

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

ИНФОРМАТИКА

INFORMATICS

TOM 20
VOL.

1 | 2023

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 for authors 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

Informatika

Том 20, № 1, январь-март 2023

Ежеквартальный научный журнал

Издается с января 2004 г.

Учредитель и издатель – государственное научное учреждение «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., чл.-корр. НАН Беларуси,
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., чл.-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., БГУИР (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, ОИПИ НАН Беларуси
(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., академик НАН Беларуси, НИИ ППМИ БГУ
(Минск, Беларусь)

Черемисинова Людмила Дмитриевна, д-р техн. наук, проф., ОИПИ НАН Беларуси (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 20, № 1, январь-март 2023

Ответственный за выпуск *Мойсейчик Светлана Сергеевна*

Редактор *Гончаренко Галина Борисовна*

Корректор *Михайлова Анна Антоновна*

Компьютерная верстка *Бутевич Ольга Борисовна*

Сдано в набор 13.02.2023. Подписано в печать 20.03.2023. Формат 60×84 1/8. Бумага офсетная. Гарнитура Таймс. Ризография. Усл. печ. л. 13,9. Уч.-изд. л. 13,7. Тираж 40 экз. Заказ 1.

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.

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

INFORMATICS

Vol. 20, no. 1, January-March 2023

Published quarterly

Issued since January 2004

Founder and publisher – State Scientific Institution "The United Institute of Informatics
Problems of the National Academy of Sciences of Belarus" (UIIP NASB)

Editor-in-Chief

Alexander V. Tuzikov, D. Sc. (Phys.-Math.), Prof., Corr. Member of NASB,
UIIP NASB (Minsk, Belarus)

Deputy Editor-in-Chief

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

Editorial Board

Sergey V. Ablameyko, D. Sc. (Eng.), Prof., Academician of NASB, BSU (Minsk, Belarus)

Uladimir V. Anishchanka, Ph. D. (Eng.), Assoc. Prof., SoftClub Ltd. (Minsk, Belarus)

Petr N. Bibilo, D. Sc. (Eng.), Prof., UIIP NASB (Minsk, Belarus)

Mikhail N. Bobov, D. Sc. (Eng.), Prof., BSUIR (Minsk, Belarus)

Alexandre B. Dolgui, D. Sc. (Eng.), Prof., IMT Atlantique (Nantes, France)

Alexander N. Dudin, D. Sc. (Phys.-Math.), Prof., BSU (Minsk, Belarus)

Alexey A. Karpov, D. Sc. (Eng.), Assoc. Prof., SPII RAS (Saint Petersburg, Russia)

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

Viktor V. Krasnoproshin, D. Sc. (Eng.), Prof., BSU (Minsk, Belarus)

Alexander M. Krot, D. Sc. (Eng.), Prof., UIIP NASB (Minsk, Belarus)

Sergey V. Kruglikov, D. Sc. (Milit.), Ph. D. (Eng.), Assoc. Prof., UIIP NASB (Minsk, Belarus)

Nikolai A. Likhoded, D. Sc. (Phys.-Math.), Prof., BSU (Minsk, Belarus)

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

Valery A. Sklyarov, D. Sc. (Eng.), Prof., University of Aveiro (Aveiro, Portugal)

Yuri N. Sotskov, D. Sc. (Phys.-Math.), Prof., UIIP NASB (Minsk, Belarus)

Alexander L. Stempkovsky, D. Sc. (Eng.), Prof., Academician of RAS, IPPM RAS (Moscow, Russia)

Yuriy S. Kharin, D. Sc. (Phys.-Math.), Prof., Academician of NASB, RI APMI BSU (Minsk, Belarus)

Ljudmila D. Cheremisinova, D. Sc. (Eng.), Prof., UIIP NASB (Minsk, Belarus)

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

Vyacheslav N. Yarmolik, D. Sc. (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. 20, no. 1, January-March 2023

Issue Head *Sviatlana S. Maiseichyk*

Editor *Halina B. Hancharenka*

Corrector *Hanna A. Mikhailava*

Computer Imposition *Volha B. Butsevich*

Sent for press 13.02.2023. Output 20.03.2023. Format 60×84 1/8. Offset paper. Headset Times. Riesography. Printed sheets 13,9. Publisher's signatures 13,7. Circulation 40 copies. Order 1.

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

ЗАЩИТА ИНФОРМАЦИИ И НАДЕЖНОСТЬ СИСТЕМ

Ярмолик В. Н., Иванюк А. А. Двухмерные физически неклонированные функции типа арбитр..... 7

БИОИНФОРМАТИКА

Красько О. В. Оценка размера эффекта воздействия в квазиэкспериментальных исследованиях..... 27

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

Игнатъева С. А., Богущ Р. П. Увеличение точности реидентификации людей на основе двухэтапного обучения сверточных нейронных сетей и аугментации 40

Бегунков В. И., Ковалев М. Я. Классификация займов с использованием логистической регрессии..... 55

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

Погтосин Ю. В. Совместное энергосберегающее кодирование состояний последовательных автоматов сети, реализующей параллельный автомат..... 75

Бутов А. А. Аппаратная реализация булевых функций на основе автоматной модели 91

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Лихачёв Д. С., Вашкевич М. И., Петровский Н. А., Азаров И. С. Малоразмерные спектральные признаки для машинного обучения в задачах анализа и классификации голосового сигнала..... 102

УЧЕНЫЕ БЕЛАРУСИ

Герт Иванович Алексеев: ученый и изобретатель 113

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

CONTENTS

INFORMATION PROTECTION AND SYSTEM RELIABILITY

Yarmolik V. N., Ivaniuk A. A. 2D physically unclonable functions of the arbiter type 7

BIOINFORMATICS

Krasko O. V. Effect size assessment in quasi-experimental studies..... 27

INTELLIGENT SYSTEMS

Ihnatsyeva S. A., Bohush R. P. Improving person re-identification based on two-stage training of convolutional neural networks and augmentation 40

Behunkou U. I., Kovalyov M. Y. Loan classification using logistic regression..... 55

LOGICAL DESIGN

Pottosin Yu. V. Joint low power state assignment of sequential automata of a net implementing a parallel automaton 75

Butov A. A. Hardware implementation of Boolean functions based on the automaton model..... 91

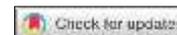
INFORMATION TECHNOLOGIES

Likhachov D. S., Vashkevich M. I., Petrovsky N. A., Azarov E. S. Small-size spectral features for machine learning in voice signal analysis and classification tasks..... 102

SCIENTISTS OF BELARUS

Gert Ivanovich Alekseev: scientist and inventor 113

ЗАЩИТА ИНФОРМАЦИИ И НАДЕЖНОСТЬ СИСТЕМ INFORMATION PROTECTION AND SYSTEM RELIABILITY



УДК 004.832.32
<https://doi.org/10.37661/1816-0301-2023-20-1-7-26>

Оригинальная статья
Original Paper

Двухмерные физически неклонлируемые функции типа арбитр

В. Н. Ярмолик[✉], А. А. Иванюк

Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
[✉]E-mail: yarmolik10ru@yahoo.com

Аннотация

Цели. Решается задача построения нового класса физически неклонлируемых функций типа арбитр (АФНФ), основанного на различии задержек по входам многочисленных модификаций базового элемента путем увеличения как количества входов, так и топологии их подключения. Подобный подход позволяет строить двухмерные физически неклонлируемые функции (2D-АФНФ), в которых в отличие от классических АФНФ запрос, формируемый для каждого базового элемента, выбирает пару путей не из двух возможных, а из большего их количества. Актуальность данного исследования связана с активным развитием физической криптографии. В работе преследуются следующие цели: построение базовых элементов АФНФ и их модификаций, разработка методики построения 2D-АФНФ.

Методы. Используются методы синтеза и анализа цифровых устройств, в том числе на программируемых логических интегральных схемах, основы булевой алгебры и схемотехники.

Результаты. Показано, что в классических АФНФ применяется стандартный базовый элемент, выполняющий две функции, а именно функцию выбора пары путей *Select* и функцию переключения путей *Switch*, которые за счет их совместного использования позволяют достичь высоких характеристик. В первую очередь это касается стабильности функционирования АФНФ, характеризующейся небольшим числом запросов, для которых ответ случайным образом принимает одно из двух возможных значений: 0 или 1. Предложены модификации базового элемента в части реализаций его функций *Select* и *Switch*. Приводятся новые структуры базового элемента с внесенными модификациями их реализаций, в том числе в части увеличения количества пар путей базового элемента, из которых путем запроса выбирается одна из них и конфигурации их переключений. Применение разнообразных базовых элементов позволяет улучшать основные характеристики АФНФ, а также нарушать регулярность их структуры, которая является главной причиной взлома АФНФ путем машинного обучения.

Заключение. Предложенный подход к построению 2D-АФНФ, основанный на различии задержек сигналов через базовый элемент, показал свою работоспособность и перспективность. Экспериментально подтвержден эффект улучшения характеристик подобных ФНФ, и в первую очередь стабильности их функционирования. Перспективным представляется дальнейшее развитие идеи построения 2D-АФНФ, экспериментальное исследование их характеристик и устойчивости к различного рода атакам, в том числе с использованием машинного обучения.

Ключевые слова: физическая криптография, физически неклонлируемые функции, физические однонаправленные функции, физически неклонлируемые функции типа арбитр

Благодарности. Авторы выражают искреннюю благодарность резиденту ПВТ компании «СК хайникс мемори солишенс Восточная Европа» за предоставленное оборудование для проведения экспериментов в рамках работы совместной учебной лаборатории с Белорусским государственным университетом информатики и радиоэлектроники.

Для цитирования. Ярмолик, В. Н. Двухмерные физически неклонлируемые функции типа арбитр / В. Н. Ярмолик, А. А. Иванюк // Информатика. – 2023. – Т. 20, № 1. – С. 7–26.
<https://doi.org/10.37661/1816-0301-2023-20-1-7-26>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 11.10.2022
Подписана в печать | Accepted 21.11.2022
Опубликована | Published 29.03.2023

2D physically unclonable functions of the arbiter type

Vyacheslav N. Yarmolik[✉], Alexander A. Ivaniuk

*Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus
✉E-mail: yarmolik10ru@yahoo.com*

Abstract

Objectives. The problem of constructing a new class of physically unclonable functions of the arbiter type (APUF) is being solved, based on the difference in delay times for the inputs of numerous modifications of the base element, due to both an increase in the number of inputs and the topology of their connection. Such an approach allows building two-dimensional physically unclonable functions (2D-APUF), in which, unlike classical APUF, the challenge generated for each basic element selects a pair of paths not from two possible, but from a larger number of them. The relevance of such a study is associated with the active development of physical cryptography. The following goals are pursued in the work: the construction of the basic elements of the APUF and their modifications, the development of a methodology for constructing 2D-APUF.

Methods. The methods of synthesis and analysis of digital devices are used, including those based on programmable logic integrated circuits, the basics of Boolean algebra and circuitry.

Results. It is shown that the classical APUF uses a standard basic element that performs two functions, namely, the function of choosing a pair of paths *Select* and the function of switching paths *Switch*, which, due to their joint use, allow achieving high performance. First of all, this concerns the stability of the APUF functioning, which is characterized by a small number of challenge, for which the response randomly takes one of two possible values 0 or 1. Modifications of the base element in terms of the implementations of its *Select* and *Switch* functions are proposed. New structures of the base element are presented in which the modifications of their implementations are made, including in terms of increasing the number of pairs of paths of the base element from which one of them is selected by the challenge, and the configurations of their switching. The use of various basic elements makes it possible to improve the main characteristics of APUF, as well as to break the regularity of their structure, which was the main reason for hacking APUF through machine learning.

Conclusion. The proposed approach to the construction of physically unclonable 2D-APUF functions, based on the difference in signal delays through the base element, has shown its efficiency and promise. The effect of improving the characteristics of such PUFs has been experimentally confirmed with noticeable improvement in the stability of their functioning. It seems promising to further develop the ideas of constructing two-dimensional physically unclonable functions of the arbiter type, as well as experimental study of their characteristics, as well as resistance to various types of attacks, including using machine learning.

Keywords: physical cryptography, physically unclonable functions, physical one-way functions, physically unclonable arbiter-type functions

Acknowledgements. The authors express their sincere gratitude to the HTP resident of the "SK hynix memory solutions Eastern Europe" company for the equipment provided for carrying out experiments within the framework of the joint laboratory with the Belarusian State University of Informatics and Radioelectronics.

For citation. Yarmolik V. N., Ivaniuk A. A. *2D physically unclonable functions of the arbiter type*. Informatika [Informatics], 2023, vol. 20, no. 1, pp. 7–26 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-1-7-26>

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

Введение. Физически неклонлируемые функции (Physical Unclonable Functions, PUF) предлагают многообещающие решения для настоящих и будущих задач информационной безопасности [1, 2]. Изначально PUF предназначались для защиты от появления поддельных нелегальных цифровых устройств; по сути, – для защиты авторских прав на цифровые устройства [3, 4]. В настоящее время сфера применения PUF значительно расширилась за счет их активного применения в криптографии для целей идентификации, аутентификации, генерирования криптографических ключей, а также реализации различных криптографических протоколов [5–8].

Первоначально PUF назывались физические однонаправленные функции (Physical One-Way Functions, POWF) или физические случайные функции (Physical Random Functions, PRF) [1, 2]. Несмотря на то что последние два определения были сформулированы исторически раньше, в настоящее время в основном употребляется название PUF, которое в русскоязычной литературе представляется как *физически неклонлируемые функции* (ФНФ) [9–11]. Идея ФНФ впервые была представлена Р. Паппу (R. Pappu) в его пионерской работе [1], где он впервые сделал попытку сформулировать основные понятия и определения в данной области.

Одно из наиболее широко используемых на сегодняшний день определений ФНФ было предложено П. Туилсом (P. Tuyls) [3] как обобщение и развитие работ Р. Паппу. Физически неклонлируемые функции, по его определению, – это физические системы, неотъемлемым свойством которых является неклонлируемость, т. е. невозможность воспроизведения двух идентичных ФНФ. У таких систем свойство неклонлируемости обусловлено тем фактом, что они состоят из множества компонентов, параметры которых в процессе создания подобных физических систем принимают случайные значения. Наличие компонентов со случайными величинами их параметров и характеристик, а также невозможность контролировать и управлять этими параметрами элементов ФНФ во время производства делают их уникальными и физически неклонлируемыми. ФНФ описываются значениями входных и соответствующих им выходных параметров (сигналов). Подобная пара, состоящая из входного физического параметра *запроса* (Challenge, C) и выходного параметра *ответа* (Response, R), называется парой «запрос-ответ» (Challenge-Response Pair, CRP). В простейшем случае ФНФ можно рассматривать как функцию $R = F(C)$, которая преобразует запросы C в ответы R [3, 9].

Исчерпывающее определение ФНФ как системы со сверхбольшим объемом информации (Super High Information Content, SHIC) было предложено У. Рухрмаером (U. Rührmair) [9, 12].

Физически неклонлируемые функции представляют собой сложные неупорядоченные физические системы с чрезвычайно большим объемом структурной информации, которые удовлетворяют следующим требованиям:

1. Структурная информация подобных систем может быть извлечена надежно и многократно путем проведения измерений для различных запросов C и получения ответов R .

2. Количество возможных запросов C должно быть настолько велико, что значения всех соответствующих ответов R не могут быть получены путем перебора всех возможных запросов C за реальный временной промежуток.

3. Ввиду наличия в системе чрезвычайно большого объема структурной информации должно быть невозможным моделировать, рассчитывать, эмулировать или каким-либо другим математическим способом предсказывать пару «запрос-ответ» (C_j, R_j), зная другую пару (C_i, R_i) или некоторое множество таких пар.

4. Для физической системы с чрезвычайно большим объемом структурной информации должно быть чрезвычайно сложным ее физическое воспроизведение или клонирование (повторение) как аналогичной физической системы, описываемой идентичным множеством пар «запрос-ответ».

Основные проблемы при создании ФНФ заключаются в противоречии первого требования, которое характеризует стабильность их функционирования, и третьего требования о непредсказуемости, случайности таких функций. Попытка повысить стабильность ФНФ увеличивает их уязвимость для различного рода атак, в особенности с применением современных достижений машинного обучения [13–15].

Для цифровых устройств фундаментальным подходом при реализации большинства разновидностей ФНФ является создание такого цифрового устройства ФНФ, выходное значение которого определяется случайными значениями временных параметров, чаще всего задержек электрических сигналов. Из-за технологических вариаций во время изготовления произвольного цифрового устройства время задержки сигналов по определенному его пути будет незначительно меняться от цифрового устройства к цифровому устройству и от кристалла к кристаллу, несмотря на идентичность их топологии и функциональности [16–18].

Исторически первой ФНФ является АФНФ (Arbiter PUF), предложенная в 2002 г. [2]. АФНФ оказалась удачным решением, основанным на различии задержек прохождения сигнала через цифровые элементы. При этом различие задержек стало основным фактором, влияющим на свойства подобной ФНФ.

Развивая идею различия задержки прохождения сигнала через элемент, в настоящей работе предлагается использовать для построения новых структур АФНФ отличия задержек сигнала по входам цифрового элемента. Показывается, что как сам цифровой элемент, так и каждый его вход уникальны и неповторимы в части задержки прохождения сигнала на выходе элемента. В статье предлагаются новые структурные решения для базового элемента АФНФ, характеризующиеся отличающимся набором значений временных задержек и их количеством, что позволило улучшить основные характеристики АФНФ, и в первую очередь стабильность ее функционирования. Впервые рассматривается идея построения 2D-АФНФ (2D-APUF).

1. ФНФ типа арбитра. В общем случае при реализации АФНФ изготавливаются два функционально и топологически идентичные электрические пути, представляющие собой последовательно подключенные элементы и их межсоединения. Очевидно, что оба пути будут иметь близкие значения величин задержек распространения по ним сигналов, однако они будут принципиально разными в силу технологических вариаций в процессе производства. Процедура измерения времени распространения сигналов заключается в одновременной подаче на входы обоих путей сигнала и определении того из них, на выходе которого сигнал появится быстрее. Пары симметричных путей задержки электрического сигнала изготавливаются таким образом, что одновременно из большого множества пар выбирается одна пара за счет формирования определенного запроса C . Далее для выбранной пары путей определяется, какой из них более быстрый, и формируется ответ R . Непредсказуемость и случайность ответов для конкретной реализации АФНФ служат фундаментальной основой подобных функций. В то же время каждая реализация АФНФ обеспечивает стабильность, т. е. повторяемость ответов на одни и те же запросы, которые не повторяют их значения для других реализаций функционально и топологически идентичных АФНФ.

Классическая схема АФНФ (рис. 1) [6] строится с использованием n последовательно подключенных пар двухвходовых мультиплексоров (MUX). Адресные входы (Adr) обоих мультиплексоров MUX_{1j} и MUX_{2j} каждой пары объединяются и применяются в качестве одного из входов для задания значения одного бита запроса c_j . Запросом в данном случае выступает n -разрядный двоичный вектор $C_i = c_0 c_1 c_2 \dots c_{n-1}$, где $c_j \in \{0, 1\}$, $j \in \{0, 1, 2, \dots, n-1\}$. Запрос C_i в схеме АФНФ формирует два пути таким образом, что если $c_j = 0$ для j -й ступени АФНФ, то для построения первого пути используется верхний мультиплексор MUX_{1j} , а для построения второго пути – нижний MUX_{2j} ; если $c_j = 1$, то наоборот. Каждая пара путей имеет общий вход, а выходы первого и второго путей подключены соответственно к входу D D -триггера и его синхронизирующему входу Clk . В данном случае D -триггер является арбитром и перед проведением эксперимента устанавливается в исходное нулевое состояние. Для конкретного запроса C_i конфигурируется уникальная пара путей и генерируется ответ $R_i \in \{0, 1\}$ как результат эксперимента по определению того, по какому из путей (первому или второму) задержка входного

импульсного сигнала меньше. Например, если по первому пути, то $R_i = 1$, а если по второму, то $R_i = 0$. Очевидно, что количество пар путей с увеличением n растет экспоненциально и равняется 2^n .

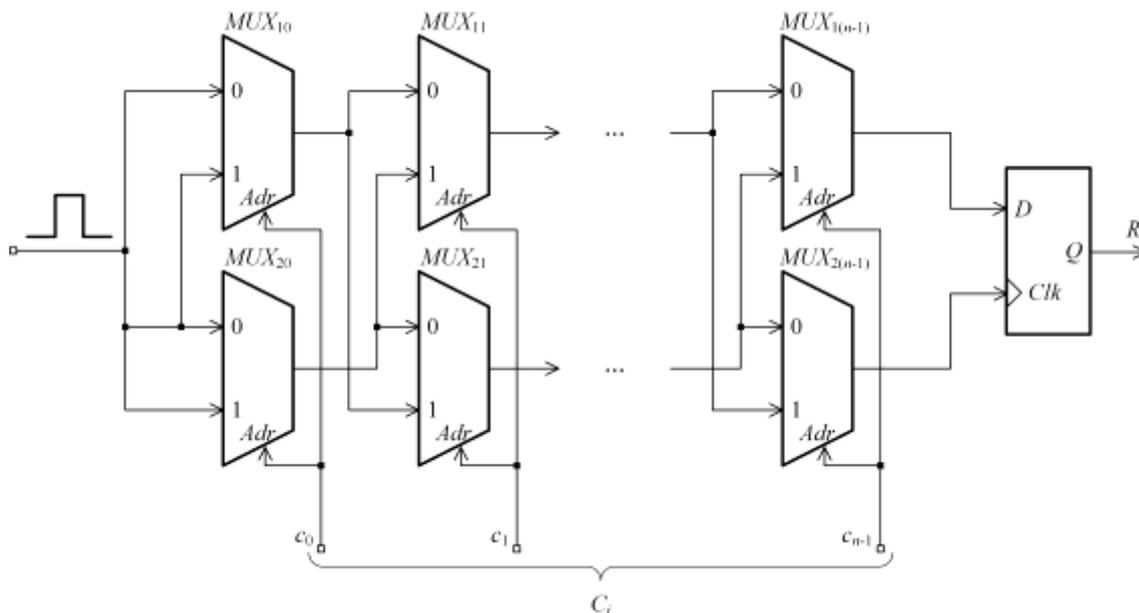


Рис. 1. АФНФ на базе двухвходовых мультиплексоров
 Fig. 1. APUF based on two-input multiplexers

Все известные решения построения ФНФ, в том числе и АФНФ, основаны на том, что задержка по конкретному пути (элементу) имеет случайное значение, определяемое множеством факторов, которые влияют на ее величину в процессе производства ФНФ. Однако у реальных ФНФ эти случайные задержки имеют неизменное и неуправляемое значение, исключая влияние внешних факторов (температуры, давления, электромагнитного излучения и др.) и временной деградации. Их неизменность, с одной стороны, обуславливает стабильность функционирования ФНФ, а с другой – открывает возможности для различного рода атак на ФНФ по предсказанию либо описанию их поведения.

Базируясь исключительно на детерминированном поведении АФНФ, чаще всего и строят различные их математические модели. Наиболее распространенная из них основана на том, что каждая j -я ступень АФНФ, состоящая из пары мультиплексоров MUX_{1j} и MUX_{2j} , может быть описана двумя параметрами, а именно разностями задержек $\delta_{0,j}$ и $\delta_{1,j}$. Разность задержки $\delta_{0,j}$ для j -й ступени АФНФ определяется как добавленная разность задержек прохождения сигнала по двум путям через MUX_{1j} и MUX_{2j} при $c_j = 0$, а разность задержки $\delta_{1,j}$ – при $c_j = 1$. Если эти два параметра известны для каждой ступени АФНФ, окончательную разницу задержки для каждой пары путей можно легко определить, если учесть возможный эффект переключения на каждой ступени. Переключение одного пути на j -й ступени АФНФ с MUX_{1j} на MUX_{2j} , а другого пути с MUX_{2j} на MUX_{1j} эквивалентно изменению знака разницы задержек сигналов пары путей на предыдущих ступенях АФНФ. Соответственно, разница задержки d_j после j -й ступени может вычисляться рекурсивно:

$$d_j = d_{j-1} \cdot (-1)^{c_j} + \delta_{c_j, j}. \quad (1)$$

Анализ соотношения (1) показывает, что ответ R_i на запрос C_i для АФНФ, представленной на рис. 1, определяется знаком разницы задержек d_{n-1} импульсного сигнала по выбранным путям в соответствии с запросом C_i . В рамках подобных моделей описания АФНФ определяющим фактором являются величины добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$, от которых зависят

основные свойства и характеристики АФНФ. В свою очередь, эти величины для j -й ступени АФНФ (рис. 1) вычисляются согласно соотношениям

$$\delta_{0,j} = \Delta(0)_{1,j} - \Delta(0)_{2,j}, \quad \delta_{1,j} = \Delta(1)_{1,j} - \Delta(1)_{2,j}. \quad (2)$$

Численное значение $\Delta(0)_{1,j}$ определяет временную задержку прохождения сигнала с нулевого входа, обозначенного символом 0, на первом мультиплексоре (MUX_{1j}) j -й ступени АФНФ на его выход, а значение $\Delta(0)_{2,j}$ – задержку на втором мультиплексоре (MUX_{2j}). Величины $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ представляют собой задержки сигналов по единичным входам соответствующих мультиплексоров. Отметим, что все четыре значения, а именно $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$, и являются источниками непредсказуемости поведения АФНФ (см. рис. 1). Все четыре величины принимают случайные значения в результате влияния множества случайных факторов на процесс изготовления АФНФ и конкретно j -й его ступени. При функционировании АФНФ эти величины в идеальном случае имеют неизменные значения и участвуют в определении величин добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$ согласно соотношениям (2). Под неизменностью указанных величин понимаются такие изменения их значений, которые не нарушают повторяемость ответов для одного и того же запроса. Далее результат R_i функционирования АФНФ при подаче запроса C_i определяется комбинацией (1) величин разности задержек $\delta_{c_j,j}$. Таким образом, для получения ответа используются только две случайные величины $\delta_{0,j}$ и $\delta_{1,j}$, каждая из которых представляет собой разность двух из четырех исходных, предварительно сгенерированных случайных значений $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$.

Для АФНФ, представленной на рис. 1, так же, как и для большинства подобных известных решений, базовым элементом является схема, которая состоит из двух двухвходовых мультиплексоров MUX_{1j} и MUX_{2j} . Копирование таких последовательно соединенных схем и лежит в основе создания АФНФ. Отметим, что в процессе производства j -го базового элемента формируются фиксированные значения задержек сигнала по всем его четырем входам. В силу различного рода случайных факторов величины этих задержек $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ принимают случайные значения. В идеальном случае указанные задержки сохраняют свои значения в процессе функционирования АФНФ. На этой гипотезе для обеспечения стабильности и повторяемости ответов на генерируемые запросы и основан принцип их функционирования. От сочетания значений задержек $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ зависят основные свойства АФНФ, которые следуют из их математического описания (1). Указанная модель позволяет убедиться в эффективности классической структуры АФНФ при различных сочетаниях задержек. Даже в случае весьма крайних и маловероятных соотношений задержек $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ существуют множества запросов, обеспечивающих устойчивое функционирование АФНФ.

В качестве примера рассмотрим классическую АФНФ для $n = 4$ (рис. 2).

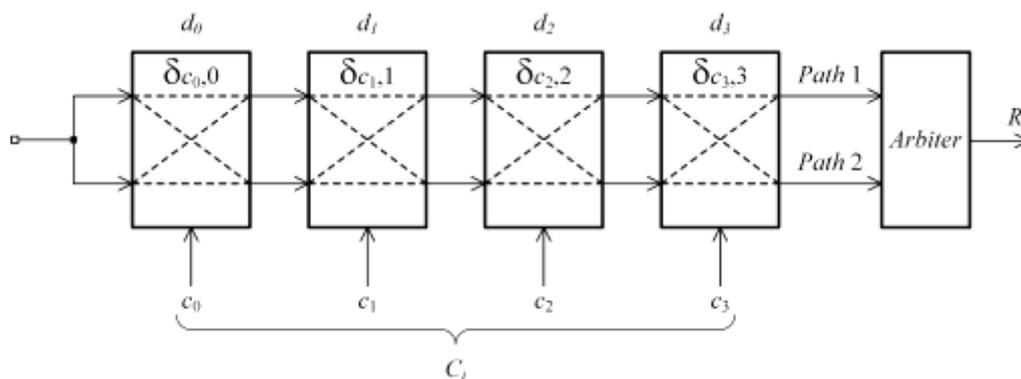


Рис. 2. АФНФ для $n = 4$

Fig. 2. APUF for $n = 4$

На каждой j -й, $j \in \{0, 1, 2, 3\}$, ступени АФНФ в зависимости от значения c_j запроса C_i формируются задержки распространения сигнала по путям *Path 1* и *Path 2* выбранной запросом пары путей (см. рис. 2). Соотношение этих задержек на каждой ступени определяется величиной их добавленной разности $\delta_{c_j, j}$, а значение разности задержки сигнала по пути *Path 2* по отношению к задержке по пути *Path 1* – величиной d_j (1). Знак плюс либо минус финального значения d_3 разности задержек и определяет значение ответа $R_i \in \{0, 1\}$.

В качестве двух реализаций АФНФ, показанной на рис. 2, рассмотрим АФНФ₁ и АФНФ₂, имеющие фиксированные задержки сигналов $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ и соответствующие им величины добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$. В табл. 1 приведены соотношения задержек $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ для обеих реализаций и численные значения добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$ в условных единицах, кратных абстрактной величине задержки d . Первая реализация АФНФ₁ в упрощенном виде представляет собой аналог стандартной реализации АФНФ, когда разности задержек каждой ступени принимают произвольные значения. Из табл. 1 видно, что абсолютные величины задержек разнятся так же, как и знаки этих разностей, принимающие оба значения – и плюс, и минус.

Таблица 1
Значения задержек сигналов для АФНФ₁ и АФНФ₂

Table 1
Signal delay value for АФНФ₁ and АФНФ₂

АФНФ ₁ АФНФ ₁	$\Delta(0)_{1,0} > \Delta(0)_{2,0}$	$\Delta(0)_{1,1} > \Delta(0)_{2,1}$	$\Delta(0)_{1,2} > \Delta(0)_{2,2}$	$\Delta(0)_{1,3} < \Delta(0)_{2,3}$
	$\delta_{0,0} = d$	$\delta_{0,1} = d$	$\delta_{0,2} = 2d$	$\delta_{0,3} = -d$
	$\Delta(1)_{1,0} < \Delta(1)_{2,0}$	$\Delta(1)_{1,1} > \Delta(1)_{2,1}$	$\Delta(1)_{1,2} < \Delta(1)_{2,2}$	$\Delta(1)_{1,3} > \Delta(1)_{2,3}$
	$\delta_{1,0} = -d$	$\delta_{1,1} = 2d$	$\delta_{1,2} = -2d$	$\delta_{1,3} = 2d$
АФНФ ₂ АФНФ ₂	$\Delta(0)_{1,0} > \Delta(0)_{2,0}$	$\Delta(0)_{1,1} > \Delta(0)_{2,1}$	$\Delta(0)_{1,2} > \Delta(0)_{2,2}$	$\Delta(0)_{1,3} > \Delta(0)_{2,3}$
	$\delta_{0,0} = d$	$\delta_{0,1} = d$	$\delta_{0,2} = d$	$\delta_{0,3} = d$
	$\Delta(1)_{1,0} > \Delta(1)_{2,0}$	$\Delta(1)_{1,1} > \Delta(1)_{2,1}$	$\Delta(1)_{1,2} > \Delta(1)_{2,2}$	$\Delta(1)_{1,3} > \Delta(1)_{2,3}$
	$\delta_{1,0} = d$	$\delta_{1,1} = d$	$\delta_{1,2} = d$	$\delta_{1,3} = d$

Функция АФНФ₂ приведена в качестве примера весьма неудачного, аномального случая синтеза АФНФ. АФНФ₂ является результатом изготовления АФНФ, когда из-за вариаций производственного процесса задержки $\Delta(0)_{1,j}$ и $\Delta(1)_{1,j}$ мультиплексоров MUX_{1j} всех $n = 4$ ступеней АФНФ₂ оказались больше задержек мультиплексоров MUX_{2j} . Следует отметить реальность такой ситуации в технологических процессах изготовления подобных функций, особенно при реализации АФНФ на программируемых структурах [19–21]. Соответственно, все величины добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$ согласно (2) примут положительные значения. Еще более усугубляя этот аномальный случай для АФНФ, предположим, что для всех четырех ступеней АФНФ₂ $\delta_{0,j} = \delta_{1,j} = d$.

Полное описание функционирования АФНФ₁ и АФНФ₂ для $n = 4$ и определенных для них величин добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$ дано в табл. 2. Символ X в табл. 2 означает метастабильное состояние АФНФ, которое относится к нежелательному ее поведению [17, 22]. Этот случай имеет место для запросов C_i , ответом на которые является равенство нулю задержки d_3 . Например, для АФНФ₁ и запроса $C_9 = c_0 c_1 c_2 c_3 = 1 0 0 1$ распространение сигнала по двум путям *Path 1* и *Path 2* в терминах разницы задержки d_i (1) описывается следующими преобразованиями:

$$d_0 = 0 \cdot (-1)^1 + (-d) = -d; d_1 = (-d) \cdot (-1)^0 + d = 0; d_2 = 0 \cdot (-1)^0 + 2d = 2d; d_3 = (2d) \cdot (-1)^1 + 2d = 0.$$

Равенство нулю значения d_3 приводит к тому, что при повторении запроса $C_i = 1 0 0 1$ АФНФ₁ будет генерировать случайное значение ответа $R_i \in \{0, 1\}$. Это и есть метастабильное

поведение АФНФ₂. На практике метастабильное состояние и соответствующие ему запросы C_i являются весьма нежелательными, так как нарушают стабильность функционирования АФНФ [17, 22].

Таблица 2
Описание функционирования АФНФ₁ и АФНФ₂
Table 2
Description of functioning APUF₁ and APUF₂

i	C_i				АФНФ ₁ APUF ₁					АФНФ ₂ APUF ₂				
					d_j				R_i	d_j				R_i
	c_0	c_1	c_2	c_3	d_0	d_1	d_2	d_3		d_0	d_1	d_2	d_3	
0	0	0	0	0	d	$2d$	$4d$	$3d$	1	d	$2d$	$3d$	$4d$	1
1	0	0	0	1	d	$2d$	$4d$	$-2d$	0	d	$2d$	$3d$	$-2d$	0
2	0	0	1	0	d	$2d$	$-4d$	$-5d$	0	d	$2d$	$-d$	0	X
3	0	0	1	1	d	$2d$	$-4d$	$6d$	1	d	$2d$	$-d$	$2d$	1
4	0	1	0	0	d	d	$3d$	$2d$	1	d	0	d	$2d$	1
5	0	1	0	1	d	d	$3d$	$-d$	0	d	0	d	0	X
6	0	1	1	0	d	d	$-3d$	$-4d$	1	d	0	d	$2d$	1
7	0	1	1	1	d	d	$-3d$	$5d$	0	d	0	d	0	X
8	1	0	0	0	$-d$	0	$2d$	d	1	d	$2d$	$3d$	$4d$	1
9	1	0	0	1	$-d$	0	$2d$	0	X	d	$2d$	$3d$	$-2d$	0
10	1	0	1	0	$-d$	0	$-2d$	$-3d$	0	d	$2d$	$-d$	0	X
11	1	0	1	1	$-d$	0	$-2d$	$4d$	1	d	$2d$	$-d$	$2d$	1
12	1	1	0	0	$-d$	$3d$	$5d$	$4d$	1	d	0	d	$2d$	1
13	1	1	0	1	$-d$	$3d$	$5d$	$-3d$	0	d	0	d	0	X
14	1	1	1	0	$-d$	$3d$	$-5d$	$-6d$	0	d	0	d	$2d$	1
15	1	1	1	1	$-d$	$3d$	$-5d$	$7d$	1	d	0	d	0	X

Главный вывод, который можно сделать в результате анализа данных в табл. 2, касается высокой эффективности АФНФ за счет того, что базовый элемент выполняет в том числе и функцию переключения *Switch* путей j -й ступени АФНФ, одного с MUX_{1j} на MUX_{2j} , а другого с MUX_{2j} на MUX_{1j} . Переключение путей позволяет нивелировать асимметрию задержек двух путей, вызванную аномальными отличиями характеристик элементов, которые реализуют АФНФ. Этот факт подтверждается примером АФНФ₂, которая, несмотря на детерминированное отклонение задержек ее элементов, в принципе может рассматриваться в качестве рабочей версии АФНФ. Отметим, что реализация АФНФ₂ по той же методологии выбора пары путей, но на основе базового элемента без функции переключения путей, как в работе [23] на базе буфера с тремя состояниями, привела бы к созданию неработоспособной версии АФНФ. Реализация АФНФ₂ с использованием двух отдельных множеств путей, когда первый путь пары строится из первого множества элементов, а второй путь – из второго множества, привела бы к получению неизменного ответа на любой запрос АФНФ₂.

Наличие различного рода асимметричных аномалий в большей мере присуще реализациям АФНФ на программируемой логике типа FPGA. Показано, что асимметрия задержки АФНФ, реализованной на FPGA, из-за асимметрии маршрутизации более чем в 10 раз выше, чем случайная ее вариация из-за производственного процесса [19]. Однако, по мнению авторов, большинство существующих реализаций АФНФ на FPGA позволяет достичь неплохих результатов только за счет удачного выбора базового элемента, состоящего из двух мультиплексоров, и в большей мере за счет его функции переключения путей.

В то же время однородность структуры АФНФ, состоящей из последовательно подключенных одинаковых базовых элементов, является ее основным недостатком и позволяет проводить различного рода атаки по их взлому [13, 14].

2. Модификации базового элемента АФНФ. Классическая структура АФНФ строится на основе базового элемента из двух двухвходовых мультиплексоров MUX_{1j} и MUX_{2j} , подробно описанного в разд. 1. Базовый элемент имеет два входа и два выхода, что позволяет путем последовательного подключения n подобных элементов строить основную структуру АФНФ, отвечающую за выбор пары путей из 2^n возможных пар. Уникальность такого базового элемента заключается в простоте его аппаратной реализации и достаточно высокой эффективности, позволяющей создавать как классические схемы АФНФ [24, 25], так и различные их модификации, в том числе на программируемых структурах типа FPGA [19–21]. Как было показано в разд. 1, для базового элемента весьма важна функция переключения *Switch*, которая реализуется одновременно с функцией выбора *Select* одной из двух пар путей через базовый элемент. Аргументом этой функции являются значения бита $c_j \in \{0, 1\}$ запроса C_i , который определяет одну из двух пар путей через базовый элемент с переключением общей пары путей АФНФ либо без переключения.

Главный недостаток классического базового элемента заключается в неиспользовании им всех четырех случайных значений задержек ($\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$), внесенных на этапе изготовления. В процессе функционирования АФНФ применяются только два, а именно $\delta_{0,j}$ и $\delta_{1,j}$ (2), из четырех приведенных ниже возможных случайных значений, представляющих собой добавленные значения задержки j -м базовым элементом:

$$\begin{aligned} \delta(1)_j &= \Delta(0)_{1,j} - \Delta(0)_{2,j}, & \delta(2)_j &= \Delta(0)_{1,j} - \Delta(1)_{2,j}, \\ \delta(3)_j &= \Delta(1)_{1,j} - \Delta(0)_{2,j}, & \delta(4)_j &= \Delta(1)_{1,j} - \Delta(1)_{2,j}. \end{aligned} \quad (3)$$

Отметим, что в данном случае рассматривается классический базовый элемент, для которого $\delta(1)_j = \delta_{0,j}$ и $\delta(4)_j = \delta_{1,j}$.

Как модификации базового элемента, состоящего из двух двухвходовых мультиплексоров, можно предложить структуры, изображенные на рис. 3. На первый взгляд, данные структуры повторяют друг друга, однако в их поведении, описываемом задержками (3), которые добавляет базовый элемент, наблюдаются заметные различия. Эти различия заключаются как в наборе функций базового элемента, так и в используемых им произвольных значениях $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ задержек, внесенных на стадии производства. Суммируя поведение классического базового элемента и четырех его модификаций (рис. 3), дадим их подробное описание (табл. 3).

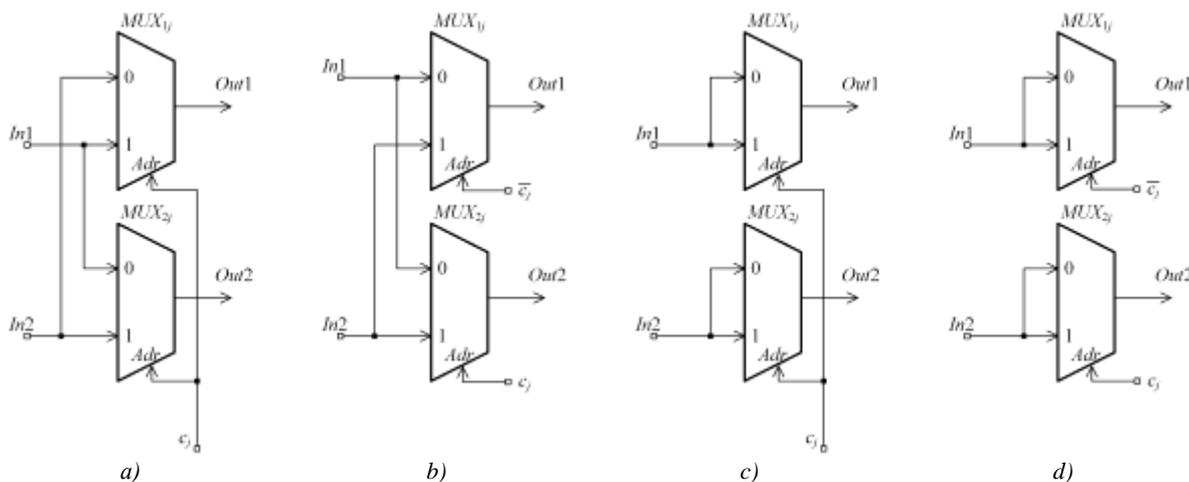


Рис. 3. Модификации базового элемента
 Fig. 3. Base element modifications

Таблица 3
Описание базового элемента и его модификаций

Table 3
Description of basic element and its modifications

Базовый элемент <i>Basic Element</i>	Функция <i>Function</i>	Математическое описание <i>Mathematical description</i>	Значение δ_{c_j} <i>Value δ_{c_j}</i>
Классический базовый элемент <i>Classical base element</i>	<i>Switch Select</i>	$d_j = d_{j-1} \cdot (-1)^{c_j} + \delta_{c_j}$	$\delta_{0j} = \delta(1)_j = \Delta(0)_{1,j} - \Delta(0)_{2,j}$, $\delta_{1j} = \delta(4)_j = \Delta(1)_{1,j} - \Delta(1)_{2,j}$
Модификация a) <i>Modification a)</i>	<i>Switch Select</i>	$d_j = d_{j-1} \cdot (-1)^{\bar{c}_j} + \delta_{c_j}$	$\delta_{0j} = \delta(1)_j = \Delta(0)_{1,j} - \Delta(0)_{2,j}$, $\delta_{1j} = \delta(4)_j = \Delta(1)_{1,j} - \Delta(1)_{2,j}$
Модификация b) <i>Modification b)</i>	<i>Switch Select</i>	$d_j = d_{j-1} \cdot (-1)^{\bar{c}_j} + \delta_{c_j}$	$\delta_{0j} = \delta(3)_j = \Delta(1)_{1,j} - \Delta(0)_{2,j}$, $\delta_{1j} = \delta(2)_j = \Delta(0)_{1,j} - \Delta(1)_{2,j}$
Модификация c) <i>Modification c)</i>	<i>Select</i>	$d_j = d_{j-1} + \delta_{c_j}$	$\delta_{0j} = \delta(1)_j = \Delta(0)_{1,j} - \Delta(0)_{2,j}$, $\delta_{1j} = \delta(4)_j = \Delta(1)_{1,j} - \Delta(1)_{2,j}$
Модификация d) <i>Modification d)</i>	<i>Select</i>	$d_j = d_{j-1} + \delta_{c_j}$	$\delta_{0j} = \delta(3)_j = \Delta(1)_{1,j} - \Delta(0)_{2,j}$, $\delta_{1j} = \delta(2)_j = \Delta(0)_{1,j} - \Delta(1)_{2,j}$

Возможны и другие модификации базового элемента, часть из которых неприменима для создания пары путей последовательным подключением базовых элементов, как это предполагает методология построения АФНФ. Некоторые модификации базового элемента могут быть получены несколькими путями. Так, модификация a) может быть получена при использовании классического элемента, у которого заменено обозначение входа *In1* на *In2* и наоборот, а соединение между элементами остается прежним. Следовательно, выходы предыдущего элемента *Out1* и *Out2* подключаются к входам *In1* и *In2* следующего модифицированного таким образом элемента.

Анализируя данные, приведенные в табл. 3, можно сделать очевидный вывод о том, что невозможно построить базовый элемент, который реализует обе функции *Switch* и *Select* и одновременно позволяет генерировать пару путей с одной из четырех добавленных задержек $\delta(1)_j$, $\delta(2)_j$, $\delta(3)_j$ или $\delta(4)_j$ (3). Это ограничение объясняется тем, что запрос c_j , подаваемый на j -й базовый элемент, имеет только два значения, которые позволяют выбрать один из двух, а не из четырех вариантов.

В качестве новых структур АФНФ можно рассматривать последовательное подключение различных базовых элементов и всевозможных их модификаций в различных сочетаниях. Таким образом нарушаются однородность и регулярность АФНФ, негативно сказывающиеся на их основных свойствах. Весьма интересны решения, которые используют базовые элементы, выполняющие обе функции *Switch* и *Select*, и базовые элементы, выполняющие только функцию *Select*.

Одним из многообещающих решений является модификация d), в которой мультиплексоры MUX_{1j} и MUX_{2j} управляются независимо, каждый своим битом c_{1j} и c_{2j} запроса $c_j = c_{1j}c_{2j}$. В этом случае запрос принимает четыре возможных значения, что позволяет использовать все четыре добавленные задержки $\delta(1)_j$, $\delta(2)_j$, $\delta(3)_j$ и $\delta(4)_j$ (3). Принимая во внимание значимость функции *Switch* как инструмента для нарушения структурной и топологической регулярности в АФНФ, ее аналог – операция *Switch* – реализуется путем соединений базовых элементов. Таким образом, выход *Out1* $j-1$ -й ступени подключается к входу *In2* j -й ступени, а выход *Out2* – к *In1*. Отметим, что операция *Switch* не повторяет функцию *Switch*, зависящую от запроса, но также является действенным инструментом для нарушения регулярности и повторяемости в схеме АФНФ. Особенно большое значение функция *Switch* имеет для реализаций АФНФ на FPGA. Функциональная схема подобной АФНФ, когда операция *Switch* используется на каждой ступени, изображена на рис. 4.

