* и 11, *б*. На рис. 10 и 11 желтым цветом показаны графики зависимости количества идентифицированных изображений движущихся объектов на кадрах видеопоследовательностей от времени отслеживания, синим – графики зависимости количества верно отслеженных изображений движущихся объектов от времени отслеживания. На рис. 11 изображены графики зависимости вероятности верного отслеживания движущихся объектов на кадрах видеопоследовательностей от времени отслеживания.

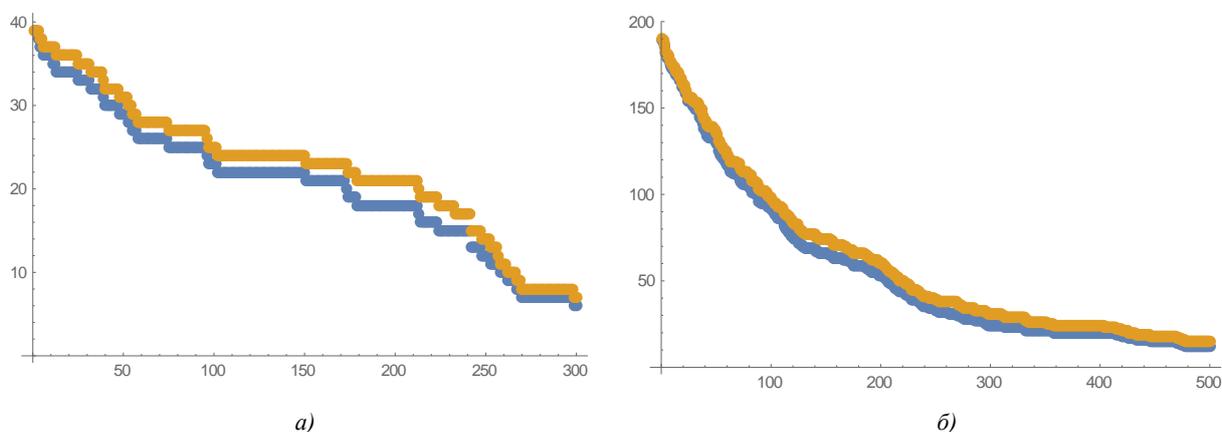


Рис. 10. Графики зависимости количества идентифицированных и верно отслеженных изображений движущихся объектов от времени отслеживания на кадрах видеопоследовательностей: *а*) первой группы; *б*) второй группы

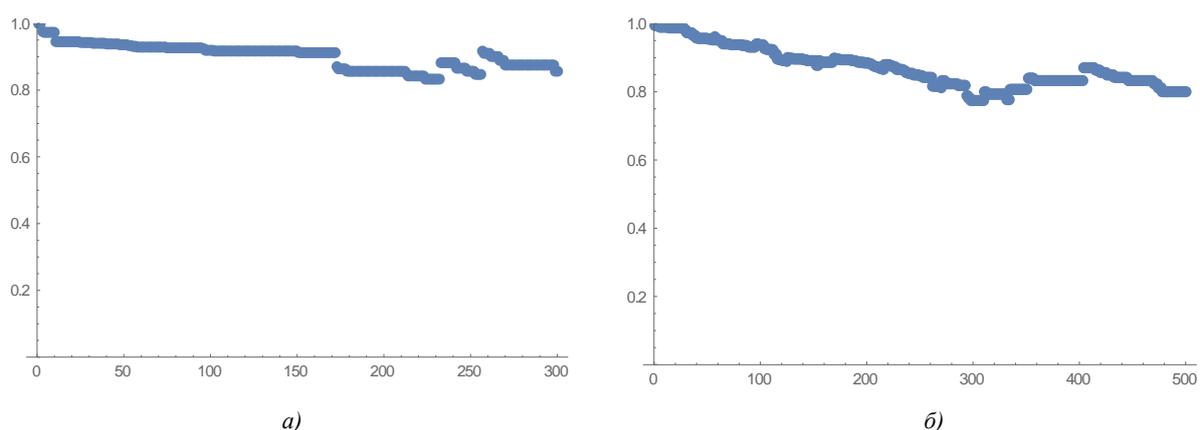


Рис. 11. Графики зависимости вероятности верного отслеживания движущихся объектов от времени отслеживания для видеопоследовательностей: *а*) первой группы; *б*) второй группы

На рис. 10 и 11 видно, что алгоритм надежно сопровождает движущиеся объекты, вероятность верного сопровождения незначительно снижается с течением времени, однако остается высокой даже для объектов, которые на тестовых видеопоследовательностях двигались наиболее продолжительное время.

