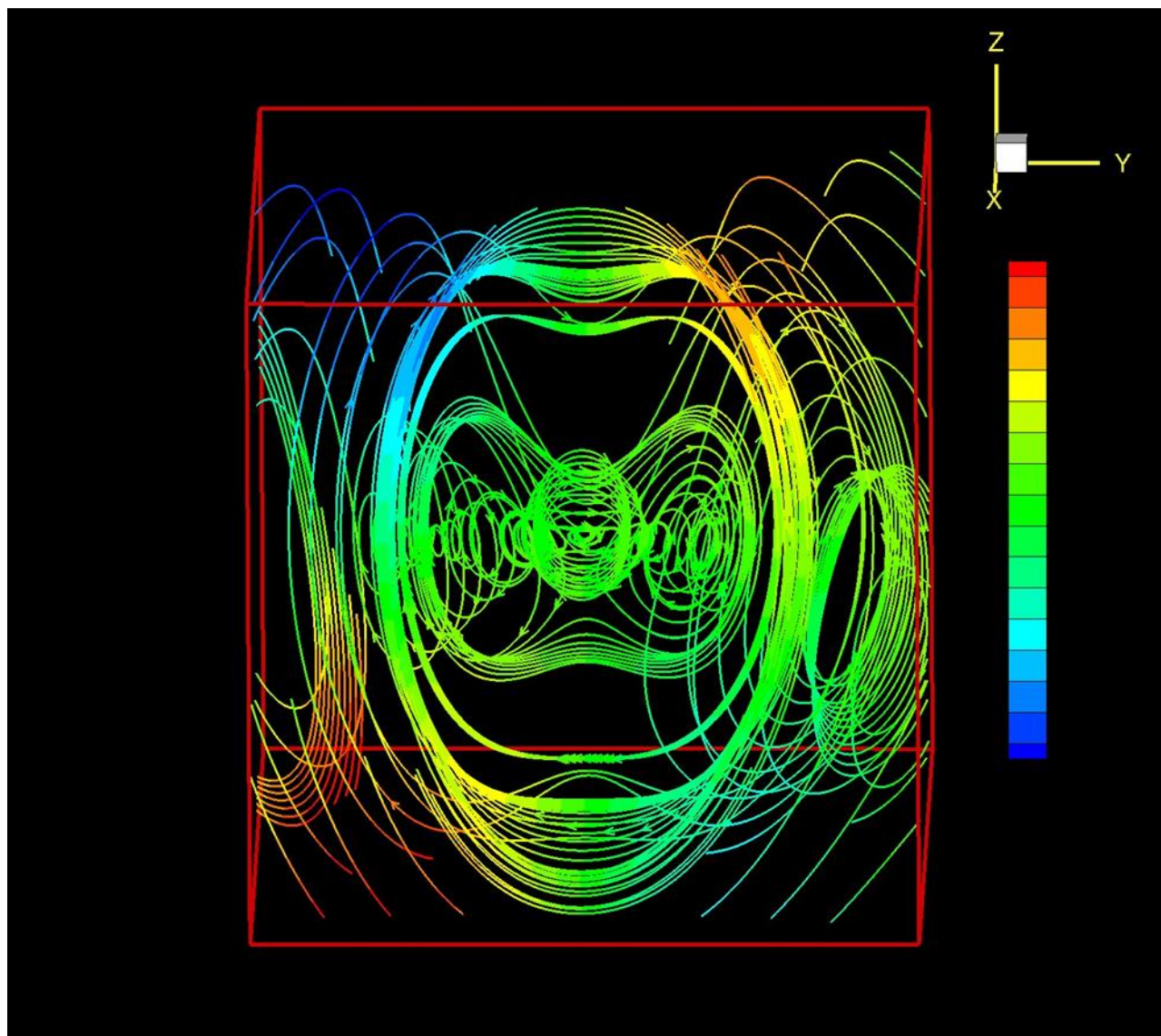




Вестник кибернетики

Proceedings in Cybernetics

№ 3 (47)

12+

2022

**БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
ХАНТЫ-МАНСКИЙСКОГО АВТОНОМНОГО ОКРУГА – ЮГРЫ
«СУРГУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Вестник кибернетики

Proceedings in Cybernetics

№ 3 (47)

Сургут
2022

Учредитель и издатель

Бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры
«Сургутский государственный университет»

Главный редактор

Острейковский Владислав Алексеевич, д. т. н., профессор

Заместитель главного редактора

Тараканов Дмитрий Викторович, к. т. н., доцент

Ответственный секретарь

Брагинский Михаил Яковлевич, к. т. н., доцент

Члены редакционной коллегии:

Советов Б. Я., академик РАН и РАО, д. т. н., профессор
Юсупов Р. М., член-корреспондент РАН, д. т. н., профессор
Абрамов О. В., д. т. н., профессор
Анохин А. Н., д. т. н., профессор
Антонов А. В., д. т. н., профессор
Бурханов Р. А., д. филос. н., профессор
Бушмелева К. И., д. т. н., профессор
Гетман А. Ф., д. т. н., профессор
Григорьев Л. И., д. т. н., профессор
Древс Ю. Г., д. т. н., профессор
Ельников А. В., д. ф.-м. н., профессор
Инютин С. А., д. т. н., профессор
Каштанов В. А., д. ф.-м. н., профессор
Керимов Т. Х., д. филос. н., профессор
Леонов Д. Г., д. т. н., профессор
Малышев Д. С., д. ф.-м. н., профессор
Мельников А. В., д. т. н., профессор
Певзнер Л. Д., д. т. н., профессор
Увайсов С. У., д. т. н., профессор
Цибульский В. Р., д. т. н., профессор
Юрков Н. К., д. т. н., профессор

Выпускающий редактор

Хасанова А. Ш.

Редактор

Бобрус Ю. Р.

Верстка

Мельниковой Е. А.

Переводчик

Петрова А. В.

Рисунок на обложке:

Визуализация магнитогидродинамических расчетов для точного решения внутри сферы
© Галкин В. А., Гореликов А. В.

Решением Высшей аттестационной комиссии с 29 мая 2017 года журнал включен в «Перечень рецензируемых научных изданий, в которых должны быть опубликованы основные научные результаты диссертаций на соискание ученой степени кандидата наук, на соискание ученой степени доктора наук».
С 29.03.2022 издание принимает к публикации статьи по следующим научным специальностям: 2.3.1. – системный анализ, управление и обработка информации, 1.2.2. – математическое моделирование, численные методы и комплексы программ, 2.2.2. электронная компонентная база микро- и нанoeлектроники, квантовых устройств

Журнал зарегистрирован Федеральной службой по надзору в сфере связи, информационных технологий и массовых коммуникаций. Свидетельство о регистрации средства массовой информации

Эл № ФС77-83015 от 31.03.2022

Издается с 2002 года. Выпускается 4 раза в год

Адрес редакции:

БУ ВО «Сургутский государственный университет»,
628412, Россия, Ханты-Мансийский автономный округ – Югра, г. Сургут, пр. Ленина, 1
Тел.: (3462) 76-29-88, факс: (3462) 76-29-29, e-mail: proceed.cyber@mail.ru. Сайт: vestcyber.ru

Founder and Publisher
Surgut State University

Chief Editor

Prof. Vladislav A. Ostreikovskiy, Doctor of Sciences (Engineering)

Vice Chief Editor

Dmitry V. Tarakanov, Candidate of Sciences (Engineering), Associate Professor

Executive Editor

Mikhail Ya. Braginsky, Candidate of Sciences (Engineering), Associate Professor

Editorial Board:

Prof. Sovetov B. Ya., Doctor of Sciences (Engineering), member of the Russian Academy of Sciences (RAS), the Russian Academy of Education (RAE)

Prof. Yusupov R. M., Doctor of Sciences (Engineering), member of the Russian Academy of Sciences (RAS)

Prof. Abramov O. V., Doctor of Sciences (Engineering)

Prof. Anokhin A. N., Doctor of Sciences (Engineering)

Prof. Antonov A. V., Doctor of Sciences (Engineering)

Prof. Burkhanov R. A., Doctor of Sciences (Philosophy)

Prof. Bushmeleva K. I., Doctor of Sciences (Engineering)

Prof. Getman A. F., Doctor of Sciences (Engineering)

Prof. Grigoryev L. I., Doctor of Sciences (Engineering)

Prof. Dreys Yu. G., Doctor of Sciences (Engineering)

Prof. Elnikov A. V., Doctor of Sciences (Phys&Math)

Prof. Inyutin S. A., Doctor of Sciences (Engineering)

Prof. Kashtanov V. A., Doctor of Sciences (Phys&Math)

Prof. Kerimov T. Kh., Doctor of Sciences (Philosophy)

Prof. Leonov D. G., Doctor of Sciences (Engineering)

Prof. Malyshev D. S., Doctor of Sciences (Phys&Math)

Prof. Melnikov A. V., Doctor of Sciences (Engineering)

Prof. Pevzner L. D., Doctor of Sciences (Engineering)

Prof. Uvaisov S. U., Doctor of Sciences (Engineering)

Prof. Tsibulsky V. R., Doctor of Sciences (Engineering)

Prof. Yurkov N. K., Doctor of Sciences (Engineering)

Publishing Editor

Khasanova A. Sh.

Editor

Bobrus Yu. R.

Layout

Melnikova E. A.

Translator

Petrova A. V.

Cover Image:

A visualization of magnetohydrodynamic analysis used to find an exact solution within a sphere

© Galkin V. A., Gorelikov A. B.

Since 29.05.2017 the journal is included in the List of Leading Peer-Reviewed Scientific Journals Published in the RF by the Higher Attestation Commission, which publishes main scientific results of doctor's and candidate's theses.

Since March 29, 2022, the journal is included in the List on the following subject groups:

2.3.1. – System Analysis, Data Management and Processing; 1.2.2. – Mathematical Modeling, Calculus and Program Complexes;

2.2.2. – Electronic Component Base of Micro- and Nanoelectronics, Quantum Devices

The journal is registered with the Federal Service for Supervision of Communications, Information Technology and Mass Media. Mass media registration certificate El No. FS77-83015 dated on 31.03.2022

Published since 2002. 4 issues per year

Editorial Board Address:

Surgut State University, Russia 628412, Khanty-Mansi Autonomous Okrug – Ugra, Surgut, Lenina pr., 1

Tel.: +7 (3462) 76-2988, fax: +7 (3462) 76-2929, e-mail: proceed.cyber@mail.ru. Web: vestcyber.ru

СОДЕРЖАНИЕ

ТЕХНИЧЕСКИЕ НАУКИ

Акперов Г. И., Александрова Т. С., Дегтярев В. В., Храмов В. В.	
Методика выбора языка программирования на основе мягких моделей	6
Бобровская О. П., Лысенкова С. А.	
Математическое моделирование и анализ для определения силы игрока сёги	14
Веремчук Н. С.	
Экспериментальное исследование системы массового обслуживания клиентов в банке	25
Григоренко В. В., Назина Н. Б.	
Нелинейный анализ параметров кардиосигналов с хаотической динамикой	32
Крамаров С. О., Овсянников В. А., Сахарова Л. В., Усатый Р. С., Лукьянова Г. В.	
Автоматизированный сбор данных ключевых финансовых показателей предприятий ИТ-отрасли региона	39
Полухин П. В.	
Применение байесовских методов для оптимизации обучения нейросетевых моделей тестирования приложений	46
Серобабов А. С.	
Выбор ключевых параметров для диагностики заболевания печени на основе метода анализа иерархий	57
Урманцева Н. Р., Курамишина А. В.	
Система поддержки принятия решений в медицине: подходы и методы применения во флебологии	66

ФИЗИКО-МАТЕМАТИЧЕСКИЕ НАУКИ

Алексеев М. М., Семенов О. Ю.	
Изучение распространения пламени в расширяющейся сфере	75
Гиниятуллин В. М., Хлыбов А. В., Федоров М. А., Асадуллин Т. А., Крутин А. С., Осипов И. А., Зарипов Д. М.	
Сравнение типов ядер в сверточных слоях нейронных сетей	84
Дубанов А. А.	
Геометрические модели уклонения цели от преследователя	99
Заводовский А. Г.	
Температурная зависимость удельного вращения плоскости поляризации для водного раствора сахара	109
Инютин С. А., Иванов В. О.	
Модельный комплекс поиска оснований модулярной арифметики для предотвращения бивалентного эффекта	114

CONTENTS

ENGINEERING

Akperov G. I., Aleksandrova T. S., Degtyarev V. V., Khramov V. V. Method for Selecting a Programming Language Based on Soft Models	6
Bobrovskaya O. P., Lysenkova S. A. Mathematical Modeling and Analysis to Determine a Shogi Player's Skill Level	14
Veremchuk N. S. Experimental Study of the Mass Customer Service System in the Bank	25
Grigorenko V. V., Nazina N. B. Non-Linear Analysis of Cardiac Signals Parameters with Chaotic Dynamics	32
Kramarov S. O., Ovsyannikov V. A., Sakharova L. V., Usatyi R. S., Lukyanova G. V. Automated Data Collection of Key Financial Indicators of IT Enterprises in the Region	39
Polukhin P. V. Using Bayesian Methods to Optimize Training of Neural Network Models for Application Testing	46
Serobabov A. S. Selecting Key Parameters for the Diagnosis of Liver Disease Based on the Analytic Hierarchy Process	57
Urmantseva N. R., Kuramshina A. V. Decision-Making Support System in Medicine: Approaches and Application Methods in Phlebology	66

PHYSICS AND MATHEMATICS

Alekseev M. M., Semenov O. Yu. Study of Flame Propagation in an Expanding Sphere	75
Giniyatullin V. M., Khlybov A. V., Fedorov M. A., Asadullin T. A., Krutin A. S., Osipov I. A., Zaripov D. M. Comparing Types of Kernels in Convolutional Layers of Neural Networks	84
Dubnov A. A. Geometric Models of Target Evasion from a Pursuer	99
Zavodovsky A. G. Temperature Dependence of the Specific Rotation of Sugar Solution	109
Inyutin S. A., Ivanov V. O. Model Complex for Searching for Modular Arithmetic Bases to Prevent the Bivalent Effect	114

ТЕХНИЧЕСКИЕ НАУКИ

Научная статья

УДК 004.43

doi: 10.34822/1999-7604-2022-3-6-13

МЕТОДИКА ВЫБОРА ЯЗЫКА ПРОГРАММИРОВАНИЯ НА ОСНОВЕ МЯГКИХ МОДЕЛЕЙ

**Гурру Имранович Акперов¹, Татьяна Сергеевна Александрова²,
Владимир Владимирович Дегтярев³, Владимир Викторович Храмов^{4✉}**

^{1, 2, 3, 4} Южный университет (ИУБуП), Ростов-на-Дону, Россия

¹ pr@iubip.ru, <https://orcid.org/0000-0002-9057-0001>

² tanysa08@yandex.ru, <https://orcid.org/0000-0001-9852-0423>

³ degvladimir@mail.ru, <https://orcid.org/0000-0002-7574-4130>

⁴ vxramov@inbox.ru ✉, <https://orcid.org/0000-0003-1848-8174>

Аннотация. Предлагается методика выбора языка программирования для разработки средств поддержки управления образовательным процессом в региональном вузе, ориентированная на мягкие модели и вычисления. Показано, что в условиях «турбулентности» региональным вузам необходимо иметь дополнительный запас устойчивости при наличии значительного числа НЕ-факторов, влияющих как на сам учебный процесс, так и на его организацию. Рассмотрен вариант реализации заявленной методики. Сделан вывод о том, что в условиях неопределенности в качестве программных средств предлагается использовать следующие языки программирования: DELPHI, Visual Basic, а также более востребованный в современных приложениях язык программирования Python.

Ключевые слова: сложность алгоритмов, условия и ограничения для программных средств, динамика изменений требований, устойчивость решения

Для цитирования: Акперов Г. И., Александрова Т. С., Дегтярев В. В., Храмов В. В. Методика выбора языка программирования на основе мягких моделей // Вестник кибернетики. 2022. № 3 (47). С. 6–13. DOI 10.34822/1999-7604-2022-3-6-13.

Original article

METHOD FOR SELECTING A PROGRAMMING LANGUAGE BASED ON SOFT MODELS

**Gurru I. Akperov¹, Tatyana S. Aleksandrova²,
Vladimir V. Degtyarev³, Vladimir V. Khramov^{4✉}**

^{1, 2, 3, 4} Southern University (IMBL), Rostov-on-Don, Russia

¹ pr@iubip.ru, <https://orcid.org/0000-0002-9057-0001>

² tanysa08@yandex.ru, <https://orcid.org/0000-0001-9852-0423>

³ degvladimir@mail.ru, <https://orcid.org/0000-0002-7574-4130>

⁴ vxramov@inbox.ru ✉, <https://orcid.org/0000-0003-1848-8174>

Abstract. The methodology for selecting a programming language is proposed to develop support tools for managing the educational process in a regional university, with focus on soft models and calculations. The study shows that it is necessary to provide additional stability for a regional university in “turbulent” conditions in the case of a significant number of non-factors affecting both the educational process itself and its management. The study presents an implementation of the method claimed. It is proposed to use

the following programming languages under conditions of uncertainty: DELPHI, Visual Basic, and Python, the most common one in modern applications.

Keywords: complexity of algorithms, conditions and restrictions for software tools, dynamics of changes in requirements, solution stability

For citation: Akperov G. I., Aleksandrova T. S., Degtyarev V. V., Khramov V. V. *Method for Selecting a Programming Language Based on Soft Models // Proceedings in Cybernetics. 2022. No. 3 (47). P. 6–13. DOI 10.34822/1999-7604-2022-3-6-13.*

ВВЕДЕНИЕ

В условиях цифровизации всех сторон жизни общества, внедрение интеллектуальных компонентов в управление образовательным процессом представляется вполне предсказуемым [1]. Другое дело, если этот процесс усложняется из-за непредсказуемых, даже неблагоприятных внутренних и внешних факторов, вызванных социально-экономическими и террористическими угрозами. В этих условиях вузам в регионах, особенно граничащих с «нестабильными» соседями, приходится постоянно вносить коррективы в программные средства, поддерживающие управление учебными заведениями. Следует учитывать, что их финансовые возможности, необходимые для обновления программ, не соизмеримы с возможностями центральных вузов. Поэтому для региональных вузов актуальной становится задача программной адаптации имеющихся возможностей к постоянно усложняющимся требованиям.

Под сложностью разработки программных средств, поддерживающих управление учебным процессом, будем понимать «трудность решения указанной задачи, измеренную в терминах некоторого ресурса, потребляемого в процессе вычисления» [2, с. 23]. Речь идет об алгоритмах и реализующих их программах, для которых сложность имеет временной, программный и информационный аспект.

В алгоритме решения задачи адаптации программных дидактических средств [3] в общем случае используются неточные, субъективные исходные данные, которые характерны для дидактического процесса. Учитывая быстроедействие штатного комплекса вычислительных средств, применение эвристических алгоритмов решения задачи с нечетким

набором данных позволяет повысить эффективность вычислений. Время выступает как функция от объема входных данных, требуемых для описания задачи. В зависимости от типа функции времени алгоритм можно отнести к одному из классов сложности: P-задачи, NP-задачи, NP-полные задачи [2, 3].

МАТЕРИАЛЫ И МЕТОДЫ

Оценка сложности программного средства. Программный аспект сложности средств поддержки методики адаптации заключается в оценке числа символов n_c в тексте программы, реализованной на языке программирования, или числа операторов n_o программы. Наряду с таким достоинством указанного подхода, как простота в определении n_c и n_o , представляется трудным сравнение программ, описанных различными языками программирования с соответствующими им структурами команд, а также с неодинаковым «количеством используемых байт на одну операцию» [3, с. 34].

При исследовании сложности программы, состоящей, как правило, из программных модулей, основываются на анализе внутренней структуры модулей. «О структурной сложности программ можно судить по числу взаимодействующих компонент, количеству связей между компонентами и сложности их взаимодействия» [2, с. 71]. В результате «структурная сложность во многом определяется набором маршрутов M_k , по которым выполняется программа, или суммарным числом условий γ_k , которое необходимо задать для прохождения всех маршрутов k -й программы» [4, с. 36]:

$$\gamma_k = \sum_{i=1}^{M_k} \gamma_i, \quad (1)$$

где γ_i – условия, определяющие i -й маршрут.

Маршруты разделяются на вычислительные процессы и логические решения. При оценке первых необходимо учитывать количество операндов, участвующих в вычислении. Тогда сложность можно определить следующим образом [2]:

$$\eta_k = \sum_{i=1}^{M_k} \sum_j l_i v_{ij}, \quad (2)$$

где l_i – число обрабатываемых операндов на каждом i -м маршруте;

v_{ij} – количество значений для каждой исходной j -й переменной на i -м маршруте [3, с. 72].

Существуют другие подходы к оценке сложности алгоритмических и программных средств, например:

1) в виде оценки, связанной с реализацией интерфейса между двумя и более модулями через один раздел данных [5]:

$$S_u = m K (m-1), \quad (3)$$

где m – число модулей;

K – число связанных элементов среди m ;

2) в виде цикломатической сложности Маккейба [6] для графа с a вершинами, b ребрами и c компонентами связи:

$$S_M = a - b + c. \quad (4)$$

3) в виде сложности понимания как осознание смысла работы модуля с последующим воспроизведением выражается формулой:

$$S_n = f(\varepsilon, t), \quad (5)$$

где ε – воспроизведение программы в процентном отношении;

t – время изучения программы.

Таким образом, «сложность реальных модулей можно охарактеризовать положением модуля в системе иерархий; особенностями и типом структуры ациклической части модуля; наличием в модуле циклов и их характеристиками; характером вычислительного процесса в модуле» [7, с. 23]; «характеристиками путей реализации исполнения программы» [2].

В результате анализа алгоритмов поддержки управления в вузе [8] выяснилось, что их полиномиальная сложность не пре-

вышает уровня 2. Следовательно, все алгоритмы могут быть выполнены в реальном времени вычислительными средствами персональных компьютеров.

Выбор компонентов пакета программных средств обучения. Реализацию ПО целесообразно ориентировать на комплекты персональных компьютеров (далее – ПК). Применение ПК, а не систем на специализированных микропроцессорах в качестве базовых ПК, обусловлено требованиями универсальности, минимальной стоимости и достаточности для решения поставленных задач. Из существующего парка ПК широкое распространение в российских вузах получили компьютеры, оснащенные продуктами Microsoft с развитым программным обеспечением.

Программная среда, используемая в персональных компьютерах, представляется в виде следующих оболочек: 1) операционная система; 2) файловая система; 3) система программирования на каком-либо языке с соответствующими инструментальными средствами; 4) прикладные программы. При разработке программных средств реализации методики адаптации важным является выбор языка программирования (далее – ЯП) и инструментальных средств из множества существующих.

При процедуре выбора ЯП целесообразно, прежде всего, «сформировать характеристики сравниваемых языков с целью оценки их качества» [9]. Необходимо иметь в виду, «что некоторый язык программирования, получивший очень высокую оценку своего качества по сравнению с другими, может оказаться не самым лучшим для определенной области применения» [10]. При выборе наилучшего ЯП для решения конкретной задачи целесообразно придерживаться следующей последовательности:

1. Сформулировать характеристику задачи с точки зрения программирования. Очень часто решаемая задача захватывает сразу несколько областей применения, наиболее распространенными среди них являются следующие: научно-технические расчеты (численное программирование); машинно-зависимое системное программирование (доступ к используе-

мым аппаратным средствам); машинно-независимое программирование (параллельная обработка); обработка текстов (манипуляция над строками символов); программирование систем реального времени (контроль планирования выполнения).

2. Выделить наиболее существенные средства сравниваемых ЯП, являющиеся критическими с точки зрения удобства, безопасности и эффективности для решаемой задачи в соответствии с областью ее применения. Если решаемая задача охватывает сразу несколько областей применения, то это целесообразно проделать для каждой из областей.

3. Определить пригодность каждого из сравниваемых языков для использования в соответствующей области применения.

Отмеченная процедура обеспечивает возможность принятия обоснованного решения о выборе того или иного языка программирования и/или другого инструментального средства для решения соответствующей задачи.

Таким образом, актуальным становится вопрос о выборе подходящего ЯП. Цель выбора – выделение из множества существующих ЯП наиболее «перспективных и их ранжирование по степени удовлетворения требованиям к программным средствам ИУМ на основании имеющихся к моменту начала решения задачи исходных данных» [9]. Неопределенность исходных данных требует использования методов принятия решений, основанных, например [10], на применении нечетких множеств для задания исходных данных и представления результатов. Исходя из сказанного, требования системы зададим в виде вектора требований (далее – ВТ), представленных нечетко.

Введем понятие лингвистической переменной и на ее базе реализуем лингвистический подход к принятию решения выбора ЯП в условиях неопределенности. В рамках этого подхода «лингвистической будем называть переменную, значениями которой могут являться слова или предложения естественного или искусственного языка» [11].

Пусть Y – универсальное множество выходных характеристик ЯП, а Z_i – множество значений характеристики $y_i \in Y, i = \overline{1, n}$. Нечеткое множество S_i , определяемое на мно-

жестве Z_i с функцией принадлежности μ_i , задает требование g_i к характеристике y_i ЯП. Совокупность таких требований образует вектор $G = \{g_i, \dots, g_j\}$.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В рамках нечетких множеств и лингвистических переменных может быть предложена процедура определения ЯП, удовлетворяющего вектору требований, которая включает следующие шаги:

Шаг 1. Формирование множества языков программирования $X = \{x_i\}$.

Шаг 2. Отображение «множества X на множество значений характеристик» Z , которое имеет вид $f_i : X \rightarrow z_i, i = \overline{1, n}$ с последующим нормированием множеств Z_i , если соответствующее множество S_i нормировано.

Шаг 3. Определение нечетких подмножеств Q_i с функциями принадлежности:

$$\mu_i(f_i(X_i)), i = \overline{1, n}, \quad (6)$$

задаваемых на множестве ЯП X по требованиям g_i к характеристикам $y_i \in Y$.

Шаг 4. Нахождение пересечения подмножеств Q :

$$Q = \bigcap_{i=1}^n Q_i, \quad (7)$$

где Q – искомое нечеткое подмножество ЯП, удовлетворяющее вектору требований G .

При определении пересечения нечетких множеств воспользуемся «мультипликативной формой представления функции принадлежности множества-пересечения $P(x_j)$ через функции принадлежности составляющих множеств» [12]:

$$P(x_j) = \bigcap_{i=1}^n \mu_i^\alpha(f_i(x_j)), j = \overline{1, n}. \quad (8)$$

Выражение учитывает вес α отдельных характеристик.

Шаг 5. Определение подходящего ЯП с использованием минимаксного критерия:

$$P \times (x_j) = \min_j \{\max(\mu_{ij})\}. \quad (9)$$

Основываясь на приведенной выше методике, модифицированной применительно

к классу решаемых задач в части шагов 4 и 5, проведем оценку и выбор программного комплекса применительно к возможностям учебно-материальной базы вуза. Пусть требуется выбрать вариант вычислительного средства и ЯП, удовлетворяющий некоторым требованиям.

Обозначим x_j -й вариант использования вычислительного средства и ЯП, причем

$$\begin{aligned} x_1 &= (p_1, u_1), x_2 = (p_2, u_1), \dots, \\ x_9 &= (p_{10}, u_2), \dots, x_{24} = (p_{13}, u_8). \end{aligned} \quad (10)$$

На шаге 2 представим отображение $f_i: X \rightarrow z_i$ в виде таблицы размерностью $(N \times n)$, в которой столбцы соответствуют вариантам выбора, а строки – характеристикам. Элемент a_{ij} , стоящий на пересечении j -й строки и i -го столбца в табл. 1, равен z_{ij} – значению характеристики y_i для варианта выбора с номером j , то есть $z_{ij} = f_i(x_j)$.

Предположим, что используются следующие характеристики:

y_1 – относительная стоимость – стоимость выполнения элементарной базовой операции в вычислительной системе, y . е./бит;

y_2 – полная стоимость, понимаемая как сумма стоимостей вычислительного средства

и программной реализации ЯП и выраженная в условных единицах;

y_3 – поддержка мультимедиа-средств, выраженная в относительных единицах;

y_4 – удобство эксплуатации – показатель качества системы, отражающий простоту доступа и использования, возможность осуществления эксплуатации с минимальными затратами пользователя.

При использовании указанных характеристик возможны следующие исходы выбора программного комплекса:

а) если из совокупности y_i исключить характеристику y_1 , т. е. для (y_2, y_3, y_4) , то процедура выбора осуществляется без учета перспектив развития средств вычислительной техники, исходя из имеющихся в наличии средств;

б) учет совокупности (y_1, y_3, y_4) характеризует выбор программных средств с перспективой развития в условиях стабильного финансирования;

в) совокупность всех характеристик (y_1, y_2, y_3, y_4) необходимо учитывать в период перехода к условиям стабильности финансирования.

Таблица 1

Варианты выбора и их свойства

Характеристика y_i	1	2	3	4	...	23	24
y_1	0,05	0,01	0,005	0,001	...	0,0009	0,0005
y_2	264	735	1322	2000	...	2453	2645
y_3	0	0	0,1	0,3	...	0,8	0,96
y_4	0,1	0,3	0,5	0,7	...	0,76	0,89

Примечание: составлено авторами.

Источники исходных данных в табл. 1 – данные по характеристикам y_1 и y_2 – взяты из [12], данные по характеристикам y_3 и y_4 получены методом опроса экспертов в рамках проведения НИР по договору № 03/1 ЧОУ ВО «ЮУ (ИУБИП)» и АНО ВО «РосНОУ» «Исследование связности компонентов единого информационного пространства» [13].

Таблица задает на непрерывном множестве значений характеристик z_i дискретные подмножества:

$$z_i^* = \bigcup_{j=1}^N z_{ij}^*, \quad (11)$$

где i – тип характеристики ($i = \overline{1, n}$ в нашем случае $n = 4$);

j – вариант выбора.

В [14, 15] предполагается то, что z_i^* – обычные точечные множества, поэтому каждый вариант характеризуется одним значением каждой из характеристик y_i .

Для нормирования значений z_{ij} применим выражение из [16]:

$$\mu_{ij} = \frac{z_{ij}}{z_{i\max}}, \quad (12)$$

или

$$\mu_{ij} = 1 - \frac{z_{ij}}{z_{i\max}}, \quad (13)$$

где $z_{i\max}$ – максимальное значение характеристики u_i .

Использование (12) или (13) зависит от вкладываемого смысла в процедуру нормирования. Так, например, для характеристики относительной стоимости выгодно за единицу принять наименьшее значение характеристики (13),

а для полной стоимости – максимальное (12). В результате получаем табл. 1 с нормированными значениями характеристик.

На шаге 3 запишем функции принадлежности (6) и определим множества:

$$Q_i = \bigcup_{j=1}^n \mu(z_{ij})x_j, \quad (14)$$

элементы которых определяют степень соответствия j -го варианта требованию i -й характеристики. Для наглядности результаты, полученные на данном шаге, занесем в табл. 2.

Таблица 2

Элементы соответствия j -го варианта требованию i -й характеристики

Характеристика u_i	1	2	3	4	...	23	24
y_1	0/1	0,8/2	0,9/3	0,98/4	...	0,982/23	0,99/24
y_2	0,1/1	0,35/2	0,5/3	0,8/4	...	0,95/23	1/24
y_3	0/1	0/2	0,01/3	0,03/4	...	0,83/23	1/24
y_4	0,15/1	0,26/2	0,54/3	0,74/4	...	0,86/23	1/24

Примечание: составлено авторами.

На шаге 4 определяем пересечение нечетких множеств Q_i и функцию принадлежности полученного пересечения по формулам (7) и (8) соответственно.

На шаге 5, используя выражение минимаксного критерия (9), определим наиболее подходящую с точки зрения указанных характеристик (y_i) конфигурацию вычислительного средства и ЯП:

$$Q = \{0,9/18; 0,85/19. \quad (15)$$

Список источников

1. Крамаров С. О., Храмов В. В. Системно-инженерный подход к исследованиям сложных многомерных систем на основе мягких моделей // Интеллектуальные ресурсы – региональному развитию : сб. науч. тр. Ростов-н/Дону, 2018. Т. 4, № 1. С. 222–228.
2. Липаев В. В. Тестирование программ. М. : Радио и связь, 1986. 296 с.
3. Kramarov S. O., Khramov V. V., Belyaev A. V., Grebenyuk E. V. The Formalization of the Characteristics of Automatic Workplace (AWP) with the Use of Linguistic Variables (LV) // IOP Conference Series: Materials Science and Engineering, Rostov-on-Don, September 11–13, 2020. Rostov-on-Don, 2021. P. 012082. DOI 10.1088/1757-899X/1029/1/012082.
4. Храмов В. В. Особенности использования принципа информационного следа при поиске программных закладок // Вопр. защиты информ. 2001. № 3 (54). С. 39–40.

References

1. Kramarov S. O., Khramov V. V. A Systems Engineering Approach to the Study of Complex Multivariate Systems Based on Soft Models // Intellectualnye resursy – regionalnomu razvitiu : Collection of scientific works. Rostov-on-Don, 2018. Vol. 4, No. 1. P. 222–228. (In Russian).
2. Lipaev V. V. Testirovanie programm. Moscow : Radio i sviaz, 1986. 296 p. (In Russian).
3. Kramarov S. O., Khramov V. V., Belyaev A. V., Grebenyuk E. V. The Formalization of the Characteristics of Automatic Workplace (AWP) with the Use of Linguistic Variables (LV) // IOP Conference Series: Materials Science and Engineering, Rostov-on-Don, September 11–13, 2020. Rostov-on-Don, 2021. P. 012082. DOI 10.1088/1757-899X/1029/1/012082.
4. Khramov V. V. Osobennosti ispolzovaniia printsipa informatsionnogo sleda pri poiske programmnykh zakladok // Information Security Questions. 2001. No. 3 (54). P. 39–40. (In Russian).

5. Акперов И. Г., Крамаров С. О., Лукасевич В. И. и др. Способ формирования цифровой план-схемы объектов сельскохозяйственного назначения и система для его реализации : патент № 2612326 С Российская Федерация, МПК G06F 17/18. № 2015105923 ; заявл. 24.02.2015 ; опубл. 07.03.2017 ; заявитель ЧОУ ВО «Южный университет (ИУБиП)». URL: <https://www.elibrary.ru/item.asp?id=38278561> (дата обращения: 27.06.2022).
6. Крамаров С. О., Митясова О. Ю., Соколов С. В. и др. Криптографическая защита информации. М. : РИОР : ИНФРА-М, 2018. 321 с.
7. Акперов И. Г., Дынник Д. И., Миронова О. А. и др. Системные проблемы надёжности, качества, математического моделирования и инфотелекоммуникационных технологий в инновационных проектах : моногр. М., 2014. 138 с.
8. Линденбаум Т. М., Крамаров С. О., Митясова О. Ю. Оперативное обнаружение и мониторинг технических аварий на железнодорожном транспорте средствами АИКС «САКС» // Актуальные проблемы и перспективы развития транспорта, промышленности и экономики России : сб. науч. тр. Ростов-н/Дону, 2020. С. 131–135.
9. Алекперов И. Д., Горбачева А. А., Фомичев Д. П. и др. Информационная безопасность и защита информации в цифровой экономике: элементы теории и тестовые задания. Ростов-н/Дону : Южный ун-т (ИУБиП), 2020. 114 с.
10. Панфилов А. Н. Об одном решении задачи выбора способов реализации информационных запросов // Изв. высш. учеб. заведений. Северо-Кавказ. регион. Сер. Технич. науки. 2010. № 3 (155). С. 20–22.
11. Ежов А. А., Шумский С. А. Нейрокомпьютинг и его применения в экономике и бизнесе. М. : Национал. открытый ун-т «Интуит», 2016. 306 с.
12. Ковтун О. Г. Сравнительный анализ спутниковых средств дистанционного зондирования земли в интересах проектирования и строительства высокоскоростных железных дорог // Перспективы развития и эффективность функционирования транспортного комплекса Юга России : сб. тр. Междунар. науч.-практ. конф., Ростов-на-Дону, 20–21 ноября 2014 г. Ростов-н/Дону, 2015. С. 97–99.
13. Акишина А. А., Аникьева М. А., Антипина И. В. и др. Национальные интересы и вопросы регионального развития в системе приоритетов международной деятельности российских университетов. М. : РИОР, 2021. 266 с.
14. Kramarov S. O., Mityasova O. Y., Grebenyuk E. V. Formation Method of Delaunay Lattice for Describing and Identifying Objects in Fuzzy Images // Lecture Notes in Networks and Systems. 2022. Vol. 295. P. 305–311. DOI 10.1007/978-3-030-82196-8_23.
15. Грошев А. Р., Крамаров С. О., Пелихов Н. В. и др. Международное сотрудничество российских университетов в императивах регионального развития. М. : РИОР, 2018. 127 с. DOI 10.29039/2006-7.
5. Akperov I. G., Kramarov S. O., Lukasevich V. I. et al. Method of Forming Digital Plan-Scheme of Agricultural Facilities and System for Realising Said Method : patent No. 2612326 S, Russian Federation, MPK G06F 17/18. No. 2015105932 ; Claim 24.02.2015 ; Published 07.03.2017 ; Applicant Southern University (IMBL). URL: <https://www.elibrary.ru/item.asp?id=38278561> (accessed: 27.06.2022). (In Russian).
6. Kramarov S. O., Mityasova O. Yu., Sokolov S. V. et al. Kriptograficheskaya zashchita informatsii. Moscow : RIOR : INFRA-M, 2018. 321 p. (In Russian).
7. Akperov I. G., Dynnik D. I., Mironova O. A. et al. Sistemnye problemy nadezhnosti, kachestva, matematicheskogo modelirovaniia i infotelekomunikatsionnykh tekhnologii v innovatsionnykh proektakh : Monograph. Moscow, 2014. 138 p. (In Russian).
8. Lindenbaum T. M., Kramarov S. O., Mityasova O. Yu. Operativnoe obnaruzhenie i monitoring tekhnicheskikh avarii na zheleznodorozhnom transporte sredstvami AIKS “SAKS” // Aktualnye problemy i perspektivy razvitiia transporta, promyshlennosti i ekonomiki Rossii : Collection of scientific works. Rostov-on-Don, 2020. P. 131–135. (In Russian).
9. Alekperov I. D., Gorbacheva A. A., Fomichev D. P. et al. Informatsionnaia bezopasnost i zashchita informatsii v tsifrovoi ekonomike: elementy teorii i testovye zadaniia. Rostov-on-Don : Southern University (IMBL), 2020. 114 p. (In Russian).
10. Panfilov A. N. Ob odnom reshenii zadachi vybora sposobov realizatsii informatsionnykh zaprosov // Bulletin of Higher Educational Institutions. North Caucasus Region. Technical Sciences. 2010. No. 3 (155). P. 20–22. (In Russian).
11. Ezhov A. A., Shumsky S. A. Neurokompiuting i ego primeneniia v ekonomike i biznese. Moscow : Intuit NOU, 2016. 306 p. (In Russian).
12. Kovtun O. G. Comparative Analysis of Satellite Means of Remote Probing of the Earth for Designing and Construction of High Speed Railways // Development Perspectives and Functioning Effectiveness of Transport Complex of the South of Russia : Proceedings of the International Research-to-Practice Conference, Rostov-on-Don, November 20–21, 2014. Rostov-on-Don, 2015. P. 97–99. (In Russian).
13. Akishina A. A., Anikyeva M. A., Antipina I. V. et al. Natsionalnye interesy i voprosy regionalnogo razvitiia v sisteme prioritetov mezhdunarodnoi deiatelnosti rossiiskikh universitetov. Moscow : RIOR, 2021. 266 p. (In Russian).
14. Kramarov S. O., Mityasova O. Y., Grebenyuk E. V. Formation Method of Delaunay Lattice for Describing and Identifying Objects in Fuzzy Images // Lecture Notes in Networks and Systems. 2022. Vol. 295. P. 305–311. DOI 10.1007/978-3-030-82196-8_23.
15. Groshev A. R., Kramarov S. O., Pelikhov N. V. et al. Mezhdunarodnoe sotrudnichestvo rossiiskikh universitetov v imperativakh regionalnogo razvitiia. Moscow : RIOR, 2018. 127 p. DOI 10.29039/2006-7. (In Russian).
16. Belyaev A., Kramarov S., Mityasova O., Popov O., Khramov V. Research of Tools for Monitoring

16. Belyaev A., Kramarov S., Mityasova O., Popov O., Khramov V. Research of Tools for Monitoring Changes in Natural and Anthropogenic-Transformed Ecosystems // IOP Conference Series: Earth and Environmental Science, Ussuriysk, June 20–21, 2021. Ussuriysk, 2021. P. 022047. DOI 10.1088/1755-1315/937/2/022047.

Changes in Natural and Anthropogenic-Transformed Ecosystems // IOP Conference Series: Earth and Environmental Science, Ussuriysk, June 20–21, 2021. Ussuriysk, 2021. P. 022047. DOI 10.1088/1755-1315/937/2/022047.

Информация об авторах

Г. И. Акперов – руководитель центра маркетинга и развития.

Т. С. Александрова – аспирант.

В. В. Дегтярев – аспирант.

В. В. Храмов – кандидат технических наук, доцент, ведущий научный сотрудник.

Information about the authors

G. I. Akperov – Head of the Marketing and Development Center.

T. S. Aleksandrova – Postgraduate.

V. V. Degtyarev – Postgraduate.

V. V. Khramov – Candidate of Sciences (Engineering), Associate Professor, Leading Researcher.

Научная статья
УДК 794.24:004.42
doi: 10.34822/1999-7604-2022-3-14-24

МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И АНАЛИЗ ДЛЯ ОПРЕДЕЛЕНИЯ СИЛЫ ИГРОКА СЁГИ

Ольга Павловна Бобровская^{1✉}, Светлана Александровна Лысенкова²

^{1,2} Сургутский государственный университет, Сургут, Россия

¹ o-bobrovskaya@mail.ru ✉, <https://orcid.org/0000-0001-7045-9085>

² lysenkova_sa@surgu.ru, <https://orcid.org/0000-0003-1007-7610>

Аннотация. Рассматриваются численные оценки позиций фигур на доске в партии сёги, получаемые от компьютерной программы. С использованием этих оценок предполагается рассчитать силу игрока, для чего требуется собрать данные, рассмотреть и повторить существующие подходы, предложить и реализовать свой подход, сравнить результаты. Проводится повторение подхода Феррейры, в котором на основании распределения выигрыша двух игроков рассчитывается разница их силы. При реализации подхода Ямаситы ищется зависимость между рейтингом Эло и средним ухудшением оценки позиции в результате плохого хода. Предлагается собственный подход к оценке силы игрока. Для этого рассматривается зависимость процента побед, хороших и плохих ходов, разницы оценки реальных и идеальных ходов и среднего улучшения позиции от рейтинга Эло. Предпринимаются попытки кластеризовать партии и ходы.

Ключевые слова: компьютерный движок, кластеризация, рейтинг Эло

Для цитирования: Бобровская О. П., Лысенкова С. А. Математическое моделирование и анализ для определения силы игрока сёги // Вестник кибернетики. 2022. № 3 (47). С. 14–24. DOI 10.34822/1999-7604-2022-3-14-24.