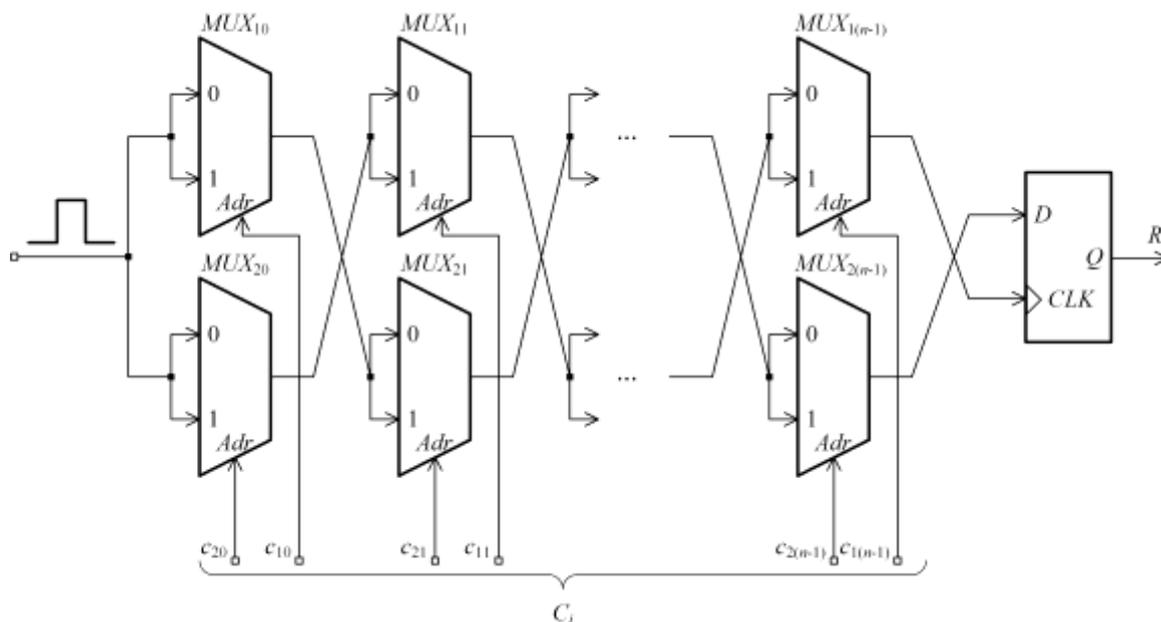


Рис. 4. Функциональная схема АФНФ

Fig. 4. Functional diagram of APUF

Математическая модель, описывающая АФНФ на рис. 4 рекурсивным соотношением для разницы задержки d_j прохождения сигнала по первому пути по отношению ко второму пути после j -й ступени, имеет вид

$$d_j = d_{j-1} \cdot (-1) + (\overline{c_{1j}c_{2j}}) \cdot \delta(1)_j + (\overline{c_{1j}c_{2j}}) \delta(2)_j + (c_{1j}\overline{c_{2j}}) \cdot \delta(3)_j + (c_{1j}c_{2j}) \delta(4)_j, \quad (4)$$

где величины задержек представляют собой действительные числа, а переменные c_{1j} и c_{2j} , $j \in \{0, 1, 2, \dots, n-1\}$, являются булевыми переменными, булевы операции над которыми формируют значения нуля либо единицы как одного из сомножителей.

Анализ изображенной на рис. 4 АФНФ показывает, что аппаратная реализация подобной АФНФ характеризуется высокой эффективностью. На один бит n -разрядного запроса C_i необходим только один двухвходовый мультиплексор. Отметим, что все существующие решения для АФНФ требуют как минимум двух таких мультиплексоров [12–14].

Сопоставляя предложенную схему АФНФ с классической, отметим, что в первом случае базовый элемент j -й ступени всегда выполняет переключение путей, а во втором случае только при $c_j = 1$. В то же время в результате операции *Select* в предложенной структуре осуществляется выбор одного из четырех возможных добавленных значений задержки, а в классическом варианте – одного из двух.

Наличие двух битов c_{1j} и c_{2j} в запросе c_j для j -й ступени означает, что общее количество битов запроса C_i удваивается, а их общее число для АФНФ (рис. 4), состоящей из n последовательно подключенных базовых элементов, равняется 2^{2n} . Такую АФНФ можно рассматривать как двухмерную, так как конфигурация конкретной пары путей осуществляется и по горизонтали, и по вертикали.

3. Синтез базового элемента для 2D-АФНФ. В дальнейшем будем основываться на идее выбора одного из возможных значений задержки прохождения сигнала через многовходовый мультиплексор. Таким образом, базовый элемент будет включать многовходовые мультиплексоры с уникальными значениями задержки по их входам. Главным параметром базового элемента является количество реконфигурируемых путей прохождения сигнала через базовый элемент. Отметим, что не только две пары путей позволяют обеспечить эффективность АФНФ, как это имеет место в большинстве известных решений. Базовый элемент должен формировать множество независимых путей прохождения через него сигналов. Рассмотрение, например,

АФНФ с четырьмя одновременно функционирующими путями, очевидно, позволит получить новые положительные эффекты. Увеличение количества путей в АФНФ и извлечение из них ответов на запросы представляются перспективными направлениями дальнейших исследований.

Ключевым параметром базового элемента АФНФ является количество $Q \geq 2$ одновременно функционирующих путей при фиксированном запросе, подаваемом на него. Формально значение Q определяет число входов $In1, In2, \dots, InQ$ базового элемента и такое же число его выходов $Out1, Out2, \dots, OutQ$, а также количество мультиплексоров в базовом элементе и минимальное число входов в каждом из них. Функционально произвольный базовый элемент выполняет соединение входов с выходами, и эти соединения являются непересекающимися, а общее количество их разнообразных конфигураций определяется величиной $Q!$. Например, для $Q = 4$ одной из возможных конфигураций четырех путей прохождения сигнала через базовый элемент является $In1-Out2, In2-Out3, In3-Out1$ и $In4-Out4$, а максимальное количество подобных конфигураций $4! = 24$. Для каждого адреса, одновременно подаваемого на входы четырех мультиплексоров, задается своя конфигурация путей базового элемента. Если базовый элемент строится, например, с использованием мультиплексоров с четырьмя входами, то конфигурация четырех путей строится для каждого из четырех адресов. Конфигурация $In1-Out2, In2-Out3, In3-Out1$ и $In4-Out4$ для значения адреса Adr четырехвходового мультиплексора, равного 00, означает, что вход $In1$ базового элемента подключен к нулевому (00) входу второго мультиплексора, $In2$ – к нулевому входу третьего мультиплексора, $In3$ – к одноименному входу первого мультиплексора и, наконец, $In4$ – к такому же входу четвертого мультиплексора. Графически четыре возможные конфигурации путей из 24 возможных для всех адресов базового элемента, построенного на четырех мультиплексорах с четырьмя входами каждый, показаны на рис. 5.

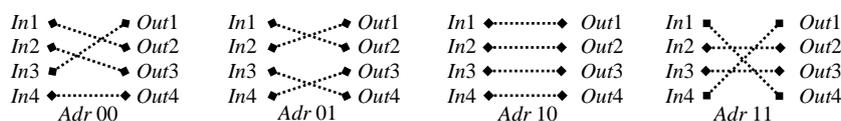


Рис. 5. Конфигурации путей базового элемента на четырех мультиплексорах

Fig. 5. Base element path configurations on four multiplexers

Конфигурации путей через базовый элемент, как отмечалось ранее, реализуются подключением входов $In1, In2, \dots, InQ$ базового элемента к входам мультиплексоров, входящих в этот базовый элемент. Каждый из Q используемых мультиплексоров идентифицируется одним из выходов $Out1, Out2, \dots, OutQ$ базового элемента, а входы – значениями его адресов Adr . Например, для рассмотренного выше примера каждый из четырех мультиплексоров имеет входы 00, 01, 10 и 11. Отметим что значение Adr в простейшем случае представляет собой запрос c_j , подаваемый на j -ю ступень АФНФ, которая представляет собой базовый элемент. Для рассмотренного выше примера базового элемента и его конфигураций путей (рис. 5) топология связей мультиплексоров с входами $In1, In2, In3$ и $In4$ базового элемента представлена в табл. 4.

Таблица 4
Топология связей мультиплексоров базового элемента

Table 4
Topology of multiplexer's connections of basic element

Вход <i>Input</i>	<i>Out1</i>	<i>Out2</i>	<i>Out3</i>	<i>Out4</i>
00	<i>In3</i>	<i>In1</i>	<i>In2</i>	<i>In4</i>
01	<i>In2</i>	<i>In1</i>	<i>In4</i>	<i>In3</i>
10	<i>In1</i>	<i>In2</i>	<i>In3</i>	<i>In4</i>
11	<i>In4</i>	<i>In2</i>	<i>In3</i>	<i>In1</i>

Анализ табл. 4 показывает, что при равенстве количества входов мультиплексора величине Q формально она должна удовлетворять требованию использовать в каждой строке таблицы все идентификаторы входов $In1, In2, \dots, InQ$, причем каждый идентификатор – только один раз.

По умолчанию любой известный базовый элемент, применяемый для построения АФНФ, выполняет операцию *Select*, которая реализует выбор пары путей либо в общем случае большего количества путей базового элемента из множества возможных. В рассмотренном примере при фиксированном подключении мультиплексоров выбор осуществляется из четырех возможных конфигураций четырех путей (см. рис. 5). Отметим, что каждый из выбранных путей характеризуется своей индивидуальной задержкой сигнала, определяемой как вариациями параметров элементов, внесенных на этапе производства, так и их межсоединениями. Второй фактор имеет определяющее значение для реализации АФНФ на программируемых структурах [19]. Более того, возможность перепрограммирования подобных структур на FPGA раскрывает еще большие возможности для АФНФ.

На основе параметра Q , определяющего количество одновременно функционирующих путей, строится схема арбитра, формирующего ответы на подаваемые запросы. Для обеспечения высокой стабильности и непредсказуемости 2D-АФНФ в качестве эффективной схемы арбитра можно использовать, например, арбитра, основанный на операции сложения по модулю два (XOR arbiter) [26, 27]. Возможны и другие решения построения схемы арбитра в зависимости от количества генерируемых путей и формируемых битов ответа. Следует отметить, что увеличение количества битов ответа уменьшает стабильность и непредсказуемость 2D-АФНФ и, наоборот, его уменьшение повышает стабильность и одновременно непредсказуемость значений ответа.

Функция *Switch* в предлагаемых решениях базового элемента трансформируется в операцию реконфигурирования (*Reconfiguration*) путей, при которой возможны различные переключения путей из множества генерируемых.

Рассмотренная методика синтеза базового элемента основана на расширении функциональных возможностей АФНФ за счет увеличения альтернатив выбора возможных путей, т. е. не за счет количества последовательно подключенных базовых элементов, а за счет расширения возможностей выбора путей по запросу и за счет большего множества их вариантов. Для подтверждения данного тезиса в качестве простейшего примера рассмотрим классическую реализацию АФНФ, для которой построим альтернативную структуру 2D-АФНФ. Предположим, что в обоих случаях будут использоваться только два мультиплексора ($Q = 2$). Отличием 2D-АФНФ от классической ее реализации, приведенной на рис. 1, является применение для построения базового элемента вместо двухвходовых мультиплексоров с большим количеством входов, например с четырьмя. Для этого случая базовый элемент будет реализовывать конкретную конфигурацию пар путей из множества возможных. Каждый адрес двух мультиплексоров с четырьмя входами базового элемента идентифицирует свою пару путей (рис. 6).

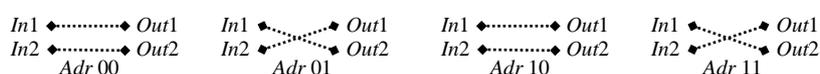


Рис. 6. Конфигурации путей базового элемента на двух мультиплексорах

Fig. 6. Base element path configurations on two multiplexers

В силу ограниченности количества путей (только два), проходящих через базовый элемент, число возможных конфигураций, формируемых им, также ограничено. В рассматриваемом примере их число равняется $2^4 = 16$, одна из указанных конфигураций изображена на рис. 6. Это означает, что при реализации 2D-АФНФ для анализируемого случая базового элемента его структура может быть разной, не повторяющейся для различных ее ступеней. На рис. 7 показана структура, соответствующая конфигурации путей на рис. 6.

Использование для случая $Q = 2$ мультиплексоров с восемью входами увеличивает число возможных конфигураций пар путей базового элемента до $2^8 = 256$. Во всех рассмотренных примерах базовый элемент реализует большее число уникальных пар путей. Следует отметить, что в качестве базовых элементов 2D-АФНФ могут быть предложены их реализации не только на мультиплексорах и не только с их общим количеством, равным размерности запроса C . Например, при реализации 2D-АФНФ эффективным представляется применение базовых элементов, использующих элементы И, ИЛИ и сумматор по модулю два [18].

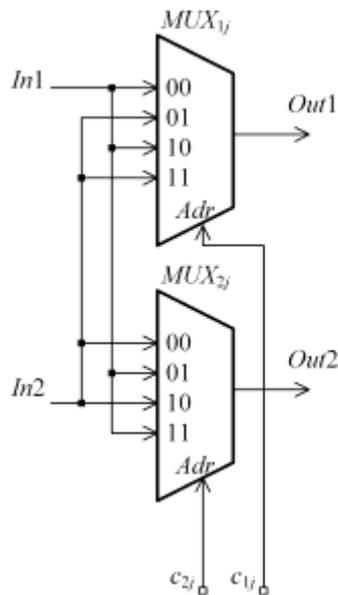


Рис. 7. Функциональная схема базового элемента
Fig. 7. Functional diagram of base element

Таким образом, для реальных структур 2D-АФНФ каждый их базовый элемент может быть реализован по уникальной, отличающейся от других базовых элементов, функциональной схеме и может управляться более чем одним битом запроса. В предельном случае 2D-АФНФ может быть реализована на одном базовом элементе, который для каждого из 2^n запросов C будет формировать свою уникальную пару путей.

4. Описание экспериментальных исследований. Для подтверждения предложенных в статье новых решений по построению АФНФ был проведен ряд экспериментов на программируемых логических интегральных схемах FPGA Xilinx Zynq7, входящих в состав плат быстрого прототипирования цифровых устройств Digilent Zybo Z7-10. Общая структура аппаратуры экспериментальной установки изображена на рис. 8.

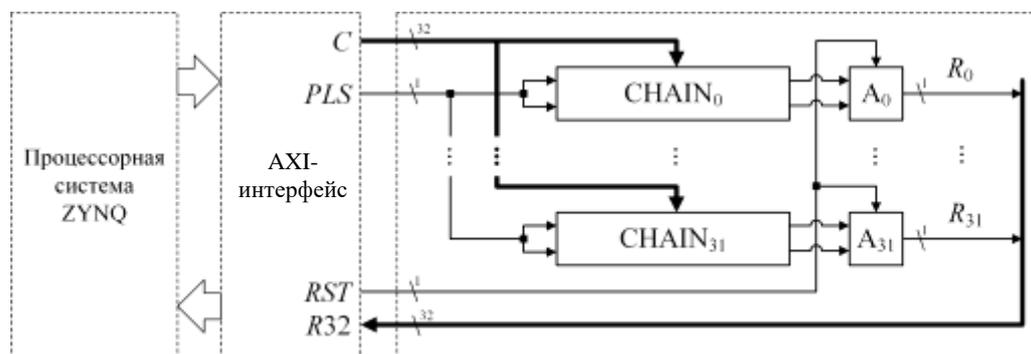


Рис. 8. Структурная схема экспериментальной установки
Fig. 8. Block diagram of experimental installation

Основу экспериментальной схемы составляют 32 реализации АФНФ (APUF_j), структурно разделенные на рисунке блоками CHAIN_j и A_j, $j \in \{0,1,2,\dots,31\}$. Каждый блок CHAIN_j представляет собой $n = 32$ последовательно соединенные ступени схемы APUF_j, на которые подается 32-разрядное значение запроса C_i и фронт тестового импульса PLS. Выводы последней ступени каждого экземпляра APUF_j подключены к независимым схемам арбитров A_j, которые, в свою очередь, вырабатывают значения ответов $R_j^i = \text{APUF}_j(C_i)$, объединенных в единую 32-разрядную шину ответов R32. Для инициализации схем арбитров применяется асинхронный сигнал RST, значение которого, как и значения сигнала PLS и шины C, формируется программно процессорной системой ZYNQ, входящей в состав кристалла FPGA.

Суть экспериментов заключалась в оценке такого важного показателя для схемных реализаций ФНФ, как стабильность [28]. Для его вычисления введем частоту встречаемости единичного ответа $R_j^i = 1$ при многократной подаче запроса C_i в E повторяющихся экспериментах:

$$P_1^E(C_i, j) = \frac{1}{E} \sum_{e=1}^E R_j^i. \quad (5)$$

Введем параметр нестабильности пары «запрос-ответ» $S^X(C_i, R_j^i) \in \{0,1\}$. Тогда если $0 < P_1^E(C_i, j) < 1$, то пара «запрос-ответ» (C_i, R_j^i) считается нестабильной и $S^X(C_i, R_j^i) = 1$. В противных случаях, когда $P_1^E(C_i, j) = 0$ либо $P_1^E(C_i, j) = 1$, пара считается стабильной и $S^X(C_i, R_j^i) = 0$.

Метрику стабильности конкретного схемного экземпляра APUF_j будем вычислять следующим образом:

$$S(\text{APUF}_j) = 1 - \frac{1}{M} \sum_{m=1}^M S^X(C_m, R_j^m), \quad (6)$$

где $j = \{0,1,2,\dots,31\}$ – индекс АФНФ, M – число поданных уникальных запросов.

В ходе проведенных экспериментов были выбраны следующие параметры: число повторений каждого запроса $E = 100$ и число сгенерированных запросов $M = 10^5$. Все запросы были сгенерированы программной моделью 32-разрядного генератора псевдослучайной M-последовательности, которая обеспечивает уникальность и равномерность подаваемых запросов из 2^{32} возможных. В качестве схем арбитров A_j был использован синхронный D-триггер (технологический примитив FDCE). Оценке стабильности были подвергнуты три различные реализации блоков CHAIN_j: SCH_1, SCH_2 и SCH_3, которые были построены по схемам на рис. 1, 4 и 7 соответственно.

На рис. 9 изображены графики отсортированных по убыванию значений $S(\text{APUF}_j)$ для перечисленных схемных реализаций. Индексы схем АФНФ на оси абсцисс являются условными в силу осуществленной сортировки значений для трех различных реализаций. Для классической реализации АФНФ (схема SCH_1) только пять экземпляров из 32 обладают максимально возможным значением стабильности $S(\text{APUF}_j) = 1$. Остальные 27 экземпляров имеют значения метрики стабильности, принадлежащие диапазону $[0,99348; 0,99477]$. Схемный вариант SCH_2 показал наличие 27 стабильных реализаций, а вариант SCH_3 – 16 стабильных реализаций из 32 возможных.

Данные, полученные в ходе проведенных экспериментов, показывают бóльшую стабильность в сравнении с классической реализацией АФНФ и состоятельность применения предложенных схемотехнических решений по построению 2D-АФНФ.

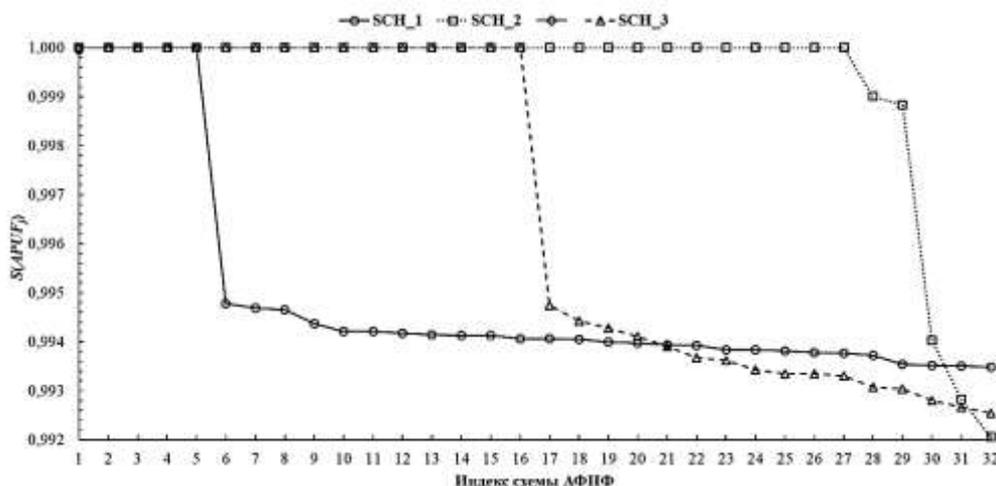


Рис. 9. Значения метрики стабильности $S(APUF_j)$ для различных схемных реализаций блоков $CHAIN_j$

Fig. 9. $S(APUF_j)$ stability parameter values for different implementation of $CHAIN_j$ blocks

Кроме параметра стабильности одной из важнейших характеристик ФНФ является соотношение единичных и нулевых ответов для различных подаваемых запросов. Подобная характеристика оценивается соответствующей метрикой единообразия (uniformity) [28]:

$$Un(APUF_j) = 1 - 2 \cdot \left| \frac{WH(R_j^M)}{M} - 0,5 \right|, \quad (7)$$

где $R_j^M = (R_j^0, R_j^1, R_j^2, \dots, R_j^{M-1})$ – вектор ответов экземпляра ФНФ $APUF_j$ на однократно поданные M уникальных запросов, WH – вес по Хэммингу. Значение $Un(APUF_j) = 1$ достигается только при условии равенства числа полученных единичных и нулевых ответов соответствующего экземпляра $APUF_j$. Нулевое значение метрики (7) свидетельствует о равенстве всех ответов на поданные запросы.

На рис. 10 изображены графики отсортированных по убыванию значений $Un(APUF_j)$ для перечисленных схемных реализаций для $M = 10^5$ запросов. Индексы схем АФНФ на оси абсцисс также являются условными.

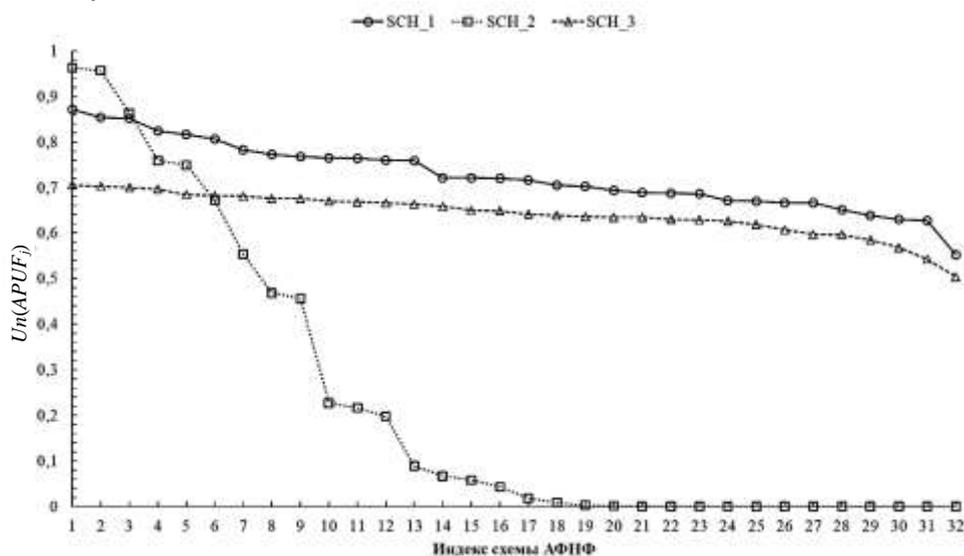


Рис. 10. Значения метрики единообразия $Un(APUF_j)$ для различных схемных реализаций блоков $CHAIN_j$

Fig. 10. $Un(APUF_j)$ uniformity parameter values for different implementation of $CHAIN_j$ blocks

На рис. 10 видно, что величины метрики $Un(APUF_i)$ сравнимы со значениями для классической АФНФ, а в отдельных случаях даже их превосходят.

Предложенные в статье схемы двухмерных АФНФ нуждаются в более детальном исследовании других статистических и вероятностных характеристик, в том числе при их реализациях на различных типах FPGA.

Заключение. В статье предложен подход к построению АФНФ, основанный на применении разнообразных модификаций базовых элементов. Главной характеристикой новых АФНФ является их двухмерность, заключающаяся не только в линейном наращивании количества базовых элементов, но и в придании более широких функциональных возможностей каждому базовому элементу. Интересным представляется дальнейшее развитие идеи построения 2D-АФНФ за счет комбинированного применения различных базовых элементов и реализации их на программируемых структурах типа FPGA.

Вклад авторов. Ярмолик В. Н. предложил идею построения двухмерных физически неклонированных функций, Иванюк А. А. принял участие в обобщении и анализе полученных результатов и проведении экспериментальных исследований.

Список использованных источников

1. Pappu, R. Physical One-Way Functions: PhD Thesis in Media Arts and Sciences / R. Pappu. – Cambridge : Massachusetts Institute of Technology, 2001. – 154 p.
2. Silicon physical random functions / B. Gassend [et al.] // Proc. of 9th Computer and Communications Security Conf. (CCS'02), Washington, DC USA, 18–22 Nov. 2002. – Washington, 2002. – P. 148–160.
3. Tuyls, P. Security with Noisy Data: On Private Biometrics, Secure Key Storage and Anti-Counterfeiting / P. Tuyls, B. Skoric, T. Kevenaar ; ed.: P. Tuyls. – N. Y. : Springer, 2007. – 339 p.
4. Rührmair, U. Strong PUFs: models, constructions, and security proofs / U. Rührmair, H. Busch, S. Katzenbeisser // Towards Hardware-Intrinsic Security / eds.: A.-R. Sadeghi, D. Naccache. – Berlin, Heidelberg : Springer, 2010. – P. 79–96.
5. Skoric, B. Robust key extraction from physical uncloneable functions / B. Skoric, P. Tuyls, W. Oprey // Proc. of Intern. Conf. Applied Cryptography and Network Security, N. Y., USA, 7–10 June 2005. – N. Y., 2005. – P. 407–422.
6. A technique to build a secret key in integrated circuits for identification and authentication applications / J. W. Lee [et al.] // Proc. of Intern. Symp. VLSI Circuits (VLSI'04), Honolulu, Hawaii, USA, 7–19 June 2004. – Honolulu, 2004. – P. 176–179.
7. Extracting secret keys from integrated circuits / D. Lim [et al.] // IEEE Transactions on Very Large Scale Integration (VLSI) Systems. – 2005. – Vol. 13, no. 10. – P. 1200–1205.
8. Maes, R. PUFKY: A fully functional PUF-based cryptographic key generator / R. Maes, A. van Herrewege, I. Verbauwhede // Proc. of 14th Intern. Workshop on Cryptographic Hardware and Embedded Systems (CHES 2012), Leuven, Belgium, 9–12 Sept. 2012. – Leuven, 2012. – P. 302–319.
9. Ярмолик, В. Н. Физически неклонированные функции / В. Н. Ярмолик, Ю. Г. Вашинко // Информатика. – 2011. – № 2(30). – С. 92–103.
10. Иванюк, А. А. Физическая криптография и защита цифровых устройств / А. А. Иванюк, С. С. Заливако // Доклады БГУИР. – 2019. – № 2(120). – С. 50–58.
11. Программная реализация физически неклонированных функций / Г. А. Мартвель [и др.] // Труды МФТИ. – 2020. – Т. 12, № 2. – С. 55–63.
12. Rührmair, U. On the foundations of Physical Uncloneable Functions / U. Rührmair, J. Sölter, F. Sehnke // IACR Cryptology ePrint Archive. – 2009. – Vol. 2009. – 20 p.
13. Delvaux, J. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise / J. Delvaux, I. Verbauwhede // Proc. of IEEE Intern. Symp. on Hardware-Oriented Security and Trust (HOST), Austin, TX, USA, 2–3 June 2013. – Austin, 2013. – P. 137–142.
14. PUF modeling attacks on simulated and silicon data / U. Rührmair [et al.] // IEEE Transactions on Information Forensics and Security. – 2013. – Vol. 11, no. 8. – P. 1876–1891.
15. Xu, X. Using statistical models to improve the reliability of delay-based PUFs / X. Xu, W. Burleson, D. E. Holcomb // Proc. of IEEE Computer Society Annual Symp. on VLSI, Pittsburgh, PA, USA, 11–13 July 2016. – Pittsburgh, 2016. – P. 547–552.

16. Agarwal, A. Statistical timing analysis for intra-die process variations with spatial correlations / A. Agarwal, D. Blaauw, V. Zolotov // Proc. of Intern. Conf. on Computer Aided Design (ICCAD03), San Jose, CA, USA, 9–13 Nov. 2003. – San Jose, 2003. – P. 900–907.
17. Клыбик, В. П. Метод увеличения стабильности физически неклонированной функции типа «арбитр» / В. П. Клыбик, С. С. Заливако, А. А. Иванюк // Информатика. – 2017. – № 1(53). – С. 31–43.
18. Ярмолик, В. Н. Физически неклонированные функции с управляемой задержкой распространения сигналов / В. Н. Ярмолик, А. А. Иванюк, Н. Н. Шинкевич // Информатика. – 2022. – Т. 19, № 1. – С. 32–49.
19. Morozov, S. An analysis of delay based PUF implementations on FPGA / S. Morozov, A. Maiti, P. Schaumont // Proc. of Intern. Symp. on Applied Reconfigurable Computing: Tools and Applications (ARC 2010), Los Angeles, CA, US, 25–27 Mar. 2010. – Los Angeles, 2010. – P. 382–387.
20. FPGA implementation of a cryptographically-secure PUF based on learning parity with noise / C. Jin [et al.] // Cryptography. – 2017. – Vol. 23, no. 1. – P. 1–20.
21. Gu, C. Improved reliability of FPGA-based PUF identification generator design / C. Gu, N. Hanley, M. O'neil // ACM Transactions on Reconfigurable Technology and Systems. – 2017. – Vol. 10, no. 3. – P. 1–23.
22. Kumar, A. METAPUF a challenge response pair generator / A. Kumar, S. L. Tripathi, R. Mishra // Periodicals of Engineering and Natural Sciences (PEN) . – 2018. – Vol. 2, no. 6. – P. 58–63.
23. Ozturk, E. Physical unclonable function with tristate buffers / E. Ozturk, G. Hammouri, B. Sunar // Proc. of IEEE Intern. Symp. on Circuits and Systems (ISCAS 2008), Seattle, Washington, USA, 18–21 May 2008. – Seattle, 2008. – P. 3194–3197.
24. Böhm, C. Physical Unclonable Functions in Theory and Practice / C. Böhm, M. Hofer. – N. Y. : Springer Science + Business Media, 2013. – 270 p.
25. Ярмолик, В. Н. Физически неклонированные функции типа арбитра с заведомо асимметричными параметрами путей / В. Н. Ярмолик, А. А. Иванюк // Доклады БГУИР. – 2022. – Т. 20, № 4. – С. 71–79.
26. A new Arbiter PUF for enhancing unpredictability on FPGA / T. Machida [et al.] // The Scientific World Journal. – 2015. – Vol. 2015, art. ID 864812. – 13 p.
27. Zhou, C. Secure and reliable XOR arbiter PUF design: An experimental study based on 1 trillion challenge response pair measurements / C. Zhou, K. K. Parhi, C. H. Kim // Proc. of 54th ACM/EDAC/IEEE Design Automation Conf. (DAC 2017), Austin, TX, USA, 18–22 June 2017. – Austin, 2017. – P. 1–6.
28. Maiti, A. A systematic method to evaluate and compare the performance of Physical Unclonable Functions / A. Maiti, V. Gunreddy, P. Schaumont ; eds.: P. Athanas, D. Pnevmatikatos, N. Sklavos // Embedded Systems Design with FPGAs. – N. Y., Springer, 2013. – P. 245–267.

References

1. Pappu R. *Physical One-Way Functions: PhD Thesis in Media Arts and Sciences*. Cambridge, Massachusetts Institute of Technology, 2001, 154 p.
2. Gassend B., Clarke D., Dijk M. S., Devadas S. Silicon physical random functions. *Proceedings of the 9th Computer and Communications Security Conference (CCS'02), Washington, DC USA, 18–22 November 2002*. Washington, 2002, pp. 148–160.
3. Tuyls P., Skoric B., Kevenaar T. *Security with Noisy Data: On Private Biometrics, Secure Key Storage and Anti-Counterfeiting*. In P. Tuyls (ed.). New York, Springer, 2007, 339 p.
4. Rührmair U., Busch H., Katzenbeisser S. Strong PUFs: models, constructions, and security proofs. *Towards Hardware-Intrinsic Security*. In A.-R. Sadeghi, D. Naccache (eds.). Berlin, Heidelberg, Springer, 2010, pp. 79–96.
5. Skoric B., Tuyls P., Oprea W. Robust key extraction from physical uncloneable functions. *Proceedings of International Conference Applied Cryptography and Network Security, New York, USA, 7–10 June 2005*. New York, 2005, pp. 407–422.
6. Lee J. W., Lim D., Gassend B., Suh G. E., ..., Devadas S. A technique to build a secret key in integrated circuits for identification and authentication applications. *Proceedings of International Symposium VLSI Circuits (VLSI'04), Honolulu, Hawaii, USA, 7–19 June 2004*. Honolulu, 2004, pp. 176–179.
7. Lim D., Lee J. W., Gassend B., Suh G. E., ..., Devadas S. Extracting secret keys from integrated circuits. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2005, vol. 13, no. 10, pp. 1200–1205.

8. Maes R., Van Herrewege A., Verbauwhede I. PUFKY: A fully functional PUF-based cryptographic key generator. *Proceedings of 14th International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2012), Leuven, Belgium, 9–12 September 2012*. Leuven, 2012, pp. 302–319.
9. Yarmolik V. N., Vashinko Y. G. *Physical unclonable functions*. Informatika [Informatics], 2011, no. 2(30), pp. 92–103 (In Russ.).
10. Ivaniuk A. A., Zalivaka S. S. *Physical cryptography and security of digital devices*. Doklady Belorusskogo gosudarstvennogo universiteta informatiki i radioelektroniki [Reports of the Belarusian State University of Informatics and Radioelectronics], 2019, no. 2(120), pp. 50–58 (In Russ.).
11. Martvel G. A., Chuprakov F. M., Nedostoev K. A., Baburin N. S. *Software implementation of physically non-cloneable functions*. Trudy Moskovskogo fiziko-tehnicheskogo instituta [Proceedings of Moscow Institute of Physics and Technology], 2020, vol. 12, no. 2, pp. 55–63 (In Russ.).
12. Rührmair U., Sölter J., Schnke F. On the foundations of Physical Unclonable Functions. *IACR Cryptology ePrint Archive*, 2009, vol. 2009, 20 p.
13. Delvaux J., Verbauwhede I. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise. *Proceedings of IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Austin, TX, USA, 2–3 June 2013*. Austin, 2013, pp. 137–142.
14. Rührmair U., Sölter J., Schnke F., Xu X., Mahmoud A., ..., Devadas S. PUF modeling attacks on simulated and silicon data. *IEEE Transactions on Information Forensics and Security*, 2013, vol. 11, no. 8, pp. 1876–1891.
15. Xu X., Burleson W., Holcomb D. E. Using statistical models to improve the reliability of delay-based PUFs. *Proceedings of IEEE Computer Society Annual Symposium on VLSI, Pittsburgh, PA, USA, 11–13 July 2016*. Pittsburgh, 2016, pp. 547–552.
16. Agarwal A., Blaauw D., Zolotov V. Statistical timing analysis for intra-die process variations with spatial correlations. *Proceedings of International Conference on Computer Aided Design (ICCAD03), San Jose, CA, USA, 9–13 November 2003*. San Jose, 2003, pp. 900–907.
17. Klybik V. P., Zalivaka S. S., Ivaniuk A. A. *Reliability enhancement method for "arbiter" physically unclonable function*. Informatika [Informatics], 2017, no. 1(53), pp. 31–43 (In Russ.).
18. Yarmolik V. N., Ivaniuk A. A., Shynkevich N. N. *Physically unclonable functions with controlled propagation delay*. Informatika [Informatics], 2022, vol. 19, no. 1, pp. 32–49 (In Russ.).
19. Morozov S., Maiti A., Schaumont P. An analysis of delay based PUF implementations on FPGA. *Proceedings of International Symposium on Applied Reconfigurable Computing: Tools and Applications (ARC 2010), Los Angeles, CA, US, 25–27 March 2010*. Los Angeles, 2010, pp. 382–387.
20. Jin C., Herder C., Ren L., Nguyen P. H., Fuller B., ..., Dijk M. van. FPGA implementation of a cryptographically-secure PUF based on learning parity with noise. *Cryptography*, 2017, vol. 23, no. 1, pp. 1–20.
21. Gu C., Hanley N., O'neil M. Improved reliability of FPGA-based PUF identification generator design. *ACM Transactions on Reconfigurable Technology and Systems*, 2017, vol. 10, no. 3, pp. 1–23.
22. Kumar A., Tripathi S. L., Mishra R. METAPUF a challenge response pair generator. *Periodicals of Engineering and Natural Sciences (PEN)*, 2018, vol. 2, no. 6, pp. 58–63.
23. Ozturk E., Hammouri G., Sunar B. Physical unclonable function with tristate buffers. *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS 2008), Seattle, Washington, USA, 18–21 May 2008*. Seattle, 2008, pp. 3194–3197.
24. Böhm C., Hofer M. *Physical Unclonable Functions in Theory and Practice*. New York, Springer Science + Business Media, 2013, 270 p.
25. Yarmolik V. N., Ivaniuk A. A. *Arbiter physical unclonable functions with asymmetric pairs of paths*. Doklady Belorusskogo gosudarstvennogo universiteta informatiki i radioelektroniki [Reports of the Belarusian State University of Informatics and Radioelectronics], 2022, vol. 20, no. 4, pp. 71–79 (In Russ.).
26. Machida T., Yamamoto D., Iwamoto M., Sakiyama K. A new Arbiter PUF for enhancing unpredictability on FPGA. *The Scientific World Journal*, 2015, vol. 2015, art. ID 864812, 13 p.
27. Zhou C., Parhi K. K., Kim C. H. Secure and reliable XOR arbiter PUF design: An experimental study based on 1 trillion challenge response pair measurements. *Proceedings of 54th ACM/EDAC/IEEE Design Automation Conference (DAC 2017), Austin, TX, USA, 18–22 June 2017*. Austin, 2017, pp. 1–6.
28. Maiti A., Gunreddy V., Schaumont P. A systematic method to evaluate and compare the performance of Physical Unclonable Functions. In P. Athanas, D. Pnevmatikatos, N. Sklavos (eds.). *Embedded Systems Design with FPGAs*. New York, Springer, 2013, pp. 245–267.

Информация об авторах

Ярмолик Вячеслав Николаевич, доктор технических наук, профессор, Белорусский государственный университет информатики и радиоэлектроники.

E-mail: yarmolik10ru@yahoo.com

Иваниук Александр Александрович, доктор технических наук, доцент, профессор кафедры информатики, заведующий совместной учебной лабораторией «СК хайникс мемори солишенс Восточная Европа», Белорусский государственный университет информатики и радиоэлектроники.

E-mail: ivaniuk@bsuir.by

Information about the authors

Vyacheslav N. Yarmolik, D. Sc. (Eng.), Prof., Belarusian State University of Informatics and Radioelectronics.

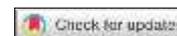
E-mail: yarmolik10ru@yahoo.com

Alexander A. Ivaniuk, D. Sc. (Eng.), Assoc. Prof., Prof. of Computer Science Department, Head of the Joint Educational Laboratory "SK Hynix Memory Solutions Eastern Europe", Belarusian State University of Informatics and Radioelectronics.

E-mail: ivaniuk@bsuir.by

БИОИНФОРМАТИКА

BIOINFORMATICS



УДК 519.23
<https://doi.org/10.37661/1816-0301-2023-20-1-27-39>

Оригинальная статья
Original Paper

Оценка размера эффекта воздействия в квазиэкспериментальных исследованиях

О. В. Красько

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

Аннотация

Цели. Поставлена задача оценить размер эффекта в квазиэкспериментальных исследованиях.
Методы. Используются методы теории оценивания и методы математической статистики.
Результаты. Оценен размер эффекта на порядковой и бинарной шкалах в случае разнонаправленных эффектов в группах в квазиэкспериментальных исследованиях для аналитического метода «различия в различиях».
Заключение. В работе представлены подходы к оценке абсолютных и стандартизированных размеров эффектов в экспериментальных и квазиэкспериментальных исследованиях. Дан краткий обзор оценок абсолютных и стандартизированных размеров эффектов для количественных и бинарных переменных исследования. Рассмотрен практический подход к оценке размеров эффектов порядковой и бинарной переменных в случае разнонаправленных эффектов в группах в квазиэкспериментальных исследованиях для аналитического метода «различия в различиях». Приведен пример расчетов абсолютных и стандартизированных размеров эффектов количественной и бинарной переменных в квазиэкспериментальных исследованиях в клинической эпидемиологии.

Ключевые слова: экспериментальные исследования, квазиэкспериментальные исследования, абсолютный размер эффекта, стандартизированный размер эффекта, аналитический метод «различия в различиях»

Для цитирования. Красько, О. В. Оценка размера эффекта воздействия в квазиэкспериментальных исследованиях / О. В. Красько // Информатика. – 2023. – Т. 20, № 1. – С. 27–39.
<https://doi.org/10.37661/1816-0301-2023-20-1-27-39>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 25.11.2022
Подписана в печать | Accepted 23.01.2023
Опубликована | Published 29.03.2023

Effect size assessment in quasi-experimental studies

Olga V. Krasko

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

Abstract

Objectives. To estimate effect sizes in quasi-experimental studies.

Methods. Methods of the theory of estimation, methods of mathematical statistics.

Results. Estimation of the effect size on an ordinal scale, estimation of the effect size on a binary in the case of opposite direction effects in groups, in quasi-experimental studies for the analytical method "differences in differences".

Conclusion. The paper considers approaches to assessing absolute and standardized effect sizes in experimental and quasi-experimental studies. A brief review of the estimators of absolute and standardized effect sizes for quantitative and binary study variables is provided. The applied approach is proposed to assess the effect sizes of a binary variable in the case of opposite direction effects in groups within a quasi-experimental studies for the "differences in differences" analytical method. An example of assessment of absolute and standardized effect sizes of quantitative and binary variables in quasi-experimental studies in clinical epidemiology is considered.

Keywords: experimental study design, quasi-experimental study design, absolute effect size, standardized effect size, "difference-in-differences" analytical method

For citation. Krasko O. V. *Effect size assessment in quasi-experimental studies*. Informatika [Informatics], 2023, vol. 20, no. 1, pp. 27–39 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-1-27-39>

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

Введение. Большие объемы медицинских данных предоставляют уникальные возможности для исторического анализа. Изучение баз медицинских данных может дать новые знания о диагностике и лечении различных заболеваний. На сегодняшний день изучение таких данных представляет собой отдельный аспект клинической эпидемиологии со своим дизайном и особенностями.

Поскольку данные историй болезней изначально не предназначались для исследования, а фиксировали историю лечения пациента, то использование их напрямую для научных исследований затруднительно из-за различных особенностей лечения, многочисленных конфаудеров и т. п. [1, 2]. Тем не менее существует возможность построения дизайна исследования и методов обработки данных для определения размера эффекта лечения или иного вмешательства для получения доказательств наличия или отсутствия эффектов.

В настоящей статье рассмотрены особенности анализа данных в квазиэкспериментальных исследованиях, дано понятие абсолютного и стандартизированного размера эффекта, на примере показаны простые расчеты размеров эффектов для непрерывных и бинарных откликов.

Особенности квазиэкспериментального дизайна исследований. Квазиэкспериментальный дизайн исследований имеет свою область применения, свои преимущества и недостатки [3–5]. Как и экспериментальный, квазиэкспериментальный дизайн исследования направлен на установление причинно-следственной связи между независимой и зависимой переменными. Однако в отличие от настоящего эксперимента при квазиэкспериментальном дизайне исследования не используется рандомизация. Вместо этого объекты исследования распределяются по группам на основе некоторых критериев, часто связанных с проявлением или течением заболевания. Квазиэкспериментальный дизайн исследования является полезным инструментом в ситуациях, когда настоящие эксперименты не могут быть использованы по этическим или практическим причинам.

Перечислим основные особенности квазиэкспериментального дизайна исследования в клинической эпидемиологии:

1. Назначение пациентов в группы лечения (или лечения и контроля) происходит неслучайным образом, в отличие от настоящего экспериментального исследования в нем отсутствует рандомизация.

2. Часто используются исторические данные, лечение не проектируется, а назначается исходя из экспертного мнения специалиста. Таким образом, исследователь может не разрабатывать лечение, а только анализировать исторические результаты.

3. Исследуются группы пациентов, которые не могли попасть в рандомизированные клинические испытания (РКИ) по причине ограничений в нормативных актах, например запрета проведения РКИ на беременных, детях и т. п.

Квазиэкспериментальные исследования могут иметь более низкую внутреннюю и более высокую внешнюю валидность, чем настоящие эксперименты, поскольку есть возможность использовать ситуации реального мира вместо искусственно созданных условий эксперимента [3]. Отсутствие рандомизации, а также неспособность контролировать с помощью дизайнера уровень воздействия изучаемого фактора и иных связанных с исследованием факторов делают данный вид исследования менее желательным для установления причинно-следственной связи между фактором риска и исходом. Однако он является более распространенным в клинической эпидемиологии, когда врач, занимающийся клинической врачебной практикой, переходит к научным исследованиям. Типичный пример – это исследования, которые планируются не заранее, а исходя из опыта врача-клинициста. Например, врач-психолог несколько лет ведет прием пациентов с повышенной тревожностью. Лечение назначается по стандартной методике. Исходом является состояние пациента спустя год после начала лечения. Накопив практический опыт, врач понимает, что методику можно улучшить. Следующие несколько лет новые пациенты проходят лечение по измененной методике, также фиксируется их состояние спустя год после начала лечения. Фактический материал, накопленный по двум группам (стандартного и экспериментального лечения), должен послужить доказательством, что измененная методика дает лучшие результаты лечения.

Описанный тип исследования хотелось бы назвать клиническими испытаниями (нерандомизированными), однако клинические испытания являются разновидностью когортных исследований, поэтому наиболее точное определение таких исследований – когортные исследования. Лечение по разным методикам будет рассматриваться как фактор, влияющий на исход заболевания. Исследование будет квазиэкспериментальным, и после учета всех влияющих на него факторов можно будет оценить эффект от изучаемого фактора (измененной методики лечения).

Таким образом, методы и приемы обработки и анализа данных в квазиэкспериментальном исследовании актуальны и востребованы в прикладных исследованиях.

Одним из распространенных аналитических методов квазиэкспериментального исследования является метод «различие в различиях» (difference-in-differences, DID) – аналитический подход, который облегчает вывод о причинно-следственной связи, даже когда рандомизация невозможна. В самом простом случае применения DID сравниваются изменения результатов с течением времени между двумя группами: популяцией, участвующей в эксперименте (экспериментальная группа), и популяцией, не участвующей в нем (группа сравнения).

РКИ и исследования по методу DID имеют некоторые общие характеристики: они включают четко определенную исследуемую популяцию и набор условий лечения, где легко отличить группу лечения от контрольной группы, а также периоды времени до лечения и после. Наиболее важным их различием является то, что условия воздействия случайным образом распределяются между группами в РКИ, но не в исследованиях по методу DID. При случайном назначении воздействие лечения статистически не зависит от любого (измеренного или неизмеренного) фактора, который также может повлиять на результаты. В исследованиях по методу DID невозможно полагаться на случайное распределение, чтобы избежать систематической ошибки из-за неизмеренных вмешивающихся факторов. Вместо этого вводятся предположения, которые ограничивают множество возможных вмешивающихся факторов. В частности, метод DID предполагает, что вмешивающиеся факторы, имеющие неоднородность в различных группах, не изменяются во времени, а изменяющиеся во времени вмешивающиеся факторы инвариантны относительно групп. Эти утверждения называются общим предположением о трендах [1, 3–5].

Рассмотрим метод DID для простейшего исследования, где анализируются две группы, экспериментальная и контрольная, до начала исследования и после его окончания.

Сделаем некоторые предположения. Пусть $g = 1, 2$ – две группы, которые наблюдаются в два периода времени $t = 1, 2$, а $Y(1)_{gt}$ представляет собой интересующий результат для группы g в период t при гипотетическом сценарии, в котором воздействие было активным в группе g в момент времени t . Величина $Y(0)_{gt}$ является результатом при той же единице времени при альтернативном сценарии отсутствия воздействия. Эффект воздействия для этой конкретной группы и периода времени определяется выражением $\Delta_{gt} = Y(1)_{gt} - Y(0)_{gt}$, которое представляет собой разницу в значениях переменных интересующего результата для одной и той же группы при двух гипотетических сценариях.