Кроме представленного выше анализа работы алгоритма, было проведено его сравнение с известными алгоритмами индикации движущихся объектов, основанными на компенсации движения. Сравнение проводилось на базе данных PESMOD, в которой находятся шесть сложных видеопоследовательностей с небольшими движущимися объектами. Размер кадров в PESMOD равен 1920×1080 пикселей. На рис. 16 приведены примеры работы алгоритма на кадрах последовательностей из PESMOD. Зелеными рамками показаны движущиеся объекты, выделенные вручную, синими – найденные алгоритмом.



Рис. 16. Примеры работы алгоритма на кадрах видеопоследовательностей PESMOD

Благодаря программному обеспечению, разработанному создателями базы PESMOD, для каждой видеопоследовательности удалось вычислить следующие критерии качества работы алгоритма индикации: точность (precision) индикации, Pr ; полноту (recall) индикации, R и обобщающий их критерий $F_1 = 2 \cdot Pr \cdot R / (Pr + R)$.

В табл. 3 производится сравнение рассматриваемого в настоящей статье алгоритма с известными алгоритмами MCD, SCBU, а также алгоритмом, разработанным создателями базы данных PESMOD. Данные, содержащиеся в таблице, – это средние значения критериев Pr , R и F_1 , полученные для всех шести последовательностей из PESMOD. Предлагаемый алгоритм тестировался при $n=1000$, $d_{\max}=100$, $d_{\min}=0,5$, $\tau_{\min}=5$, $\tau=5$, $\nu=3$, $\xi=100$, $\gamma=30$, $\lambda=10$.

Таблица 3

Сравнительные характеристики представленного алгоритма и известных алгоритмов

Показатели	MCD	SCBU	PESMOD	Представленный алгоритм
Pr	0,5424	0,4268	0,5452	0,8636
R	0,5607	0,4416	0,5070	0,6338
F_1	0,3828	0,4108	0,4302	0,6718

Из табл. 3 видно, что разработанный алгоритм превосходит рассмотренные известные алгоритмы, основанные на компенсации движения. Наибольшим преимуществом алгоритма является высокий показатель точности индикации Pr .

Проведенный анализ показывает, что предварительное отслеживание движущихся объектов позволяет значительно понизить вероятность ложных обнаружений движущихся объектов.

Заключение. Разработан алгоритм автоматического обнаружения и отслеживания движущихся объектов, предназначенный для использования на борту БПЛА. Эксперименты с видеопоследовательностями, снятыми видекамерами с различных БПЛА, показали работоспособность предложенного алгоритма. Достоинством алгоритма является то, что он редко обнаруживает ложное движение. Вместе с тем данный алгоритм не сразу обнаруживает движущиеся объекты, а только через некоторое время после начала их движения – это, несомненно, его недостаток.

Численное сравнение работы представленного алгоритма на базе данных PESMOD с тремя известными алгоритмами, основанными на компенсации движения, показало, что он превосходит другие алгоритмы по всем важным показателям качества индикации движущихся объектов, поскольку надежно сопровождает движущиеся объекты и способен в реальном времени решать задачу сопровождения их большого количества. Кроме того, представленный алгоритм способен обнаруживать и отслеживать движущиеся объекты малого размера (до 40 пикселей), смещение которых между соседними кадрами в среднем может составлять 0,25 пикселя. Таким образом, разработанный алгоритм может успешно применяться на борту БПЛА.