Original article

MATHEMATICAL MODELING AND ANALYSIS TO DETERMINE A SHOGI PLAYER'S SKILL LEVEL

Olga P. Bobrovskaya^{1✉}, Svetlana A. Lysenkova²

^{1,2} Surgut State University, Surgut, Russia

¹ o-bobrovskaya@mail.ru ✉, <https://orcid.org/0000-0001-7045-9085>

² lysenkova_sa@surgu.ru, <https://orcid.org/0000-0003-1007-7610>

Abstract. The study deals with obtaining numerical estimates of shogi pieces' disposition on the board with the help of software. Using these estimates, the authors intend to evaluate a player's skill. That requires collecting data, reviewing and repeating existing approaches, proposing and implementing the authors' approach, and comparing the results. The difference in strength of two players is calculated based on their win distribution via re-simulation of the Ferreira approach. In implementing Yamashita's approach, the relation between Elo rating and the average degradation of the position estimate as a result of a bad move is searched for. The authors' approach is proposed to estimate a player's skill. Hence, the dependence of wins, good and bad moves, the deviation of the estimates of real and ideal moves, and the average improvement of the position from the Elo rating are analyzed. The study attempts to cluster plays and moves.

Keywords: computer engine, clustering, Elo rating

For citation: Bobrovskaya O. P., Lysenkova S. A. Mathematical Modeling and Analysis to Determine a Shogi Player's Skill Level // Proceedings in Cybernetics. 2022. No. 3 (47). P. 14–24. DOI 10.34822/1999-7604-2022-3-14-24.

ВВЕДЕНИЕ

Люди играют в шахматы (различного вида и с разными правилами) более двух тысячелетий. Компьютерные программы, называемые «движками» и способные играть в шахматы, создаются на протяжении полувека. Начиная с 70-х гг. прошлого века, Международная шахматная федерация осуществляет оценку силы игроков, составление их рейтинга и присвоение званий на основе результатов партий с использованием формул Эло. В случае победы рейтинг игрока увеличивается, в случае поражения – уменьшается на величину, зависящую от исходных значений рейтингов противников. Чем больше значение рейтинга игрока – тем он сильнее, чем больше разность рейтингов между двумя игроками – тем больше вероятность того, что сильнееший выиграет партию. Та же рейтинговая система используется в играх шахматного типа (в том числе сёги), также ее модификации используются во многих других играх.

Из-за недостатков, присущих рейтингу Эло, как в предсказании исходов партий, так и в скорости изменения, исследователями предпринимаются попытки найти лучшие альтернативы. Например, в работе [1] авторы разрабатывают нейронную сеть, способную предсказывать результат шахматной партии, основываясь на уже совершенных ходах (достигается точность предсказания 59 % для длины последовательности в 80 ходов).

С 2006 г. развивается идея оценки силы игрока по ходам партии с помощью компьютерных движков в шахматах: используются оценки движка для получения численной оценки позиции [2–5], количества возможных ходов (интерпретирующихся как сложность позиции) [2], определения соответствующих рейтингу Эло параметров чувствительности и согласованности [3], создания марковского классификатора, предсказывающего исход игры [4], подсчета распределения выигрыша (изменение в оценке позиции в результате хода) [5]. После чего полученные значения используются для создания альтернативы рейтингу Эло.

Согласно [6], в сёги до 2014 г. не существовало сравнений игроков на основе анализа игровых записей. Но автор начинает эту

работу и обнаруживает линейную зависимость между средним плохим ходом (изменением оценки позиции после хода игрока в худшую сторону) и рейтингом Эло. В статье [7] предлагается оценивать игроков через два параметра: значение среднего плохого хода (разница между оценкой хода и оценкой наилучшего хода от движка, если она больше 700 очков) и коэффициент совпадения (количество раз, когда лучший ход совпадает с фактическим ходом, деленное на количество ходов в партии), которые, как выяснилось, обязательно коррелируют между собой.

Цель данного исследования – проверить применимость уже существующих подходов и предложить свой подход определения силы игрока на основании его ходов в течение партии. Выдвигается гипотеза о том, что если ходы охарактеризовать численной оценкой улучшения или ухудшения позиции (что можно осуществить с помощью эталона – компьютерного движка), которые они производят, то все ходы, совершаемые в партиях, можно будет разбить на отдельные классы.

Для достижения цели следует выполнить следующие задачи:

1. Собрать и подготовить данные (записи партий).
2. Рассмотреть существующие подходы и разработать свою модель, в соответствии с этим обработать данные движком.
3. Повторить существующие подходы и реализовать свою модель.
4. Сравнить полученные результаты.

МАТЕРИАЛЫ И МЕТОДЫ

Объект исследования – партии сёги. Игра с нулевой суммой, с двумя игроками, которые ходят по очереди. После захвата фигуры противника, в свой ход ее можно выставить на поле как свою. В среднем в партии около 120 ходов, но в используемом наборе данных присутствуют как короткие игры (когда игрок сдался), так и длинные (вплоть до 1000 ходов).

В процессе исследования применялись теоретические методы: анализ и синтез, сравнение, абстрагирование и конкретизация, обобщение, формализация, моделирование; математические: статистические методы, методы визуализации данных (функции

и графики), методы искусственного интеллекта (машинное обучение).

Для численной оценки позиций использовался движок YaneuraOu с функцией оценки Elmo на глубине поиска 17. Все ходы партии отправлялись движку, чтобы получить от него 3 оценки: численную оценку получаемой в результате совершенного хода позиции, потенциально наилучший ход и численную оценку получаемой в его результате позиции. Выдаваемые движком значения оценок нормированы (не превышают по модулю 10000 единиц). Из пошаговой разницы полученных оценок позиций находились как оценки наилучших возможных ходов, так и реально совершенных.

Из собранного набора данных (с сайта 81dojo.com игры за 2011–2012 гг.) были выбраны пары игроков, у которых между собой больше 50 игр. Получилось 27 игроков. Для них, на основании имеющихся игр, посчитан процент выигрышей из всех сыгранных ими игр и проведен подсчет рейтинга Эло (начальное количество баллов 1500, как в работах [6, 7]). Полученный рейтинг Эло оказался слишком далек от рейтингов игроков, рассчитанных онлайн-платформой на основании значительно большего количества их игр. Поэтому для сравнений использовался текущий рейтинг Эло с сайта, несмотря на возможные изменения силы игры за 10 лет.

На основании полученных от компьютерного движка оценок позиций для каждого игрока подсчитан процент хороших, увеличивающих выигрыш, и плохих, ухудшающих позицию ходов, а также среднее количество очков, на которые изменяется оценка позиции в результате хороших и плохих ходов. Также подсчитано среднее значение разницы между оценкой наилучшего возможного хода, предлагаемого движком, и ходом игрока.

Данные, полученные с помощью компьютерного движка, используются для осуществления кластеризации методом k-средних. Для полученных кластеров оценивается сбалансированность. Для определения количе-

ства кластеров используются метод локтя, метод силуэта и Calinski-Harabasz index.

Подход Феррейры (Ferreira [4]). Расчетный рейтинг, вычисляемый Феррейрой, нельзя назвать точным рейтингом (каким является рейтинг Эло), поэтому автор интерпретирует его как силу игрока.

Для реализации подхода, описанного в статье [4], были использованы результаты игр двадцати семи пар игроков, у которых между собой больше 50 игр, и их рейтинг Эло. Для каждого игрока взята его последняя игра (предположительно, являющаяся наиболее показательной и соответствующей имеющемуся рейтингу Эло). На рис. 1 показаны гистограммы выигрыша за ход (разность между оценкой позиции до и после хода) для двух игроков. Значения частот на этих гистограммах были нормализованы, что означает, что сумма высот всех столбцов равна 1,0.

Приведенные на графиках значения интерпретируются как распределение выигрыша.

Далее, для игроков считается корреляция их распределений выигрыша по следующей формуле:

$$f_{X-Y}[n] = (f_Y \times f_X) \triangleq \sum_{m=-\infty}^{\infty} f_Y[m] \times f_X[n+m], \quad (1)$$

где X и Y – две дискретные случайные величины с распределениями вероятностей f_X и f_Y соответственно; тогда распределение вероятности $X - Y$ (победы первого игрока) определяется взаимной корреляцией.

Определяется вероятностная оценка выигрыша партии первым игроком:

$$p_{XY} \triangleq 0,5 \times P(X = Y) + 1 \times P(X > Y) = 0,5 \times f_{X-Y}[0] + \sum_{n=1}^{\infty} f_{X-Y}[n]. \quad (2)$$

Разница в рейтинговых баллах, соответствующая процентному ожиданию p , определяется по формуле:

$$d = \Phi^{-1}(p) \times 200 \times \sqrt{2}, \quad (3)$$

где Φ^{-1} – обратная функция плотности нормального распределения.

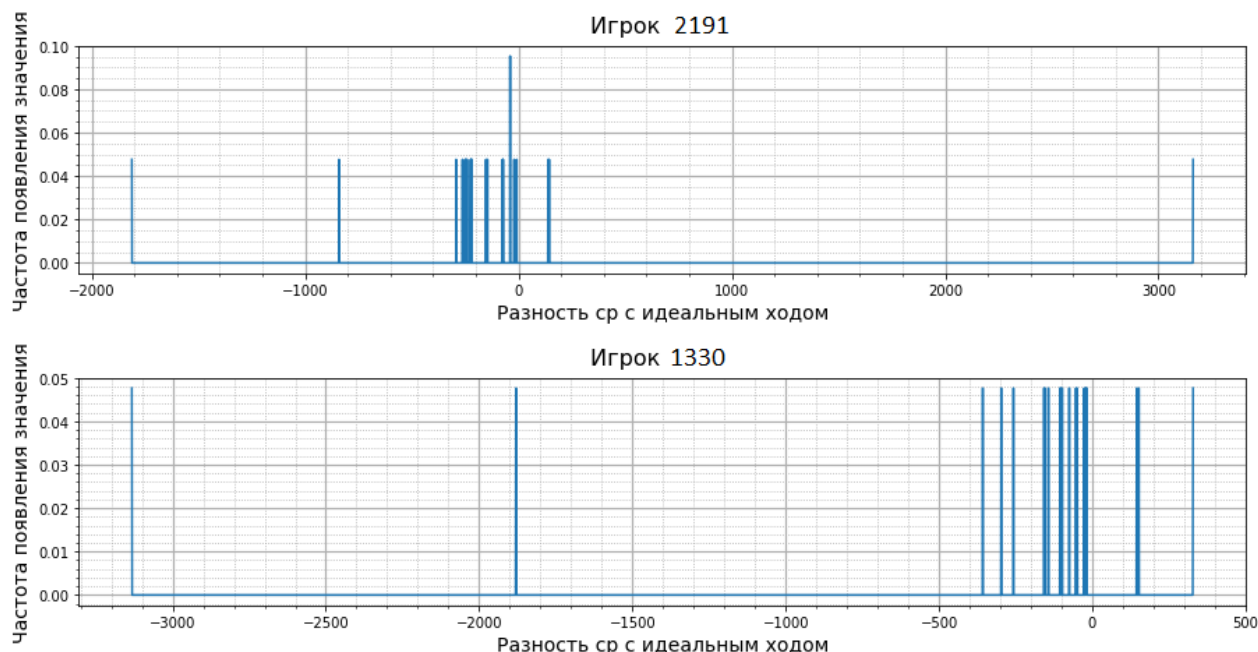


Рис. 1. Нормализованная гистограмма выигрыша для обоих игроков (строка № 23 из табл. 1)

Примечание: составлено авторами.

Такие расчеты были проведены для всех 27 игроков, которые могли оказаться как первым, так и вторым игроком в рассматриваемом наборе игр. Результаты представлены в табл. 1.

Таблица 1

Значения для крайних игр каждого рассматриваемого игрока

№	Индекс первого игрока	Индекс второго игрока	Рассчитанная разность рейтинга	Разность текущих рейтингов Эло
1	1	159	36,42	-33,6
2	1379	7	18,71	-264,53
3	9	407	-20,16	182,01
4	66	29	25,81	-490,1
5	66	187	51,92	-17,9
6	1330	204	98,6	92,9
7	1379	304	45,89	-238,7
8	9	407	-20,16	182,01
9	1330	780	67,09	-189,5
10	1330	789	1,18	-50,9
11	822	799	43,91	121,9
12	1330	884	27,33	-475,1
13	1379	1012	-33,58	255,03
14	1198	1377	-103,34	-651,8
15	1379	1230	26,14	-198,13
16	2191	1329	-36,72	807,45
17	2032	1377	92,44	-1153,42
18	2032	1379	-40,56	-322,42
19	1379	1432	146,63	-72,48
20	1438	1329	-44,61	635,57
21	1785	1379	32,65	16,3
22	2032	1379	-40,56	-322,42
23	2191	1330	-37,89	147,45

Примечание: составлено авторами.

Как видно из табл. 1, значения разности рейтингов Эло и рассчитанных разностей (по Феррейре) совпадают примерно в двух случаях из двадцати. Чтобы убедиться в нали-

чий или отсутствии связи между ними, был построен график по всем играм рассматриваемых игроков (рис. 2).

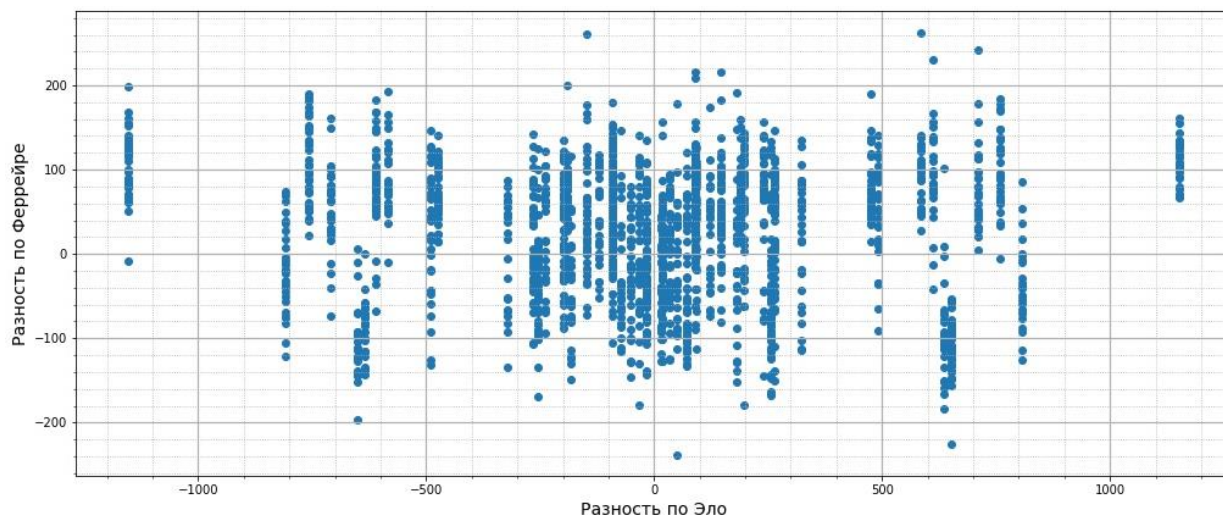


Рис. 2. Значения разности по Эло и по Феррейре для всех игр рассматриваемых игроков
 Примечание: составлено авторами.

Значения по оси X фиксированы для игроков, а значения по оси Y рассчитываются для каждой партии, и, как видно по графику, сгруппированы в диапазоне около 200 единиц. Но как порядок величин, так и знак сильно отличаются: в то время как разница рейтинга Эло у игроков превышает 1000 ед. по модулю, разница по Феррейре не выходит за границы 300 ед. по модулю; значения симметричны относительно оси X как для положи-

тельной разности рейтинга Эло, так и для отрицательной, хотя можно было бы ожидать, что график будет сгруппирован в первой и третьей четвертях.

Подход Ямаситы. Попробуем повторить внешний вид графика (рис. 3) из статьи [6], на котором по оси X откладываются значения рейтинга Эло, а по оси Y – средний плохой ход.

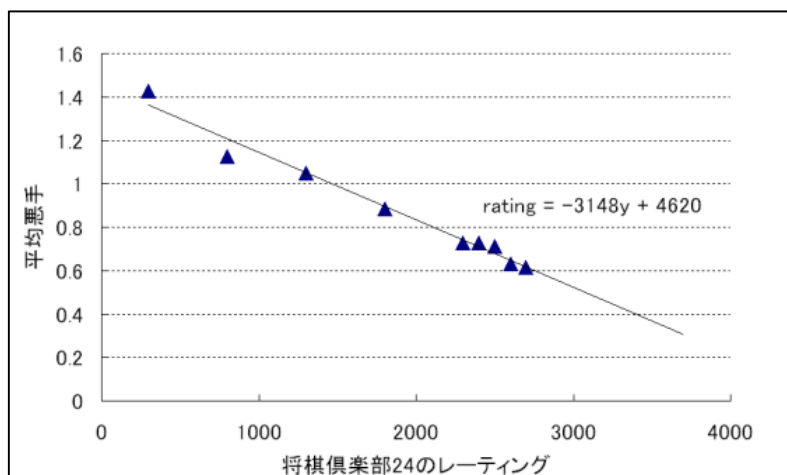


Рис. 3. Линейная зависимость значения рейтинга Эло от среднего плохого хода, полученная методом наименьших квадратов
 Примечание: составлено по [6].

Значения рейтинга Эло были отсортированы по возрастанию для 27 игроков с количеством игр между собой больше 50 шт. Слева направо, сверху вниз на рис. 4 изображены графики: процент побед от количества сыгранных игр (рис. 4а); процент хороших ходов, в результате которых увеличивалась оценка позиции (рис. 4б); процент плохих ходов, в результате которых уменьшалась оценка

позиции (рис. 4в); средняя разница между реальным ходом и лучшим рекомендуемым ходом от движка (рис. 4г); среднее увеличение оценки позиции в результате хорошего хода (рис. 4д); среднее уменьшение оценки позиции в результате плохого хода (рис. 4е). Похожим на убывающую прямую должен быть шестой график.

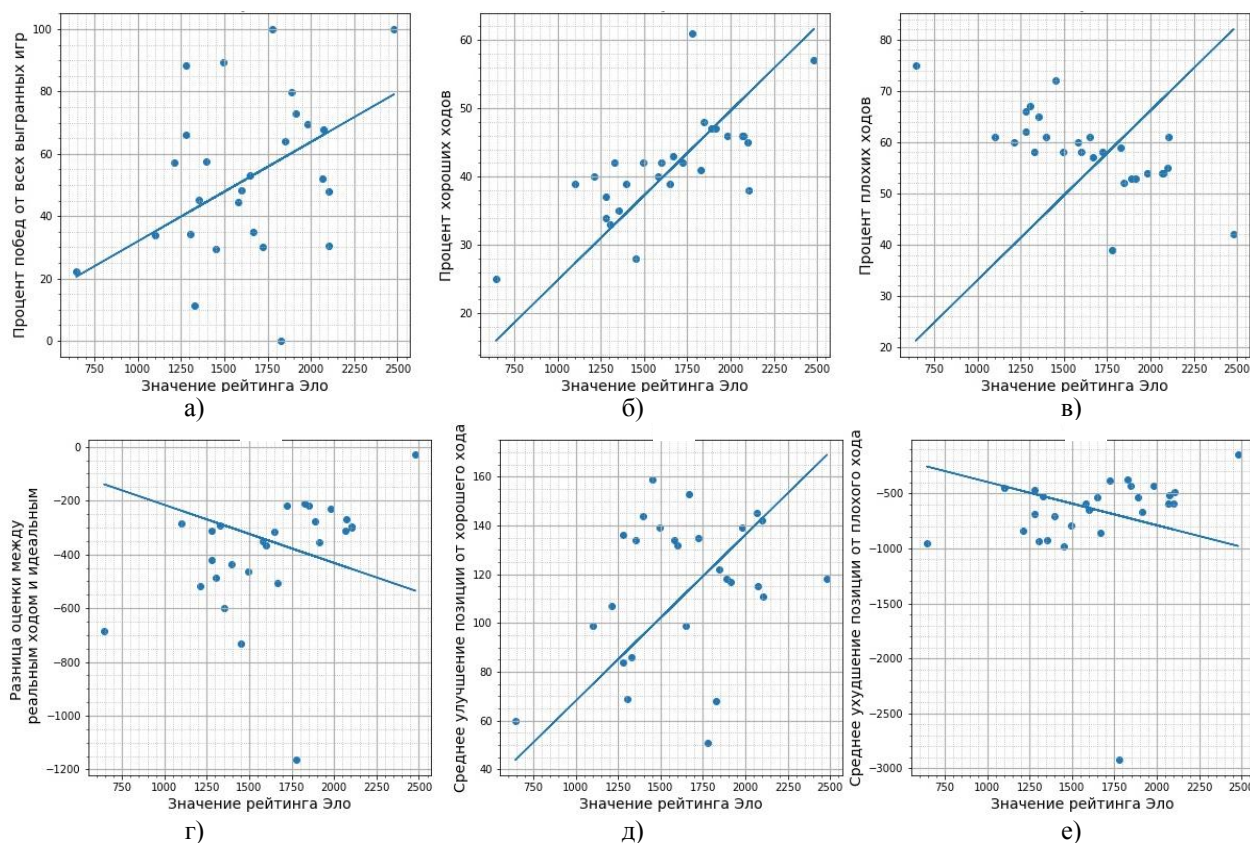


Рис. 4. Сравнение оценок игроков и их рейтинга Эло

Примечание: составлено авторами.

Прямые построены методом линейной регрессии. Для каждого графика был посчитан коэффициент R^2 (коэффициент детерминации). Для графиков под буквами: а, г, д, е, — значение коэффициента меньше 0,20, что говорит о большой разбросанности значений. Для графиков процентов хороших и плохих ходов (рис. 4б, 4в) значение коэффициента равно 0,53 и 0,54 соответственно, и, как видно по самому графику, точки лежат довольно близко к построенной прямой.

Кластеризация партий. Здесь используется только оценка позиции в результате

реального хода, на основе которой вычисляется ее изменение (разность последовательных значений). Последовательность значений изменения позиции для каждой партии представляет собой временной ряд. Для получения параметров этих рядов и последующей кластеризации был использован экстрактор признаков [8], обеспечивающий систематическое извлечение признаков временных рядов (комбинирующий установленные алгоритмы из статистики, анализа временных рядов, обработки сигналов и нелинейной динамики с надежным алгоритмом

выбора признаков). После разбиения на 4 кластера методом k-средних получены кла-

Таблица 2

Кластеризация партий

№ кластера	Количество партий
2	23846
0	9960
1	4219
3	1708

Примечание: составлено авторами.

Количество партий в кластерах значительно различается (на порядок), что, возможно, свидетельствует о принадлежности всех рассматриваемых партий к одному кластеру или крайне несбалансированных данных. Но, так как тут рассматривались все имеющиеся в нашем распоряжении партии (около 47 тысяч), можно предположить, что на основе данной кластеризации осуществлять последующую классификацию партий будет невозможно.

Так как в партии участвуют два игрока, было выдвинуто предположение, что удастся лучше разделить на кластеры не целые игры,

а половины игр, за которые отвечает каждый игрок. Игра разделена пополам – для каждого игрока его ходы: из последовательных разниц оценок позиции в одной партии строится два временных ряда: для первого и второго игроков (изменения численной оценки преимущества за 1-й, 3-й, 5-й и т. д. ходы; 2-й, 4-й, 6-й и далее ходы соответственно). Результат кластеризации представлен в табл. 3.

Результат стал чуть лучше, но различия в размерах кластеров все еще велики. Это позволяет утверждать, что кластеризовать партии не удалось.

Таблица 3

Кластеризация половин партий

№ кластера	Количество партий
3	40356
1	17431
2	12085
0	9594

Примечание: составлено авторами.

Кластеризация ходов. Кластеризация ходов осуществлялась по следующим параметрам: изменению оценки позиции, совпадению хода с наилучшим ходом от движка и разнице между ними.

В случае деления на 4 кластера (табл. 4) размеры кластеров получились сбалансированными (все числа одного порядка).

Таблица 4

Кластеризация ходов

№ кластера	Количество ходов
3	900244
1	889880
0	836716
2	783252

Примечание: составлено авторами.

Так как измерений немного, график с параметрами ходов представлен на рис. 5, где каждая точка – это ход. Темнота точек (от черного до светло-оранжевого) отражает номер хода в партии. По оси X отражено изменение оценки позиции, произошедшее в результате хода – выигрыш или проигрыш очков. По оси Y –

упущенный выигрыш (или проигрыш), который был бы возможен в случае выбора игроком наилучшего хода. Таким образом, точки, находящиеся под прямой $y = x$, являются ходами, в которых игрок сходил лучше рекомендуемого движком хода.

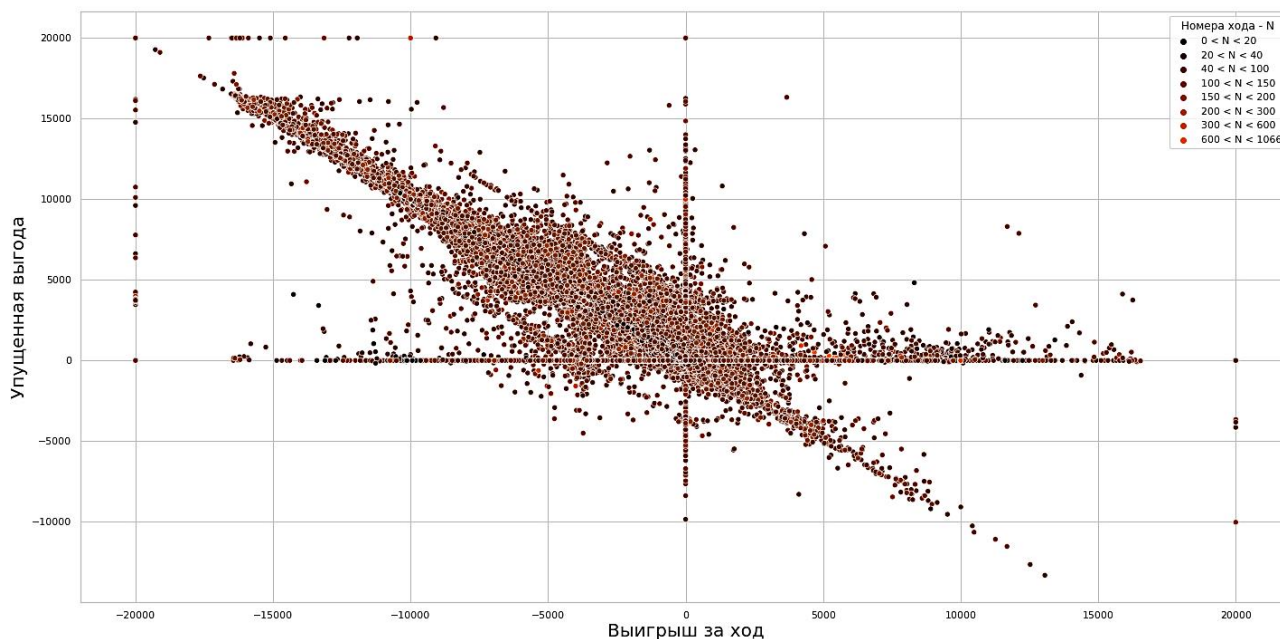


Рис. 5. Ходы

Примечание: составлено авторами.

Большая часть ходов находится выше прямой $y = x$ и в отрицательных значениях x , что говорит о том, что игроки преимущественно ходят хуже, чем могли бы, и вместо возможного в результате хода улучшения позиции ухудшают ее.

Для определения наилучшего количества кластеров были использованы метод локтя, метод силуэта и быстрее считающийся Calinski-Harabasz index, представленные ниже на графиках (рис. 6).

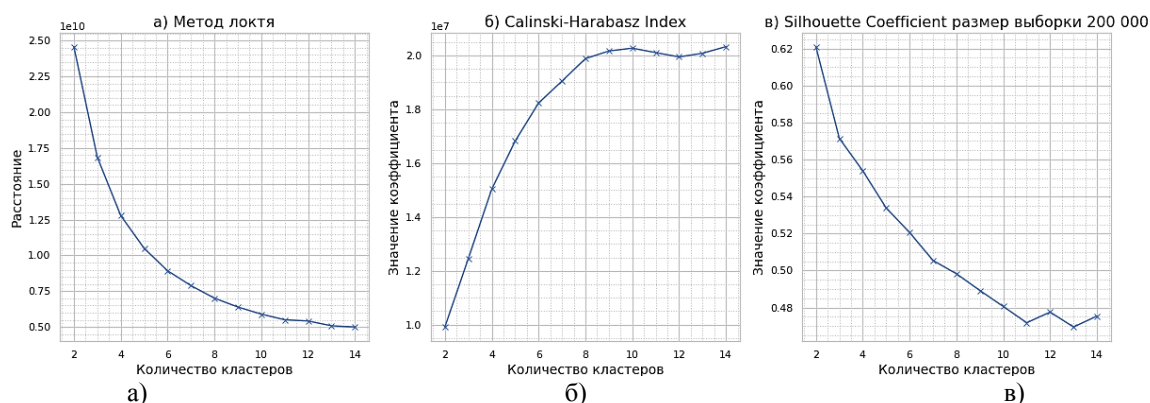


Рис. 6. Определение количества кластеров:

а – по методу локтя; б – Calinski-Harabasz index; в – по методу силуэта

Примечание: составлено авторами.

Метод локтя (Elbow rule) – визуальный метод определения наилучшего количества кластеров путем сравнения разницы суммы квадратов ошибок (SSE) каждого кластера; самая большая разница, образующая угол изгиба, показывает лучший номер кластера [9].

Calinski-Harabasz index известен как критерий отношения дисперсии. Индекс *Калински – Харабаз* определяется как отношение сумм межкластерной дисперсии и внутрикластерной дисперсии [10]:

$$CH = \frac{(T - W_K) / (K - 1)}{W_K / (N - K)}, \quad (4)$$

где $T = \sum_{i=1}^N \sum_{v \in V} (y_{iv} - \bar{y}_v)^2$ – разброс данных;

$W_K = W(S, C) = \sum_{k=1}^K \sum_{i \in S_k} d(y_i, c_k)$ – минимизируемое внутрикластерное расстояние между экземплярами кластера и центроидами. Наилучшим количеством кластеров считается то, у которого значение индекса наибольшее.

Метод силуэта [11] – метод для оценки потенциально оптимального числа кластеров, который использует среднее расстояние между одной точкой данных и другими в том же кластере и среднее расстояние между различными кластерами для оценки результата кластеризации. Метрика оценки этого метода называется коэффициентом

силуэта (S) и определяется как $\frac{(b-a)}{\max(a, b)}$,

где a и b представляют собой среднее расстояние внутри кластера и среднее расстояние до ближайшего кластера соответственно. Лучшее значение равно 1, а худшее значение равно –1. Значения около 0 указывают на перекрывающиеся кластеры. Отрицательные значения обычно указывают на то, что образец был отнесен к неправильному кластеру, так как другой кластер более похож.

Метод локтя говорит о том, что кластеров от четырех до шести (плавное место перегиба графика), в то время как Calinski-Harabasz index и коэффициент силуэта не имеют четкого максимума, из чего можно сделать вывод, что произвести деление на кластеры методом k -средних в данном случае не удастся.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В процессе исследования было обнаружено, что попытка рассчитать рейтинг Эло на малом количестве игр для малого количества игроков приводит к неадекватным результатам. Что подтверждает указанный вначале недостаток рейтинга Эло, требующего большого количества игр, и актуальность исследования, нацеленного на поиск альтернативы.

Полученные результаты не удается легко сопоставить с рейтингом Эло, на который мы опираемся в данном исследовании.

Подход Феррейры. Наблюдаемое как в табл. 1, так и на рис. 2 несоответствие знаков и масштабов разниц рейтингов Эло и Феррейры может иметь следующие возможные объяснения:

- неточности со стороны компьютерного движка, поскольку от запуска к запуску оценка одного и того же хода может меняться, чего, возможно, не наблюдалось у движка, используемого Феррейрой;

- неучтенные особенности шахмат и сёги: разность в средней продолжительности партии, характер этапов партии (дебют, миттельшпиль, эндшпиль) и переходов между ними, которые могут оказывать влияние на оценку рассматриваемых ходов;

- неточность рейтинга Эло как по временным промежуткам, так и по качеству для игроков-любителей;

- отмеченная автором [5] необходимость учитывать сложность позиций, возникающих за доской в ходе игры: «В конце концов, в некоторых позициях допустимых ходов всего несколько и ошибиться практически невозможно, а в других позициях сделать правильный выбор из множества возможных ходов становится очень сложно»;

- результаты в работе Феррейры по рассмотренной оценке разности рейтингов также не совпадали с настоящими рейтингами Эло игроков. Возможно, следовало продолжить анализ до получения финальных значений рейтинга по Феррейре, но отказ от этого мотивирован возможностью не использовать в анализе гипотезу о распределении движка с частотой 1 на значении выигрыша 0 (по наблюдениям, движок иногда выбирает ходы

не с самой большой оценкой) и последующем смещении результатов относительно известных рейтингов Эло.

Подход Ямаситы. В качестве положительного результата можно отметить, что точки значений на рис. 4б и рис. 4в расположены довольно близко к построенным прямым. Поэтому стоит рассмотреть идею использования процента хороших и плохих ходов, оцененных с помощью компьютерного движка, в качестве оценки силы игроков сёги.

Несоответствие графика на рис. 4е оригинальному графику (рис. 3) может быть объяснено неточностью повторения оригинального подхода, а именно:

- не были удалены первые 40 ходов построения крепости, не были убраны ходы, оценка которых превышала некоторый порог, чтобы исключить влияние форсированного мата (цуме);

- не исключены ходы, сумма абсолютных различий оценок которых при каждом переходе к лучшему ходу в ходе итерационного углубления превышает 1000 ед.

Кластеризация партий. Попытка кластеризовать партии как временные ряды не удалась. Предположение о том, что партии могут составлять классы, например по разности рейтингов противников или сочетанию стилей их игры, не нашло подтверждения. Если подобные классы и присутствуют в рассматриваемых данных, то в слишком малом количестве.

Кластеризация ходов. Как видно по графикам на рис. 5 и 6, деление на кластеры методом k-средних для рассматриваемых данных невозможно. Но ходы в отмеченных на графике (рис. 5) координатах можно разделить на области с помощью прямых, параллельных прямой $y = -x$. Полученные классы будут характеризоваться разной успешностью хода в сравнении с наилучшей оценкой движка.

Чтобы выделить из совершаемых игроками ходов различные классы методом k-средних, потребуется добавление большего количества параметров, характеризующих ход (промежуточные оценки, проводимые движком: статическая оценка позиции, все перерасчеты, проводимые в процессе поиска в глубину). Эти параметры можно либо достать из существующего движка с открытым кодом, либо разработать свой движок так, чтобы он выдавал эти оценки.

ЗАКЛЮЧЕНИЕ

Были подготовлены данные в виде записей партий, которые обрабатывались компьютерным движком для получения численных оценок позиций фигур на доске.

Рассмотрено несколько существующих подходов для определения силы игрока на основе их ходов, два подхода (Феррейры и Ямаситы) повторены. Достигнуть тех же результатов, что и у авторов подходов, не удалось.

Предложенные авторские подходы к оценке силы игрока (использование процента побед, разницы оценки между реальным и идеальным ходами и среднего улучшения позиции от хорошего хода, а также кластеризация партий и их половин) не дали удовлетворительных результатов.

Использование же процента хороших и плохих ходов игрока и кластеризация ходов являются перспективными. Поставленная в начале исследования цель – проверить применимость существующих и предложить свой подход для определения силы игрока на основании его ходов с помощью компьютерного движка – была достигнута.

Планируется продолжать работу над определением классов ходов, возможно, сопоставлением набора таких классов и рейтинга Эло игрока.

Список источников

1. Drezewski R., Wątor G. Chess as Sequential Data in a Chess Match Outcome Prediction Using Deep Learning with Various Chessboard Representations // Procedia Computer Science. 2021. Vol. 192. P. 1760–1769.

References

1. Drezewski R., Wątor G. Chess as Sequential Data in a Chess Match Outcome Prediction Using Deep Learning with Various Chessboard Representations // Procedia Computer Science. 2021. Vol. 192. P. 1760–1769.

2. Guid M., Bratko I. Computer Analysis of World Chess Champions // ICGA Journal. 2006. Vol. 29, No. 2. P. 65–73.
3. Regan K. W., Haworth G. M. Intrinsic Chess Ratings // Proceedings of the AAAI Conference on Artificial Intelligence. 2011. Vol. 25, No. 1. P. 834–839.
4. Alliot J.-M. Who is the Master? // ICGA Journal. 2017. Vol. 39, No. 1. P. 3–43.
5. Ferreira D. R. Determining the Strength of Chess Players Based on Actual Play // ICGA Journal. 2012. Vol. 35, No. 1. P. 3–19.
6. Yamashita H. Pro Shogi Players Rating and Game Record Analysis // Proceedings of the Game Programming Workshop. 2014. P. 9–16.
7. Takatsu K., Takada H., Hirata T. How to Estimate the Rating of the Youngest Professional Shogi Player Souta Fujii // Memoir of Graduate School of Engineering, University of Fukui. 2019. Vol. 67. P. 1–9.
8. Christ M., Braun N., Neuffer J., Kempa-Liehr A. W. Time Series Feature Extraction on Basis of Scalable Hypothesis Tests (TsFresh – A Python Package) // Neurocomputing. 2018. Vol. 307. P. 72–77. URL: <https://www.sciencedirect.com/science/article/pii/S0925231218304843> (дата обращения: 23.05.2022).
9. Umargono E., Suseno J. E., Gunawan S. K. V. K-Means Clustering Optimization Using the Elbow Method and Early Centroid Determination Based on Mean and Median Formula // Proceedings of the 2nd International Seminar on Science and Technology (ISSTEC 2019). 2020. P. 121–129. URL: https://www.researchgate.net/publication/346349075_K-Means_Clustering_Optimization_Using_the_Elbow_Method_and_Early_Centroid_Determination_Based_on_Mean_and_Median_Formula (дата обращения: 29.05.2022).
10. De Amorim R. C., Hennig C. Recovering the Number of Clusters in Data Sets with Noise Features Using Feature Rescaling Factors // Information Sciences. 2015. Vol. 324. P. 126–145. URL: <https://arxiv.org/pdf/1602.06989.pdf> (дата обращения: 29.05.2022).
11. Shi C., Wei B., Wei S. et al. A Quantitative Discriminant Method of Elbow Point for the Optimal Number of Clusters in Clustering Algorithm // J Wireless Com Network. 2021. P. 31. DOI 10.1186/s13638-021-01910-w (дата обращения: 29.05.2022).
2. Guid M., Bratko I. Computer Analysis of World Chess Champions // ICGA Journal. 2006. Vol. 29, No. 2. P. 65–73.
3. Regan K. W., Haworth G. M. Intrinsic Chess Ratings // Proceedings of the AAAI Conference on Artificial Intelligence. 2011. Vol. 25, No. 1. P. 834–839.
4. Alliot J.-M. Who is the Master? // ICGA Journal. 2017. Vol. 39, No. 1. P. 3–43.
5. Ferreira D. R. Determining the Strength of Chess Players Based on Actual Play // ICGA Journal. 2012. Vol. 35, No. 1. P. 3–19.
6. Yamashita H. Pro Shogi Players Rating and Game Record Analysis // Proceedings of the Game Programming Workshop. 2014. P. 9–16. (In Japanese).
7. Takatsu K., Takada H., Hirata T. How to Estimate the Rating of the Youngest Professional Shogi Player Souta Fujii // Memoir of Graduate School of Engineering, University of Fukui. 2019. Vol. 67. P. 1–9. (In Japanese).
8. Christ M., Braun N., Neuffer J., Kempa-Liehr A. W. Time Series Feature Extraction on Basis of Scalable Hypothesis Tests (TsFresh – A Python Package) // Neurocomputing. 2018. Vol. 307. P. 72–77. URL: <https://www.sciencedirect.com/science/article/pii/S0925231218304843> (accessed: 23.05.2022).
9. Umargono E., Suseno J. E., Gunawan S. K. V. K-Means Clustering Optimization Using the Elbow Method and Early Centroid Determination Based on Mean and Median Formula // Proceedings of the 2nd International Seminar on Science and Technology (ISSTEC 2019). 2020. P. 121–129. URL: https://www.researchgate.net/publication/346349075_K-Means_Clustering_Optimization_Using_the_Elbow_Method_and_Early_Centroid_Determination_Based_on_Mean_and_Median_Formula (accessed: 29.05.2022).
10. De Amorim R. C., Hennig C. Recovering the Number of Clusters in Data Sets with Noise Features Using Feature Rescaling Factors // Information Sciences. 2015. Vol. 324. P. 126–145. URL: <https://arxiv.org/pdf/1602.06989.pdf> (accessed: 29.05.2022).
11. Shi C., Wei B., Wei S. et al. A Quantitative Discriminant Method of Elbow Point for the Optimal Number of Clusters in Clustering Algorithm // J Wireless Com Network. 2021. P. 31. DOI 10.1186/s13638-021-01910-w (accessed: 29.05.2022).

Информация об авторах

О. П. Бобровская – магистрант.

С. А. Лысенкова – кандидат физико-математических наук.

Information about the authors

O. P. Bobrovskaya – Master's Degree Student.

S. A. Lysenkova – Candidate of Sciences (Physics and Mathematics).

Научная статья

УДК 519.876.5:336.7

doi: 10.34822/1999-7604-2022-3-25-31

ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ СИСТЕМЫ МАССОВОГО ОБСЛУЖИВАНИЯ КЛИЕНТОВ В БАНКЕ

Наталья Сергеевна Веремчук

Сибирский государственный автомобильно-дорожный университет, Омск, Россия

n-veremchuk@rambler.ru, <https://orcid.org/0000-0002-2709-9755>

Аннотация. Описывается разработка и исследование имитационной модели для системы обслуживания клиентов в отделении банка. Модель основана на агентном и дискретно-событийном моделировании. Приведены результаты экспериментов, целью которых было исследование функционирования системы обслуживания клиентов банка, изучение статистики очередей, среднего времени обслуживания при различных режимах работы. Полученные результаты могут применяться для проведения экономического планирования и принятия решений по улучшению работы банка. Материалы статьи также могут использоваться для формирования необходимых профессиональных компетенций у обучающихся в процессе освоения основных образовательных программ бакалавриата и магистратуры.

Ключевые слова: дискретно-событийное моделирование, имитационный подход, система обслуживания, учебный процесс

Для цитирования: Веремчук Н. С. Экспериментальное исследование системы массового обслуживания клиентов в банке // Вестник кибернетики. 2022. № 3 (47). С. 25–31. DOI 10.34822/1999-7604-2022-3-25-31.