Пусть во втором периоде воздействие происходит в группе $g = 2$, а не группе $g = 1$ и фиктивная переменная $T_g = 1[g = 2]$ идентифицирует наблюдения в группе 2, $T_g = 0[g = 1]$. Переменная T_g не имеет индекса времени, поскольку членство в группах постоянно и не меняется во времени. Пусть фиктивная переменная $P_t = 1[t = 2]$ идентифицирует наблюдения из периода времени 2, а переменная $P_t = 0[t = 1]$ – из периода времени 1. Тогда переменная P_t не имеет группового индекса, поскольку периоды времени не различаются между группами. В таком исследовании переменная воздействия является произведением двух фиктивных переменных: $S_{gt} = T_g \times P_t$. Для гипотетического результата в каждой группе в каждый момент времени можем записать $Y_{gt} = Y(0)_{gt} + (Y(1)_{gt} - Y(0)_{gt}) \times S_{gt}$. Данное формальное представление описывает проблему контрфактического вывода, которая возникает в каждом исследовании причинно-следственного вывода. В исследовании имеется доступ к данным Y_{gt} и S_{gt} , и одна из его задач – установить среднее значение $D = Y(1)_{gt} - Y(0)_{gt}$ в рамках изучаемой когорты. Однако для группы без воздействия (контрольной) невозможно оценить размер эффекта воздействия, но возможен расчет, связанный с изменением результата во времени (D_k). Общее предположение о трендах в этом случае позволяет трактовать оценку изменения D_k как связанную с периодом наблюдения. Для группы с воздействием (экспериментальной) невозможно рассчитать чистый размер эффекта воздействия, но возможен расчет, связанный с изменением результата во времени, который ассоциирован с воздействием (D_s). Если предположить аддитивность эффектов, связанных с воздействием и временем, то $D = D_s - D_k$. Конкретная формула расчета для D_k и D_s в данном случае зависит от шкалы, в которой измеряется соответствующая переменная. Это может быть разница в пропорциях для бинарной переменной, разница в порядках для порядковой шкалы или разница в измерениях до и после исследования на некоторой физической шкале. Расстояние между D_k и D_s может служить оценкой размера эффекта при соблюдении предположения о параллельных трендах. Далее приводятся основные формулы для расчетов различий в группах в разные моменты времени (повторные измерения) и для различных групп в некоторый момент времени.

Метод DID отражает логику построения оценки размера эффекта, связанного с воздействием. Далее будет показан практический пример использования данного метода в квазиэкспериментальном исследовании. Полный теоретический подход, связанный с несколькими группами и периодами наблюдения, а также возможными конфаундерами, не рассматривается в настоящей статье, поскольку ее цель – показать, что исследования, не связанные с РКИ, могут представлять интерес для специалистов в предметной области при оценке размера эффекта, и привести пример простого исследования для освоения основ данного метода.

Понятие размера эффекта. Размер эффекта – это численная оценка изменений, связанных с воздействием (лечением) (d -семейство оценок), или численная оценка связи переменных в исследовании (r -семейство оценок). Соответственно этому определению оценки размера эф-

фекта делятся на два больших семейства: d -семейство оценок, рассчитывающее размер различий в форме абсолютной разницы, стандартизированной разницы или отношения, и r -семейство оценок, оценивающее тесноту связи между переменными.

Согласно работе [6] размер эффекта определяется как количественное отражение величины некоторого явления, которое используется для решения интересующей задачи. На сегодняшний день существует более сотни различных мер оценки размеров эффектов для разных типов переменных и для ответов на научные вопросы.

Далее рассматриваются основные меры d -семейства оценок размера эффекта как наиболее распространенные в рандомизированных контролируемых испытаниях и квазиэкспериментальных исследованиях.

Абсолютные размеры эффектов. В экспериментальных и квазиэкспериментальных исследованиях традиционно используется d -семейство оценок для демонстрации лечебного эффекта (воздействия). Рассматривается исход в экспериментальной группе и группе контроля. Различия оцениваются в виде точечной оценки 95 % доверительного интервала (ДИ). Однако нельзя не отметить, что наряду с оценкой абсолютных различий на основе ДИ оценка стандартизированного размера эффекта в последнее время приобретает популярность в научных публикациях.

Рассмотрим сначала оценки абсолютных различий для количественной и бинарной переменных, которые чаще всего используются для оценки исхода воздействия в экспериментальных и квазиэкспериментальных исследованиях (табл. 1). В квазиэкспериментальных исследованиях необходимо в обязательном порядке контролировать исходные состояния наблюдаемых в группах, поскольку они могут значительно различаться.

Таблица 1
Основные оценки абсолютных размеров эффектов в экспериментальных и квазиэкспериментальных исследованиях

Table 1

Main absolute effect sizes in experimental and quasi-experimental studies

Количественная переменная <i>Quantitative variable</i>		
Расстояние между двумя независимыми группами	Нормальное распределение переменной в каждой группе	Оценка среднего разности между двумя случайными величинами $d = \bar{x}_2 - \bar{x}_1$ с расчетом ДИ
	Распределение, отличное от нормального	Оценка Ходжеса – Лемана (Hodges-Lehmann Estimator) с расчетом ДИ [7]
Расстояние между двумя зависимыми группами (парные сравнения), рассчитывается переменная различий между измерениями в каждой паре	Для переменной различий в случае ее нормального распределения	Оценка среднего \bar{x} с расчетом ДИ среднего
	Для переменной различий в случае ее распределения, отличного от нормального	Расчет псевдомедианы (оценка Ходжеса – Лемана для одной выборки) с расчетом ДИ
Бинарная переменная <i>Binary variable</i>		
Различия в пропорциях между двумя независимыми группами	Биномиальное распределение	Оценка разницы в пропорциях с расчетом ДИ
Различия в пропорциях между двумя зависимыми группами (парные сравнения)		

Оценка Ходжеса – Лемана [7, 8] определяет разницу между двумя популяциями. Для двух наборов данных с m - и n -наблюдениями набор составленных из них двухэлементных наборов представляет собой декартово произведение, содержащее $m \times n$ пар точек (по одной из каждого набора). Каждая такая пара определяет одну разность значений. Оценка Ходжеса – Лемана является медианой разностей набора $m \times n$. Псевдомедиана (оценка Ходжеса – Лемана для единственной популяции) определяется следующим образом. Пусть данные содержат n измерений. Определим $n(n+1)/2$ возможных пар. Для каждой пары рассчитаем среднее

$(x_i + x_j)/2, i \leq j, i=1, \dots, n, j=1, \dots, n$. Медиана всех возможных средних является псевдомедианой. Данная величина называется одновыборочной оценкой Ходжеса – Лемана и не оценивает ни среднее значение, ни медиану.

Стандартизированные размеры эффектов. Преимущество стандартизированного размера эффекта заключается в возможности сравнения различных исследований, а также унификации интерпретации.

Самая распространенная оценка стандартизированного размера эффекта количественной переменной в двух группах – это индекс d Коэна [9]. Он рассчитывается как $d = \frac{\bar{x}_1 - \bar{x}_2}{s}$, где \bar{x}_1 – оценка среднего в первой группе; \bar{x}_2 – оценка среднего во второй группе; s – общее стандартное отклонение двух групп, которое рассчитывается при равном размере групп как $s = \sqrt{\frac{s_1^2 + s_2^2}{2}}$, где s_1 и s_2 – стандартные отклонения в соответствующих группах.

Помимо индекса d Коэна существуют еще две похожие оценки: индекс Δ Гласса (Glass' Δ) и индекс g Хеджеса (Hedges' g). Они отличаются от индекса d Коэна способом расчета знаменателя. В случае индекса Δ Гласса это стандартное отклонение контрольной группы, а в случае индекса g Хеджеса – взвешенное стандартное отклонение двух групп.

Гласс предложил индекс Δ для случаев, когда две выборки представляют экспериментальную и контрольную группы, и рекомендовал использовать для стандартизации стандартное отклонение контрольной группы [10].

Индекс g Хеджеса [11] обеспечивает лучшую оценку размеров эффекта, если размеры выборки различаются, поскольку при этом используются размеры выборки в качестве весов при определении общего стандартного отклонения: $s = \sqrt{\frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}}$. Также эта оценка рекомендуется для малых выборок.

Одной из интересных оценок размера эффекта, которая построена на совершенно другом подходе (вероятностном), является индекс δ Клиффа (Cliff's δ) [12]. Он не зависит от вида распределения и предназначен для измерений в шкале не ниже порядковой. Пусть существуют выборки $X = \{x_1, \dots, x_m\}$ и $Y = \{y_1, \dots, y_n\}$. Тогда дельта-функция определяется как

$$\delta(i, j) = \begin{cases} +1, & x_i > y_j; \\ -1, & x_i < y_j; \\ 0, & x_i = y_j, \end{cases}$$

а размер эффекта – как $\delta = \frac{1}{m \times n} \sum_{i=1}^m \sum_{j=1}^n \delta(i, j)$.

Согласно определению $-1 \leq \delta \leq 1$. Значения, близкие к ± 1 , означают минимальное перекрытие двух распределений, в то время как значения, близкие к 0, означают значительное перекрытие наблюдений в двух выборках.

Еще одна уникальная оценка размеров эффектов – ранговая бисериальная корреляция [13–15], которая принадлежит к d -семейству эффектов (эффектов различий), и r -семейству эффектов (эффектов связи): $r_{rb} = \frac{2}{n_1} \times \left(\bar{Y}_2 - \frac{n+1}{2} \right) = \frac{2}{n_2} \times \left(\frac{n+1}{2} - \bar{Y}_1 \right)$, где \bar{Y}_1 и \bar{Y}_2 – среднее рангов в первой и второй группах соответственно; n_1 и n_2 – размер первой и второй группы соответственно, $n = n_1 + n_2$. Эта оценка используется для определения размера эффекта между двумя группами для показателей, имеющих шкалу измерений: порядковую, шкалу интервалов или шкалу

отношений. В случае двух групп ранговая бисериальная корреляция совпадает с индексом δ Клиффа [16].

Для оценки стандартизированных размеров эффектов бинарных переменных при рассмотрении различий в пропорциях в двух независимых группах можно использовать индекс h Коэна (Cohen's h) [9], который рассчитывается как $h = 2 \times (\arcsin \sqrt{p_1} - \arcsin \sqrt{p_2})$, где p_1, p_2 – пропорции интересующего исхода (бинарной переменной) в двух независимых группах.

При оценке размера эффекта как для зависимых групп, так и для независимых возможно применение стандартизированного размера эффекта, который рассчитывается аналогично индексу d Коэна или индексу Δ Гласса с учетом различий в расчете стандартного отклонения в зависимых и независимых группах для бинарного отклика.

В качестве оценки стандартизированного размера эффекта для зависимых групп Коэн предложил использовать индекс g (Cohen's g) [9]. Изначально Коэн разработал эту оценку для сравнения одновыборочной пропорции с вероятностью 0,5. Для зависимых групп в этом случае определяются несогласованные пары (пары, которые изменили значение бинарного признака). Среди них выделяется число однонаправленных пар и определяются пропорции $p_{0 \rightarrow 1}$ и $p_{1 \rightarrow 0}$, $p_{0 \rightarrow 1} + p_{1 \rightarrow 0} = 1$. Одна из пропорций сравнивается с 0,5. Коэн называет эту оценку индексом размера эффекта, который может быть направленным (g определяется как $g = p - 0,5$) или ненаправленным (g определяется как $g = |p - 0,5|$), где p – одна из двух пропорций $p_{0 \rightarrow 1}$ или $p_{1 \rightarrow 0}$. Ненаправленный индекс может принимать значения от 0 (полная симметрия, равные пропорции) до 0,5 (полная асимметрия).

Интерпретация оценок размера эффекта. Помимо количественной оценки различий часто необходимо понять, насколько эти различия интересны с точки зрения предметной области, а не только с точки зрения статистической значимости. Является размер эффекта большим или малым, – это вопрос интерпретации специалиста предметной области.

Основные пороги интерпретации стандартизированного размера эффекта количественных показателей приведены в табл. 2 и 3, а бинарных показателей – в табл. 4.

Таблица 2
 Интерпретация оценок стандартизированных размеров эффектов количественных показателей (индекса d Коэна, индекса Δ Гласса и индекса g Хеджеса) в различных научных источниках

Table 2
 Effect size interpretation for Cohen's d , Glass' Δ , and Hedges' g in various scientific sources

[9]	[17]	[18]	[19]
$d(\Delta, g) < 0,2$ Очень малый	$d(\Delta, g) < 0,1$ Крошечный	$d(\Delta, g) < 0,15$ Очень малый	$d(\Delta, g) < 0,2$ Очень малый
$0,2 \leq d(\Delta, g) < 0,5$ Малый	$0,1 \leq d(\Delta, g) < 0,2$ Очень малый	$0,15 \leq d(\Delta, g) < 0,36$ Малый	$0,2 \leq d(\Delta, g) < 0,41$ Малый
$0,5 \leq d(\Delta, g) < 0,8$ Средний	$0,2 \leq d(\Delta, g) < 0,5$ Малый	$0,36 \leq d(\Delta, g) < 0,65$ Средний	$0,41 \leq d(\Delta, g) < 0,63$ Умеренный
$d(\Delta, g) \geq 0,8$ Большой	$0,5 \leq d(\Delta, g) < 0,8$ Средний	$d(\Delta, g) \geq 0,65$ Большой	$d(\Delta, g) \geq 0,63$ Большой
–	$0,8 \leq d(\Delta, g) < 1,2$ Большой	–	–
–	$1,2 \leq d(\Delta, g) < 2$ Очень большой	–	–
–	$d(\Delta, g) \geq 2$ Огромный	–	–

Таблица 3

Интерпретация оценок стандартизированных размеров эффектов ранговой бисериальной корреляции в различных научных источниках

Table 3

Standardized effect size interpretation for rank biserial correlation in various scientific sources

[20]	[19]	[9]	[18]	[21]
$r_{pb} < 0,05$ Крошечный	$r_{pb} < 0,1$ Очень малый	$r_{pb} < 0,1$ Очень малый	$r_{pb} < 0,12$ Очень малый	$r_{pb} < 0,2$ Очень слабый
$0,05 \leq r_{pb} < 0,1$ Очень малый	$0,1 \leq r_{pb} < 0,2$ Малый	$0,1 \leq r_{pb} < 0,3$ Малый	$0,12 \leq r_{pb} < 0,24$ Малый	$0,2 \leq r_{pb} < 0,4$ Малый
$0,1 \leq r_{pb} < 0,2$ Малый	$0,2 \leq r_{pb} < 0,3$ Умеренный	$0,3 \leq r_{pb} < 0,5$ Умеренный	$0,24 \leq r_{pb} < 0,41$ Умеренный	$0,4 \leq r_{pb} < 0,6$ Умеренный
$0,2 \leq r_{pb} < 0,3$ Средний	$r_{pb} \geq 0,3$ Большой	$r_{pb} \geq 0,5$ Большой	$r_{pb} \geq 0,41$ Большой	$0,6 \leq r_{pb} < 0,8$ Сильный
$0,3 \leq r_{pb} < 0,4$ Большой	–	–	–	$r_{pb} \geq 0,8$ Очень сильный
$r_{pb} \geq 0,4$ Очень большой	–	–	–	–

Таблица 4

Интерпретация оценок стандартизированных размеров эффектов бинарных переменных

Table 4

Standardized effect size interpretation for binary variables

Индекс Коэна [9] Cohen's [9]	
h	g
$h = 0,2$ Небольшой	$g < 0,05$ Очень малый
$h = 0,5$ Средний	$0,05 \leq g < 0,15$ Малый
$h = 0,8$ Большой	$0,15 \leq d < 0,25$ Средний
–	$d \geq 0,25$ Большой

Индекс δ Клиффа не имеет авторской интерпретации, однако если предположить нормальное распределение исследуемого показателя в двух группах, то при индексе d Коэна, равном $d = 0,2$, $\delta = 0,147$; при $d = 0,5$ $\delta = 0,33$; при $d = 0,8$ $\delta = 0,474$.

Связь между индексом δ Клиффа и индексом d Коэна определяется выражением $\delta = 2 \times P(X \leq d/2) - 1 / P(X \leq d/2)$, где $P(X \leq x)$ – кумулятивная функция распределения Гаусса, $N(0,1)$.

Связь между индексом δ Клиффа и индексом d Коэна при распределении исследуемого показателя, отличного от нормального, рассмотрена в работах [22–24].

Пример квазиэкспериментального исследования. Рассмотрим условное квазиэкспериментальное исследование, в котором лечение назначалось с учетом некоторого бинарного признака и согласия пациента на его участие в исследовании и лечении. Контрольная группа формировалась из пациентов, которые не давали согласие на лечение, а соглашались только на участие в исследовании. Также мониторился уровень качества жизни пациента по 10-балльной визуально-аналоговой шкале (ВАШ). Контроль осуществлялся через три месяца после начала лечения. В каждой группе было 30 пациентов, группы не различались по возрасту, полу и степени тяжести заболевания.

Примером квазиэкспериментального исследования может служить исследование эффективности некоторого лечения, связанного с нарушением деятельности желудочно-кишечного тракта. Уровень боли оценивается по ВАШ (используется в медицинских исследованиях с 1921 г. [25]). Наличие рефлюкса желудка у пациента является бинарной переменной, которая определяет нерандомизированность назначения лечения.

Код примера использования воспроизводим в статистическом пакете R, для которого приведены коды генерации исходных данных, а также коды расчетов в виде отдельных чанков (см. https://drive.google.com/file/d/16bGni_smkDZchxCR4vN_p62tKJeP7WRF/view). Все интервальные оценки даны с уровнем доверия 95 %.

Анализ результатов исследования количественной переменной. Рассмотрим изменение показателя ВАШ в основной и контрольной группах. Учитывая, что этот показатель измеряется в балльной (порядковой) шкале, выбираем для его анализа непараметрические методы сравнения. Проверка распределений интересующих нас переменных показала, что распределения отличны от нормального.

Рассчитаем абсолютные и стандартизированные размеры эффектов в группах и различия в эффектах между группами («различия в различиях») (табл. 5 и 6). На основании расчетов различия в баллах между началом и концом исследования по ВАШ боли в контрольной группе (D_k) составили 3 (95 % ДИ 1,5; 3,5), а в экспериментальной группе (D_3) – -1 (95 % ДИ -2; 0). Искомый эффект «различия в различиях» $\Delta D = D_3 - D_k$ по ВАШ боли составил -2 (95 % ДИ -3; -1).

Таблица 5

Результаты оценки абсолютных и стандартизированных размеров эффектов количественной переменной в группах

Table 5

The assessment of the absolute and standardized effect sizes of the quantitative variable within groups

Группа Group	Статистическая значимость по критерию Вилкоксона при парных измерениях Statistical significance, paired Wilcoxon test	Абсолютный размер эффекта между началом и концом исследования, псевдомедиана (95 % ДИ) Absolute effect size within the study, pseudomedian (95 % CI)	Стандартизированный размер эффекта между началом и концом исследования, ранговая бисериальная корреляция (95 % ДИ) Standardized effect size within the study, the rank-biserial correlation coefficient (95 % CI)
Экспериментальная	0,083	-1 (-2; 0)	-0,45 (-0,76; 0,02)
Контрольная	<0,001	3 (1,5; 3,5)	0,90 (0,77; 0,96)

Таблица 6

Результаты оценки различий в эффектах между группами

Table 6

The assessment of the absolute and standardized effect sizes of the quantitative variable between groups

Экспериментальная группа, размер эффекта, псевдомедиана (95 % ДИ) Experimental group effect size, pseudomedian (95 % CI)	Контрольная группа, размер эффекта, псевдомедиана (95 % ДИ) Control group effect size, pseudomedian (95 % CI)	Абсолютный размер «различия в различиях» ΔD , оценка Ходжеса – Лемана (95 % ДИ) Absolute "difference-in-difference" distance ΔD , Hodges-Lehmann Estimator (95 % CI)	Стандартизированный размер «различия в различиях», ранговая бисериальная корреляция (95 % ДИ) Standardized "difference-in-difference" distance, the rank-biserial correlation coefficient (95 % CI)
-1 (-2; 0)	3 (1,5; 3,5)	-2 (-3; -1), $p < 0,001$	-0,62 (-0,77; -0,40)

Статистически значимого эффекта по снижению боли в экспериментальной группе не наблюдается, но вывод о его отсутствии некорректен. Учитывая усиление боли в контрольной группе, можно заключить, что воздействие (лечение) является эффективным для контроля уровня боли при данном заболевании с абсолютным размером эффекта по ВАШ до двух баллов. Интерпретация стандартизированных размеров эффекта может быть взята из табл. 3, но клинически более понятно, что при лечении уровень боли снижается в целом на один-три балла.

Анализ исследования результатов бинарной переменной. Код генерации данных приведен в приложении, распределение в группах исследования бинарной переменной (наличие рефлюкса) представлено в табл. 7.

Таблица 7

Бинарная переменная, распределение в группах в начале и конце исследования, размер эффектов в группах

Table 7

Binary variable distribution in groups at baseline and end of study, within groups effect sizes

Экспериментальная группа <i>Experimental group</i>		Конец исследования, n (%) <i>End of the study, n (%)</i>		Эффекты <i>Effects</i>	
		Да 14 (47 %)	Нет 16 (53 %)	Абсолютный размер эффекта между началом и концом исследования, D_3 (95 % ДИ)	Стандартизированный размер эффекта между началом и концом исследования, индекс g_3 Коэна (95 % ДИ)
Начало исследования, n (%) <i>Experimental group, start of the study, n (%)</i>	Да 28 (93 %)	14	14	-0,46 (-0,65; -0,29) $p < 0,001$	0,50 (0,28; 0,50)
	Нет 2 (7 %)	0	2		
Контрольная группа <i>Control group</i>		Конец исследования, n (%) <i>End of the study, n (%)</i>		Эффекты <i>Effects</i>	
		Да 20 (67 %)	Нет 10 (33 %)	Абсолютный размер эффекта между началом и концом исследования, D_k (95 % ДИ)	Стандартизированный размер эффекта между началом и концом исследования, индекс g_k Коэна (95 % ДИ)
Начало исследования, n (%) <i>Control group, start of the study, n (%)</i>	Да 14 (47 %)	13	1	0,20 (0,03; 0,37) $p = 0,033$	0,38 (0,03; 0,48)
	Нет 16 (53 %)	7	9		

Видно, что лечение преимущественно назначалось пациентам с наличием бинарного признака (рефлюкса) при отношении шансов 15, $p < 0,001$. В конце исследования группы в процентном соотношении по бинарной переменной сравнивались ($p = 0,192$). Однако данное исследование является квазиэкспериментальным, поэтому необходимо учитывать состояние пациента не только в конце исследования, но и в его начале.

Для расчета размера эффекта с учетом начальных условий исследования следует оценить разницу в эффектах. Сначала рассчитаем размеры эффектов в группах как различия в пропорциях до начала и после окончания исследования с учетом повторных измерений.

Для определения вариации оценки абсолютного эффекта как пропорции в группе используем формулу Мак-Немара для парных пропорций [26]: $Var = (p_{0 \rightarrow 1} + p_{1 \rightarrow 0}) - (p_{0 \rightarrow 1} - p_{1 \rightarrow 0})^2$.

Стандартная ошибка составит $se = \sqrt{\frac{Var}{n}}$, где $p_{0 \rightarrow 1}, p_{1 \rightarrow 0}$ – доли несогласованных пар; n – размер выборки.

Абсолютное значение различия в эффектах («различия в различиях») оценивается как разность размеров эффектов в группах. Однако с учетом того, что в одной из групп эффект отрицательный, будем использовать для расчета формулы, приведенные ниже.

Для нахождения абсолютного размера эффекта «различия в различиях» воспользуемся формулой $\Delta D = D_3 - D_k$, стандартная ошибка по правилу распространения ошибки составит $se = \sqrt{se_3^2 + se_k^2}$. Тогда ДИ может быть рассчитан как $\Delta D \pm z_{1-\alpha/2} \times se$, где $z_{1-\alpha/2}$ – значение квантиля стандартного нормального распределения, α – уровень ошибки первого рода.

Для определения стандартизированного размера эффекта «различия в различиях» применим формулу для индекса h Коэна с учетом знака полученного эффекта в группе:

$$h = 2 \times \left(\text{sgn}(D_s) \times \arcsin \sqrt{|D_s|} - \text{sgn}(D_k) \times \arcsin \sqrt{|D_k|} \right).$$

Для нахождения стандартной ошибки результата углового преобразования воспользуемся формулой $se = \sqrt{\frac{1}{4n}}$ [27] и по правилу распространения ошибки рассчитаем ошибку стандартизированного размера эффекта «различия в различиях».

Результаты расчета абсолютных и стандартизированных размеров эффектов в группах приведены в табл. 7, результаты расчета «различия в различиях» – в табл. 8.

Таблица 8
Оценка размера эффекта «различия в различиях» для бинарной переменной

Table 8
Estimate of the “differences in differences” effect size for binary variable

Экспериментальная группа, абсолютный размер эффекта, D_s (95 % ДИ) <i>Experimental group, absolute effect size, D_s (95 % CI)</i>	Контрольная группа, абсолютный размер эффекта, D_k (95 % ДИ) <i>Control group, absolute effect size, D_k (95 % CI)</i>	Абсолютный размер эффекта «различия в различиях», ΔD (95 % ДИ) <i>Absolute “difference-in-difference” distance, ΔD (95 % CI)</i>	Стандартизированный размер эффекта «различия в различиях», индекс h Коэна (95 % ДИ) <i>Standardized “difference-in-difference” distance, Cohen’s h (95 % CI)</i>
-0,46 (-0,65; -0,29)	0,20 (0,03; 0,37)	-0,67 (-0,91; -0,42)	-2,43 (-2,68; -2,18)

Интерпретацию результатов анализа можно выбрать согласно табл. 3 и 4.

Закключение. Аналитический метод DID является достаточно популярным на практике для оценки результатов с использованием данных наблюдений. Фактически этот метод пытается имитировать РКИ, изучая дифференциальное влияние воздействия (лечения, иного вмешательства) в естественных экспериментах. В своей канонической форме метод DID определяет средний эффект воздействия на испытуемых, участвующих в эксперименте, путем сравнения разницы в исходах до и после лечения двух групп: получающей и не получающей воздействие (экспериментальная и группа сравнения соответственно). Чтобы придать причинно-следственную интерпретацию оценкам метода DID, используется безусловное предположение о параллельных трендах: в отсутствие воздействия средние результаты для групп эксперимента и сравнения следовали бы с течением времени параллельными путями. Хотя данное предположение принципиально не поддается проверке, его правдоподобие обычно подвергается сомнению, если наблюдаемые характеристики, которые, как считается, связаны с эволюцией исхода, не сбалансированы между экспериментальной группой и группой сравнения. В таких случаях исследователи обычно включают ковариаты до лечения в анализ DID и считают, что предположения о трендах могут быть удовлетворены только после включения этих ковариат в более сложные модели оценки результатов [28, 29], которые опираются на линейную регрессию с взаимодействием и контролем ковариат и чаще применяются в эконометрике и социологии при анализе панельных данных. Переменная отклика (исхода) в таких моделях имеет количественную природу.

Медицинские данные не часто отвечают предположениям, лежащим в основе линейной регрессии, поэтому для оценки результатов могут выполняться более понятные расчеты с учетом баланса групп по ковариатам, принятым в клинических исследованиях. Таким образом, алгоритмы подбора групп, например propensity score matching (PSM) [30], могут быть использованы до начала анализа по методу DID.

Более того, чаще исход фиксируется как бинарная переменная, и эффект на основе, например, логистической регрессии (отношения шансов) не дает такой же клинической картины результатов лечения, как обычные пропорции.

В клинической практике размер эффекта играет важную роль. На основе результатов исследований принимаются решения, которые должны вести к более эффективным и качественным тактикам лечения, к выбору оптимальных подходов и получению клинически благоприятных исходов заболевания.

Статистические методы оценки результатов применения определенных методов и способов лечения пациента в конечном счете должны давать практикующему врачу понятные данные об эффектах лечения (воздействия). Безусловно, абсолютный размер эффекта воспринимается врачом легче, чем стандартизированный, однако для понимания того, насколько велик размер эффекта, необходима его стандартизация. Это помогает проводить мультицентровые исследования, сравнения различных тактик и т. п. Насколько важен результат различий, может решить только специалист в предметной области.

References

1. Lederer D. J., Bell S. C., Branson R. D., Chalmers J. D., Marshall R., ..., Vincent J.-L. Control of confounding and reporting of results in causal inference studies. Guidance for authors from editors of respiratory, sleep, and critical care journals. *Annals of the American Thoracic Society*, 2019, vol. 16, no. 1, pp. 22–28.
2. Fletcher R. H., Fletcher S. W., Wagner E. H. *Clinical Epidemiology: The Essentials*, 3rd ed. Baltimore, Williams & Wilkins, 1996, 276 p.
3. Campbell D. T., Riecken H. W. Quasi-experimental design. *International Encyclopedia of the Social Sciences*, 1968, vol. 5, no. 3, pp. 259–263.
4. Pearl J. Causal inference in statistics: An overview. *Statistics Surveys*, 2009, vol. 3, pp. 96–146.
5. Shadish W. R., Luellen J. K. Quasi-experimental design. *Handbook of Complementary Methods in Education Research*. Mahwah, Routledge, 2012, pp. 539–550.
6. Kelley K., Preacher K. J. On effect size. *Psychological Methods*, 2012, vol. 17, no. 2, pp. 137–152.
7. Hodges J. L., Lehmann E. L. Estimates of location based on rank tests. *The Annals of Mathematical Statistics*, 1963, vol. 34, pp. 598–611.
8. Hollander M., Wolfe D. A., Chicken E. *Nonparametric Statistical Methods*, 3rd ed. John Wiley & Sons, 2013, 848 p.
9. Cohen J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. Hillsdale, Lawrence Earlbaum Associates, 1988, 590 p.
10. Glass G. V. Primary, secondary, and meta-analysis of research. *Educational Researcher*, 1976, vol. 5, no. 10, pp. 3–8.
11. Hedges L. V. Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics*, 1981, vol. 6, no. 2, pp. 107–128.
12. Cliff N. *Ordinal Methods for Behavioral Data Analysis*, 1st ed. Psychology Press, 1996, 212 p.
13. Cureton E. E. Rank-biserial correlation. *Psychometrika*, 1956, vol. 21, no. 3, pp. 287–290.
14. Cureton E. E. Rank-biserial correlation when ties are present. *Educational and Psychological Measurement*, 1968, vol. 28, no. 1, pp. 77–79.
15. Glass G. V. Note on rank biserial correlation. *Educational and Psychological Measurement*, 1966, vol. 26, no. 3, pp. 623–631.
16. Willson V. L. Critical values of the rank-biserial correlation coefficient. *Educational and Psychological Measurement*, 1976, vol. 36, no. 2, pp. 297–300.
17. Sawilowsky S. S. New effect size rules of thumb. *Journal of Modern Applied Statistical Methods*, 2009, vol. 8, no. 2, pp. 597–599.
18. Lovakov A., Agadullina E. R. Empirically derived guidelines for effect size interpretation in social psychology. *European Journal of Social Psychology*, 2021, vol. 51, no. 3, pp. 485–504.
19. Gignac G. E., Szodorai E. T. Effect size guidelines for individual differences researchers. *Personality and Individual Differences*, 2016, vol. 102, pp. 74–78.
20. Funder D. C., Ozer D. J. Evaluating effect size in psychological research: Sense and nonsense. *Advances in Methods and Practices in Psychological Science*, 2019, vol. 2, no. 2, pp. 156–168.
21. Evans J. D. *Straightforward Statistics for the Behavioral Sciences*. Thomson Brooks/Cole Publishing Co, 1996, 624 p.
22. Hess M. R., Kromrey J. D. Robust confidence intervals for effect sizes: A comparative study of Cohen's d and Cliff's delta under non-normality and heterogeneous variances. *Annual meeting of the American Educational Research Association*, 2004, vol. 1.

23. Romano J., Kromrey J., Coraggio J., Skowronek J. Appropriate statistics for ordinal level data: Should we really be using t-test and Cohen's d for evaluating group differences on the NSSE and other surveys. *Annual meeting of the Florida Association of Institutional Research*, 2006, vol. 177, p. 34.

24. Marfo P., Okyere G. A. The accuracy of effect-size estimates under normals and contaminated normals in meta-analysis. *Heliyon*, 2019, vol. 5, no. 6, p. e01838.

25. Couper M. P., Tourangeau R., Conrad F. G., Singer E. Evaluating the effectiveness of visual analog scales: A web experiment. *Social Science Computer Review*, 2006, vol. 24, no. 2, pp. 227–245.

26. McNemar Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 1947, vol. 12, no. 2, pp. 153–157.

27. Anscombe F. J. The transformation of Poisson, binomial and negative-binomial data. *Biometrika*, 1948, vol. 35, no. 3/4, pp. 246–254.

28. Abadie A. Semiparametric difference-in-differences estimators. *The Review of Economic Studies*, 2005, vol. 72, no. 1, pp. 1–19.

29. Sant'Anna P. H. C., Zhao J. Doubly robust difference-in-differences estimators. *Journal of Econometrics*, 2020, vol. 219, no. 1, pp. 101–122.

30. Rosenbaum P. R., Rubin D. B. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 1983, vol. 70, no. 1, pp. 41–55.

Информация об авторе

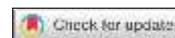
Красько Ольга Владимировна, кандидат технических наук, доцент, ведущий научный сотрудник, Объединенный институт проблем информатики Национальной академии наук Беларуси.
E-mail: krasko@newman.bas-net.by
<https://orcid.org/0000-0002-4150-282X>

Information about the author

Olga V. Krasko, Ph. D. (Eng.), Assoc. Prof., Leading Researcher, The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.
E-mail: krasko@newman.bas-net.by
<https://orcid.org/0000-0002-4150-282X>

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

INTELLIGENT SYSTEMS



УДК 004.932

<https://doi.org/10.37661/1816-0301-2023-20-1-40-54>Оригинальная статья
Original Paper

Увеличение точности реидентификации людей на основе двухэтапного обучения сверточных нейронных сетей и аугментации

С. А. Игнатьева[✉], Р. П. Богуш*Полоцкий государственный университет имени Евфросинии Полоцкой,
ул. Блохина, 29, Новополоцк, 211440, Беларусь*[✉]E-mail: s.ignatieva@pdu.by

Аннотация

Цели. Основной целью исследования является повышение точности повторной (ре-) идентификации людей в распределенных системах видеонаблюдения.

Методы. Применяются методы машинного обучения.

Результаты. Представлена технология двухэтапного обучения сверточных нейронных сетей (СНС), отличающаяся использованием аугментации изображений для предварительного этапа и точной настройки весовых коэффициентов на основе исходного набора изображений. На первом этапе обучение осуществляется на аугментированных данных, на втором этапе выполняется точная настройка СНС на исходных изображениях, что способствует повышению эффективности реидентификации за счет уменьшения потерь при обучении. Применение на двух этапах разных данных не позволяет СНС запоминать тренировочные примеры, тем самым предотвращая переобучение.

Предложенный метод расширения набора данных для обучения отличается тем, что совмещает циклический сдвиг пикселей изображения, исключение цветности и замещение одного фрагмента изображения уменьшенной копией другого из пакета, подаваемого на вход СНС. При использовании этого метода аугментации можно увеличить разнообразие обучающих данных, что повысит робастность СНС ко многим факторам: перекрытию людей, изменению освещенности, уменьшению разрешения изображения, зависимости от местоположения отличительных особенностей объекта интереса.

Заключение. Применение технологии двухэтапного обучения и предложенного метода аугментации данных позволяет повысить точность повторной идентификации людей для разных СНС и наборов данных в метриках: Rank1 на 4–21 %, mAP на 10–31 %, mINP на 39–60 %.

Ключевые слова: реидентификация людей, сверточные нейронные сети, предварительное обучение, точная настройка, расширение обучающей выборки

Для цитирования. Игнатьева, С. А. Увеличение точности реидентификации людей на основе двухэтапного обучения сверточных нейронных сетей и аугментации / С. А. Игнатьева, Р. П. Богуш // Информатика. – 2023. – Т. 20, № 1. – С. 40–54. <https://doi.org/10.37661/1816-0301-2023-20-1-40-54>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 05.10.2022
Подписана в печать | Accepted 25.11.2022
Опубликована | Published 29.03.2023

Improving person re-identification based on two-stage training of convolutional neural networks and augmentation

Sviatlana A. Ihnatsyeva[✉], Rykhard P. Bohush

*Euphrosyne Polotskaya State University of Polotsk,
st. Blokhina, 29, Novopolotsk, 211440, Belarus*

[✉]*E-mail: s.ignatieva@pdu.by*

Abstract

Objectives. The main goal is to improve person re-identification accuracy in distributed video surveillance systems.

Methods. Machine learning methods are applied.

Result. A technology for two-stage training of convolutional neural networks (CNN) is presented, characterized by the use of image augmentation for the preliminary stage and fine tuning of weight coefficients based on the original images set for training. At the first stage, training is carried out on augmented data, at the second stage, fine tuning of the CNN is performed on the original images, which allows minimizing the losses and increasing model efficiency. The use of different data at different training stages does not allow the CNN to remember training examples, thereby preventing overfitting.

Proposed method as expanding the training sample differs as it combines an image pixels cyclic shift, color exclusion and fragment replacement with a reduced copy of another image. This augmentation method allows to get a wide variety of training data, which increases the CNN robustness to occlusions, illumination, low image resolution, dependence on the location of features.

Conclusion. The use of two-stage learning technology and the proposed data augmentation method made it possible to increase the person re-identification accuracy for different CNNs and datasets: in the Rank1 metric by 4–21 %; in the mAP by 10–31 %; in the mINP by 39–60 %.

Keywords: Person re-identification, convolutional neural network, pre-train, fine tuning, augmentation

For citation. Ihnatsyeva S. A., Bohush R. P. *Improving person re-identification based on two-stage training of convolutional neural networks and augmentation*. Informatika [Informatics], 2023, vol. 20, no. 1, pp. 40–54 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-1-40-54>

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

Введение. СНС являются наиболее эффективным инструментом извлечения признаков изображений людей для реидентификации в многокамерных системах видеонаблюдения. Известно, что в значительной мере на эффективность работы СНС оказывают влияние количество изображений в обучающей выборке и их разнообразие. Недостаток тренировочных данных может спровоцировать переобучение СНС, запоминание исходных данных и неспособность к обобщению признаков в целом. Решением этой проблемы может служить концепция переноса обучения (transfer learning). В таком случае СНС обучается на наборе данных достаточно большого размера, например ImageNet (3,2 млн аннотированных изображений) [1] или LUPerson (около 4 млн изображений более 200 тыс. человек) [2]. После обучения возможны два сценария (URL: https://pytorch.org/tutorials/beginner/transfer_learning_tutorial.html): использование обученной модели в качестве сети с фиксированными связями (fixed feature extractor) и выполнение точной настройки (fine tuning). В первом случае обучение СНС на исследуемом наборе данных выполняется только для полносвязного слоя, а весовые коэффициенты скрытых слоев остаются без изменений. При точной настройке СНС скрытые слои обучаются с меньшей скоростью, чем последний слой, который является классификационным. Таким образом, СНС обучается в два этапа. На первом этапе выполняется предварительное обучение на большом наборе данных, а на втором этапе веса корректируются на обучающей выборке для решения конкретной задачи, например обнаружения и классификации объектов разных классов, повторной идентификации человека [3].

Эффективность предобученных моделей СНС подтверждена результатами исследований, опубликованными в работах [2–4], и обусловлена тем, что на этапе предварительного обучения выделяются признаки, несущие основную информацию об изображениях: вертикальные и горизонтальные линии, цвет, текстура и форма объектов и др. Использование предварительно обученных нейронных сетей широко распространено не только для решения задач классификации и распознавания объектов на изображениях, но и для сегментации изображений, поиска ключевых точек и обработки текста (URL: <https://pytorch.org/vision/stable/models.html>). Применение таких сетей ограничивается наличием предварительно обученных моделей для конечного числа архитектур СНС и высоким потреблением вычислительных ресурсов и времени.

Другим решением проблемы переобучения является регуляризация. В машинном обучении под регуляризацией понимают добавление ограничений к архитектуре нейронной сети, функции потерь или к обучающим наборам данных [5]. Примерами регуляризации являются: прореживание нейронных связей СНС [6]; использование разных функций активации [7]; L1- и L2-регуляризации (URL: <https://www.analyticssteps.com/blogs/l2-and-l1-regularization-machine-learning>); аугментация данных, при которой часть изображения может быть удалена или замещена другой информацией [5, 8, 9]. Аугментация применяется для увеличения разнообразия обучающей выборки на основе имеющихся данных [9] за счет преобразований изображений. При этом могут использоваться такие преобразования, как изменение яркости и контраста, зеркальное отражение, поворот, размытие и др. Количество изображений при аугментации не увеличивается, а разнообразие достигается за счет того, что на различных эпохах обучения к изображениям применяются разные преобразования. Расширение обучающей выборки позволяет улучшить обобщающую способность СНС и увеличить точность работы, в том числе при различных мешающих факторах, таких как высокая вариация освещения, низкое разрешение, перекрытие объектов и др.

Анализ существующих методов аугментации. При увеличении обучающей выборки преобразование выполняется для одного изображения [8] или для пакета изображений [9–11], используемых при обучении СНС. Изображения группируются в пакеты для уменьшения времени обучения за счет распараллеливания вычислений. Размер пакета является гиперпараметром и, как правило, изменяется в диапазоне от 16 до 64. Следует отметить, что уменьшение размера пакета может приводить к взрывным градиентам и неспособности СНС к обобщению, а увеличение – к снижению точности работы СНС.

В работе [8] для аугментации предлагается метод *random erasing*, при котором изображения выбираются для преобразований из исходного набора данных с применением генератора псевдослучайных чисел (ГПСЧ). На основе ГПСЧ также определяются размер и координаты прямоугольного фрагмента изображения, пиксели которого заполняются нулевыми или случайными значениями (рис. 1, *a*). Поэтому при загрузке пакета изображений на разных эпохах обучения одно и то же изображение может быть представлено как в исходном виде, так и с разными измененными фрагментами, что позволяет повысить устойчивость СНС к окклюзиям, однако при этом часть информации теряется.

Метод регуляризации с использованием аугментации *CutMix* основан на применении преобразования в пакете изображений, которое предполагает замещение области фрагмента одного изображения областью такого же размера другого изображения из этого же пакета [9] (рис. 1, *b*). Применение *CutMix* позволяет СНС повысить устойчивость к перекрытию и при этом сохранить ту часть информации, которая теряется при использовании метода *random erasing*. В алгоритме *Mixed Single Thumbnail (MST)* [10] фрагмент изображения замещается на уменьшенную копию фиксированного размера другого изображения из пакета (рис. 1, *c*). Это позволяет одновременно изучать два изображения при извлечении признаков, повышает устойчивость к окклюзиям и низкому разрешению изображений. Одновременно два изображения из пакета обрабатываются и при использовании аугментации *mixup* [11], однако их объединение выполняется с помощью линейной интерполяции (рис. 1, *d*). Такой подход позволяет улучшить обобщающую способность обученной СНС и снижает чувствительность к запоминанию ошибочных меток (идентификаторов людей), которые могут встречаться в существующих наборах данных.



a)



b)



c)



d)

Рис. 1. Примеры использования методов аугментации данных: a) random erasing; b) CutMix; c) MST; d) mixup
Fig. 1. Examples of using data augmentation methods: a) random erasing; b) CutMix; c) MST; d) mixup

Результаты исследований, представленные в работах [12, 13], показывают, что при обучении для задачи классификации нейронная сеть больше внимания уделяет текстуре объекта, а не его форме. Так, в [12] отмечено, что если образ кота заполнить текстурой кожи слона, то СНС распознает класс обнаруженного объекта как «слон», тогда как человек будет считать, что это кот, т. е. для человека решающее значение имеет именно форма объекта, а не его текстура.

Возможны ситуации, когда люди носят одежду одного фасона и разного цвета, причем даже у одного человека цвет одежды при изменении освещения может различаться. В таком случае при реидентификации важное значение будет иметь силуэт человека. Поэтому для увеличения значимости формы объекта на предсказание СНС в статье [13] предлагается исключать цветность на прямоугольных фрагментах изображения.

В работе [14] предлагается метод аугментации, состоящий из нескольких этапов. Вначале СНС обучается на исходных данных до того момента, когда значения функции потерь перестают уменьшаться. После обучения на изображениях определяются области, которые являются наиболее важными для принятия решения нейронной сетью. Для этого методом скользящего окна исключаются разные участки изображения и оценивается влияние данного шага на ре-

зультат. При обнаружении фрагмента изображения, без которого сеть не может верно определить идентификационную метку человека, область исключается и изображение добавляется к обучающей выборке. После расширения обучающей выборки изображениями с удаленными фрагментами продолжается обучение СНС. Применение такого метода аугментации дает возможность обучить сеть нахождению дополнительных отличий людей, позволяющих повысить точность повторной идентификации.

Методы random erasing и CutMix повышают устойчивость лишь к окклюзиям, а MST – к окклюзиям и разрешению изображения. Метод, предложенный в работе [14], требует дополнительных затрат на поиск изображений, которые будут использоваться для аугментации.

Для исключения указанных недостатков существующих подходов и повышения эффективности повторной идентификации людей в системах распределенного видеонаблюдения предлагается технология двухэтапного обучения с помощью нового метода аугментации данных.

Технология двухэтапного обучения с аугментацией данных. Предлагается использовать подход, включающий технологию двухэтапного обучения СНС и новый метод аугментации данных. При этом на первом этапе выполняется предварительное обучение на аугментированных данных, а на втором этапе применяется точная настройка СНС, при которой весовые коэффициенты корректируются на исходных изображениях с меньшей скоростью обучения. Это позволяет извлекать наиболее надежные отличительные признаки. Для увеличения разнообразия обучающей выборки на основе имеющихся данных предлагается использовать циклический сдвиг пикселей по вертикали и горизонтали, исключение цветности и замещение фрагмента другим уменьшенным изображением. Циклический сдвиг и исключение цветности применяются к отдельным изображениям, а замещение фрагмента осуществляется уменьшенной копией другого из пакета, подаваемого на вход СНС.

Предварительное обучение модели и точная настройка. Для снижения значения функции потерь при обучении, высокие уровни которой вызваны неправдоподобностью обучающих данных по отношению к тестовым данным, и повышения точности повторной идентификации людей предлагается применять двухэтапное обучение. При этом используются следующие особенности: скорость тренировки классификационного слоя выше скорости всех остальных слоев СНС, на первом этапе осуществляется предварительное обучение с применением аугментации данных, на втором этапе тренировка СНС продолжается только на исходном наборе данных.

Циклический сдвиг изображения. В большинстве случаев голова человека находится в верхней части изображения, а ноги – в нижней. В процессе обучения СНС запоминает эти особенности. При обнаружении человека детектором часть его изображения может не попасть в ограничительную рамку или быть скрыта другими объектами. Если человек наклонился, присел, поднял руки вверх, то нижняя или верхняя часть его тела оказывается скрыта, поэтому выделение признаков СНС выполняется не совсем корректно. Применение циклического сдвига позволяет получить изображение человека, на котором отдельные части тела могут оказаться не на своих местах и отдельно от самого человека. Например, голова будет находиться в нижней части изображения, ступни – выше плеч, а правая рука – слева. Представляется, что такой подход обеспечит повышение устойчивости СНС к местоположению отличительных характеристик фигуры.

Циклический сдвиг применяется к некоторой части изображений (p) из всего набора исходных данных до начала обучения как гиперпараметр наряду с размером пакета B , количеством эпох $K_{\text{эпох}}$ и скоростью обучения lr . Для реализации выбора изображений (листинг), к которым применяется данное преобразование, используется ГПСЧ, с помощью которого определяется число r , принимающее значение от нуля до единицы. Пороговое значение

$$t = \frac{p}{100} \quad (1)$$

служит для определения необходимости применения циклического сдвига к изображению. Если $r < t$, то входное изображение видоизменяется.

Листинг. Алгоритм выбора изображений из обучающей выборки, для которых выполняется преобразование

```

1  Input:  $I_{in}$  – входное изображение
2       $p$  – процент изображений, к которым применяется преобразование
3      Transform – преобразование
4  Output:  $I_{out}$  – выходное изображение
5  Initialization:  $r \leftarrow \text{Rand}(0,1)$ ;
6       $t = p/100$ 
7  if  $r > t$  then
8       $I_{out} \leftarrow I_{in}$ ;
9      return  $I_{out}$ .
10 else
11      $I_{out} \leftarrow \text{Transform}(I_{in})$ ;
12     return  $I_{out}$ .
13 end
    
```

На рис. 2 показаны результаты циклического сдвига по вертикали и горизонтали на некоторое количество пикселей изображений.