References

1. Chapel M.-N., Bouwmans T. *Moving Objects Detection with a Moving Camera: A Comprehensive Review*. Available at: <https://arxiv.org/abs/2001.05238> (accessed 19.04.2021).
2. Fiaz M., Mahmood A., Jung S.-K. *Tracking Noisy Targets: A Review of Recent Object Tracking Approaches*. Available at: <https://arxiv.org/abs/1802.03098> (accessed 19.04.2021).
3. Ye D. H., Li J., Chen Q., Wachs J., Bouman C. Deep learning for moving object detection and tracking from a single camera in unmanned aerial vehicles (UAVs). *IS&T International Symposium on Electronic Imaging 2018*. Burlingame, California, USA, 2018, pp. 4661–4666.
4. Lucas B., Kanade T. An iterative image registration technique with an application to stereo vision. *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)*. Vancouver, 1981, vol. 2, pp. 674–679.
5. Mueller M., Smith N., Ghanem B. A benchmark and simulator for uav tracking. *Proceedings of the 14th European Conference on Computer Vision (ECCV '16)*. Amsterdam, 2016, pp. 445–461. https://doi.org/10.1007/978-3-319-46448-0_27
6. Du D., Zhu P., Wen L., Bian X., Ling H., ..., Liu Z. VisDrone-DET2019: the vision meets drone object detection in image challenge results. *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. Seoul, 2019, pp. 213–226. <https://doi.org/10.1109/ICCVW.2019.00030>
7. Delibasoglu I. *UAV Images Dataset for Moving Object Detection from Moving Cameras*. Available at: <https://arxiv.org/abs/2103.11460> (accessed 19.04.2021).
8. Yi K. M., Yun K., Kim S. W., Chang H., Choi J. Detection of moving objects with non-stationary cameras in 5.8ms: bringing motion detection to your mobile device. *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPR)*. Portland, OR, USA, 2013, pp. 27–34.
9. Yun K., Lim J., Choi J. Y. Scene conditional background update for moving object detection in a moving camera. *Pattern Recognition Letters*, 2017, vol. 88, pp. 57–63. <https://doi.org/10.1016/j.patrec.2017.01.017>

Информация об авторе

Жук Роман Сергеевич, младший научный сотрудник, Объединенный институт проблем информатики Национальной академии наук Беларуси, Минск, Беларусь.
E-mail: ramanzhuck@gmail.com

Information about the author

Raman S. Zhuk, Junior Researcher, the United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Minsk, Belarus.
E-mail: ramanzhuck@gmail.com

УЧЕНЫЕ БЕЛАРУСИ SCIENTISTS OF BELARUS



**Киркоров Николай Иванович –
талантливый разработчик,
изобретатель и руководитель**

**Kirkorov Nikolai Ivanovich –
talented developer,
inventor and leader**

19 января 2021 г. на 76-м году ушел из жизни Николай Иванович Киркоров – талантливый и высококвалифицированный специалист в сфере создания специализированного программного обеспечения, используемого в аэрокосмической и картографической областях знаний.

Николай Иванович родился 22 мая 1945 г. в г. Горячий Ключ Краснодарского края России, школу закончил в Минске. После окончания Минского радиотехнического института в 1968 г. по специальности «Математические и счетно-решающие приборы и устройства» был распределен в лабораторию программирования Специального конструкторского бюро Минского завода счетных машин им. Г. К. Орджоникидзе (ныне ОАО «МПОВТ»). Николай Иванович занимался вопросами системного программирования и цифровой обработки изображений.

В период с 1978 по 1989 г. работал начальником отдела № 110 СКТБ с ОП Института технической кибернетики АН БССР (ныне Объединенный институт проблем информатики НАН Беларуси) и являлся одним из основных руководителей и разработчиков ряда проектов оборонной тематики, в частности проекта по разработке методов и средств картографического обеспечения крылатых ракет. С 1989 г. (после присоединения СКТБ с ОП к институту) по 1997 г. занимал должность главного конструктора проекта в лаборатории геоинформатики.