Original article

EXPERIMENTAL STUDY OF THE MASS CUSTOMER SERVICE SYSTEM IN THE BANK

Natalya S. Veremchuk

Siberian State Automobile and Highway University, Omsk, Russia

n-veremchuk@rambler.ru, <https://orcid.org/0000-0002-2709-9755>

Abstract. A simulation model of a customer service system in the bank based on agent and discrete-event modeling is developed, analyzed and described. The study presents the results of experiments aimed at analyzing the performance of the customer service system in the bank. The statistics of queues and average service time at different working hours are studied. The results obtained can be used for economic planning and decision-making to improve the bank's performance. The materials discussed in the article can also be used to provide students with the necessary professional skills when studying for bachelor's and master's degrees.

Keywords: discrete-event modeling, simulation approach, service system, educational process

For citation: Veremchuk N. S. Experimental Study of the Mass Customer Service System in the Bank // Proceedings in Cybernetics. 2022. No. 3 (47). P. 25–31. DOI 10.34822/1999-7604-2022-3-25-31.

ВВЕДЕНИЕ

Одним из инструментов анализа, управления и прогнозирования является имитационное моделирование [1–3]. С применением имитационного подхода осуществляется поиск

эффективных решений, в том числе и в науке, а также формируется представление о функционировании сложных систем в режиме реального времени [4–6]. Как следствие, имитационное моделирование широко применя-

ется на практике в процессах решения задач из различных сфер деятельности [7–9]. Имитационные эксперименты можно проводить, задав различные значения параметров, описывающих практические ситуации.

В процессе построения имитационных моделей применяются компьютерные технологии [10]. Компьютерное моделирование используется в случаях, когда проведение экспериментов на реальной системе сложно или практически невозможно. Причинами таких ситуаций могут служить, например, стоимость или длительность моделируемых процессов.

Применение инструментария имитационного моделирования к процессам анализа, прогнозирования и управления деятельностью банка является перспективным направлением, поскольку позволяет в режиме реального времени, с учетом визуализации внутренних процессов в системе, выявить наиболее значимые проблемы и принять решения по их устранению. Для повышения эффективности их деятельности необходимо проводить имитационные эксперименты с учетом конкретных параметров организации.

В работе описаны разработка и исследование имитационной модели системы обслуживания посетителей банка. Целью исследования является анализ и оценка эффективности процессов обслуживания клиентов. Для реализации модели использовалась программная среда AnyLogic PLE. По результатам моделирования сделаны выводы об особенностях функционирования системы в различных режимах работы. Описаны эксперименты, целью которых было изучение статистики очередей, среднего времени обслуживания, получение рекомендаций по улучшению эффективности оказания услуг и экспериментальное доказательство этого факта. Полученные результаты могут применяться для принятия решений по улучшению работы банковского отделения, а также при формировании необходимых профессиональных компетенций обучающихся в процессе изучения образовательных программ.

МАТЕРИАЛЫ И МЕТОДЫ

Различают три основных подхода при создании имитационных моделей. К ним относят

системную динамику, агентное и дискретно-событийное моделирование [6–7].

Наиболее высокий уровень абстракции осуществляется в системной динамике. Суть заключается в изучении взаимодействия различных процессов или явлений. При использовании системной динамики применяется информация только о глобальных зависимостях [3]. Дискретно-событийное моделирование используется при имитации процессов обслуживания, создания продукции [7]. В таких случаях система описывается как процесс. Агентные модели применяются в случаях, когда необходимо показать деятельность объектов [6]. Оно предполагает наличие индивидуальных объектов.

На практике часто используется многоподходное имитационное моделирование, в котором различные методы [1, 2, 4, 5] комбинируются таким образом, что достоинства одних подходов компенсируют недостатки других. Благодаря объединению методов создаются модели без привлечения дополнительных средств. Можно отметить следующие плюсы от использования многоподходного моделирования:

- в процессе построения модели всегда выбираются некоторые допустимые уровни абстракции с учетом поставленных целей и задач;
- чаще всего задачи реального мира очень сложны, поэтому их при моделировании разбивают на составные части. Каждую часть описывают, применяя разные методы. Сочетание методов моделирования одновременно придает гибкость, необходимую для решения задачи. Например, можно совместить задачи производства, распределения и реализации продукции в одной модели.

На рынке представлен обширный набор современных систем моделирования, которые поддерживают графические оболочки, анимацию, объектно-ориентированное программирование и многие другие технологии. Один из российских программных продуктов для имитационного моделирования – это AnyLogic PLE (Personal Learning Edition), бесплатная версия, применяемая в образовательных целях [10]. Он поддерживает проектирование, разработку модели. В режиме реального времени осуществляется выпол-

нение экспериментов, оптимизация параметров в модели. Для удобства можно использовать графические элементы, диаграммы состояний, события и т. д. Кроме того, в комплект входит набор уже готовых типовых библиотек, которые можно применять при разработке моделей из соответствующих областей деятельности.

Опишем процесс построения имитационной модели системы обслуживания клиентов

банка в среде AnyLogic PLE. На рис. 1 изображена концептуальная модель, или схема, банковского отделения, где: а – это кассы работников банка (клерков), d – банкомат. В модели два типа агентов: рабочие банка (b) и посетители (c). Придя в офис, клиент выбирает один из способов обслуживания и встает в очередь e или f, получает необходимые услуги и покидает банк.

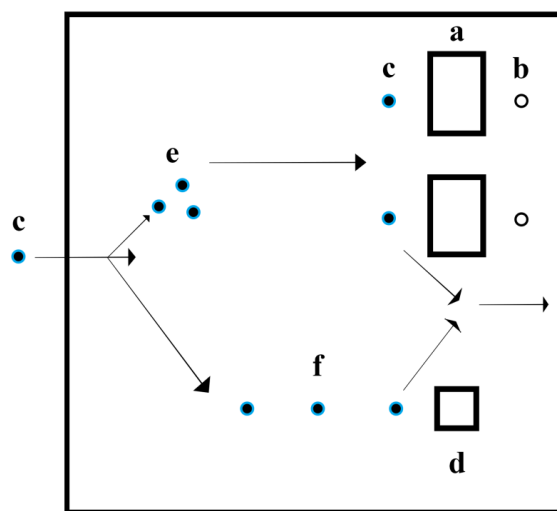


Рис. 1. Концептуальная модель банковского отделения

Примечание: составлено автором.

В системе есть банкомат, две кассы обслуживания; моделируется входящий поток посетителей с определенной интенсивностью. Считается, что посетители для обслуживания с равной вероятностью могут пойти как к банкомату, так и к кассам работников банка. В случаях, когда превышен лимит ожидания, посетители покидают банк, не дождавшись обслуживания. Данные о числе обслуженных и не обслуженных посетителей отражаются на диаграмме. Осуществляется сбор информации

о времени, которое посетители провели в банке. Построен график, отражающий длину очереди. В качестве единиц модельного времени использованы минуты.

Для имитации процессов обслуживания посетителей в банкомате построена диаграмма процесса, изображенная на рис. 2. Для этого на рабочую область перенесены объекты source, queue, delay и sink из библиотеки моделирования процессов.

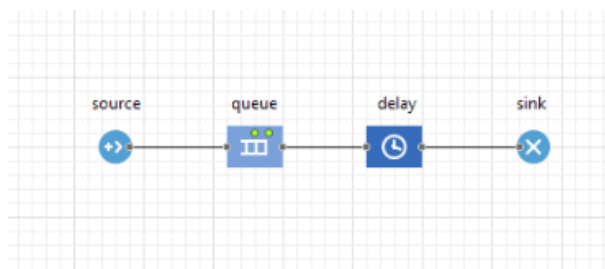


Рис. 2. Диаграмма процесса обслуживания посетителей в банкомате

Примечание: составлено автором.

Появление клиентов в банке моделируется с помощью объекта source, в свойствах которого выставлена интенсивность прибытия 0,67 посетителей (заявок) в минуту. Очередь из ожидающих обслуживания клиентов моделируется блоком queue, в свойствах которого указано максимальное количество мест в очереди 15. Процесс обслуживания в банкомате имитируется объектом delay (ATM), который имеет вероятностные характеристики с треугольным распределением triangular (0,8, 1,5, 3,5). Конец блок-схемы обозначен как sink.

В модель (рис. 3) добавлен блок service из той же библиотеки. С его применением имитируется выбор вариантов обслуживания посетителями у клерков или в банкомате. Этот

процесс осуществляется с одинаковой вероятностью. Время обслуживания у клерков имеет треугольное распределение с параметрами triangular (3, 5, 7) в минутах.

Чтобы осуществить возможность выхода из очереди, у объектов service и queue установлен тайм-аут на 40 мин. Для большей наглядности добавлено 3D-окно и сцена.

Сбор статистических данных по обслуживанию осуществляется с применением различных диаграмм из библиотеки статистики. Столбиковая диаграмма отражает среднюю занятость банкомата и длину очереди. Круговая диаграмма предназначена для вывода информации о проценте обслуженных и необслуженных посетителей. На рис. 3 представлен результат запуска модели.

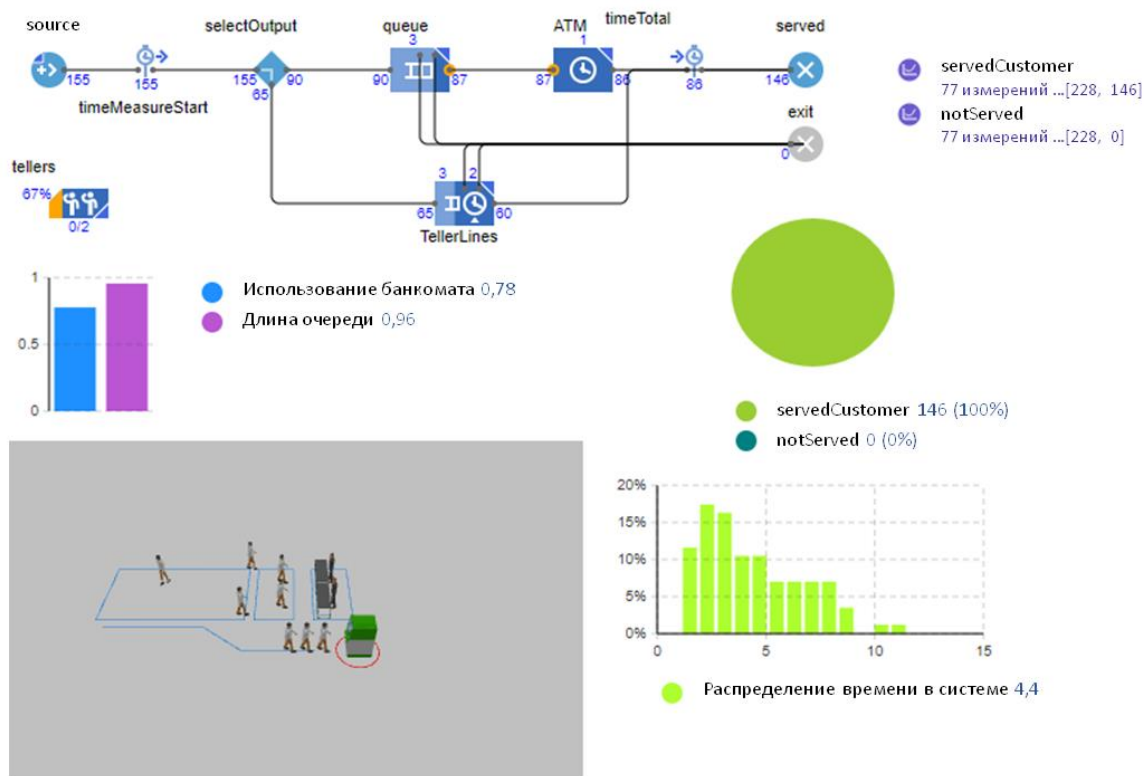


Рис. 3. Результаты экспериментов
 Примечание: составлено автором.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Смоделируем ситуацию, когда посетители проводят при обслуживании в банкомате в среднем 1,5 мин, а в кассе 5 мин. При увеличении этих значений на одну минуту, при неизменной интенсивности, наблюдается рост не обслуженных посетителей. Количество ушед-

ших людей составляет 2 %. При этом увеличилось среднее время, которое посетители проводят в банке, оно составило 16,67 мин. Результаты этого эксперимента представлены на рис. 4.

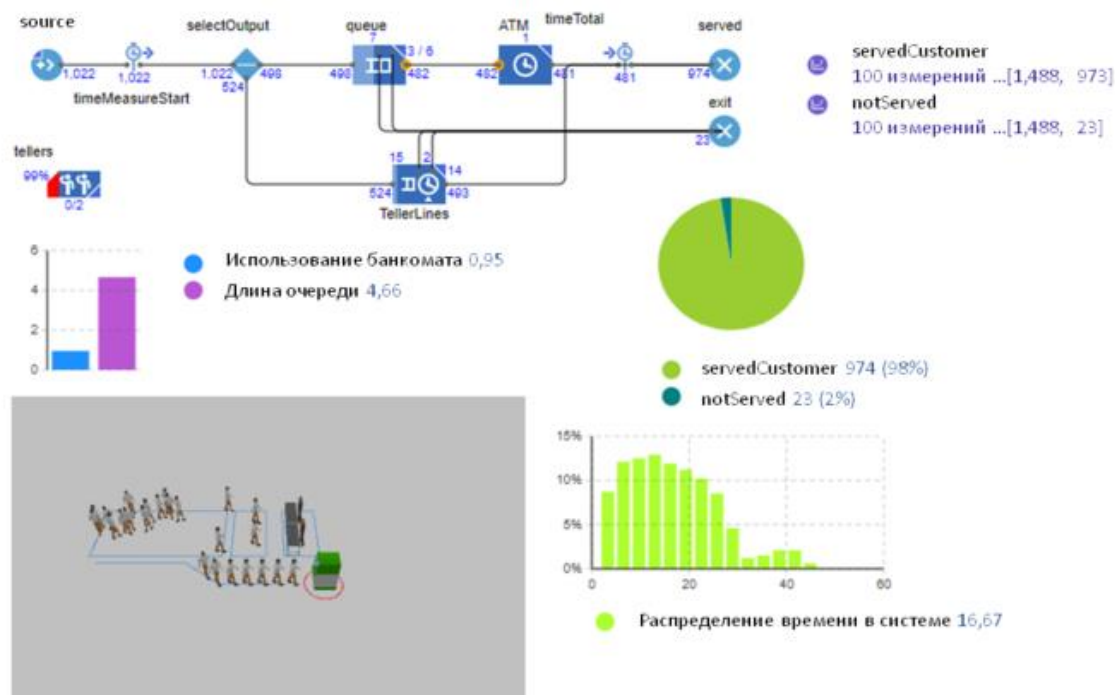


Рис. 4. Эксперимент с увеличением среднего времени обслуживания

Примечание: составлено автором.

В экспериментальных целях в модель была добавлена еще одна касса для улучшения обслуживания клиентов банка. По результатам эксперимента наблюдалось значительное сокращение очереди к кассам. В то время как

очередь к банкомату по-прежнему оставалась большой, а среднее время, которое посетители проводят в банке, составило 21,35 мин, что отражено на рис. 5.

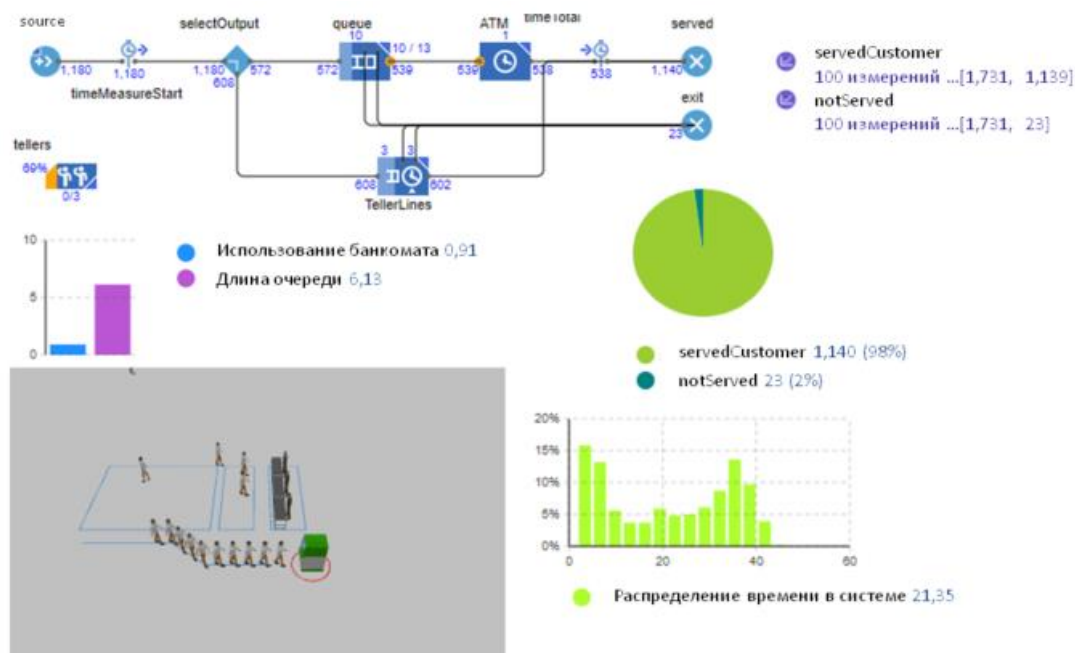


Рис. 5. Результат эксперимента с тремя кассами

Примечание: составлено автором.

Для сокращения очереди на обслуживание в модели реализована возможность перехода клиентов от банкомата к кассам. Диаграмма модернизированного процесса представлена на рис. 6, из которого видно, что все показатели системы обслуживания значительно улучшаются. Сокращается длина очереди

с 6,13 до 2,87. Отмечено, что посетители проводят в офисе в среднем 11,85 мин, то есть среднее время обслуживания уменьшилось на 9,5 мин. А процент ушедших без обслуживания клиентов сократился до 1 %. Результаты данного эксперимента представлены на рис. 6.

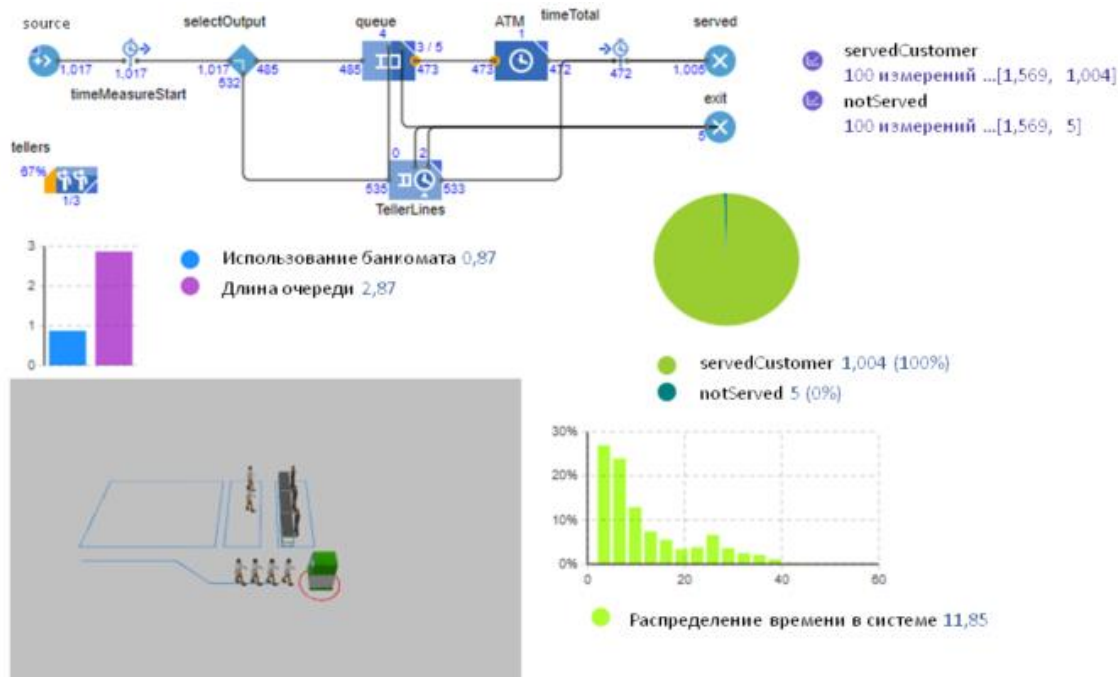


Рис. 6. Результаты эксперимента с модернизированной моделью

Примечание: составлено автором.

ЗАКЛЮЧЕНИЕ

Разработана имитационная модель системы обслуживания клиентов банковского отделения с применением методов агентного и дискретно-событийного моделирования.

Проведены эксперименты по исследованию работы системы в различных режимах, изучена статистика очередей и среднего времени обслуживания, сделаны выводы об особенностях функционирования системы обслуживания клиентов. Проведены эксперименты с улучшенной моделью банка: наглядно представлена эффективность принятых мер, отмечено сокращение среднего времени обслуживания и изменение статистики очередей в положительной динамике.

На сегодняшний день построенная имитационная модель обслуживания клиентов актуальна. Основываясь на результатах моде-

лирования, можно проводить экономическое планирование, анализировать деятельность банка и принимать решения с целью улучшения обслуживания посетителей.

Полученные результаты могут применяться не только при решении вопросов планирования и управления деятельностью банка, но и в рамках освоения учебных дисциплин основных образовательных программ бакалавриата и магистратуры для формирования профессиональных компетенций обучающихся. В ходе разработки имитационной модели студенты приобретают знания, умения и навыки по изучению поведения сложных систем. Применение в учебном процессе имитационного подхода при анализе и прогнозировании поведения сложных систем приводит к тому, что обучение становится более наглядным.

Список источников

1. Казеев И. М. Имитационное моделирование транспортно-технологического комплекса // Образоват. ресурсы и технологии. 2020. № 1 (30). С. 69–74.
2. Шамраева В. В., Кузовлев Е. Г., Баранник С. В. Реализация геоинформационных систем в дорожной области как одного из направлений информационного моделирования // Вестн. компьютер. и информ. технологий. 2018. № 6 (168). С. 20–26.
3. Романов В. П., Ахмадеев Б. А. Моделирование инновационной экосистемы на основе модели «хищник–жертва» // Бизнес-информатика. 2015. № 1. С. 7–17.
4. Веремчук Н. С. Об имитационном моделировании элементов дорожной сети // Перспективы науки. 2021. № 12. С. 38–41.
5. Акопов А. С., Хачатрян Н. К. Имитационная модель нефтеперерабатывающего предприятия со сложной схемой взаимодействия оборудования // Программная инженерия. 2021. Т. 12, № 6. С. 319–328.
6. Макаров В. Л., Бахтизин А. Р., Бекларян Г. Л., Акопов А. С., Ровенская Е. А., Стрелковский Н. В. Агентное моделирование социально-экономических последствий миграции при государственном регулировании занятости // Экономика и математ. методы. 2022. Т. 58, № 1. С. 113–130. DOI 10.31857/S042473880018960-5.
7. Макаров В. Л., Бахтизин А. Р., Бекларян Г. Л., Акопов А. С. Цифровой завод: методы дискретно-событийного моделирования и оптимизации производственных характеристик // Бизнес-информатика. 2021. Т. 15, № 2. С. 7–20.
8. Ordu M., Demir E., Tofallis C., Gunal M. M. A Novel Healthcare Resource Allocation Decision Support Tool: A Forecasting-Simulation-Optimization Approach // Journal of the Operational Research Society. 2021. Vol. 72, Is. 3. P. 485–500.
9. Oueida S., Aloqaily M., Ionescu S. A Smart Healthcare Reward Model for Resource Allocation in Smart City // Multimedia Tools and Applications. 2019. Vol. 78. P. 24573–24594.
10. Боев В. Д. Компьютерное моделирование в среде AnyLogic. М. : Юрайт, 2021. 298 с.

Информация об авторе

Н. С. Веремчук – кандидат физико-математических наук, доцент.

References

1. Kazeev I. M. Simulation Modeling Transport and Technological Complex // Educational Resources and Technologies. 2020. No. 1 (30). P. 69–74. (In Russian).
2. Shamraeva V. V., Kuzovlev E. G., Barannik S. V. Implementation of Geographic Information Systems in the Road Sector as One of the Directions of Information Modeling // Herald of Computer and Information Technologies. 2018. No. 6 (168). P. 20–26. (In Russian).
3. Romanov V. P., Akhmadeev B. A. Innovation Ecosystem Modelling Based on “Predator-Prey” Model // Business Informatics. 2015. No. 1. P. 7–17. (In Russian).
4. Veremchuk N. S. On Simulation Modeling of Road Network Elements // Science Prospects. 2021. No. 12. P. 38–41. (In Russian).
5. Akopov A. S., Khachatryan N. K. Simulation Model of an Oil Refinery with a Complex Scheme of Equipment Interaction // Software Engineering. 2021. Vol. 12, No. 6. P. 319–328. (In Russian).
6. Makarov V. L., Bakhtizin A. R., Beklaryan G. L., Akopov A. S., Rovenskaya E. A., Strelkovsky N. V. Agent-Based Modeling of Social and Economic Impacts of Migration under the Government Regulated Employment // Economics and Mathematical Methods. 2022. Vol. 58, No. 1. P. 113–130. DOI 10.31857/S042473880018960-5. (In Russian).
7. Makarov V. L., Bakhtizin A. R., Beklaryan G. L., Akopov A. S. Digital Plant: Methods of Discrete-Event Modeling and Optimization of Production Characteristics // Business Informatics. 2021. Vol. 15, No. 2. P. 7–20. (In Russian).
8. Ordu M., Demir E., Tofallis C., Gunal M. M. A Novel Healthcare Resource Allocation Decision Support Tool: A Forecasting-Simulation-Optimization Approach // Journal of the Operational Research Society. 2021. Vol. 72, Is. 3. P. 485–500.
9. Oueida S., Aloqaily M., Ionescu S. A Smart Healthcare Reward Model for Resource Allocation in Smart City // Multimedia Tools and Applications. 2019. Vol. 78. P. 24573–24594.
10. Boev V. D. Kompiuternoe modelirovanie v srede AnyLogic. Moscow : Iurait, 2021. 298 p. (In Russian).

Information about the author

N. S. Veremchuk – Candidate of Sciences (Physics and Mathematics), Associate Professor.

Научная статья
УДК 519.246.85 + 616.1
doi: 10.34822/1999-7604-2022-3-32-38

НЕЛИНЕЙНЫЙ АНАЛИЗ ПАРАМЕТРОВ КАРДИОСИГНАЛОВ С ХАОТИЧЕСКОЙ ДИНАМИКОЙ

Виолетта Вячеславовна Григоренко^{1✉}, Нина Борисовна Назина²

^{1,2} Сургутский государственный университет, Сургут, Россия

¹ grigv_84@mail.ru ✉, <https://orcid.org/0000-0002-9073-4184>

² nnb@mail.ru, <https://orcid.org/0000-0001-9363-160X>

Аннотация. В статье рассматривается исследование и оценка хаотической динамики параметров сложных нестационарных биосистем, в частности особый тип хаоса, возникающий в параметрах, которыми эти системы описаны. В отличие от детерминированного хаоса Лоренса, в таких системах не только невозпроизводим процесс, но и никогда невозпроизводимо начальное состояние системы. В основе исследования лежит оценка степени хаотичности временных рядов кардиосигналов сердечно-сосудистой системы человека с использованием методов непараметрической математической статистики, методов нелинейной динамики. При помощи автокорреляционного анализа на примере элементов кардиосигналов (временных рядов кардиоинтервалов) сердечно-сосудистой системы показано, что параметры биосистем имеют строго хаотическую структуру и нелинейную тенденцию. Расчет старшего показателя Ляпунова показал, что происходит постоянная смена знака (+, –) при каждой регистрации кардиосигналов, что доказывает наличие хаотических процессов в работе сердца. Исследование может быть использовано для идентификации параметров функционального состояния организма человека.

Ключевые слова: нелинейная динамика, старший показатель Ляпунова, анализ временных рядов, коэффициент автокорреляции, коррелограмма

Для цитирования: Григоренко В. В., Назина Н. Б. Нелинейный анализ параметров кардиосигналов с хаотической динамикой // Вестник кибернетики. 2022. № 3 (47). С. 32–38. DOI 10.34822/1999-7604-2022-3-32-38.

Original article

NON-LINEAR ANALYSIS OF CARDIAC SIGNALS PARAMETERS WITH CHAOTIC DYNAMICS

Violetta V. Grigorenko^{1✉}, Nina B. Nazina²

^{1,2} Surgut State University, Surgut, Russia

¹ grigv_84@mail.ru ✉, <https://orcid.org/0000-0002-9073-4184>

² nnb@mail.ru, <https://orcid.org/0000-0001-9363-160X>

Abstract. The article studies and evaluates the chaotic dynamics parameters of complex non-stationary biosystems, in particular, a specific chaos type occurring in the parameters that describe the systems mentioned. Contrary to Lawrence deterministic chaos, these systems provide neither the process of repetition nor the repetition of the system's initial state. The study is based on the evaluation of the degree of chaotic time series of cardiac signals of a human cardiovascular system using methods of non-parametric mathematical statistics and those of non-linear dynamics. On the example of elements of cardiac signals (time series cardiac intervals) of a cardiovascular system and with autocorrelation analysis, the study shows that biosystem parameters have a strictly chaotic structure and non-linear trend. The chaotic processes in cardiac performance are demonstrated by computing the largest Lyapunov exponent and, as a result, a constant sign (+, –) change in each cardiac signal recording. The study can be used to identify parameters of the functional state of a human organism.

Keywords: non-linear dynamics, largest Lyapunov exponent, time series analysis, autocorrelation coefficient, correlogram

For citation: Grigorenko V. V., Nazina N. B. *Non-Linear Analysis of Cardiac Signals Parameters with Chaotic Dynamics // Proceedings in Cybernetics*. 2022. No. 3 (47). P. 32–38. DOI 10.34822/1999-7604-2022-3-32-38.

ВВЕДЕНИЕ

Несмотря на значительный прогресс в расширении инструментальных способов неинвазивной диагностики состояний параметров биосистем, идет постоянный поиск повышения информационной отдачи от полученных данных исследований. На сегодняшний день актуальной проблемой является построение эффективных способов и алгоритмов обработки и анализа параметров биосистем, а также получения более точных и качественных результатов для дальнейшего прогнозирования и оценки функционального состояния организма человека.

По классификации W. Weaver, предложенной в 1948 г., биосистемы (и в частности, сердечно-сосудистая система) относятся к системам третьего типа (далее – СТТ) [1]. Это динамически изменяющиеся, нестационарные, открытые (подверженные воздействию шумов как внешней, так и внутренней природы) системы, а их параметры строго хаотичны [2, 3]. В отличие от детерминированного хаоса Лоренса, в таких системах не только невозпроизводим процесс, но и никогда невозпроизводимо начальное состояние системы $x(t_0)$ [4].

Оценка состояния СТТ зависит от возможностей количественного описания протекающих процессов в рамках ограниченной информации, в условиях большого количества внешних факторов воздействия, а также индивидуальных особенностей человека [5, 6]. Основной причиной создания моделей поведения вектора состояния СТТ является именно невозпроизводимость результатов экспериментов. Каждый раз регистрируемые выборки вектора кардиосигнала будут «уникальны» даже при одинаковых условиях эксперимента и, более того, уникальностью будет обладать каждый временной участок такого сигнала [7].

В современной медицине, в том числе российской, широко используют детерминистско-стохастический подход (далее – ДСП) обработки данных о состоянии систем третьего типа (нормативы, инструкции и т. д.). В ДСП параметры, которыми описана система организма человека или его подсистема, всегда усреднены. С позиции детерминистского подхода многократное повторение любого процесса должно обеспечивать идентификацию моделей сложной динамической системы в фазовом пространстве состояний. В стохастическом подходе такой вектор состояния системы $x(t)$ должен иметь повторяющееся начальное значение $x(t_0)$ и воспроизводимую функцию распределения $f(x)$ для всех конечных состояний $x(t_k)$. Если начальное состояние $x(t_0)$ воспроизвести точно невозможно, то применение стохастического подхода уже ограничено (нет повторений результатов испытаний) [8].

В последние годы активно развивается новое направление в естествознании – теория хаоса-самоорганизации [9]. В ее основе лежит идентификация особого типа хаоса, свойственного сложным, нестационарным, динамическим биосистемам, с использованием методов непараметрической математической статистики, методов нелинейной динамики (расчет старшего показателя Ляпунова), а также расчета параметров квазиаттракторов, которые качественно и количественно описывают хаотическую динамику биомедицинских сигналов [9].

Сердечно-сосудистая система (далее – ССС) относится к СТТ и представляет собой сложную, динамическую систему, параметры которой меняются в зависимости от времени и целого ряда внешних и внутренних факторов (в том числе шумов), которые непрерывно влияют на ее состояние [10]. Современные представления об особенностях кардиосигналов, которыми описана ССС, указывают на

необходимость исследования и оценки степени хаотичности такого рода сигналов. Такая количественная оценка может стать важным признаком для идентификации функциональных состояний организма человека, а также ряда патологий в работе сердца.

МАТЕРИАЛЫ И МЕТОДЫ

Одним из основных параметров для анализа состояния ССС является кардиосигнал. Кардиосигнал является нестационарным сигналом, который обладает индивидуальными биологическими свойствами организма конкретного человека. Одним из элементов структуры кардиосигнала является кардиоинтервал (далее – КИ). КИ ССС характеризуют сердечный ритм. Нормогенезу соответствуют интервалы между ударами сердца, которые варьируются, но без значительных отклонений, а равенство или существенное отклонение от среднего значения кардиоинтервалов свидетельствует о наличии патологии, например, различного рода аритмии (брадикардия, тахикардия) [11, 12]. Для регистрации кардиосигналов в исследовании использовался стационарный медицинский прибор – пульсоксиметр «Элокс-01М», который регистрирует пульсовую волну с одного из пальцев в положении сидя, в течение пятиминутного интервала времени (согласно рекомендациям Ассоциации кардиологов Европы) [13]. Сертифицированный медицинский прибор, пульсоксиметр «Элокс-01М», активно используется для экспресс-анализа состояния функциональных систем организма человека (применялся инфракрасный датчик для регистрации пульсовой волны и уровня насыщения кислорода – SpO_2).

В качестве способа идентификации хаотических процессов работы параметров сердечно-сосудистой системы в исследовании используется, во-первых, расчет коэффициента автокорреляции и построение автокорреляционных функций (далее – коррелограмм), по которым можно сделать вывод о структуре временного ряда (линейности, тенденции, хаотичности); во-вторых, определение скорости разбегания фазовых траекторий кардиосигналов, которая оценивается на основе расчета старшего показателя Ляпунова [14].

Хаотичность процесса работы сердца подтверждает использование метода анализа временных рядов КИ, расчета коэффициентов автокорреляции (далее – АСФ) и визуализации полученных функций в виде коррелограмм. При помощи автокорреляционной функции и по виду коррелограммы определялось наличие нелинейности в сигнале.

Старший показатель Ляпунова (далее – СПЛ) позволяет идентифицировать тип динамической системы с точки зрения присутствия хаотического поведения, а также способен отражать уровень хаотичности сигнала [15]. Знак старшего показателя Ляпунова полностью характеризует тип колебаний динамической системы: если он положителен, то это является признаком хаотичности биосистемы. Для расчета СПЛ в работе используется интерполяционный метод, представленный в работах А. В. Беспалова, Н. Д. Поляхова [15, 16].

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Методами математической непараметрической статистики определялась корреляционная зависимость между последовательными уровнями временного ряда со сдвинутыми на несколько шагов во времени рядами данных. Для каждой обследуемой число повторов регистраций кардиосигналов в эксперименте составило 15. Проводились расчеты коэффициента автокорреляции для лагов от 1 до 50. Пример результата расчета коэффициента автокорреляции с соответствующим ему лагом, а также коррелограмма временных рядов кардиоинтервалов обследуемых в различных функциональных состояниях представлены на рис. 1а, 1б, 1в. По оси абсцисс отложены значения лагов $\tau = 50$, по оси ординат – значения коэффициента автокорреляции АСФ.

На рис. 1а, 1б, 1в видно, что для всех состояний превалирует значение коэффициента корреляции меньше 0,5, что говорит о хаотичности процесса для всех временных рядов кардиоинтервалов, также визуализируется значительная разница в форме самой коррелограммы. В результате проведенных исследований было выявлено, что кардиоинтервалы имеют только хаотическую структуру временных рядов, а также сильную нелинейную составляющую.

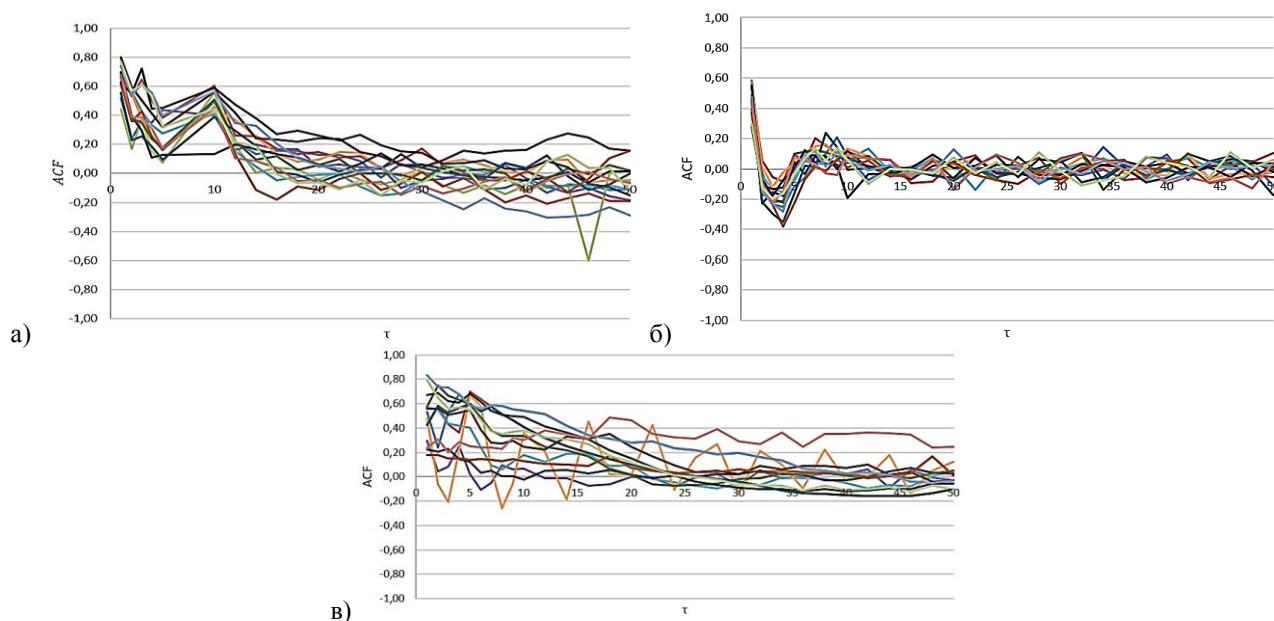


Рис. 1. Коррелограмма:

а – временных рядов кардиоинтервала (15 регистраций) обследуемой в состоянии нормогенеза;
б – временных рядов кардиоинтервала (15 регистраций) обследуемой в состоянии брадикардии;
в – временных рядов кардиоинтервала (15 регистраций) обследуемой в состоянии тахикардии

Примечание: составлено авторами.

На рис. 2а, 2б, 2в представлены рассчитанные коэффициенты СПЛ для трех человек (обследуемых) в различных функциональных состояниях (15 регистраций кардиосигнала).

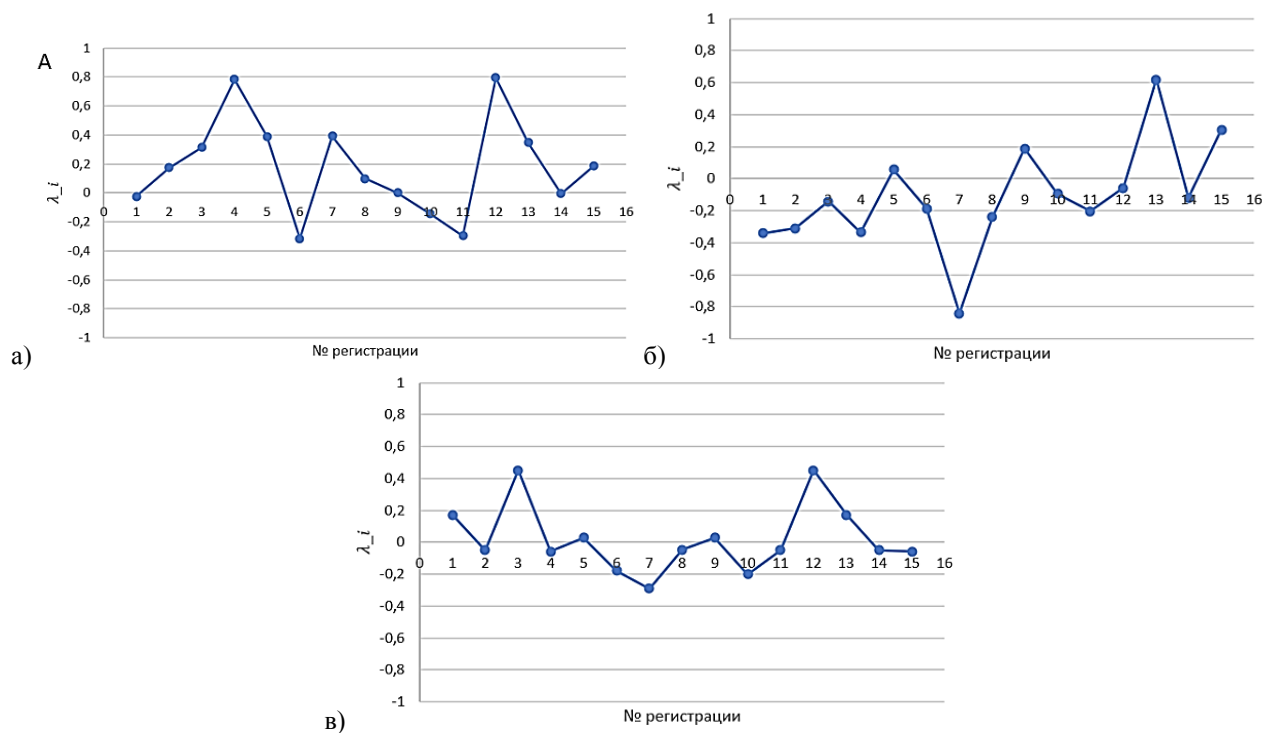


Рис. 2. Расчет СПЛ:

а – для временных рядов КИ обследуемой в состоянии нормогенеза;
б – для временных рядов КИ обследуемой в состоянии брадикардии;
в – для временных рядов КИ обследуемой в состоянии тахикардии

Примечание: составлено авторами.

В таблице представлены результаты расчета СПЛ (λ_i) для 15 повторов регистрации кардиосигналов испытуемых в различных

функциональных состояниях, а также рассчитанные абсолютные (Δ_x) и относительные (ε , %) погрешности.