Рис. 2. Примеры циклического сдвига пикселей изображений по вертикали и горизонтали

Fig. 2. Examples of image pixel's cyclic shift vertically and horizontally

Исключение цветности. Цвет одежды является важной характерной особенностью, играющей значительную роль при реидентификации людей. Однако цветопередача различных видеокамер и освещенность сцены видеонаблюдения могут значительно исказить цветовые характеристики одного и того же объекта: на рис. 3, *a* при разном освещении платье женщины воспроизводится серым цветом вместо белого; на рис. 3, *b* с одной камеры майка фиксируется голубой, с другой – бирюзовой; пример на рис. 3, *c* демонстрирует, как коричневые шорты меняют цвет ближе к черному. В этом случае наиболее важными представляются такие признаки, как силуэт человека или форма одежды. Поэтому исключение цветности может повысить устойчивость к уровням освещения за счет увеличения влияния других признаков изображения человека.



Рис. 3. Примеры изображений с видеокамер при разном освещении

Fig. 3. Color distortion examples in different lighting conditions

Замещение фрагмента одного изображения уменьшенной копией другого из пакета.

Добавление копии одного изображения на другое выполняется методом, похожим на метод из работы [10]. Его отличие заключается в том, что уменьшенные копии имеют нефиксированный размер: изменяют значения в диапазоне от $0,125S$ до $0,25S$, где S – площадь исходного изображения.

На входе СНС изображение x_i имеет метку (идентификатор) y_i , где i – индекс изображения в пакете. На каждом изображении (x_i, y_i) фрагмент замещается на $(x_j^{\text{mini}}, y_j^{\text{mini}})$. Индекс j – порядковый номер изображения в пакете, которое будет уменьшено; x_j^{mini} – уменьшенная копия x_j ; y_j^{mini} – соответствующий идентификатор. Для определения индекса j для x_j^{mini} , замещающего фрагмент на i -м изображении, с помощью ГПСЧ формируется массив J размером B , заполненный числами от 1 до B , где B – количество изображений в пакете без повторений. Полученное в результате преобразования изображение можно описать следующим образом:

$$\tilde{x} = M \odot x_i + \Phi(x_j^{\text{mini}}), \quad \tilde{y} = \lambda y_i + (1 - \lambda) y_j^{\text{mini}}, \quad (2)$$

где (\tilde{x}, \tilde{y}) – аугментированное изображение; $M \in \{0,1\}^{W \times H}$ – бинарная маска, определяющая местоположение фрагмента, который будет заменен на уменьшенное изображение из пакета; W и H – ширина и высота x_i ; \odot – поэлементное умножение; $\Phi(\cdot)$ – операция заполнения изображения.

Для бинарной маски необходимо сформировать координаты ограничительной рамки вырезанного фрагмента $E = (r_x, r_y, r_w, r_h)$, где r_x, r_y – координаты левого нижнего угла; r_w и r_h – высота и ширина удаленного участка, которые определяются с помощью ГПСЧ таким образом, что r_h составляет $(0,25 \dots 0,5 H)$, где H – высота исходного изображения. Ширина определяется как $r_w = \left\lfloor \frac{r_h}{\eta} \right\rfloor$, где $\eta = \left\lfloor \frac{H}{W} \right\rfloor$ – соотношение сторон исходного изображения.

Область размером E на x_i удаляется и заполняется уменьшенной копией x_j^{mini} такого же размера, что и E . Для $\Phi(\cdot)$ формируется бинарная маска $\tilde{M} = 1 - M$, заполненная единицами, что позволяет разместить в удаленной области x_i уменьшенную копию x_j^{mini} . При таком методе аугментации на каждом изображении присутствуют два человека с соответствующими метками y_i и y_j^{mini} . Метка \tilde{y} будет зависеть от соотношения площади уменьшенного изображения к исходному, определяемого коэффициентом λ :

$$\lambda = 1 - \frac{r_w \cdot r_h}{W \cdot H}. \quad (3)$$

Рассмотрим пример применения предлагаемого метода аугментации для пакета из восьми изображений. Изображения на рис. 4, *a–d, f* и *h* содержат циклический сдвиг; на изображении на рис. 4, *b* отсутствует информация о цвете. Для всех изображений выполнена замена фрагмента на уменьшенное изображение из пакета. Уменьшенные изображения содержат циклический сдвиг в примерах на рис. 4, *a, d–h*.



Рис. 4. Примеры аугментации данных с циклическим сдвигом пикселей, добавлением изображений в градациях серого и замещением фрагмента изображения уменьшенной копией
Fig. 4. Examples of data augmentation with cycle shift pixel, adding grayscale images and replacing image fragment with a reduced copy

Представленные на рис. 4 изображения в значительной мере отличаются от реальных, полученных при видеонаблюдении. Предполагается, что с их помощью можно обучить СНС обнаруживать признаки независимо от их расположения, целостности частей тела человека, попавших в кадр, цвета одежды или разрешения изображения и таким образом повысить чувствительность СНС к форме (силуэту) объекта, увеличить устойчивость к окклюзиям, уменьшить зависимость от уровня освещенности.

Оценка точности работы алгоритма повторной идентификации людей. Для оценки результатов повторной идентификации использовались такие метрики, как mAP, Rank1 и mINP [15].

Метрика mAP является оценкой точности алгоритма повторной идентификации, отражающей среднее значение средних точностей для всех запросов, и рассчитывается по формуле

$$mAP = \frac{1}{Q} \sum_{i=1}^Q AP_i, \quad (4)$$

где Q – общее число запросов; $AP = \frac{\sum \text{precision}}{I}$ – средняя точность для каждого i -го запроса;

I – число изображений в тестовой выборке; $\text{precision} = \frac{TP}{TP + FP}$ – точность запроса; TP – количество верноположительных предсказаний запроса; FP – количество ложноположительных предсказаний запроса.

Группа метрик RankN характеризует качество ранжирования и показывает проценты от общего числа запросов, для которых верный выданный результат был среди первых N полученных. Соответственно, метрика Rank1 показывает число процентов запросов, для которых идентификатор первого изображения-кандидата совпадает с идентификатором запроса. Если $N = 5$, то Rank5 показывает число процентов запросов, для которых среди первых пяти выданных изображений-кандидатов было определено верное решение. Для вычисления RankN определяется отношение суммы числа запросов, для которых верное решение было найдено среди первых выданных результатов, к общему числу запросов Q :

$$\text{RankN} = \frac{\sum K_{i,N}}{Q}, \quad (5)$$

где i – номер запроса; $K_{i,N}$ – i -й запрос, для которого верное решение было получено в ряду первых N выданных результатов.

В системах повторной идентификации приоритетно, чтобы верные предсказания находились в начале ранжированного списка и имели как можно меньше ложных предсказаний. Метрики RankN и mAP не отражают сложность поиска правильно идентифицированных изображений людей для поступающего запроса, и при одинаковых показателях Rank значения точности AP могут отличаться. Для учета поиска наиболее сложных правильных предсказаний используется метрика mINP (mean Inverse Negative Penalty) [15], позволяющая исключить доминирование легких совпадений, которые влияют на метрики Rank и mAP. Для вычисления mINP вводятся дополнительные метрики: NP (Negative Penalty) – отрицательный штраф, назначаемый за неверные предсказания для i -го запроса и уменьшающий вероятность правильной реидентификации при неправильном нахождении самого сложного совпадения; INP (Inverse Negative Penalty) – обратная величина для NP, рост которой свидетельствует об улучшении производительности системы. Метрика mINP характеризует среднее значение INP для всех запросов и вычисляется по формуле

$$mINP = \frac{1}{Q} \sum_i (1 - NP_i) = \frac{1}{Q} \sum_i \left(1 - \frac{R_i^{\text{hard}} - |G_i|}{R_i^{\text{hard}}} \right) = \frac{1}{Q} \sum_i \frac{|G_i|}{R_i^{\text{hard}}}, \quad (6)$$

где Q – общее количество запросов; $NP_i = \frac{R_i^{\text{hard}} - |G_i|}{R_i^{\text{hard}}}$ – отрицательный штраф; R_i^{hard} – позиция самого сложного верного предсказания; $|G_i|$ – общее количество верных предсказаний для запроса.

Результаты экспериментальных исследований. Для определения наиболее эффективного метода аугментации данных использовался алгоритм повторной идентификации (URL: https://github.com/layumi/Person_reID_baseline_pytorch), реализованный на фреймворке pyTorch. Оптимальные значения скорости обучения и размера пакета определялись эмпирически, наилучшие показатели точности повторной идентификации для модели получены при $lr = 0,07$ и $B = 16$. Применяемый алгоритм повторной идентификации включает метод регуляризации dropout [6] и такие преобразования, как заполнение по границам изображения (padding), отражение по горизонтали (random horizontal flip) и кадрирование (random crop). В процессе обучения используется планировщик скорости, уменьшающий lr в 0,1 раз каждые 40 эпох. Скорость обучения скрытых слоев СНС в 0,1 раза меньше скорости обучения классификационного слоя. Для предложенного подхода в течение первых 45 эпох обучение осуществляется на аугментированных данных, а затем продолжается на исходных изображениях.

Применение аугментированных данных в технологии двухэтапного обучения позволяет повысить точность повторной идентификации по сравнению с базовым подходом и уменьшить временные затраты по сравнению с методом, использующим предварительное обучение на миллионах изображений, обеспечивая при этом сопоставимую точность. Процесс обучения с использованием предложенного подхода может занимать от полутора до шести часов на персональном компьютере с такими основными характеристиками, как Intel Core i5 3.11 GHz, 16 Gb RAM, Nvidia GeForce RTX-3060 6 Gb, в зависимости от набора данных, а предварительное обучение на ImageNet может длиться до нескольких недель (URL: <https://cs231n.github.io/transfer-learning/>). После этого этапа необходимо выполнить точную настройку на наборе данных для повторной идентификации.

В табл. 1 рассматриваются примеры предсказаний алгоритма повторной идентификации с использованием СНС ResNet-50, обученной с помощью различных методов аугментации для двух запросов. Запрос обозначен как query, ошибочные предсказания – false, верные – true.

Анализ данных табл. 1 показал, что для первого запроса при использовании СНС, обученной без аугментации, не было ни одного верного ответа среди пяти первых вариантов предсказаний. При тестировании СНС, для обучения которой применялись методы аугментации random erasing, Cutmix, MixUp и MST, было допущено от двух до четырех ошибок. При использовании предложенного подхода, включающего технологию двухэтапного обучения и аугментацию, была допущена лишь одна ошибка для первого запроса.

Для второго запроса при применении для обучения предложенного подхода не было ни одного ошибочного прогноза. При обучении СНС с аугментацией данных методами random erasing и MST при тестировании получены четыре верных предсказания среди первых пяти результатов. СНС, для обучения которой применялись методы аугментации MixUp и CutMix, допущены две и три ошибки соответственно. Модель, обученная без аугментации, позволила получить одно верное предсказание.

Необходимое количество эпох обучения для обеспечения наибольшей эффективности реидентификации определялось экспериментально. Модель обучалась на протяжении 110 эпох с использованием двухэтапного обучения и предлагаемого метода аугментации. Для каждой пятой эпохи определялась точность работы алгоритма повторной идентификации в метриках Rank1, mAP и mINP. На рис. 5 показаны графики зависимости точности реидентификации от количества эпох обучения с использованием СНС ResNet-50 и набора данных Market-1501.

На графиках, представленных на рис. 4, видно, что наибольшая точность для двухэтапного обучения с применением предлагаемого метода аугментации достигается на 80-й эпохе: Rank1 = 0,8815, mAP = 0,7025, mINP = 0,3506.

Результаты работы алгоритма повторной идентификации людей с использованием ResNet-50, для обучения которой применялись различные методы аугментации

Table 1

The results of the work of the algorithm for person re-identification using ResNet-50, trained by various augmentation methods

Метод аугментации <i>Augmentation method</i>	Запрос 1 <i>Query 1</i>	Результат реидентификации для запроса 1 <i>Re-identification result for query 1</i>	Запрос 2 <i>Query 2</i>	Результат реидентификации для запроса 2 <i>Re-identification result for query 2</i>
Без аугментации <i>Without augmentation</i>				
Random erasing				
Cutmix				
MixUP				
MST				
Предлагаемый подход <i>Proposed approach</i>				

Для определения эффективности применения двухэтапного обучения по сравнению с одноэтапным СНС ResNet-50 обучалась на наборе данных Market-1501 с использованием предложенного метода аугментации на протяжении 80 эпох. Была получена точность Rank1 = 0,8708, mAP = 0,6886, mINP = 0,3401. Это подтверждает, что применение аугментации только на предварительном этапе более эффективно, чем ее применение на протяжении всего процесса обучения.

Для сравнения предлагаемой технологии двухэтапного обучения и метода аугментации с существующими методами, использующими расширение обучающей выборки, применялись СНС ResNet-50 [16] и DenseNet-121 [17]. Для обучения и тестирования использовались наборы данных Market-1501 [18], Duke-MTMC [19] и MSMT17 [20]. Результаты экспериментов представлены в табл. 2.

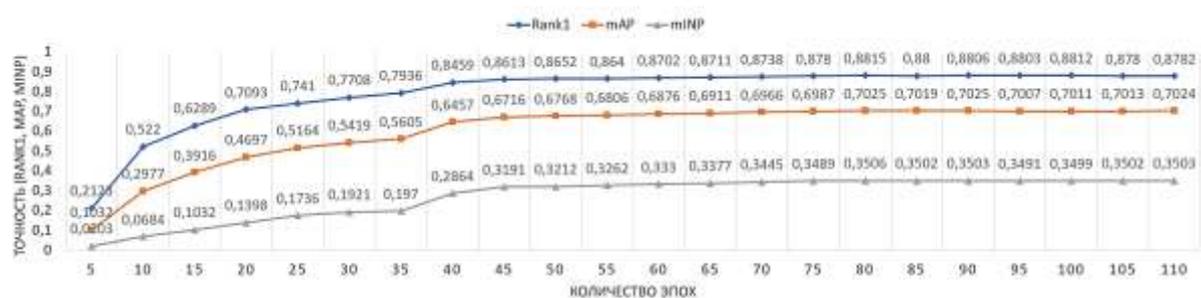


Рис. 5. Зависимость точности повторной идентификации людей в метриках Rank1, mAP и mINP от количества эпох обучения при использовании CNN ResNet-50 и набора данных Market-1501

Fig. 5. Dependence of person re-identification accuracy in the metrics Rank1, mAP and mINP on the training epochs number using CNN ResNet-50 and the Market-1501 dataset

Таблица 2

Сравнение точности методов аугментации для CNN ResNet-50 и DenseNet-121 на наборах данных Market-1501, Duke-MTMC и MSMT17

Table 2

Comparison of the accuracy of augmentation methods for CNN ResNet-50 and DenseNet-121 on Market-1501, Duke-MTMC and MSMT17 datasets

Метод Method	Метрики Metrics	Набор данных для обучения и тестирования Dataset for train and test					
		Market-1501		Duke-MTMC		MSMT17	
		ResNet-50	DenseNet-121	ResNet-50	DenseNet-121	ResNet-50	DenseNet-121
Без аугментации Without augmentation	Rank1	0,8319	0,8376	0,7285	0,7280	0,4954	0,5498
	mAP	0,6108	0,6248	0,5253	0,5316	0,2483	0,2667
	mINP	0,2395	0,2686	0,1453	0,1588	0,251	0,261
Random erasing [8]	Rank1	0,8453	0,8424	0,7361	0,7437	0,4931	0,5476
	mAP	0,6508	0,6490	0,5506	0,5634	0,2469	0,2881
	mINP	0,2773	0,2822	0,1650	0,1775	0,240	0,320
CutMix [9]	Rank1	0,8622	0,8548	0,7424	0,7459	0,4951	0,5500
	mAP	0,6645	0,6632	0,5610	0,5702	0,2562	0,2885
	mINP	0,2792	0,2897	0,1738	0,1778	0,279	0,3185
MixUP [11]	Rank1	0,8480	0,8521	0,7365	0,7428	0,5265	0,5820
	mAP	0,6402	0,6482	0,5401	0,5509	0,2585	0,2985
	mINP	0,2601	0,2802	0,1539	0,1688	0,237	0,310
MST [10]	Rank1	0,8607	0,8607	0,7478	0,7531	0,5071	0,5625
	mAP	0,6757	0,6831	0,5872	0,5911	0,2683	0,3035
	mINP	0,3064	0,3218	0,1957	0,1931	0,318	0,372
Предлагаемый подход Proposed approach	Rank1	0,8815	0,8732	0,7590	0,7666	0,6022	0,6116
	mAP	0,7025	0,6916	0,5823	0,5867	0,3259	0,3366
	mINP	0,3506	0,3431	0,2022	0,1980	0,3950	0,4170

Из данных табл. 2 следует, что предлагаемый подход позволяет получить наилучшие результаты (выделены жирным шрифтом) точности повторной идентификации практически во всех проведенных экспериментах при сравнении с другими методами аугментации. Только для набора данных Duke-MTMC он незначительно уступает методу аугментации MST в метрике mAP для CNN ResNet-50 и DenseNet-121. Наибольшая эффективность предлагаемого подхода отмечается на наборе данных MSMT, так как для ResNet-50 увеличена точность по сравнению с моделью, обученной без аугментации, в метрике Rank1 с 0,4954 до 0,6022, в mAP с 0,2483 до 0,3259 и mINP с 0,251 до 0,395.

Для набора данных Market-1501 значительное увеличение достигнуто по всем трем метрикам оценки точности для СНС ResNet-50: для Rank1 с 0,8319 до 0,8815; mAP с 0,6108 до 0,7025; mINP с 0,2395 до 0,3506.

При сравнении приведенных в табл. 2 результатов экспериментов со значениями, представленными на графике (см. рис. 5), видно, что уже на 50-й эпохе при использовании двухэтапного обучения и предложенного метода аугментации модель достигает значений Rank1 = 0,8652 и mAP = 0,6768. Это превосходит точность СНС, обученной в течение 80 эпох с CutMix (Rank1 = 0,8622, mAP = 0,6645) и MST (Rank1 = 0,8607, mAP = 0,6757). В метрике mINP предлагаемый метод аугментации уже на 45-й эпохе обучения продемонстрировал лучший показатель по сравнению с другими, при которых СНС обучалась на протяжении 80 эпох. Предлагаемый подход подразумевает, что аугментация применяется только на первых 45 эпохах. Таким образом, использование предложенного метода аугментации оказывается более эффективным как для СНС, обученной без расширения обучающей выборки, так и для модели, обученной с использованием других методов аугментации.

Заключение. В работе предложен подход для повышения точности повторной идентификации людей, включающий двухэтапное обучение СНС и аугментацию данных. Технология двухэтапного обучения основана на концепции переноса обучения (transfer learning) и отличается от классического подхода, предполагающего использование для предобучения больших наборов данных, применением на данном этапе наборов, расширенных за счет аугментации, а для точной настройки весовых коэффициентов – применением для обучения изображений из исходного набора. Преимущество предложенной технологии заключается в том, что из-за невысокой вычислительной сложности для аугментации не потребуются значительные вычислительные затраты на выполнение предварительного обучения СНС. Использование на разных этапах обучения различных данных не позволяет СНС запоминать обучающие примеры, тем самым предотвращая переобучение.

В основе предложенного метода аугментации лежат циклический сдвиг пикселей изображения, исключение цветности и замещение фрагмента уменьшенной копией другого изображения. Такой метод аугментации позволяет получить большое разнообразие обучающих данных, что увеличивает робастность СНС к окклюзиям, освещенности, низкому разрешению изображения, зависимости от местоположения отличительных признаков.

Применение технологии двухэтапного обучения и предложенного метода аугментации данных для СНС ResNet-50, обученной на разных наборах данных, позволило повысить точность Rank1 на 4,18–21,55 %, mAP на 10,85–31,25 %, mINP на 39,16–57,37 % по сравнению с СНС, обученной без аугментации. Для СНС DenseNet-121 увеличение точности алгоритма реидентификации людей составило для Rank1 4,25–11,24 %, mAP 10,36–26,21 %, mINP 24,69–60 % при разных наборах данных.

Вклад авторов. С. А. Игнатьева разработала технологию двухэтапного обучения и метод аугментации данных, провела экспериментальные исследования. Р. П. Богуш сформулировал цель и задачи исследования, выполнил анализ и интерпретацию полученных результатов.

Список использованных источников

1. ImageNet: A large-scale hierarchical image database / J. Deng [et al.] // 2009 IEEE Conf. on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009. – Miami, 2009. – P. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
2. Unsupervised pre-training for person re-identification / D. Fu [et al.] // 2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021. – Nashville, 2021. – P. 14745–14754. <https://doi.org/10.1109/CVPR46437.2021.01451>
3. Богатырева, А. А. Исследование способности к transfer learning сверточных нейронных сетей, обученных на ImageNet / А. А. Богатырева, А. Р. Виноградова, С. А. Тихомирова // Междунар. журнал прикладных и фундаментальных исследований. – 2019. – № 7. – С. 106–111.

4. Конарев, Д. И. Повышение точности предварительно обученных нейронных сетей путем тонкой настройки / Д. И. Конарев, А. А. Гуламов // Материалы конф. «Информационные технологии в управлении», Санкт-Петербург, 6–8 окт. 2020 г. – СПб., 2020. – С. 200–212.
5. DeVries, T. Improved Regularization of Convolutional Neural Networks with CutOut / T. DeVries, G. W. Taylor. – 2017. – Mode of access: <https://doi.org/10.48550/arXiv.1708.04552>. – Date of access: 09.08.2022.
6. Dropout: A simple way to prevent neural networks from overfitting / N. Srivastava [et al.] // *J. of Machine Learning Research*. – 2014. – No. 15. – P. 1929–1958. <https://doi.org/10.5555/2627435.2670313>
7. Choice of activation function in convolution neural network for person re-identification in video surveillance systems / H. Chen [et al.] // *Programming and Computer Software*. – 2022. – Vol. 48, no. 5. – P. 312–321. <http://doi.org/10.1134/S0361768822050036>
8. Random Erasing Data Augmentation / Z. Zhong [et al.]. – 2020. – Mode of access: <https://doi.org/10.1609/AAAI.V34I07.7000>. – Date of access: 09.08.2022.
9. CutMix: Regularization strategy to train strong classifiers with localizable features / S. Yun [et al.] // 2019 IEEE/CVF Intern. Conf. on Computer Vision (ICCV), Seoul, Korea (South), 27 Oct. – 2 Nov. 2019. – Seoul, 2019. – P. 6022–6031. <https://doi.org/10.1109/ICCV.2019.00612>
10. Cut-thumbnail: A novel data augmentation for convolutional neural network / T. Xie [et al.] // *Proc. of the 29th ACM Intern. Conf. on Multimedia, Virtual Event, China, 20–24 Oct. 2021. – Virtual Event, China, 2021. – P. 1627–1635. <https://doi.org/10.1145/3474085.3475302>*
11. Mixup: Beyond Empirical Risk Minimization / H. Zhang [et al.]. – 2018. – Mode of access: <https://doi.org/10.48550/arXiv.1710.09412>. – Date of access: 09.08.2022.
12. ImageNet-Trained CNNs are Biased Towards Texture; Increasing Shape Bias Improves Accuracy and Robustness / R. Geirhos [et al.]. – 2019. – Mode of access: <https://doi.org/10.48550/arXiv.1811.12231>. – Date of access: 09.08.2022.
13. Gong, Y. An Effective Data Augmentation for Person Re-identification / Y. Gong, Z. Zeng. – 2021. – Mode of access: <https://doi.org/10.48550/arXiv.2101.08533>. – Date of access: 09.08.2022.
14. Adversarially occluded samples for person re-identification / H. Huang [et al.] // 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018. – Salt Lake City, 2018. – P. 5098–5107. <https://doi.org/10.1109/CVPR.2018.00535>
15. Deep learning for person re-identification: A survey and outlook / M. Ye [et al.] // *IEEE Transactions on Pattern Analysis and Machine Intelligence*. – 2021. – Vol. 44, iss. 6. – P. 2872–2893. <https://doi.org/10.1109/TPAMI.2021.3054775>
16. Deep residual learning for image recognition / K. He [et al.] // 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016. – Las Vegas, 2016. – P. 770–778. <https://doi.org/10.1109/cvpr.2016.90>
17. Huang, G. Densely connected convolutional networks / G. Huang, Z. Liu, K. Q. Weinberger // 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017. – Honolulu, 2017. – P. 2261–2269. <https://doi.org/10.1109/CVPR.2017.243>
18. Scalable person re-identification: A benchmark / L. Zheng [et al.] // 2015 IEEE Intern. Conf. on Computer Vision (ICCV), Santiago, Chile, 7–13 Dec. 2015. – Santiago, 2015. – P. 1116–1124. <https://doi.org/10.1109/ICCV.2015.133>
19. Performance Measures and a Data Set for Multi-target, Multi-camera Tracking / E. Ristani [et al.]. – 2016. – Mode of access: https://doi.org/10.1007/978-3-319-48881-3_2. – Date of access: 09.08.2022.
20. Person transfer GAN to bridge domain gap for person re-identification / L. Wei [et al.] // 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018. – Salt Lake City, 2018. – P. 79–88. <https://doi.org/10.1109/CVPR.2018.00016>

References

1. Deng J., Dong W., Socher R., Li L., ..., Fei-Fei L. ImageNet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009*. Miami, 2009, pp. 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
2. Fu D., Chen D., Bao J., Yang H., Yuan L., ..., Chen D. Unsupervised pre-training for person re-identification. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20–25 June 2021*. Nashville, 2021, pp. 14745–14754. <https://doi.org/10.1109/CVPR46437.2021.01451>

3. Bogatyreva A. A., Vinogradova A. R., Tikhomirova S. A. *Study of the ability of convolution neural networks pretrained on ImageNet to transfer learning*. *Mezhdunarodnyj zhurnal prikladnyh i fundamental'nyh issledovanij [International Journal of Applied and Fundamental Research]*, 2019, no. 7, pp. 106–111 (In Russ.).
4. Konarev D. I., Gulamov A. A. *Improving the accuracy of pretrained neural networks by fine-tuning*. *Materialy konferencii "Informacionnye tehnologii v upravlenii"*, Sankt-Peterburg, 6–8 oktjabrja 2020 g. [*Proceedings of the Conference "Information Technologies in Management", Saint Petersburg, 6–8 October 2020*]. Saint Petersburg, 2020, pp. 200–212 (In Russ.).
5. DeVries T., Taylor G. W. *Improved Regularization of Convolutional Neural Networks with CutOut*, 2017. Available at: <https://doi.org/10.48550/arXiv.1708.04552> (accessed 09.08.2022).
6. Srivastava N., Hinton G. E., Krizhevsky A., Sutskever I., Salakhutdinov R. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014, no. 15, pp. 1929–1958. <https://doi.org/10.5555/2627435.2670313>
7. Chen H., Ihnatsyeva S., Bohush R., Ablameyko S. Choice of activation function in convolution neural network for person re-identification in video surveillance systems. *Programming and Computer Software*, 2022, vol. 48, no. 5, pp. 312–321. <http://doi.org/10.1134/S0361768822050036>
8. Zhong Z., Zheng L., Kang G., Li S., Yang Y. *Random Erasing Data Augmentation*, 2020. Available at: <https://doi.org/10.1609/AAAI.V34I07.7000> (accessed 09.08.2022).
9. Yun S., Han D., Oh S., Chun S., Choe J., Yoo Y. J. CutMix: Regularization strategy to train strong classifiers with localizable features. *2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 27 October – 2 November 2019*. Seoul, 2019, pp. 6022–6031. <https://doi.org/10.1109/ICCV.2019.00612>
10. Xie T., Cheng X., Wang X., Liu M., Deng J., ..., Liu M. Cut-thumbnail: A novel data augmentation for convolutional neural network. *Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021*. Virtual Event, China, 2021. <https://doi.org/10.1145/3474085.3475302>
11. Zhang H., Cissé M., Dauphin Y., Lopez-Paz D. *Mixup: Beyond Empirical Risk Minimization*, 2018. Available at: <https://doi.org/10.48550/arXiv.1710.09412> (accessed 09.08.2022).
12. Geirhos R., Rubisch P., Michaelis C., Bethge M., Wichmann F., Brendel W. *ImageNet-Trained CNNs are Biased Towards Texture; Increasing Shape Bias Improves Accuracy and Robustness*, 2019. Available at: <https://doi.org/10.48550/arXiv.1811.12231> (accessed 09.08.2022).
13. Gong Y., Zeng Z. *An Effective Data Augmentation for Person Re-identification*, 2021. Available at: <https://doi.org/10.48550/arXiv.2101.08533> (accessed 09.08.2022).
14. Huang H., Li D., Zhang Z., Chen X., Huang K. Adversarially occluded samples for person re-identification. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018*. Salt Lake City, 2018, pp. 5098–5107. <https://doi.org/10.1109/CVPR.2018.00535>
15. Ye M., Shen J., Lin G., Xiang T., Shao L., Hoi S. C. Deep learning for person re-identification: A survey and outlook. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021, vol. 44, iss. 6, pp. 2872–2893. <https://doi.org/10.1109/TPAMI.2021.3054775>
16. He K., Zhang X., Ren S., Sun J. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016*. Las Vegas, 2016, pp. 770–778. <https://doi.org/10.1109/cvpr.2016.90>
17. Huang G., Liu Z., Weinberger K. Q. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017*. Honolulu, 2017, pp. 2261–2269. <https://doi.org/10.1109/CVPR.2017.243>
18. Zheng L., Shen L., Tian L., Wang S., Wang J., Tian Q. Scalable person re-identification: A benchmark. *2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015*. Santiago, 2015, pp. 1116–1124. <https://doi.org/10.1109/ICCV.2015.133>
19. Ristani E., Solera F., Zou R. S., Cucchiara R., Tomasi C. *Performance Measures and a Data Set for Multi-target, Multi-camera Tracking*, 2016. Available at: https://doi.org/10.1007/978-3-319-48881-3_2 (accessed 09.08.2022).
20. Wei L., Zhang S., Gao W., Tian Q. Person transfer GAN to bridge domain gap for person re-identification. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018*. Salt Lake City, 2018, pp. 79–88. <https://doi.org/10.1109/CVPR.2018.00016>

Информация об авторах

Игнатъева Светлана Александровна, магистр технических наук, аспирант кафедры вычислительных систем и сетей, Полоцкий государственный университет имени Евфросинии Полоцкой.

E-mail: s.ignatieva@pdu.by

<https://orcid.org/0000-0002-9780-5731>

Богуш Рихард Петрович, доктор технических наук, доцент, заведующий кафедрой вычислительных систем и сетей, Полоцкий государственный университет имени Евфросинии Полоцкой.

E-mail: r.bogush@pdu.by

<https://orcid.org/0000-0002-6609-5810>

Information about the authors

Sviatlana A. Ihnatsyeva, M. Sc. (Eng.), Postgraduate Student of the Department of Computing Systems and Networks, Euphrosyne Polotskaya State University of Polotsk.

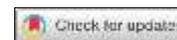
E-mail: s.ignatieva@pdu.by

<https://orcid.org/0000-0002-9780-5731>

Rykhard P. Bohush, D. Sc. (Eng.), Assoc. Prof., Head of the Department of Computing Systems and Networks, Euphrosyne Polotskaya State University of Polotsk.

E-mail: r.bogush@pdu.by

<https://orcid.org/0000-0002-6609-5810>



УДК 004.89
<https://doi.org/10.37661/1816-0301-2023-20-1-55-74>

Оригинальная статья
Original Paper

Классификация займов с использованием логистической регрессии

В. И. Бегунков[✉], М. Я. Ковалев¹

¹Объединенный институт проблем информатики
Национальной академии наук Беларуси,
ул. Сурганова, 6, Минск, 220012, Беларусь
[✉]E-mail: vbegunkov@gmail.com

Аннотация

Цели. Решение задачи классификации займов имеет большое значение для финансовых институтов, которые должны эффективно распределять денежные активы между субъектами в рамках предоставления финансовых услуг. Поэтому финансовым организациям необходим инструмент наиболее точного определения надежных заемщиков. Одним из инструментов принятия таких решений служит машинное обучение. Целью работы является анализ возможности эффективного применения логистической регрессии для решения задачи классификации займов.

Метод. На основе алгоритма логистической регрессии с использованием исторических данных по выданным займам рассчитываются следующие метрики: стоимостная функция, *Accuracy*, *Precision*, *Recall* и мера F_1 . Полиномиальная регрессия и метод главных компонент применяются для определения оптимального набора входных данных для исследуемого алгоритма логистической регрессии.

Результаты. Оценено влияние нормализации данных на конечный результат, дана оценка влияния сбалансированности целевых значений, рассчитано оптимальное граничное значение для алгоритма логистической регрессии, рассмотрено влияние увеличения входных показателей посредством заполнения отсутствующих значений и использования полиномов разной степени. Имеющийся набор входных показателей проанализирован на избыточность.

Заключение. Результаты исследований подтверждают, что применение алгоритма логистической регрессии для решения задач классификации займов является целесообразным. Данный алгоритм позволяет быстро получить работающий инструмент классификации займов.

Ключевые слова: классификация займов, скоринг, логистическая регрессия, машинное обучение, нормализация данных

Для цитирования. Бегунков, В. И. Классификация займов с использованием логистической регрессии / В. И. Бегунков, М. Я. Ковалев // Информатика. – 2023. – Т. 20, № 1. – С. 55–74.
<https://doi.org/10.37661/1816-0301-2023-20-1-55-74>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 11.01.2023
Подписана в печать | Accepted 10.02.2023
Опубликована | Published 29.03.2023

Loan classification using logistic regression

Uladzimir I. Behunkou[✉], Mikhail Y. Kovalyov¹

¹*The United Institute of Informatics Problems
of the National Academy of Sciences of Belarus,
st. Surganova, 6, Minsk, 220012, Belarus*

[✉]*E-mail: vbegunkov@gmail.com*

Abstract

Objectives. The studied problem of loan classification is particularly important for financial institutions, which must efficiently allocate monetary assets between entities as part of the provision of financial services. Therefore, it is more important than ever for financial institutions to be able to identify reliable borrowers as accurately as possible. At the same time, machine learning is one of the tools for making such decisions. The aim of this work is to analyze the possibility of efficient use of logistic regression for solving the task of loan classification.

Methods. Based on the logistic regression algorithm using historical data on loans issued, the following metrics are calculated: cost function, *Accuracy*, *Precision*, *Recall* и F_1 score. Polynomial regression and principal component analysis are used to determine the optimal set of input data for the being studied logistic regression algorithm.

Results. The impact of data normalization on the final result is estimated, the optimal regularization parameter for solving this problem is determined, the impact of the balance of target values is assessed, the optimal boundary value for the logistic regression algorithm is calculated, the influence of increasing input indicators by means of filling in missing values and using polynomials of different degrees is considered and the existing set of input indicators is analyzed for redundancy.

Conclusion. The research results confirm that the application of the logistic regression algorithm for solving loan classification problems is appropriate. The use of this algorithm allows to get quickly a working loan classification tool.

Keywords: loan classification, scoring, logistic regression, machine learning, data normalization

For citation. Behunkou U. I., Kovalyov M. Y. *Loan classification using logistic regression*. *Informatika [Informatics]*, 2023, vol. 20, no. 1, pp. 55–74 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-1-55-74>

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

Введение. Кредитование давно стало важным инструментом ускорения роста экономик как развитых, так и развивающихся стран. В связи с тем что в любой экономике имеются субъекты хозяйствования с избытком и недостатком денежных средств, кредитование позволяет распределять денежные активы с максимальной выгодой для всех участников, а значит, максимально использовать потенциал экономики отдельно взятой страны. Ключевая роль в данном процессе принадлежит финансовым институтам, которые должны эффективно распределять денежные активы между субъектами в рамках предоставления своих услуг. Поэтому финансовым организациям как никогда важно иметь возможность наиболее точно определять надежных заемщиков, необходимы современные инструменты, посредством которых можно удаленно и максимально быстро принимать решения по поступающим запросам на предоставление займов. Машинное обучение является одним из методов, на основе которого такие инструменты принятия решений могут быть разработаны [1]. В свою очередь, решаемую с помощью данных инструментов задачу можно представить как бинарную задачу классификации займа. Хэнд и Хэнли в своей работе [2] представили задачу классификации займа как бинарную следующим образом: разделили претендентов на кредиты на два класса в соответствии с вероятностью погашения займа на хороших (без дефолта) и плохих (с дефолтом). Развитие и использование различных алгоритмов для кредитного скоринга привели к необходимости проведения сравнительного анализа таких алгоритмов, который Баесенс с коллегами успешно реализовали в рамках исследования, опубликованного в 2003 г. [3]. Однако с течением времени стало ясно, что проведенные эксперименты начали терять свою актуальность по следующим причинам [4]:

1. В алгоритмах скоринга использованы в основном небольшие наборы данных и малое количество независимых показателей. В свою очередь, результатом развития информационных технологий является возможность получения доступа ко все большему объему данных с более широким набором исходных показателей.

2. Появились новые алгоритмы скоринга, сравнительные исследования по которым не проводились ранее.

3. Исследование было сфокусировано на анализе индивидуальных классификаторов (использующих какую-либо одну модель для получения результата), но с течением времени широкое распространение получили ансамблевые классификаторы (использующие множество моделей).

Поэтому Баесенс с группой исследователей провели новый сравнительный анализ алгоритмов скоринга, который был опубликован в 2015 г. [4]. В этом эксперименте рассмотрены три основные группы алгоритмов классификации: индивидуальные, однородные и разнородные ансамблевые классификаторы. По результатам анализа в финальную выборку попал и классификатор, основанный на логистической регрессии. Целью настоящей работы является исследование возможности эффективного применения логистической регрессии для решения задачи классификации займа.

Описание данных. Для решения задачи все используемые данные можно разделить на три группы: входные данные, настраиваемые параметры рассматриваемых методов и выходные данные.

Входные данные. Для настройки параметров и проведения экспериментов рассматриваемыми методами используются исторические данные по выданным на платформе для кредитования от человека человеку LendingClub займам¹, состоящие из 2 260 668 строк. Пусть t является отдельной позицией (строкой, которая соответствует определенному выданному займу). При этом для решения задачи были взяты все доступные позиции в количестве 2 260 668 (займы, выданные за период с апреля 2016 по сентябрь 2018 г.). Каждый займ характеризуется $n = 55$ независимыми показателями, которые удовлетворяют следующим свойствам:

- показатели были известны на момент принятия решения о выдаче займа;
- для них определены итоговые результаты по каждому займу;
- значения показателей определены (т. е. не отсутствуют) по всему перечню выданных займов.

Приведем названия и определения независимых показателей:

1. *loan_amnt* – запрошенная сумма займа.
2. *term* – количество платежей по займу в месяцах, что эквивалентно сроку займа.
3. *int_rate* – процентная ставка по займу.
4. *installment* – ежемесячный платеж по займу.
5. *emp_length* – продолжительность трудоустройства заемщика, выраженная в годах. Значения находятся в диапазоне от 0 до 10, где 0,5 означает занятость менее года, а 10 – занятость в течение 10 лет или выше. При этом допускается, что все значения n/a (not available) приравниваются к нулю. Таким образом, считается, что заемщики с этим значением не имеют трудового опыта.
6. *home_ownership* – информация о наличии в собственности недвижимости, которая была предоставлена заемщиком.
7. *annual_inc* – сумма годового дохода, представленная заемщиком при регистрации.
8. *verification_status* – статус, который определяет, что был проверен доход и источник дохода или проверка не проводилась.
9. *issue_d* – месяц, в котором займ был предоставлен.
10. *purpose* – назначение займа, которое указывается заемщиком.
11. *addr_state* – штат, в котором заемщик проживает.

¹Loan data // Lending Club. – Mode of access: <https://www.kaggle.com/datasets/wordsforthewise/lending-club>. – Date of access: 04.09.2019.

12. *dti* – соотношение суммы всех месячных платежей по займам (за исключением ипотеки) и месячного дохода заемщика.
13. *delinq_2yrs* – количество просрочек продолжительностью более 30 дней по платежам за последние два года согласно кредитной истории заемщика.
14. *earliest_cr_line* – месяц, в котором заемщику была открыта первая кредитная линия.
15. *inq_last_6mths* – количество запросов на займ за последние шесть месяцев (за исключением запросов на кредит на покупку автомобиля и на ипотеку).
16. *open_acc* – количество открытых кредитных линий у заемщика согласно отчету о кредитной истории.
17. *pub_rec* – количество отрицательных публичных записей в кредитной истории заемщика.
18. *revol_bal* – общий кредитный оборотный баланс.
19. *revol_util* – отношение суммы, используемой заемщиком, ко всей доступной сумме револьверного (возобновляемого) кредита.
20. *total_acc* – общее количество кредитных линий (включая закрытые) у заемщика согласно отчету о кредитной истории.
21. *initial_list_status* – изначальный статус листинга займа: частичный или полный (т. е. заемщик возьмет займ, если будет предоставлена полностью запрошенная сумма).
22. *application_type* – определяет, является ли обращение за займом индивидуальным или совместным (с другими заемщиками).
23. *acc_now_delinq* – количество просроченных счетов у заемщика.
24. *tot_coll_amt* – общая сумма задолженности, когда-либо подвергнутая взысканию.
25. *tot_cur_bal* – итоговый текущий баланс всех счетов.
26. *total_rev_hi_lim* – общий лимит по револьверному (возобновляемому) кредиту.
27. *acc_open_past_24mths* – количество открытых кредитных линий за последние 24 месяца.
28. *avg_cur_bal* – средний текущий баланс всех счетов.
29. *bc_open_to_buy* – суммарная доступная сумма кредита по возобновляемым банковским картам.
30. *bc_util* – соотношение суммарного текущего баланса к кредитному лимиту по всем банковским картам.
31. *chargeoff_within_12_mths* – количество списаний (безнадежной задолженности) в течение последних 12 месяцев.
32. *mo_sin_old_rev_tl_op* – количество месяцев с момента открытия старейшего револьверного счета.
33. *mo_sin_rcnt_rev_tl_op* – количество месяцев с момента открытия последнего револьверного счета.
34. *mo_sin_rcnt_tl* – количество месяцев с момента открытия последнего счета.
35. *mort_acc* – количество ипотечных счетов.
36. *mths_since_recent_bc* – количество месяцев с момента открытия последнего карточного счета.
37. *num_accts_ever_120_pd* – количество счетов, по которым были просрочки платежей продолжительностью 120 дней и более.
38. *num_actv_bc_tl* – количество текущих активных карточных счетов.
39. *num_actv_rev_tl* – количество текущих активных возобновляемых кредитных счетов.
40. *num_bc_sats* – количество удовлетворительных карточных счетов (по которым погашения производились вовремя).
41. *num_bc_tl* – общее количество карточных счетов.
42. *num_il_tl* – количество счетов с периодическими платежами.
43. *num_op_rev_tl* – количество открытых возобновляемых кредитных счетов.
44. *num_rev_accts* – количество возобновляемых кредитных счетов.
45. *num_rev_tl_bal_gt_0* – количество возобновляемых кредитных счетов с балансом больше нуля.
46. *num_sats* – количество удовлетворительных счетов.
47. *num_tl_op_past_12m* – количество счетов, открытых за последние 12 месяцев.

- 48. *pct_tl_nvr_dlq* – процент счетов, по которым не было просрочек платежей.
- 49. *pub_rec_bankruptcies* – количество публично зарегистрированных банкротств.
- 50. *tax_liens* – количество налоговых залогов.
- 51. *tot_hi_cred_lim* – итоговый кредитный лимит.
- 52. *total_bal_ex_mort* – итоговый кредитный баланс за исключением ипотеки.
- 53. *total_bc_limit* – итоговый кредитный лимит по картам.
- 54. *total_il_high_credit_limit* – итоговый лимит по счетам с периодическими платежами.
- 55. *disbursement_method* – метод предоставления заемщику кредита: наличными средствами или прямым платежом.

Предполагается, что значения данных показателей были известны до принятия решения о выдаче соответствующего займа. Обозначим значение показателя j в займе i из исходного набора данных через элементы $x_j^{(i)}$ матрицы X размером n на m , где $j = 1, \dots, n$, $i = 1, \dots, m$. Обозначим через x_j столбец матрицы X , а через $x^{(i)}$ – строку матрицы X , которая содержит значения независимых показателей в отдельной позиции (займе) i набора данных.

Также в качестве исходных данных используются целевые значения $y^{(i)}$ (итоговый результат по займам, где $i = 1, \dots, m$), которые определены в поле *loan_status* исходного набора данных. Показатель $y^{(i)}$ принимает два значения:

1. Возвратный займ (со значением *Fully Paid*). Такие займы были погашены. Соответствует значению $y^{(i)} = 1$.

2. Невозвратный займ (*Charged Off* или *Default*). Займы, по которым был объявлен дефолт или погашение займа просрочено более чем на 180 дней. Соответствует значению $y^{(i)} = 0$.

Займы со значениями *Current*, *In Grace period*, *Late (16–30 days)* и *Late (31–120 days)* исключаются из анализа, так как однозначно нельзя определить, будут они возвратными или невозвратными.

Параметры используемого алгоритма. В рассматриваемом алгоритме задействованы настраиваемые параметры:

θ_j – набор коэффициентов модели, где $j = 0, \dots, n$. Эти коэффициенты определяются в результате решения задачи оптимизации с помощью алгоритма путем минимизации стоимостной функции, как описано далее;

$a_\theta(x)$ – функция активации.

Выходные данные. Предположим, что выходными данными поставленной ранее бинарной задачи классификации (т. е. определения займа как возвратного или потенциально невозвратного) являются величины $\hat{y}^{(i)} \in \{0, 1\}$, где единица соответствует возвратному, а ноль – потенциально невозвратному займу i , $i = 1, \dots, m$.

Преобразование входных данных. Для решения бинарной задачи классификации с использованием рассматриваемого алгоритма необходимо, чтобы значения всех входных показателей были числами. В связи с тем что некоторые входные показатели имеют качественные и текстовые значения, следует провести их соответствующие преобразования:

Показатель *term* содержит текстовые значения. Его преобразование в числовой вид осуществляется в соответствии с табл. 1.

Таблица 1
Преобразование показателя *term*

Table 1
Conversion of term feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
36 months	1
60 months	2

Преобразование показателей *emp_length*, *verification_status*, *home_ownership* и *purpose* осуществляется в соответствии с табл. 2–5.

Таблица 2
Преобразование показателя *emp_length*

Table 2
Conversion of *emp_length* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
<i>n/a</i>	0
<i>< 1 year</i>	0,5
<i>1 year</i>	1
<i>2 years</i>	2
<i>3 years</i>	3
<i>4 years</i>	4
<i>5 years</i>	5
<i>6 years</i>	6
<i>7 years</i>	7
<i>8 years</i>	8
<i>9 years</i>	9
<i>10 + years</i>	10

Таблица 3
Преобразование показателя *purpose*

Table 3
Conversion of *purpose* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
<i>Car</i>	1
<i>credit_card</i>	2
<i>debt_consolidation</i>	3
<i>home_improvement</i>	4
<i>House</i>	5
<i>major_purchase</i>	6
<i>Medical</i>	7
<i>Moving</i>	8
<i>Other</i>	9
<i>renewable_energy</i>	10
<i>small_business</i>	11
<i>Vacation</i>	12
<i>Wedding</i>	13

Таблица 4
Преобразование показателя *home_ownership*

Table 4
Conversion of *home_ownership* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
<i>ANY</i>	1
<i>MORTGAGE</i>	2
<i>OWN</i>	3
<i>RENT</i>	4

Таблица 5
Преобразование показателя *verification_status*

Table 5
Conversion of *verification_status* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
<i>Source Verified</i>	1
<i>Verified</i>	2
<i>Not Verified</i>	3

Так как показатели *earliest_cr_line* и *issue_d* содержат месяц и год выдачи первого и текущего займа, которые по отдельности мало полезны для решения поставленной задачи, то в качестве новых значений показателя *earliest_cr_line* примем разницу между месяцем выдачи текущего займа (*issue_d*) и месяцем, в котором заемщику была открыта первая кредитная линия (*earliest_cr_line*). Соответственно, после преобразования показатель *issue_d* удаляется.