С 1998 г. и вплоть до 2020 г. Николай Иванович работал в научно-конструкторском управлении (НКУ) «Космос» ОАО «Пеленг»: сначала главным специалистом по программному обеспечению цифрового картографирования, а затем заместителем начальника отдела. В НКУ «Космос» руководил и лично участвовал в разработке систем наземной обработки информации целевой аппаратуры космических аппаратов, в том числе БКА. С 1998 по 2001 г. являлся членом некоммерческой организации «ГИС-Ассоциация» (Москва), принимал участие в международных конференциях по проблемам ДЗЗ, проводимых компанией «Совзонд» и Институтом космических исследований РАН.

Н. И. Киркоров запатентовал четыре изобретения в области распознавания образов, опубликовал ряд работ по распознаванию образов и цифровой фотограмметрической обработке данных ДЗЗ. В период с 1997 по 1998 г. в инициативном порядке, самостоятельно и без консультаций ученых-фотограмметристов им была разработана цифровая фотограмметрическая система DPS-97, а затем ее модификация RealisticMap для обработки данных космосъемки и данных блока аэрофотостереосъемки, выходные результаты расчетов которой были сопоставимы с данными известных зарубежных систем. На базе разработок Николая Ивановича был создан пакет программ по обработке материалов аэрофотостереосъемки, который затем использовался в навигационно-топографической службе Вооруженных Сил Республики Беларусь.

Николай Иванович являлся одним из тех специалистов, благодаря которым была успешно введена в эксплуатацию и функционирует Белорусская космическая система дистанционного зондирования Земли. Созданное под его руководством специализированное программное обеспечение позволило успешно обработать первую в Беларуси собственную космическую информацию, полученную от целевой аппаратуры БКА. Результаты обработки первого сброса целевой информации БКА показали корректность реализованных математических алгоритмов, а сформированные изображения земной поверхности совокупно с их сопроводительной информацией позволили убедиться в том, что штатно работает не только целевая аппаратура, но и некоторые другие системы космического аппарата. Созданное программное обеспечение (первое в Беларуси с таким функционалом) по сути было соизмеримо с тем, которое разработал коллектив российских ученых и программистов. Н. И. Киркоров также принимал активное участие в летных испытаниях БКА, оперативно реализовывал дополнительный функционал. Он показал себя специалистом, способным творчески, на высоком уровне и в сжатые сроки решать сложные технические задачи.

За годы своей трудовой деятельности Николай Иванович активно сотрудничал с коллегами из навигационно-топографического управления Генерального штаба Вооруженных Сил Республики Беларусь, предприятий «Белгеодезия» и «БелПСХАГИ», РУП «Проектный институт Белгипрозем», Института природопользования НАН Беларуси, факультета географии и геоинформатики Белгосуниверситета, Объединенного института проблем информатики и УП «Геоинформационные системы» НАН Беларуси.

Н. И. Киркоров пользовался заслуженным авторитетом и уважением среди специалистов по обработке данных ДЗЗ, обладал высоким уровнем квалификации в областях обработки изображений, картографии, геодезии, фотограмметрии, низкоуровневого и высокоуровневого программирования, отлично знал технологии, следил за публикациями в отечественной и зарубежной научной литературе. Его компетентность и высокий профессионализм обеспечивали плодотворное сотрудничество не только с отечественными специалистами, но и коллегами из России, Китая, Индии и Египта. За успешные результаты выполненных проектов оборонной тематики Николай Иванович Киркоров был награжден орденом «Знак Почета», удостоен медали «90 лет Вооруженным Силам Республики Беларусь».

Более 40 лет Николай Иванович являлся руководителем ряда коллективов разработчиков, создававших пакеты программ специализированного назначения, был прекрасным организатором и энтузиастом своего дела, поражал работоспособностью и заряжал коллег своей увлеченностью работой, всегда непосредственно участвовал во всех проектах. Он брал на себя разработку алгоритмов и реализацию самых ответственных элементов системы, уважал и ценил специалистов высокого класса, с удовольствием с ними сотрудничал, при этом никогда не отказывал в помощи своим сотрудникам, терпеливо объяснял пути решения поставленной задачи, обучал и охотно делился своими знаниями и накопленным опытом. Благодаря его усилиям за все эти годы большому количеству молодых специалистов был дан серьезный стартовый уровень успешной трудовой деятельности. Николай Иванович постоянно совершенствовал свои разработки и построил хороший фундамент для создания новых и дальнейшего развития существующих программных пакетов, разрабатываемых НКУ «Космос». Отдельные элементы системы в настоящее время используются в работах по созданию целевой аппаратуры нового космического аппарата Беларуси. Н. И. Киркоров внес большой вклад в решение задач аэрокосмической области в нашей стране, трудился на благо ее развития до последних дней жизни.