Таблица

Результаты расчета СПЛ (λ_i) для 15-ти повторов регистраций кардиосигналов

№ рег-ии КИ	Состояния								
	Нормогенез			Брадикардия			Тахикардия		
	λ_i	Δ_x	ε , %	λ_i	Δ_x	ε , %	λ_i	Δ_x	ε , %
1	-0,0247	0,11	1,37 %	-0,3415	0,9798	1,03 %	0,1696	0,77	0,85 %
2	0,1750	0,40	0,70 %	-0,3102	0,17	0,20 %	-0,0512	0,73	2,70 %
3	0,3131	0,71	0,70 %	-0,1427	0,1	0,25 %	0,4498	0,26	0,12 %
4	0,7822	0,56	0,22 %	-0,3354	0,26	0,28 %	-0,0580	0,02	0,07 %
5	0,3893	0,69	0,54 %	0,0557	0,65	4,33 %	0,0278	0,83	6,38 %
6	-0,3144	0,13	0,13 %	-0,1871	0,58	1,12 %	-0,1779	0,35	0,36 %
7	0,3930	0,91	0,71 %	-0,8426	0,76	0,32 %	-0,2898	0,88	0,54 %
8	0,0977	0,03	0,09 %	-0,2414	0,82	1,22 %	-0,0470	0,73	2,70 %
9	0,0012	0,39	0,00 %	0,1884	0,93	1,79 %	0,0278	0,86	6,62 %
10	-0,1461	0,91	1,94 %	-0,0940	0,4	1,54 %	-0,2006	0,5	0,50 %
11	-0,2959	0,05	0,05 %	-0,2027	0,97	1,73 %	-0,0512	0,73	2,70 %
12	0,7939	0,41	0,16 %	-0,0583	0,39	2,44 %	0,4498	0,26	0,12 %
13	0,3469	0,80	0,71 %	0,6189	0,9	0,52 %	0,1696	0,77	0,85 %
14	-0,0056	0,83	0,75 %	-0,1207	0,9	2,73 %	-0,0494	0,82	3,16 %
15	0,1859	0,98	1,63 %	0,3058	0,94	1,11 %	-0,0580	0,02	0,07 %

Примечание: составлено авторами.

Как видно из рис. 2а, 2б, 2в и табл. 1, знак СПЛ все время меняет знак от регистрации к регистрации кардиосигнала вне зависимости от функционального состояния организма человека, что говорит о хаотической природе процесса работы сердца.

Необходимым условием возникновения хаоса в динамических нестационарных системах является размерность фазового пространства $N > 3$. В исследовании количество значений временного ряда КИ варьируется от 290 до 500 в зависимости от функционального состояния обследуемого (брадикардия: $\bar{x} = 280$; нормогенез: $\bar{x} = 300 \sim 350$; тахикардия: $\bar{x} = 500$ и выше).

ЗАКЛЮЧЕНИЕ

В результате проведенного исследования было выявлено и экспериментально подтверждено, что кардиосигналы, которыми описана сердечно-сосудистая система человека, имеют особую строго хаотическую структуру:

1. Результаты расчета коэффициентов автокорреляции и построение коррелограмм временных рядов кардиоинтервалов для обследуемых как в норме, так и при наличии патологии показали непрерывную смену знака с каждым увеличением лага, также нет стремления ACF к нулю.

2. Расчет старшего показателя Ляпунова для временных рядов кардиоинтервалов демонстрирует постоянную смену знака, а фазовые траектории непрерывно и хаотически пересекаются, что говорит о наличии хаоса, отличного от детерминированного (динамического) хаоса Лоренца. В аттракторах Лоренца их пересечение невозможно, так как они всегда расходятся.

Таким образом, представленный в работе стохастический и хаотический анализ временных рядов кардиоинтервалов ССС человека позволяет повысить качество диагностики физиологического состояния организма человека и расширить функциональные возможности существующих миниатюрных медицинских приборов кардиодиагностики.

Список источников

References

1. Чемпалова Л. С., Яхно Т. А., Манина Е. А., Игнатенко А. П., Оразбаева Ж. А. Гипотеза W. Weaver при изучении произвольных и непроизвольных движений // Вестн. новых мед. технологий. 2021. Т. 28, № 1. С. 75–77.
2. Grigorenko V. V., Eskov V. M., Filatov M. A., Pavlyk A. V. New Information Technologies in the Analysis of Electroencephalograms // J Phys: Conf Ser. 2020. Vol. 1679. P. 032081.
3. Grigorenko V. V., Eskov V. M., Nazina N. B., Egorov A. A. Information-Analytical System of Cardiographic Information Functional Diagnostics // J Phys: Conf Ser. 2020. Vol. 1515. P. 052027.
4. Еськов В. М., Еськов В. В., Гавриленко Т. В., Вохмина Ю. В. Формализация эффекта «повторение без повторения» Н. А. Бернштейна // Биофизика. 2017. Т. 62, № 1. С. 168–176.
5. Филатова О. Е., Русак С. Н., Майстренко Е. В., Добрынина И. Ю. Возрастная динамика параметров сердечно-сосудистой системы населения Севера РФ // Сложность. Разум. Постнеклассика. 2016. № 2. С. 40–49.
6. Filatova O. E., Bashkatova Yu. V., Shakirova L. S., Filatov M. A. Neural Network Technologies in System Synthesis // IOP Conference Series: Materials Science and Engineering. 2021. Vol. 1047. P. 012099.
7. Филатова О. Е., Хадарцева К. А., Филатова Д. Ю., Живаева Н. В. Биофизика сложных систем – complexity // Вестн. новых мед. технологий. 2016. Т. 23, № 2. С. 9–17.
8. Григоренко В. В. Обработка кардиографической информации на основе стохастического и хаотического подходов // Вестник НГИЭИ. 2019. № 4 (95). С. 78–88.
9. Vokhmina Yu. V., Eskov V. M., Gavrilenko T. V., Filatova O. E. Measuring Order Parameters Based on Neural Network Technologies // Measurement Techniques. 2015. Vol. 58. P. 462–466.
10. Аль-Хулейди Н. А. Система обработки и нейросетевого анализа биоэлектрических сигналов для решения задач медицинской диагностики : дис. ... канд. техн. наук. Владимир, 2014. 150 с.
11. Антипов О. И., Нагорная М. Ю. Показатель Херста биоэлектрических сигналов // Инфокоммуникационные технологии. 2011. № 1 (9). С. 75–77.
12. Хачатрян К. С., Манило Л. А. Метод анализа псевдофазового портрета в задаче распознавания биомедицинских сигналов // Биотехносфера. 2016. № 5 (47). С. 14–18.
13. Physiological Basis of Heart Rate Variability (HRV). URL: https://www.vitalscan.com/dt_hrv1.html (дата обращения: 03.06.2022).
14. Старченкова К. С., Манило Л. А. Оценка энтропии ритмограмм для разных видов сердечных аритмий // Биотехнические, медицинские и экологические системы, измерительные устройства и робототехнические комплексы : сб. тр. XXXII Всерос. науч.-техн. конф. студентов, молодых ученых и специалистов. Рязань, 2019. С. 206–209.
1. Chempalova L. S., Yakhno T. A., Manina E. A., Ignatenko A. P., Orazbaeva Zh. A. W. Weaver Hypothesis in Voluntary and Involuntary Movement's Studying // Journal of New Medical Technologies. 2021. Vol. 28, No. 1. P. 75–77. (In Russian).
2. Grigorenko V. V., Eskov V. M., Filatov M. A., Pavlyk A. V. New Information Technologies in the Analysis of Electroencephalograms // J Phys: Conf Ser. 2020. Vol. 1679. P. 032081.
3. Grigorenko V. V., Eskov V. M., Nazina N. B., Egorov A. A. Information-Analytical System of Cardiographic Information Functional Diagnostics // J Phys: Conf Ser. 2020. Vol. 1515. P. 052027.
4. Eskov V. M., Eskov V. V., Gavrilenko T. V., Vokhmina Yu. V. Formalization of the Effect of "Repetition without Repetition" Discovered by N. A. Bernshtein // Biophysics. 2017. Vol. 62, No. 1. P. 168–176. (In Russian).
5. Filatova O. E., Rusak S. N., Maystrenko E. V., Dobrynina I. Yu. Aging Dynamics of Cardiovascular System Parameters of the Russian North Citizen // Slozhnost. Razum. Postneklassika. 2016. No. 2. P. 40–49. (In Russian).
6. Filatova O. E., Bashkatova Yu. V., Shakirova L. S., Filatov M. A. Neural Network Technologies in System Synthesis // IOP Conference Series: Materials Science and Engineering. 2021. Vol. 1047. P. 012099.
7. Filatova O. E., Khadartseva K. A., Filatova D. Yu., Zhivaeva N. V. Biophysics of Complex Systems – Complexity // Journal of New Medical Technologies. 2016. Vol. 23, No. 2. P. 9–17. (In Russian).
8. Grigorenko V. V. Processing of Cardiographic Information on the Basis of Stochastic and Chaotic Approaches // Bulletin NGIEI. 2019. No. 4 (95). P. 78–88. (In Russian).
9. Vokhmina Yu. V., Eskov V. M., Gavrilenko T. V., Filatova O. E. Measuring Order Parameters Based on Neural Network Technologies // Measurement Techniques. 2015. Vol. 58. P. 462–466.
10. Al-Khuleydi N. A. Sistema obrabotki i neurosetevogo analiza bioelektricheskikh signalov dlia resheniia zadach meditsinskoi diagnostiki : Cand. Sci. Dissertation (Engineering). Vladimir, 2014. 150 p. (In Russian).
11. Antipov O. I., Nagornaya M. Yu. Bioelectric Signal Hurst Index // Infokommunikatsionnye tekhnologii. 2011. No. 1 (9). P. 75–77. (In Russian).
12. Khachatryan K. S., Manilo L. A. The Method of Analysis Pseudophase Portrait in the Problem of Recognition of Biomedical Signals // Biotekhnosfera. 2016. No. 5 (47). P. 14–18. (In Russian).
13. Physiological Basis of Heart Rate Variability (HRV). URL: https://www.vitalscan.com/dt_hrv1.html (accessed: 03.06.2022).
14. Starchenkova K. S., Manilo L. A. Estimation of Entropy of Rhythmograms for Different Types of Cardiac Arrhythmias // Biotekhnicheskie, meditsinskie i ekologicheskie sistemy, izmeritelnye ustroystva i robototekhnicheskie komplekсы : Proceedings of the XXXII All-Russian Scientific and

15. Беспалов А. В., Поляхов Н. Д., Якупов О. Э. Алгоритм оценки первого показателя Ляпунова по временному ряду // Междунар. конф. по мягким вычислениям и измерениям. 2010. Т. 1. С. 151–155.
16. Поляхов Н. Д., Беспалов А. В. Сравнительный анализ методов оценки первого показателя Ляпунова // Современ. проблемы науки и образования. 2012. № 6. С. 7.
- Engineering Conference of Students, Young Scientists and Specialists. Ryazan, 2019. P. 206–209. (In Russian).
15. Bespalov A. V., Polyakhov N. D., Yakupov O. E. Algoritm otsenki pervogo pokazatelya Lyapunova po vremennomu riadu // International Conference on Soft Computing and Measurement. 2010. Vol. 1. P. 151–155. (In Russian).
16. Polyakhov N. D., Bespalov A. V. Comparative Analysis of First Lyapunov Exponent Estimation Methods // Modern Problems of Science and Education. 2012. No. 6. P. 7. (In Russian).

Информация об авторах

В. В. Григоренко – старший преподаватель.
Н. Б. Назина – доцент.

Information about the authors

V. V. Grigorenko – Senior Lecturer.
N. B. Nazina – Associate Professor.

Научная статья

УДК 004.41

doi: 10.34822/1999-7604-2022-3-39-45

АВТОМАТИЗИРОВАННЫЙ СБОР ДАННЫХ КЛЮЧЕВЫХ ФИНАНСОВЫХ ПОКАЗАТЕЛЕЙ ПРЕДПРИЯТИЙ IT-ОТРАСЛИ РЕГИОНА

**Сергей Олегович Крамаров^{1✉}, Виктор Андреевич Овсянников²,
Людмила Викторовна Сахарова³, Роман Сергеевич Усатый⁴,
Галина Викторовна Лукьянова⁵**

¹ Южный федеральный университет, Ростов-на-Дону, Россия

² Национальный исследовательский университет «Высшая школа экономики»,
Москва, Россия

^{3,4,5} Ростовский государственный экономический университет (РИНХ),
Ростов-на-Дону, Россия

¹ maoovo@yandex.ru ✉, <https://orcid.org/0000-0003-3743-6513>

² viktordee41@gmail.com, <https://orcid.org/0000-0003-1824-9536>

³ L_Sakharova@mail.ru, <https://orcid.org/0000-0002-4897-4926>

⁴ rs.usaty@gmail.com, <https://orcid.org/0000-0002-0993-7575>

⁵ lukyanova.g@yandex.ru, <https://orcid.org/0000-0001-6674-2240>

Аннотация. Рассматривается новое программное обеспечение для автоматизированного сбора данных, позволяющее на основе парсинга открытых сайтов за короткое время получать в Excel-формате данные о наиболее значимых финансово-экономических показателях фирм конкретной отрасли, что позволяет провести полноценный финансово-экономический анализ выбранной отрасли в данном регионе. Важным условием для сбора данных является наличие списка необходимых для изучения фирм на официальном сайте export-base.ru. В качестве основного инструмента написания программы был выбран язык программирования Python и ключевые библиотеки, такие как *request*, *selenium*, *beautifulsoup4* и *csv*. Рассматриваемое в работе ПО апробировано на примере быстрого получения комплексных данных IT-компаний Ростовской области для проведения финансово-экономического анализа IT-отрасли региона.

Ключевые слова: парсинг, скрейпинг, Python, автоматизированный сбор информации

Для цитирования: Крамаров С. О., Овсянников В. А., Сахарова Л. В., Усатый Р. С., Лукьянова Г. В. Автоматизированный сбор данных ключевых финансовых показателей предприятий IT-отрасли региона // Вестник кибернетики. 2022. № 3 (47). С. 39–45. DOI 10.34822/1999-7604-2022-3-39-45.

Original article

AUTOMATED DATA COLLECTION OF KEY FINANCIAL INDICATORS OF IT ENTERPRISES IN THE REGION

**Sergey O. Kramarov^{1✉}, Viktor A. Ovsyannikov², Lyudmila V. Sakharova³,
Roman S. Usaty⁴, Galina V. Lukyanova⁵**

¹ Southern Federal University, Rostov-on-Don, Russia

² Higher School of Economics, Moscow, Russia

^{3,4,5} Rostov State University of Economics, Rostov-on-Don, Russia

¹ maoovo@yandex.ru ✉, <https://orcid.org/0000-0003-3743-6513>

² viktordee41@gmail.com, <https://orcid.org/0000-0003-1824-9536>

³ L_Sakharova@mail.ru, <https://orcid.org/0000-0002-4897-4926>

⁴ rs.usaty@gmail.com, <https://orcid.org/0000-0002-0993-7575>

⁵ lukyanova.g@yandex.ru, <https://orcid.org/0000-0001-6674-2240>

Abstract. The article deals with a new software for automated data collection. Using parsing of open websites, this software allows obtaining a set of the most significant financial and economic indicators of firms for a particular industry in the region in a short time and presents it in an Excel document, resulting in a comprehensive financial and economic analysis of a given industry in the region. A list of required firms published at export-base.ru is a necessary condition for data collection. Python and such key libraries as *request*, *selenium*, *beautifulsoup4*, and *csv* are selected as the main tool for writing the program. The software developed is tested for quickly obtaining complex data from IT companies in the Rostov Oblast to conduct the financial and economic analysis of the IT industry in the region.

Keywords: parsing, scraping, Python, automated data collection

For citation: Kramarov S. O., Ovsyannikov V. A., Sakharova L. V., Usatyi R. S., Lukyanova G. V. *Automated Data Collection of Key Financial Indicators of IT Enterprises in the Region // Proceedings in Cybernetics. 2022. No. 3 (47). P. 39–45. DOI 10.34822/1999-7604-2022-3-39-45.*

ВВЕДЕНИЕ

Исследование проводилось с целью разработки универсальной методологии сбора данных из открытых источников, предназначенных для финансово-экономического анализа конкретной отрасли в регионе на основе открытых данных показателей предприятий.

Несмотря на наличие большого количества программного обеспечения (далее – ПО), с помощью которого возможно проведение финансово-экономического анализа отдельных предприятий, отсутствуют программы, осуществляющие полный анализ конкретной отрасли. Препятствием для создания подобного ПО, помимо проблем с методиками расчета интегральных показателей, является то, что данные о фирмах должны удовлетворять нескольким требованиям одновременно:

- а) используются стандартные для финансового анализа данные;
- б) данные получены законным путем, из открытых источников, для выборки предприятий отрасли, позволяющей провести полноценный статистический и нечетко-множественный анализ;
- в) проверены и достоверны;
- г) представлены в единой форме за ряд лет;
- д) указаны на самой html-странице или в виде таблицы.

Кроме того, информация на сайте должна быть разрешена к копированию.

Обеспечить быстрый сбор и обработку информации в удобном виде из открытых источников данных достаточно сложно, потому что:

1. Бухгалтерская отчетность предприятий, которая представлена на ряде сайтов, находится в полузакрытом либо в малодоступном

виде. Встречаются также проблемы CAPTCHA (блокировки данных после ряда обращений, проверка cookie-файлов, задержка перед действиями, ограничения по количеству обращений в течение определенного времени и т. д.).

2. Многие предприятия постоянно меняют дизайн сайтов, где убирают информацию за предыдущие годы, а за текущий год вводят платный доступ, добавляют или убирают новые блоки информации, меняют названия запросов данных с сервера и др.

3. Информация часто представлена в малопригодном для парсига (нетиповом) виде, то есть данные могут находиться в JS-файле, для каждого предприятия организация xpath может быть разной, что часто мешает точно определить собранные данные.

В настоящем исследовании была поставлена задача разработки ПО, которое ориентировано на решение задач автоматического сбора данных, необходимых и достаточных для проведения анализа финансово-экономического состояния конкретной отрасли по необходимым показателям. Имеющиеся в настоящее время программы, осуществляющие автоматизированный сбор общедоступной информации из интернета без использования API сайтов (Scraping (далее – СКРП), или его российский вариант – Parsing (далее – ПРС)), используются для анализа данных в достаточно узких целях [1–2].

Разрабатываемое в данной статье ПО должно быть основано на предварительном автоматическом анализе данных из различных источников с учетом их полноты, достоверности и корректности, а также отвечать современным требованиям корректного ис-

пользования информации из сети с учетом имеющихся законов, ограничивающих парсинг [2–3].

В настоящее время в различных методах сбора данных из сети для аналитических исследований активно используют СКРП, который является одной из основных технологий получения веб-данных путем извлечения их со страниц веб-ресурсов, что помогает автоматизировать процесс сбора данных путем создания различных библиотек [2].

Однако при использовании СКРП в большинстве случаев решаются только конкретные, практико-ориентированные задачи, подразумевающие сбор типовых данных из одного источника, упорядоченных особым способом, на относительно коротком промежутке времени. Всякий раз, когда источники данных меняются, ПО должно создаваться заново. Как следствие, имеется крайне мало теоретических исследований, посвященных возможностям СКРП по извлечению из сети репрезентативных выборок, пригодных для полноценного статистического и эконометрического анализа данных, особенно за короткое время.

Между тем в настоящее время назрела необходимость в сборе данных для полноценного информационного анализа данных (далее – ИАД) сложных социальных, экономических и других систем на основе больших массивов данных за длительные промежутки времени. Особое значение такие исследования имеют для системного анализа при формировании региональных кластеров. Современные методы ИАД позволяют это сделать, а необходимость управления системами требует разработки соответствующего универсального исследовательского аппарата.

МАТЕРИАЛЫ И МЕТОДЫ

Для решения поставленных в работе задач нами были проведены следующие исследования:

- сделан подробный анализ сайтов с данными о финансовом состоянии предприятий с целью выбора оптимального сайта, позволяющего получать для выборки предприятий заданного ОКВЭД сведения о таких показателях, как прибыль, сумма уплаченных нало-

гов и социальных выплат (по видам), коэффициентов оценки финансового состояния и др.;

- выполнена разработка соответствующего ПО, осуществляющего автоматизированный сбор данных с отобранных сайтов посредством использования СКРП и ПРС;

- на примере ИТ-предприятий Ростовской области был проведен автоматический перевод данных в удобный для анализа формат Excel.

На первом этапе работы нами были проанализированы сайты, пригодные для получения необходимой информации: «Государственный информационный ресурс бухгалтерской (финансовой) отчетности», «За честный бизнес», «Спектр-аудит», «Бухгалтерская (финансовая) отчетность предприятий» и др. [4–7]. Однако все они, как правило, имеют ряд недостатков, перечисленных выше.

В качестве основного нами был выбран сайт TestFirm.ru, как наиболее удовлетворяющий вышеперечисленным требованиям [7]. Использование других сайтов проводилось нами также для извлечения дополнительной информации.

На следующем этапе работы нами был написан соответствующий код ПРС, который позволяет выполнять настройку создаваемого ПО под блоки изучаемых сайтов. В качестве инструмента для написания программы нами был выбран Python и его ключевые библиотеки (*request*, *selenium*, *beautifulsoup4*, *csv*) [8].

В качестве технических новшеств предложенного нами ПО можно отметить те, которые позволили не ограничивать работу во времени с сайтами, с момента начала работы с клавиатурой (в любом удобном виде). Это дало возможность решить проблему быстрого сбора информации после начала работы с клавиатурой и проводить работу без временных задержек и ограничений на количество обращений к ПО, проводить проверку ПРС, по ключевым словам, а также осуществлять упорядоченное сохранение собранных данных.

При разработке ПО нами была также создана новая технология использования СКРП для систематизации информации, размещенной на различных сайтах, при этом програм-

ма позволяет подстроить ПРС и под другие сайты.

Разработанное ПО было разделено нами на три блока:

1. Сбор URL-адресов.
2. Парсинг финансовых данных с сайта.
3. Отдельный парсер для налогов и страховых взносов.

Остановимся более подробно на каждом из них.

Для сбора URL-адресов была использована библиотека selenium для Python, а в качестве эмулированного браузера использовался Google Chrome. Сам код разрабатывался с добавлением *Chromedriver* (часть *Chromium*, которая отвечает за управление им с помощью внешних инструментов) [9] с последующим подключением необходимых электронных библиотек и созданием функции, которая отвечает за вставку ИНН конкретных фирм и их URL-адресов. При внесении ИНН фирм, которые рассматриваются в качестве поиска, используем библиотеку csv для сохранения наших адресов. Для сбора информации некоторых блоков был использован также xpath (язык запросов к элементам XML-документа). Разработан для орга-

низации доступа к частям документа XML в файлах трансформации XSLT, что является стандартом консорциума W3C из библиотеки requests [10–12].

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В качестве примера был проведен анализ открытых данных предприятий Ростовской области, осуществляющих деятельность в области разработки компьютерного программного обеспечения, оказывающих консультационные услуги в данной области и другие сопутствующие услуги (код ОКВЭД 62). В этой базе данных имеются характеристики 447 предприятий, однако в результате парсинга были получены данные всего для 363, поскольку остальные не сдавали налоговую отчетность более года по данным ФНС.

Список предприятий Ростовской области, осуществляющих деятельность по разработке компьютерного программного обеспечения, оказанию консультационных услуг в данной области и других сопутствующих услуг, получен на сайте export-base.ru, откуда была сделана выгрузка названий компаний и их ИНН (рис. 1) [13].

База скачана с https://export-base.ruКод ОКВЭД				ИНН	ОГРН	Стационарный телефон	Мобильный телефон	Дата регистрации	Город
Ростовская область									
62. Разработка компьютерного программного обеспечения, консультационные услуги в данной области и другие сопутствующие услуги									
62.01. Разработка компьютерного программного обеспечения									
ВС САППОРТ, ООО	62.01	Разработка компьютерного программного обеспечения	6165211290	1176196061546		+7 (906) 4160220	28.12.2017	Ростовна-Дону	
ФЛАГМАН, ООО	62.01	Разработка компьютерного программного обеспечения	6163106798	1116195002274	+7 (8632) 709169	+7 (928) 2293363	18.03.2011	Ростовна-Дону	
БУРАЛО ТЕХНОЛОГИИ, ООО	62.01	Разработка компьютерного программного обеспечения	6165209816	1176196046290			29.09.2017	Ростовна-Дону	
ИМПЕКС ГРУПП, ООО	62.01	Разработка компьютерного программного обеспечения	6147037843	1146191000229	+7 (863) 6065200	+7 (928) 6065200	18.02.2014	Каменск-Шахтинский	
ТАЛАН, ООО	62.01	Разработка компьютерного программного обеспечения	6143063225	1066143050797	+7 (863) 9248683		04.09.2006	Волгодонск	
					+7 (863) 2191600, +7 (863) 2990446, +7 (863) 2236600, +7 (863) 2506897, +7 (863) 3000230, +7 (863) 3001000, +7 (863) 3001006		+7 (988) 5882968, +7 (951) 2563130	26.02.1997	Ростовна-Дону
ГЭНДАЛЬФ СЕРВИС, ООО ВЦ	62.01	Разработка компьютерного программного обеспечения	6163154865	1176196024641					
СПАРК ДЕВЕЛОПМЕНТ, ООО	62.01	Разработка компьютерного программного обеспечения	6164301047	1106164005639	+7 (863) 2993718		+7 (918) 5025607	28.12.2010	Ростовна-Дону
АЛИН, ООО	62.01	Разработка компьютерного программного обеспечения	6165120653	1056165030085	+7 (863) 2450574			02.03.2005	Ростовна-Дону
ГЭНДАЛЬФ, ООО	62.01	Разработка компьютерного программного обеспечения	6167084305	1066167006421	+7 (863) 2236500, +7 (863) 3001001, +7 (863) 3001000		+7 (928) 1713951	07.02.2006	Ростовна-Дону
ГЭНДАЛЬФ, ООО ЦИТ	62.01	Разработка компьютерного программного обеспечения	6167132823	1166196062834	+7 (863) 2236500, +7 (863) 3001000	+7 (903) 4063130	23.11.1993	Ростовна-Дону	
ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ ЮФУ, НТЦ	62.01	Разработка компьютерного программного обеспечения	6163027810	1026103165241	+7 (8634) 312350, +7 (8634) 315941, +7 (8634) 318000, +7 (8634) 328725, +7 (8634) 371635, +7 (8634) 615459, +7 (8634) 362550			03.02.1994	Таганрог

Рис. 1. Пример таблицы данных по предприятиям из export-base.ru по Ростовской области

Примечание: составлено авторами на основании данных, полученных в исследовании.

Разработанное нами ПО выполняет автоматически следующие действия: эмулирует браузер, вводит ИНН фирмы в поиск и копирует URL-адрес страниц в отдельный csv файл для дальнейшей работы с ними, собирает необходимую информацию (ИНН, название компании, ОКВЭД, доходы, расходы, налог на прибыль, земельный, социальное страхование и др.). Эти данные вносятся автоматически в таблицы Excel со всеми необходимыми финансовыми показателями предприятий. Такие таблицы удобны для редактирования и пригодны для работы дополнительного программного комплекса, осу-

ществляющего анализ конкретной отрасли на основе выбранных показателей.

Для такого сбора информации при использовании разработанного ПО требуется от 30 секунд до 1 минуты, а после на парсинг одной страницы уходит одна-две секунды, что приводит к значительной экономии времени при анализе от 100 и более фирм.

В итоге мы получаем таблицу Excel со всеми финансовыми показателями предприятий (рис. 2), которые удобны для дальнейшего редактирования и пригодны для работы программного комплекса, осуществляющего комплексный анализ ИТ-отрасли на основе выбранных показателей.

A8	✕	✓	<i>f_x</i>	ООО «ГЭНДАЛЬФ»
	A	B	C	
1	ООО «Флагман»	(Малое предприятие)	6163106798	
2	ООО «БУРАЛО ТЕХНОЛОГИИ»	(Микропредприятие)	6165209816	
3	ООО «ИМПЕКС ГРУПП»	(Микропредприятие)	6147037843	
4	ООО «Талан»	(Микропредприятие)	6143063225	
5	ООО ВЦ «ГЭНДАЛЬФ сервис»	(Малое предприятие)	6163154865	
6	ООО «Спарк Девелопмент»	(Микропредприятие)	6164301047	
7	ООО «АЛИН»	(Микропредприятие)	6165120653	
8	ООО «ГЭНДАЛЬФ»	(Малое предприятие)	6167084305	
9	ООО ЦЕНТР ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ «ГЭНДАЛЬФ»		6167132823	
10	ООО «ПРОГА»	(Микропредприятие)	6167040234	
11	ООО ТОРГОВО-ПРОИЗВОДСТВЕННАЯ КОМПАНИЯ «РОСТСТРОЙПАНЕЛЬ»	(Микропредприятие)	6165206685	
12	ООО «РУФОРМАТ»	(Микропредприятие)	6165216813	
13	ЗАО «ТАМГА»	(Малое предприятие)	6167046035	
14	ООО «АСТОРМЕДИА»	(Микропредприятие)	6163152297	
15	ООО «ДА-ВИНЧИ»	(Микропредприятие)	6165128998	
16	ООО «Информационные Технологии»	(Микропредприятие)	6154140351	
17	ООО «ЮГ-ИНТЕГРАТОР»	(Микропредприятие)	6161042796	
18	ООО «АЛМУР»	(Микропредприятие)	6161004945	

Рис. 2. Excel-таблица финансовых показателей предприятий ИТ-отрасли Ростовской области

Примечание: составлено авторами на основании данных, полученных в исследовании.

На рис. 3 приведен пример анализа финансовых показателей предприятий ИТ-отрасли Ростовской области.

Предложенное программное обеспечение ориентировано на решение научно-исследовательской задачи сбора данных, необходимых и достаточных для проведения анализа финансово-экономического состояния отрасли по конкретным показателям, что отличает его от распространенных в настоящее время программ скрейпинга для анализа данных в практических, сиюминутных целях. Как след-

ствие, разработанное программное обеспечение основано на предварительном анализе источников данных с учетом их полноты, достоверности и корректности.

Благодаря разработке данного ПО сделан шаг в направлении развития технологий использования скрейпинга для решения долговременных научно-исследовательских задач и систематизации информации, размещенной на различных сайтах. Блоки кода позволяют подстроить парсер под другие сайты.

	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN
	Налог, взимаемый в связи с применением упрощенной системы налогообложения	Неналоговые доходы	Налог на добавленную стоимость	Налог на прибыль	Земельный налог	Транспортный налог	Налог на имущество организаций	Задолженность и перерасчеты по отмененным доходам	Страховые и другие взносы на обязательное пенсионное страхование	Страховые взносы на обязательное медицинское страхование	Страховые взносы на обязательное социальное страхование
1											
8		0 руб.	0 руб.	0 руб.				0 руб.	0 руб.		
9	3,4 млн. руб.	0 руб.	0 руб.	0 руб.			0 руб.	0 руб.	2,23 млн. руб.	1,11 млн. руб.	550 тыс. руб.
10			0 руб.	0 руб.			0 руб.	0 руб.	0 руб.	0 руб.	20 руб.
11	40,2 тыс. руб.		0 руб.	0 руб.					37,3 тыс. руб.	9,11 тыс. руб.	4,69 тыс. руб.
12	0 руб.	0 руб.	126 тыс. руб.	8,28 тыс. руб.					1,81 тыс. руб.	422 руб.	240 руб.
13	71,9 тыс. руб.								161 тыс. руб.	37,6 тыс. руб.	21,1 тыс. руб.
14	0 руб.		35,3 млн. руб.	404 тыс. руб.	222 тыс. руб.	7,85 тыс. руб.	5,05 тыс. руб.	0 руб.	5,76 млн. руб.	1,96 млн. руб.	402 тыс. руб.
15	16,4 тыс. руб.	0 руб.							54,1 тыс. руб.	12,5 тыс. руб.	7,14 тыс. руб.
16		0 руб.	419 тыс. руб.	68,2 тыс. руб.		1,78 тыс. руб.	0 руб.	0 руб.	50 тыс. руб.	14,2 тыс. руб.	5,19 тыс. руб.
17	10 тыс. руб.								16,3 тыс. руб.	3,8 тыс. руб.	2,16 тыс. руб.
18	12,5 тыс. руб.								16,4 тыс. руб.	3,81 тыс. руб.	2,16 тыс. руб.
19	890 тыс. руб.		0 руб.	0 руб.			0 руб.	0 руб.	519 тыс. руб.	120 тыс. руб.	68,5 тыс. руб.
20									0 руб.		
21	12,6 тыс. руб.	0 руб.							28,8 тыс. руб.	6,69 тыс. руб.	3,8 тыс. руб.
22	12,6 тыс. руб.								90 тыс. руб.	26,9 тыс. руб.	11,8 тыс. руб.
23	188 тыс. руб.		0 руб.	0 руб.			0 руб.		71,4 тыс. руб.	18,5 тыс. руб.	8,41 тыс. руб.
24			0 руб.	0 руб.			0 руб.		0 руб.		
25	0 руб.								0 руб.	0 руб.	0 руб.
26	24,2 тыс. руб.	0 руб.							148 тыс. руб.	37,2 тыс. руб.	18,3 тыс. руб.
27		0 руб.	2,49 млн. руб.	0 руб.	0 руб.	2,56 тыс. руб.	24,3 тыс. руб.	0 руб.	1,69 млн. руб.	896 тыс. руб.	384 тыс. руб.
28	63,1 тыс. руб.		0 руб.						99,4 тыс. руб.	24,8 тыс. руб.	12,2 тыс. руб.

Рис. 3. Финансовые показатели предприятий IT-отрасли Ростовской области
 Примечание: составлено авторами на основании данных, полученных в исследовании.

ЗАКЛЮЧЕНИЕ

Разработанное программное обеспечение для автоматизированного сбора данных позволяет на основе парсинга открытого сайта TestFirm.ru в компактной форме Excel-файла за кратчайшее время получать сведения о значимых финансово-экономических показателях фирм заданной отрасли по интересующему региону. Новизна разработанного ПО состоит в том, что оно ориентировано не на решение текущих практических задач анализа информации в сети, а на сбор информации для полноценного научного исследования. Как следствие, при разработке программного обеспечения выполнены следующие требования: собраны стандартные данные, используемые в финансовом анализе; данные получены законным путем, из открытых источников; данные получены для выборки предприятий отрасли, позволяющей провести полноценный статистический и нечетко-множественный анализ; данные прове-

рены и достоверны, представлены в единой форме за ряд лет; данные указаны на самой html-странице или в виде таблицы; информация на сайте разрешена к копированию.

Разработанное ПО имеет четко продуманную структуру, оптимизированное быстродействие и отвечает современным требованиям корректного использования информации из сети с учетом имеющихся в российском законодательстве законов, ограничивающих парсинг. В качестве технических новшеств предложенного ПО следует отметить следующие: нажатие кнопок и сбор информации после нажатия, сокращение времени задержки для обхода ограничений на количество нажатий по времени, проверка парсинга по ключевым словам, а также упорядоченное сохранение собранных данных.

Условием сбора данных является наличие списка фирм, использованного в данном исследовании, на официальном сайте export-base.ru.

Список источников

1. Krotov V., Silva L. Legality and Ethics of Web Scraping // Proceedings of the 24th Americas Conference on Information Systems, New Orleans, 2018. URL: https://www.researchgate.net/publication/324907302_Legality_and_Ethics_of_Web_Scraping (дата обращения: 30.08.2022).
2. Рожкова М. А. Базы данных и сервисы онлайн-классифайдов: пользование базой и использование информации // Журн. суда по интеллектуал. правам. 2019. № 26. С. 25–32.

References

1. Krotov V., Silva L. Legality and Ethics of Web Scraping // Proceedings of the 24th Americas Conference on Information Systems, New Orleans, 2018. URL: https://www.researchgate.net/publication/324907302_Legality_and_Ethics_of_Web_Scraping (accessed: 30.08.2022).
2. Rozhkova M. A. Databases and Services of Online Classifieds // Zhurn. suda po intellektual. pravam. 2019. No. 26. P. 25–32. (In Russian).

3. Государственный информационный ресурс бухгалтерской (финансовой) отчетности. URL: <https://bo.nalog.ru> (дата обращения: 30.08.2022).
4. Всероссийская система проверки контрагентов. URL: <https://zachestnyibiznes.ru> (дата обращения: 30.08.2022).
5. Спектр-аудит. URL: <https://spektr-audit.com> (дата обращения: 30.08.2022).
6. Бухгалтерская (финансовая) отчетность предприятий. URL: <https://e-ecolog.ru/buh> (дата обращения: 30.08.2022).
7. Сравнение финансового состояния фирмы с отраслевыми показателями и конкурентами. URL: <https://www.testfirm.ru> (дата обращения: 30.08.2022).
8. Mitchell R. Web Scraping with Python. 2nd Ed. 2018. 288 p.
9. Chromedriver. URL: <https://chromedriver.chromium.org/downloads> (дата обращения: 30.08.2022).
10. Adams M. D. Principled Parsing for Indentation-Sensitive Languages: Revisiting Landin's Offside Rule // ACM SIGPLAN Notices. Vol. 48, Is. 1. P. 511–522. (дата обращения: 30.08.2022).
11. Ramu C., Gemünd C., Gibson T. J. Object-Oriented Parsing of Biological Databases with Python // Bioinformatics. 2000. Vol. 16, Is. 7. P. 628–638. URL: <https://academic.oup.com/bioinformatics/article/16/7/628/2279282?login=false> (дата обращения: 30.08.2022).
12. Chapagain A. Hands-On Web Scraping with Python: Perform Advanced Scraping Operations Using Various Python Libraries and Tools Such as Selenium, Regex, and Others. 2019. 189 p. URL: https://books.google.ru/books?hl=ru&lr=&id=j-6iDwAAQBAJ&oi=fnd&pg=PP1&dq=parsing+python+selenium&ots=J9F2ScFdG8&sig=y0U5WR-xLhvNR3bGpJdAOyDrXR8&redir_esc=y#v=onepage&q=parsing%20python%20selenium&f=false (дата обращения: 30.08.2022).
13. ExportBase. URL: <https://export-base.ru/> (дата обращения: 30.08.2022).
3. Gosudarstvennyi informatsionnyi resurs bukhgalterskoi (finansovoi) otchetnosti. URL: <https://bo.nalog.ru> (accessed: 30.08.2022). (In Russian).
4. Vserossiiskaia sistema proverki kontragentov. URL: <https://zachestnyibiznes.ru> (accessed: 30.08.2022). (In Russian).
5. Spektr-audit. URL: <https://spektr-audit.com> (accessed: 30.08.2022). (In Russian).
6. Bukhgalterskaia (finansovaia) otchetnost predpriatii. URL: <https://e-ecolog.ru/buh> (accessed: 30.08.2022). (In Russian).
7. Sravnenie finansovogo sostoiianiia firmy s otraslevymi pokazateliami i konkurentami. URL: <https://www.testfirm.ru> (accessed: 30.08.2022). (In Russian).
8. Mitchell R. Web Scraping with Python. 2nd Ed. 2018. 288 p.
9. Chromdriver. URL: <https://chromedriver.chromium.org/downloads> (accessed: 30.08.2022).
10. Adams M. D. Principled Parsing for Indentation-Sensitive Languages: Revisiting Landin's Offside Rule // ACM SIGPLAN Notices. Vol. 48, Is. 1. P. 511–522. (accessed: 30.08.2022).
11. Ramu C., Gemünd C., Gibson T. J. Object-Oriented Parsing of Biological Databases with Python // Bioinformatics. 2000. Vol. 16, Is. 7. P. 628–638. (accessed: 30.08.2022).
12. Chapagain A. Hands-On Web Scraping with Python: Perform Advanced Scraping Operations Using Various Python Libraries and Tools Such as Selenium, Regex, and Others. 2019. 189 p. URL: https://books.google.ru/books?hl=ru&lr=&id=j-6iDwAAQBAJ&oi=fnd&pg=PP1&dq=parsing+python+selenium&ots=J9F2ScFdG8&sig=y0U5WR-xLhvNR3bGpJdAOyDrXR8&redir_esc=y#v=onepage&q=parsing%20python%20selenium&f=false (accessed: 30.08.2022).
13. ExportBase. URL: <https://export-base.ru/> (accessed: 30.08.2022). (In Russian).

Информация об авторах

С. О. Крамаров – доктор физико-математических наук, профессор.

В. А. Овсянников – магистрант.

Л. В. Сахарова – доктор физико-математических наук, доцент.

Р. С. Усатый – аспирант.

Г. В. Лукьянова – кандидат технических наук, доцент.

Information about the authors

S. O. Kramarov – Doctor of Sciences (Physics and Mathematics), Professor.

V. A. Ovsyannikov – Master's Degree Student.

L. V. Sakharova – Doctor of Sciences (Physics and Mathematics), Associate Professor.

R. S. Usatyi – Postgraduate.

G. V. Lukyanova – Candidate of Sciences (Engineering), Associate Professor.

Научная статья
УДК 519.2
doi: 10.34822/1999-7604-2022-3-46-56

ПРИМЕНЕНИЕ БАЙЕСОВСКИХ МЕТОДОВ ДЛЯ ОПТИМИЗАЦИИ ОБУЧЕНИЯ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ ТЕСТИРОВАНИЯ ПРИЛОЖЕНИЙ

Павел Валерьевич Полухин

Воронежский государственный университет, Воронеж, Россия
alfa_force@bk.ru, <https://orcid.org/0000-0001-5645-6312>

Аннотация. Нейросетевые модели являются универсальным инструментом для решения вероятностных задач в различных сферах и областях деятельности. Однако при решении ряда задач применение классических нейросетевых моделей и алгоритмов обучения может приводить к переобучению сети, вследствие чего происходит снижение точности апостериорного распределения выходного слоя нейросети и наблюдается слабая адаптивность сети к самообучению на основе ранее непредставленных данных в обучающей выборке. Тем самым снижаются возможности сети по выявлению аномалий. Одним из методов решения данной проблемы является использование байесовских методов для обучения и вероятностного вывода в нейросетях. При использовании байесовского подхода в нейросетях подразумевается задание весов сети в виде вероятностного распределения по всем допустимым значениям, которые могут принимать параметры сети. Тогда для определения весов можно использовать классические механизмы обучения и вероятностного вывода, которые используются в байесовских сетях, в частности алгоритмы на основе метода Монте-Карло и цепей Маркова. В работе рассмотрены вопросы применения байесовских нейросетей для моделирования процесса тестирования веб-приложений.

Ключевые слова: метод Монте-Карло, цепь Маркова, метод максимального правдоподобия, максимум апостериорной вероятности, вариационный вывод, байесовская нейронная сеть, многослойный перцептрон, методы тестирования

Для цитирования: Полухин П. В. Применение байесовских методов для оптимизации обучения нейросетевых моделей тестирования приложений // Вестник кибернетики. 2022. № 3 (47). С. 46–56. DOI 10.34822/1999-7604-2022-3-46-56.