Преобразование показателей *addr_state*, *initial_list_status*, *application_type* и *disbursement_method* осуществляется в соответствии с табл. 6–9.

Таблица 6
Преобразование показателя *addr_state*

Table 6
Conversion of *addr_state* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>	Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
AK	1	MT	30
AL	2	NA	31
AR	3	NC	32
AS	4	ND	33
AZ	5	NE	34
CA	6	NH	35
CO	7	NJ	36
CT	8	NM	37
DC	9	NV	38
DE	10	NY	39
FL	11	OH	40
GA	12	OK	41
GU	13	OR	42
HI	14	PA	43
IA	15	PR	44
ID	16	RI	45
IL	17	SC	46
IN	18	SD	47
KS	19	TN	48
KY	20	TX	49
LA	21	UT	50
MA	22	VA	51
MD	23	VI	52
ME	24	VT	53
MI	25	WA	54
MN	26	WI	55
MO	27	WV	56
MP	28	WY	57
MS	29		

Таблица 7
Преобразование показателя *initial_list_status*

Table 7
Conversion of *initial_list_status* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
W	1
F	2

Таблица 8
Преобразование показателя *application_type*

Table 8
Conversion of *application_type* feature

Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
Joint App	1
Individual	2

Таблица 9
Преобразование показателя *disbursement_method*

Table 9
Conversion of *disbursement_method* feature

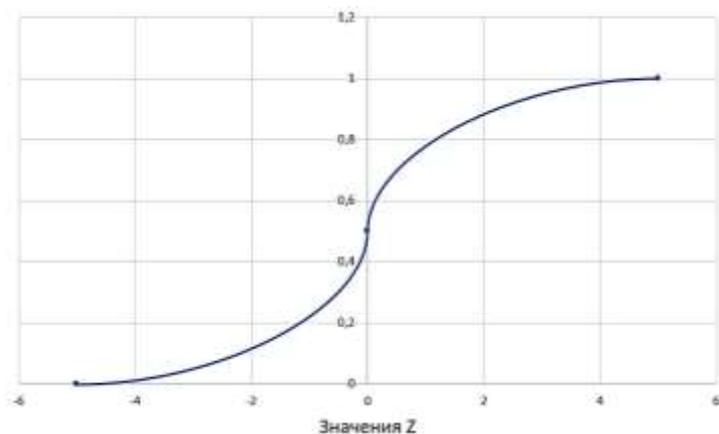
Исходное текстовое значение <i>Original text value</i>	Преобразованное альтернативное числовое значение <i>Converted alternative numeric value</i>
Cash	1
DirectPay	2

Результатом выполненных преобразований является конечный набор входных данных в числовом виде, который состоит из $m = 1\,221\,731$ позиций и $n = 54$ входных показателей.

Постановка задачи. Значения функции логистической регрессии (рисунок) всегда находятся в диапазоне $0 \leq a_{\theta}(x^{(i)}) \leq 1$. Для бинарной задачи классификации займа i под значением $a_{\theta}(x^{(i)})$ можно понимать вероятность возврата. При этом значение функции определяется выражением [5]

$$a_{\theta}(x^{(i)}) = \frac{1}{1+e^{-Z}}, \quad (1)$$

где $Z_{\theta}(x^{(i)}) = \theta_0 + \theta_1 \cdot x_1^{(i)} + \dots + \theta_n \cdot x_n^{(i)}$.



Логистическая (сигмовидная) функция

Logistic (sigmoid) function

Таким образом, функция $a_{\theta}(x^{(i)})$ определяет вероятность того, что значение выходного показателя \hat{y} будет равно единице, т. е. предоставляемый займ считается возвратным.

Для того чтобы иметь возможность классифицировать займ и определить значение выходного показателя $\hat{y}^{(i)} \in \{0, 1\}$ с учетом симметричности логистической функции относительно значения 0,5, примем следующие неравенства [6]:

$$\begin{aligned} a_{\theta}(x^{(i)}) \geq 0,5 &\rightarrow \hat{y}^{(i)} = 1; \\ a_{\theta}(x^{(i)}) < 0,5 &\rightarrow \hat{y}^{(i)} = 0. \end{aligned}$$

В качестве альтернативы алгоритм может определять вероятность возврата займа от 0 до 1 и предоставлять сотруднику возможность классифицировать займ как возвратный или невозвратный.

Для определения параметров θ_j используется стоимостная функция, которую требуется минимизировать. Данную функцию можно представить в виде равенства²

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}), \quad (2)$$

где $x^{(i)}$ и $y^{(i)}$ – заданные показатели, θ – вектор неизвестных переменных. Задачи минимизации невыпуклых функций весьма сложные. Стоимостная функция от переменных θ будет выпуклой, если функцию Cost представить следующим образом [6]:

$$\text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}) := \begin{cases} -\ln(a_{\theta}(x^{(i)})), & \text{если } y^{(i)} = 1; \\ -\ln(1 - a_{\theta}(x^{(i)})), & \text{если } y^{(i)} = 0. \end{cases} \quad (3)$$

Функция (3) обладает свойствами

$$\begin{cases} \text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}) \rightarrow 0, & \text{если } y^{(i)} = a_{\theta}(x^{(i)}); \\ \text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}) \rightarrow \infty, & \text{если } y^{(i)} = 0 \text{ и } a_{\theta}(x^{(i)}) \rightarrow 1; \\ \text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}) \rightarrow \infty, & \text{если } y^{(i)} = 1 \text{ и } a_{\theta}(x^{(i)}) \rightarrow 0. \end{cases} \quad (4)$$

Формулу (4) можно представить в более компактном виде³:

$$\text{Cost}(a_{\theta}(x^{(i)}), y^{(i)}) = -y^{(i)} \cdot \ln(a_{\theta}(x^{(i)})) - (1 - y^{(i)}) \cdot \ln(1 - a_{\theta}(x^{(i)})). \quad (5)$$

В результате полная стоимостная функция, которая учитывает весь набор данных m , рассчитывается по формуле [6]

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} \cdot \ln(a_{\theta}(x^{(i)})) + (1 - y^{(i)}) \cdot \ln(1 - a_{\theta}(x^{(i)})) \right]. \quad (6)$$

В задачах подобного типа при большом количестве входных показателей часто возникает проблема переобучения, при которой функция $a_{\theta}(x^{(i)})$ достаточно точно описывает имеющийся набор данных, но не очень полезна для прогнозирования, так как не справляется с задачей обобщения имеющегося набора данных.

При решении проблемы переобучения можно либо вручную выбрать некоторые из имеющихся входных показателей для расчета функции $a_{\theta}(x^{(i)})$, либо задействовать l_2 -регуляризацию [7], при которой включаются в расчет все имеющиеся входные показатели, специальным образом уменьшаются величины соответствующих им коэффициентов θ_j и изменяется функция $J(\theta)$. Стоит отметить, что коэффициент θ_0 не подлежит изменению, так как не связан с каким-либо входным показателем.

В настоящей работе при необходимости используется l_2 -регуляризация и к стоимостной функции⁴ добавляется $\frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$, где λ – параметр регуляризации. В результате итоговая стоимостная функция рассчитывается следующим образом:

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m \left[y^{(i)} \cdot \ln(a_{\theta}(x^{(i)})) + (1 - y^{(i)}) \cdot \ln(1 - a_{\theta}(x^{(i)})) \right] + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2. \quad (7)$$

²Cost function for logistic regression // Supervised Machine Learning: Regression and Classification. – Mode of access: <https://www.coursera.org/learn/machine-learning/lecture/0hpr8/cost-function-for-logistic-regression>. – Date of access: 16.11.2019.

³Simplified Cost Function for Logistic Regression // Supervised Machine Learning: Regression and Classification. – Mode of access: <https://www.coursera.org/learn/machine-learning/lecture/Zjj2j/simplified-cost-function-for-logistic-regression>. – Date of access: 16.11.2019.

⁴Regularized logistic regression // Supervised Machine Learning: Regression and Classification. – Mode of access: <https://www.coursera.org/learn/machine-learning/lecture/cAxpF/regularized-logistic-regression>. – Date of access: 16.11.2019.

Далее для минимизации стоимостной функции применяется метод градиентного спуска [8], посредством которого определяются оптимальные коэффициенты θ_j . Изначально для этого коэффициентам θ_j присваиваются нулевые значения. Затем путем последовательного изменения данных коэффициентов оптимизируется (в данном случае уменьшается) стоимостная функция $J(\theta)$. Таким образом, согласно определению метода градиентного спуска требуется последовательно повторять шаги по обновлению коэффициентов θ_j с помощью следующей формулы до тех пор, пока стоимостная функция не достигнет своего минимального значения⁵:

$$\theta_j := \theta_j - \alpha \cdot \frac{d}{d\theta_j} J(\theta). \quad (8)$$

При этом все коэффициенты θ_j должны обновляться одновременно, а параметр α определяет размер шага градиентного спуска.

Учитывая, что θ_0 не подлежит регуляризации, и определив частную производную стоимостной функции, пошаговое нахождение оптимальных коэффициентов θ_j с помощью метода градиентного спуска можно осуществить посредством преобразованных формул

$$\theta_0 := \theta_0 - \alpha \cdot \left[\frac{1}{m} \sum_{i=1}^m (a_{\theta}(x^{(i)}) - y^{(i)}) \cdot x_0^{(i)} \right], \quad (9)$$

$$\theta_j := \theta_j - \alpha \cdot \left[\frac{1}{m} \sum_{i=1}^m (a_{\theta}(x^{(i)}) - y^{(i)}) \cdot x_j^{(i)} + \frac{\lambda}{m} \cdot \theta_j \right], \quad j=1, \dots, n. \quad (10)$$

Для проведения экспериментов использовался объектно-ориентированный язык программирования *Python version 3.7.5*⁶. Оптимальное значение α вычислялось на основе экспериментов.

После нахождения коэффициентов θ_j необходимо рассчитать точность прогнозирования, т. е. с использованием оптимальных коэффициентов θ_j и входных показателей x_j определить целевые величины $\hat{y}^{(i)}$ по всему набору данных для обучения и сравнить с известными целевыми значениями $y^{(i)}$. Далее требуется провести оценку эффективности данной логистической регрессии. Для этого необходимо, используя $\hat{y}^{(i)}$ и $y^{(i)}$, рассчитать четыре основных параметра [9]: истинно положительный (*TP*), ложноположительный (*FP*), истинно отрицательный (*TN*) и ложноотрицательный (*FN*). На основе данных параметров строится матрица несоответствий (табл. 10).

Таблица 10
Матрица несоответствий

Table 10
Confusion matrix

Прогнозное значение $\hat{y}^{(i)}$ <i>Forecast value $\hat{y}^{(i)}$</i>	Тестовое значение $y^{(i)}$ <i>Test value $y^{(i)}$</i>	
	Класс 1 <i>Class 1</i>	Класс 0 <i>Class 0</i>
Класс 1	Истинно положительный	Ложноположительный
Класс 0	Ложноотрицательный	Истинно отрицательный

⁵Gradient Descent Implementation // Supervised Machine Learning: Regression and Classification. – Mode of access: <https://www.coursera.org/learn/machine-learning/lecture/Ha1RP/gradient-descent-implementation>. – Date of access: 16.11.2019.

⁶Python version 3.7.5 // Python Software Foundation. – Mode of access: <https://www.python.org/downloads/release/python-375/>. – Date of access: 21.10.2019.

С помощью матрицы рассчитывается базовый коэффициент эффективности *Accuracy* (A), который означает отношение количества правильно спрогнозированных классов ко всему набору спрогнозированных значений, т. е. $A = (TP + TN) / (TP + FP + TN + FN)$ [10]. Данный коэффициент отражает общую эффективность прогнозирования регрессии, но не учитывает возможный дисбаланс в распределении классов по всему набору данных, т. е. не учитывает, что в имеющемся наборе данных количество позиций класса 1 существенно больше, чем класса 0. Таким образом, для более детального измерения эффективности нейронной сети на основе матрицы несоответствий определяются дополнительные метрики:

Precision (P) – точность, или соотношение правильно спрогнозированных положительных классов к общему количеству положительно спрогнозированных классов, т. е. $P = TP / (TP + FP)$. В контексте рассматриваемой задачи означает, какой процент займов, спрогнозированных как возвратные, действительно таковыми являются.

Recall (R) – полнота, также известная как чувствительность, или соотношение правильно спрогнозированных положительных классов к общему количеству действительно положительно классов: $R = TP / (TP + FN)$. В данном случае показывает, какая доля возвратных займов предсказана верно.

Поскольку однозначный ответ о превосходстве какой-либо из метрик P и R отсутствует, для их совместного использования при оценке эффективности прогнозирования определяется мера F_1 как среднее гармоническое метрик P и R по формуле $F_1 = (2 \cdot P \cdot R) / (P + R)$. Однако при решении задачи многоклассовой классификации мера F_1 может быть рассчитана с помощью класса `sklearn.metrics.classification_report`⁷ как микроусредненная величина (без учета разделения данных на классы), макроусредненная величина (рассчитываются значения F_1 для каждого класса в отдельности, а далее находится среднее), а также как средневзвешенная величина (рассчитываются значения F_1 для каждого класса, а затем общее значение F_1 с учетом веса каждого класса в наборе данных). В связи с тем что значение коэффициента эффективности A равняется микроусредненному значению F_1 , микроусредненной величине P и микроусредненной величине R , а эти макроусредненные величины не учитывают вес каждого из классов в наборе исходных данных, для оценки эффективности будет использоваться коэффициент эффективности A и средневзвешенная мера F_1 на основе средневзвешенных значений P и R . Чем больше значения данных метрик, тем выше точность прогнозирования модели. Для получения метрик весь набор исходных данных разделяется на два множества: тренировочный и тестовый наборы данных. Первый используется для обучения, а второй – для проверки точности прогнозирования, так как наиболее корректной является проверка точности с использованием данных, которые алгоритм еще не обрабатывал. При этом тренировочный набор данных будет состоять из 70 %, а тестовый набор – из 30 % от всего множества займов.

Нормализация исходных данных. Многие входные показатели x_j имеют значения, которые сильно отличаются по своему диапазону. Например, показатель `'loan_amnt'` имеет значения в диапазоне (1000, 40 000), а значения показателя `'annual_inc'` находятся в интервале (16, 10 999 200). Такие различия в диапазонах величин могут привести к сложностям для некоторых алгоритмов машинного обучения [11] или более медленному выполнению метода градиентного спуска⁸. Это связано с тем, что метод может сходиться быстрее к точке экстремума при меньшем диапазоне значений входных показателей либо найти более оптимальное значение.

Для решения обозначенной проблемы необходимо привести значения входных показателей (m -мерных векторов-столбцов $x_j = (x_j^{(1)}, x_j^{(2)}, \dots, x_j^{(m)})^T$) к примерно одинаковому диапазону (в векторном виде удовлетворяющему соотношению $-1 \leq x_j \leq 1$ для большинства значений

⁷Sklearn.decomposition.classification_report // Sklearn Decomposition. – Mode of access: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html#sklearn.metrics.classification_report. – Date of access: 20.11.2022.

⁸Feature scaling part 1 // Supervised Machine Learning: Regression and Classification. – Mode of access: <https://www.coursera.org/learn/machine-learning/lecture/KMDV3/feature-scaling-part-1>. – Date of access: 25.11.2019.

векторов). Для этого используется инструмент средней нормализации, при котором значения каждого входного показателя изменяются в соответствии с векторной формулой [11]

$$x_j := \frac{x_j - \mu_j}{sd(x_j)}, \quad (11)$$

где μ_j – среднее значение всех компонент вектора x_j , а $sd(x_j)$ – стандартное отклонение компонент вектора x_j . При этом средняя нормализация применяется только к непрерывным⁹ показателям, а к номинальным¹⁰, ординальным¹¹ и категориальным¹² показателям (в данном случае *term*, *verification_status*, *home_ownership*, *purpose*, *addr_state*, *initial_list_status*, *application_type*, *disbursement_method*) не применяется. Все значения данных показателей делятся на максимальное число этих же показателей, что позволяет привести значения показателей к диапазону от нуля до единицы, т. е. делает их размерность сопоставимой с количественными показателями.

Для оценки влияния использования нормализации входных показателей на результаты машинного обучения в рамках логистической регрессии и для рассматриваемой задачи были проведены компьютерные исследования, состоящие из 10 000 итераций (количество итераций алгоритма до останова оптимизации). В рамках данных исследований для каждого из вариантов (с нормализацией и без нормализации) также было найдено значение α , которое обеспечило минимальное значение стоимостной функции на основе логарифмической шкалы, предполагающей, что следующее число получается умножением предыдущего на 10. При этом диапазон исследования для α составил от $1e-10$ до 10. Полученные результаты представлены в табл. 11 и 12.

Таблица 11
Результаты эксперимента при использовании нормализации

Table 11
Experiment results when using normalization

Исследуемый параметр <i>Parameter under study</i>	Значение без нормализации <i>Value without normalization</i>	Значение с нормализацией <i>Normalized value</i>
Оптимальный α	1e-10	1
Длительность обучения алгоритма (с)	0,153 65	0,180 76
Значение стоимостной функции	0,501 43	0,458 00
<i>Accuracy training</i> (%)	79,646 78	79,906 36
<i>Accuracy testing</i> (%)	79,727 44	80,000 82

Таблица 12
Ключевые метрики при использовании нормализации на тестовых данных

Table 12
Key metrics when using normalization on test data

Класс <i>Class</i>	<i>Precision</i>	<i>Recall</i>	Мера F_1 <i>Measure F_1</i>
Невозвратные займы	0,538 95	0,092 92	0,158 51
Возвратные займы	0,809 46	0,979 79	0,886 52
Средневзвешенное	0,754 63	0,800 01	0,738 94

⁹Continuous or discrete variable // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Continuous_or_discrete_variable. – Date of access: 19.11.2022.

¹⁰Level of measurement // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Level_of_measurement#Nominal_level. – Date of access: 19.11.2022.

¹¹Ordinal data // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Ordinal_data. – Date of access: 19.11.2022.

¹²Categorical variable // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Categorical_variable. – Date of access: 19.11.2022.

По результатам использования указанных ранее входных показателей с исходными данными для решения задачи классификации займа можно сделать следующие выводы:

1. Применение регуляризации на этапе нормализации нецелесообразно, так как значения коэффициента эффективности A , полученные на тестовых и тренировочных данных, отличаются незначительно, что свидетельствует об отсутствии проблемы переобучения.

2. Использование нормализации уменьшает значение стоимостной функции на 8,66 % и улучшает точность прогнозирования (значение A) на 0,27 %.

3. Значение $P = 0,80946$ для возвратных займов означает, что 80,946 % займов, определенных с помощью модели как возвратные, действительно таковыми являются. При этом значение $R = 0,97979$ свидетельствует о том, что при рассмотрении всего множества действительно возвратных займов модель правильно классифицировала 97,979 % из них.

Очевидно, что нормализация ведет к улучшению точности прогнозирования. Поэтому в дальнейших компьютерных экспериментах будет использован именно такой вариант.

Не менее важным является определение коэффициентов θ_j ($j = 1, \dots, n$) для каждого из входных векторных показателей x_j . По результатам проведенного исследования рассчитаны значения θ_j (табл. 13).

Таблица 13
Значения θ_j

Table 13
 θ_j values

Номер показателя <i>Indicator number</i>	θ_j	ID показателя <i>Indicator ID</i>	Номер показателя <i>Indicator number</i>	θ_j	ID показателя <i>Indicator ID</i>
1	-0,075 79	<i>loan_amnt</i>	28	0,079 38	<i>avg_cur_bal</i>
2	-1,232 21	<i>term</i>	29	-0,033 85	<i>bc_open_to_buy</i>
3	-0,344 14	<i>int_rate</i>	30	-0,041 07	<i>bc_util</i>
4	-0,073 28	<i>installment</i>	31	0,000 31	<i>chargeoff_within_12_mths</i>
5	0,074 93	<i>emp_length</i>	32	0,134 07	<i>mo_sin_old_rev_tl_op</i>
6	-0,680 81	<i>home_ownership</i>	33	-0,002 03	<i>mo_sin_rcnt_rev_tl_op</i>
7	0,063 93	<i>annual_inc</i>	34	0,021 66	<i>mo_sin_rcnt_tl</i>
8	0,217 13	<i>verification_status</i>	35	0,055 89	<i>mort_acc</i>
9	0,586 29	<i>disbursement_method</i>	36	0,040 98	<i>mths_since_recent_bc</i>
10	-0,380 98	<i>purpose</i>	37	-0,003 42	<i>num_accts_ever_120_pd</i>
11	-0,001 91	<i>addr_state</i>	38	0,009 26	<i>num_actv_bc_tl</i>
12	-0,168 92	<i>dti</i>	39	-0,085 44	<i>num_actv_rev_tl</i>
13	-0,074 16	<i>delinq_2yrs</i>	40	-0,064 60	<i>num_bc_sats</i>
14	-0,142 87	<i>earliest_cr_line</i>	41	0,025 55	<i>num_bc_tl</i>
15	-0,048 90	<i>inq_last_6mths</i>	42	-0,068 90	<i>num_il_tl</i>
16	-0,007 80	<i>open_acc</i>	43	0,042 27	<i>num_op_rev_tl</i>
17	-0,027 50	<i>pub_rec</i>	44	-0,077 29	<i>num_rev_accts</i>
18	-0,000 35	<i>revol_bal</i>	45	-0,051 10	<i>num_rev_tl_bal_gt_0</i>
19	-0,023 65	<i>revol_util</i>	46	0,039 87	<i>num_sats</i>
20	0,202 42	<i>total_acc</i>	47	-0,012 82	<i>num_tl_op_past_12m</i>
21	0,201 88	<i>initial_list_status</i>	48	0,008 09	<i>pct_tl_nvr_dlq</i>
22	-0,198 58	<i>application_type</i>	49	-0,001 17	<i>pub_rec_bankruptcies</i>
23	-0,009 82	<i>acc_now_delinq</i>	50	0,008 20	<i>tax_liens</i>
24	0,002 06	<i>tot_coll_amt</i>	51	0,009 95	<i>tot_hi_cred_lim</i>
25	0,024 63	<i>tot_cur_bal</i>	52	-0,164 83	<i>total_bal_ex_mort</i>
26	0,113 75	<i>total_rev_hi_lim</i>	53	0,132 93	<i>total_bc_limit</i>
27	-0,162 46	<i>acc_open_past_24mths</i>	54	0,172 28	<i>total_il_high_credit_limit</i>

Как следует из результатов эксперимента, $\theta_2 = -1,232\ 21$ является коэффициентом с наибольшим абсолютным значением. Таким образом, показатель x_2 (срок займа) оказывает наибольшее отрицательное влияние на результат функции $a_\theta(x^{(i)})$. Соответственно, чем больше срок кредита, тем менее вероятно, что займ будет классифицирован как возвратный.

Влияние сбалансированности исторических целевых значений на классификацию займов. Как было указано ранее, набор входных данных состоит из 1 221 731 позиции (заявки на займ). Однако возвратным займам соответствует 973 421 (~ 79,7 %) позиция, а невозвратным – 248 310 (~ 20,3 %) позиций. Следовательно, набор входных данных не сбалансирован по целевым значениям и смещен в сторону позиций с возвратными займами.

Так как некоторые подходы к машинному обучению показывают лучшие результаты при обучении сбалансированными данными [11], необходимо провести анализ влияния сбалансированности входных данных на результаты прогнозирования в рамках рассматриваемой задачи. Для этого из входного набора данных создается подмассив [8], который состоит из всех 248 310 позиций входных данных, соответствующих невозвратным займам, и только из 248 310 позиций, соответствующих возвратным займам. В итоге набор входных данных в подмассиве будет сбалансирован, но общее количество позиций уменьшится до 496 620.

Результаты компьютерного эксперимента, состоящего из 10 000 итераций при $\alpha = 1$, представлены в табл. 14 и 15.

Таблица 14

Результаты эксперимента при сбалансированности исторических целевых значений

Table 14

Experiment results when the historical target values are balanced

Исследуемый параметр <i>Parameter under study</i>	Значение <i>Value</i>
Длительность обучения алгоритма (с)	0,072 73
Значение стоимостной функции	0,590 68
<i>Accuracy training (%)</i>	68,677 98
<i>Accuracy testing (%)</i>	68,882 31

Таблица 15

Ключевые метрики при использовании сбалансированных данных

Table 15

Key metrics when using balanced data

Класс <i>Class</i>	<i>Precision</i>	<i>Recall</i>	Мера F_1 <i>Measure F_1</i>
Невозвратные займы	0,691 74	0,679 36	0,685 49
Возвратные займы	0,686 02	0,698 26	0,692 09
Средневзвешенное	0,688 87	0,688 82	0,688 79

По итогам проведенного исследования можно сделать вывод, что абсолютная сбалансированность не привела к улучшению значения стоимостной функции, величин A и меры F_1 модели при использовании предложенного алгоритма машинного обучения в задаче классификации займа. Уменьшение точности прогнозирования и увеличение значения стоимостной функции объясняются существенным уменьшением набора входных позиций с 1 221 731 до 496 620 в связи с намерением сбалансировать набор входных данных. Вместе с тем следует отметить улучшение метрик P , R и F_1 для невозвратных займов. Поэтому применять сбалансированные входные данные можно в случае, когда точность прогнозирования невозвратных займов более важна, чем возвратных. Учитывая, что величины A и F_1 модели оказались хуже значений, полученных при отсутствии сбалансированности, в дальнейшем будет использован весь набор входных данных, состоящий из 1 221 731 позиции.

Классификация займов при увеличении входных показателей. При формировании входных данных задействовались показатели с определенными значениями по всему перечню вы-

данных займов. Однако в первоначальном наборе имелись показатели, которые были известны на момент выдачи займа, но имели до 30 % отсутствующих значений. Задействование этих входных показателей может привести к увеличению точности прогнозирования. Таким образом, необходимо оценить влияние исключенных из-за неполноты данных, но известных на момент выдачи входных показателей. Для этого требуется преобразовать входные показатели путем заполнения отсутствующих значений следующим образом: так как все показатели содержат количественные значения, то пустые позиции будут заменены поочередно на модальную величину соответствующего показателя, а также на среднее и медианное значения. Далее входные показатели нормализуются аналогично другим непрерывным показателям, как было описано ранее. Впоследствии требуется провести исследования и представить сравнительный анализ данных трех вариантов.

Приведем перечень дополнительных входных показателей:

mths_since_last_delinq – количество месяцев с момента последней просрочки;
mths_since_last_record – количество месяцев с момента последней публичной записи;
open_acc_6m – количество открытых кредитных счетов за последние шесть месяцев;
open_act_il – количество текущих активных счетов с рассрочкой платежа;
open_il_12m – количество счетов с рассрочкой платежа, открытых за последние 12 месяцев;
open_il_24m – количество счетов с рассрочкой платежа, открытых за последние 24 месяца;
mths_since_rcnt_il – количество месяцев с момента открытия последнего счета с рассрочкой платежа;
total_bal_il – текущий баланс по всем счетам с рассрочкой платежа;
il_util – соотношение суммарного текущего баланса к кредитному лимиту по всем счетам с рассрочкой;
open_rv_12m – количество револьверных счетов, открытых за последние 12 месяцев;
open_rv_24m – количество револьверных счетов, открытых за последние 24 месяца;
max_bal_bc – максимальный текущий баланс задолженности по всем револьверным счетам;
all_util – соотношение баланса к кредитному лимиту по всем счетам;
inq_fi – количество персональных финансовых запросов;
total_cu_tl – количество финансовых счетов;
inq_last_12m – количество запросов на кредит за последние 12 месяцев;
mo_sin_old_il_acct – количество месяцев со времени открытия самого старого счета с рассрочкой платежа;
mths_since_recent_inq – количество месяцев с момента последнего запроса;
percent_bc_gt_75 – процент всех счетов по банковским картам, которые превышают 75 % лимита.

При проведении исследования были использованы модальные, средние и медианные значения соответствующих дополнительных показателей для устранения пустых позиций и расчета точности прогнозирования для каждого случая при решении задачи классификации займа. Результаты исследования представлены в табл. 16 и 17.

Таблица 16
Результаты исследования при увеличении количества входных показателей

Table 16
Research results with an increase in the number of input features

Исследуемый параметр <i>Parameter under study</i>	Модальные значения <i>Modal values</i>	Средние значения <i>Averages</i>	Медианные значения <i>Median values</i>
Длительность обучения алгоритма (с)	0,223 65	0,208 26	0,219 02
Значение стоимостной функции	0,456 66	0,457 07	0,457 05
<i>Accuracy training</i> (%)	79,906 36	79,930 57	79,914 66
<i>Accuracy testing</i> (%)	79,985 54	80,027 28	80,005 73

Таблица 17
Ключевые метрики при заполнении средними значениями

Table 17
Key metrics when filled with averages

Класс Class	Precision	Recall	Мера F_1 Measure F_1
Невозвратные займы	0,541 73	0,095 67	0,162 62
Возвратные займы	0,809 87	0,979 42	0,886 61
Средневзвешенное	0,755 52	0,800 27	0,739 85

Как следует из результатов эксперимента, включение дополнительных показателей в набор выходных исходных данных и их преобразование с помощью средних и медианных значений привели к улучшению точности прогнозирования по сравнению с точностью $A = 80,000\ 82\ %$, определенной ранее. При этом преобразование с помощью средних значений привело к лучшим результатам, чем преобразование с помощью модальных и медианных значений. Относительно стоимостной функции все три варианта показали лучший результат, чем полученный ранее. Поэтому в дальнейших исследованиях будут использоваться показатели, у которых отсутствующие значения заполнены средними величинами, а количество входных показателей n станет равным 73.

Классификация займов при различных граничных значениях. Функция логистической регрессии является симметричной относительно значения 0,5 (см. рисунок). Поэтому это значение было выбрано как граничное для определения того, будет займ возвратным или же невозвратным. Однако данная постановка задачи не может исключать возможности существования более подходящего пограничного значения при решении текущей задачи. В связи с этим требуется провести исследование влияния различных граничных значений на результаты решения задачи классификации займа. Следовательно, необходимо проанализировать влияние разных граничных значений от 0,01 до 1 с шагом 0,01 на точность прогнозирования при классификации займа на основе 10 000 итераций обучения алгоритма логистической регрессии. По результатам обучения находится оптимальное пограничное значение, которому соответствует наибольшая точность прогнозирования, выраженная значением A на тестовых данных.

Результаты анализа показали, что оптимальным граничным значением являлась величина 0,49. При этом значении средняя точность прогнозирования на тренировочных данных составила 79,93267 %, а на тестовых – 80,028102 %. Граничные значения не оказывают влияние на величину стоимостной функции.

Применительно к текущей задаче полученная точность в некоторой степени больше точности 80,027 28 %, рассчитанной при использовании пограничного значения 0,5. Поэтому при дальнейшем анализе предложенного алгоритма машинного обучения будет применяться граничное значение 0,49.

Таким образом, выявлено, что исследование различных граничных значений при решении задачи классификации займа является целесообразным. Вместе с тем из табл. 18 следует, что значения метрик P , R и F_1 также изменились. В частности, значение F_1 всей модели ухудшилось до 0,73767 в сравнении с 0,73985. Следовательно, оптимальное граничное значение модели может варьироваться в зависимости от показателя, выбранного для оптимизации.

Таблица 18
Ключевые метрики при оптимальном граничном значении

Table 18
Key metrics at the optimal boundary value

Класс Class	Precision	Recall	Мера F_1 Measure F_1
Невозвратные займы	0,542 46	0,087 62	0,151 01
Возвратные займы	0,808 83	0,981 48	0,886 83
Средневзвешенное	0,755 56	0,800 28	0,737 67

Классификация займов при использовании полиномиальных показателей. Согласно результатам исследований, проведенных ранее, увеличение количества показателей входных данных может привести к улучшению ключевых показателей рассматриваемого алгоритма обучения. Альтернативным вариантом увеличения числа входных показателей является создание дополнительных входных показателей на основе каждого из уже имеющихся показателей с использованием полиномиальной регрессии. В упрощенном варианте, предполагающем наличие одного входного показателя, полиномиальная регрессия выражается следующим образом [5]:

$$Z(x^{(i)}) = \theta_0 + \theta_1 \cdot x_1^{(i)1} + \theta_2 \cdot x_1^{(i)2} + \dots + \theta_p \cdot x_1^{(i)p}, \quad (12)$$

где p – степень полинома. В результате каждый входной показатель может быть возведен поочередно до степени p и добавлен к исходному набору входных данных. При этом регрессия приобретает более сложный нелинейный характер.

Для решения задачи классификации займа в качестве эксперимента имеющийся набор входных показателей x_j требуется расширить с помощью полинома от второй до пятой степени включительно и сравнить результаты. Учитывая, что состав входных данных изменился, величина $\alpha = 1$ может перестать быть оптимальной. Поэтому для каждого полинома необходимо найти оптимальное значение α (соответствующее минимальному значению стоимостной функции) в диапазоне от $1e-10$ до 10 на основе логарифмической шкалы аналогично подходу, использованному ранее.

Результаты компьютерного эксперимента, состоящего из $10\,000$ итераций для каждой степени полинома, представлены в табл. 19.

Таблица 19
Результаты исследования при использовании полиномиальных показателей

Table 19
Research results when using polynomial features

Исследуемый параметр <i>Parameter under study</i>	Полином второй степени <i>Second degree polynomial</i>	Полином третьей степени <i>Third degree polynomial</i>	Полином четвертой степени <i>Fourth degree polynomial</i>	Полином пятой степени <i>Fifth degree polynomial</i>
α	1e-2	1e-4	1e-6	1e-10
Длительность обучения алгоритма (с)	0,346 42	0,433 01	0,532 92	0,663 71
Значение стоимостной функции	0,455 68	0,493 49	0,646 89	0,704 09
<i>Accuracy training (%)</i>	79,899 58	79,640 11	79,384 04	79,248 74
<i>Accuracy testing (%)</i>	79,959 62	79,708 88	79,449 42	79,312 45

Таблица 20
Ключевые метрики при использовании полинома второй степени

Table 20
Key metrics when using of the second degree polynomial

Класс <i>Class</i>	<i>Precision</i>	<i>Recall</i>	Мера F_1 <i>Measure F_1</i>
Невозвратные займы	0,548 34	0,064 66	0,115 68
Возвратные займы	0,805 75	0,986 46	0,886 99
Средневзвешенное	0,753 57	0,799 60	0,730 64

Согласно результатам анализа увеличение количества исходных входных показателей с использованием полинома второй степени привело к улучшению (уменьшению) значения стоимостной функции. Вместе с тем из табл. 20 следует, что длительность алгоритма обучения

существенно увеличилась, а значения коэффициента эффективности A и средневзвешенной меры F_1 оказались ниже полученных ранее. Поэтому дальнейшее увеличение количества входных показателей на основе полиномиальной регрессии нецелесообразно.

Использование метода главных компонент для задачи классификации займа. Как следует из полученных ранее результатов, увеличение количества входных показателей с 54 до 73 привело к улучшению метрик исследуемой модели. Учитывая, что дальнейшее увеличение количества показателей с применением полиномов не привело к улучшению точности прогнозирования, целесообразно проанализировать имеющийся набор входных показателей на избыточность. Основными причинами устранения избыточности являются [8]: упрощение использования набора данных, уменьшение вычислительных затрат и шума в данных. Целью исследования является уменьшение размерности входных данных без ухудшения значений основных результирующих показателей рассматриваемой модели. Для такого исследования будет использован метод главных компонент (*principal component analysis*)¹³ [7, 8], суть которого заключается в уменьшении линейной размерности с использованием разложения по сингулярным значениям¹⁴. Для последовательного расчета главных компонент (начиная с первого и до заданного количества) будет использован класс *sklearn.decomposition.PCA*¹⁵. Главные компоненты берутся в диапазоне от 1 до 73 включительно с целью выявления их оптимального количества, которое обеспечит максимальное значение коэффициента эффективности A на тестовых данных.

Результаты проведенных расчетов показали, что оптимальное количество главных компонент равнялось 72, значения остальных метрик приведены в табл. 21 и 22.

Таблица 21

Результаты исследования при использовании метода главных компонент

Table 21

Results of the study using the method of principal components

Исследуемый параметр <i>Parameter under study</i>	Значение <i>Value</i>
Длительность обучения алгоритма (с)	0,206 32
Значение стоимостной функции	0,457 06
<i>Accuracy training</i> (%)	79,934 08
<i>Accuracy testing</i> (%)	80,040 65

Таблица 22

Ключевые метрики при использовании метода главных компонент

Table 22

Key metrics when using principal component analysis

Класс <i>Class</i>	<i>Precision</i>	<i>Recall</i>	Мера F_1 <i>Measure F_1</i>
Невозвратные займы	0,547 79	0,088 25	0,152 01
Возвратные займы	0,808 94	0,981 48	0,886 89
Средневзвешенное	0,756 00	0,800 41	0,737 92

Из полученных результатов следует, что применение метода главных компонент привело к получению наилучшего значения коэффициента эффективности $A = 80,0401\%$. Отсюда можно сделать вывод, что использование данного метода является целесообразным при решении

¹³Principal component analysis // Wikipedia. – Mode of access: https://en.wikipedia.org/wiki/Principal_component_analysis. – Date of access: 22.12.2022.

¹⁴Principal component analysis (PCA) // Sklearn Decomposition. – Mode of access: <https://scikit-learn.org/stable/modules/decomposition.html#decompositions>. – Date of access: 22.12.2022.

¹⁵Sklearn.decomposition.PCA // Sklearn Decomposition. – Mode of access: <https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html>. – Date of access: 22.12.2022.

задачи классификации займа. При этом значение $R = 98,148\%$ для возвратных займов означает, что модель правильно классифицировала 98,148 % из них.

Так как количество входных показателей равно 73, а оптимальное количество главных компонент 72, то можно сделать вывод, что исходное пространство показателей в основном не является избыточным. Вместе с тем обнаружено, что применение метода главных компонент с количеством компонент, равным 55, покрывает минимум 99 % дисперсии входных данных. Этот факт может быть полезен при необходимости уменьшения вычислительных затрат.

Заключение. В работе представлены принципы формирования и преобразования данных для задачи классификации займов и рассмотрено применение алгоритма логистической регрессии для ее решения. Выявлено, что использование нормализации улучшает точность прогнозирования при оптимальной величине $\lambda = 1$, а абсолютная сбалансированность целевых значений не приводит к улучшению конечных результатов. Установлено, что оптимальным граничным значением для алгоритма логистической регрессии является 0,49 вместо используемого по умолчанию 0,5. Определено, что увеличение показателей в наборе входных данных и их преобразование с помощью средних и медианных значений способствует улучшению точности прогнозирования. Вместе с тем увеличение количества входных показателей с использованием полиномов не привело к однозначному улучшению показателей модели, но существенно увеличило длительность обучения задействованного алгоритма. Применение метода главных компонент позволило получить максимальное значение коэффициента эффективности A и является целесообразным при решении задачи классификации займов.

Вклад авторов. В. И. Бегунков разработал принципы обработки данных и программной модели решения задачи классификации займов, провел эксперименты с интерпретированием результатов. М. Я. Ковалев сформулировал задачу исследования и выполнил научное редактирование статьи.

References

1. Gerhard F., Harlalka A., Suvanam R. The coming opportunity in consumer lending. *McKinsey Quarterly*, 2021. Available at: <https://www.mckinsey.com/business-functions/risk-and-resilience/our-insights/the-coming-opportunity-in-consumer-lending> (accessed 01.05.2021).
2. Hand D. J., Henley W. E. Statistical classification methods in consumer credit scoring: a review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 1997, vol. 160, no. 3, pp. 523–541.
3. Baesens B., Van Gestel T., Viaene S., Stepanova S., Suykens J., Vanthienen J. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, 2003, vol. 54, no. 6, pp. 627–635.
4. Lessmann S., Baesens B., Seow H.-V., Thomas L. C. Benchmarking state-of-the-art classification algorithms for credit scoring: an update of research. *European Journal of Operational Research*, 2015, vol. 247, no. 1, pp. 124–136.
5. Shalev-Shwartz S., Ben-David S. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014, pp. 125, 126–127.
6. Geron A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd edition. O'Reilly Media, 2019, pp. 144–149.
7. Murphy K. P. *Machine Learning: A Probabilistic Perspective (Adaptive Computation and Machine Learning Series)*. The MIT Press, 2012, pp. 225–227, 387–407.
8. Harrington P. *Machine Learning in Action*, 1st edition. Manning Publication Co, 2012, pp. 86–91, 148, 269–279.
9. Fawcett T. An introduction to ROC analysis. *Pattern Recognition Letters*, 2006, vol. 27, no. 8, pp. 861–874.
10. Metz C. E. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 1978, vol. 8, no. 4, pp. 283–298.
11. Kelleher J. D., Namee B. M., D'Arcy A. *Fundamentals of Machine Learning for Predictive Data Analytics*, 1st edition. The MIT Press, 2015, pp. 142–143, 539.

Информация об авторах

Бегунков Владимир Иванович, магистр технических наук.

E-mail: vbegunkov@gmail.com

Ковалев Михаил Яковлевич, член-корреспондент НАН Беларуси, доктор физико-математических наук, профессор, Объединенный институт проблем информатики Национальной академии наук Беларуси.

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

<https://orcid.org/0000-0003-0832-0829>

Information about the authors

Uladzimir I. Behunkou, M. Sc. (Eng.).

E-mail: vbegunkov@gmail.com

Mikhail Y. Kovalyov, Corresponding Member of the National Academy of Sciences of Belarus, D. Sc. (Eng.), Prof., The United Institute of Informatics Problems of the National Academy of Sciences of Belarus.

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

<https://orcid.org/0000-0003-0832-0829>

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ

LOGICAL DESIGN



УДК 519.711
<https://doi.org/10.37661/1816-0301-2023-20-1-75-90>

Оригинальная статья
Original Paper

Совместное энергосберегающее кодирование состояний последовательных автоматов сети, реализующей параллельный автомат

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

Цели. Рассматривается задача энергосберегающего кодирования частичных состояний параллельного автомата. Целью работы является исследование возможности использования приема декомпозиции при кодировании частичных состояний для снижения размерности задачи.

Методы. Заданный параллельный автомат разлагается в сеть последовательных автоматов, состояния которых кодируются затем трюичными векторами. Метод кодирования использует поиск максимального разреза во взвешенном графе, представляющем пары состояний, связанных переходами. Весами ребер графа являются величины, связанные с вероятностями переходов.

Результаты. Описан способ построения сети из последовательных автоматов, реализующей заданный параллельный автомат. Вероятности переходов между состояниями вычисляются путем решения системы линейных уравнений согласно методу Чэпмена – Колмогорова. Значения внутренних переменных, кодирующих состояния каждого компонентного последовательного автомата, находятся по двухблочным разбиениям множества его состояний, которые определяются разрезами соответствующего графа переходов.

Заключение. Использование декомпозиции параллельного автомата позволяет снизить размерность трудоемкой задачи кодирования состояний. Предлагаемый метод предназначен для применения в системах автоматизированного проектирования дискретных устройств.

Ключевые слова: параллельный автомат, частичное состояние, декомпозиция автоматов, кодирование состояний автомата, граф переходов, вероятность перехода между состояниями

Для цитирования. Поттосин, Ю. В. Совместное энергосберегающее кодирование состояний последовательных автоматов сети, реализующей параллельный автомат / Ю. В. Поттосин // Информатика. – 2023. – Т. 20, № 1. – С. 75–90. <https://doi.org/10.37661/1816-0301-2023-20-1-75-90>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 18.12.2022
Подписана в печать | Accepted 20.01.2023
Опубликована | Published 29.03.2023

Joint low power state assignment of sequential automata of a net implementing a parallel automaton

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

Objectives. The problem of low power state assignment of partial states of a parallel automaton is considered. The objective of the paper is to investigate the possibilities of using the decomposition in state assignment of partial states in order to decrease the task dimension.

Methods. Parallel automaton is decomposed into a net of sequential automata whose states are assigned then with ternary vectors. The method for assignment uses searching for a maximal cut in a weighted graph that represents pairs of states connected by transitions. The edge weights of the graph are the values related to the probabilities of transitions.

Results. A method to construct a net of sequential automata that realizes the given parallel automaton is described. The probabilities of transitions between sets are calculated by means of solving a system of linear equations according to the Chapman – Kolmogorov method. The values of inner variables assigned to the states of every component sequential automaton are obtained from two-block partitions of its set of states that are determined by the cuts of corresponding transition graph.

Conclusion. Applying parallel automaton decomposition allows decreasing the dimension of the laborious problem of state assignment. The proposed method is intended for application in computer aided systems for design of discrete devices.

Keywords: parallel automaton, partial state, decomposition of automata, state assignment of automata, transition graph, probability of transition between states

For citation. Pottosin Yu. V. *Joint low power state assignment of sequential automata of a net implementing a parallel automaton*. Informatika [Informatics], 2023, vol. 20, no. 1, pp. 75–90 (In Russ.).
<https://doi.org/10.37661/1816-0301-2023-20-1-75-90>

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

Введение. Одним из важных критериев оптимизации при проектировании дискретных устройств является величина потребляемой энергии. Это обусловлено, с одной стороны, стремлением увеличить время действия источника энергии в портативных приборах, а с другой – стремлением снизить остроту проблемы отвода тепла при проектировании сверхбольших интегральных схем.

Как отмечено в работах [1, 2], потребляемая мощность схемы, построенной на основе КМОП-технологии, пропорциональна интенсивности переключений логических элементов и элементов памяти. Это дает возможность частично решать данную проблему на уровне логического проектирования. В частности, снижения энергопотребления можно добиваться на этапе кодирования состояний автомата, т. е. когда абстрактным символам состояний приписываются булевы векторы, что необходимо для получения системы булевых функций, представляющей структурную модель проектируемого устройства. Для последовательных автоматов при синхронной и асинхронной реализации эта задача решалась в работах [3–8]. Метод энергосберегающего кодирования частичных состояний параллельного автомата описан в статье [9], где используется подход [10], сводящий данную задачу к поиску покрытия графа его полными двудольными подграфами.

Параллельный автомат является функциональной моделью дискретного устройства управления, с достаточной степенью удобства представляющей параллелизм управляемых взаимодействующих процессов [11]. Эта модель близка к широко известной сети Петри [12]. В отличие от последовательного автомата, который в отдельный момент времени находится только

в одном состоянии, параллельный автомат может одновременно находиться в нескольких частичных состояниях. Любая совокупность частичных состояний, в которых параллельный автомат может находиться одновременно, не находясь ни в каком другом состоянии, называется глобальным состоянием. Задачу кодирования состояний параллельного автомата можно решать путем нахождения всех его глобальных состояний и последующего их кодирования каким-либо из методов, разработанных применительно к последовательным автоматам. Однако при сравнительно небольшом количестве частичных состояний параллельного автомата число его глобальных состояний, как показано в работе [13], может быть настолько велико, что задача их кодирования не может быть решена за приемлемое время. Кроме того, при асинхронной реализации параллельный автомат не сводится к последовательному [13] и указанный подход вообще не может быть использован. Поэтому были разработаны методы специально для кодирования частичных состояний параллельного автомата [14–16], использующие приемы, отличающиеся от тех, которые применяются при кодировании состояний последовательных автоматов.

В настоящей работе предлагается сводить кодирование частичных состояний параллельного автомата к совместному кодированию состояний компонентных автоматов сети, реализующей заданный параллельный автомат. Каждый из компонентных автоматов данной сети является последовательным автоматом, и поэтому для кодирования его состояний можно использовать какой-либо из известных методов, предназначенных для последовательных автоматов. В статье показано, как использовать для совместного кодирования состояний компонентных автоматов сети, реализующей параллельный автомат, итеративный метод, описанный в работе [17]. Все такие автоматы могут быть синтезированы на основе программируемых логических интегральных схем, как показано, например, в работе [18]. Использование приема декомпозиции параллельного автомата, т. е. разложения его в сеть последовательных автоматов, позволяет снизить размерность задачи. Разложение проектируемого устройства на блоки дает возможность снизить энергопотребление путем блокировки синхронизирующих сигналов, подаваемых на некоторые блоки [19].