Николай Иванович всегда практически полностью был погружен в работу и вместе с тем оставался разносторонним человеком: занимался спортом, увлекался туризмом и горными лыжами, ценил хорошую поэзию и искренне восхищался талантливыми людьми из любой сферы деятельности. Это был любящий отец, неравнодушный и порядочный человек, умевший поддерживать в трудной ситуации и оказать помощь.

А. В. Тузиков, Д. А. Сикорский, Г. В. Лосик, Т. Г. Царик



О научной деятельности Николая Романовича Торопова

About scientific activity of Nikolai Romanovich Toropov

50 лет тому назад – 26 ноября 1971 г. – в Институте технической кибернетики АН БССР по инициативе доктора технических наук А. Д. Закревского и его учеников была создана лаборатория системного программирования и логического синтеза, впоследствии переименованная в лабораторию логического проектирования ОИПИ НАН Беларуси. Николай Романович Торопов (13.06.1935–15.04.2021) возглавлял сектор системного программирования с момента основания лаборатории, в которой он проработал до 2009 г. Основным направлением его научной деятельности была разработка систем программирования для алгоритмического языка ЛЯПАС, автором которого являлся А. Д. Закревский. ЛЯПАС (Логический Язык Представления Алгоритмов Синтеза дискретных автоматов) появился на заре распространения универсальных цифровых вычислительных машин (УЦВМ) в Советском Союзе. Позже вместо УЦВМ стали употреблять термины «электронные вычислительные машины» и «компьютеры». Работы по созданию языка ЛЯПАС и первой системы программирования с ним были начаты в 1962 г. К марту 1964 г. был продемонстрирован в действии отечественный язык программирования ЛЯПАС на первой в Сибири машине «Урал-1» и были представлены результаты экспериментально-статистических испытаний оригинальных алгоритмов синтеза дискретных автоматов.

Вычислительная машина «Урал-1» была приобретена Томским университетом в 1958 г. К этому времени Н. Р. Торопов окончил радиофизический факультет Томского университета, и ему довелось осваивать новую технику. В составе группы своих товарищей-выпускников он выполнял обязанности инженера по эксплуатации машины «Урал-1» приблизительно в течение двух лет, до того момента, когда их сменили новые выпускники радиофизического факультета. Таким образом, Николай Романович был одним из пионеров освоения вычислительных машин в Сибири. После этого он занимался разработкой элементов вычислительной техники на полупроводниковых приборах, сменивших впоследствии громоздкие и малонадежные вакуумные лампы, на которых тогда строилась вся вычислительная техника.

Момент ввода программирующей системы ПС-ЛЯПАС в эксплуатацию совпал с пиком серийного выпуска машин «Урал-2» (Пенза) и М-20 (Казань) и их широкого распространения в СССР. В результате не только сам ЛЯПАС, но и прикладные программные продукты, созданные на его базе, оказались востребованы во многих городах Советского Союза (Москве, Ленинграде, Риге, Новосибирске, Свердловске, Киеве, Севастополе, Таганроге, Кишиневе, Фрунзе и др.). Наибольшее распространение получили две версии языка: ЛЯПАС-70 на машинах М-20, БЭСМ-3М, БЭСМ-4, СМ-4, «Минск-2», «Минск-22» и ЛЯПАС-71 на машинах М-220, БЭСМ-6. В 1974 г. в Минске начались исследования по реализации языка ЛЯПАС и созданию системы программирования для ЭВМ «Минск-32». Николай Романович добился высоких результатов в области системного программирования. Так, группа разработчиков под его руководством создала диалоговую систему ДИСМО, реализованную на машине