Original article

USING BAYESIAN METHODS TO OPTIMIZE TRAINING OF NEURAL NETWORK MODELS FOR APPLICATION TESTING

Pavel V. Polukhin

Voronezh State University, Voronezh, Russia
alfa_force@bk.ru, <https://orcid.org/0000-0001-5645-6312>

Abstract. Neural network models are a universal tool for solving probabilistic problems in various fields and areas. However, when solving a set of problems, typical neural networks and training algorithms can lead to overfitted networks, resulting in a decrease in the accuracy of the posterior distribution of the output layer of the neural network. In this case, the network fails to adapt fully to self-training due to no data in the training sample, leading to a decrease in the network's ability to detect anomalies. One of the methods to solve the problem is by using Bayesian methods for training of and inference in neural networks. When using the Bayesian approach in neural networks, it is intended to set the weights in the network as a probability distribution over all values that are permissible for the parameters of the network. Then, classical learning mechanisms and probabilistic inference used in Bayesian networks, such as algorithms based on the Monte Carlo methods and Markov chains, can be used to determine weights. The article considers the issues of using Bayesian neural networks for modeling the process of web application testing.

Keywords: Monte Carlo method, Markov chain, maximum likelihood method, maximum posterior probability, variation inference, Bayesian neural network, multilayer perceptron, testing methods

For citation: Polukhin P. V. Using Bayesian Methods to Optimize Training of Neural Network Models for Application Testing // *Proceedings in Cybernetics*. 2022. No. 3 (47). P. 46–56. DOI 10.34822/1999-7604-2022-3-46-56.

ВВЕДЕНИЕ

Применение байесовского подхода для решения задач обучения и вероятностного вывода является общепринятым. Наряду с классическими вероятностными графическими моделями, особый интерес представляет возможность применения байесовского подхода к нейронным сетям и формирование байесовской нейронной сети, позволяющей исключить недостатки классических нейросетей общего назначения. Первое полное описание байесовских нейронных сетей приведено в работах Нила [1] и Маккея [2], рассматривающих возможности оптимизации обучения нейронных сетей на основе применения байесовского подхода. Основное преимущество таких сетей заключается в вероятностном определении весов между слоями нейронной сети. Такой подход позволяет учитывать априорное распределение по всем выходным параметрам в процессе обучения сети и вычисления апостериорного распределения в соответствии с генерируемыми предположениями (выборками). Использование различных подходов формирования выборок, основывающихся на методе Монте-Карло с применением цепей Маркова, позволяет сократить объем данных, необходимых для обеспечения требуемого уровня обученности сети, а также исключить вероятность ложного срабатывания. Процедура обучения параметров происходит за счет вероятностного вывода с использованием байесовского вывода. Для оптимизации процедуры обучения байесовской нейронной сети возникает необходимость исследования новых алгоритмов вероятностного вывода. Хинтоном и Ван Кампом предложен один из таких алгоритмов, который основывается на методе вариационного вывода. Сущность данного метода заключается в определении вероятностного распределения для весов из всех слоев, учитывая факт условной независимости весов каждого из слоев. Применение такого подхода позволяет учитывать веса всех слоев в процессе формирования результиру-

ющих выборок байесовских нейронных сетей (далее – БНС) и повысить точность процедуры вариационного вывода. На практике при моделировании БНС могут рассматриваться различные последовательные алгоритмы Монте-Карло с применением цепей Маркова, в частности алгоритм фильтрации частиц. Каждая последующая выборка будет формироваться с учетом предшествующей выборки и пропорционально ее весу. Тогда переход между слоями можно будет представить в виде однородного марковского процесса.

МАТЕРИАЛЫ И МЕТОДЫ

Современный подход к обучению БНС основывается на различных алгоритмах, основанных на методе вариационного вывода и генерации выборок. Специфика примененного вариационного вывода (далее – ВВ) предложена Хинтоном для обучения нейронной сети типа «многослойный перцептрон», имеющей неограниченное число скрытых слоев. Типовая нейронная сеть в виде многослойного перцептрона представлена на рис. 1, где видно, что X – входной слой, Y – выходной слой, $H = (H_1, H_2, \dots, H_n)$ – множество скрытых слоев нейронной сети. Основным подходом, используемым при построении нейронной сети, заключается в вычисленных выходных параметрах Y с учетом X и H .

Для установления связи между данными параметрами используется активационная функция.

В качестве примера такой функции рассмотрим гиперболический тангенс:

$$H(X) = \tanh\left(a + \sum_{i=1}^n W_i X_i\right), \quad (1)$$

где W_i – вес связи между входным параметром X_i и одним из параметров скрытого слоя H_{ij} ;

a – смещение, соответствующее параметрам слоев H .

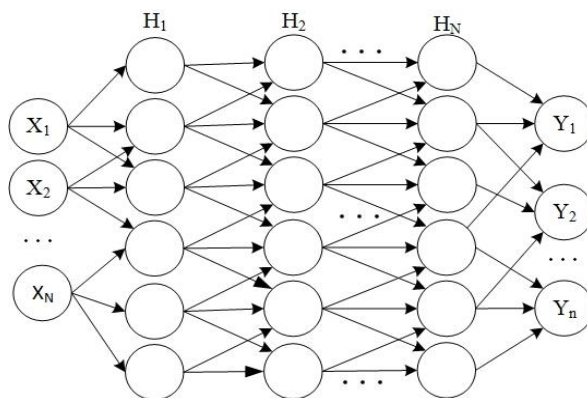


Рис. 1. Пример нейронной сети типа «многослойный перцептрон»
 Примечание: составлено по [1].

В соответствии с активационной функцией происходит вычисление значений весов для каждого из параметров скрытого слоя H_{ij} с учетом предыдущего слоя. Модель нейронной сети, представленной на рис. 1, может быть использована для решения задач регрессии и классификации. Обе задачи предусматривают установление связи между входным и выходным набором параметров нейронной сети при условии наличия выборки $D = \{(X^i, Y^i)\}_{i=1}^N$. Решение задачи регрессии может быть задано в виде поиска условного распределения $P(Y|X)$, которое, в случае гауссовского распределения входных параметров X_i , будет иметь следующий вид [3]:

$$P(Y|X) = \prod_{k=1}^n \frac{1}{\sqrt{2\pi}\sigma_k} e^{-\frac{(f_k(X)-Y_k)^2}{2\sigma_k^2}}, \quad (2)$$

$$f_k(X) = b + \sum_{i=1}^n W_i H_i(X), \quad (3)$$

где b – смещение для всех параметров из H .

Решение задачи классификации для идентификации k -объектов, согласно Нилу, может быть сведено к поиску распределения $P(Y=k|X)$. Тогда получим:

$$P(Y=k|X) = e^{f_k(X)} / \sum_K e^{f_K(X)}, K \subseteq Y. \quad (4)$$

Определение параметров $P(Y|X)$ и $P(Y=k|X)$ реализуется в нейронной сети за счет обучения на основе выборки D . В таком

случае для определения весов W можно воспользоваться классическими методами математической статистики: максимального правдоподобия (далее – МП) и максимума апостериорной вероятности (далее – МАВ). Выражение для вычисления весов W^{ML} на основе МП имеет следующий вид:

$$W^{ML} = \arg \max_W \log P(D|W) = \arg \max_W \sum_{i=1}^n \log P(Y_i | X_i, W_k). \quad (5)$$

В таком случае выражение для МАВ W^{MAP} имеет следующий вид:

$$W^{MAP} = \arg \max_W \log P(W|D) = \arg \max_W \log P(D|W) + \log P(W). \quad (6)$$

Для решения уравнений (4) и (5) можно воспользоваться различными численными методами, в частности методом градиентного спуска, а также одним из квазиньютоновских алгоритмов. В общем случае задача сводится к поиску максимального значения $\log(f(x))$. Рассмотрим решение данных уравнений методом Бройдена – Флетчера – Гольдфарба – Шанно (далее – БФГШ). На каждом шаге алгоритма БФГШ используется процедура разложения оценочной функции на полином вида:

$$P(Y=k|X) = e^{f_k(X)} / \sum_K e^{f_K(X)}, K \subseteq Y, \quad (7)$$

где λ – шаг сдвига алгоритма.

Основная идея алгоритма БФГШ заключается в расчете приближенной матрицы Гессе $B(x_k)$ вместо прямого вычисления $H(x_k)$. Тогда значение параметра λ можно выразить через соответствующую приближенную матрицу $B(x_k)$:

$$\lambda_k = -B^{-1}(x_k) \nabla f(x_k). \quad (8)$$

Далее рассмотрим вычисление приближенной матрицы Гессе. Для этого воспользуемся условием Вольфе:

$$\begin{aligned} f(x_k + \alpha_k \lambda_k) &\leq f(x_k) + c_1 \alpha_k \nabla f_k^T(x) \lambda_k, \\ \nabla f(x_k + \alpha_k \lambda_k)^T &\geq c_2 \nabla f_k^T(x) \lambda_k, \end{aligned} \quad (9)$$

где c_1 и c_2 – константы, соответствующие условиям $0 < c_1 < c_2$.

Если приближение функции $f(x)$ будет находиться в пределах локального экстремума, то будет справедливо строгое условие Вольфе:

$$\begin{aligned} f(x_k + \alpha_k \lambda_k) &\leq f(x_k) + c_1 \alpha_k \nabla f_k^T(x) \lambda_k, \\ |\nabla f(x_k + \alpha_k \lambda_k)^T| &\leq c_2 |\nabla f_k^T(x) \lambda_k|. \end{aligned} \quad (10)$$

Следовательно, каждый следующий шаг для x_k будет удовлетворять условию (10).

$$x_{k+n} = x_k + \alpha_k \lambda_k. \quad (11)$$

Тогда для определения окончательного алгоритма БФГШ рассмотрим метод Дэвидсона – Флетчера – Пауэлла для расчета приближенной матрицы Гессе $B_{x_{k+n}}$, являющейся частью алгоритма БФГШ:

$$\begin{aligned} B_{x_{k+n}} &= (I - \delta_{k+n-1} s_{k+n-1} y_{k+n-1}^T) \times M_{x_{k+n-1}} \times \\ &\times (I - \delta_{k+n-1} y_{k+n-1} s_{k+n-1}^T) + \delta_{k+n-1} s_{k+n-1} s_{k+n-1}^T, \end{aligned} \quad (12)$$

где $\delta_{k+n-1} = 1 / y_{k+n-1} s_{k+n-1}^T$;

$$\begin{aligned} s_{k+n-1} &= x_{k+n} - x_{k+n-1}; \\ y_{k+n-1} &= \nabla f(x_{k+n}) - \nabla f(x_{k+n-1}). \end{aligned}$$

Обозначим обратную матрицу Гессе как $H_{x_{k+n}} = B_{x_{k+n}}^{-1}$. Тогда, воспользовавшись тождеством Шермана – Моррисона – Вудбери,

можно получить приближенное значение инверсной матрицы $H_{x_{k+n}}$:

$$\begin{aligned} H_{k+n} &= H_{k+n-1} - H_{k+n-1}' + \frac{y_{k+n-1} y_{k+n-1}^T}{y_{k+n-1}^T s_{k+n-1}}, \\ H_{k+n-1}' &= \frac{H_{k+n-1} s_{k+n-1} s_{k+n-1}^T H_{k+n-1}}{s_{k+n-1}^T H_{k+n-1} s_{k+n-1}}. \end{aligned} \quad (13)$$

Отметим, что обратная матрица Гессе H_{k+n-1} может быть выражена через соответствующую матрицу Якоби. Это позволяет исключить необходимость расчета вторых производных для функции $f(x)$ на каждом шаге:

$$H_{k+n} = J_{k+n-1}^T J_{k+n-1} + H_{k+n-1}. \quad (14)$$

Из выражения (13) получим важное следствие, заключающееся в том, что матрица H_{k+n} будет положительно определенной только в том случае, если H_{k+n-1} также является положительно определенной. Данное следствие – достаточно важное, так как является одним из необходимых условий для определения направленности поиска.

Для определения искомого выражения расчета H_{k+n} на основе алгоритма БФГШ введем ограничения относительно матрицы H_{k+n} : матрица должна быть симметричной и положительно определенной, должна удовлетворять уравнению секущих $M_{k+n} y_{k+n-1} = s_{k+n-1}$. Тогда алгоритм БФГШ, с учетом выражения (13), описывается следующим выражением:

$$\begin{aligned} H_{x_{k+n}} &= (I - \delta_{k+n-1} s_{k+n-1} y_{k+n-1}^T) \times H_{x_{k+n-1}} \times \\ &\times (I - \delta_{k+n-1} y_{k+n-1} s_{k+n-1}^T) + \delta_{k+n-1} s_{k+n-1} s_{k+n-1}^T, \end{aligned} \quad (15)$$

где $\delta_{k+n-1} = 1 / y_{k+n-1} s_{k+n-1}^T$.

БФГШ является достаточно эффективным алгоритмом для решения задач поиска экстремума функции $f(x)$. Однако на практике применение для методов максимального правдоподобия и максимума апостериорной вероятности при обучении нейронной сети часто приводит к ее переобучению и снижает в целом эффективность решения задач ре-

грессии и классификации, несмотря на достаточно точное решение задач поиска максимального значения с использованием методов МП и МАВ.

Альтернативным для преодоления проблем методов МП и МАВ является байесовский подход в процессе обучения нейронной сети. Нейронные сети с вероятностным распределением весов получили название «байесовские нейронные сети». Для данного типа нейронных сетей весам соответствует распределение вероятностей $P(W_{n+1}|D)$, формируемое на основе обучающей выборки D . В таком случае апостериорное распределение по всем весам W при наличии свидетельства относительно X и Y задается в виде условного распределения $P(W|X, Y)$. Полное совместно апостериорное распределение для байесовской нейронной сети будет иметь следующий вид:

$$\begin{aligned} P(Y_{n+1}|X_{n+1}, (X_{1:n}, Y_{1:n})) &= \\ &= \int P(X, Y, W) dW_{n+1}, P(X, Y, W) = \\ &= P(Y_{n+1}|X_{n+1}, W_{n+1}) P(W_{n+1} | X_{1:n}, Y_{1:n}). \end{aligned} \quad (16)$$

Тогда распределение вероятностей $P(W_{n+1} | X_{1:n}, Y_{1:n})$ может быть найдено в соответствии с формулой Байеса:

$$\begin{aligned} P(W_{n+1}|D) &= \\ &= \frac{P(D_n|W_{n+1})P(W_{n+1})}{P(D_n)} = \prod_{i=1}^n P(Y_i | X_i, W_k). \end{aligned} \quad (17)$$

Тогда в соответствии с выражением (15) процедура нормализации распределения $P(Y_{n+1}|X_{n+1})$ имеет следующий вид:

$$\begin{aligned} P(Y_{n+1}|X_{n+1}) &= \\ &= \int P(Y_{n+1}|X_{n+1}, W_{n+1}) P(W_{n+1}) dW_{n+1}. \end{aligned} \quad (18)$$

В таком случае оценку параметров выходного слоя БНС, соответствующего шагу, можно определить согласно следующему интегралу:

$$\hat{Y}_{n+1} = \int f(x, W) P(W, D) dW, \quad (19)$$

где $f(x, W)$ – функция выходов Y_{n+1} , имеющая непосредственную зависимость от весов W .

Для нахождения $P(W|X, Y)$ воспользуемся методом вариационного вывода (далее – ВВ), имеющим сходство с методом МП, однако вместо логарифма используется дистанция Кульбака – Лейблера (далее – КЛ). Для формулировки метода ВВ требуется задать приближенное вариационное распределение по всем весам $Q(W)$. Степень близости распределения $Q(W)$ к $P(W|X, Y)$ будет обеспечиваться за счет расчета дистанции КЛ. Определим выражение дистанции КЛ в следующем виде [4]:

$$\begin{aligned} D_{KL}(Q(W), P(W|X, Y)) &= \\ &= \int Q(W) \log \frac{Q(W)}{P(W|X, Y)} dW = \\ &= \int Q(W) \log Q(W) dW - \int Q(W) P(W|X, Y) dW = \\ &= E_W \left(\log \frac{Q(W)}{P(W)} \right) - E_W(P(W|X, Y)) + \log P(W). \end{aligned} \quad (20)$$

С учетом того, что дистанция D_{KL} является строго неотрицательной, можно получить следующее неравенство:

$$\begin{aligned} D_{KL}(Q(W), P(W|X, Y)) - \\ - E_W(P(W|X, Y)) + \log P(W) \geq 0, \\ \log P(W) \geq E_W \left(\log \frac{Q(W|X, Y)}{P(W)} \right). \end{aligned} \quad (21)$$

Введем следующие обозначения для вариационно-независимой функции:

$$\begin{aligned} F(D, Q) &= D_{KL}(Q(W), P(W|X, Y)) - \\ &- E_W(\log P(D|W)). \end{aligned} \quad (22)$$

Тогда с учетом дистанции КЛ и функции (19) можно получить оценку весов $\hat{W}$ на основе вариационного распределения $Q(W)$:

$$\hat{W} = \arg \min_W F(D, Q). \quad (23)$$

В таком случае оценка $\hat{W}$ для $F(D, Q)$ будет являться оценкой максимального правдоподобия, ее расчет можно выполнить

по аналогии с классической нейронной сетью. Для этого достаточно воспользоваться описанным ранее квазиньютоновским алгоритмом БФГШ и оценочной функцией $F(D, Q)$.

Алгоритм ВВ является точным алгоритмом байесовского вывода, используемого для обучения БНС. Основным его недостатком является повышение временной сложности алгоритма в случае увеличения слоев нейронной сети, в частности общего числа скрытых слоев. В связи с этим наибольший интерес представляют стохастические алгоритмы на основе байесовского вывода и на основе метода Монте-Карло. В основе метода Монте-Карло используется порождение случайных значений переменных БНС в соответствии с заданной функцией распределения. Особый интерес представляет возможность формирования выборок для весов W в соответствии с моделью перехода между скрытыми слоями БНС. Обозначим $Q(W)$ как апостериорное распределение вероятности (распределение по значимости) в момент перехода между слоями БНС. Тогда математическую формулировку метода Монте-Карло запишем в следующем виде:

$$Ef(x) = \int_0^1 f(x, W) Q(x) dx \approx \frac{1}{N} \sum_{i=1}^N f(x_i, W_i). \quad (24)$$

Так, из (23) следует, что W_i формируется из распределения по значимости $Q(W)$. На практике особую значимость представляют методы, основанные на применении метода Монте-Карло совместно с цепью Маркова (далее – МКМЦ). Такой подход обладает одним важным свойством по сравнению с методом Монте-Карло – позволяет накапливать выборки, то есть каждая следующая генерация выполняется путем случайного изменения уже существующей. Серия таких случайных событий образует цепь Маркова в соответствии со стационарным распределением $Q(W)$. Тогда переходная вероятность будет задаваться в виде распределения $P(W_{i+1} | W_i)$. Алгоритм вычисления апостериорного распределения на основе ме-

тода МКМЦ выполняется путем разворачивания цепи Маркова на n шагов, характеризующихся состояниями W_{i+1} и W_i с соответствующими вероятностями $P(W_{i+1})$ и $P(W_i)$. Определим цепь Маркова для случайного распределения $P(W_{i+1} | W_i)$, которая будет иметь следующий вид [5]:

$$P(\xi(t_{i+1}) = W_{i+1} | \xi(t_1) = W_1, W_1, \dots, \xi(t_i) = W_i) = P(\xi(t_{i+1}) = W_{i+1} | \xi(t_i) = W_i), \quad (25)$$

где распределение $Q(W)$ будет сходиться к исходному распределению $P(W)$ в силу закона больших чисел при $N \rightarrow \infty$. В таком случае точность алгоритма будет зависеть от N . Условие стационарности для распределения $P(W)$ будет иметь следующий вид:

$$P(W_{i+1}) = \int P(W_{i+1} | W_i) P(W_i) dW_i \forall W_{i+1} \in W. \quad (26)$$

Достаточное условие стационарности для распределения $P(W_i)$ будет справедливо только в том случае, если выполняется принцип детального равновесия [6].

$$P(W_{i+1}) = \int P(W_{i+1} | W_i) P(W_i) dW_i \forall W_{i+1} \in W. \quad (27)$$

На методе МКМЦ основаны алгоритмы Метрополиса – Гастингса (далее – МГ), Гиббса. Наряду с этими общераспространенными, рассмотрим гибридные алгоритмы на основе динамики Ланжевена и Гамильтона. Приведем алгоритм МГ, который также используется в гибридных алгоритмах в качестве условия принятия или отклонения выборки. Запишем условие, соответствующее оценке принятия (отбрасывания) выборки на основе алгоритма МГ [7]:

$$E_\phi \min \{1, \phi\} = \int dW_i dW_{i+1} \min(1, \Omega), \quad \Omega = \frac{P(W_{i+1}) P(W_i | W_{i+1})}{P(W_i) P(W_{i+1} | W_i)}, W_i \neq W_{i+1}, \quad (28)$$

$$\phi(W_{i+1} | W_i) = \frac{P(W_{i+1}) P(W_i | W_{i+1})}{P(W_i) P(W_{i+1} | W_i)}, \quad (29)$$

$$W_i \sim P(W_i), W_{i+1} \sim P(W_{i+1} | W_i).$$

Тогда в соответствии с (26) докажем условие стационарности $P(W_i, W_{i+1})$:

$$\begin{aligned} \min \left(\frac{P(W_i)P(W_{i+1}|W_i)}{P(W_{i+1})P(W_{i+1}|W_i)} \right) \varphi(W_{i+1}|W_i) = \\ = \min \left(\frac{P(W_{i+1})P(W_{i+1}|W_i)}{P(W_i)P(W_{i+1}|W_i)} \right) = \\ = P(W_{i+1})P(W_{i+1}|W_i) \varphi(W_{i+1}|W_i). \end{aligned} \quad (30)$$

Тогда, с учетом того, что распределение $P(W_i)$ является стационарным, выражение (28) перепишем в следующем виде [8, 9]:

$$\varphi(W_{i+1}|W_i) = \frac{P_0(W_{i+1}) \prod_{i=1}^n P(X_i, Y_i | W_{i+1}) P(W_i | W_{i+1})}{P_0(W_i) \prod_{i=1}^n P(X_i, Y_i | W_i) P(W_{i+1} | W_i)}. \quad (31)$$

В соответствии с выражением (30) определим условие принятия (отбрасывания) выборки на каждом шаге алгоритма МГ с учетом равномерного распределения $U(0,1)$:

$$W_i = \begin{cases} W_{i+1}, \varphi(W_{i+1}|W_i) > U(0,1) \\ W_i, \varphi(W_{i+1}|W_i) \leq U(0,1) \end{cases}. \quad (32)$$

Из выражения (31) видно, что при формировании условия с учетом распределения $U(0,1)$ на практике будем наблюдать достаточно большой разброс выборок. Как следствие, для достижения требуемого уровня точности требуется генерация достаточно большого числа выборок при $N \rightarrow \infty$. Для решения данной проблемы в работах Барденета [10] были сделаны попытки уменьшения области разброса выборок за счет использования оценки правдоподобия с нормализацией. Запишем оценку правдоподобия в соответствии с выражением (30):

$$L(W_{i+1}, W_i) = \frac{1}{N} \sum_{i=1}^N \log \left(\frac{P(X_i, Y_i | W_{i+1})}{P(X_i, Y_i | W_i)} \right). \quad (33)$$

Тогда вместо распределения $U(0,1)$ в качестве критерия правдоподобия соответствия выборки будем использовать функцию распределения $\xi(U, W_{i+1}, W_i)$:

$$\xi(W_{i+1}, W_i) = \frac{1}{N} \log \left(U(0,1) \frac{P(W_i)P(W_{i+1}|W_i)}{P(W_{i+1})P(W_i|W_{i+1})} \right). \quad (34)$$

Используя выражения (32) и (33), перепишем условие (31) в следующем виде:

$$W_i = \begin{cases} W_{i+1}, L(W_{i+1}, W_i) > \xi(W_{i+1}, W_i) \\ W_i, \varphi(W_{i+1}|W_i) \leq \xi(W_{i+1}, W_i) \end{cases}. \quad (35)$$

Использование критерия на основе оценки правдоподобия позволяет сузить область разброса выборок, однако окончательно решить задачу генерации согласованных выборок не может. В связи с этим особую значимость приобретают алгоритмы на основе динамики Ланжевена, довольно часто используемой для описания различных физических процессов. Тогда запишем классическое уравнение динамики Ланжевена для Броуновского движения в дифференциальном виде [11]:

$$\begin{aligned} dP(W_i) = \frac{1}{2} \nabla L(P(W_i)) dW_i + d\xi(W_i), \\ i \in [0, \infty), \end{aligned} \quad (36)$$

где $\xi(W_i)$ – производная случайного Броуновского процесса, $L(P(W_i)) = \log P(W_i)$.

Тогда, решая уравнение (35) методом Эйлера, получим:

$$\begin{aligned} P(W_{i+1}) = P(W_i) + \frac{1}{2} \nabla L(P(W_i)) + \\ + \sqrt{\epsilon} \pi_i, \pi_i \sim N(\mu, I), \end{aligned} \quad (37)$$

где I – единичная ковариационная матрица, соответствующая многомерному нормальному распределению $N(0, I)$.

В таком случае, подставляя выражения динамики Ланжевена, соответствующие распределениям $P(W_{i+1})$ и $P(W_i)$, в выражение (34), можно получить обновленное условие получения согласованной выборки по методу Метрополиса – Гастингса. В свою очередь распределение $P_n(W_{i+1}|W_i)$ для всех переходов N запишем в виде следующего произведения [12]:

$$P_N(W_{i+1}|W_i) = \prod_{i=1}^N P(W_{i+1}|W_i). \quad (38)$$

Далее рассмотрим применение БНС, а также описанных выше алгоритмов обучения для моделирования процессов тестирования веб-приложений. Для этого исследуем общую схему тестирования веб-приложений на основе БНС. Процедура тестирования на этапе обучения реализуется методом «черного ящика». В данном случае структура веб-приложения неизвестна, а каждый последующий тест генерируется на основе случайного изменения предыдущего. Опишем основные особенности тестирования ошибок межсайтового скриптинга (далее – МСС). МСС представляет собой ошибку в веб-

приложении, связанную с некорректной фильтрацией входных данных. В таком случае в веб-приложении могут передаваться конструкции языка JavaScript, которые затем передаются в веб-браузер клиента с целью его компрометации. Среди МСС-ошибок выделяют три основных типа: отраженная, хранимая и dom. Рассмотрим типовую схему тестирования межсайтового скриптинга в процессе анализа веб-приложений на предмет наличия ошибок на основе сети БНС.

Из рис. 2 видно, что БНС в процессе обучения производит анализ успешных МСС-тестов, показавших факт присутствия МСС, а также возможность выполнения эксплуатации данных ошибок.

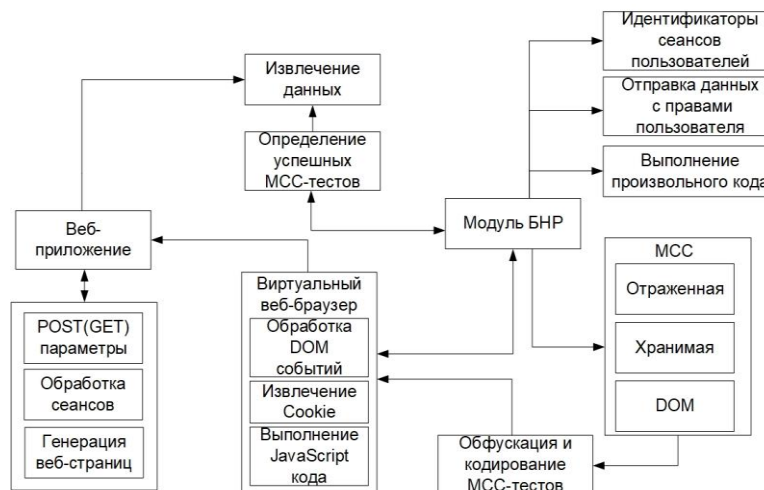


Рис. 2. Модель тестирования МСС на основе БНС

Примечание: составлено автором на основании данных, полученных в исследовании.

В режиме обучения БНС производится случайная генерация тестовых данных с учетом особенностей анализа МСС-ошибок, включая синтаксические правила языка JavaScript. Тестовые генерации подаются на вход группы разнородных приложений $A = (A^1, A^2, \dots, A^n)$, после чего анализируются выходные данные приложения. Под выходными данными подразумевается факт присутствия МСС, а также выполнение полезной нагрузки. В этом случае входная последовательность тестов, а также результаты тестирования используются для обучения БНС. Для процедуры обучения входной информацией является выбор-

ка $D = \{(X^i, Y^i)\}_{i=1}^N$, $X = (X^1, X^2, \dots, X^N)$ – множество тестовых генераций, $Y = (Y^1, Y^2, \dots, Y^N)$ – информация относительно генераций, успешно прошедших тестирование. Из этого следует, что БНС накапливает информацию исключительно относительно успешно завершившихся тестовых генераций. Для более точного обучения БНС для тестирования МСС-ошибок производится настройка блоков генерации тестов с учетом различных платформ веб-браузеров: Gecko, Chromium.WebKit. Такой подход позволяет производить обучение БНС с учетом особен-

ностей обработки параметров каждой из таких платформ, позволяя выполнять процедуру более осмысленно и точно, по аналогии с тем, как это делает тестировщик в процессе анализа веб-приложения. В результате разработанная модель БНС может работать как в режиме самообучения, так и обучения с учителем. Такой подход дает возможность непрерывной коррекции тестовых генераций и оптимизации процедуры обучения модели тестирования, а также адаптации сети к возникновению аномальных ошибок в процессе реализации процедуры тестирования.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Таким образом, в результате исследования разработаны гибридные алгоритмы обучения БНС на основе динамики Ланжевена, рассмотрены возможные пути их адаптации для решения задач тестирования прикладного программного обеспечения, представленного в виде веб-приложений. Для построения среды тестирования модели, приведенной на рис. 1, а также для реализации процедур обучения БНС, была развернута группа веб-приложений (1С Битрикс 16.5, 16.5.4, WordPress 4.0, 4.1, Joomla 1.5.15, 2.5), включающих в себя средства хранения, обработки поступающей информации, а также имитирующих работу реальных веб-приложений. В качестве платформы для запуска виртуальных веб-браузеров используется Selenium GRID, представляющий собой распределенную платформу, предназначенную для решения задачи оптимизации тестирования веб-приложений динамической обработкой

и отображением данных. Отличительной особенностью Selenium является возможность программного доступа к DOM-модели документа, а также к адресному пространству переменных языка JavaScript и его функциональным возможностям. В качестве среды развертывания Selenium GRID используем среду Docker, запускающую отдельные экземпляры веб-браузеров Firefox, Chromium и Edge. Взаимодействие с Selenium производится на основе вызова удаленных процедур, позволяющих производить поэтапное выполнение тестов для обнаружения ошибок межсайтового скриптинга. В процессе выполнения эксперимента приведем оценки вычислительной эффективности классических алгоритмов обучения БНС на основе вариационного вывода, Метрополиса – Гастингса, а также разработанного алгоритма на основе динамики Ланжевена.

Из сравнительных характеристик алгоритмов табл. 1 видно, что алгоритм, разработанный на основе динамики Ланжевена, обладает наилучшими показателями. Это, в первую очередь, связано с тем, что динамика Ланжевена позволяет ограничить область согласованных выборок, то есть в процессе каждого последующего шага в соответствии с выражением (35) производится постепенное смещение формируемых выборок к области истинных значений. Следовательно, апостериорное распределение вероятностей по всем выборкам $P(W|X, Y)$, формируемое по результатам реализации процедуры тестирования, будет обладать наибольшей точностью.

Таблица 1

Сравнение производительности алгоритмов обучения БНС

№	Объем выборки	Алгоритм ВВ	Алгоритм МГ	Алгоритм ДЛ
1	1000	11,114263 сек.	14,742326 сек.	10,761308 сек.
2	100000	42,800446 сек.	56,590105 сек.	40,128704 сек.
3	1000000	114,214165 сек.	147,883367 сек.	92,710642 сек.
4	100000000	1151,117164 сек.	1484,762449 сек.	855,414459 сек.

Примечание: составлено автором на основании данных, полученных в исследовании.

Отличительной особенностью разработанного алгоритма является применение критерия (34), сочетающего в себе критерий согласованности на основе логарифма правдоподобия и подхода, предложенного Метрополисом. Это позволяет использовать

распределение $L(W_{i+1}, W_i)$ взамен классического равномерного распределения $U(0,1)$, что повышает степень согласованности формируемых выборок.

ЗАКЛЮЧЕНИЕ

Моделирование процессов тестирования является сложной и многогранной задачей, требующей создания различного инструментария тестирования, средств накопления статистической информации и интеллектуального анализа данных. Байесовские нейронные сети являются универсальным механизмом моделирования сложных процессов, протекающих в условиях определенности, расширяющим классические модели нейронных сетей за счет использования байесовских методов. Особый научно-практический интерес представляют расширение и оптимизация алгоритмов на основе метода Монте-Карло с применением цепей Маркова. Рассмотренный подход на основе использования динамики Ланжевена в сочетании с логарифмическим правдоподобием позволяют использовать преимущества обоих подходов для по-

вышения степени согласования выборок, формируемых в соответствии с распределениями $P(W_i | W_{i+1})$. Предложенный алгоритм обладает хорошей масштабируемостью. Кроме того, разработан параллельный алгоритм, позволяющий оптимизировать генерацию выборок и снизить временные затраты на реализацию процедур обучения. Использование БНС в процессе тестирования веб-приложений позволяет оптимизировать процедуру обучения сети за счет повышения точности определения весов тестовых выборок для каждого скрытого слоя, адаптировать модель нейронной сети для обнаружения аномального поведения приложений. Вычислительный эксперимент и полученные результаты подтверждают правильность разработанных алгоритмов для моделирования тестирования веб-приложений.

Список источников

1. Skansi S. Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence. 2018. 191 p.
2. Зуев В. Н., Кемайкин В. Л. Модифицированные алгоритмы обучения нейронных сетей // Программные продукты и системы. 2019. Т. 4, № 2. С. 258–262.
3. Вершков Н. А., Кучуков В. А., Кучукова Н. Н. Теоретический подход к поиску глобального экстремума при обучении нейронных сетей // Тр. ИСП РАН. 2019. Т. 31, № 2. С. 41–52.
4. Russell S., Norvig P. Artificial Intelligence: A Modern Approach. Hoboken : Pearson, 2020. 1023 p.
5. Bunch P., Godsill S. Improved Particle Approximations to the Joint Smoothing Distribution Using Markov Chain Monte Carlo // IEEE Transactions Signal Processing. 2013. Vol. 61, Is. 4. P. 946–953.
6. Mohan K., Pearl J. Graphical Models for Processing Missing Data // Journal of American Statistical Association. 2018. Vol. 116, Is. 534. P. 1023–1037.
7. Азарнова Т. В., Полухин П. В., Аснина Н. Г., Проскурин Д. К. Формирование структуры байесовской сети процесса тестирования надежности информационных систем // Вестн. Воронеж. гос. техн. ун-та. 2017. Т. 13, № 6. 156 с.
8. Moral P. D., Doucet A. Particle Methods: An Introduction with Application // ESAIM. 2014. Vol. 44. P. 1–46.
9. Мельникова И. В., Сметанников Д. И. Исследование уравнений для вероятностных характеристик случайных процессов, заданных стохастическими уравнениями // Тр. ИММ УрО РАН. 2018. Т. 24, № 2. С. 185–193.
10. Bardenet R., Doucet A., Holmes C., Bardenet R. Towards Scaling up Markov Chain Monte Carlo:

References

1. Skansi S. Introduction to Deep Learning: From Logical Calculus to Artificial Intelligence. 2018. 191 p.
2. Zuev V. N., Kemaikin V. L. An Improved Neural Network Training Algorithm // Software & Systems. 2019. Vol. 4, No. 2. P. 258–262. (In Russian).
3. Vershkov N. A., Kuchukov V. A., Kuchukova N. N. The Theoretical Approach to the Search for a Global Extremum in the Training of Neural Networks // Proceedings of the ISP RAS. 2019. Vol. 31, No. 2. P. 41–52. (In Russian).
4. Russell S., Norvig P. Artificial Intelligence: A Modern Approach. Hoboken : Pearson, 2020. 1023 p.
5. Bunch P., Godsill S. Improved Particle Approximations to the Joint Smoothing Distribution Using Markov Chain Monte Carlo // IEEE Transactions Signal Processing. 2013. Vol. 61, Is. 4. P. 946–953.
6. Mohan K., Pearl J. Graphical Models for Processing Missing Data // Journal of American Statistical Association. 2018. Vol. 116, Is. 534. P. 1023–1037.
7. Azarnova T. V., Polukhin P. V., Asnina N. G., Proskurin D. K. Bayesian Network Structure Formation of Information Systems Reliability Testing Process // Bulletin of Voronezh State Technical University. 2017. Vol. 13, No. 6. 156 p. (In Russian).
8. Moral P. D., Doucet A. Particle Methods: An Introduction with Application // ESAIM. 2014. Vol. 44. P. 1–46.
9. Melnikova I. V., Smetannikov D. I. The Study of Equations for Probability Characteristics of Random Processes Described by Stochastic Equations // Tr. IMM UrO RAN. 2018. Vol. 24, No. 2. P. 185–193. (In Russian).
10. Bardenet R., Doucet A., Holmes C., Bardenet R. Towards Scaling up Markov Chain Monte Carlo:

An Adaptive Subsampling Approach // Proceedings of the 31st International Conference on Machine Learning, June 22–24, 2014, Beijing, China. 2014. Vol. 32, Is. 1. P. 405–413.

11. LeCun Y., Bengio G., Hinton G. Deep Learning // Nature. 2015. Vol. 521. P. 436–444.
12. Durmus A., Majewski S., Miasojedow B. Analysis of Langevin Monte Carlo via Convex Optimization // Journal of Machine Learning Research. 2019. Vol. 20. P. 1–46.

Информация об авторе

П. В. Полухин – кандидат технических наук.

An Adaptive Subsampling Approach // Proceedings of the 31st International Conference on Machine Learning, June 22–24, 2014, Beijing, China. 2014. Vol. 32, Is. 1. P. 405–413.

11. LeCun Y., Bengio G., Hinton G. Deep Learning // Nature. 2015. Vol. 521. P. 436–444.
12. Durmus A., Majewski S., Miasojedow B. Analysis of Langevin Monte Carlo via Convex Optimization // Journal of Machine Learning Research. 2019. Vol. 20. P. 1–46.

Information about the author

P. V. Polukhin – Candidate of Sciences (Engineering).

Научная статья

УДК 519.816

doi: 10.34822/1999-7604-2022-3-57-65

ВЫБОР КЛЮЧЕВЫХ ПАРАМЕТРОВ ДЛЯ ДИАГНОСТИКИ ЗАБОЛЕВАНИЯ ПЕЧЕНИ НА ОСНОВЕ МЕТОДА АНАЛИЗА ИЕРАРХИЙ

Александр Сергеевич Серобабов

Омский государственный технический университет, Омск, Россия

aserobabow95@mail.ru, <https://orcid.org/0000-0001-8850-0947>

Аннотация. В статье рассмотрены вопросы выбора ключевых параметров диагностики заболевания печени. Применен метод анализа иерархии для определения важности критериев на основании оценки экспертами прикладной области медицины. Построена иерархическая структура выбора ключевых параметров диагностики заболевания печени. Получены матрицы попарных сравнений для определения важности каждого введенного критерия. Выполнены расчеты противоречивости оценок экспертов с помощью индекса согласованности, которые предложили эксперты при заполнении матрицы попарных сравнений. Проведен сравнительный анализ параметров по полученным значениям глобальных приоритетов со значениями корреляции Пирсона. Выбраны параметры с наиболее эффективным соотношением критериев, что в дальнейшем позволит сократить область исследования взаимосвязи выбранных параметров, а также классифицировать стадии заболевания печени.

Ключевые слова: матрица парных сравнений, метод анализа иерархий, случайная согласованность, отношение согласованности, индекс согласованности

Для цитирования: Серобабов А. С. Выбор ключевых параметров для диагностики заболевания печени на основе метода анализа иерархий // Вестник кибернетики. 2022. № 3 (47). С. 57–65. DOI 10.34822/1999-7604-2022-3-57-65.

Original article

SELECTING KEY PARAMETERS FOR THE DIAGNOSIS OF LIVER DISEASE BASED ON THE ANALYTIC HIERARCHY PROCESS

Aleksandr S. Serobabov

Omsk State Technical University, Omsk, Russia

aserobabow95@mail.ru, <https://orcid.org/0000-0001-8850-0947>

Abstract. The article considers the issues of selecting key parameters for the diagnosis of liver disease. The analytic hierarchy process is used to determine the importance of criteria based on the assessment of the applied medicine provided by the experts. The selection of key parameters outline for the diagnosis of liver disease is built. Matrices of pair-wise comparisons are obtained to determine the significance of each introduced criterion. When filling in the matrix of pair-wise comparisons, the experts proposed their assessments, and their contradiction have been calculated with the consistency index. The comparative analysis of parameters according to the obtained values of global priorities with Pearson correlation values is carried out. In the course of the study, parameters with the most effective correlation of criteria are selected. Those will help reduce the study area of interrelation of selected parameters, as well as liver disease stages.

Keywords: matrix of pair-wise comparisons, analytic hierarchy process, random consistency, relation of consistency, consistency index

For citation: Serobabov A. S. Selecting Key Parameters for the Diagnosis of Liver Disease Based on the Analytic Hierarchy Process // Proceedings in Cybernetics. 2022. No. 3 (47). P. 57–65. DOI 10.34822/1999-7604-2022-3-57-65.

ВВЕДЕНИЕ

В диагностике заболевания важной задачей является выбор значимых параметров. Каждая стадия заболевания печени сопоставляется с определенными диапазонами значений параметров. Процесс выбора соответствующего параметра усложняется наличием многокритериальных предпочтений, поэтому целесообразно применить метод анализа иерархии (далее – МАИ), чтобы декомпозировать проблему на более простые составляющие. Для специалиста прикладной области МАИ принятие решения в результате сопоставления матрицы попарных сравнений критериев и параметров является понятным и простым способом.

В рамках данной статьи рассматривается задача выявления ключевых параметров для постановки диагноза неалкогольной жировой болезни печени (далее – НАЖБП). Данное исследование является продолжением [1], где проведен первичный анализ данных и построена матрица корреляций. Информация для исследования предоставлена экспертами в области гастроэнтерологии и гепатологии. При выявлении ключевых параметров эксперты должны реализовать аналитическую процедуру, состоящую из следующих этапов:

1. Выделить в рамках задачи параметры, которые могут быть связаны с диагностированием заболевания.
2. Определить множество критериев оценки.
3. Оценить важность критериев между собой.
4. Оценить каждый параметр между собой по выбранному критерию.

5. На основе полученных результатов определить ключевые параметры для диагностики заболевания.

МАТЕРИАЛЫ И МЕТОДЫ

В качестве исходных данных использованы результаты, полученные при обследовании 149 пациентов с выявленной НАЖБП. Результаты подвергнуты первичной обработке и вычислению корреляционных связей между стадией заболевания и лабораторными результатами [2].