Описание модели. Параллельный автомат состоит из следующих объектов: $Q = \{q_1, q_2, \dots, q_r\}$ – множества частичных состояний; $X = \{x_1, x_2, \dots, x_n\}$ – множества входных булевых переменных; $Y = \{y_1, y_2, \dots, y_m\}$ – множества выходных булевых переменных и последовательности строк-переходов $T = \{\tau_1, \tau_2, \dots, \tau_r\}$, имеющих вид [13]

$$\tau_i = S_i : -K_i \rightarrow K_i' \rightarrow S_i', \quad (1)$$

где S_i и S_i' – подмножества множества Q ; K_i – элементарная конъюнкция переменных из множества X и K_i' – элементарная конъюнкция переменных из множества Y .

Все множество частичных состояний, в которых рассматриваемый параллельный автомат находится в некоторый момент времени, называется *глобальным состоянием*. Смысл строки (1) заключается в следующем. Если автомат находится одновременно в состояниях, составляющих множество S_i , и булевы переменные приняли значения, обращающие конъюнкцию K_i в единицу, то конъюнкция K_i' приобретает значение единица и автомат переходит из частичных состояний, составляющих множество S_i , в частичные состояния, составляющие множество S_i' . Другими словами, пусть $P = \{P_1, P_2, \dots, P_p\}$ – множество всех достижимых глобальных состояний заданного параллельного автомата. Тогда если $S_i \subseteq P_g$, где P_g – текущее глобальное состояние автомата, и моделируемое устройство принимает на входе двоичные сигналы, обращающие конъюнкцию K_i в единицу, то глобальным состоянием в следующий момент времени будет $P_h = (P_g \setminus S_i) \cup S_i'$, а на выходе устройство выдаст двоичные сигналы, обращающие конъюнкцию K_i' в единицу. Любая из конъюнкций K_i или K_i' может отсутствовать в строке (1). Отсутствие K_i означает ее тождественное равенство единице. Отсутствие K_i' означает в зависимости от интерпретации данной модели либо то, что все переменные из множества Y обращаются в нуль, либо то, что значения сигналов на выходе не меняются. В данной работе не будут рассматриваться выходные сигналы и конъюнкция K_i' будет опускаться. Так же, как для последовательного автомата, возможны синхронная и асинхронная реализации параллельного автома-

та. Далее будет рассматриваться синхронная реализация. При такой реализации время делится на фиксированные промежутки – такты, в течение одного такта автомат совершает переход между состояниями и выдает определенные выходные сигналы.

В данную модель внесены следующие ограничения.

1. Введено начальное глобальное состояние в виде одноэлементного множества. Для определенности можно взять $\{q_1\}$.

2. Для двух различных строк, i -й и j -й, если $S_i \cap S_j \neq \emptyset$, то $S_i = S_j$.

Существует ряд других ограничений, связанных с корректностью задания параллельного автомата и указанных в работе [11], которых не будем касаться, так как там решается другой круг задач. Не будем также обращать внимание на выходные сигналы, ограничимся рассмотрением переключений элементов памяти.

Примером параллельного автомата может служить следующая последовательность строк:

1: $-\bar{x}_1 x_2 \rightarrow 10$;	τ_1
10: $-\bar{x}_2 \rightarrow 2.3.4$;	τ_2
2: $\rightarrow 5$;	τ_3
3.5: $-x_2 \rightarrow 8$;	τ_4
4: $-\bar{x}_1 \rightarrow 7$;	τ_5
4: $-x_1 \rightarrow 9$;	τ_6
7: $-\bar{x}_2 \rightarrow 9$;	τ_7
8.9: $\rightarrow 6$;	τ_8
6: $-x_1 \rightarrow 1$.	τ_9

Здесь $Q = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ и $X = \{x_1, x_2\}$. За начальное глобальное состояние примем одноэлементное множество $\{1\}$. Первая строка (переход τ_1) означает, что если $x_1 = 0$ и $x_2 = 1$, то автомат выходит из состояния $\{1\}$ и в следующем такте входит в состояние $\{10\}$, которое также является глобальным. При любой другой комбинации значений x_1 и x_2 автомат остается в состоянии $\{1\}$. Из состояния $\{10\}$ при $x_2 = 0$ автомат переходит в частичные состояния 2, 3 и 4, которые составляют глобальное состояние $\{2, 3, 4\}$ (переход τ_2). В следующем такте автомат меняет частичное состояние 2 на частичное состояние 5 независимо от значений входных переменных. Поскольку рассматривается синхронная реализация, где переходы могут совершаться одновременно, переход τ_3 совершается одновременно с переходом τ_5 при $x_1 = 0$ или одновременно с переходом τ_6 при $x_1 = 1$. Глобальными состояниями при этом будут соответственно $\{3, 5, 7\}$ или $\{3, 5, 9\}$. Проследив таким образом функционирование данного параллельного автомата, получим еще глобальные состояния $\{7, 8\}$, $\{8, 9\}$ и $\{6\}$.

Построение сети последовательных автоматов, реализующей заданный параллельный автомат. Пусть задан некоторый параллельный автомат B , описанием которого является последовательность строк вида (1). Рассмотрим некоторую сеть $N = (X, A_1, A_2, \dots, A_n)$, где X – множество тех же входных булевых переменных, которые присутствуют в задании автомата B , и A_1, A_2, \dots, A_n – компонентные последовательные автоматы с множествами внутренних состояний Q_1, Q_2, \dots, Q_n . Каждый из автоматов A_j является автоматом без выходов, т. е. для него заданы только переходы, которые представим в форме, аналогичной (1):

$$q^j : -\alpha_j \rightarrow q^j, \quad (2)$$

где α_j – предикат над переменными из множества X и состояниями компонентных автоматов $A_1, A_2, \dots, A_{j-1}, A_{j+1}, \dots, A_n$, принимающий значение единица при определенной комбинации значений некоторых входных переменных и определенной совокупности состояний некоторых компонентных автоматов. Если $\alpha_j = 1$, то автомат A_j переходит из состояния q^j в состояние q^j , в противном случае остается в состоянии q^j .

Сеть N реализует автомат B , если существует такое отображение φ множества $D \subseteq Q_1 \times Q_2 \times \dots \times Q_n$ в некоторую совокупность подмножеств множества Q , что для любого

перехода (1) автомата B при $S_i \subseteq \Phi(q^1, q^2, \dots, q^n)$ имеем $S_i' \subseteq \Phi(q^{1'}, q^{2'}, \dots, q^{n'})$, если $\alpha_j = 1$, и $S_i' \subseteq \Phi(q^1, q^2, \dots, q^n)$, если $\alpha_j = 0$, согласно (2). В случае синхронной реализации параллельного автомата B можно построить эквивалентный ему последовательный автомат A с состояниями, соответствующими глобальным состояниям автомата B . Описанная сеть в этом случае реализует автомат A в смысле работ [20, 21].

Определим множество состояний Q_j компонентного автомата A_j ($j = 1, 2, \dots, n$) сети N следующим образом. Каждому состоянию $q^j \in Q_j$ поставим во взаимно-однозначное соответствие некоторое частичное состояние q автомата B , причем среди всех частичных состояний, соответствующих состояниям из Q_j , не должно быть ни одной пары параллельных состояний, т. е. таких, в которых автомат B может находиться одновременно. Указанное соответствие обозначим $f_j(q^j) = q$. Пусть множество компонентных автоматов $\{A_1, A_2, \dots, A_n\}$ сети N достаточное для того, чтобы для каждого частичного состояния q автомата B существовала по крайней мере одна компонента A_j с таким состоянием q^j , что $f_j(q^j) = q$.

Переходы в компонентных автоматах определим согласно выражению (2) следующим образом. Каждой строке τ_i вида (1) из описания автомата B поставим в соответствие совокупность переходов в тех компонентных автоматах A_j , любой из которых имеет такое состояние q^j , что $f_j(q^j) \in S_i \cup S_i'$. Для компонентного автомата A_j в качестве левой части выражения (2) возьмем то состояние q^j , для которого $f_j(q^j) \in S_i$. Такое состояние в Q_j единственное, поскольку по определению множества Q_j оно не содержит ни одной пары состояний, соответствующих параллельным частичным состояниям автомата B . Состояние $q^{j'}$ в выражении (2) определяем так, чтобы имело место $f_j(q^{j'}) \in S_i'$. При этом полагаем, что $\alpha_j = 1$ тогда и только тогда, когда $K_i = 1$ и $\{f_1(q^1), f_2(q^2), \dots, f_{j-1}(q^{j-1}), f_{j+1}(q^{j+1}), \dots, f_n(q^n)\} \supseteq S_i'$.

Множество последовательных автоматов $\{A_1, A_2, \dots, A_n\}$, образующих сеть N , которая реализует заданный параллельный автомат B , строим следующим образом. Получим все максимальные множества взаимно непараллельных частичных состояний автомата B . Максимальных в том смысле, что любое частичное состояние, не принадлежащее какому-либо из этих множеств, оказывается параллельным хотя бы одному из состояний, принадлежащих ему. Далее следует получить покрытие этими множествами всех пар частичных состояний, связанных переходами. Каждое множество из полученного покрытия соответствует множеству состояний одного из компонентных последовательных автоматов, составляющих искомую сеть N .

В статье [22] описан алгоритм установления параллельности частичных состояний, который для параллельных автоматов рассматриваемого класса решает данную задачу за полиномиальное время. В случае энергосберегающего кодирования, как отмечено далее, приходится рассматривать глобальные состояния параллельного автомата, и по ним легко установить параллельность частичных состояний. Для примера параллельного автомата, описание которого приведено выше и перечислены глобальные состояния, матрица отношения параллельности на множестве частичных состояний имеет вид

1	2	3	4	5	6	7	8	9	10	
0	0	0	0	0	0	0	0	0	0	1
0	0	1	1	0	0	0	0	0	0	2
0	1	0	1	1	0	1	0	1	0	3
0	1	1	0	0	0	0	0	0	0	4
0	0	1	0	0	0	1	0	1	0	5
0	0	0	0	0	0	0	0	0	0	6
0	0	1	0	1	0	0	1	0	0	7
0	0	0	0	0	0	1	0	1	0	8
0	0	1	0	1	0	0	1	0	0	9
0	0	0	0	0	0	0	0	0	0	10

Такую матрицу можно рассматривать как матрицу смежности графа, представляющего отношение параллельности. Упомянутые множества непараллельных частичных состояний соот-

ветствуют независимым множествам этого графа. Способы нахождения максимальных независимых множеств в графе описаны в работе [17]. В данном графе максимальными независимыми множествами являются $\{1, 2, 5, 6, 8, 10\}$, $\{1, 3, 6, 8, 10\}$, $\{1, 4, 5, 6, 8, 10\}$, $\{1, 2, 6, 7, 9, 10\}$ и $\{1, 4, 6, 7, 9, 10\}$. Табл. 1 является таблицей покрытия. В ней строкам соответствуют максимальные множества взаимно непараллельных частичных состояний, а столбцам – пары частичных состояний, связанных переходами.

Таблица 1

Table 1

	1-10	2-10	3-10	4-10	2-5	3-8	5-8	4-7	4-9	7-9	6-8	6-9	1-6
1,2,5,6,8,10	1	1			1		1						1
1,3,6,8,10	1		1			1					1		1
1,4,5,6,8,10	1			1			1						1
1,2,6,7,9,10	1	1								1		1	1
1,4,6,7,9,10	1			1				1	1	1		1	1

Кратчайшее покрытие (естественно потребовать минимум компонентных автоматов) составляют множества $\{1, 2, 5, 6, 8, 10\}$, $\{1, 3, 6, 8, 10\}$ и $\{1, 4, 6, 7, 9, 10\}$. Состояния компонентных автоматов обозначим теми же символами, которые обозначают соответствующие им частичные состояния заданного параллельного автомата. Таким образом, искомую сеть составляют три автомата: A_1 с множеством состояний $Q_1 = \{1, 2, 5, 6, 8, 10\}$, A_2 с множеством состояний $Q_2 = \{1, 3, 6, 8, 10\}$ и A_3 с множеством состояний $Q_3 = \{1, 4, 6, 7, 9, 10\}$. Переходы между состояниями определяются согласно выражению (2), как описано выше. Поведение получаемой сети N зададим в виде системы не полностью определенных функций $q^{1'}$, $q^{2'}$ и $q^{3'}$, принимающих значения соответственно из множеств Q_1 , Q_2 и Q_3 . Их аргументами являются булевы переменные x_1 , x_2 и многозначные переменные q^1 , q^2 и q^3 , принимающие значения из тех же множеств, что и указанные функции. Данную систему функций представляет табл. 2 – обычное табличное задание дискретных функций. В ней символом «–» обозначено безразличное значение переменной.

Таблица 2

Table 2

x_1x_2	q^1	q^2	q^3	$q^{1'}$	$q^{2'}$	$q^{3'}$
0 1	1	1	1	10	10	10
1 –	1	1	1	1	1	1
– 0	1	1	1	1	1	1
– 0	10	10	10	2	3	4
– 1	10	10	10	10	10	10
– –	2	–	–	5	–	–
– 1	5	3	–	8	8	–
– 0	5	3	–	5	3	–
0 –	–	–	4	–	–	7
1 –	–	–	4	–	–	9
– 0	–	–	7	–	–	9
– 1	–	–	7	–	–	7
– –	8	8	9	6	6	6
1 –	6	6	6	1	1	1
0 –	6	6	6	6	6	6

В результате кодирования состояний, т. е. замены каждой переменной q^j и, соответственно, $q^{j'}$ булевым вектором с компонентами z_i^j и, соответственно, $z_i^{j'}$, представляющими состояния двоичных элементов памяти, заданная табл. 2 система функций преобразуется в систему булевых функций.

Метод кодирования состояний последовательного автомата. Для кодирования состояний автомата используем итеративный способ, описанный в работе [17]. В его основе лежит поиск

максимального разреза во взвешенном графе. Пусть требуется закодировать состояния некоторого автомата A , т. е. каждому его состоянию q поставить в соответствие булев вектор (z_1, z_2, \dots, z_k) , называемый кодом состояния. Различным состояниям должны быть приписаны различные коды. Текущая ситуация в процессе выполнения данного способа характеризуется частичными кодами состояний (z_1, z_2, \dots, z_j) , $j < k$, и взвешенным графом $G = (V, E)$, вершины которого соответствуют состояниям автомата. Две вершины этого графа связаны ребром, если и только если соответствующие состояния имеют один и тот же частичный код. В начальной ситуации частичные коды пустые и граф является полным. Каждое ребро $v_s v_t \in E$ имеет вес w_{st} , пропорциональный величине $1 - p_{st}$, где p_{st} – вероятность перехода между состояниями q_s и q_t , соответствующими вершинам v_s и v_t , независимо от направления перехода. Ясно, что вероятность p_{st} равна сумме вероятностей переходов от q_s к q_t и от q_t к q_s . Очевидно, что для снижения переключательной активности элементов памяти расстояние по Хэммингу между кодами состояний q_s и q_t в пространстве булевых переменных z_1, z_2, \dots, z_k должно быть как можно малым, если вероятность p_{st} высока.

Процесс кодирования состояний заданного автомата представляет собой последовательность шагов. На i -м шаге находится разбиение множества вершин V графа G на подмножества V_1 и V_2 , вводится переменная z_i , которая получает значение 0 (или 1) для состояний, соответствующих вершинам из V_1 , и значение 1 (или 0) для состояний, соответствующих вершинам из множества V_2 . Затем удаляются ребра, соединяющие вершины из V_1 с вершинами из V_2 , и выполняется следующий $(i + 1)$ -й шаг. Процесс заканчивается, когда граф G становится пустым.

Задача разбиения множества V на подмножества V_1 и V_2 сводится к нахождению максимального разреза в графе G , т. е. такого разбиения, что сумма весов ребер, соединяющих вершины из V_1 с вершинами из V_2 , была бы максимальной. На последнем шаге остаются только те ребра, которые соответствуют парам состояний, связанных переходами со сравнительно большой вероятностью. Расстояния между кодами этих состояний равны единице. Для нахождения разреза можно применять описанный в статье [23] метод, представляющий собой последовательность шагов, на каждом из которых выбирается вершина v из множества V_2 и переносится в множество V_1 . Начальными значениями этих множеств являются $V_1 = \emptyset$ и $V_2 = V$, а вершина v выбирается следующим образом. Пусть d – сумма весов ребер, инцидентных вершине v , и c – сумма весов ребер, связывающих вершину v с вершинами из множества V_1 . Перенос вершины v из V_2 в V_1 увеличивает сумму весов ребер, соединяющих вершины из множества V_1 с вершинами из множества V_2 , на величину $d - 2c$, если она положительна. На первом шаге она равна d . На каждом шаге выбирается такая вершина v , для которой величина $d - 2c$ максимальна. Процесс заканчивается, когда эта величина окажется отрицательной или нулем для всех вершин из множества V_1 .

Вычисление вероятностей переходов. Для подсчета вероятностей переходов между состояниями последовательного автомата принимаются следующие предположения: автомат является полностью определенным; все состояния являются взаимно достижимыми, т. е. для любых двух состояний существует последовательность входных сигналов, переводящая автомат из одного состояния в другое; автомат работает достаточно долго.

Вероятность перехода последовательного автомата из состояния q_i в состояние q_j , вызываемого входным сигналом $\mathbf{x} = (x_1, x_2, \dots, x_n)$, равна вероятности прихода входного сигнала \mathbf{x} . Если имеется несколько входных сигналов, переводящих автомат из состояния q_i в состояние q_j , условная вероятность p'_{ij} такого перехода равна сумме вероятностей этих сигналов как вероятность несовместимых событий. Условием является то, что автомат находится в состоянии q_i . Абсолютная вероятность p_{ij} перехода из состояния q_i в состояние q_j в течение всего времени работы автомата равна произведению $p(q_i) p'_{ij}$, где $p(q_i)$ – вероятность того, что автомат находится в состоянии q_i (это событие и приход сигналов, меняющих состояние q_i на q_j , являются независимыми событиями).

Для вычисления вероятностей $p(q_i)$, $i = 1, 2, \dots, m$, где m – число состояний автомата, можно использовать уравнения Чэпмена – Колмогорова для дискретных Марковских цепей [24]. Подобно закону Кирхгоффа из электротехники сумма вероятностей переходов в некоторое состо-

яние равно сумме вероятностей переходов из этого состояния. На основании приведенных соображений можно вывести следующую систему уравнений с неизвестными $p(q_i)$ ($i = 1, 2, \dots, m$):

$$\sum_{i=1}^m p(q_i) p'_{ij} = p(q_j), \quad j = 1, 2, \dots, m,$$

$$\sum_{i=1}^m p(q_i) = 1.$$

Вероятности p'_{ij} должны быть известны. Таким образом, решив эту систему уравнений, получим вероятности $p(q_i)$. Как было сказано ранее, абсолютная вероятность p_{ij} определяется как $p_{ij} = p(q_i) p'_{ij}$.

Совместное кодирование состояний компонентных автоматов. Для компонентного автомата в сети входной сигнал, кроме внешних сигналов сети, могут составлять сигналы о состояниях других компонентных автоматов. Благодаря внутренним связям то, что компонентный автомат находится в определенном состоянии, и приход некоторого входного для него сигнала нельзя считать независимыми событиями. Для вычисления вероятностей переходов между состояниями компонентных автоматов предлагается использовать моделирование синхронного параллельного автомата последовательным автоматом, состояниями которого являются глобальные состояния заданного параллельного автомата [11]. Вероятности переходов между глобальными состояниями определяются, как показано выше для моделирующего последовательного автомата, а вероятности переходов между частичными состояниями и, соответственно, между состояниями компонентных автоматов определяются через вероятности переходов между глобальными состояниями. Вероятность перехода между частичными состояниями q_i и q_j равна сумме вероятностей переходов между теми глобальными состояниями P_s и P_r , для которых $q_i \in P_s$ и $q_j \in P_r$ или $q_i \in P_r$ и $q_j \in P_s$.

Переходы между глобальными состояниями параллельного автомата из рассматриваемого примера представлены в виде табл. 3, где строки и столбцы соответствуют глобальным состояниям, и на пересечении строки и столбца стоит условие перехода из состояния, соответствующего строке, в состояние, соответствующее столбцу. По этой таблице легко определяются условные вероятности переходов, представленные в табл. 4. Считается, что входные двоичные сигналы x_1 и x_2 независимы и равновероятны.

Таблица 3

Table 3

	1	10	2.3.4	3.5.7	3.5.9	7.8	8.9	6
1	$x_1 \vee \bar{x}_2$	$\bar{x}_1 x_2$						
10		x_2	\bar{x}_2					
2.3.4				\bar{x}_1	x_1			
3.5.7					\bar{x}_2	x_2		
3.5.9					x_2		\bar{x}_2	
7.8						x_2	\bar{x}_2	
8.9								1
6	x_1							\bar{x}_1

Таблица 4

Table 4

	1	10	2.3.4	3.5.7	3.5.9	7.8	8.9	6
1	3/4	1/4						
10		1/2	1/2					
2.3.4				1/2	1/2			
3.5.7					1/2	1/2		
3.5.9					1/2		1/2	
7.8						1/2	1/2	
8.9								1
6	1/2							1/2

Вероятности переходов между частичными состояниями параллельного автомата совпадают с вероятностями переходов между соответствующими состояниями компонентных автоматов в реализующей его сети.

Матрица смежности взвешенного графа G , общего для автоматов A_1 , A_2 и A_3 , где в связи с ее симметричностью опущены значения ниже главной диагонали ввиду симметричности матрицы и сама диагональ, имеет следующий вид:

$$\begin{array}{cccccccccc}
 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
 \left[\begin{array}{cccccccccc}
 25 & 25 & 25 & 25 & 23 & 25 & 25 & 25 & 23 & \\
 & 0 & 0 & 23 & 25 & 25 & 25 & 25 & 23 & \\
 & & 0 & 0 & 25 & 0 & 23 & 0 & 23 & \\
 & & & 25 & 25 & 24 & 25 & 24 & 23 & \\
 & & & & 25 & 0 & 23 & 0 & 25 & \\
 & & & & & 25 & 23 & 23 & 25 & \\
 & & & & & & 0 & 23 & 25 & \\
 & & & & & & & 0 & 25 & \\
 & & & & & & & & & 25
 \end{array} \right] \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array}
 \end{array}$$

Нули в матрице относятся к тем парам состояний, которые принадлежат различным автоматам, и поэтому соответствующие вершины не должны быть связаны ребрами. В качестве весов ребер w_{ij} , как было сказано выше, должны быть величины, пропорциональные разностям $1 - p_{ij}$. Здесь в качестве значений w_{ij} взяты числители этих дробей при общем знаменателе 25.

При формировании разреза графа G начальными значениями множеств V_1 и V_2 являются \emptyset и $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Суммы весов ребер, инцидентных вершинам, принимают следующие значения:

$$d(1) = 221, d(2) = 171, d(3) = 96, d(4) = 171, d(5) = 146, d(6) = 219, d(7) = 147, d(8) = 169, \\
 d(9) = 145, d(10) = 217.$$

Максимум имеет вершина 1, поэтому она переносится в множество V_1 : $V_1 = \{1\}$, $V_2 = \{2, 3, 4, 5, 6, 7, 8, 9, 10\}$. Теперь для вершин из множества V_2 вычисляются значения $h(v) = d(v) - 2c(v)$, где $c(v)$ – сумма весов ребер, связывающих вершину v с вершинами из множества V_1 :

$$h(2) = 171 - 2 \cdot 25 = 121, h(3) = 96 - 2 \cdot 25 = 46, h(4) = 171 - 2 \cdot 25 = 121, \\
 h(5) = 146 - 2 \cdot 25 = 96, h(6) = 219 - 2 \cdot 23 = 173, h(7) = 147 - 2 \cdot 25 = 97, \\
 h(8) = 169 - 2 \cdot 25 = 119, h(9) = 145 - 2 \cdot 25 = 95, h(10) = 217 - 2 \cdot 23 = 171.$$

По максимуму значения h из V_2 в V_1 переносится вершина 6 и получается разбиение $\{1, 6\}$, $\{2, 3, 4, 5, 7, 8, 9, 10\}$. Процесс переноса вершин из V_2 в V_1 продолжается до тех пор, пока значение h для всех вершин в V_2 не станет нулевым или отрицательным. В результате получим разбиение $\{1, 2, 4, 6, 10\}$, $\{3, 5, 7, 8, 9\}$, или разрез максимального веса. (Возможно, этот максимум не является наибольшим в силу эвристического характера применяемого алгоритма.) Соответственно, вводим кодирующую переменную z_1 со следующими значениями:

$$\begin{array}{cccccccccc}
 \text{состояния компонентных автоматов:} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\
 \text{значения переменной } z_1: & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0
 \end{array}$$

Из графа G удаляются ребра, принадлежащие найденному разрезу, и в полученном графе так же находится разрез. Теперь матрица смежности примет вид

$$\begin{array}{cccccccccc} 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & \\ \left[\begin{array}{cccccccccc} 25 & 0 & 25 & 0 & 23 & 0 & 0 & 0 & 23 & \\ & 0 & 0 & 0 & 25 & 0 & 0 & 0 & 23 & \\ & & 0 & 0 & 0 & 0 & 23 & 0 & 0 & \\ & & & 0 & 25 & 0 & 0 & 0 & 23 & \\ & & & & 0 & 0 & 23 & 0 & 0 & \\ & & & & & 0 & 0 & 0 & 25 & \\ & & & & & & 0 & 23 & 0 & \\ & & & & & & & 0 & 0 & \\ & & & & & & & & 0 & \end{array} \right] \begin{array}{l} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \end{array}$$

Разрез в этом графе определяет разбиение $\{1, 5, 6, 7, 8\}$, $\{2, 3, 4, 9, 10\}$, а переменная z_2 имеет следующие значения:

$$\begin{array}{l} \text{состояния компонентных автоматов: } 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10 \\ \text{значения переменной } z_2: \quad 0 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \end{array}$$

Для определения значений переменной z_3 необходимо удалить из графа G не только ребра найденного разреза, но и изолированные вершины. Для состояний, соответствующих изолированным вершинам, переменная z_3 приобретает неопределенное значение. Матрица смежности графа G теперь имеет вид

$$\begin{array}{cccccc} 2 & 4 & 5 & 6 & 8 & 10 \\ \left[\begin{array}{cccccc} 0 & 0 & 0 & 23 & 0 & 0 \\ & 0 & 0 & 0 & 0 & 23 \\ & & 0 & 0 & 0 & 23 \\ & & & 0 & 23 & 0 \\ & & & & 0 & 0 \\ & & & & & 0 \end{array} \right] \begin{array}{l} 1 \\ 2 \\ 4 \\ 5 \\ 6 \\ 8 \end{array} \end{array}$$

Все ребра из этого графа составляют единственный разрез, разделяющий множество вершин на $\{1, 5, 10\}$ и $\{2, 4, 6, 8\}$, и по этому разрезу находим значения переменной z_3 :

$$\begin{array}{l} \text{состояния компонентных автоматов: } 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9 \ 10 \\ \text{значения переменной } z_2: \quad 0 \ 1 \ - \ 1 \ 0 \ 1 \ - \ 1 \ - \ 0 \end{array}$$

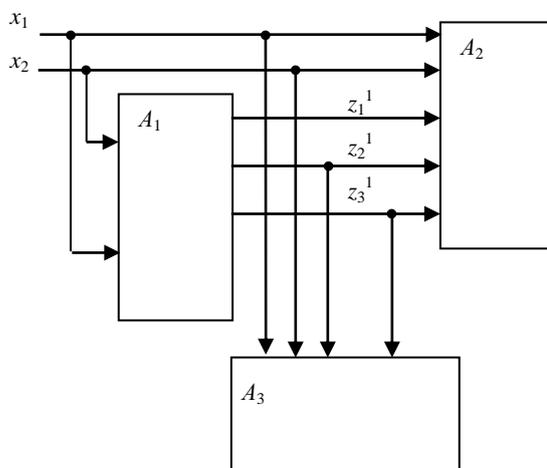
Таким образом, получаем матрицы C_1 , C_2 и C_3 кодирования состояний автоматов A_1 , A_2 и A_3 соответственно:

$$C_1 = \begin{array}{ccc} z_1^1 & z_2^1 & z_3^1 \\ \left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{array} \right] \begin{array}{l} 1 \\ 2 \\ 5 \\ 6 \\ 8 \\ 10 \end{array} \end{array}, \quad C_2 = \begin{array}{ccc} z_1^2 & z_2^2 & z_3^2 \\ \left[\begin{array}{ccc} 0 & 0 & 0 \\ 1 & 1 & - \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{array} \right] \begin{array}{l} 1 \\ 3 \\ 6 \\ 8 \\ 10 \end{array} \end{array}, \quad C_3 = \begin{array}{ccc} z_1^3 & z_2^3 & z_3^3 \\ \left[\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & - \\ 1 & 1 & - \\ 0 & 1 & 0 \end{array} \right] \begin{array}{l} 1 \\ 4 \\ 6 \\ 7 \\ 9 \\ 10 \end{array} \end{array}.$$

Качество кодирования состояний автомата можно оценить введенной в работе [5] величиной $D = \sum p_{ij}(d_{ij} - 1)$, где p_{ij} – вероятность перехода между состояниями q_i и q_j в любом направлении, d_{ij} – расстояние по Хэммингу между кодами состояний q_i и q_j , а суммирование ведется по всем парам состояний автомата. Очевидно, что чем меньше эта величина, тем лучше решение

$$U_3 = \begin{bmatrix} x_1 & x_2 & z_2^1 & z_3^1 & z_1^3 & z_2^3 & z_3^3 \\ 0 & 1 & - & - & 0 & - & 0 \\ - & 0 & 1 & - & - & - & - \\ - & - & 1 & - & - & 1 & 0 \\ - & - & - & - & 0 & 1 & 1 \\ 1 & - & - & - & 0 & 1 & - \\ - & 0 & - & - & 1 & 0 & - \\ - & - & - & - & 1 & 0 & - \\ - & - & - & - & 1 & - & - \\ 0 & - & - & 1 & - & - & - \end{bmatrix}, V_3 = \begin{bmatrix} z_1^3 & z_2^3 & z_3^3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$

На рисунке показана полученная сеть последовательных автоматов, реализующая заданный параллельный автомат.



Сеть последовательных автоматов
Network of sequential automata

Заключение. Описанный подход к кодированию частичных состояний параллельного автомата может быть применен при синтезе распределенных систем управления. В таких системах, управляющих некоторой совокупностью отдаленных друг от друга объектов, все блоки связаны информационно в единую сеть и каждый блок находится непосредственно на объекте управления. При этом возникает проблема распределения выходных сигналов по компонентам сети. В настоящей статье выходные сигналы не рассматривались, все внимание было сосредоточено на функции возбуждения элементов памяти. Метод энергосберегающего кодирования частичных состояний параллельного автомата рассчитан на использование его в автоматизированной системе логического проектирования. Сравнение результатов применения предложенного метода с результатами кодирования частичных состояний без учета интенсивности переключений элементов памяти показывает, что данный метод показывает лучший результат. Использование декомпозиции параллельного автомата позволяет снизить размерность трудоемкой задачи кодирования состояний.

Список использованных источников

1. Мурога, С. Системное проектирование сверхбольших интегральных схем : в 2-х кн. / С. Мурога. – М. : Мир, 1985. – Кн. 1. – 288 с.
2. Pedram, M. Power minimization in IC design: Principles and applications / M. Pedram // ACM Transactions on Design Automation of Electronic Systems. – 1996. – Vol. 1. – P. 3–56.

3. Kashirova, L. State assignment of finite state machine for decrease of power dissipation / L. Kashirova, A. Keevallik, M. Meshkov // *Second Intern. Conf. Computer-Aided Design of Discrete Devices (CAD DD'97)*, Minsk, Republic of Belarus, 12–14 Nov. 1997. – Minsk : Institute of Engineering Cybernetics of the NASB, 1997. – Vol. 1. – P. 60–67.
4. Sudnitson, A. Partition search for FSM low power synthesis / A. Sudnitson // *Fourth Intern. Conf. Computer-Aided Design of Discrete Devices (CAD DD'2001)*, Minsk, Republic of Belarus, 14–16 Nov. 2001. – Minsk : Institute of Engineering Cybernetics of the NASB, 2001. – Vol. 1. – P. 44–49.
5. Закревский, А. Д. Алгоритмы энергосберегающего кодирования состояний автомата / А. Д. Закревский // *Информатика*. – 2011. – № 1(29). – С. 68–78.
6. Поттосин, Ю. В. Кодирование состояний дискретного автомата, ориентированное на уменьшение энергопотребления реализующей схемы / Ю. В. Поттосин // *Прикладная дискретная математика*. – 2011. – № 4(14). – С. 62–71.
7. Поттосин, Ю. В. Энергосберегающее противогоночное кодирование состояний асинхронного автомата / Ю. В. Поттосин // *Информатика*. – 2015. – № 2(46). – С. 94–101.
8. Pottosin, Yu. Race-free state assignment for low power asynchronous automaton / Yu. Pottosin // *Further Improvements in the Boolean Domain* / ed. B. Steinbach. – Cambridge Scholars Publishing, 2018. – P. 253–267.
9. Pottosin, Yu. V. Low power assignment of partial states of a parallel automaton / Yu. V. Pottosin // *Прикладная дискретная математика*. – 2022. – № 56. – С. 113–122.
10. Pottosin, Yu. Optimal state assignment of synchronous parallel automata / Yu. Pottosin // *Design of Embedded Control Systems*. – N. Y. : Springer, 2005. – P. 111–124.
11. Закревский, А. Д. Параллельные алгоритмы логического управления / А. Д. Закревский. – М. : УРСС, 2003. – 304 с.
12. Питерсон, Дж. Теория сетей Петри и моделирование систем / Дж. Питерсон. – М. : Мир, 1984. – 264 с.
13. Закревский, А. Д. Логические основы проектирования дискретных устройств / А. Д. Закревский, Ю. В. Поттосин, Л. Д. Черемисинова. – М. : Физматлит, 2007. – 592 с.
14. Закревский, А. Д. Блочное кодирование частных состояний у автоматов, реализующих параллельные алгоритмы логического управления / А. Д. Закревский // *Изв. АН СССР. Техническая кибернетика*. – 1983. – № 5. – С. 3–11.
15. Черемисинова, Л. Д. Реализация асинхронными автоматами параллельных алгоритмов логического управления / Л. Д. Черемисинова // *Автоматика и вычислительная техника*. – 1985. – № 2. – С. 65–69.
16. Поттосин, Ю. В. Декомпозиционный метод кодирования состояний параллельного автомата / Ю. В. Поттосин // *Автоматика и вычислительная техника*. – 1987. – № 1. – С. 84–91.
17. Поттосин, Ю. В. Комбинаторные задачи в логическом проектировании дискретных устройств / Ю. В. Поттосин. – Минск : Беларус. навука, 2021. – 175 с.
18. Соловьев, В. В. Проектирование цифровых систем на основе программируемых логических интегральных схем / В. В. Соловьев. – М. : Горячая линия – Телеком, 2001. – 636 с.
19. Piguet, C. Low-power and low-voltage CMOS digital design / C. Piguet // *Microelectronic Engineering*. – 1997. – No. 39. – P. 179–208.
20. Hartmanis, J. Algebraic structure theory of sequential machines / J. Hartmanis, R. E. Stearns. – Englewood Cliffs, N. J. Prentice Hall Inc., 1966. – 208 p.
21. Кээваллик, А. Э. Теорема декомпозиции конечных автоматов / А. Э. Кээваллик // *Автоматика и вычислительная техника*. – 1974. – № 1. – С. 17–24.
22. Ковалев, А. В. О нахождении отношения параллельности на множестве мест одного подкласса сетей Петри / А. В. Ковалев // *Вест. Акад. наук Беларуской ССР. Сер. фіз.-матэм. навук*. – 1989. – № 2. – С. 106–110.
23. Закревский, А. Д. Раскраска графов при декомпозиции булевых функций / А. Д. Закревский // *Логическое проектирование*. – Минск : Ин-т техн. кибернетики НАН Беларуси, 2000. – Вып. 5. – С. 83–97.
24. Macii, E. High-level power modeling, estimation and optimization / E. Macii, M. Pedram, F. Somenzi // *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*. – 1998. – Vol. 17, no. 11. – P. 1061–1079.

References

1. Muroga S. *VLSI System Design: When and How to Design Very-Large-Scale Integrated Circuits*, 1st edition. Wiley, 1982, 496 p.
2. Pedram M. Power minimization in IC design: Principles and applications. *ACM Transactions on Design Automation of Electronic Systems*, 1996, vol. 1, pp. 3–56.
3. Kashirova L., Keevallik A., Meshkov A. M. State assignment of finite state machine for decrease of power dissipation. *Second International Conference Computer-Aided Design of Discrete Devices (CAD DD'97)*, Minsk, Republic of Belarus, 12–14 November 1997. Minsk, Institute of Engineering Cybernetics of the National Academy of Sciences of Belarus, 1997, vol. 1, pp. 60–67.
4. Sudnitson A. Partition search for FSM low power synthesis. *Fourth International Conference Computer-Aided Design of Discrete Devices (CAD DD'2001)*, Minsk, Republic of Belarus, 14–16 November 2001. Minsk, Institute of Engineering Cybernetics of the National Academy of Sciences of Belarus, 2001, vol. 1, pp. 44–49.
5. Zakrevskij A. D. *Algorithms for low power state assignment of an automaton*. Informatika [Informatics], 2011, no. 1(29), pp. 68–78 (In Russ.).
6. Pottosin Yu. V. *State assignment of a discrete automaton to decrease power consumption of the implementing circuit*. Prikladnaya diskretnaya matematika [Discrete Applied Mathematics], 2011, no. 4(14), pp. 62–71 (In Russ.).
7. Pottosin Yu. V. *Low power race-free state assignment of an asynchronous automaton*. Informatika [Informatics], 2015, no. 2(46), pp. 94–101 (In Russ.).
8. Pottosin Yu. Race-free state assignment for low power asynchronous automaton. *Further Improvements in the Boolean Domain*. In B. Steinbach (ed.). Cambridge Scholars Publishing, 2018, pp. 253–267.
9. Pottosin Yu. V. Low power assignment of partial states of a parallel automaton. *Prikladnaya diskretnaya matematika [Discrete Applied Mathematics]*, 2022, no. 56, pp. 113–122.
10. Pottosin Yu. Optimal state assignment of synchronous parallel automata. *Design of Embedded Control Systems*. New York, Springer, 2005, pp. 111–124.
11. Zakrevskij A. D. Parallel'nye algoritmy logicheskogo upravleniya. *Parallel Algorithms for Logical Control*. Moscow, URSS, 2003, 304 p. (In Russ.).
12. Peterson J. L. *Petri Net Theory and the Modeling of Systems*, 1st edition. Prentice Hall, 1981, 290 p.
13. Zakrevskij A. D., Pottosin Yu. V., Cheremisinova L. D. Logicheskie osnovy proektirovaniya diskretnyh ustrojstv. *Logical Fundamentals of Discrete Devices Design*. Moscow, Fizmatlit, 2007, 592 p. (In Russ.).
14. Zakrevskij A. D. *Block partial state assignment of automata that implement parallel algorithms for logical control*. Izvestija Akademii nauk Sojuza Sovetskikh Socialisticheskikh Respublik. Tehnicheskaja kibernetika [Proceedings of the Academy of Sciences of the Union of Soviet Socialist Republics. Technical Cybernetics], 1983, no. 5, pp. 3–11 (In Russ.).
15. Cheremisinova L. D. *Implementation of parallel algorithms for logical control by asynchronous automata*. Avtomatika i vychislitel'naya tehnika [Automation and Computer Engineering], 1985, no. 2, pp. 65–69 (In Russ.).
16. Pottosin Yu. V. *A decomposition method for state assignment of a parallel automaton*. Avtomatika i vychislitel'naya tehnika [Automation and Computer Engineering], 1987, no. 1, pp. 84–91 (In Russ.).
17. Pottosin Yu. V. *Kombinatornye zadachi v logicheskom proektirovanii diskretnyh ustrojstv*. *Combinatorial Problems in Logical Design of Discrete Devices*. Minsk, Belaruskaya navuka, 2021, 175 p. (In Russ.).
18. Solov'yov V. V. *Proektirovanie cifrovyyh sistem na osnove programmiruemykh logicheskikh integral'nyh shem*. *The Design of Digital Systems Based on Programmable Logical Integrated Circuits*. Moscow, Goryachaya linia – Telekom, 2001, 636 p. (In Russ.).
19. Pigué C. Low-power and low-voltage CMOS digital design. *Microelectronic Engineering*, 1997, no. 39, pp. 179–208.
20. Hartmanis J., Stearns R.E. *Algebraic Structure Theory of Sequential Machines*. Englewood Cliffs, N. J. Prentice Hall Inc., 1966, 208 p.
21. Keevallik A. *Decomposition theorem of finite automata*. Avtomatika i vychislitel'naya tehnika [Automation and Computer Engineering], 1974, no. 1, pp. 17–24 (In Russ.).
22. Kovalyov A. V. *About finding parallelism relation on the set of positions of a subclass of Petri nets*. Vesci Akademiï navuk Belaruskaj Saveckaj Sacyjalistychnaj Rjespubliki. Seryja fizika-matjematychnyh navuk [Proceedings of the Academy of Sciences of the Belarusian Soviet Socialist Republic. Physics and Mathematics Series], 1989, no. 2, pp. 106–110 (In Russ.).

23. Zakrevskij A. D. *Graph coloring during the decomposition of Boolean functions*. Logicheskoe proektirovanie [*Logical Design*]. Minsk, Institut tehniczeskoj kibernetiki Nacional'noj akademii nauk Belarusi, 2000, iss. 5, pp. 83–97 (In Russ.).

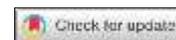
24. Macii E., Pedram M., Somenzi F. High-level power modeling, estimation and optimization. *IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems*, 1998, vol. 17, no. 11, pp. 1061–1079.

Информация об авторе

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

Information about the author

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



УДК 519.711
<https://doi.org/10.37661/1816-0301-2023-20-1-91-101>

Оригинальная статья
Original Paper

Аппаратная реализация булевых функций на основе автоматной модели

А. А. Бутов

Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
E-mail: tmkrb9@gmail.com

Аннотация

Цели. В настоящее время электронные устройства управления все шире внедряются в различные изделия бытового и производственного назначения. В качестве таких устройств широко применяются микроконтроллеры самой разной конфигурации. Можно предложить другой подход, где устройство управления со стандартной структурой синтезируется из типовых интегральных схем и реализует булеву функцию, описывающую требуемые управляющие воздействия. Целью работы является исследование возможности реализации булевых функций с помощью устройств со стандартной структурой, проектирование которых основано на использовании модели дискретного автомата.

Методы. Исходная булева функция, подлежащая реализации, задается в виде дизъюнктивной нормальной формы. Для нее строится бинарная диаграмма решений (англ. Binary Decision Diagram, BDD), оптимизированная по числу вершин, на основе которой формируется граф переходов синхронного автомата Мура с абстрактным состоянием. Далее после выполнения этапа кодирования состояний автомата на основе его таблицы переходов формируется входная информация для прошивки (программирования) матричной памяти постоянного запоминающего устройства (ПЗУ).

Результаты. Устройство, реализующее булеву функцию на основе автоматной модели, синтезируется из типовых микросхем. Основным компонентом служит ПЗУ, которое в соответствии со стандартной структурой устройства дополняется сдвиговым регистром, регистром состояний, триггером и тремя селекторами начального и двух финальных состояний.

Заключение. Процесс проектирования устройства со стандартной структурой, реализующего булеву функцию, в итоге сводится к программированию матричной памяти ПЗУ на основе автоматной таблицы переходов. Использование многократно программируемого ПЗУ позволяет изменять функциональность устройства при сохранении прежней схемной реализации. Недостатком такого устройства, так же как и устройств, реализованных на основе микроконтроллеров, является низкое быстродействие, достоинством – возможность использования в различных изделиях и приборах, прежде всего бытового назначения, которые не требуют высокоскоростной реакции на изменение входного сигнала.

Ключевые слова: синтез комбинационных схем, булева функция, дискретный автомат, бинарная диаграмма решений, постоянное запоминающее устройство

Для цитирования. Бутов, А. А. Аппаратная реализация булевых функций на основе автоматной модели / А. А. Бутов // Информатика. – 2023. – Т. 20, № 1. – С. 91–101.

<https://doi.org/10.37661/1816-0301-2023-20-1-91-101>

Конфликт интересов. Автор заявляет об отсутствии конфликта интересов.

Поступила в редакцию | Received 19.12.2022

Подписана в печать | Accepted 20.01.2023

Опубликована | Published 29.03.2023

Hardware implementation of Boolean functions based on the automaton model

Alexey A. Butov

*Belarusian State University of Informatics and Radioelectronics,
st. P. Brovka, 6, Minsk, 220013, Belarus
E-mail: tmkrb9@gmail.com*

Abstract

Objectives. Currently, electronic control devices are increasingly being introduced into various household and production products. Microcontrollers of a wide variety of configurations are widely used as such devices. Another approach can be proposed where a control device with a standard structure is synthesized from typical integrated circuits and implements a Boolean function describing the required control actions. The purpose of the work is to investigate the possibility of implementing Boolean functions using devices with a standard structure, the design of which is based on the use of a discrete automaton model.

Methods. The original Boolean function to be implemented is given as a disjunctive normal form. A binary decision diagram (BDD) is built for such function, optimized by the number of vertices, on the basis of which a graph of transitions of a synchronous Moore automaton with an abstract state is formed. Further, after performing the state encoding step of the machine, input information for flashing (programming) of the matrix memory of the read-only memory (ROM) is generated based on its transition table.

Results. A device that implements a Boolean function based on an automaton model is synthesized from typical microcircuits. The main component is ROM, which, according to the standard structure of the device, is supplemented by a shift register, a state register, a trigger and three selectors of the initial and two final states.

Conclusion. The process of designing a device with standard structure that implements the Boolean function, as a result, comes down to programming the ROM matrix memory based on an automaton transition table. The use of a reprogrammable ROM allows to change the functionality of the device while maintaining the previous circuit implementation. The disadvantage of such a device, as well as devices implemented on the basis of microcontrollers, is the low speed, the advantage is the possibility of use it in various products and devices, primarily for household purposes, which do not require a high-speed response to the change of input signal.

Keywords: synthesis of combination circuits, Boolean function, discrete automaton, binary decision diagram, read-only memory

For citation. Butov A. A. *Hardware implementation of Boolean functions based on the automaton model*. *Informatika [Informatics]*, 2023, vol. 20, no. 1, pp. 91–101 (In Russ.).
<https://doi.org/10.37661/1816-0301-2023-20-1-91-101>

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

Введение. В настоящее время электронные устройства управления применяются не только в сложных технических устройствах и системах, но и в различных широко распространенных предметах повседневного пользования: бытовой технике, медицинском оборудовании, измерительной технике, электронных детских игрушках, торговых автоматах, роботах и др. Это в значительной мере обусловлено применением в них микроконтроллеров [1], выполняющих функции управления и контроля. Микросхема микроконтроллера предварительно программируется на выполнение определенных функций по управлению устройством, в которое она интегрирована, путем анализа значений входных переменных и выдачи команд, направленных на изменение состояния этого устройства. Микроконтроллеры применяются там, где не предъявляются жесткие требования к обеспечению быстродействия устройств, что характерно для огромного числа реальных объектов, допускающих автоматизацию их работы.

Электронной промышленностью выпускается широкий спектр различных микроконтроллеров, для которых имеются соответствующие среды разработки, автоматизирующие процесс прошивки памяти микроконтроллера. Для использования микроконтроллера в каком-либо проектируемом изделии необходимо выбрать не только микроконтроллер с нужной функциональностью, но и подходящую интегрированную среду разработки (Integrated Development

Environment, IDE), а также освоить работу в этой среде, включая изучение используемого в ней языка программирования (возможно, специализированного), программных библиотек, отладчика (симулятора) и др.