«Минск-32», которая была оснащена терминалами по типу пишущей машинки. ДИСМО была одной из первых интерактивных систем, показала хорошие эксплуатационные характеристики по параллельному обслуживанию многих абонентов и явилась предвестником будущих систем с новой версией языка ЛЯПАС-М для различных моделей компьютеров. На протяжении всей своей деятельности Н. Р. Торопов был одним из ближайших соратников А. Д. Закревского. Результатом их совместной работы по языку ЛЯПАС-М стала монография «Система программирования ЛЯПАС-М» [1]. О реализации диалоговой системы программирования ЛЕС (ЛЯПАС-М для ЕС ЭВМ) Николай Романович написал отдельную монографию [2] и серию брошюр по математическому обеспечению ЭВМ [3, 4]. В языке ЛЯПАС-М символика была приближена к стандартным алфавитам отечественных устройств отображения информации, включены новые операции над символами и двухмерными логическими массивами, названными комплексами. Созданные программирующие системы ЛЯПАС-М для ЭВМ серии ЕС содержали развитый набор программных средств, достаточных для разработки сложных комплексов автоматизации логического проектирования. Обзор по языку ЛЯПАС и его реализациям был представлен в работе [5].

Николай Романович руководил разработкой не только программных средств системного программирования, но и инструментальных средств логического проектирования дискретных устройств [6], сам разрабатывал эффективные алгоритмы и программы решения оптимизационных комбинаторных задач в области логического проектирования дискретных устройств. Одной из них является классическая задача совместной и раздельной минимизации систем полностью и не полностью определенных булевых функций в классе дизъюнктивных нормальных форм. Описания алгоритмов и программ на языке ЛЯПАС для решения задачи минимизации булевых функций были представлены Н. Р. Тороповым в отдельной главе коллективной монографии «Синтез асинхронных автоматов на ЭВМ» [7], затем он написал эффективные программы минимизации и на языке C++ [8]. Данные программы оказались конкурентоспособными по качеству получаемых решений с лучшими из зарубежных программ и по сей день используются в разработках лаборатории логического проектирования при создании систем автоматизированного проектирования заказных сверхбольших интегральных схем.

Много усилий Н. Р. Торопов направлял на разработку эффективных алгоритмов и программ для решения задач полиномиальных представлений булевых функций. Монография «Полиномиальная реализация частичных булевых функций и систем» [9] была посвящена оптимизации полиномиальных представлений булевых функций (полиномов Жегалкина и Рида – Маллера). Предложенные методы и алгоритмы базировались на теории линейных векторных пространств. Практически все предложенные в книге оригинальные методы оптимизации полиномиальных представлений булевых функций были программно реализованы и испытаны на потоках примеров практической размерности и используются в лаборатории логического проектирования.

Николай Романович разработал эффективные программы решения больших систем логических уравнений (сотни уравнений), но с ограниченным числом переменных в каждом уравнении (порядка десяти), которые нашли практическое применение в криптографии. Появление в ОИПИ НАН Беларуси суперкомпьютеров стимулировало разработку соответствующих программных средств логического проектирования. Н. Р. Торопов принимал активное участие в выполнении программы «Триада» Союзного государства, разрабатывал методы параллельных комбинаторных вычислений, алгоритмы и программы выполнения операций над дизъюнктивными нормальными формами, а также программы решения линейных логических уравнений [10, 11]. С 1989 по 2009 г. он опубликовал более 60 научных работ. Под научным руководством Н. Р. Торопова были защищены кандидатские диссертации Д. И. Черемисинова, Н. А. Кириенко, В. И. Романова и А. А. Прихожего.

Николай Романович Торопов остался в памяти коллег талантливым ученым, высококвалифицированным специалистом в области программирования, трудолюбивым и ответственным научным сотрудником.