Для выбора ключевых входных параметров экспертной системы диагностики заболевания представлен метод анализа иерархий. Осуществлен выбор нескольких ключевых параметров (кортежа), где глобальные приоритеты полученных параметров удовлетворяют следующему условию: $X_1 > X_2 > X_3$, где X – множество глобальных приоритетов параметров, X_i – значение глобального приоритета параметра, i – ранг по убывающей множества X .

Постановка задачи. В рамках данной работы рассматривается задача выбора ключевых параметров диагностики заболевания печени. Введем обозначения: $K = \{K_1, K_2 \dots K_4\}$ – множество критериев оценки [2]; K_1 – точность полученных значений; K_2 – уровень достоверности доказательности связи параметра с заболеванием; K_3 – информативность параметра; K_4 – статистическая взаимосвязь. В табл. 1 представлены обозначения параметров и их полные наименования.

Таблица 1

Наименование параметров и их обозначения

Обозначение параметра	Наименование параметра	Обозначение параметра	Наименование параметра
L_{lep}	Лептин	D_{oc}	Наличие остеоартроза у пациента
L_{ObR}	Рецепторы лептина	L_{ggt}	Гамма-глутамилтранспептидаза
D_{nash}	Неалкогольный стеатогепатит	D_{bit}	Наличие дискомфорта у пациента
D_P	Наличие рубиновых пятен у пациента	P_{din}	Динамика массы
D_o	Наличие ожирения у пациента	P_{ob}	Обхват талии
L_{timp2}	Тканевой ингибитор матриксных протеиназ 2	L_{ttg}	Тиреотропный гормон натошак

Примечание: составлено по материалам статьи [1].

Метод анализа иерархий. Для применения метода анализа иерархий необходимо

построить иерархическую структуру, представленную на рис. 1. Иерархия строится,

начиная с вершины, где на первом уровне указывается глобальная цель, выраженная нахождением нескольких ключевых параметров, таких что $X_1 > X_2 > X_3$, где X – множество значений глобального авторитета, i –

ранг множества X глобального авторитета, упорядоченного в порядке убывания. На втором уровне – критерии, на третьем – параметры диагностирования заболевания.

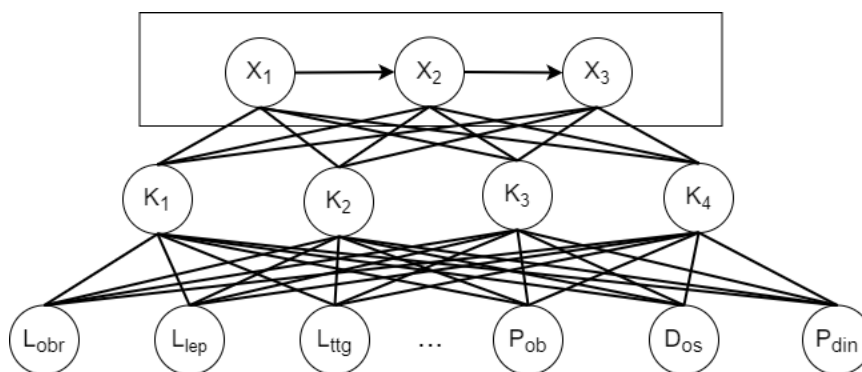


Рис. 1. Иерархическая структура выбора ключевых параметров диагностики заболевания печени

Примечание: составлено авторами на основании задачи о нахождении ключевых параметров методом анализа иерархий, где $X_1 > X_2 > X_3$.

Важность критериев множества K определяется в результате их попарного сравнения [3], при котором выполняется построение матрицы A размером $(h \times h)$ парных сравнений критериев в результате соотношения: $A = (a_{ij}) = k_i/k_j$, где k_i – числовая значимость критерия, полученная в результате экспертной оценки, представленная числовым значением от 1 до 9:

- 1 – аналогичная значимость;
- 3 – незначительная значимость;
- 5 – значительная значимость;
- 7 – существенная значимость;
- 9 – основная значимость.

При неопределенности выбора выстав-ляются четные значения: 2, 4, 6, 8 (в частности,

4 – между незначительным и значительным преимуществом) [4, 5].

По построенной матрице A рассчитывается значение важности i -го критерия:

$$a_i = v_i / \sum_{j=1}^k v_j, \quad (1)$$

где $v_i = (\prod_{j=1}^h a_{ij})^{1/h}$ – значение i -й компоненты собственного вектора матрицы A .

Первым шагом задачи выбора ключевых параметров является построение матрицы парных сравнений важности критериев, представленной в табл. 2.

Таблица 2

Матрица попарных сравнений для определения важности критериев

Матрица критериев	K_1	K_2	K_3	K_4	v_i	a_i
K_1	1	0,25	0,33	0,33	0,41	0,09
K_2	4	1	2	2	2	0,43
K_3	3	0,5	1	1	1,11	0,24
K_4	3	0,5	1	1	1,11	0,24
Сумма по столбцу (S)	11	2,25	4,33	4,33	4,62	1
$S \times a_i$	0,97	0,97	1,04	1,04	$\lambda_{\max}$	12,4
Индекс согласованности (ω)					0,0053	

Примечание: составлено авторами на основе сравнений критериев, полученных в результате опроса врачей.

Полученные значения α_i показывают важность критериев при решении задачи выбора ключевых параметров диагностики заболевания печени. Из расчетов видно, что явное предпочтение отводится критерию K_2 .

Далее строится матрица парных сравнений по каждому критерию для каждого па-

раметра. Эксперт каждые имеющиеся параметры сравнивает попарно и оценивает их с помощью девятибалльной шкалы Саати [3].

В табл. 3 представлена матрица попарных сравнений по критерию «Точность полученных результатов».

Таблица 3

Матрица попарных сравнений по критерию «Точность полученных результатов»

K_i	L_{obr}	L_{lep}	L_{ttg}	L_{ggt}	L_{tmp2}	D_{OC}	D_{nash}	D_{bit}	D_o	P_{ob}	D_{os}	P_{din}	α_i
L_{obr}	1	0,5	0,5	1	2	2	1	3	2	1	2	2	0,1
L_{lep}	2	1	1	2	2	3	1	3	2	1	2	2	0,13
L_{ttg}	2	1	1	2	2	3	1	3	2	1	2	2	0,13
L_{ggt}	1	0,5	0,5	1	1	2	0,5	2	1	0,5	1	1	0,07
L_{tmp2}	0,5	0,5	0,5	1	1	2	0,5	2	1	0,5	1	1	0,07
D_{OC}	0,5	0,3	0,3	0,5	2	1	0,3	0,5	1	0,3	1	1	0,05
D_{nash}	1	1	1	2	0,5	3	1	3	3	1	3	2	0,12
D_{bit}	0,3	0,3	0,3	0,5	0,5	2	0,3	1	1	0,3	1	0,5	0,04
D_o	0,5	0,5	0,5	1	1	1	0,3	1	1	0,3	1	1	0,05
P_{ob}	1	1	1	2	2	3	1	3	3	1	1	1	0,11
D_{os}	0,5	0,5	0,5	1	1	1	0,3	1	1	1	1	1	0,06
P_{din}	0,5	0,5	0,5	1	1	1	0,5	2	1	1	1	1	0,07
Сумма (S)	10,8	7,7	7,7	15	16	24	7,8	24	19	9	17	15	1
$S \times \alpha_i$	1,09	1	1	1,04	1,04	1,14	0,93	1,07	1,02	1,02	1,01	1,01	
λ_{max}				12,4			I_s				0,0361		
Случайная согласованность (I_r)				1,49			Отношение согласованности (I_o)				0,0242		

Примечание: составлено авторами на основе сравнений параметров по критерию «Точность полученных результатов», полученных в результате опроса врачей.

Для выявления противоречивости оценок, которые предложили эксперты при заполнении матрицы попарных сравнений, используется количественная оценка – индекс согласованности [6], вычисляемый по формуле:

$$I_s = \frac{\lambda_{max} - n}{n - 1}, \quad (2)$$

где λ_{max} – собственное значение матрицы;
 n – количество параметров.

Индекс согласованности показывает степень непротиворечивости суждений эксперта. Идеальный случай I_s равен нулю.

Для сопоставления матриц различной размерности по показателю согласованности

используется нормированный показатель «отношение согласованности», вычисляемый по формуле:

$$I_o = \frac{I_s}{I_r}, \quad (3)$$

где I_r – случайная согласованность, т. е. среднестатистическое значение индекса согласованности при случайном выборе коэффициентов матрицы сравнения.

Значения случайной согласованности для матриц различного порядка приведены в табл. 4 [7].

Таблица 4

Значения случайной согласованности для матриц различного порядка

Размер матрицы	1	2	3	4	5	6	7	8	9	10
Случайная согласованность	0	0	0,58	0,9	1,12	1,24	1,32	1,41	1,45	1,49

Примечание: составлено по материалам статьи [7].

Вывод о противоречивости результатов по критерию «Точность полученных результатов»: величина $I_o < 0,1$ [8], следовательно, противоречивость в суждениях отсутствует.

В табл. 5 представлена матрица попарных сравнений по критерию «Уровень достоверности доказательности связи параметра с заболеванием».

Таблица 5

**Матрица попарных сравнений по критерию
«Уровень достоверности доказательности связи параметра с заболеванием»**

K_2	L_{obr}	L_{lep}	L_{ttq}	L_{ggt}	L_{tmp2}	D_{OC}	D_{nash}	D_{bit}	D_o	P_{ob}	D_{os}	P_{din}	α_i
L_{obr}	1	1	1	2	1	3	1	3	3	2	2	2	0,13
L_{lep}	1	1	1	2	1	3	1	3	3	2	2	2	0,13
L_{ttq}	1	1	1	2	0,5	2	0,5	2	2	1	1	1	0,13
L_{ggt}	1	0,5	0,5	1	1	2	0,5	2	1	0,5	1	1	0,07
L_{tmp2}	1	1	1	2	1	0,5	0,5	1	1	1	1	1	0,07
D_{OC}	0,3	0,3	0,3	0,5	2	1	0,3	1	1	1	1	1	0,05
D_{nash}	1	1	1	2	2	3	1	3	3	2	2	2	0,14
D_{bit}	0,3	0,3	0,3	0,5	1	1	0,3	1	1	1	1	1	0,05
D_o	0,3	0,3	0,3	0,5	1	1	0,3	1	1	1	1	1	0,05
P_{ob}	0,5	0,5	0,5	1	1	1	0,5	1	1	1	1	1	0,06
D_{os}	0,5	0,5	0,5	1	1	1	0,3	1	1	1	1	1	0,06
P_{din}	0,5	0,5	0,5	1	1	1	0,5	2	1	1	1	1	0,06
Сумма (S)	8	8	8	15,5	13,5	20,5	7,5	21	21	16	16	16	1
$S \times \alpha_i$	1,02	1,02	1,02	1,06	0,98	1,09	1,01	1,06	1,06	0,98	0,98	0,98	
λ_{max}	12,3					I_s					0,0276		
Случайная согласованность (R)	1,49					I_o					0,0185		

Примечание: составлено авторами на основе сравнений параметров по критерию «Уровень достоверности доказательности связи параметра с заболеванием», полученных в результате опроса врачей.

Вывод о противоречивости результатов по критерию «Уровень достоверности доказательности связи параметра с заболеванием»: величина $I_o < 0,1$, следовательно, противоречивость в суждениях отсутствует.

В табл. 6 представлена матрица попарных сравнений по критерию «Информативность параметра».

Таблица 6

Матрица попарных сравнений по критерию «Информативность параметра»

K_3	L_{obr}	L_{lep}	L_{ttq}	L_{ggt}	L_{tmp2}	D_{OC}	D_{nash}	D_{bit}	D_o	P_{ob}	D_{os}	P_{din}	α_i
L_{obr}	1	1	2	2	2	3	1	2	3	2	2	3	0,14
L_{lep}	1	1	2	2	2	3	1	2	3	2	2	3	0,14
L_{ttq}	0,5	0,5	1	1	1	2	0,5	1	2	2	1	2	0,08
L_{ggt}	0,5	0,5	1	1	1	2	0,5	1	2	1	1	2	0,07
L_{tmp2}	0,5	0,5	1	1	1	2	0,3	1	1	1	1	0,5	0,06
D_{OC}	0,3	0,3	0,5	1	3	1	0,5	1	0,5	1	1	0,5	0,06
D_{nash}	1	1	2	2	1	2	1	2	3	2	1	3	0,12
D_{bit}	0,5	0,5	1	1	1	1	0,5	1	2	1	1	2	0,07
D_o	0,3	0,3	0,5	0,5	1	2	0,3	0,5	1	0,5	0,3	1	0,05
P_{ob}	0,5	0,5	0,5	1	1	1	0,5	1	2	1	1	0,5	0,06
D_{os}	0,5	0,5	1	1	1	1	1	1	1	3	1	0,5	0,07
P_{din}	0,3	0,3	0,5	0,5	2	2	0,3	0,5	1	2	2	1	0,06
Сумма (S)	7	7	13	14	17	21	7,5	14	23,5	16,5	14,3	19	1
$S \times \alpha_i$	1,01	1,01	1,06	1,02	0,95	1,53	0,93	1,02	1,06	1,01	1,02	1,18	
λ_{max}	12,8					I_s					0,0757		
Случайная согласованность (R)	1,49					I_o					0,0508		

Примечание: составлено авторами на основе сравнений параметров по критерию «Информативность параметра», полученных в результате опроса врачей.

Вывод о противоречивости результатов «Информативность параметра»: величина $I_o < 0,1$, следовательно, противоречивость в суждениях отсутствует.

В табл. 7 представлена матрица попарных сравнений по критерию «Статистическая взаимосвязь».

Таблица 7

Матрица попарных сравнений по критерию «Статистическая взаимосвязь»

K_4	L_{obr}	L_{lep}	L_{ttg}	L_{ggt}	L_{imp2}	D_{oc}	D_{nash}	D_{bit}	D_o	P_{ob}	D_{os}	P_{din}	α_i
L_{obr}	1	1	1	1	1	2	1	2	1	1	2	1	0,1
L_{lep}	1	1	1	1	1	2	1	2	1	1	2	2	0,1
L_{ttg}	1	1	1	1	1	2	1	2	1	1	2	1	0,1
L_{ggt}	1	1	1	1	1	3	1	2	1	1	2	1	0,1
L_{imp2}	1	1	1	1	1	2	1	2	2	1	2	1	0,1
D_{oc}	0,5	0,5	0,5	0,3	0,5	1	0,3	0,5	0,5	1	0,5	0,3	0,04
D_{nash}	1	1	1	1	1	3	1	2	1	2	1	1	0,1
D_{bit}	0,5	0,5	0,5	0,5	0,5	2	0,5	1	0,5	0,5	1	1	0,05
D_o	1	1	1	1	0,5	2	1	2	1	1	2	1	0,09
P_{ob}	1	1	1	1	1	1	0,5	2	1	1	0,5	1	0,08
D_{os}	0,5	0,5	0,5	0,5	0,5	2	1	1	0,5	2	1	1	0,06
P_{din}	1	0,5	1	1	1	3	1	1	1	1	1	1	0,08
Сумма (S)	10,5	10	10,5	10,3	10	25	10,3	19,5	11,5	13,5	17	12,3	1
$S \times \alpha_i$	1	1,01	1	1,02	1,01	1,02	1,02	1,05	1,04	1,03	1,09	1,03	
λ_{max}	12,3					I_s					0,0297		
R	1,49					I_o					0,0199		

Примечание: составлено авторами на основе сравнений параметров по критерию «Статистическая взаимосвязь», полученных в результате опроса врачей.

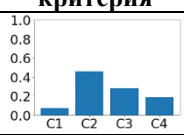
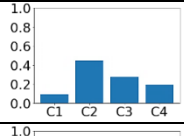
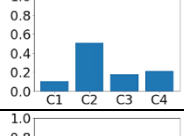
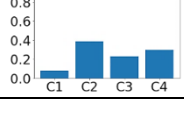
Вывод о противоречивости результатов по критерию «Статистическая взаимосвязь»: величина $I_o < 0,1$, следовательно, противоречивость в суждениях отсутствует.

На последнем этапе принятия решения подсчитываются значения глобальных прио-

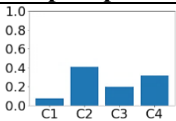
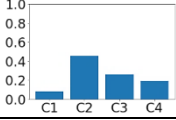
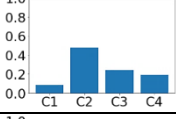
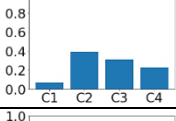
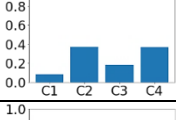
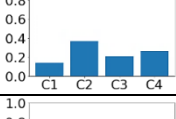
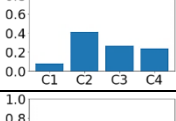
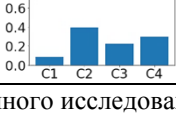
ритетов как сумма произведений значения вектора приоритета для критерия и значения вектора локального приоритета этой альтернативы в отношении данного критерия, представленные в табл. 8.

Таблица 8

Значения глобальных приоритетов

Вектор глобальных приоритетов	C_1	C_2	C_3	C_4	Относительный вклад критерия	Глобальные приоритеты
L_{obr}	0,0089	0,0554	0,0345	0,0229		0,121
L_{lep}	0,0116	0,0554	0,0345	0,0243		0,125
L_{ttg}	0,0116	0,0554	0,0197	0,0229		0,109
L_{ggt}	0,0061	0,0298	0,0175	0,0229		0,077

Окончание табл. 8

Вектор глобальных приоритетов	C_1	C_2	C_3	C_4	Относительный вклад критерия	Глобальные приоритеты
L_{tmp2}	0,0057	0,0316	0,0151	0,0243		0,077
D_{oc}	0,0042	0,0232	0,0134	0,0098		0,051
D_{nash}	0,0104	0,0587	0,0297	0,0237		0,123
D_{bit}	0,0038	0,0219	0,0175	0,0129		0,056
D_o	0,0048	0,0219	0,0108	0,0216		0,059
P_{ob}	0,01	0,0265	0,0147	0,0182		0,07
D_P	0,0052	0,0265	0,0171	0,0153		0,064
P_{din}	0,0057	0,0265	0,0149	0,0199		0,067

Примечание: составлено авторами на основе таблиц 2, 3, 5, 6, 7 данного исследования.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В результате исследования получены значения глобальных приоритетов (табл. 7), которые изображены на рис. 2а. В качестве критерия достоверности полученных результатов использованы результаты исследования [9], в котором получены значения корреляции параметров со стадией болезни, полученные результаты представлены на рис. 2б. Как видно из двух диаграмм, имеются некоторые сходства в поведении тренда. Так, параметры L_{obr} , L_{lep} , D_{nash} имеют наибольшие значения глобального приоритета и значения корреляции выше среднего. С учетом того, что МАИ включает в себя оценку параметров по нескольким критериям, то предпочтительней использовать его результаты для выбора ключевых параметров диагностики

заболевания печени, а значения корреляции выступают еще одним методом подтверждения правильности выбранных параметров. Представленные значения имеют слабую и среднюю силу корреляции, что не является достаточным критерием выбора ключевых параметров. Также корреляция отражает линейную взаимосвязь, но связь между параметром и стадией болезни может иметь не только линейную, но и более сложную связь, поэтому проведена оценка параметров экспертами по четырем критериям (точность полученных значений, уровень достоверности доказательности связи параметра с заболеванием, информативность параметра, статистическая взаимосвязь).

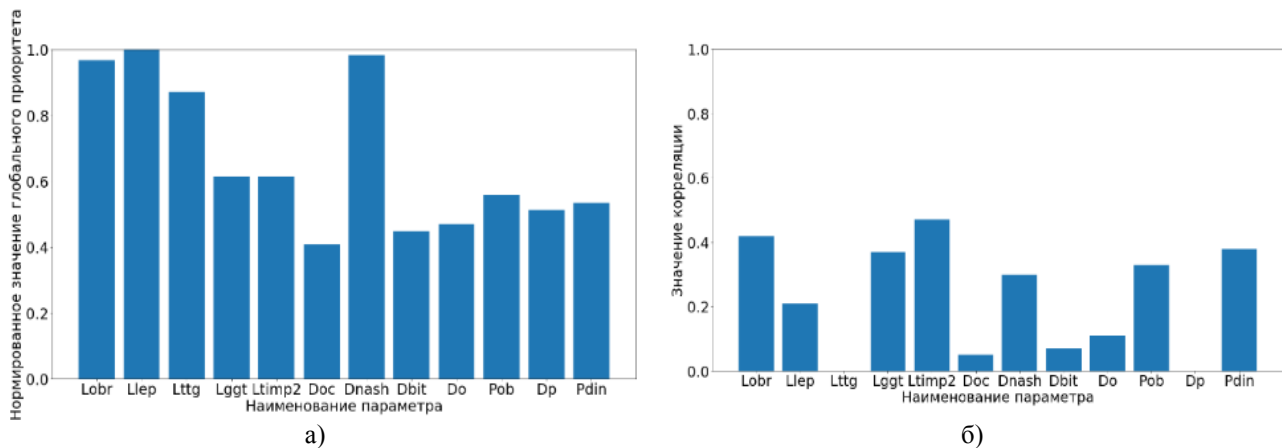


Рис. 2. Гистограммы сравнения глобального приоритета и полученного значения корреляции:
 а) гистограмма нормированных значений глобального приоритета; б) гистограмма значений корреляций
Примечание: составлено авторами на основе данных, полученных в исследовании и в статье [1].

ЗАКЛЮЧЕНИЕ

В результате опроса экспертов сформированы матрицы парных сравнений критериев: точность полученных значений, уровень достоверности доказательности связи параметра с заболеванием, информативность параметра, статистическая взаимосвязь. После обработки суждений экспертов вычислены оценки соответствующих собственных векторов и векторов приоритетов для каждого уровня иерархии. Согласно методу анализа иерархий по максимальному значению ком-

поненты глобального вектора, получены ключевые параметры для диагностики НАЖБП.

Кроме того, представляется целесообразным использовать полученные параметры для создания автоматизированной экспертной системы поддержки принятия решения [9–10], работающей с большими объемами числовых показателей, для автоматической классификации стадии заболевания НАЖБП.

Список источников

1. Серобабов А. С., Чебаненко Е. В., Денисова Л. А., Кролевец Т. С. Разработка экспертной системы ранней диагностики заболеваний: программные средства первичной обработки и выявление зависимостей // Омск. науч. вестн. 2018. № 4 (160). С. 179–184.
2. Фурцев Д. Г., Чиклаева А. А. Алгоритм выбора лучшего решения в системах поддержки принятия решений // Прикладная математика, управление и информатика : сб. тр. Междунар. молодежная конф., 3–5 октября 2012 г. Белгород, 2012. С. 607–609.
3. Кузьмина Н. М., Ридли А. Н. Методика оценки степени значимости категорий условий применения тарифа // Науч. вестн. МГТУ ГА. 2016. № 224 (2). С. 138–146.
4. Чирухин В. А., Прохоров В. М. О практике применения метода анализа иерархий в логистике // Logistics. 2018. № 6. С. 44–48.
5. Волокобинский М. Ю., Пекарская О. А., Рязи Д. А. Принятие решений на основе метода анализа иерархий // Финансы: теория и практика. 2016. № 2 (92). С. 33–42.

References

1. Serobabov A. S., Chebanenko E. V., Denisova L. A., Krolevets T. S. Early Disease Detection Expert System Development: Primary Machining Software Applications and Dependences // Omsk Scientific Bulletin. 2018. No. 4 (160). P. 179–184. (In Russian).
2. Furtsev D. G., Chikulaeva A. A. Algoritm vybora luchshego resheniia v sistemakh podderzhki priniatiia reshenii // Prikladnaia matematika, upravlenie i informatika : Proceedings of the International Youth Conference, October 3–5, 2012. Belgorod, 2012. P. 607–609. (In Russian).
3. Kuzmina N. M., Ridley A. N. A Method for Estimating the Impact of the Fare Rules Conditions // Civil Aviation High Technologies. 2016. No. 224 (2). P. 138–146. (In Russian).
4. Chirukhin V. A., Prokhorov V. M. On the Practice of Application of Analytic Hierarchy Process in Logistics // Logistics. 2018. No. 6. P. 44–48. (In Russian).
5. Volokobinsky M. Yu., Pekarskaya O. A., Razi D. A. Decision-Making Based on the Hierarchy Analysis Method // Finance: Theory and Practice. 2016. No. 2 (92). P. 33–42. (In Russian).

6. Фурцев Д. Г., Коваленко А. Н., Ткаченко Е. А. Об оптимизации на основе метода анализа иерархий // Науч. вед. 2014. № 1 (172). С. 142–150.
7. Картвелишвили В. М., Лебедюк Э. А. Метод анализа иерархий: критерии и практика // Вестн. РЭА им. Г. В. Плеханова. 2013. № 6 (60). С. 97–112.
8. Socaciu L. G., Unguresan P. V. Using the Analytic Hierarchy Process to Prioritize and Select Phase Change Materials for Comfort Application in Buildings // Mathematical Modelling in Civil Engineering. 2014. Vol. 10, Is. 1. P. 21–28.
9. Ткаченко А. Л., Мещеряков В. А., Денисова Л. А. Проектирование информационно-аналитической системы для поддержки образовательного процесса технического вуза // Автоматизация в промышленности. 2022. № 4. С. 7–14.
10. Серобабова А. С. Анализ входных параметров экспертной системы ранней диагностики заболевания // Вестник кибернетики. 2020. № 4 (40). С. 33–41.
6. Furtsev D. G., Kovalenko A. N., Tkachenko E. A. On the Optimization Method Based on the Analytic Hierarchy Process // Nauch. ved. 2014. No. 1 (172). P. 142–150. (In Russian).
7. Kartvelishvili V. M., Lebedyuk E. A. Metod analiza ierarkhii: kriterii i praktika // Vestnik of the Plekhanov Russian University of Economics. 2013. No. 6 (60). P. 97–112. (In Russian).
8. Socaciu L. G., Unguresan P. V. Using the Analytic Hierarchy Process to Prioritize and Select Phase Change Materials for Comfort Application in Buildings // Mathematical Modelling in Civil Engineering. 2014. Vol. 10, Is. 1. P. 21–28.
9. Tkachenko A. L., Meshcheryakov V. A., Denisova L. A. Design of an Analytic Support System for the Educational Process in a Technical University // Avtomatizatsiia v promyshlennosti. 2022. No. 4. P. 7–14. (In Russian).
10. Serobabov A. S. Analysis of Input Parameters of the Expert System for Early Diagnosis of the Disease // Proceedings in Cybernetics. 2020. No. 4 (40). P. 33–41. (In Russian).

Информация об авторе

А. С. Серобабова – аспирант.

Information about the author

A. S. Serobabov – Postgraduate.

Научная статья
УДК 519.816:616.14
doi: 10.34822/1999-7604-2022-3-66-74

СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ В МЕДИЦИНЕ: ПОДХОДЫ И МЕТОДЫ ПРИМЕНЕНИЯ ВО ФЛЕБОЛОГИИ

Нелли Руслановна Урманцева^{1✉}, Алсу Винировна Курамшина²

^{1,2} Сургутский государственный университет, Сургут, Россия

¹ nel-u@yandex.ru ✉, <https://orcid.org/0000-0001-9163-6132>

² kuramshina_av@surgu.ru, <https://orcid.org/0000-0001-9891-4959>

Аннотация. Способы принятия решений, которые опираются на теорию вероятностей, специальные главы математики и другие инструменты, определяют необходимость актуализации подходов и методов приобретения новых компетенций, а иногда и возврата к принятию решений на основе интуиции. Осуществление выбора для человека (лица, принимающего решение) становится все более сложной задачей, особенно это касается медицины, где задачи слабо поддаются формализации, а степень ответственности за принятое решение является высокой. В статье рассматривается проектирование и внедрение системы поддержки принятия решений, разработанной как модуль медицинской информационной системы, для оказания помощи сердечно-сосудистым хирургам в решении задач, связанных с принятием клинических решений в области флебологии.

Ключевые слова: теория принятия решений, система поддержки принятия решений, хронические заболевания вен, флебология

Для цитирования: Урманцева Н. Р., Курамшина А. В. Система поддержки принятия решений в медицине: подходы и методы применения во флебологии // Вестник кибернетики. 2022. № 3 (47). С. 66–74. DOI 10.34822/1999-7604-2022-3-66-74.

Original article

DECISION-MAKING SUPPORT SYSTEM IN MEDICINE: APPROACHES AND APPLICATION METHODS IN PHLEBOLOGY

Nelly R. Urmantseva^{1✉}, Alsu V. Kuramshina²

^{1,2} Surgut State University, Surgut, Russia

¹ nel-u@yandex.ru ✉, <https://orcid.org/0000-0001-9163-6132>

² kuramshina_av@surgu.ru, <https://orcid.org/0000-0001-9891-4959>

Abstract. Decision-making methods based on probability theory, specific chapters in mathematics, and other tools dictate the necessity to update approaches and methods for acquiring new competencies, and even to return to intuitive decision-making. Decision-making for a person (a decision maker) is becoming a more complex objective, particularly in medicine, as there is no certainty in the formalization of tasks, with high responsibility for the decision made. The article considers the design and implementation of a decision-making support system made as a module of a medical information system to assist cardiovascular surgeons in solving problems related to clinical decision-making in phlebology.

Keywords: decision-making theory, decision-making support system, chronic venous disorders, phlebology

For citation: Urmantseva N. R., Kuramshina A. V. Decision-Making Support System in Medicine: Approaches and Application Methods in Phlebology // Proceedings in Cybernetics. 2022. No. 3 (47). P. 66–74. DOI 10.34822/1999-7604-2022-3-66-74.

ВВЕДЕНИЕ

Говоря о процессе принятия решения в современных сложных условиях, целесообразно продумывать многие аспекты в системе поддержки принятия решений (далее – СППР), которая представлена как компьютерная автоматизированная система. Основное предназначение СППР – это помощь лицу, принимающему решение. Такая система может включать в себя комплекс подходов, методов и моделей как математических, так и эвристических, которые объединены общей методикой формирования альтернатив управленческих решений в организационных системах, определения последствий реализации каждой альтернативы и обоснования выбора наиболее подходящего управленческого решения [1]. При применении математических и эвристических подходов, методов и моделей, а также разработка адекватной, комплексной СППР приобретают актуальность в медицине и требуют детального изучения.

Целью исследования является формирование основных модулей СППР во флебологии. Она будет достигаться с помощью решения следующих задач:

- изучение и обобщение возможностей, трудностей, основных подходов и методов процесса принятия решений;
- выделение субъективных факторов, влияющих на процессы принятия решений;
- проектирование и реализация СППР во флебологии, в частности, построение дерева принятия решений предметной области, функциональной модели IDEF0 системы, базы знаний, реализация интерфейса.

В исследовании используется система математических и эвристических методов принятия решений, при этом авторами выделяются возможности, недостатки и влияние субъективных факторов на процесс принятия решений. В результате выявляется, что при использовании практически любого подхода, метода невозможно полностью исключить влияние человеческого фактора, а при осуществлении выбора в медицине это является существенным моментом. Применение СППР при приеме пациента позволит минимизиро-

вать влияние субъективного фактора на процесс принятия решений врача-флеболога.

МАТЕРИАЛЫ И МЕТОДЫ

Процесс принятия и реализации решения зависит от множества факторов. С одной стороны, это процесс, в котором многие моменты субъективны и зависят от качеств, компетентности, опыта участников, в частности, формирование таких способов выявления предпочтений, как шкала и критерии, процедура оценки характеристик предмета/явления лицом, принимающим решение (далее – ЛПР). Особенно это касается плохо структурируемых проблем.

С другой стороны, в науке управления сформировалось множество методов, использование которых обеспечивает объективность и эффективность. Осуществление выбора в условиях определенности является самой простой задачей, однако такая ситуация в современном мире практически недостижима.

При этом выбор в условиях неопределенности зависит от множества переменных:

- неопределенности, связанной с участниками процесса принятия решения;
- неизвестности и неизученности факторов внешней среды;
- невозможности четкого описания задачи;
- невозможности структурировать проблемную ситуацию в форму, удобную для анализа и оптимального выбора.

Существующие подходы и методы принятия решений предоставляют следующие возможности: теоретическое обоснование рационального выбора [2], осуществляемого человеком; учет разнородных характеристик сравниваемых результатов; сочетание последовательно выполняемых итераций, реализуемых человеком (фаза анализа) и компьютером (фаза расчетов), позволяет сужать область допустимых решений; упрощение решения при помощи разбиения исходной задачи на ряд менее сложных задач оптимизации; разработка и использование алгоритмов для решения транспортной задачи и др.

Однако у всех подходов есть свои недостатки (рис. 1).

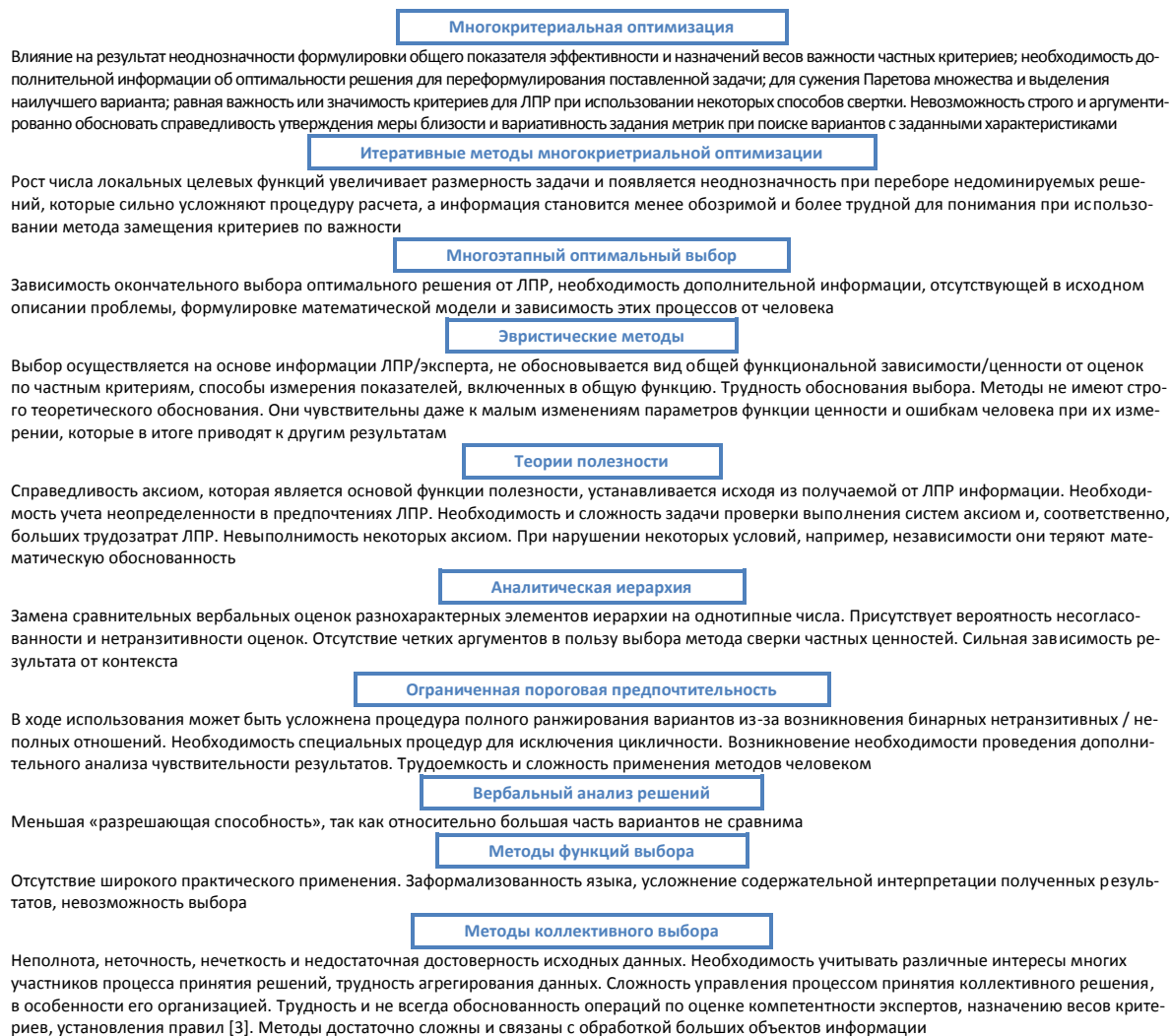


Рис. 1. Недостатки некоторых подходов и методов в процессе принятия решений

Примечание: составлено авторами с использованием материалов из [3].

Таким образом, можно констатировать, что накопленный опыт не позволяет исключить субъективные факторы в процессе принятия решений, а влияние человеческого фактора настолько многообразно и распространяется также и на выбор конкретного подхода и метода в конкретной ситуации. Представленный анализ возможностей и трудностей применения подходов, методов и моделей в большей степени связан с субъективными моментами, а именно – с предпочтениями ЛПР.

Можно говорить о том, что рациональность поведения человека, который не только определяет отдельные категории и активно участвует в процессе принятия решений, но также организует этот процесс, управляет им, в том числе осуществляет выбор подхо-

дов и методов, используемых в процессе принятия решения, зависит от множества факторов. При этом особенно важно, что выбор связан не только с неопределенностью, но и несравнимостью вариантов, а ЛПР (эксперт, консультант) затрудняется его осуществить, либо их предпочтительность является слабой. Все это находит отражение в различных предметных областях.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Особую актуальность применение описанных выше методов для создания адекватной СППР находит в медицине, т. к. принятое решение – это чья-то жизнь или ее качество. Еще одной важной причиной для создания СППР является также и комфорт работы врача в случаях, когда заболевание является рас-

пространственным, опасным, а поток пациентов очень большим. На приеме сердечно-сосудистого хирурга-флеболога возникает именно такая ситуация. На данный момент в Российской Федерации варикозом страдают 20 % мужчин и 40 % женщин, из которых у 15 % уже есть серьезные осложнения. Также каждый год число людей, страдающих варикозом, увеличивается на 2,5 %. Этот рост происходит за счет тяжелых форм заболевания. А самая высокая распространенность встречается у людей в возрасте 25–50 лет, т. е. у работоспособной части населения.

СППР, создаваемая для решения всех обозначенных проблем, направлена на определение наличия варикоза у пациента при помощи сбора анамнеза и визуального осмотра.

После получения информации система будет отображать наличие болезни и ее стадию по международной классификации СЕАР. При необходимости она дает возможность хранения результатов опроса пациента.

При проектировании СППР во флебологии следует представить предметную область с использованием дерева принятия решений – метода представления решающих правил в виде иерархии, включающей в себя элементы двух типов – узлов и листьев. Узлы включают в себя решающие правила и производят проверку примеров на соответствие выбранного атрибута обучающего множества [4].

Дерево принятия решений (далее – ДПР) предметной области представлено на рис. 2.

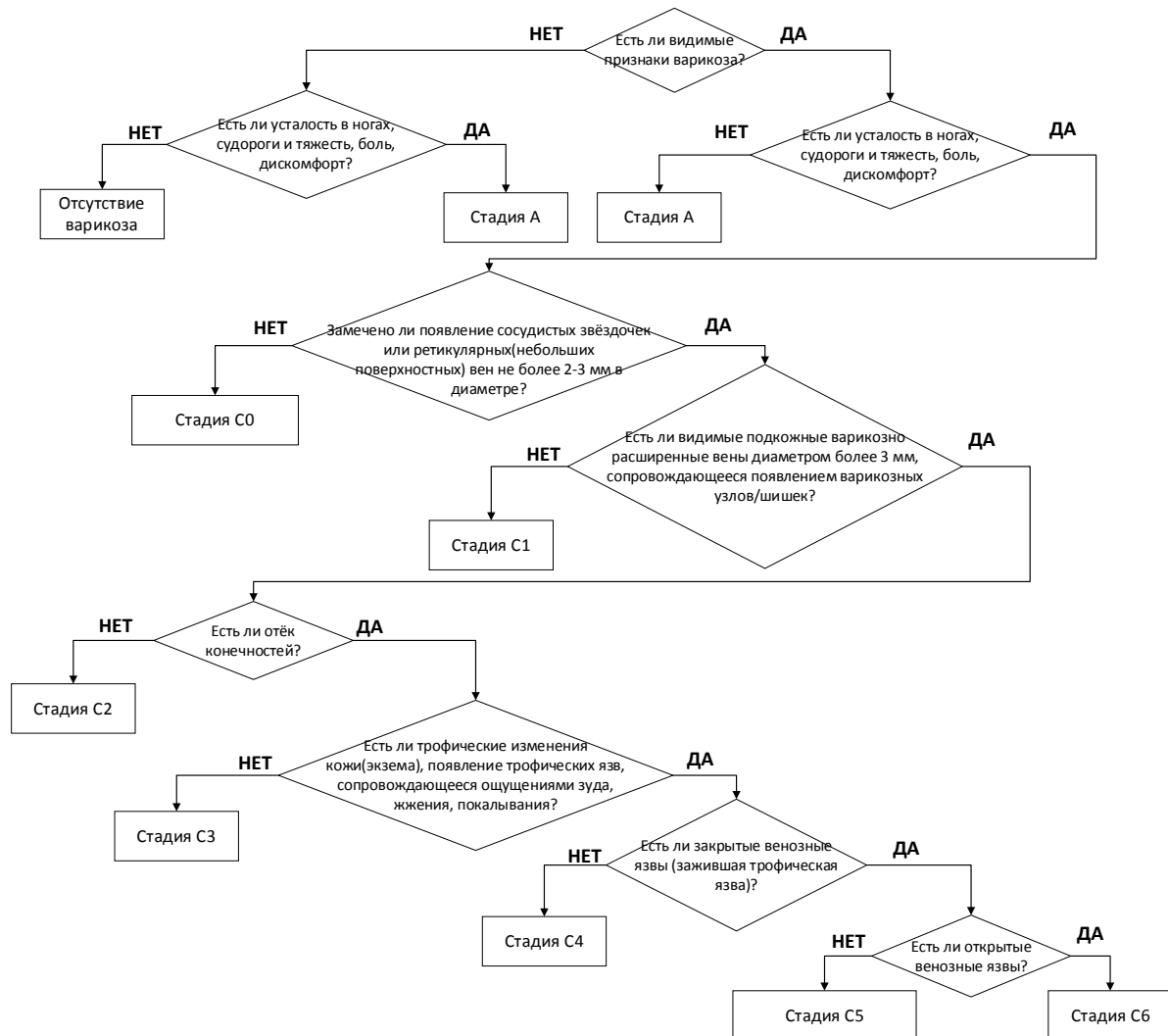


Рис. 2. Дерево принятия решений предметной области

Примечание: разработано авторами.

ДПР предметной области имеет глубину 8, описывает все возможные стадии варикоза, в том числе такие диагнозы, как «отсутствие варикоза» и стадия А – «бессимптомное течение болезни», и является модификацией ДПР из [5].

На рис. 3 представлена декомпозиция контекстной диаграммы IDEF0 для СППР сердечно-сосудистого хирурга-флеболога.

На диаграмме изображены 5 блоков и следующие связи между ними (рис. 4).