В настоящей статье предлагается другой подход к разработке подобных средств автоматизации, развиваемый в русле таких работ, как, например, [2–4]. Согласно этому подходу сначала строится автоматная модель, по которой с помощью типовых интегральных схем серий K155, K176, K555 или аналогичных им синтезируется устройство управления со стандартной структурой. Это устройство вычисляет значение булевой функции за некоторое число шагов, выполняемых в соответствии с автоматной таблицей переходов.

Описание подхода. Комбинационную схему (комбинационное устройство) с n входами и одним выходом на уровне функциональной модели можно задать булевой функцией $y = f(\mathbf{x})$, где компонентами вектора $\mathbf{x} = (x_1, x_2, \dots, x_n)$ являются булевы переменные, образующие в совокупности множество X .

Можно считать, что комбинационные схемы отличаются от последовательностных схем тем, что имеют лишь одно внутреннее состояние, в то время как последние, описываемые автоматной моделью, могут изменять свое внутреннее состояние и, соответственно, поведение.

Суть предлагаемого подхода к проектированию устройства, реализующего булеву функцию f , заключается в том, что оно проектируется как последовательностная схема, основанная на синхронной автоматной модели Мура с абстрактным состоянием. В данной модели в каждом состоянии автомата анализируется значение лишь одной из переменных множества X . В следующий дискретный момент времени в зависимости от значения этой переменной выполняется переход в одно из двух новых состояний, в каждом из которых будет анализироваться значение какой-то другой переменной из множества X и т. д.

Процесс последовательного анализа значений переменных множества X , сопровождающийся изменением состояний автомата, заканчивается тогда, когда автомат окажется в одном из двух финальных состояний, соответствующих значениям нуля и единицы функции f . При этом будут последовательно проанализированы значения всех переменных, входящих в множество X или в некоторое его подмножество. В последнем случае процесс достижения финального состояния заканчивается раньше, так как значение функции f в какой-то момент времени становится известным и дальнейший анализ уже не требуется. После достижения одного из финальных состояний будет определено значение функции f . В следующий дискретный момент времени выполняется безусловный переход в начальное состояние автомата и тем самым запускается новый аналогичный процесс последовательного анализа входных переменных с соответствующей сменой внутренних состояний для нахождения обновленного значения функции f . Значения входных переменных x_1, x_2, \dots, x_n каждый раз обновляются в момент перехода автомата в начальное состояние, а значение функции f может корректироваться только в момент очередного достижения автоматом одного из финальных состояний. С использованием данной модели значение сигнала на выходе устройства будет обновляться с некоторой задержкой, поскольку каждый переход автомата из одного состояния в другое требует одного такта работы тактового генератора.

Обобщенная структура устройства. Компоненты устройства, работающего на основе автоматной модели, показаны на рис. 1. Для простоты предполагается, что изменение состояния каждого из ее компонентов, обладающих памятью, происходит (с некоторой задержкой, вызванной инерцией срабатывания электронных схем) только в момент изменения управляющего сигнала от уровня логического нуля к уровню логической единицы, т. е. по положительному перепаду сигнала или, как говорят, по его переднему фронту. Отметим, что разные серии микросхем и входящие в них компоненты имеют свои особенности и используют в целях управления своими состояниями не только положительные, но и отрицательные перепады. Во втором случае соответствующие входы называются инверсными и обозначаются на рисунках небольшим кружком.

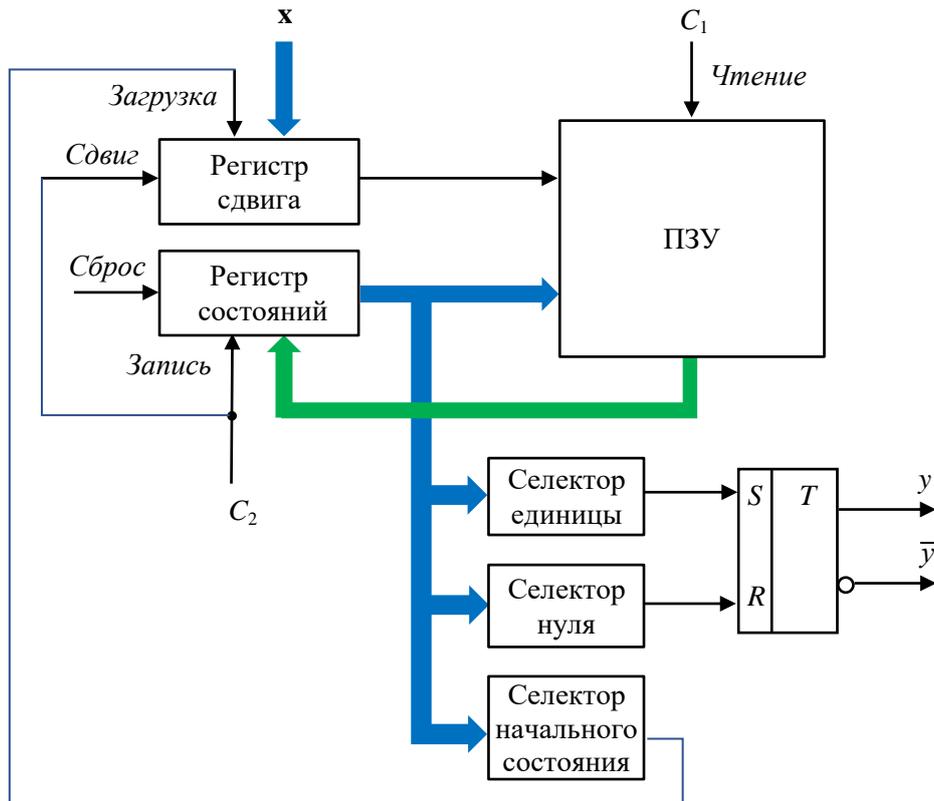


Рис. 1. Обобщенная структура устройства, реализующего булеву функцию
 Fig. 1. Generalized structure of the device implementing the Boolean function

Основным компонентом устройства является ПЗУ, которое имеет v входов и w выходов. ПЗУ обладает двухуровневой структурой, которая включает в себя:

- полный дешифратор, преобразующий v -разрядный двоичный код, который поступает на его входы, в 2^v -разрядный унитарный код, в котором только один разряд может иметь значение единицы, а все остальные – нуля;
- программируемую матричную память, организованную в виде 2^v w -разрядных двоичных слов.

При подаче на входы ПЗУ v -разрядного двоичного кода в матричной памяти посредством дешифратора выбирается соответствующее w -разрядное слово, которое передается на выходы ПЗУ. Все входы ПЗУ разбиты на две неравные части: одна часть включает в себя лишь один вход, другая – остальные $v-1$ входов, объединенных для удобства графического изображения в параллельную шину. Также в параллельную шину объединены и w выходов, где $w = v-1$. Информация на выходах ПЗУ обновляется (в соответствии с адресуемым словом) передним фронтом тактового импульса C_1 в момент поступления его на управляющий вход «Чтение».

В n -разрядный регистр сдвига, входы которого для удобства объединены в шину, записываются значения входных переменных x_1, x_2, \dots, x_n передним фронтом сигнала, поступающего на управляющий вход «Загрузка» (в самом младшем разряде будет храниться значение переменной x_1 , затем переменной x_2 и т. д.). Эта запись происходит в момент старта устройства и в момент перехода его в начальное состояние. С выхода регистра сдвига снимается значение самого младшего разряда, хранимого в регистре слова. Передним фронтом тактового импульса C_2 , поступающего на его управляющий вход «Сдвиг», хранимая в регистре информация сдвигается вправо с потерей значения, содержащегося до этого момента в самом младшем разряде. Таким образом регистр сдвига позволяет поочередно выводить значения переменных x_1, x_2, \dots, x_n .

Регистр состояний предназначен для хранения двоичного кода текущего состояния устройства и имеет $v-1$ входов и столько же выходов, которые для удобства объединены в соответствующие шины. Запись информации в регистр выполняется передним фронтом тактового импульса C_2 при его поступлении на управляющий вход «Запись». Для очистки регистра (установки на его выходах уровней логического нуля), необходимой для установки устройства в начальное состояние, используется управляющий вход «Сброс».

Компоненты под названием «Селектор единицы», «Селектор нуля» и «Селектор начального состояния» реализуют элементарные конъюнкции ранга $v-1$ и устанавливают на своих выходах уровень логической единицы тогда, когда устройство оказывается соответственно в одном или другом финальном состоянии либо в начальном состоянии.

RS-триггер управляется сигналами, формируемыми компонентами «Селектор единицы» и «Селектор нуля», и предназначен для хранения найденного значения функции f в течение интервала времени, необходимого для подсчета обновленного значения.

Для работы устройства необходим также генератор тактовых импульсов C_1 , которые в свою очередь порождают инвертированные тактовые импульсы C_2 .

Необходимо отметить, что поскольку используется модель инициального автомата, то началу работы устройства должна предшествовать установка его в начальное состояние путем очистки регистра состояний и сброса RS-триггера в нулевое состояние, а также загрузки в регистр сдвига значений переменных x_1, x_2, \dots, x_n .

Автоматная модель устройства. Способ построения автоматной модели основан на использовании разложения Шеннона булевой функции f по переменным, входящим в множество X [5]. Представление вида

$$f(x_1, x_2, \dots, x_n) = x_i f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n) \vee \bar{x}_i f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$$

называется разложением функции f по переменной x_i . Функции $f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n)$ и $f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n)$ называются коэффициентами разложения функции f по переменной x_i . Они не зависят от переменной x_i и получаются путем подстановки в исходную функцию констант 1 и 0 соответственно вместо переменной x_i . В свою очередь, каждый из полученных коэффициентов разложения можно далее разложить по другой переменной, например x_j , получая в результате четыре коэффициента разложения функции f по переменным x_i и x_j . Продолжая процесс разложения по оставшимся входным переменным и учитывая, что некоторые из получаемых при этом коэффициентов разложения вырождаются в константы 1 и 0 или оказываются дубликатами, можно построить ориентированный ациклический граф, называемый в литературе бинарной диаграммой решений. В зарубежной литературе используется название binary decision diagram, сокращенно BDD [6–9].

BDD содержит начальную, внутренние и две концевые вершины, которые распределены по уровням. Если переменные для разложения выбирать в порядке увеличения их индекса, то начальная вершина, расположенная на первом уровне, помечается переменной x_1 . Внутренние вершины, расположенные на уровнях 2, 3, ..., n , помечаются переменными x_2, x_3, \dots, x_n соответственно. Концевые вершины, расположенные на уровне $n+1$, будут помечены константами 0 и 1, соответствующими двум возможным значениям булевой функции f . Каждая из вершин, кроме концевых, связана с двумя вершинами-потомками. Переход к одной из них выполняется в зависимости от значения переменной, которой помечена вершина. Значениям 0 и 1 этой переменной соответствуют переходы, графически отображаемые пунктирной и непрерывной линиями соответственно. Тогда значение, которое принимает функция f на конкретном наборе значений своих переменных x_1, x_2, \dots, x_n , в BDD однозначно определяется путем из начальной вершины в одну из концевых вершин.

Рассмотрим пример. На рис. 2, а изображена BDD для функции

$$f(x_1, x_2, \dots, x_6) = x_1 \bar{x}_5 \vee \bar{x}_2 x_4 x_6 \vee \bar{x}_3 \vee x_3 \bar{x}_4 \bar{x}_6,$$

заданной в дизъюнктивной нормальной форме.

Размер BDD, определяемый числом ее вершин, зависит не только от исходной булевой функции, но и от выбора порядка входных переменных, используемых при построении BDD. В общем случае задача выбора порядка входных переменных, приводящая к построению BDD наименьшего размера, сопряжена с громоздкими вычислениями, резко возрастающими с ростом числа аргументов функции f . Имеются работы (например, [10–12]), посвященные решению этой задачи. В настоящей статье ограничимся рассмотрением удобного на практике приближенного варианта решения данной задачи, заключающегося в следующем. На очередном шаге построения BDD выбирается переменная, использование которой приводит к появлению наименьшего числа различающихся коэффициентов разложения, отличных от нуля и единицы. Если таких вариантов несколько либо их вообще нет, то выбирается вариант, у которого больше коэффициентов разложения, вырожденных в одиночную переменную (возможно, с инверсией). Если и здесь вариантов несколько либо они отсутствуют, то выбирается вариант с наименьшим суммарным числом литералов, входящих в элементарные конъюнкции дизъюнктивной нормальной формы всех коэффициентов разложения. Применение такого способа выбора переменных при построении BDD часто позволяет значительно уменьшить ее размер (рис. 2, *b*).

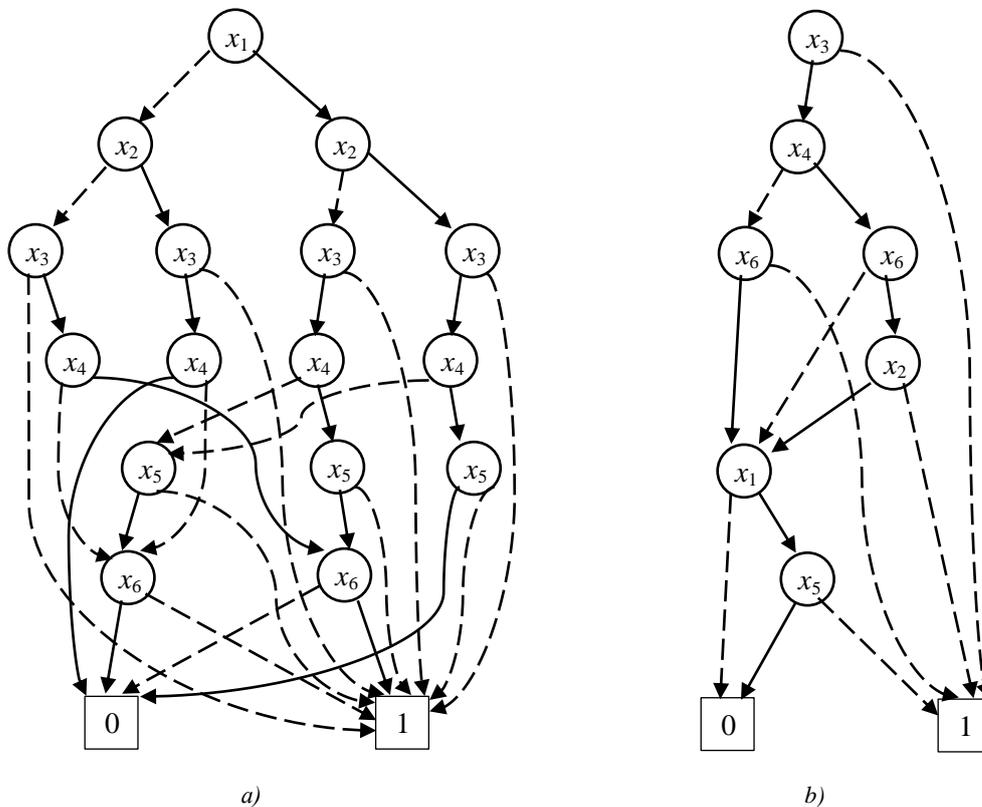


Рис. 2. BDD: *a*) порядок разложения $x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_6$; *b*) порядок разложения $x_3 \rightarrow x_4 \rightarrow x_6 \rightarrow x_2 \rightarrow x_1 \rightarrow x_5$
 Fig. 2. BDD: *a*) decomposition order: $x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_6$; *b*) decomposition order $x_3 \rightarrow x_4 \rightarrow x_6 \rightarrow x_2 \rightarrow x_1 \rightarrow x_5$

После того как для булевой функции f построена BDD, ее можно достаточно просто отобразить в граф поведения инициального автомата Мура с абстрактным состоянием, который и будет служить функциональной моделью устройства рассматриваемого здесь типа. Автомат может находиться в различных внутренних состояниях, образующих множество $Q = \{q_1, q_2, \dots, q_m\}$, где m – общее число вершин в BDD. Начальным состоянием служит состояние q_1 . Финальные состояния, соответствующие конечным вершинам BDD, дополняются дугами для безусловного перехода в начальное состояние, поскольку в отличие от BDD значение функции f будет вычисляться многократно. Например, BDD, изображенной на рис. 2, *b*, соответствует граф поведения автомата Мура с абстрактным состоянием, который показан на рис. 3. Финальные состояния изображены в форме эллипсов.

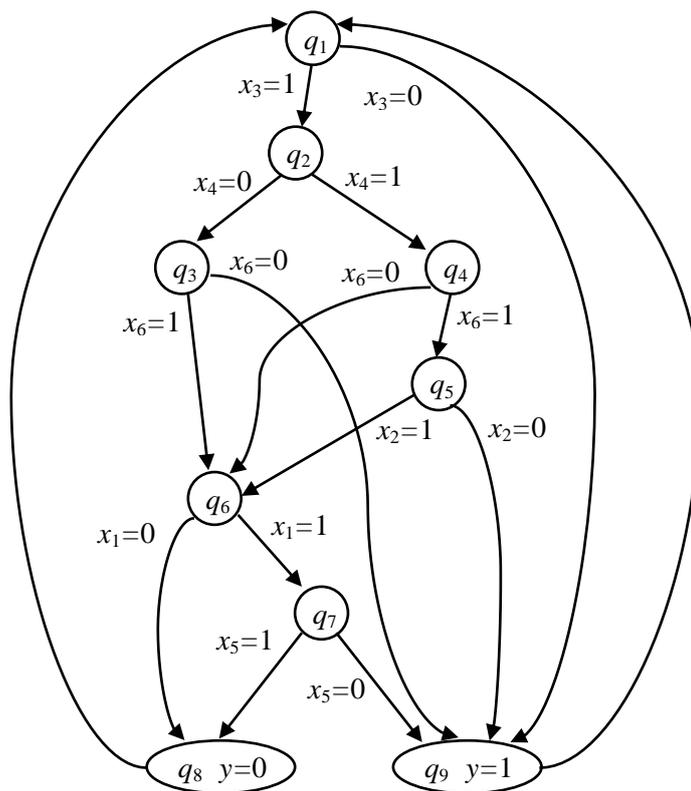


Рис. 3. Граф поведения автомата Мура с абстрактным состоянием
Fig. 3. A graph of the behavior of a Moore automaton with an abstract state

Автомат Мура с абстрактным состоянием можно задать также с помощью таблицы переходов (табл. 1), каждая строка которой определяет состояние q^+ , в которое переходит автомат из состояния q , если на его вход подано значение соответствующей входной переменной. Для автомата Мура таблица переходов дополняется столбцом, задающим значения функции f . Особенностью этой таблицы является то, что символ «-», используемый обычно при задании интервалов булева пространства входных переменных, означает еще и то, что значение соответствующей переменной не принимается во внимание, поскольку не оказывает никакого влияния на функционирование автомата. В частности, интервал - - - - - указывает, что из состояний q_8 и q_9 переход выполняется всегда в состояние q_1 независимо от значений входных переменных. Символ «*», используемый при задании значения функции f , означает, что переход в данное состояние не меняет значение функции, т. е. она сохранит то же значение, которое было и в предыдущем состоянии.

Чтобы перейти к технической реализации устройства рассматриваемого типа, необходимо сначала выполнить кодирование состояний автомата путем замены многозначной внутренней переменной q на булев вектор $\mathbf{z} = (z_1, z_2, \dots, z_w)$. В этом случае автомат с абстрактным состоянием будет преобразован в булев автомат [5, 13]. Синхронная реализация автомата на основе ПЗУ не требует решения задачи противогоночного кодирования и выполнения операций, связанных с упрощением булевых выражений, так как какая-либо иная управляющая логика отсутствует. Поэтому задача кодирования состояний сводится к присваиванию состояниям автомата уникальных w -разрядных двоичных кодов. При этом начальное состояние необходимо всегда кодировать вектором, состоящим из одних нулей, поскольку автомат является инициальным. Требуемое число разрядов $w = \lceil \log_2 m \rceil$, где $\lceil a \rceil$ означает минимальное целое число, не меньшее a . Для простоты будем использовать следующее правило: состояние q_i , $i \in \{1, 2, \dots, m\}$,

кодируется двоичным кодом, имеющим десятичное значение $i-1$. Таким образом, состояния q_1, q_2, \dots, q_m будут закодированы двоичными числами с десятичными эквивалентами $0, 1, \dots, m-1$ соответственно. В рассматриваемом примере $w = 4$. С учетом этого получаем таблицу переходов булева автомата Мура (табл. 2).

Таблица 1
Таблица переходов автомата Мура с абстрактным состоянием

Table 1
The transition table of the Moore automaton with an abstract state

x	q	q ⁺	y	x	q	q ⁺	y
- - 0 - - -	q ₁	q ₉	*	- 0 - - - -	q ₅	q ₉	*
- - 1 - - -	q ₁	q ₂	*	- 1 - - - -	q ₅	q ₆	*
- - - 0 - -	q ₂	q ₃	*	0 - - - - -	q ₆	q ₈	*
- - - 1 - -	q ₂	q ₄	*	1 - - - - -	q ₆	q ₇	*
- - - - 0	q ₃	q ₉	*	- - - - 0 -	q ₇	q ₉	*
- - - - 1	q ₃	q ₆	*	- - - - 1 -	q ₇	q ₈	*
- - - - 0	q ₄	q ₆	*	- - - - - -	q ₈	q ₁	0
- - - - 1	q ₄	q ₅	*	- - - - - -	q ₉	q ₁	1

Таблица 2
Таблица переходов булева автомата Мура

Table 2
Transition table of the Boolean Moore automaton

x	z	z ⁺	y	x	z	z ⁺	y
- - 0 - - -	0 0 0 0	1 0 0 0	*	- 0 - - - -	0 1 0 0	1 0 0 0	*
- - 1 - - -	0 0 0 0	0 0 0 1	*	- 1 - - - -	0 1 0 0	0 1 0 1	*
- - - 0 - -	0 0 0 1	0 0 1 0	*	0 - - - - -	0 1 0 1	0 1 1 1	*
- - - 1 - -	0 0 0 1	0 0 1 1	*	1 - - - - -	0 1 0 1	0 1 1 0	*
- - - - 0	0 0 1 0	1 0 0 0	*	- - - - 0 -	0 1 1 0	1 0 0 0	*
- - - - 1	0 0 1 0	0 1 0 1	*	- - - - 1 -	0 1 1 0	0 1 1 1	*
- - - - 0	0 0 1 1	0 1 0 1	*	- - - - - -	0 1 1 1	0 0 0 0	0
- - - - 1	0 0 1 1	0 1 0 0	*	- - - - - -	1 0 0 0	0 0 0 0	1

Прошивка памяти ПЗУ. Обозначим входы и выходы ПЗУ через A_0, A_1, \dots, A_{v-1} и Q_1, Q_2, \dots, Q_w соответственно. При подаче на входы ПЗУ v -разрядного кода в матричной памяти посредством дешифратора выбирается соответствующее w -разрядное слово, которое передается на выходы ПЗУ.

Если для адресации используются входы A_1, A_2, \dots, A_{v-1} , то в матричной памяти будет выбрано одно из двух слов, адреса которых различаются лишь значением самого младшего разряда, связанного с адресным входом A_0 . Значение сигнала, поступающего на этот вход, определит, какое из двух слов будет выбрано. Адресный вход A_0 связан с выходом сдвигового регистра, в который предварительно записываются значения входных переменных из множества X в порядке, обеспечивающем наименьший размер BDD. Адресные входы A_1, A_2, \dots, A_{v-1} связаны с соответствующими выходами регистра состояний, хранящего код текущего внутреннего состояния устройства. С учетом этого можно сделать вывод, что прошивка памяти ПЗУ должна выполняться согласно таблице переходов булева автомата. Тогда каждое из двух слов, соответствующих конкретному внутреннему состоянию, будет хранить код следующего состояния, т. е. состояния, в которое перейдет устройство в зависимости от текущего значения сигнала на входе A_0 . На рис. 4 показана прошивка памяти ПЗУ, обеспечивающая техническую реализацию булева автомата, заданного табл. 2.

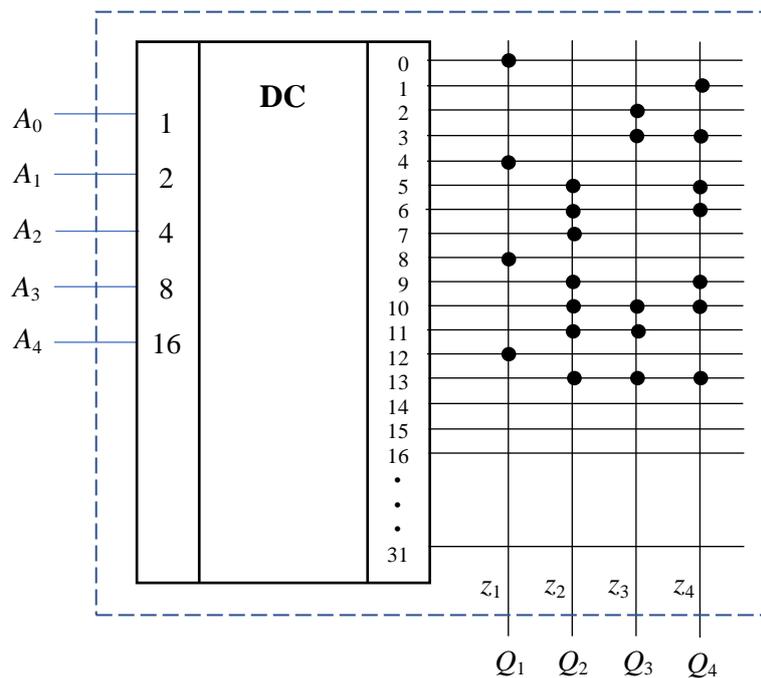


Рис. 4. Прошивка памяти ПЗУ
Fig. 4. ROM Memory Firmware

Заключение. Устройства, реализованные на основе автоматной модели (как и проектируемые на базе микроконтроллеров), могут найти применение в самых разнообразных изделиях, прежде всего предметах бытового назначения, не требующих обеспечения высоких показателей по быстродействию и времени реакции. В отношении этих характеристик данные устройства значительно уступают комбинационным схемам, синтезированным из традиционных логических элементов И, ИЛИ, НЕ, И-НЕ, ИЛИ-НЕ и др., например из программируемых логических матриц [5, 13] или LUT-компонентов [14], представляющих собой разновидность программируемых логических интегральных схем комбинационного типа. С другой стороны, преимуществом первых является возможность изменения их функциональности при сохранении той же самой схемной реализации. Для этого в устройстве необходимо использовать не однократно, а многократно программируемое ПЗУ, которое путем перепрошивки своей матричной памяти позволит реализовать любую другую булеву функцию при следующих условиях:

- 1) число ее аргументов не будет превосходить разрядность регистра сдвига;
- 2) размер BDD, оптимизированной по числу вершин, не будет превышать 2^n ;
- 3) финальные состояния булева автомата сохраняют свои прежние кодировки (чтобы избежать внесения изменений в компоненты «Селектор единицы» и «Селектор нуля»).

Первое условие можно не учитывать, если изначально использовать сдвиговый регистр большей разрядности, резервируя при этом старшие разряды для возможных изменений функциональности устройства в будущем.

Рассматриваемое устройство в некоторых случаях можно упростить, удалив регистр состояний и соединив выход ПЗУ с его входом, а также с компонентами «Селектор единицы», «Селектор нуля» и «Селектор начального состояния». В этом случае дискретный автомат, моделирующий работу устройства, перестает быть инициальным. В свою очередь это может привести к тому, что самый первый переход устройства в одно из финальных состояний может быть ошибочным, так как устройство начнет работу из какого-то состояния, которое невозможно заранее предсказать. Дальнейшее же функционирование устройства будет правильным. Поэтому, если такой «начальный сбой» допустим, указанное упрощение устройства можно выполнять.

Устройства, реализованные на базе микроконтроллеров, по сравнению с описываемым в статье устройством оказываются менее быстродействующими. Особенно это касается тех из них, которые программируются не на ассемблере, а на языке высокого уровня, так как после компиляции образуются довольно длинные последовательности машинных команд процессора. С другой стороны, использование микроконтроллеров позволяет реализовать значительно более сложные алгоритмы управления, чем те, которые описываются булевыми функциями.

Следует отметить, что предложенный способ реализации булевых функций опирается на классическую автоматную модель, хотя тот же конечный результат можно получить, основываясь на моделях микропрограммного или секвенциального автоматов [5, 13].

Список использованных источников

1. Белов, А. В. Программирование микроконтроллеров для начинающих и не только... / А. В. Белов. – СПб. : Наука и техника, 2016. – 352 с.
2. Кузьмин, В. А. Реализация функций алгебры логики автоматами, нормальными алгоритмами и машинами Тьюринга / В. А. Кузьмин // Проблемы кибернетики. – 1965. – Вып. 13. – С. 75–96.
3. Хопкрофт, Д. Введение в теорию автоматов, языков и вычислений / Д. Хопкрофт, Р. Мотвани, Дж. Ульман. – 2-е изд., пер. с англ. – М. : Изд. дом «Вильямс», 2008. – 528 с.
4. Шальто, А. А. Логическое управление. Методы аппаратной и программной реализации алгоритмов / А. А. Шальто. – СПб. : Наука, 2000. – 780 с.
5. Закревский, А. Д. Логические основы проектирования дискретных устройств / А. Д. Закревский, Ю. В. Поттосин, Л. Д. Черемисинова. – М. : Физматлит, 2007. – 592 с.
6. Andersen, H. R. An Introduction to Binary Decision Diagrams. Lecture Notes / H. R. Andersen. – Copenhagen : IT University of Copenhagen, 1999. – 35 p.
7. Akers, S. B. Binary decision diagrams / S. B. Akers // IEEE Transactions on Computers. – 1978. – Vol. C-27, no. 6. – P. 509–516.
8. 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.
9. Yang, S. BDS: a BDD-based logic optimization system / S. Yang, M. Ciesielski // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. – 2002. – Vol. 21, no. 7. – P. 866–876.
10. Бибило, П. Н. Применение диаграмм двойного выбора при синтезе логических схем / П. Н. Бибило. – Минск : Беларус. навука, 2014. – 231 с.
11. Meinel, C. Algorithms and Data Structures in VLSI Design: OBDD – Foundations and Applications / C. Meinel, T. Theobald. – Berlin, Heidelberg : Springer-Verlag, 1998. – 267 p.
12. Ebdndt, R. Advanced BDD Optimization / R. Ebdndt, G. Fey, R. Drechsler. – Springer, 2005. – 222 p.
13. Закревский, А. Д. Логический синтез каскадных схем / А. Д. Закревский. – М. : Наука, 1981. – 416 с.
14. Vemuri, N. BDD-based logic synthesis for LUT-6-based FPGAs / N. Vemuri, P. Kalla, R. Tessier // ACM Transactions on Design Automation of Electronic Systems. – 2002. – Vol. 7, no. 4. – P. 501–525.

References

1. Belov A. V. Programmirovaniye mikrokontrollov dlya nachinayushchih i ne tol'ko... *Microcontroller Programming for Beginners and Beyond...* Saint-Petersburg, Nauka i tekhnika, 2016, 352 p. (In Russ.).
2. Kuz'min V. A. *Realization of functions of the algebra of logic by automata, normal algorithms and Turing machines.* Problemy kibernetiki [Cybernetics Issues], 1965, vol. 13, pp. 75–96 (In Russ.).
3. Hopcroft J. E., Motwani R., Ullman J. D. *Introduction to Automata, Theory, Languages and Computation*, 2nd edition. Addison-Wesley, 2000, 521 p.
4. Shalyto A. A. Logicheskoye upravleniye. Metody apparatnoy i programmnoy realizacii algoritmov. *Logical Control. Methods of Hardware and Software Implementation of Algorithms.* Saint-Petersburg, Nauka, 2000, 780 p. (In Russ.).
5. Zakrevskij A. D., Pottosin Yu. V., Cheremisinova L. D. Logicheskie osnovy proektirovaniya diskretnyh ustrojstv. *Logical Fundamentals of Discrete Devices Design.* Moscow, Fizmatlit, 2007, 592 p. (In Russ.).
6. Andersen H. R. *An Introduction to Binary Decision Diagrams. Lecture Notes.* Copenhagen, IT University of Copenhagen, 1999, 35 p.

7. Akers S. B. Binary decision diagrams. *IEEE Transactions on Computers*, 1978, vol. C-27, no. 6, pp. 509–516.
8. Bryant R. E., Meinel C. Ordered binary decision diagrams. *Logic Synthesis and Verification*. In S. Hassoun, T. Sasao, R. K. Brayton (eds.). Kluwer Academic Publishers, 2002, pp. 285–307.
9. Yang S., Ciesielski M. BDS: a BDD-based logic optimization system. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2002, vol. 21, no. 7, pp. 866–876.
10. 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.).
11. Meinel C., Theobald T. *Algorithms and Data Structures in VLSI Design: OBDD – Foundations and Applications*. Berlin, Heidelberg, Springer-Verlag, 1998, 267 p.
12. Ebendt R., Fey G., Drechsler R. *Advanced BDD Optimization*. Springer, 2005, 222 p.
13. Zakrevskij A. D. Logicheskij sintez kaskadnyh skhem. *Logical Synthesis of Cascading Circuit*. Moscow, Nauka, 1981, 416 p. (In Russ.).
14. Vemuri N., Kalla P., Tessier R. BDD-based logic synthesis for LUT-6-based FPGAs. *ACM Transactions on Design Automation of Electronic Systems*, 2002, vol. 7, no. 4, pp. 501–525.

Информация об авторе

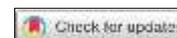
Бутов Алексей Александрович, кандидат технических наук, доцент, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: tmkrb9@gmail.com

Information about the author

Alexey A. Butov, Ph. D. (Eng.), Assoc. Prof., Belarusian state university of informatics and radioelectronics.
E-mail: tmkrb9@gmail.com

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

INFORMATION TECHNOLOGIES



УДК 004.934.2 + 004.85
<https://doi.org/10.37661/1816-0301-2023-20-1-102-112>

Оригинальная статья
Original Paper

Малоразмерные спектральные признаки для машинного обучения в задачах анализа и классификации голосового сигнала

Д. С. Лихачёв[✉], М. И. Вашкевич, Н. А. Петровский, И. С. Азаров

Белорусский государственный университет
информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
[✉]E-mail: likhachov@bsuir.by

Аннотация

Цели. Решается задача разработки метода вычисления малоразмерных спектральных признаков, повышающего эффективность существующих систем машинного обучения для анализа и классификации голосовых сигналов.

Методы. Спектральные признаки извлекаются с помощью генеративного подхода, который предполагает вычисление дискретного спектра Фурье последовательности отчетов, сгенерированной с использованием авторегрессионной модели входного голосового сигнала. Сгенерированная последовательность, обрабатываемая дискретным преобразованием Фурье, учитывает периодичность преобразования, позволяя тем самым повысить точность спектральной оценки анализируемого сигнала.

Результаты. Предложен и описан генеративный метод вычисления спектральных признаков, предназначенных для применения в системах машинного обучения при анализе и классификации голосовых сигналов. Проведен экспериментальный анализ точности и стабильности представления спектра тестового сигнала с известным спектральным составом с использованием огибающих. Огибающие вычислялись с помощью предложенного генеративного метода и дискретного преобразования Фурье с различными окнами анализа (прямоугольным окном и окном Ханна). Проведенный анализ показал, что генеративный метод получения спектральных огибающих позволил добиться более точного представления спектра тестового сигнала по критерию минимума квадратичной ошибки. Проведено сравнение эффективности классификации голосового сигнала при использовании предложенных признаков и признаков на основе мел-частотных кепстральных коэффициентов. В качестве базовой тестовой системы для оценки эффективности предлагаемого подхода на практике использовалась система диагностики бокового амиотрофического склероза по голосу.

Заключение. Результаты экспериментов показали ощутимое повышение точности классификации при использовании предлагаемых признаков по сравнению с признаками на основе мел-частотных кепстральных коэффициентов.

Ключевые слова: анализ голоса, генеративный метод, авторегрессия, машинное обучение, спектральные признаки, классификация

Для цитирования. Малоразмерные спектральные признаки для машинного обучения в задачах анализа и классификации голосового сигнала / Д. С. Лихачёв [и др.] // Информатика. – 2023. – Т. 20, № 1. – С. 102–112. <https://doi.org/10.37661/1816-0301-2023-20-1-102-112>

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 07.02.2023

Подписана в печать | Accepted 22.02.2023

Опубликована | Published 29.03.2023

Small-size spectral features for machine learning in voice signal analysis and classification tasks

Denis S. Likhachov[✉], Maxim I. Vashkevich, Nick A. Petrovsky, Elias S. Azarov

*Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus*

[✉]E-mail: likhachov@bsuir.by

Abstract

Objectives. The problem of developing a method for calculating small-sized spectral features that increases the efficiency of existing machine learning systems for analyzing and classifying voice signals is being solved.

Methods. Spectral features are extracted using a generative approach, which involves calculating a discrete Fourier spectrum for a sequence of samples generated using an autoregressive model of input voice signal. The generated sequence processed by the discrete Fourier transform considers the periodicity of the transform and thereby increase the accuracy of spectral estimation of analyzed signal.

Results. A generative method for calculating spectral features intended for use in machine learning systems for the analysis and classification of voice signals is proposed and described. An experimental analysis of the accuracy and stability of the spectrum representation of a test signal with a known spectral composition has been carried out using the envelopes. The envelopes were calculated using proposed generative method and using discrete Fourier transform with different analysis windows (rectangular window and Hanna window). The analysis showed that spectral envelopes obtained using the proposed method more accurately represent the spectrum of test signal according to the criterion of minimum square error. A comparison of the effectiveness of voice signal classification with proposed features and the features based on the mel-frequency kepsral coefficients is carried out. A diagnostic system for amyotrophic lateral sclerosis was used as a basic test system to evaluate the effectiveness of proposed approach in practice.

Conclusion. The obtained experimental results showed a significant increase of classification accuracy when using proposed approach for calculating features compared with the features based on the mel-frequency kepsral coefficients.

Keywords: voice analysis, generative method, autoregression, machine learning, spectral features, classification

For citation. Likhachov D. S., Vashkevich M. I., Petrovsky N. A., Azarov E. S. *Small-size spectral features for machine learning in voice signal analysis and classification tasks*. Informatika [Informatics], 2023, vol. 20, no. 1, pp. 102–112 (In Russ.). <https://doi.org/10.37661/1816-0301-2023-20-1-102-112>

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

Введение. В настоящее время большое количество систем для обработки речевых и голосовых сигналов с использованием машинного обучения оперируют признаками в частотной области [1–4]. Во многих случаях извлечение признаков в таких системах сводится к получению в том или ином виде спектральной оценки анализируемого сигнала. Основным требованием к полученной оценке является как можно более высокое спектральное разрешение при удовле-

творительной точности оценки для целевой задачи. В большинстве применяемых в настоящее время методов вычисления признаков для машинного обучения в частотной области используется дискретное преобразование Фурье (ДПФ). Это преобразование хорошо подходит для частотно-временного анализа периодических или частично периодических сигналов, в том числе речевых и голосовых [4]. Спектральная оценка при помощи ДПФ является точной при соблюдении двух условий: анализируемый сигнал должен быть строго периодическим и стационарным на протяжении всего интервала анализа; период анализируемого сигнала должен точно совпадать с размером ДПФ. Для речевых и аудиосигналов два вышеприведенных требования в строгом виде не выполняются. Это обуславливает необходимость применения временного окна для обработки входных данных. Обработка окном хотя и позволяет ослабить влияние боковых спектральных лепестков, но в то же время приводит к значительным искажениям полученной спектральной оценки по отношению к истинной [5].

В настоящей работе для получения спектральных признаков предлагается подход, использующий генеративный метод спектрального оценивания на базе авторегрессионной модели анализируемого сигнала. Как будет показано далее, применение данного метода позволяет отказаться от использования окна при выполнении спектрального оценивания и в конечном итоге существенно улучшить эффективность работы системы обработки речевого сигнала на основе машинного обучения для диагностики бокового амиотрофического склероза (БАС) [6].

Генеративный метод вычисления дискретной оценки огибающей спектра авторегрессионного процесса. Основная идея предлагаемого метода получения дискретной оценки огибающей спектра авторегрессионного процесса (ОСАП) состоит в том, чтобы использовать ДПФ без применения оконной функции для обработки входных данных, обеспечив при этом согласованность периода анализируемого сигнала и размера ДПФ. С этой целью в качестве входных данных для ДПФ предлагается использовать специальным образом сгенерированную последовательность, которая имела бы равный с размером ДПФ период и соответствующий анализируемому сигналу спектральный состав.

Схема вычисления дискретной оценки ОСАП в общем виде изображена на рис. 1.

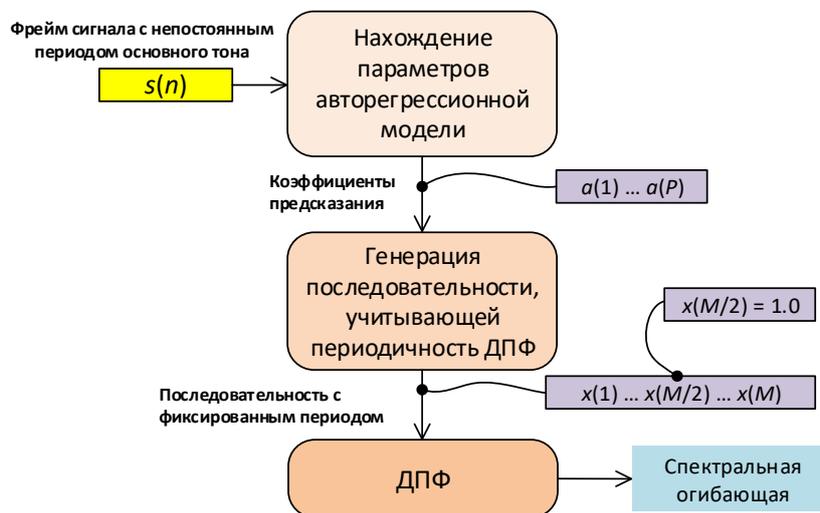


Рис. 1. Схема вычисления спектральной огибающей

Fig. 1. Scheme for calculating the spectral envelope

Видно, что процесс вычисления дискретной оценки ОСАП включает несколько основных этапов:

1. По входному кадру анализируемого сигнала с использованием автокорреляционного метода определяются параметры авторегрессионной модели P -го порядка: $a_k, k = \overline{1, P}$ [7].

2. С помощью авторегрессионной модели анализируемого процесса генерируется последовательность с периодом, равным размеру ДПФ. Размер ДПФ зависит от целевой размерности вектора признаков и покрываемого им частотного диапазона.

Предположим, необходимо сгенерировать требуемую входную последовательность $x(m)$ для ДПФ размером M . Искомая последовательность не должна быть нулевой, т. е. должно выполняться условие

$$\sum_{m=1}^M |x(m)| \neq 0. \quad (1)$$

Зная параметры авторегрессионной модели анализируемого процесса, можно записать следующее равенство [7]:

$$x(m) = -\sum_{k=1}^P a_k \cdot x(m-k), \quad m = \overline{1, M+P}. \quad (2)$$

Положив $x(m) = 0$ при $m = \overline{M+1, M+P}$ и $m = \overline{-P+1, 0}$, выражение (2) для $m = \overline{1, M}$ можно записать в виде равенства

$$\begin{pmatrix} -1 & 0 & 0 & \dots & 0 & 0 \\ -a_1 & -1 & 0 & \dots & 0 & 0 \\ -a_2 & a_1 & -1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & -a_p \end{pmatrix} \cdot \begin{pmatrix} x(1) \\ x(2) \\ x(3) \\ \vdots \\ x(M) \end{pmatrix} = 0. \quad (3)$$

Введя дополнительное ограничение $x(M/2) = 1$ для выполнения условия (1) и решив полученную из (3) систему линейных уравнений относительно x , можно найти искомую последовательность $x(m)$, $m = \overline{1, M}$, которая будет затухать к краям кадра аналогично оконной функции.

3. С помощью ДПФ вычисляется дискретный спектр одного периода сгенерированной на этапе 2 последовательности:

$$S_{est}(k) = \left| \sum_{m=0}^{M-1} x(m) e^{-i \frac{2\pi}{M} km} \right|, \quad k = \overline{1, \frac{M}{2}}. \quad (4)$$

Полученный дискретный спектр $S_{est}(k)$ является искомой дискретной оценкой ОСАП.

Анализ эффективности предлагаемого метода вычисления дискретной оценки ОСАП. С целью анализа эффективности предлагаемого метода проведено сравнение точности и стабильности представления спектра тестового сигнала с помощью спектральных огибающих, полученных тремя способами: предложенным методом, ДПФ без применения окна, ДПФ с применением окна Ханна.

В качестве тестового сигнала использовался полигармонический речеподобный сигнал с заданными параметрами (рис. 2). В ходе эксперимента генерировались сегменты тестового сигнала с постоянной частотой основного тона и длительностью до 200 мс. Каждый сегмент тестового сигнала анализировался кадрами длительностью от 1,5 до 20 мс с шагом 0,068 мс. Частота основного тона варьировалась от 85 до 240 Гц с шагом 1 Гц. Для корректного сопоставления

частотной сетки спектра тестового сигнала и полученных огибающих использовалась линейная интерполяция. Все огибающие при сравнении нормировались по энергии. Пример спектральных огибающих, вычисленных для одного кадра тестового сигнала, приведен на рис. 2.

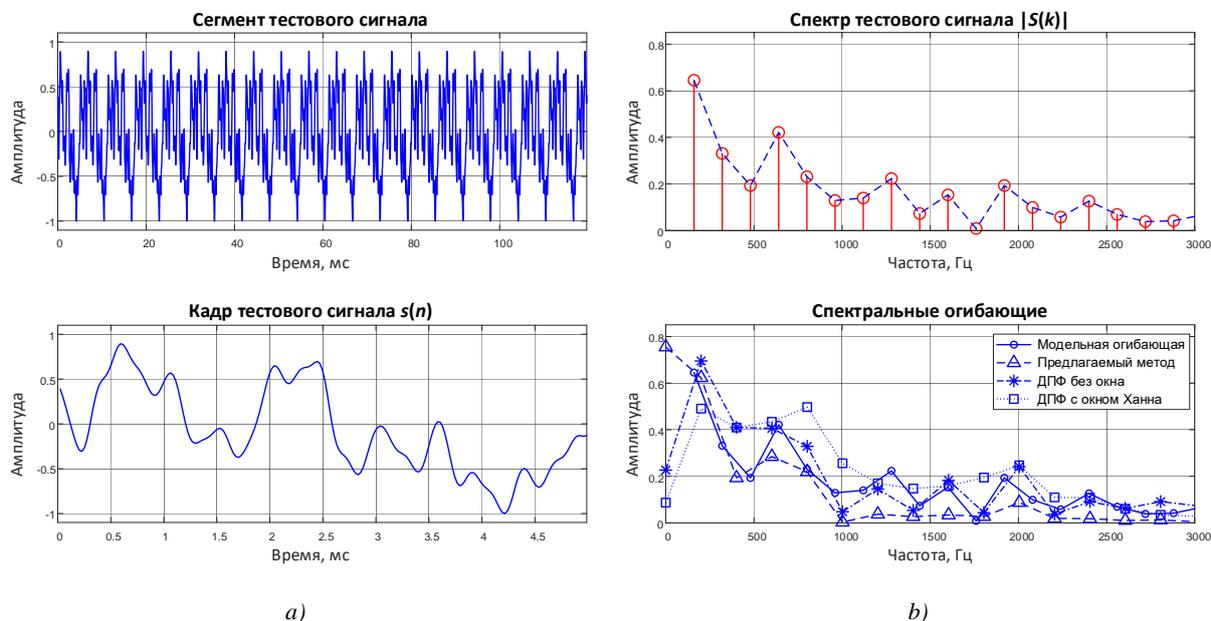


Рис. 2. Тестовый сигнал (a) и соответствующие спектральные огибающие для одного кадра (b)

Fig. 2. Test signal (a) and corresponding spectral envelopes for one frame (b)

В качестве критериев точности и стабильности представления спектра тестового сигнала с помощью полученных огибающих использовались относительные квадратичные ошибки (%), вычисляемые согласно следующим выражениям:

$$SE_{acc}^{i,j} = \frac{\sum_{k=1}^K (S_{mod}^i(k) - S_{est}^{i,j}(k))^2}{\sum_{k=1}^K (S_{mod}^i(k))^2} \cdot 100, \quad SE_{stab}^i = \frac{\sum_{k=1}^K (S_{est}^{i,1}(k) - S_{est}^{i-1,F}(k))^2}{\sum_{k=1}^K (S_{est}^{i-1,F}(k))^2} \cdot 100, \quad (5)$$

где $SE_{acc}^{i,j}$ – относительная квадратичная ошибка для j -го кадра сигнала из i -го сегмента; SE_{stab}^i – относительная квадратичная ошибка для i -го сегмента; $S_{mod}^i(k)$ – k -й частотный отсчет спектра тестового сигнала для i -го сегмента; $S_{est}^{i,j}(k)$ – k -й отсчет полученной спектральной огибающей для j -го кадра сигнала из i -го сегмента; $S_{est}^{i,1}(k)$ и $S_{est}^{i-1,F}(k)$ – k -й частотный отсчет спектральной огибающей для первого кадра из i -го сегмента и последнего кадра из $i-1$ -го сегмента; K – количество частотных отсчетов, участвующих в оценке; F – количество кадров сигнала в сегменте.