Список литературы

1. Закревский, А. Д. Система программирования ЛЯПАС-М / А. Д. Закревский, Н. Р. Торопов. – Минск : Наука и техника, 1978. – 220 с.
2. Торопов, Н. Р. Диалоговая система программирования ЛЕС / Н. Р. Торопов. – Минск : Наука и техника, 1985. – 263 с.
3. Система программирования ЛЕС-4 : инструкт.-метод. материалы / Академия наук БССР, Ин-т техн. кибернетики ; авт.-сост.: Н. Р. Торопов, В. И. Романов ; науч. ред. А. Д. Закревский. – Минск, 1984. – 134 с.
4. Система программирования ЛЕС-7 : материалы по мат. обеспечению ЭВМ / Академия наук БССР, Ин-т техн. кибернетики ; авт.-сост.: Н. Р. Торопов [и др.] ; науч. ред. А. Д. Закревский. – Минск, 1990. – 83 с.
5. Торопов, Н. Р. Алгоритмический язык ЛЯПАС / Н. Р. Торопов // Логическое проектирование : сб. науч. тр. – Минск : Ин-т техн. кибернетики НАН Беларуси, 2001. – Вып. 6. – С. 6–25.
6. Торопов, Н. Р. Исследовательская САПР дискретных управляющих устройств / Н. Р. Торопов // Материалы по математическому обеспечению ЭВМ. – Минск : Ин-т техн. кибернетики АН Беларуси, 1994. – 60 с.
7. Торопов, Н. Р. Минимизация системы булевых функций / Н. Р. Торопов // Синтез асинхронных автоматов на ЭВМ / авт.-сост.: А. Д. Закревский [и др.] ; под общ. ред. А. Д. Закревского. – Минск : Наука и техника, 1975. – Гл. 4. – С. 92–118.
8. Торопов, Н. Р. Минимизация систем булевых функций в классе ДНФ / Н. Р. Торопов // Логическое проектирование : сб. науч. тр. – Минск : Ин-т техн. кибернетики НАН Беларуси, 1999. – Вып. 4. – С. 4–19.
9. Закревский, А. Д. Полиномиальная реализация частичных булевых функций и систем / А. Д. Закревский, Н. Р. Торопов. – Минск : Ин-т техн. кибернетики НАН Беларуси, 2001. – 200 с. ; 2-е изд., стер. – М. : УРСС, 2003. – 200 с.
10. Торопов, Н. Р. Параллельная проверка ДНФ на тавтологию / Н. Р. Торопов // Информатика. – 2005. – № 2(6). – С. 35–42.
11. Торопов, Н. Р. Параллельные логико-комбинаторные вычисления в среде MPI / Н. Р. Торопов // Информатика. – 2005. – № 3(7). – С. 82–90.

П. Н. Бибило, Ю. В. Поттосин

Правила для авторов

Редакция журнала «Информатика» просит авторов руководствоваться приведенными ниже правилами.

I. Статьи принимаются в редакцию через электронную систему подачи по адресу <http://inf.grid.by> в формате файлов текстовых редакторов Microsoft Word. Объем оригинальной статьи – от 8 до 16 стр., включая рисунки, таблицы и достаточное количество наиболее актуальных ссылок; объем обзорной статьи – от 16 до 32 стр., включая все основные ссылки. Текст набирается с переносами, шрифт Times New Roman 11 пт, интервал между строками – одинарный, абзацный отступ 0,5 см, поля по 2,5 см со всех сторон.

Материал статьи должен быть четко структурированным: Введение; основные разделы, в которых изложены цели и задачи, методы, результаты; Заключение (выводы).

II. Статьи о результатах работ, проведенных в научных учреждениях, должны иметь разрешение на публикацию (сопроводительное письмо за подписью руководителя или выписку из заседания ученого совета, отдела или кафедры, акт экспертизы).

III. Статьи в обязательном порядке должны включать аннотацию, ключевые слова, список литературы, информацию об авторах на русском и английском языках.

На титульной странице располагаются следующие метаданные:

1. Индекс по универсальной десятичной классификации (УДК); на русском и английском языках тип статьи (оригинальная или обзорная), название статьи, инициалы и фамилии всех авторов, полное наименование учреждений, где работают авторы, с указанием почтового адреса, при наличии указывается ученая степень и ORCID, e-mail ответственного лица.