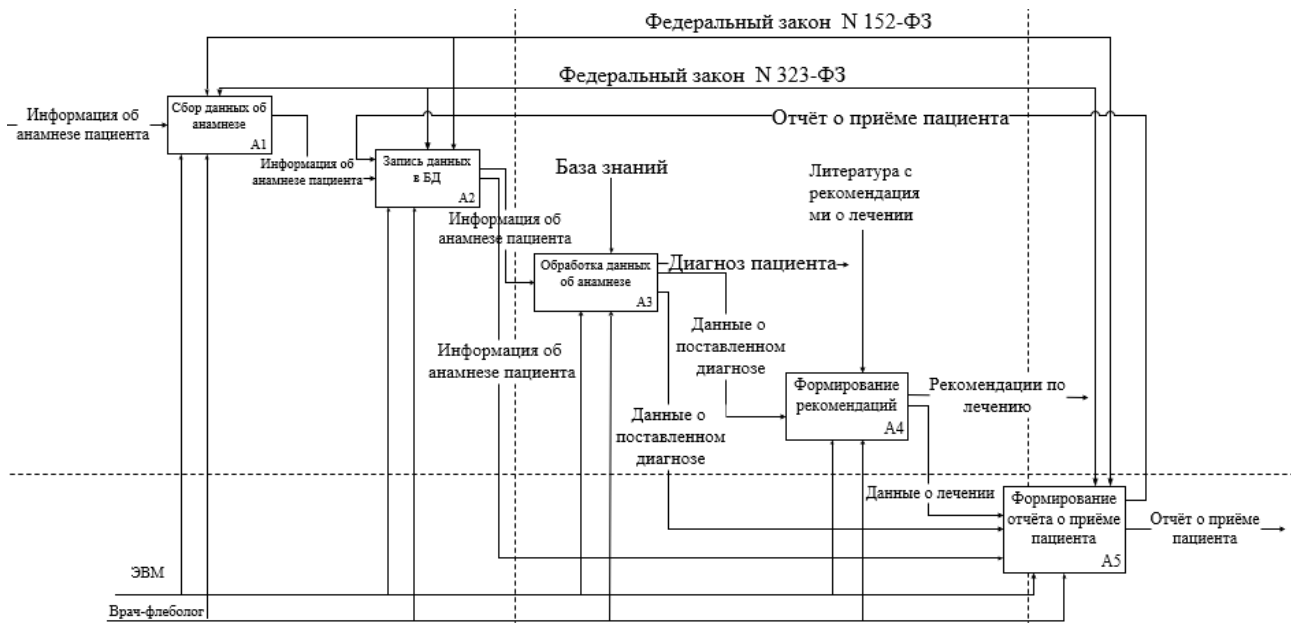


Рис. 3. Диаграмма декомпозиции функциональной модели СППР врача-флеболога
 Примечание: разработано авторами.

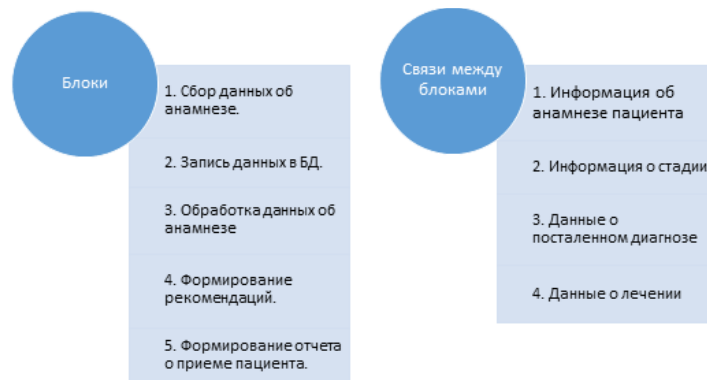


Рис. 4. Названия функциональных блоков и связей между ними контекстной диаграммы IDEF0 СППР врача-флеболога

Примечание: разработано авторами.

База знаний, формируемая для разработки системы, – текстовый файл. Он затем может быть зашифрован. База знаний может быть представлена структурой (рис. 5) и состоять из трех секций.

Перечисление правил вывода (последняя секция) заканчивается концом файла. Отдельная строка – отдельное правило вывода. Правило вывода начинается с описания исхода. При этом вероятность исхода зависит

от изменения этого правила. Оно чаще всего формулируется в форме текста, который может включать любые символы (кроме запятых). Априорная вероятность каждого исхода (Р) указывается после запятой, являющейся разделителем. Эта вероятность представлена в том случае, если отсутствует дополнительная информация. После указывается поле, состоящее из трех элементов.

Оно является итерационным и формируется через запятую-разделитель.

Первый элемент (свидетельство): номер соответствующего симптома – i .

Второй элемент: вероятность получения ответа «Да» для первого элемента (свидетельства i), в случае, если возможный исход верен – $P_{y_i} = P(E_i | H)$.

Третий элемент: вероятность получения ответа «Да» для каждого симптома, относящегося к данному исходу $[6] - P_n = P(E_i | -H)$.

Элемент одного исхода для секции 3 (рис. 5), автоматически сформированный редактором базы знаний после ее наполнения, представлен ниже:

Отсутствие варикоза, 0.75, 1,0.8,0.2, 2,0.95,0.005, 3,0.6,0.2, 4,0.2,0.01, 5,0.0001,0.01, 6,0.0001,0.01, 7,0.0001,0.01, 8,0.0001,0.01, 9,0.0001,0.01, 10,0.0001,0.01, 11,0.0001,0.01, 12,0.0001,0.01.

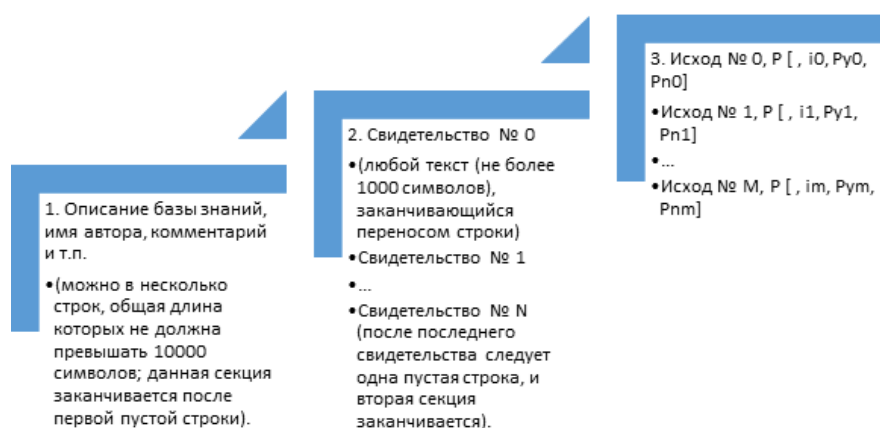


Рис. 5. Структура файла базы знаний СППР врача-флеболога

Примечание: разработано авторами.

Созданная в выбранном для разработки СППР приложении «Малая экспертная систе-

ма 2.0» база знаний выглядит для пользователя так, как представлено на рис. 6.

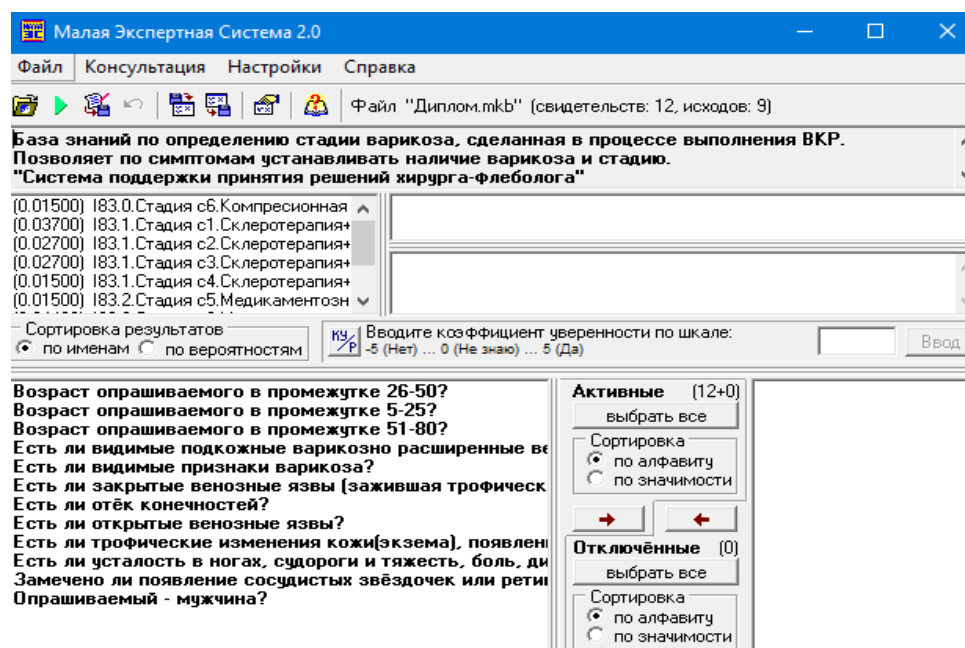


Рис. 6. Окно приложения с запущенной базой знаний

Примечание: скриншот авторов.

После запуска начала консультации необходимо ответить на все вопросы в соответствии с анамнезом и визуальным осмотром пациента, поставив значение в активное поле слева от кнопки «Ввод». Значение 0 воспринимается системой как ответ «Нет», а значение «1» как ответ «Да» соответственно. Если нельзя дать точного ответа, система воспринимает также и промежуточные значения,

желательно использовать значения не меньше десятых.

После ответа на все свидетельства область результатов изменится (рис. 7), некоторые исходы окрасятся в серый (стадии С0–С6), а оставшиеся (их может быть несколько) – в чёрный цвет и снабжены вероятностями на основе ответов пользователя – отсутствие варикоза ($p = 0,76463$), стадия А ($p = 0,03778$).

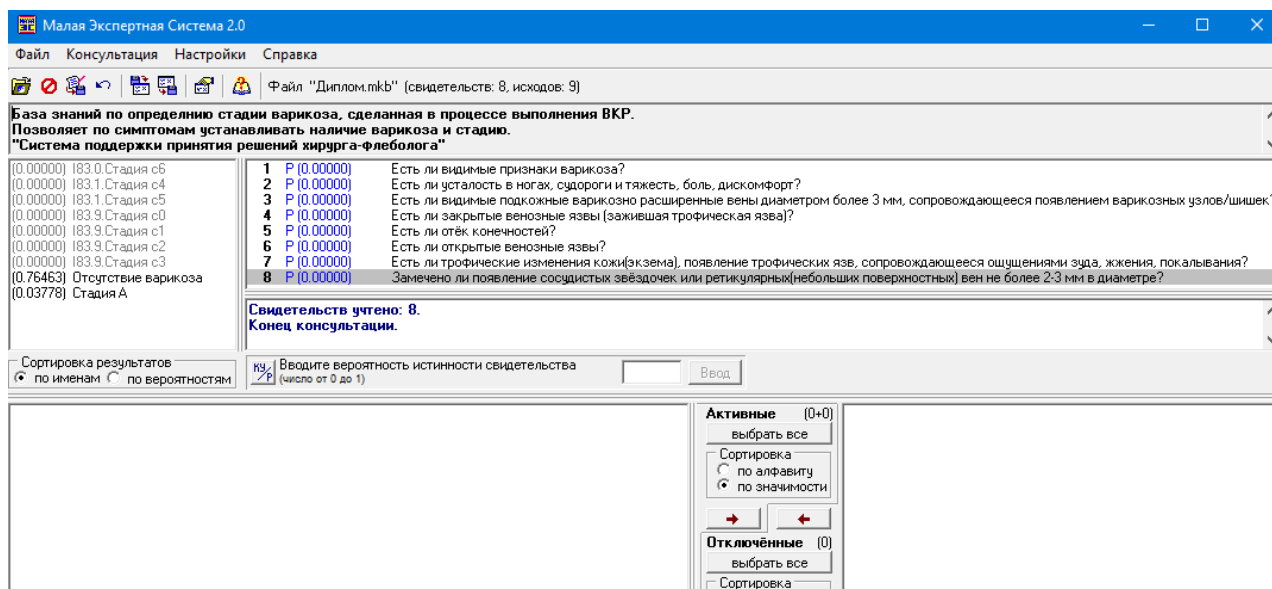


Рис. 7. Окончание консультации
 Примечание: скриншот авторов.

После окончания консультации необходимо сохранить протокол консультации с фамилией, инициалами и возрастом для

дальнейшего использования протокола в базе данных (рис. 8).

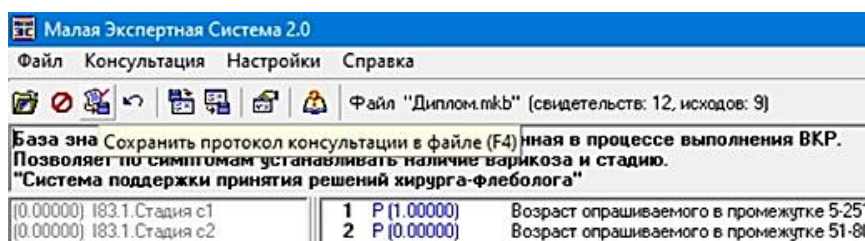


Рис. 8. Сохранение протокола консультации
 Примечание: скриншот авторов.

Система содержит 12 основных вопросов, исходя из которых рассчитывается класс патологии [7]:

1. Опрашиваемый – мужчина?
2. Возраст опрашиваемого в промежутке 5–25?
3. Возраст опрашиваемого в промежутке 26–50?

4. Возраст опрашиваемого в промежутке 51–80?
5. Есть ли видимые признаки варикоза?
6. Есть ли усталость в ногах, судороги и тяжесть, боль, дискомфорт?
7. Замечено ли появление сосудистых звездочек или ретикулярных (небольших поверхностных) вен не более 2–3 мм в диаметре?

8. Есть ли видимые подкожные варикозно-расширенные вены диаметром более 3 мм, сопровождающиеся появлением варикозных узлов/шишек?

9. Есть ли отек конечностей?

10. Есть ли трофические изменения кожи (экзема), появление трофических язв, сопровождающееся ощущениями зуда, жжения, покалывания?

11. Есть ли закрытые венозные язвы (зажившая трофическая язва)?

12. Есть ли открытые венозные язвы?

Также в данном приложении существует два подхода к вводу данных:

1. С использованием вероятности истинности свидетельства, где «0» – это «Нет», а «1» – «Да»;

2. С использованием коэффициента уверенности [8], которой дает возможность ответа «Не знаю», диапазон по умолчанию: –5 – ответ «Нет», 0 – «Не знаю», 5 – «Да».

Изменить диапазон коэффициента уверенности можно с использованием настройки предпочтений (рис. 9).

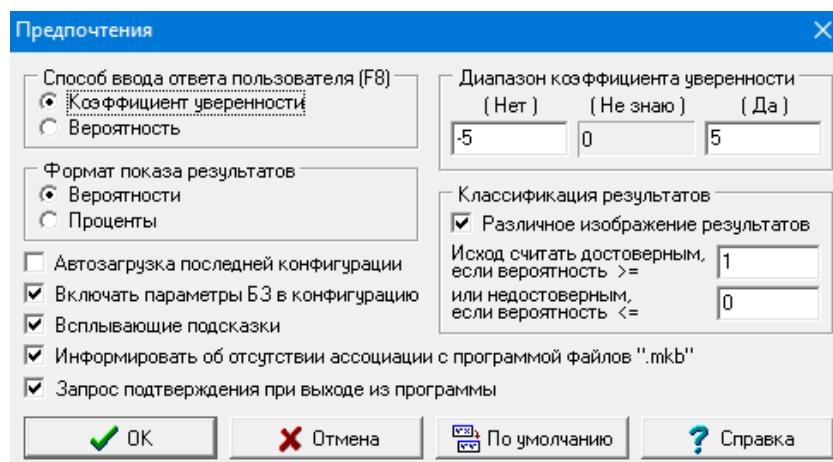


Рис. 9. Настройка предпочтений ЛПР

Примечание: скриншот авторов.

ЗАКЛЮЧЕНИЕ

Принятие решений в медицине является сложной и слабо формализованной задачей. СППР как модуль медицинской информационной системы предназначена в первую очередь для врачей-флебологов, так как для ее полноценного использования требуется

визуальный осмотр пациента, а также и для пациентов, которые могут получить консультацию самостоятельно. Но данные, полученные по окончании консультации, не являются диагнозом и требуют уточнений и комментариев врача.

Список источников

1. Доан Д. Х., Крошили А. В., Крошили С. В. Обзор подходов к проблеме принятия решений в медицинских информационных системах в условиях неопределенности // Фундаментал. исслед. 2015. № 12–1. С. 26–30.
2. Маленкова И. Н., Шакиров В. А. Многокритериальный анализ вариантов строительства электростанций методом PROMETHEE // Тр. Братск. гос. ун-та. Сер. Естеств. и инженер. науки. 2016. Т. 1. С. 41–50.
3. Низамутдинова Р. И. Система поддержки принятия коллективных решений при управлении взаимодействующими деловыми процессами в промышленности : дис. ... канд. техн. наук. Уфа, 2011. 217 с.

References

1. Doan D. Kh., Kroshilin A. V., Kroshilina S. V. Overview of Approaches to the Problem of Decision Making in Medical Information Systems in the Conditions of Uncertainty // Fundamental Research. 2015. No. 12–1. P. 26–30. (In Russian).
2. Malenkova I. N., Shakirov V. A. Mnogokriterialnyi analiz variantov stroitelstva elektrostantsii metodom PROMETHEE // Tr. Bratsk. gos. un-ta. Ser. estestv. i inzhener. nauki. 2016. Vol. 1. P. 41–50. (In Russian).
3. Nizamutdinova R. I. Sistema podderzhki priniatiia kolektivnykh reshenii pri upravlenii vzaimodeistviushchimi delovymi protsessami v promyshlennosti : Cand. Sci. Dissertation (Engineering). Ufa, 2011. 217 p. (In Russian).

4. Что такое дерево решений и где его используют? URL: <https://habr.com/ru/company/productstar/blog/523044/> (дата обращения: 25.06.2022).
5. Урманцева Н. Р., Громов В. А. Коллаборативная система сбора и обработки флебологических данных PHLEBOCLOUD // Вестник кибернетики. 2021. № 4 (44). С. 40–47.
6. Экспертные системы. Малая экспертная система 2.0. Редактор баз знаний 1.0. URL: <https://bourabai.ru/alg/mes2.htm> (дата обращения: 01.07.2022).
7. Иванова Ю. Ю. Клиника варикозного расширения вен нижних конечностей в различных стадиях по классификации CEAP // Научные исследования и разработки студентов : сб. материалов II Междунар. студ. науч.-практ. конф., Чебоксары, 1 декабря 2016 г. Чебоксары, 2016. С. 49–52.
8. Кобринский Б. А. Нечеткость и факторы уверенности вербальных и визуальных экспертных знаний // Нечеткие системы, мягкие вычисления и интеллектуальные технологии (НСМВИТ-2017) : тр. VII Всерос. науч.-практ. конф., Санкт-Петербург, 3–7 июля 2017 г. СПб., 2017. С. 83–91.
4. Chto takoe derevo reshenii i gde ego ispolzuiut? URL: <https://habr.com/ru/company/productstar/blog/523044/> (accessed: 25.06.2022). (In Russian).
5. Urmantseva N. R., Gromov V. A. PHLEBOCLOUD – A Collaborative System for Collecting and Processing of Phlebological Data // Proceedings in Cybernetics. 2021. No. 4 (44). P. 40–47. (In Russian).
6. Ekspertnye sistemy. Malaia ekspertnaia sistema 2.0. Redaktor baz znanii 1.0. URL: <https://bourabai.ru/alg/mes2.htm> (accessed: 01.07.2022). (In Russian).
7. Ivanova Yu. Yu. Klinika varikoznogo rasshireniia ven nizhnikh konechnostei v razlichnykh stadiiakh po klassifikatsii CEAP // Nauchnye issledovaniia i razrabotki studentov : Proceedings of the II International Student Research-to-Practice Conference, Cheboksary, December 1, 2016. Cheboksary, 2016. P. 49–52. (In Russian).
8. Kobrinsky B. A. Fuzzy and Certainty Factor of Verbal and Visual Expert Knowledge // Nechetkie sistemy, miagkie vychisleniia i intellektualnye tekhnologii (NSMVIT-2017) : Proceedings of the VII All-Russian Research-to-Practice Conference, Saint Petersburg, July 3–7, 2017. Saint Petersburg, 2017. P. 83–91. (In Russian).

Информация об авторах

Н. Р. Урманцева – старший преподаватель.

А. В. Курамшина – кандидат экономических наук, доцент.

Information about the authors

N. R. Urmantseva – Senior Lecturer.

A. V. Kuramshina – Candidate of Sciences (Economics), Associate Professor.

ФИЗИКО-МАТЕМАТИЧЕСКИЕ НАУКИ

Научная статья

УДК 536.46

doi: 10.34822/1999-7604-2022-3-75-83

ИЗУЧЕНИЕ РАСПРОСТРАНЕНИЯ ПЛАМЕНИ В РАСШИРЯЮЩЕЙСЯ СФЕРЕ

Максим Михайлович Алексеев¹, Олег Юрьевич Семенов^{2✉}

^{1,2} Сургутский государственный университет, Сургут, Россия

¹ amm.iff@gmail.com, <https://orcid.org/0000-0002-8657-5803>

² ous.tutor.phinma@mail.ru ✉, <https://orcid.org/0000-0002-3553-7356>

Аннотация. Исследовано распространение пламени, развивающегося в сферическом объеме однородного предварительно смешанного горючего газа, находящегося в мыльном пузыре и воспламененного с помощью электрического разряда. Проведена визуализация распространения фронта пламени с помощью высокоскоростной цифровой видеокамеры. Получены зависимости скорости фронта пламени от времени и радиальных величин мыльного пузыря в широком диапазоне условий по видеозаписям серии экспериментов.

Результаты экспериментальных исследований могут использоваться для создания современных камер сгорания и более совершенных горелочных устройств.

Ключевые слова: фронт пламени, скорость, горение, пропан, мыльный пузырь, радиус сферы, теплодиффузионная неустойчивость, искровое зажигание, визуализация

Для цитирования: Алексеев М. М., Семенов О. Ю. Изучение распространения пламени в расширяющейся сфере // Вестник кибернетики. 2022. № 3 (47). С. 75–83. DOI 10.34822/1999-7604-2022-3-75-83.

Original article

STUDY OF FLAME PROPAGATION IN AN EXPANDING SPHERE

Maksim M. Alekseev¹, Oleg Yu. Semenov^{2✉}

^{1,2} Surgut State University, Surgut, Russia

¹ amm.iff@gmail.com, <https://orcid.org/0000-0002-8657-5803>

² ous.tutor.phinma@mail.ru ✉, <https://orcid.org/0000-0002-3553-7356>

Abstract. The article studies the flame propagation in a soap bubble filled with a spherical volume of a homogeneous pre-mixed combustible gas ignited by an electrical discharge. Visualization of the flame front propagation is carried out using a high-speed digital video camera. The video recordings of an experiment series show the relationship between the flame front velocity and the time and radial values of a soap bubble in various conditions.

The experiment results can be used for the invention of a modern combustion chamber and more sophisticated burner arrangements.

Keywords: flame front, velocity, combustion, propane, soap bubble, sphere radius, thermal diffusion instability, spark ignition, visualization

For citation: Alekseev M. M., Semenov O. Yu. Study of Flame Propagation in an Expanding Sphere // Proceedings in Cybernetics. 2022. No. 3 (47). P. 75–83. DOI 10.34822/1999-7604-2022-3-75-83.

ВВЕДЕНИЕ

Изучение пламени является очень важным для расчетов при возгорании газоздушных смесей в производственных и бытовых помещениях, создания огнепреградителей и защитных мембран. Данным исследованиям посвящено достаточно много работ, но имеются вопросы, требующие дополнительного изучения. Для описания процессов распространения пламени в мыльном пузыре авторами работ [1–5] были проведены исследования сферического пламени, постоянного объема и давления; выявлены большие расхождения в скоростях ламинарного пламени, измеренных в различных экспериментах для одной и той же смеси. В работах [6, 7] приводится описание радиального расширения сферической мыльной оболочки под действием экзотермической реакции горения при распространении фронта пламени в свободном пространстве, заполненном водородно-воздушной смесью.

Эксперименты, описанные в работах [8, 9], по изучению ускорения пламени проводили в трубке с критическим составом газоздушной смеси. Распространение и гашение пламени в каналах изучалось для понимания процессов теплообмена между пламенем и стенками канала, а также их влияния на скорость горения смеси газов. В работе [10] описывается нестационарное распространение пламени в узком канале Хеле-Шоу, оно имеет сложный характер для пламени метана и пропана. В [11] исследователи применили визуализацию потока при горении газа в мыльном пузыре с использованием лазерной томографии и шпирен-методов. Высокоскоростная шпирен-визуализация использовалась для изучения роста ядер возмущений пламени при различной их эквивалентности. В работах [12, 13] применялась технология высокоскоростной обработки изображений, проведено исследование турбулентного распространения пламени для получения новых данных движения фронта пламени.

В работе [14] произведены численные расчеты с помощью программы расчета объемных течений пропано- и метано-воздушных смесей. Нестационарное распространение пламени смешанных с воздухом метана и пропана было исследовано с использова-

нием дисковой горелки с параллельными пластинами в работах [15, 16]. Влияние тепловых и диффузионных процессов на структуру пламени в камерах сгорания описано в работах [17–19]. Значительное количество экспериментальных работ по исследованию процессов горения не дают достаточных сведений по описанию влияния теплодиффузионных и гидродинамических условий на характер движения и структуру пламени в сферически расширяющейся оболочке.

МАТЕРИАЛЫ И МЕТОДЫ

Изучение распространения пламени в замкнутых объемах проводилось с использованием сферической емкости, в которой источник инициации находился в центре. На рис. 1 показана схема экспериментальной установки. Газоздушные смеси с разной концентрацией пропана в воздухе готовили в газометре вытеснения, объем подаваемой смеси фиксировали с помощью расходомера газа. Для предотвращения обратного движения пламени в газовой трубке устанавливали пламегаситель.

Мыльный пузырь надували с помощью пропано-воздушной газовой смеси через газовую трубку системы подачи газа с воздухом. Далее, при достижении требуемых размеров мыльного пузыря – ситуации, когда искровой промежуток между электродами, выполненными из тонкой стальной проволоки, оказывался в центре сферы – производили воспламенение смеси электрической искрой с помощью электронной системы зажигания, подключенной к источнику напряжения.

После зажигания образовывалось пламя в форме сферы, постепенно распространяющееся по всему объему газовой смеси – от центра мыльного пузыря к его краям. Для видеофиксации результатов экспериментов применяли высокоскоростную камеру марки AOS Technologies AGX-PRI с автоматической передачей экспериментальных данных в персональный компьютер и зеркальную цифровую камеру Sony DSLRA580. Для качественной визуализации процесса распространения пламени устанавливали электрическую лампу, световой поток которой был направлен на мыльный пузырь.

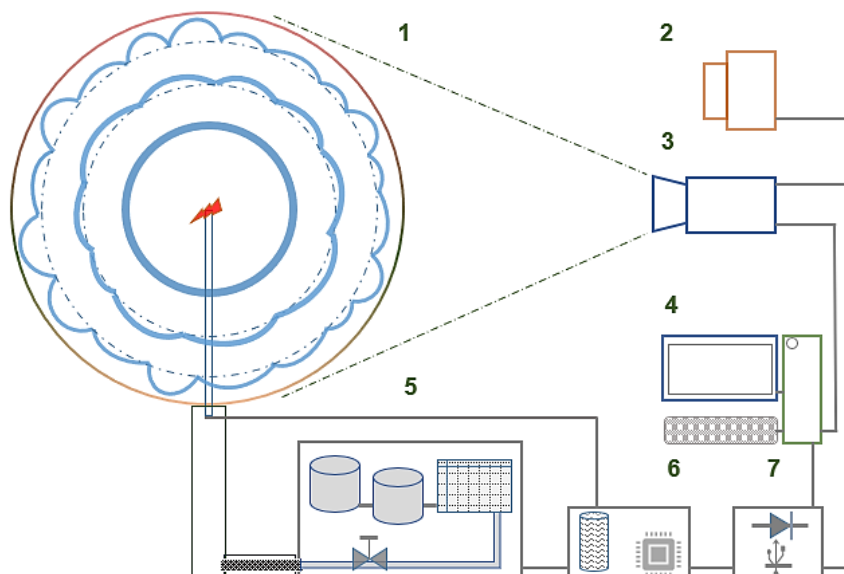


Рис. 1. Схема экспериментальной установки:

- 1 – распространяющийся фронт пламени в мыльном пузыре; 2 – электрическая лампа подсветки;
3 – высокоскоростная камера; 4 – компьютер обработки данных;
5 – система подачи газовой смеси: газометр, трубка подачи газа с пламегасителем и расходомер;
6 – электронная система зажигания; 7 – источник напряжения

Примечание: составлено авторами на основании данных, полученных в исследовании.

Фронт пламени, инициированный в замкнутой емкости – мыльной сфере, распространяется в сторону несгоревшего газа при поджигании горючей смеси электрической искрой. Нормальная скорость пламени является важным параметром процесса горения, имеющего свойство физико-химической постоянства для определенной газовой смеси. Наблюдаемую скорость пламени u в мыльной сфере можно определить, зная скорость движения газовой смеси w в процессе ее сферического расширения, а также с помощью нормальной скорости u_n пламени.

Скорость фронта пламени u горючей смеси в мыльном пузыре характеризуется следующим выражением:

$$u = u_n + w, \quad (1)$$

где u_n – нормальная скорость фронта пламени, w – радиальный поток.

Степень расширения газовой смеси зависит от объема V_2 продуктов сгорания в расширяющейся сфере:

$$V_2 = \frac{4 \times \pi \times R_2^3}{3} \quad (2)$$

и начального объема V_1 пропано-воздушной горючей смеси:

$$V_1 = \frac{4 \times \pi \times R_1^3}{3}, \quad (3)$$

где R_1 – начальный радиус;

R_2 – конечный радиус мыльного пузыря.

Таким образом, нормальная скорость распространения пламени в мыльном пузыре будет определяться из уравнения:

$$u_n = u \times \left(\frac{R_1}{R_2} \right)^3. \quad (4)$$

Производя измерения наблюдаемой скорости фронта пламени u по цифровой видеозаписи, применяя уравнение (4) с учетом начального R_1 и конечного R_2 радиусов мыльного пузыря, можно вычислить нормальную скорость распространения пламени u_n в сферическом объеме.

Из опытов было установлено, что пламя принимает форму расширяющейся сферы с неровной поверхностью, увеличивающейся в объеме от центра к стенкам. Через некото-

рое время после воспламенения смеси давление в мыльном пузыре сначала растет пропорционально времени, после чего происходит незначительное его изменение, связанное с возрастанием скорости горения в результате нагревания новой несгоревшей смеси перед фронтом пламени в результате адиабатического сжатия. В сферическом мыльном пузыре возникают колебания давления газа, которые влияют на формирование ячеек на поверхности фронта пламени. Радиальные потоки смеси газа в мыльной сфере появляются из-за разности температур и давлений между продуктами сгорания и горючей смесью. Осцилляции фронта пламени вызваны теплодиффузионной и гидродинамической неустойчивостью процесса

горения при распространении пламени, развивающегося в определенных условиях эксперимента.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

На рис. 2 показана эволюция фронта пламени при воспламенении пропано-воздушной смеси. Мыльный пузырь практически не оказывает влияния на тепловое расширение находящегося в нем пропана с воздухом, и пламя распространяется сферически при постоянном давлении. Светящийся шар в мыльном пузыре состоит из горячих продуктов горения, ограниченных по периферии узким сферическим фронтом пламени, разделяющим сгоревший и несгоревший газ.

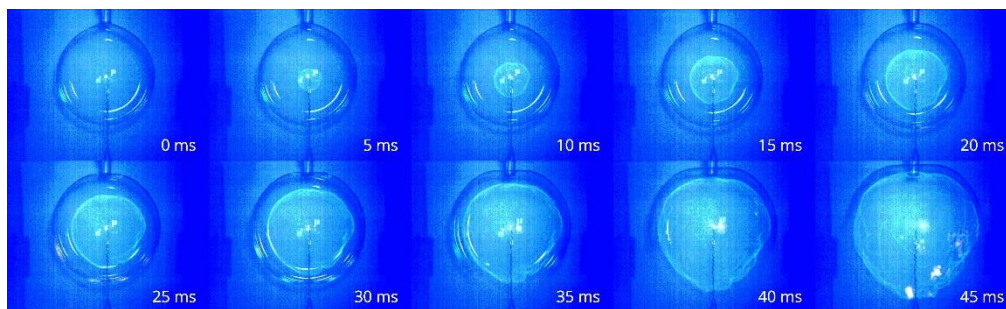


Рис. 2. Фотокадры дефлаграционного горения 5 %-го пропана и 95 %-го воздуха в мыльном пузыре (интервал времени между кадрами – 5 мс)

Примечание: составлено авторами на основании данных, полученных в исследовании.

На фотокадрах записи хорошо видно развитие гладкой сферической формы пламени, которая сохраняется по мере распространения несгоревшей смеси. Далее, после некоторого времени возникают возмущения – об-

разуются ячейки различного масштаба на поверхности пламени.

Рис. 3 иллюстрирует динамику развития пламени с содержанием 7 %-го пропана.

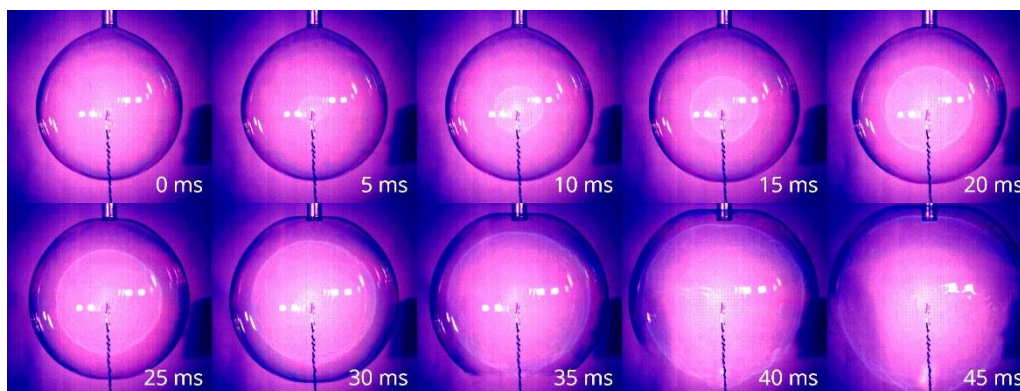


Рис. 3. Фотокадры дефлаграционного горения 7 %-го пропана и 93 %-го воздуха

Примечание: составлено авторами на основании данных, полученных в исследовании.

Пламя остается почти сферическим до момента, когда оно начинает подниматься. Рост мыльного пузыря происходит по мере распространения пламени к его краям, из фотографий экспериментов видны неустойчивости, развивающиеся в структуре пламени. Сфера сгоревших продуктов стремится подняться вверх в результате конвекции. Мыльный сферический пузырь растягивает-

ся по мере распространения пламени в соответствии с увеличением объема за счет теплового расширения. Объем зависит от диаметра пузыря в третьей степени, и мыльный пузырь расширяется постепенно – до тех пор, пока диаметр пламени не станет равным половине первоначального диаметра пузыря (рис. 4).

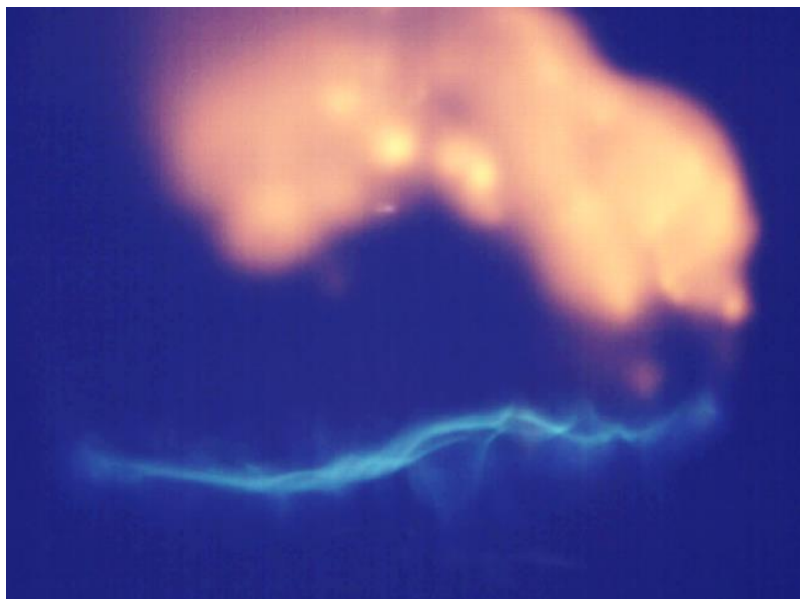


Рис. 4. Грибовидный фронт пламени

Примечание: составлено авторами на основании данных, полученных в исследовании.

Далее происходит подъем и растяжение фронта пламени по мере его приближения к верхней поверхности пленки пузыря. При достижении фронта пламени мыльной пленки она лопается. После ее разрушения сфера пламени принимает грибовидную форму и существует еще в течение некоторого времени, затем ее граница размывается и становится неровной вследствие перемешивания смеси с окружающим воздухом и остатками мыльной пленки.

При разрыве пузырьковой пленки на поверхности пламени наблюдались возмущения (рис. 5). Эти возмущения через некоторое время нарастали из-за неустойчивостей Рэлея – Тейлора. Идентификация фронта пламени становится затруднительной после разрыва мыльного пузыря. Несгоревшая часть газа остается непрореагировавшей во

время переходного конвекционного процесса, что подавляет распространение пламени.

На рис. 6 приведены зависимости диаметра фронта пламени и мыльного пузыря от времени. Как следует из графика, скорость распространения пламени относительно размера мыльной сферы остается практически постоянной. Диаметр фронта пламени возрастает линейно при достижении стенок пузыря.

Из графика зависимости диаметра пузыря от диаметра фронта пламени (рис. 7) возможно определить нормальную скорость распространения пламени в мыльном пузыре.

Значение нормальной скорости фронта пламени движущегося по горючей пропановоздушной смеси определенного состава является различным в зависимости от процентного соотношения пропана в воздухе.



Рис. 5. Кадр распада мыльного пузыря с волнообразными возмущениями на его поверхности
Примечание: составлено авторами на основании данных, полученных в исследовании.

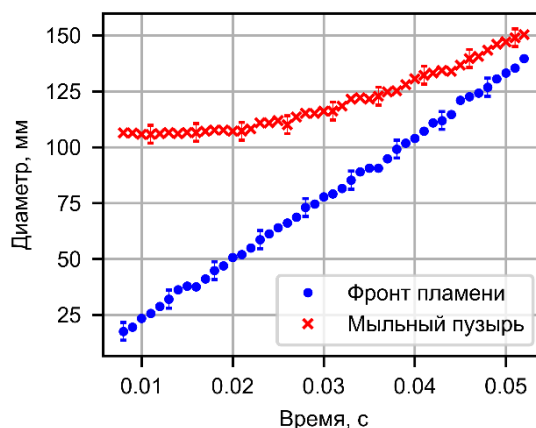


Рис. 6. График зависимости диаметра фронта пламени и мыльного пузыря от времени
Примечание: составлено авторами на основании данных, полученных в исследовании.

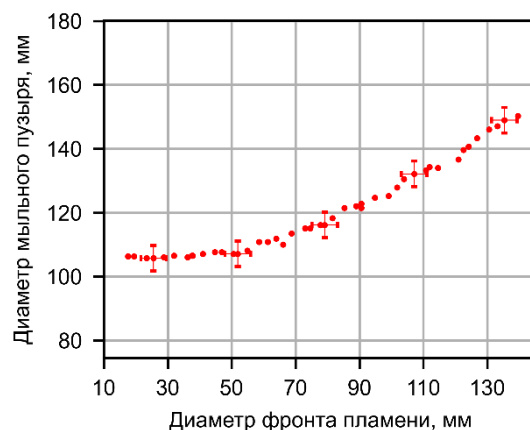


Рис. 7. График зависимости диаметра пузыря от диаметра фронта пламени
Примечание: составлено авторами на основании данных, полученных в исследовании.

Искривленная форма поверхности фронта пламени стабилизируется за счет увеличения скорости газа и уменьшения скорости горения в одних участках, а также уменьшения

скорости газа и увеличения скорости горения в других. На рис. 8 представлены зависимости скорости движения границы пузыря и фронта пламени от их диаметров. Они вы-

числены через производную зависимостей радиусов от времени. Так как радиусы измеряются по изображениям с точностью до од-

ного пикселя, исходные данные о зависимости радиусов от времени получаются «зашумленными» из-за погрешности измерений.

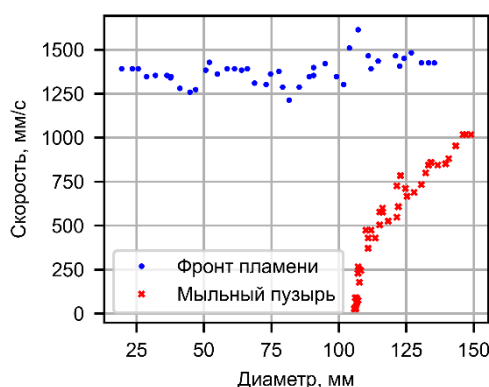


Рис. 8. Скорость движения границы пузыря и фронта пламени в зависимости от их диаметров
Примечание: составлено авторами на основании данных, полученных в исследовании.

При численном нахождении производной от таких «зашумленных» данных скорость меняется в больших пределах (рис. 9).

Линии на графиках зависимостей диаметров от времени – это и есть сглаженные данные, но даже по сглаженным фильтром данным скорость немного меняется, т. к. движения границы пузыря и фронта неравномерны.

Для того чтобы вычислить производную от радиуса по времени, исходные измерения были сглажены фильтром Савицкого – Голея. При более сильном сглаживании любым фильтром кроме устранения шумов теряются особенности на исходных данных, что совершенно не соответствует действительности, связанной с особенностью этого фильтра.



Рис. 9. Скорость движения границы пузыря и фронта от времени
Примечание: составлено авторами на основании данных, полученных в исследовании.

По графикам зависимости скорости движения границы мыльного пузыря и фронта пламени от их диаметров и времени зафиксировано, что сферически распространяющееся пламя действует на мыльную оболочку пузыря, заставляя ее радиально расширяться от центральной точки зажигания с проявлением гидродинамических, диффузионных и тепловых явлений.

ЗАКЛЮЧЕНИЕ

В работе применена методика исследования процессов распространения газоздушного пламени в мыльных пузырях. Из экспериментов установлено, что максимум нормальной скорости пламени смещен в сторону избытка горючего газа по отношению к смеси стехиометрического состава, в сферическом пламени поток тепла от сгоревшего га-

за к несгоревшему является расходящимся. Результаты экспериментальных исследований могут использоваться для создания со-

временных камер сгорания и более совершенных горелочных устройств.