Для получения итогового показателя точности $\overline{SE_{acc}}$ вычисленные по выражению (5) ошибки $SE_{acc}^{i,j}$ усреднялись по всем сегментам и кадрам. Итоговое значение показателя стабильности находилось путем усреднения ошибок SE_{stab}^i по всем сегментам, за исключением первого. Полученные результаты представлены в табл. 1 и 2, жирным шрифтом выделены лучшие показатели.

Таблица 1

Средняя квадратичная ошибка \overline{SE}_{acc} для кадра анализа с разной длительностью, %

Table 1

Mean square error \overline{SE}_{acc} for an analysis frame with different duration, %

Метод получения спектральной огибающей <i>Method for obtaining the spectral envelope</i>	Длительность кадра анализа, мс <i>Analysis frame duration, ms</i>				
	1,5	3,2	5,0	10	20
Предложенный метод <i>Suggested method</i>	24,8	22,0	14,3	12,2	16,2
ДПФ без применения окна <i>DFT without windowing</i>	46,2	33,1	28,1	29,3	42,6
ДПФ с применением окна Ханна <i>DFT using Hann window</i>	39,4	29,1	19,4	19,8	36,6

Таблица 2

Средняя квадратичная ошибка \overline{SE}_{stab} для кадра анализа с разной длительностью, %

Table 2

Mean square error \overline{SE}_{stab} for an analysis frame with different duration, %

Метод получения спектральной огибающей <i>Method for obtaining the spectral envelope</i>	Длительность кадра анализа, мс <i>Analysis frame duration, ms</i>				
	1,5	3,2	5,0	10	20
Предложенный метод <i>Suggested method</i>	22,6	23,4	19,5	2,4	0,7
ДПФ без применения окна <i>DFT without windowing</i>	28,9	27,3	19,71	8,8	6,6
ДПФ с применением окна Ханна <i>DFT using Hann window</i>	28,8	30,8	27,41	9,0	2,2

Как показали результаты экспериментов, спектральные огибающие, полученные с помощью предложенного метода, более точно по сравнению с остальными методами представляют спектр тестового сигнала с точки зрения минимальной квадратичной ошибки.

Использование малоразмерных спектральных признаков на основе дискретной оценки ОСАП. Для проверки предлагаемого подхода в качестве тестовой взята система анализа и классификации голосового сигнала с использованием мел-частотных кепстральных коэффициентов (МЧКК), описанная в работе [6] (рис. 3). Основная задача данной системы заключается в выполнении бинарной классификации. В качестве входных данных используется голосовой сигнал, который представляет собой протяжный звук /а/. По результатам классификации входной образец голосового сигнала относят к норме или патологии. В соответствии со схемой на рис. 3 на этапе предобработки выполняется фильтрация сигнала. Далее происходит выделение признаков и рассчитываются надсегментные спектральные (кепстральные) признаки. Полученные из набора надсегментных спектральных признаков векторы используются в классификаторе для отнесения исходного голосового образца к одному из классов. Для решения задачи классификации применялся метод опорных векторов (SVM, Support Vector Machine) [8].

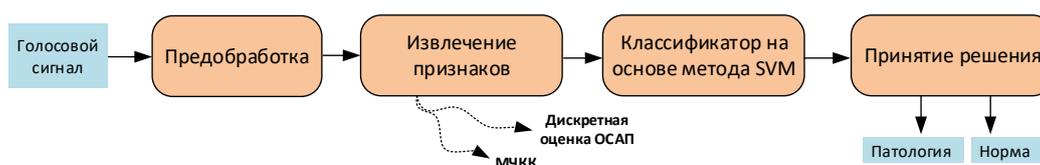


Рис. 3. Тестовая система анализа и классификации голосового сигнала

Fig. 3. Voice signal analysis and classification test system

Выделение и формирование признаков показаны на рис. 4. Исходный голосовой сигнал обрабатывался сегментами длительностью до 1 с. Для каждого такого сегмента проводился кратковременный анализ с вычислением спектральных признаков, которые достаточно точно описывают локальную структуру спектра сигнала на временном промежутке от 10 до 40 мс. Однако в результате такого кратковременного анализа формируется чрезмерно большой набор признаков для эффективной классификации с помощью машинного обучения [8]. С целью уменьшения объема данных, используемых для работы классификатора, полученные в результате кратковременного анализа признаки усреднялись и уже на их основе формировался надсегментный вектор признаков.

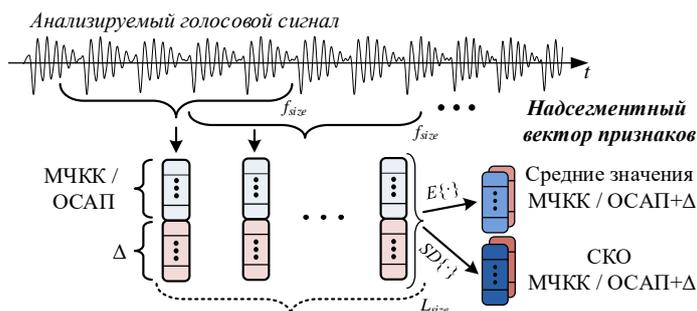


Рис. 4. Принципы формирования надсегментных признаков

Fig. 4. Principles of formation of suprasegmental features

Для формирования надсегментных признаков из наборов кратковременных спектральных признаков по каждому набору рассчитывались их средние значения и СКО.

Кратковременные спектральные признаки использовались в двух вариантах:

1. Малоразмерные спектральные признаки на основе дискретной оценки ОСАП, вычисляемые в соответствии с выражением (4).

2. Малоразмерные спектральные признаки на основе МЧКК, вычисляемые с использованием метода, описанного в работе [3]. Данный метод базируется на нахождении энергетического спектра сигнала с помощью ДПФ, который переводится в мел-шкалу путем взвешивания на треугольные окна с равной шириной в шкале мелов. Для определения МЧКК от энергетического спектра вычисляется логарифм и применяется дискретное косинусное преобразование второго типа.

На этапе проектирования классификатора с целью повышения эффективности его работы производился отбор признаков методом LASSO (Least Absolute Shrinkage and Selection Operator) [9]. В этом случае применяется регрессионный анализ, который хорошо подходит для отбора признаков, имеющих сильную корреляцию друг с другом. С помощью метода LASSO решается задача линейной регрессии со штрафной функцией, накладываемой на абсолютные значения коэффициентов модели. Задача решается итеративно для определенного числа возрастающих значений параметра регуляризации, причем веса при некоторых признаках модели приближаются к нулю или становятся равными нулю. По порядку, в котором обнуляются параметры модели (признаки), можно судить о степени влияния того или иного набора признаков на качество работы классификатора.

Для обучения и тестирования системы использовалась база, содержащая 54 образца голосовых сигналов (протяжный звук /a/), описанная в работе [10]. База содержит записи 39 здоровых человек (23 мужчин, 16 женщин) и 15 больных БАС (6 мужчин, 9 женщин).

Экспериментальные исследования эффективности предложенного формирования спектральных признаков. С помощью описанной выше системы анализа и классификации голосового сигнала (см. рис. 3) было проведено сравнение качества классификации для кратковременных спектральных признаков, полученных с применением предлагаемого метода и на основе МЧКК.

При проведении экспериментов для вычисления признаков использовались следующие значения параметров системы анализа:

- длительность сегмента около L_{size} 800 мс;
- межсегментное перекрытие f_{size} 75 %;
- количество параметров в частотной области M , которые используются для формирования вектора признаков, от 10 до 60. В случае вычисления МЧКК значение M равно числу кепстральных коэффициентов, а в случае вычисления спектральной огибающей – количеству отсчетов в частотной области в целевом диапазоне частот;
- размерность вектора надсегментных признаков равна $1 \cdot 4 \cdot (M - 1)$ и варьируется от 36 до 236. Так, например, для $M = 20$ она равна 1·76. (Кепстральные коэффициенты и отсчеты спектральной огибающей с нулевым индексом не используются для формирования вектора признаков.);
- нижняя и верхняя границы частотного диапазона, используемого для формирования признаков, от 50 до 6400 Гц;
- размер кадра анализа от 10 до 40 мс.

Для оценки производительности классификатора использовался метод перекрестной проверки LOSO (Leave-One-Subject-Out) [11]. Относительно рассматриваемой системы применение данного метода предполагало итеративное тестирование классификатора. На каждой итерации в тестовый набор включались данные, соответствующие одному диктору, а в обучающий – все оставшиеся. Количество итераций равнялось количеству дикторов. На каждой итерации определялась сбалансированная точность классификации:

$$BA = 0,5 \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right), \quad (6)$$

где TP , TN , FP , FN – количество истинно положительных, истинно отрицательных, ложноположительных и ложноотрицательных результатов классификации.

Оценки работы классификатора для каждой итерации использовались далее для вычисления среднего значения по всем итерациям \overline{BA} . Полученные результаты представлены в табл. 3 и 4, жирным шрифтом выделены лучшие показатели. Видно, что применение признаков на основе дискретной оценки ОСАП позволило добиться более высокого значения полноты классификации для всех размеров кадра анализа. Наибольшая сбалансированная точность классификации при использовании признаков на основе дискретной оценки ОСАП достигалась при количестве частотных отсчетов около 20.

Таблица 3
 Максимальная средняя сбалансированная точность классификации при использовании МЧКК признаков в зависимости от количества частотных отсчетов и длительности кадра

Table 3
 The maximum average balanced accuracy of the classification when using MFCC features, depending on the number of frequency samples and the duration of the frame

Длительность кадра анализа, мс Analysis frame duration, ms	Сбалансированная точность классификации \overline{BA} , % Balanced classification accuracy \overline{BA} , %										
	M≤10	M≤15	M≤20	M≤25	M≤30	M≤35	M≤40	M≤45	M≤50	M≤55	M≤60
10	69,3	70,5	71,7	71,7	73,2	67,2	70,3	72,4	71,9	70,7	69,7
15	70,1	70,2	70,1	70,1	69,1	71,6	71,8	73,0	73,7	69,9	70,8
20	71,3	68,3	71,3	72,5	73,9	72,6	70,8	74,4	71,2	72,1	72,1
25	72,5	68,8	72,2	72,7	75,7	69,3	70,4	72,3	69,4	72,0	70,8
30	73,0	67,5	74,3	74,7	70,9	69,7	70,2	70,1	70,3	72,2	69,7
35	74,3	67,3	69,3	75,4	71,2	69,7	75,0	69,6	72,3	77,3	69,5
40	73,9	67,1	70,9	75,0	67,9	69,3	71,4	68,8	72,5	71,3	72,8

Таблица 4

Максимальная средняя сбалансированная точность классификации при использовании предлагаемых признаков в зависимости от количества частотных отсчетов и длительности кадра

Table 4

The maximum average balanced accuracy of the classification when using proposed features, depending on the number of frequency samples and the duration of the frame

Длительность кадра анализа, мс <i>Analysis frame duration, ms</i>	Сбалансированная точность классификации \overline{BA} , % <i>Balanced classification accuracy \overline{BA}, %</i>										
	M≤10	M≤15	M≤20	M≤25	M≤30	M≤35	M≤40	M≤45	M≤50	M≤55	M≤60
10	72,5	81,0	81,6	80,4	74,2	74,6	73,1	78,5	78,2	78,5	75,7
15	69,2	79,0	80,3	79,5	76,6	74,4	74,7	76,6	78,5	77,6	78,9
20	67,4	78,6	80,5	79,4	77,9	75,2	74,8	77,9	79,7	79,4	77,7
25	67,1	77,0	81,5	77,5	77,7	75,4	73,3	75,6	78,1	78,6	77,2
30	67,0	74,4	82,7	76,2	77,4	73,9	72,7	74,1	78,6	77,5	77,1
35	66,8	74,6	81,0	76,2	78,7	74,1	72,2	73,7	77,5	77,2	76,8
40	66,2	74,5	81,5	75,7	77,5	73,6	71,7	72,9	76,5	78,3	75,6

Закключение. В работе предложены спектральные признаки на основе дискретной оценки ОСАП, которые являются альтернативой классическим мел-спектральным признакам. Результаты проведенных экспериментов для определения патологических изменений голоса свидетельствуют, что представленный метод позволяет более эффективно извлекать полезную для классификации информацию из имеющихся обучающих данных и ощутимо повышать точность классификации.

Характерной особенностью задачи определения патологических изменений голоса является крайне малый объем доступных обучающих данных, что обусловлено высокой сложностью их получения [10]. Это делает использование сложных моделей классификации, например на основе глубоких нейросетей, нецелесообразным вследствие явления переобучения [8]. Повышение точности классификации в таких условиях возможно при использовании простых, но более адекватных речевому процессу моделей, что является важной практической составляющей полученного результата. Таким образом, предложенный подход имеет хорошую перспективу применения для задач обработки речевых и голосовых сигналов, в частности, в системах машинного обучения с малым объемом обучающих данных.

Вклад авторов. Д. С. Лихачёв проработал идею использования генеративного подхода на основе авторегрессионной модели для вычисления малоразмерных признаков и провел исследование ее эффективности. М. И. Вашкевич и Н. А. Петровский приняли участие в программной реализации системы, предназначенной для постановки экспериментов. И. С. Азаров определил цели и задачи исследования. Все авторы принимали участие в подготовке текста статьи, анализе и интерпретации результатов экспериментов.

Список использованных источников

1. Towards robust voice pathology detection / P. Harar [et al.] // *Neural Computing and Applications*. – 2020. – Vol. 32, no. 20. – P. 15747–15757.
2. Benba, A. Discriminating between patients with Parkinson’s and neurological diseases using cepstral analysis / A. Benba, A. Jilbab, A. Hammouch // *IEEE Transactions on Neural Systems and Rehabilitation Engineering*. – 2016. – Vol. 24, no. 10. – P. 1100–1108.
3. Vashkevich, M. Classification of ALS patients based on acoustic analysis of sustained vowel phonations / M. Vashkevich, Y. Rushkevich // *Biomedical Signal Processing and Control*. – 2021. – Vol. 65. – P. 1–14.
4. Rabiner, L. R. *Fundamentals of Speech Recognition* / L. R. Rabiner, B. H. Juang. – Pearson Education, 1993. – 570 p.
5. Harris, F. J. On the use of windows for harmonic analysis with the discrete Fourier transform / F. J. Harris // *Proc. of the IEEE*. – Jan. 1978. – Vol. 66, no. 1. – P. 51–83.

6. Вашкевич, М. И. Система анализа и классификации голосового сигнала на основе пертурбационных параметров и кепстрального представления в психоакустических шкалах / М. И. Вашкевич, Д. С. Лихачёв, И. С. Азаров // Доклады БГУИР. – 2022. – Т. 20, № 4. – С. 73–82. <https://doi.org/10.35596/1729-7648-2022-20-1-73-82>
7. Markel, J. D. Linear Prediction of Speech / J. D. Markel, A. H. Gray. – Berlin, N. Y. : Springer-Verlag, 1976. – 290 p.
8. Flach, P. Machine Learning: The Art and Science of Algorithms That Make Sense of Data / P. Flach. – Cambridge University Press, 2012. – 416 p.
9. An Introduction to Statistical Learning with Applications in R / G. James [et al.]. – Springer, 2013. – 440 p.
10. Vashkevich, M. Bulbar ALS detection based on analysis of voice perturbation and vibrato / M. Vashkevich, A. Petrovsky, Y. Rushkevich // IEEE Intern. Conf. on Signal Processing: Algorithms, Architectures, Arrangements, and Applications, Poznan, Poland, 18–20 Sept. 2019. – Poznan, 2019. – P. 267–272.
11. The necessity of leave one subject out (LOSO) cross validation for EEG disease diagnosis / S. Kunjan [et al.] // Brain Informatics. – Springer, 2021. – P. 558–567. https://doi.org/10.1007/978-3-030-86993-9_50

References

1. Harar P., Galaz Z., Alonso-Hernandez J. B., Mekyska J., Burget R., Smekal Z. Towards robust voice pathology detection. *Neural Computing and Applications*, 2020, vol. 32, no. 20, pp. 15747–15757.
2. Benba A., Jilbab A., Hammouch A. Discriminating between patients with Parkinson's and neurological diseases using cepstral analysis. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2016, vol. 24, no. 10, pp. 1100–1108.
3. Vashkevich M., Rushkevich Y. Classification of ALS patients based on acoustic analysis of sustained vowel phonations. *Biomedical Signal Processing and Control*, 2021, vol. 65, pp. 1–14.
4. Rabiner L. R., Juang B. H. *Fundamentals of Speech Recognition*. Pearson Education, 1993, 570 p.
5. Harris F. J. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, January 1978, vol. 66, no. 1, pp. 51–83. <https://doi.org/10.1109/PROC.1978.10837>
6. Vashkevich M. I., Likhachov D. S., Azarov E. S. *Voice analysis and classification system based on perturbation parameters and cepstral presentation in psychoacoustic scales*. Doklady Belorusskogo gosudarstvennogo universiteta informatiki i radioelektroniki [Reports of the Belarusian State University of Informatics and Radioelectronics], 2022, vol. 20, no. 1, pp. 73–82 (In Russ.). <https://doi.org/10.35596/1729-7648-2022-20-1-73-82>
7. Markel J. D., Gray A. H. *Linear Prediction of Speech*. Berlin, New York, Springer-Verlag, 1976, 290 p.
8. Flach P. *Machine Learning: The Art and Science of Algorithms That Make Sense of Data*. Cambridge University Press, 2012, 416 p.
9. James G., Witten D., Hastie T., Tibshirani R. *An Introduction to Statistical Learning with Applications in R*. Springer, 2013, 440 p.
10. Vashkevich, M., Petrovsky A., Rushkevich Y. Bulbar ALS detection based on analysis of voice perturbation and vibrato. *IEEE International Conference on Signal Processing: Algorithms, Architectures, Arrangements, and Applications, Poznan, Poland, 18–20 September 2019*. Poznan, 2019, pp. 267–272.
11. Kunjan S., Grummett T. S., Pope K. J., Powers D. M. W., Fitzgibbon S. P., ..., Lewis T. W. The necessity of leave one subject out (LOSO) cross validation for EEG disease diagnosis. *Brain Informatics*, Springer, 2021, pp. 558–567. https://doi.org/10.1007/978-3-030-86993-9_50

Информация об авторах

Лихачёв Денис Сергеевич, кандидат технических наук, доцент кафедры электронных вычислительных средств, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: likhachov@bsuir.by

Вашкевич Максим Иосифович, доктор технических наук, доцент кафедры электронных вычислительных средств, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: vashkevich@bsuir.by

Information about the authors

Denis S. Likhachov, Ph. D. (Eng.), Assoc. Prof. of Computer Engineering Department, Belarusian State University of Informatics and Radioelectronics.
E-mail: likhachov@bsuir.by

Maxim I. Vashkevich, D. Sc. (Eng.), Assoc. Prof. of Computer Engineering Department, Belarusian State University of Informatics and Radioelectronics.
E-mail: vashkevich@bsuir.by

Петровский Николай Александрович, кандидат технических наук, доцент кафедры электронных вычислительных средств, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: nick.petrovsky@bsuir.by

Азаров Илья Сергеевич, доктор технических наук, доцент, заведующий кафедрой электронных вычислительных средств, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: azarov@bsuir.by

Nick A. Petrovsky, Ph. D. (Eng.), Assoc. Prof. of Computer Engineering Department, Belarusian State University of Informatics and Radioelectronics.
E-mail: nick.petrovsky@bsuir.by

Elias S. Azarov, D. Sc. (Eng.), Assoc. Prof., Head of Computer Engineering Department, Belarusian State University of Informatics and Radioelectronics.
E-mail: azarov@bsuir.by

УЧЕНЫЕ БЕЛАРУСИ SCIENTISTS OF BELARUS

**Герт Иванович Алексеев:
ученый и изобретатель**

**Gert Ivanovich Alekseev:
scientist and inventor**



Герт Иванович Алексеев родился 17 марта 1936 г. в Ульяновской области в семье военного. Его отец Иван Феропонтович воевал, был политруком, окончил войну заместителем командира полка. Мать Мария Федоровна работала учителем, а затем директором средней школы. После войны Иван Феропонтович был переведен в Харьков, где работал до 1950 г. преподавателем Харьковского пограничного училища. С 1950 по 1958 г. семья жила во Львове. Иван Феропонтович служил в штабе Прикарпатского военного округа. Мария Федоровна работала во Львовской консерватории, позднее, став кандидатом философских наук и доцентом, преподавала во Львовском политехническом институте.

В 1953 г. Герт Иванович закончил школу и поступил на механический факультет Львовского политехнического института. Получив диплом с отличием, был направлен на Минский электромеханический завод, где работал на разных должностях и вскоре познакомился со своей будущей супругой Светланой Витальевной. С 1958 по 1964 г. учился на заочном отделении радиотехнического факультета Львовского политехнического института, закончил учебу с отличием.

В 1969 г. Г. И. Алексеев был принят на работу в Институт технической кибернетики (ИТК) АН БССР и поступил в аспирантуру. Его научным руководителем стал профессор Петр Михайлович Чеголин. Будучи сотрудником лаборатории читающих автоматов под руководством П. М. Чеголина, Герт Иванович подключился к разработкам средств ввода графической информации. С 1970 г. Г. И. Алексеев начинает заниматься разработкой электромагнитных дигитайзеров, и это научное направление стало главным делом всей его жизни. В 1972 г. Герт Иванович досрочно защитил кандидатскую диссертацию «Функциональное преобразование информации на основе интерполирования».

Под руководством Герта Ивановича и при его непосредственном участии разработан ряд планшетных средств ввода графической информации, серийно освоенных промышленностью. Внедрение данных устройств способствовало созданию в СССР специализированного картографического производства. За эти разработки в 1984 г. Герт Иванович удостоился почетного звания «Заслуженный изобретатель БССР».

В 1989 г. Г. И. Алексеев избран заведующим лабораторией автоматизации ввода видеoinформации ИТК АН БССР. В лаборатории были выполнены работы по теоретическим основам электромагнитного ввода графической информации, найден ряд новых технических решений. Это позволило в трудные 1990-е гг. на скромной технологической базе разработать параметри-

ческий ряд дигитайзеров форматов с А2 по А0. Дигитайзеры использовались организациями Беларуси, России и Украины.

В 1990 г. в Институте кибернетики имени Глушкова (Киев) состоялась защита докторской диссертации Г. И. Алексеева «Теоретические основы и проектирование электромагнитных планшетных средств ввода графической информации».

С середины 1990-х гг. интерес к использованию дигитайзеров начал постепенно снижаться, стали обретать актуальность технологии растрового ввода крупноформатных документов. Предлагавшиеся в то время на рынке Беларуси сканеры не отвечали всем появляющимся у потребителей запросам. Г. И. Алексеев предложил технологию пофрагментного комбинированного ввода крупноформатных документов, которые закреплялись на рабочем поле широкоформатного дигитайзера и сканировались по фрагментам с помощью перевернутого и модернизированного для этих целей сканера. Положение каждого фрагмента фиксировалось датчиками координат. На последнем этапе из фрагментов программным путем формировалось широкоформатное растровое изображение документа. Были созданы и поставлялись заказчикам сравнительно недорогие аппаратно-программные комплексы комбинированного ввода ДИСКАН.

В 2001 г. Герту Ивановичу присвоено звание профессора.

В последнее время Герт Иванович работал над программами Союзного государства Беларуси и России, связанными с космическими исследованиями. Под его руководством были разработаны интерактивные системы ввода, визуального отображения и редактирования графической информации. Эти системы могут применяться, в частности, для совместного отображения и обработки планов городов и космических снимков.

Г. И. Алексеев подготовил четырех кандидатов технических наук и одного кандидата физико-математических наук. Неоднократно возглавлял государственные экзаменационные комиссии в БГУИР и БГТУ, работал заместителем председателя докторского совета по защите диссертаций и был членом ученого совета ОИПИ НАН Беларуси. Награжден золотой (1984) и серебряной (1986) медалями ВДНХ СССР, медалью «Ветеран труда» (1986), почетными грамотами НАН Беларуси (1998) и Мингорисполкома (2003). Является автором свыше 130 научных работ, в том числе двух монографий и 76 изобретений. Для своих сотрудников и учеников он был принципиальным и внимательным руководителем, имел большой авторитет в коллективе института.

Герт Иванович умел очень многое делать своими руками, увлекался пчеловодством, играл на нескольких музыкальных инструментах, обладал замечательным чувством юмора, любил рассказывать веселые истории. Все отмечали его энергичность, разносторонность, внимательность к людям.

Основные направления научной и изобретательской деятельности Г. И. Алексеева

Разработка электромагнитных планшетных дигитайзеров. В Беларуси начало исследований по автоматизации ввода графических изображений относят к середине 1960-х гг., когда в ИТК АН БССР в лаборатории читающих автоматов под руководством П. М. Чеголина были заложены теоретические основы автоматизации ввода графической информации в ЭВМ.

Вначале ввод графики предполагалось осуществлять с привлечением различных механических приспособлений. В конце 1960-х гг. сотрудник лаборатории А. А. Рассказов выдвинул идею использования эффекта взаимоиндукции линейного проводника (координатной шины) и короткой приемной цилиндрической катушки индуктивности (указателя координат) для построения электромагнитных (индукционных) измерителей координат – прообразов современных дигитайзеров.

В скором времени появились первые образцы планшетных устройств графического ввода развертывающего преобразования, основанные на анализе полярности информационных сигналов. В их числе так называемые дискретно-дискретные полуавтоматы для считывания графической информации (ПАСГИ) и устройства для считывания графической информации (УСГИ).

В 1973 г. лаборатория читающих автоматов была преобразована в лабораторию автоматизации ввода графической информации. С 1973 по 1978 г. выполнялись хозяйственные договоры

на основе кодировщиков семейства ПАСГИ для предприятий судо-, авиа- и машиностроения. Во всех этих работах активное участие принимал Г. И. Алексеев.

В середине 1970-х гг. на базе ЭВМ «Минск-32», указанных технических средств ввода, а также средств отображения графики в ИТК АН БССР был создан зал машинного проектирования, который стал своего рода прототипом будущих комплексов АРМ. Для ознакомления с достигнутыми результатами институт посещает президент АН СССР академик А. П. Александров. Особо высокую его оценку получил комплекс цифрования сложных графических изображений, технология функционирования которого была продемонстрирована на примере отображения крупномасштабных топографических карт на экране графического дисплея. Был отмечен значительный вклад Г. И. Алексеева в разработку этого комплекса. Через короткое время вышло постановление Совета министров СССР, в соответствии с которым ИТК АН БССР был назначен головным исполнителем крупного проекта по созданию базы цифровых картографических данных в интересах обороны страны. Одна из ключевых ролей в этом проекте отводилась оригинальным технологиям цифрования карт на базе планшетных устройств ввода графической информации. Г. И. Алексеев активно включился в работу над этим проектом.

В 1983–85 гг. было налажено сотрудничество на договорной основе ИТК АН БССР и ОКБМ при заводе «Эвистор» (Витебск), продолжавшееся до 1989 г. Ответственным исполнителем от ИТК АН БССР был Г. И. Алексеев, который в это время работал в лаборатории эргатических систем (зав. лабораторией Г. Г. Маньшин). Подготовлен коллектив разработчиков средств ввода, в том числе два кандидата наук: В. Н. Лабовкин и И. Е. Масько (научный руководитель Г. И. Алексеев). Издана монография Г. И. Алексеева «Электромагнитные планшетные устройства ввода» [1]. Накопленные знания и опыт совместного сотрудничества использованы в ОКБМ при разработке дигитайзеров ЭМ-7069, ЭМ-7079 и ЭМ-7089, выпущенных заводами «Эвистор» и «Этон» (Новолукомль) в количестве около 1000 устройств.

Задачи массового ввода картографической информации обозначили высокие требования к дигитайзерам, особенно по точности и надежности. Промышленно освоенные дигитайзеры семейства ЭМ не соответствовали таким требованиям. В этих условиях была поставлена задача по созданию отечественного крупноформатного дигитайзера, не уступающего по перечисленным техническим параметрам лучшим зарубежным аналогам. В конкурсе на выполнение этой задачи приняли участие два коллектива разработчиков. Э. Н. Леонович с сотрудниками представил дигитайзер на основе амплитудных методов измерений, а коллектив под руководством Г. И. Алексеева (В. Я. Зенин от ИТК; М. С. Люкевич, В. П. Ушаков и В. М. Рухлинский от ЦКБ с ОП) создал дигитайзер на основе фазового подхода к измерениям. Советом главных конструкторов проекта по картографической тематике был проведен тщательный анализ конструкций и характеристик обоих вариантов, в результате был выбран планшет, предложенный Г. И. Алексеевым. Затем были созданы дигитайзеры ПЛАНШЕТ-Р и ПЛАНШЕТ-РА со следующими характеристиками: формат рабочего поля 840×600 мм (максимальная погрешность ±0,1 мм), разрешающая способность 0,025 мм, подсветка и увеличение изображения в зоне считывания, скорость обновления координатной информации в потоковом режиме 10 000 пар координат в секунду, двухстрочное графическое табло. Серийное производство дигитайзеров ПЛАНШЕТ-Р и ПЛАНШЕТ-РА освоено ЦКБ с ОП АН БССР и Ульяновским ПО «Комета» (выпущено около 200 дигитайзеров). ПЛАНШЕТ-Р и ПЛАНШЕТ-РА стали основной специализированного автоматизированного картографического производства.

Дигитайзеры ПЛАНШЕТ-Р и ПЛАНШЕТ-РА в 1983 и 1984 гг. отмечены дипломами I степени, золотыми, серебряными и несколькими бронзовыми медалями ВДНХ СССР. За работы в области специальной картографии ИТК АН БССР награжден орденом Трудового Красного Знамени (1984), группе сотрудников присуждена Государственная премия СССР в области науки и техники. В это же время за создание высокоэффективных изобретений Г. И. Алексееву присуждено почетное звание «Заслуженный изобретатель БССР».

К концу 1980-х гг. в связи с распространением персональных компьютеров и широкой компьютеризацией возникла потребность в массовом отечественном дигитайзере. В 1989 г. по предложению И. К. Ростовцева, генерального директора Опытного-промышленного предприятия Научно-технического объединения (ОПП НТО) АН СССР (ныне ОАО «Приборостроительный

завод Оптрон», Минск), группа специалистов из ИТК АН БССР (Г. И. Алексеев, В. А. Мильман), ЦКБ АН БССР и ОПП НТО АН СССР приступила к разработке недорогого компактного малоформатного дигитайзера. Г. И. Алексеев предложил основные технические решения дигитайзера, где использовался нониусный принцип фазовых измерений координат. Рабочее поле планшета было впервые сконструировано и выполнено в виде трехслойной, склеенной под прессом тонкой пластины толщиной от 3 до 5 мм. Подобные дигитайзеры зарубежных фирм в то время имели объемные конструкции в виде усеченных по наклонной плоскости прямоугольных в основании пирамид. Индукционная система дигитайзера была оптимизирована путем численного моделирования. В электронном блоке дигитайзера использовался микропроцессор, управляющий обработкой сигналов с основной и нониусной системами шин. Особое значение в непростых экономических условиях того времени имела технологичность и простота изготовления дигитайзера, его сравнительно невысокая стоимость. ОПП НТО АН СССР в 1990 и 1991 гг. изготовило крупную серию из 5000 дигитайзеров модели УВГ1-01 с рабочим полем формата А3, на несколько лет обеспечив потребность стран СНГ в подобных устройствах. Этим завершилось импортозамещение аналогов западных фирм отечественными устройствами УВГ1-01 в масштабах СНГ.

В апреле 1989 г. Г. И. Алексеев становится заведующим лабораторией автоматизации ввода видеоинформации. В лаборатории развернулись работы по созданию и совершенствованию электромагнитных планшетных дигитайзеров. На разных этапах по этой тематике работали В. А. Маслоков, В. В. Бокуть, Л. Ю. Мазаник, В. А. Мильман, А. И. Гренов, Д. М. Гороховик, С. Н. Королик, Е. Ю. Васильев, А. А. Камович, А. И. Люлис, Н. Р. Новиков, С. В. Решетник, Е. И. Коновалов и С. Г. Алексеев. Большую часть разработчиков составляли ученики Герта Ивановича.

Дигитайзер УВГ1-01 как базовая модель использовался для создания модельного ряда дигитайзеров различных форматов (до А0 включительно). На протяжении почти 20 лет такие дигитайзеры поставлялись предприятиям и организациям различных отраслей экономики (машиностроения, картографии, легкой промышленности и др.) и государственных структур Беларуси, а также Российской Федерации и Украины.

Особо следует отметить поставку дигитайзеров в Минский горисполкомом, где была создана специализированная геоинформационная система для контроля раскопок на территории города. Значительная часть исходной информации для геоинформационной системы находилась на специальных топографических планах, наклеенных на алюминиевую основу, а обычный электромагнитный дигитайзер не работает с этим материалом. Потребовались существенные усилия Герта Ивановича и его коллектива, в результате необходимый дигитайзер был создан и поставлен заказчику.

В последние десятилетия усилиями Г. И. Алексеева и его сотрудников был разработан складной дигитайзер, вызвавший большой интерес у создателей подвижных пунктов управления для различных силовых структур.

Совершенствование ввода с помощью электромагнитных планшетных дигитайзеров. В процессе создания электромагнитных планшетных дигитайзеров оказалось, что для обеспечения высокого технического уровня изделий только инженерных решений недостаточно и необходим научный анализ процессов электромагнитного ввода. Г. И. Алексеев начал исследования в этом направлении еще будучи сотрудником лаборатории читающих автоматов и продолжил в качестве руководителя лаборатории автоматизации ввода видеоинформации ИТК НАН Беларуси. Исследования опирались на такие разделы физики и математики, как электромагнитная теория, теория аппроксимации функций, теория функций действительной и комплексной переменной, теория оптимизации и вычислительные методы. В процессе этих работ возникали задачи и получались результаты, выходящие за рамки тематики дигитайзерного ввода.

Центральным объектом исследований являлась индукционная система электромагнитного планшетного дигитайзера, качество работы которой определяет точность ввода графической информации. Приведем краткое описание индукционной системы с фазовым принципом измерения координат, так как именно этот тип систем получил широкое распространение.

Индукционная система дигитайзера состоит из указателя координат (УК) и матрицы проводников электрического тока. УК перемещается оператором по поверхности рабочего поля, на котором закрепляется вводимое графическое изображение. Матрица проводников располагается непосредственно под поверхностью рабочего поля и содержит две группы проводников. Одна обеспечивает измерение координаты x , другая – координаты y . Группа, обеспечивающая измерение координаты x , состоит из множества прямолинейных проводников (шин), расположенных перпендикулярно оси Ox . Шины соединяются специальным образом и образуют периодическую структуру (основную сетку). Периоды этой структуры называются зонами.

УК представляет собой короткую цилиндрическую катушку, ориентированную перпендикулярно плоскости шин. При возбуждении УК синусоидальным током частоты ω в основной сетке наводится ЭДС, пропорциональная взаимной индуктивности УК и совокупности шин сетки. Величина взаимной индуктивности зависит от взаимного расположения УК и шин, поэтому ЭДС несет в себе информацию о координате x центра УК. В результате в основной сетке возникает синусоидальный переменный ток с фазовым сдвигом φ , зависящим от координаты x . Фазовый сдвиг преобразуется в цифровую форму, и на его основе рассчитывается значение координаты. Параметры индукционной системы подбираются так, чтобы зависимость фазового сдвига φ от координаты x была почти прямо пропорциональной. В этом случае точность ввода координаты является наивысшей.

Описанное измерение координаты фазовым способом характеризуется неоднозначностью: координата измеряется от начала зоны, а номер зоны остается неопределенным. Для определения номера зоны наиболее эффективным оказался нониусный метод. Он заключается в том, что вместе с основной сеткой проводников в той же плоскости (или в параллельной близкой плоскости) располагается нониусная сетка. Она устроена и работает аналогично, отличается только величиной периода и числом зон. Сравнение координат, снятых с основной и нониусной сеток, позволяет установить номер зоны, т. е. однозначно измерить координату x в пределах рабочего поля дигитайзера.

Исследования Г. И. Алексеева по теоретическим вопросам дигитайзерного ввода графической информации можно разделить на два этапа. На первом этапе (до 1985 г.) Герт Иванович преимущественно вел исследования самостоятельно. Итоги этих исследований отражены в монографии [1]. Второй этап связан с созданием коллектива, руководителем которого был Герт Иванович.

Отметим основные результаты первого этапа исследований.

Получены формулы для расчета характеристик (напряженности, векторного потенциала) магнитных полей, порождаемых переменными токами в элементах индукционных систем дигитайзеров. Показано, что высокая точность расчетов достигается при аппроксимации реальных проводников тонкими проводниками полигональной формы.

Выведены формулы расчета взаимной индуктивности катушки указателя координат и одиночной тонкой шины как конечной, так и бесконечной длины. Оказалось, что для бесконечной шины формулы значительно проще. Показано, что в предположении о бесконечности шин получаются достаточно точные результаты в случае, если УК располагается в центральной части рабочего поля дигитайзера.

Выявлено снижение точности определения координаты фазовым способом, когда УК располагается на краю матрицы дигитайзера. Это явление, называемое краевым эффектом, заставляет уменьшать размеры рабочего поля дигитайзера по сравнению с размером матрицы. Выполнены экспериментальное и численное исследования краевого эффекта.

Выделены и изучены базовые составляющие погрешности измерения координат: методическая, технологическая и температурная погрешности электронной схемы измерительного канала. Предложены определенные меры по снижению их значений.

Выполнены теоретические и экспериментальные исследования дигитайзеров со стержневым указателем координат.

Для дальнейшего развития тематики дигитайзерного ввода требовалось последовательное развитие теории в таких направлениях, как упрощение многих математических формул, неудобных для вычислений, выбор оптимальных параметров сеток координатных шин и указате-

ля координат, разработка мер для минимизации краевого эффекта, а также алгоритмов и программных средств для повышения точности ввода координат. В рамках этих исследований Герт Иванович начал привлекать в лабораторию специалистов с математическим, радиотехническим и компьютерным образованием.

На втором этапе исследований Герт Иванович и руководимый им коллектив получили следующие основные результаты по тематике дигитайзерного ввода.

Были разработаны несколько моделей индукционного взаимодействия УК и сеток матриц планшетного дигитайзера. При определенных упрощающих предположениях получены формулы для расчета взаимодействия в виде комбинаций элементарных функций (без интегралов и бесконечных сумм), что значительно облегчило выбор оптимальных параметров индукционных систем. Получено выражение для вертикальной составляющей вектора магнитной индукции, порождаемого током в сетке с бесконечным числом шин бесконечной длины, и выражение для поправки, которую необходимо учитывать при переходе от бесконечного числа шин к конечному, что важно для анализа краевого эффекта [2].

Получены формулы для расчета индукционного взаимодействия УК и координатной шины с учетом ширины и высоты шины [3].

Предложен новый подход к анализу индукционного взаимодействия УК и матрицы дигитайзера на упрощенной модели, когда шины полагаются тонкими бесконечными и их число бесконечно [4]. Подход основан на разложении в ряд Фурье зависимости взаимной индуктивности матрицы и УК. Получено выражение зависимости тангенса методической погрешности ввода координат от параметров сетки и УК и уравнение для определения оптимального значения среднего радиуса катушки УК.

Разработан способ повышения точности ввода графической информации с помощью дигитайзеров [5]. Он предполагает однократное измерение погрешности с помощью обмера эталонного объекта, вычисление таблиц поправок, корректировку вводимых координат на основе таблиц поправок. Возможны два варианта: таблицы поправок размещаются в теле драйвера дигитайзера и корректировка координат выполняется программно; таблицы поправок размещаются в памяти микропроцессора и корректировка координат выполняется микропрограммно. В результате разработки была повышена точность ввода координат более чем в два раза без наращивания аппаратных средств дигитайзера.

В процессе расчетов индукционного взаимодействия шины дигитайзера и катушки стержневого указателя координат получено аналитическое выражение для расчета взаимной индуктивности прямолинейного бесконечного проводника и контура эллиптической формы, произвольно ориентированного в пространстве [6]. Рассмотрен расчет взаимной индуктивности прямолинейного бесконечного проводника и круговой катушки. Предложен способ вычислений, позволяющий для цилиндрической катушки вообще избежать численного интегрирования. В целом задача сводится к вычислению контурного интеграла вместо двойного, что позволяет значительно ускорить вычисления [7]. Отметим, что расчет взаимной индуктивности проводников различной формы и расположения удается провести аналитически (без численного интегрирования) в небольшом числе случаев (не более 10), все они были установлены в 1900–1960 гг. [8]. Результаты работ [6, 7] дополняют материал известных справочников по расчету индуктивностей и могут войти в последующие их издания.

С 2005 г. Г. И. Алексеев вместе с С. В. Абламейко и С. Г. Алексеевым занимался исследованием индукционных систем складных и секционных дигитайзеров [9]. Складные дигитайзеры удобны для работы в мобильных комплексах различного назначения. В секционных дигитайзерах матрица собирается из составных частей (печатных плат). Это позволяет изготавливать матрицы дигитайзеров большого формата из печатных плат ограниченного формата. Изучение индуктивностей таких матриц значительно сложнее, чем традиционных матриц. В исследованиях значительную роль играли вычислительные эксперименты в системе MathCAD.

Ввод крупноформатных изображений с помощью комбинированных устройств. С 2000 г. основной темой исследований Г. И. Алексеева и его лаборатории стал ввод в компьютер крупноформатных изображений. Существует несколько традиционных способов такого ввода, каж-

дый из них имеет свои недостатки [10, 11]. Г. И. Алексеев предложил вводить крупноформатные изображения с помощью комбинированных устройств, объединяющих дигитайзер и сканер.

Комбинированное устройство, получившее название ДИСКАН, состоит из широкоформатного планшетного дигитайзера и малоформатного планшетного сканера. Сканер используется в перевернутом положении, для чего сделаны небольшие доработки. В корпус сканера монтируются индукционные датчики координат, аналогичные указателю координат дигитайзера.

На рабочем поле дигитайзера закрепляется вводимое изображение на носителе из диэлектрического материала. Для фиксации изображения в процессе ввода его накрывают специальным тонким прозрачным стеклом. Сначала малоформатным планшетным сканером выполняется сканирование документа по фрагментам, которые сохраняются в файлах в растровом формате. Кроме того, для каждого фрагмента сохраняются координаты точек, полностью определяющих его положение в системе координат дигитайзера. Затем выполняется программное формирование широкоформатного изображения из фрагментов и результат сохраняется в файл.

Разработаны несколько модификаций комбинированного устройства ввода, отличающихся типами и форматами дигитайзера и сканера, количеством индукционных датчиков в сканере, а также дополнительными техническими доработками. Представлен вариант комбинированного устройства без дигитайзера. В этом устройстве сканер позиционируется на рабочем поле с помощью специальных механических упоров. Такое устройство предназначено для ввода планов городов фиксированного формата, в том числе выполненных на алюминиевой основе.

Для обеспечения комбинированного ввода разработаны алгоритмы и программы:

- коррекции геометрических и яркостных искажений фрагментов сканированных изображений;
- визуализации широкоформатных растровых изображений с возможностью их одновременной линейной трансформации в режиме реального времени;
- совмещения фрагментов изображения с целью формирования единого растрового файла;
- визуализации процесса комбинированного ввода.

Было изготовлено и поставлено заказчикам несколько десятков комплексов ввода ДИСКАН. Наиболее важными заказчиками являлись Навигационно-топографическое управление Генерального штаба Вооруженных Сил Республики Беларусь, РУП «Белгеодезия», Комитет архитектуры и градостроительства Мингорисполкома. На комплексы ввода ДИСКАН получен патент на полезную модель. Под руководством Герта Ивановича А. И. Люлис защитил кандидатскую диссертацию по тематике ввода крупноформатных изображений с помощью комбинированных устройств.

Хотя в настоящее время часть задач ввода крупноформатных изображений решается с помощью планшетных сканеров, комбинированные устройства сохраняют свое значение для ввода сверхбольших изображений, для работы в мобильных и других нестандартных условиях.

Список использованных источников

1. Алексеев, Г. И. Электромагнитные планшетные устройства ввода / Г. И. Алексеев. – Минск : Наука и техника, 1985. – 239 с.
2. Гренов, А. И. Математическое моделирование и проектирование нониусных индукционных систем планшетных дигитайзеров : автореф. дис. ... канд. техн. наук : 05.13.16 ; 05.13.12 / А. И. Гренов ; Ин-т техн. кибернетики НАН Беларуси. – Минск, 1997. – 20 с.
3. Алексеев, Г. И. Расчет индукционного взаимодействия координатной шины и указателя координат электромагнитного дигитайзера / Г. И. Алексеев, А. Н. Догель, В. А. Мильман // Идентификация образов : сб. науч. тр. – Минск : Ин-т техн. кибернетики АН БССР, 1990. – С. 172–185.
4. Алексеев, Г. И. Расчет и оптимизация индукционной системы дигитайзера на упрощенной модели / Г. И. Алексеев, Е. Ю. Васильев, В. А. Мильман // Вес. Нац. акад. наук Беларуси. Сер. физ.-техн. наук. – 2001. – № 3. – С. 119–115.
5. Цифровые планшеты с коррекцией погрешности измерения координат / Г. И. Алексеев [и др.] // Вес. Нац. акад. наук Беларуси. Сер. физ.-техн. наук. – 1998. – № 1. – С. 135.
6. Алексеев, Г. И. О расчете взаимной индуктивности прямолинейного проводника и эллиптического контура / Г. И. Алексеев, В. А. Мильман // Электричество. – 1993. – № 12. – С. 53–57.

7. Алексеев, Г. И. Расчет взаимной индуктивности прямолинейного проводника и круговой катушки / Г. И. Алексеев, В. А. Мильман, В. В. Шейко // Электричество. – 1995. – № 4. – С. 59–64.
8. Калантаров, П. Л. Расчет индуктивностей / П. Л. Калантаров, Л. А. Цейтлин. – Изд. 3. – М. : Энергоатомиздат, 1986. – 488 с.
9. Абламейко, С. В. Технические средства аппаратно-программных комплексов комбинированного ввода крупноформатных графических документов / С. В. Абламейко, Г. И. Алексеев, С. Г. Алексеев // Вестник компьютерных и информационных технологий. – 2009. – № 3. – С. 37–48.
10. Комбинированный ввод в компьютер крупноформатных изображений / Г. И. Алексеев [и др.] // Вес. Нац. акад. навук Беларусі. Сер. фіз.-тэхн. навук. – 2003. – № 4. – С. 115–119.
11. Комбинированный ввод в компьютер графических изображений с использованием малоформатного планшетного сканера и дигитайзера / Г. И. Алексеев [и др.] // Информатика. – 2006. – № 4(12). – С. 71–78.

С. Г. Алексеев, В. А. Мильман, О. И. Семенков

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

1 | 2023

ТОМ ■ VOL. 20

ИНФОРМАТИКА ■ INFORMATICS

ISSN 1816-0301 (Print)



9 771816 030000