2. Аннотация (Abstract) объемом 150–250 слов в оригинальной статье должна быть структурирована отдельными подразделами: Цели, Методы, Результаты, Заключение, а также максимально характеризовать содержательную часть рукописи. Сюда не следует включать впервые введенные термины, аббревиатуры (за исключением общеизвестных), ссылки на литературу.

3. Ключевые слова (Keywords) – наиболее значимые слова или словосочетания по теме работы, отражающие специфику темы, объекты и результаты исследования; перечень ключевых слов должен содержать 5–10 слов.

4. В разделе Благодарности (Acknowledgements) указываются все источники финансирования исследования, а также благодарности людям, которые участвовали в работе над статьей.

5. Автор обязан уведомить редакцию о реальном или потенциальном конфликте интересов, включив информацию в раздел Конфликт интересов (Conflict of interest).

6. Формулы, рисунки, таблицы в статье нумеруются в соответствии с порядком их упоминания в тексте. Ссылки на рисунки и таблицы в тексте обязательны. Рисунки должны быть выполнены с хорошим разрешением в масштабе, позволяющем четко различать надписи и обозначения. Цветные иллюстрации печатаются только в том случае, когда это необходимо для понимания излагаемого материала. Подрисуночные подписи с расшифровкой всех позиций, представленных на рисунке, и названия таблиц набираются шрифтом гарнитуры основного текста размером 9 пт. Перевод подрисуночной подписи и пояснений к рисунку, а также перевод названия таблицы, заголовки строк или столбцов располагаются курсивом после русскоязычной версии.

7. Набор формул выполняется в формульном редакторе Microsoft Equation или Math Type. Прямым шрифтом набираются: греческие и русские буквы; математические символы (\sin , \lg , ∞); символы химических элементов (C, Cl, CH₃); цифры (римские и арабские); индексы (верхние и нижние), являющиеся сокращениями слов. Курсивом набираются латинские буквы, символы физических величин (в том числе и в индексе).

8. Список использованной литературы оформляется в соответствии с требованиями Высшей аттестационной комиссии Республики Беларусь (ГОСТ 7.5–2008). Номер литературной ссылки в тексте дается порядковым номером в квадратных скобках. Ссылаться на неопубликованные работы не допускается.

10. Отдельно оформляется References со следующей структурой: авторы (транслитерация), транслитерированное название монографии, *Перевод названия монографии на английский язык*. Выходные данные с обозначениями на английском языке. От транслитераций названий статей можно отказаться.

Ссылки на учебно-методическую литературу, ГОСТы, авторефераты, статистические отчеты в список не включаются, а оформляются в виде сносок (с подробными рекомендациями можно ознакомиться на сайте журнала в разделе Правила для авторов).

11. В разделе Информация об авторах (Information about the authors) приводятся ФИО авторов полностью, ученая степень, звание, должность, название организации, ORCID (при наличии).

IV. Все поступающие в редакцию рукописи проходят предварительную проверку на соответствие Правилам для авторов. Статья может быть возвращена автору на доработку с просьбой устранить недостатки или дополнить информацию. После проверки на соответствие правилам статья направляется рецензенту с указанием сроков рецензирования.

V. При наличии замечаний рецензента автору предоставляется определенное время на доработку рукописи. Статьи, направляемые на доработку, должны быть возвращены в исправленном виде с ответами на все замечания. Окончательное решение о публикации или отклонении рукописи принимается редколлегией журнала. При положительном заключении рецензента статья передается редактору для подготовки к печати. Редакция оставляет за собой право на редакционные изменения, не искажающие основное содержание статьи.

VI. Редакция журнала предоставляет возможность первоочередного опубликования статей, представленных лицами, которые осуществляют послевузовское обучение (аспирантура, докторантура, соискательство) в год завершения обучения.

VII. Авторы несут ответственность за направление в редакцию статей, уже опубликованных ранее или принятых к публикации другими изданиями.

ИНДЕКСЫ

00827

для индивидуальных
подписчиков

008272

для предприятий
и организаций