Список источников

1. Faghih M., Chen Z. The Constant-Volume Propagating Spherical Flame Method for Laminar Flame Speed Measurement Science // Science Bulletin. 2016. Vol. 61, Is. 16. P. 1296–1310. DOI 10.1007/s11434-016-1143-6.
2. Vledouts A., Quinard J., Vandenberghe N., Villermaux E. Explosive Fragmentation of Liquid Shells // J Fluid Mech. 2016. Vol. 788. P. 246–273. DOI 10.1017/jfm.2015.716.
3. Li F.-S., Li G.-X., Jiang Y.-H., Li H.-M., Sun Z.-Y. Study on the Effect of Flame Instability on the Flame Structural Characteristics of Hydrogen/Air Mixtures Based on the Fast Fourier Transform // Energies. 2017. Vol. 10, No. 5. P. 678. DOI 10.3390/en10050678.
4. Askari O., Elia M., Ferrari M., Metghalchi H. Cell Formation Effects on the Burning Speeds and Flame Front Area of Synthetic Gas at High Pressures and Temperatures // Appl Energy 2017. Vol. 189, Is. C. P. 568–577.
5. Choi O., Lee M. C. Investigation into the Combustion Instability of Synthetic Natural Gases Using High Speed Flame Images and Their Proper Orthogonal Decomposition // Int J Hydrog Energy. 2016. Vol. 41, Is. 45. P. 20731–20743.
6. Володин В. В., Голуб В. В., Ельянов А. Е., Коробов А. Е., Микушкин А. Ю., Петухов В. А. Влияние объема водородно-воздушной газовой смеси, типа и энергии инициирования на распространение сферического фронта пламени // Вестн. МГТУ им. Н. Э. Баумана. Сер. Естественные науки. 2019. № 2. С. 64–80. DOI 10.18698/1812-3368-2019-2-64-80.
7. Vledouts A., Quinard J., Vandenberghe N., Villermaux E. Explosive Fragmentation of Liquid Shells // J Fluid Mech Cambridge University Press. 2016. Vol. 788. P. 246–273. DOI 10.1017/jfm.2015.716.
8. Семенов О. Ю. Исследование релаксационного распространения пламени в каналах : автореф. дис. ... канд. физ.-мат. наук. Томск, 2014. 23 с. URL: <https://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000476181> (дата обращения: 01.04.2022).
9. Ciccarelli G., Chaumeix N., Mendiburu A. Z., N'Guessan K., Comandini A. Fast-Flame Limit for Hydrogen/Methane-Air Mixtures // Proceedings of the Combustion Institute. 2019. Vol. 37, Is. 3. P. 3661–3668. DOI 10.1016/j.proci.2018.06.045.
10. Alekseev M. M., Alekseev M. V., Samsonov V. P., Semenov O. Yu. Simulation of Hydrodynamic Phenomena Attendant on the Flame Front Propagation in a Tube Behind a Piston // Tech Phys. 2014. Vol. 59, No. 1. P. 52–59.

References

1. Faghih M., Chen Z. The Constant-Volume Propagating Spherical Flame Method for Laminar Flame Speed Measurement Science // Science Bulletin. 2016. Vol. 61, Is. 16. P. 1296–1310. DOI 10.1007/s11434-016-1143-6.
2. Vledouts A., Quinard J., Vandenberghe N., Villermaux E. Explosive Fragmentation of Liquid Shells // J Fluid Mech. 2016. Vol. 788. P. 246–273. DOI 10.1017/jfm.2015.716.
3. Li F.-S., Li G.-X., Jiang Y.-H., Li H.-M., Sun Z.-Y. Study on the Effect of Flame Instability on the Flame Structural Characteristics of Hydrogen/Air Mixtures Based on the Fast Fourier Transform // Energies. 2017. Vol. 10, No. 5. P. 678. DOI 10.3390/en10050678.
4. Askari O., Elia M., Ferrari M., Metghalchi H. Cell Formation Effects on the Burning Speeds and Flame Front Area of Synthetic Gas at High Pressures and Temperatures // Appl Energy 2017. Vol. 189, Is. C. P. 568–577.
5. Choi O., Lee M. C. Investigation into the Combustion Instability of Synthetic Natural Gases Using High Speed Flame Images and Their Proper Orthogonal Decomposition // Int J Hydrog Energy. 2016. Vol. 41, Is. 45. P. 20731–20743.
6. Volodin V. V., Golub V. V., Elyanov A. E., Korobov A. E., Mikushkin A. Yu., Petukhov V. A. Effects of Hydrogen-Air Gas Mixture Volume and Initiation Type and Energy on the Propagation of a Spherical Flame Front // Herald of the Bauman Moscow State Technical University. Series Natural Sciences. 2019. No. 2. P. 64–80. DOI 10.18698/1812-3368-2019-2-64-80. (In Russian).
7. Vledouts A., Quinard J., Vandenberghe N., Villermaux E. Explosive Fragmentation of Liquid Shells // J Fluid Mech Cambridge University Press. 2016. Vol. 788. P. 246–273. DOI 10.1017/jfm.2015.716.
8. Semenov O. Yu. Issledovanie relaksatsionnogo rasprostraneniia plameni v kanalakh : Extended abstract Can. Sci. Dissertation (Physics and Mathematics). Tomsk, 2014. 23 p. URL: <https://vital.lib.tsu.ru/vital/access/manager/Repository/vtls:000476181> (accessed: 01.04.2022). (In Russian).
9. Ciccarelli G., Chaumeix N., Mendiburu A. Z., N'Guessan K., Comandini A. Fast-Flame Limit for Hydrogen/Methane-Air Mixtures // Proceedings of the Combustion Institute. 2019. Vol. 37, Is. 3. P. 3661–3668. DOI 10.1016/j.proci.2018.06.045.
10. Alekseev M. M., Alekseev M. V., Samsonov V. P., Semenov O. Yu. Simulation of Hydrodynamic Phenomena Attendant on the Flame Front Propagation in a Tube Behind a Piston // Tech Phys. 2014. Vol. 59, No. 1. P. 52–59.

11. Nandan V. D., Deore N. Developments of Flame Propagation in Micro and Mesoscale Channels // IOSR Journal of Engineering. 2014. Vol. 04, Is. 06. P. 26–29.
12. Алексеев М. М., Семенов О. Ю. Физическое моделирование тюльпанообразного пламени при горении газов в цилиндрической вертикальной трубе // Вестн. кибернетики. 2021. № 1 (41). С. 63–70.
13. Jang H. J., Lee S. M., Kim N. I. Effects of Ignition Disturbance on Flame Propagation of Methane and Propane in a Narrow-Gap-Disk-Burner // Combustion and Flame. 2020. Vol. 215. P. 124–133. DOI 10.1016/j.combustflame.2020.01.019.
14. Алексеев М. М., Борисов В. Е., Семенов О. Ю., Якуш С. Е. Моделирование горения в узком плоском канале // Препринты ИПМ им. М. В. Келдыша. 2016. Т. 134. 32 с. DOI 10.20948/prepr-2016-134.
15. Badawy T., Chao X. B., Xu H. Impact of Spark Plug Gap on Flame Kernel Propagation and Engine Performance // Applied Energy. 2017. Vol. 191. P. 311–327. DOI 10.1016/j.apenergy.2017.01.059.
16. Jang H. J., Jang G. M., Kim N. I. Unsteady Propagation of Premixed Methane/Propane Flames in a Mesoscale Disk Burner of Variable-Gaps // Proceedings of the Combustion Institute. 2018. Vol. 37, Is. 2. P. 1861–1868. DOI 10.1016/j.proci.2018.06.112.
17. Radisson B., Piketty-Moine J., Almarcha C. Coupling of Vibro-Acoustic Waves with Premixed Flame // Phys Rev Fluids. 2019. Vol. 4, Is. 12. P. 121201.
18. Zhao D., Li J. Feedback Control of Combustion Instabilities Using a Helmholtz Resonator with an Oscillating Volume // Combustion Science and Technology. 2012. Vol. 184, Is. 5. P. 694–716.
19. Kiverin A., Yakovenko I. Mechanism of Transition to Detonation in Unconfined Volumes // Acta Astronautica. 2020. Vol. 176. P. 647–652. DOI 10.1016/j.actaastro.2020.02.013.
11. Nandan V. D., Deore N. Developments of Flame Propagation in Micro and Mesoscale Channels // IOSR Journal of Engineering. 2014. Vol. 04, Is. 06. P. 26–29.
12. Alekseev M. M., Semenov O. Yu. Tulip Flame Physical Modeling during Gaseous Combustion in a Cylindrical Vertical Pipe // Proceedings in Cybernetics. 2021. No. 1 (41). P. 63–70. (In Russian).
13. Jang H. J., Lee S. M., Kim N. I. Effects of Ignition Disturbance on Flame Propagation of Methane and Propane in a Narrow-Gap-Disk-Burner // Combustion and Flame. 2020. Vol. 215. P. 124–133. DOI 10.1016/j.combustflame.2020.01.019.
14. Alekseev M. M., Borisov V. E., Semenov O. Yu., Yakush S. E. Numerical Study on Combustion in a Plane Narrow Channel // Keldysh Institute Preprints. 2016. Vol. 134. 32 p. DOI 10.20948/prepr-2016-134. (In Russian).
15. Badawy T., Chao X. B., Xu H. Impact of Spark Plug Gap on Flame Kernel Propagation and Engine Performance // Applied Energy. 2017. Vol. 191. P. 311–327. DOI 10.1016/j.apenergy.2017.01.059.
16. Jang H. J., Jang G. M., Kim N. I. Unsteady Propagation of Premixed Methane/Propane Flames in a Mesoscale Disk Burner of Variable-Gaps // Proceedings of the Combustion Institute. 2018. Vol. 37, Is. 2. P. 1861–1868. DOI 10.1016/j.proci.2018.06.112.
17. Radisson B., Piketty-Moine J., Almarcha C. Coupling of Vibro-Acoustic Waves with Premixed Flame // Phys Rev Fluids. 2019. Vol. 4, Is. 12. P. 121201.
18. Zhao D., Li J. Feedback Control of Combustion Instabilities Using a Helmholtz Resonator with an Oscillating Volume // Combustion Science and Technology. 2012. Vol. 184, Is. 5. P. 694–716.
19. Kiverin A., Yakovenko I. Mechanism of Transition to Detonation in Unconfined Volumes // Acta Astronautica. 2020. Vol. 176. P. 647–652. DOI 10.1016/j.actaastro.2020.02.013.

Информация об авторах

М. М. Алексеев – кандидат физико-математических наук, доцент.

О. Ю. Семенов – кандидат физико-математических наук, доцент.

Information about the authors

M. M. Alekseev – Candidate of Sciences (Physics and Mathematics), Associate Professor.

O. Yu. Semenov – Candidate of Sciences (Physics and Mathematics), Associate Professor.

Научная статья
УДК 004.032.26
doi: 10.34822/1999-7604-2022-3-84-98

СРАВНЕНИЕ ТИПОВ ЯДЕР В СВЕРТОЧНЫХ СЛОЯХ НЕЙРОННЫХ СЕТЕЙ

Вахит Мансурович Гиниятуллин^{1✉}, **Александр Владимирович Хлыбов**²,
Михаил Андреевич Федоров³, **Тамерлан Азаматович Асадуллин**⁴,
Андрей Сергеевич Крутин⁵, **Илья Андреевич Осипов**⁶, **Дамир Мунзирович Зарипов**⁷
^{1, 2, 3, 4, 5, 6, 7} Уфимский государственный нефтяной технический университет, Уфа, Россия
¹ fentazer@mail.ru ✉, <https://orcid.org/0000-0002-4686-2752>
² brinkinvision@gmail.com, <https://orcid.org/0000-0002-9586-0681>
³ MasterOfHoMM@gmail.com, <https://orcid.org/0000-0002-6762-7831>
⁴ tealredplanet@gmail.com, <https://orcid.org/0000-0002-6660-6158>
⁵ krut_inuly@mail.ru, <https://orcid.org/0000-0001-7412-2391>
⁶ warluswarlusgg@gmail.com, <https://orcid.org/0000-0002-3570-6738>
⁷ damir.zaripov@gmail.com, <https://orcid.org/0000-0001-8912-0321>

Аннотация. Рассмотрена процедура использования фиксированных задаваемых сверточных ядер. Расчеты проведены с изменением их размерности и шага сдвига. Для исследований взяты четыре различных набора изображений и структур нейронных сетей. В рамках статьи обсуждаются четыре вида задаваемых ядер: вертикальные, горизонтальные, диагональные и кольцевые. Исследовалась зависимость точности распознавания изображений в монохромном и цветном представлениях. При дообучении фиксированных ядер наблюдается сдвиг в положительную сторону.

Ключевые слова: сверточное ядро, размер и шаг ядра, воспроизводимость результатов, глубокое обучение, первоначальное приближение, дискретность

Благодарности: работа выполнена при поддержке федерального государственного бюджетного учреждения «Фонд содействия развитию малых форм предприятий в научно-технической сфере» (Фонд содействия инновациям).

Для цитирования: Гиниятуллин В. М., Хлыбов А. В., Федоров М. А., Асадуллин Т. А., Крутин А. С., Осипов И. А., Зарипов Д. М. Сравнение типов ядер в сверточных слоях нейронных сетей // Вестник кибернетики. 2022. № 3 (47). С. 84–98. DOI 10.34822/1999-7604-2022-3-84-98.

Original article

COMPARING TYPES OF KERNELS IN CONVOLUTIONAL LAYERS OF NEURAL NETWORKS

Vakhit M. Giniyatullin^{1✉}, **Aleksandr V. Khlybov**², **Mikhail A. Fedorov**³,
Tamerlan A. Asadullin⁴, **Andrey S. Krutin**⁵, **Ilya A. Osipov**⁶, **Damir M. Zaripov**⁷
^{1, 2, 3, 4, 5, 6, 7} Ufa State Oil Technical University, Ufa, Russia
¹ fentazer@mail.ru ✉, <https://orcid.org/0000-0002-4686-2752>
² brinkinvision@gmail.com, <https://orcid.org/0000-0002-9586-0681>
³ MasterOfHoMM@gmail.com, <https://orcid.org/0000-0002-6762-7831>
⁴ tealredplanet@gmail.com, <https://orcid.org/0000-0002-6660-6158>
⁵ krut_inuly@mail.ru, <https://orcid.org/0000-0001-7412-2391>
⁶ warluswarlusgg@gmail.com, <https://orcid.org/0000-0002-3570-6738>
⁷ damir.zaripov@gmail.com, <https://orcid.org/0000-0001-8912-0321>

Abstract. The article studies the process of using fixed specified convolutional kernels. The calculations are conducted considering the change in filter size and stride. In the course of the study, four different

sets of images and structures of neural networks are selected. The article discusses four types of specified kernels: vertical, horizontal, diagonal, and ring. The dependency of the accuracy of image recognition in monochrome and color is analyzed. A positive shift is observed when training fixed kernels.

Keywords: convolutional kernel, kernel's size and stride, reproducibility, deep learning, initial approximation, discreteness

Acknowledgments: the article is funded by the Fund for Assistance to Small Innovative Enterprises (Innovation Assistance Fund).

For citation: Giniyatullin V. M., Khlybov A. V., Fedorov M. A., Asadullin T. A., Krutin A. S., Osipov I. A., Zaripov D. M. Comparing Types of Kernels in Convolutional Layers of Neural Networks // Proceedings in Cybernetics. 2022. No. 3 (47). P. 84–98. DOI 10.34822/1999-7604-2022-3-84-98.

ВВЕДЕНИЕ

Применение сверточных слоев в искусственной нейронной сети (далее – ИНС) в задачах распознавания образов весьма популярно. Обычно в парадигме глубокого обучения (deep learning) [1–3] используются десятки обучаемых ядер. В большинстве случаев их разносят по нескольким подряд идущим слоям. Размер ядер 3×3 или 5×5 , шаг по умолчанию равен 1, затем следуют полносвязные нейронные слои.

В полносвязном слое обучаемые веса нейронной сети представляют собой вектор вещественных значений. В сверточных слоях вектор весов заменен на плоскую (двумерную) матрицу и называется ядром свертки. Сверточное ядро сдвигается по изображению, преобразованному в числовую матрицу, с некоторым шагом, сокращая (сворачивая) его исходную размерность в меньшую. Сокращение размерности зависит как от размера сверточного ядра, так и от значения его шага.

Известно, что можно обойтись и без сверточных слоев, но тогда количество нейронов полносвязного слоя и количество обучаемых параметров растет пропорционально площади входного изображения [1, 2]. При использовании ядер большого размера (в работе [3] упоминаются ядра размером 22×22) за счет увеличения шага свертки можно радикально

(пропорционально квадрату шага) сократить размер выходного вектора.

Таким образом, задача состоит в сокращении вычислительных затрат как при обучении искусственной нейронной сети, так и на этапе рабочего хода, при этом вероятность удачного исхода обучения должна увеличиться.

В работе [4] было показано, что коэффициенты ядер можно задавать дискретными (использовались значения $-1, 0, +1$) в некоторой переборной процедуре, а в работе [5] упоминается большая дискретность значений весов сверточного слоя. Количество итераций в процедуре полного перебора – это степенная функция от размерности ядра, следовательно, перебор многозначных ядер большого размера практически невыполним.

Авторы статьи видоизменяли коэффициенты ядер только первого сверточного слоя и решили ограничиться:

- размером ядер от 5×5 до 9×9 ;
- значениями коэффициентов ядер в пятизначном диапазоне $(-2, -1, 0, +1, +2)$;
- четырьмя структурами ядер (вертикальные, горизонтальные, диагональные и кольцевые).

«Вертикальные», «горизонтальные» и «диагональные» ядра можно отнести к типу «линейных» ядер (рис. 1).

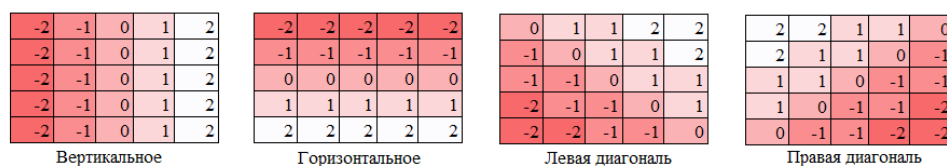


Рис. 1. Линейные задаваемые ядра

Примечание: составлено авторами.

Генерация большей размерности таких ядер происходит путем последовательного добавления для каждой большей размерности значений из выбранного диапазона дискретности с сохранением симметрии на нечетных размерах. Например, из размерности 5×5 (рис. 1) можно получить ядра размер-

ности 6×6 путем добавления дополнительных строк/столбцов и диагоналей со значением «1». Для следующего нечетного размера (7×7) добавляются элементы со значением «-1» (рис. 2). Дальнейшее масштабирование происходит аналогичным образом, добавлением «двоек».

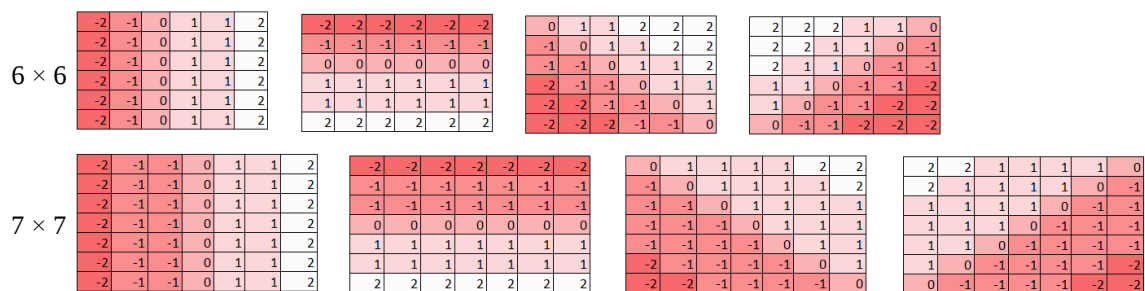


Рис. 2. Масштабирование линейных ядер

Примечание: составлено авторами.

Генерация «кольцевых» ядер происходила в обратном порядке. Бралось ядро большой размерности (например, 9×9), и уменьшался его размер за счет удаления двойных диаго-

налей по часовой стрелке (рис. 3). На рис. 3 зеленым цветом отмечены диагонали, которые будут удалены на следующем шаге.

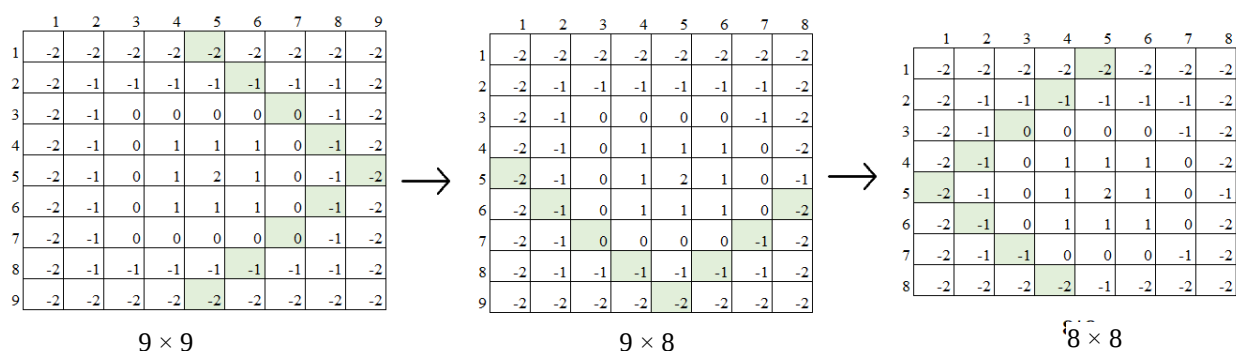


Рис. 3. Масштабирование кольцевых ядер

Примечание: составлено авторами.

Поскольку для последующих слоев используются стандартные процедуры обучения, общий результат является невоспроизводимым, авторы в результате обсуждения пришли к выводу, что можно принять отклонение в расчетах менее 4 % за несущественную разницу.

МАТЕРИАЛЫ И МЕТОДЫ

Проведенные исследования были выполнены на четырех наборах данных (датасетах) [6–9] и заранее заданных структурах нейронных сетей [10–13].

Характеристики датасетов представлены в табл. 1.

В табл. 2 предпринята попытка обобщения структур используемых нейронных сетей. В структуре ИНС PneumoniaDetection, в отличие от всех остальных, после каждого сверточного слоя присутствует слой BatchNormalization (для экономии пространства он не включен в табл. 2). Знак «+» в ячейках таблицы означает наличие слоя в структуре.

На рис. 4 приведен пример задания структуры нейронной сети для датасета Fruits 360.

Таблица 1

Описание наборов данных

Датасет	Fruits 360	Chest X-Ray Images (Pneumonia)	Malaria Dataset	Fashion MNIST
Краткое описание	Набор изображений фруктов	Рентгенограммы грудной клетки	Изображения клеток для прогнозирования заболевания малярией	Черно-белые изображения одежды
Размерность изображений, px	100 × 100	150 × 150	130 × 130	28 × 28
Количество цветовых каналов, однотонное / RGB	3	1	3	1
Обучающая выборка, шт	4306	5216	27600	60000
Тестовая выборка, шт	1445	624	15832	10000

Примечание: составлено авторами.

Таблица 2

Структуры нейронных сетей

	CNN Fruit Classification	Pneumonia Detection	Detecting Malaria	Fashion MNIST – Keras CNN
Размерность входного изображения	(100, 100, 3)	(150, 150, 1)	(130, 130, 3)	(28, 28, 1)
Сверточный слой	8 ядер 3 × 3	32 ядра 3 × 3	32 ядра 3 × 3	16 ядер 2 × 2
Maxpooling	Ядро 2×2, шаг 1	Ядро 2×2, шаг 2	Ядро 2×2, шаг 2	Ядро 2×2, шаг 2
Dropout	+			
Сверточный слой	16 ядер 3 × 3	64 ядра 3 × 3	64 ядра 3 × 3	32 ядра 2 × 2
Dropout		+		
Maxpooling	Ядро 2 × 2, шаг 2	Ядро 2 × 2, шаг 2	Ядро 2 × 2, шаг 2	Ядро 2 × 2, шаг 2
Dropout	+			
Сверточный слой	32 ядер 3 × 3	64 ядра 3 × 3	64 ядра 3 × 3	
Maxpooling	Ядро 2 × 2, шаг 2	Ядро 2 × 2, шаг 2	Ядро 2 × 2, шаг 2	
Dropout	+			
Сверточный слой		128 ядер 3 × 3		
Dropout		+		
Maxpooling		Ядро 2 × 2, шаг 2		
Сверточный слой		256 ядер 3 × 3		
Dropout		+		
Maxpooling		Ядро 2 × 2, шаг 2		
Flatten	+	+	+	+
Полносвязный слой	512 нейронов	128 нейронов	128 нейронов	1024 нейрона
Dropout	+	+	+	+
Полносвязный слой				128 нейронов
Dropout				+
Полносвязный слой	9 нейронов	1 нейрон	1 нейрон	10 нейронов
Количество выходных классов	9 (Apple, Avocado, Banana, Cherry, Cocos, Kiwi, Lemon, Mango, Orange)	2 (Normal, Pneumonia)	2 (Parasitized, Uninfected)	10 (T-shirt/top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, Ankle boot)
Количество эпох обучения	10	5	5	7
Точность по обучающей выборке, %	95,15	98,14	94,45	92,26
Точность по тестовой выборке, %	94,60	78,69	93,61	91,96

Примечание: составлено авторами.

```
model=Sequential()  
model.add(Conv2D(filters=8,kernel_size=(3,3),padding="Same",activation="relu",input_shape=(100,100,3)))  
model.add(MaxPool2D(pool_size=(2,2)))  
model.add(Dropout(0.35))  
  
model.add(Conv2D(filters=16,kernel_size=(3,3),padding="Same",activation="relu"))  
model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))  
model.add(Dropout(0.35))  
  
model.add(Conv2D(filters=32,kernel_size=(3,3),padding="Same",activation="relu"))  
model.add(MaxPool2D(pool_size=(2,2),strides=(2,2)))  
model.add(Dropout(0.35))  
model.add(Flatten())  
model.add(Dense(512,activation="relu"))  
model.add(Dropout(0.5))  
model.add(Dense(9,activation="softmax"))
```

Рис. 4. Задание структуры нейронной сети

Примечание: снимок экрана [10].

Conv2D, MaxPool2D и Dropout – это, соответственно, сверточный, maxpooling- и dropout-слои. Dense – полносвязный нейронный слой. Flatten – слой, преобразующий двумерную матрицу в вектор. В скобках задаются различные параметры для каждого из слоев: filters – количество ядер в сверточном слое, kernel_size и pool_size – размер ядер в сверточном и maxpooling-слоях соответственно. Для установления шага сдвига ядер используется параметр strides, где в скобках через запятую задается значение шага по горизонтали (первое значение) и вертикали (второе значение). На рис. 4 в сверточных слоях не указано значение шага, по умолчанию оно устанавливается в 1 по горизонтали и вертикали. В параметре input_shape задан размер исходного изображения и количество цветовых каналов (на рисунке размерность изображения 100 на 100 пикселей, 3 цветовых канала RGB). Учет цветовой многоканальности происходит за счет установки ядер в каждый цветовой канал. Первый параметр полносвязного слоя – это количество нейронов, второй – вид функции активации.

Все структуры ИНС поддаются сокращению. В табл. 2 красным цветом выделены ячейки, в которых были проведены дальнейшие упрощения структур ИНС.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Для представленных выше структур были выполнены следующие основные шаги:

- обучение исходной структуры нейронной сети;

- упрощение и обучение структур нейронных сетей с одним фиксированным задаваемым сверточным ядром с изменением шага его сдвига и фиксированием точности распознавания;

- выбор размерности ядра и шага;

- перебор комбинаций ядер;

- для цветных изображений проводилась проверка влияния трехканальности на точность результатов обучения.

Для FruitClassification исходная структура нейронной сети на выходе имеет девять классов. Изображения представлены в трехканальном виде по цветовым плоскостям RGB. Точность по обучающей и тестовой выборке исходной структуры $\approx 95\%$.

На рис. 5 представлен пример вывода результатов расчета нейронной сети для датасета Fruits 360 [10].

Для каждой эпохи выводится количество итераций, время в миллисекундах, значения потерь, а также точности по обучающей (acc) и валидационной/тестовой выборкам (val_acc). Точность вычисляется как отношение правильно распознанных ИНС экземпляров исходной выборки к общему их количеству. Результаты вычисления точности, представленные в дальнейших таблицах, переведены в проценты.

```
Epoch 1/10
191/191 [=====] - 10s 52ms/step - loss: 1.9696 - acc: 0.2173 - val_loss: 1.4683 - val_acc: 0.3805
Epoch 2/10
191/191 [=====] - 9s 46ms/step - loss: 1.2587 - acc: 0.4538 - val_loss: 0.8938 - val_acc: 0.4861
Epoch 3/10
191/191 [=====] - 9s 46ms/step - loss: 0.6846 - acc: 0.7132 - val_loss: 0.2742 - val_acc: 0.8666
Epoch 4/10
191/191 [=====] - 9s 46ms/step - loss: 0.4171 - acc: 0.8546 - val_loss: 0.1499 - val_acc: 0.9594
Epoch 5/10
191/191 [=====] - 10s 51ms/step - loss: 0.2394 - acc: 0.9177 - val_loss: 0.1827 - val_acc: 0.9374
Epoch 6/10
191/191 [=====] - 9s 49ms/step - loss: 0.1967 - acc: 0.9273 - val_loss: 0.0950 - val_acc: 0.9733
Epoch 7/10
191/191 [=====] - 9s 47ms/step - loss: 0.1768 - acc: 0.9378 - val_loss: 0.0885 - val_acc: 0.9559
Epoch 8/10
191/191 [=====] - 9s 46ms/step - loss: 0.1270 - acc: 0.9540 - val_loss: 0.0464 - val_acc: 0.9872
Epoch 9/10
191/191 [=====] - 9s 47ms/step - loss: 0.1257 - acc: 0.9575 - val_loss: 0.0423 - val_acc: 0.9861
Epoch 10/10
191/191 [=====] - 9s 46ms/step - loss: 0.1401 - acc: 0.9555 - val_loss: 0.0655 - val_acc: 0.9791
```

Рис. 5. Задание структуры нейронной сети

Примечание: снимок экрана [10].

Для упрощения переборной процедуры трехканальная цветовая схема была переведена в одноканальную с помощью функции `grayscale` [14], при этом снизилась точность как по обучающей (90,35 %), так и по тестовой выборке (85,96 %). Такое изменение входных данных также позволило сократить время обучения с 9,5 сек до 5,25 сек на одну эпоху.

Далее был усечен первый сверточный слой с восемью ядер до одного, убраны `dropout`-слои, удален второй `maxpooling`-слой. Точность такой упрощенной структуры составила по обучающей выборке = 82,56 %, по тестовой = 80,69 %.

Была проведена переборная процедура задаваемых ядер с размерностями от 5×5 до 9×9 , которая состояла в том, что в одно-ядерный первый сверточный слой упрощенной архитектуры нейронной сети устанавливались фиксированные ядра одной из заранее заданных структур (вертикальное, горизонтальное, диагональное или кольцевое) и происходило обучение сети с изменением шага сдвига этого сверточного ядра по изображению (табл. 3). В табл. 3 представлены точности нейронной сети, получаемые с применением ядер различного размера и типа.

Таблица 3

Точность распознавания для задаваемых ядер (CNNFruitClassification), в %

Ядро \ Шаг	1	2	3	4	5	6	7	8	9
5 × 5									
Вертикальное (v)	76,65	61,47	46,70	39,43	42,53				
Горизонтальное (h)	81,55	67,60	55,22	47,17	46,12				
Правая диагональ (rd)	81,03	64,48	53,33	42,56	46,96				
Левая диагональ (ld)	86,89	77,03	65,09	54,79	54,47				
Кольцо (r)	12,46	12,11	11,65	11,79	12,00				
6 × 6									
Вертикальное (v)	67,40	59,49	53,15	52,34	41,71	40,28			
Горизонтальное (h)	64,83	60,36	58,84	52,66	40,83	41,45			
Правая диагональ (rd)	63,49	58,61	53,59	50,41	39,38	41,74			
Левая диагональ (ld)	63,40	63,86	61,73	56,60	46,15	45,30			
Кольцо (r)	11,44	11,70	11,79	11,70	11,65	11,73			

Окончание табл. 3

Ядро \ Шаг	1	2	3	4	5	6	7	8	9
7 × 7									
Вертикальное (v)	59,57	55,63	52,66	45,10	35,90	32,28	29,48		
Горизонтальное (h)	76,81	67,63	62,07	61,82	55,31	38,40	39,31		
Правая диагональ (rd)	80,37	70,60	67,04	62,83	54,89	45,29	43,35		
Левая диагональ (ld)	79,19	72,13	78,30	67,59	60,90	53,49	50,70		
Кольцо (r)	11,16	11,89	11,98	12,14	11,74	10,84	12,24		
8 × 8									
Вертикальное (v)	76,86	66,15	73,28	65,23	60,86	57,84	58,22	50,04	
Горизонтальное (h)	74,54	64,75	72,47	67,43	62,65	60,09	57,59	51,33	
Правая диагональ (rd)	69,47	70,76	55,83	54,04	46,68	48,74	44,19	40,16	
Левая диагональ (ld)	69,10	73,23	63,40	67,22	62,98	52,79	49,52	45,86	
Кольцо (r)	11,23	11,24	11,20	11,93	10,90	10,80	11,08	11,61	
9 × 9									
Вертикальное (v)	68,45	60,88	52,38	46,11	44,69	40,91	41,37	34,29	30,73
Горизонтальное (h)	67,92	66,42	58,59	55,76	49,12	45,59	42,35	40,14	31,26
Правая диагональ (rd)	63,33	68,58	52,66	53,89	49,41	48,80	41,92	40,94	33,89
Левая диагональ (ld)	78,62	69,88	64,74	56,09	58,71	58,12	53,39	50,56	41,66
Кольцо (r)	11,71	11,42	10,75	11,12	11,63	12,15	11,41	10,62	12,35

Примечание: составлено авторами.

Из табл. 3 заметно уменьшение точности с ростом размера ядер. Снижение точности наблюдается и при увеличении шага сдвига сверточного ядра, при этом резкое падение прослеживается на шаге, примерно равном половине размерности исходного ядра.

Далее были составлены комбинации леводиагональной матрицы 5×5 с тремя остальными. Комбинации ядер и получаемый прирост точности сведены в табл. 4.

Таблица 4

Точность для комбинаций ядер (CNNFruitClassification)

Комбинация	Точность по обучающей выборке, %
ld + rd	85,84
ld + h	89,36
ld + v	82,49
ld + rd + h	85,33
ld + rd + v	89,14
ld + h + v	87,53
ld + rd + h + v	91,80

Примечание: составлено авторами.

Наибольшая точность достигается в комбинации всех четырех матриц (точность по обучающей выборке = 91,80 %).

Для проверки влияния количества цветочных каналов на результаты были проведены дополнительные расчеты. Одно сверточное ядро в трех цветовых каналах фактически представляет собой три ядра. В табл. 5 представлены результаты для трех случаев:

1) изображения преобразованы в одноканальный вид (операция grayscale [14]), а в единственное ядро первого сверточно-

го слоя устанавливается леводиагональное ядро;

2) изображения трехканальные, в три цветовых канала заданы одинаковые ядра – леводиагональные;

3) изображения трехканальные, в три цветовых канала заданы три разных ядра – леводиагональное, праводиагональное, вертикальное.

Преобразование цветного изображения в одноканальный вид осуществляется с помощью операции grayscale, по формуле:

$$Y = 0,299 \times R + 0,587 \times G + 0,114 \times B, \quad (1)$$

где R , G и B – соответствующие компоненты красного, зеленого и синего цвета.

Во всех случаях размер ядра равен 5×5 с шагом 1.

Оказалось (табл. 5), что для данного набора изображений цвет оказывает серьезное влияние на результаты обучения, особенно на тестовой выборке.

Датасет Malaria также имеет трехканальные (RGB) цветные изображения (рис. 6).

Таблица 5

Проверка влияния трехканальности на точность распознавания (CNNFruitClassification)

Каналы/точность	Точность по обучающей выборке, %	Точность по тестовой выборке, %
ld – 1 канал	86,60	76,75
ld, ld, ld – 3 канала	65,93	57,72
ld, v, rd – 3 канала	88,99	85,26

Примечание: составлено авторами.

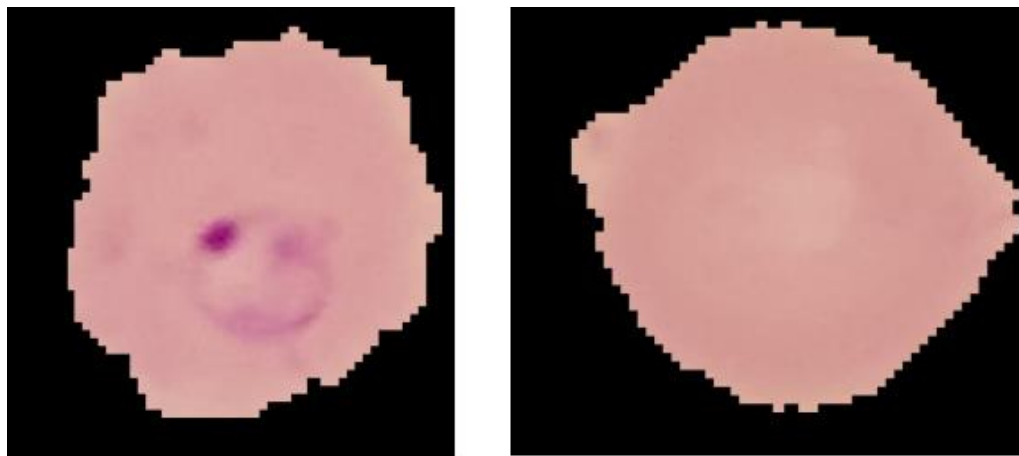


Рис. 6. Примеры изображений из MalariaDataset (слева – зараженная клетка, справа – здоровая)

Примечание: изображения сгенерированы из датасета при помощи исходной структуры НС.

В исходной структуре ИНС DetectingMalaria были проведены следующие сокращения: уменьшено количество ядер во всех сверточных слоях (в первом – с 32 ядер до одного, во втором и третьем слое – с 64 ядер до 16 в каждом из слоев), убраны dropout-слои [15]. Исходная точность такой сокра-

щенной структуры составила: по обучающей выборке = 94,13 %, по тестовой = 93,97 %. Также проведена аналогичная предыдущему примеру переборная процедура задаваемых ядер, результаты по получаемым точностям сведены в табл. 6.

Таблица 6

Точность распознавания для задаваемых ядер (MalariaDataset), в %

Ядро \ Шаг	1	2	3	4	5	6	7	8	9
5×5									
Вертикальное (v)	92,67	90,29	84,10	84,22	80,17				
Горизонтальное (h)	92,81	90,08	85,54	84,16	79,55				
Левая диагональ (ld)	93,14	91,16	85,88	84,83	80,12				
Правая диагональ (rd)	92,53	90,45	86,99	85,09	79,26				
Кольцо (r)	55,56	53,64	54,74	52,91	52,47				

Окончание табл. 6

Ядро \ Шаг	1	2	3	4	5	6	7	8	9
6 × 6									
Вертикальное (v)	92,60	90,86	89,75	88,93	84,07	81,76			
Горизонтальное (h)	92,59	90,74	89,23	87,23	83,86	81,24			
Левая диагональ (ld)	92,61	90,96	86,79	83,64	80,91	81,95			
Правая диагональ (rd)	96,20	91,16	86,84	84,22	80,94	82,69			
Кольцо (r)	55,17	53,86	54,41	53,58	51,34	51,24			
7 × 7									
Вертикальное (v)	92,56	89,77	85,62	83,77	80,31	82,52	84,70		
Горизонтальное (h)	92,25	92,40	91,01	87,87	85,43	83,71	84,99		
Левая диагональ (ld)	92,64	92,50	91,38	89,85	86,63	84,53	84,70		
Правая диагональ (rd)	92,66	92,22	91,67	90,76	87,35	86,01	85,93		
Кольцо (r)	51,43	52,40	51,40	50,38	50,18	50,47	50,62		
8 × 8									
Вертикальное (v)	89,84	91,01	89,69	89,01	87,29	82,96	87,71	83,97	
Горизонтальное (h)	91,62	90,84	89,70	89,92	87,75	85,30	88,09	84,55	
Левая диагональ (ld)	92,46	92,29	89,93	89,46	86,31	83,50	82,88	82,36	
Правая диагональ (rd)	92,20	92,18	90,62	89,50	86,39	84,83	84,57	83,69	
Кольцо (r)	49,42	50,48	50,46	50,44	49,95	50,22	49,92	49,10	
9 × 9									
Вертикальное (v)	90,96	91,52	88,35	87,08	85,81	84,28	84,68	83,34	82,88
Горизонтальное (h)	91,17	91,62	90,36	86,89	85,68	84,41	85,00	82,87	83,68
Левая диагональ (ld)	91,86	90,97	88,71	86,87	85,57	83,19	84,44	83,05	82,36
Правая диагональ (rd)	92,03	91,33	89,94	88,38	86,60	84,88	82,11	83,36	81,44
Кольцо (r)	50,36	49,44	50,19	50,15	49,69	49,56	49,71	49,67	49,83

Примечание: составлено авторами.

Как и в предыдущем примере, с увеличением размера ядра и шага его сдвига наблюдается некоторое снижение точности.

По табл. 6 было выбрано леводиAGONАЛЬНОЕ ядро 7×7 с шагом 3 (в табл. 6 красным

цветом выделена соответствующая ячейка). К нему подбирались комбинации оставшихся ядер (табл. 7). В результате было получено, что комбинация из четырех разных ядер дает наибольший прирост точности.

Таблица 7

Точность для комбинаций ядер (MalariaDataset)

Комбинация	Точность по обучающей выборке, %
ld + rd	92,83
ld + h	92,79
ld + v	93,02
ld + rd + h	93,72
ld + rd + v	93,81
ld + h + v	93,77
ld + rd + h + v	93,86

Примечание: составлено авторами.

При проверке влияния трехканальности на конечный результат выяснилось, что цвет в данном случае практически не влияет на исходный результат (табл. 8).

Для датасета Pneumonia в результате упрощения структуры НС в первом сверточном слое вместо 32 ядер 3×3 было оставлено одно ядро. В последующих сверточных

слоях количество ядер было уменьшено в 2 раза. Также были удалены dropout-слои. Точность такой упрощенной структуры составила 99,79 % по обучающей выборке и 76,28 % по тестовой выборке. Результаты точности в процессе перебора задаваемых ядер сведены в табл. 9.

Таблица 8

Проверка влияния трехканальности на точность распознавания (MalariaDataset)

Каналы/точность	Точность по обучающей выборке, %	Точность по тестовой выборке, %
ld – 1 канал	93,01	92,62
ld, ld, ld – 3 канала	92,33	91,82
ld, v, rd – 3 канала	93,16	92,86

Примечание: составлено авторами.

Таблица 9

Точность распознавания для задаваемых ядер (PneumoniaDetection), в %

Ядро \ Шаг	1	2	3	4	5	6	7	8	9
5 × 5									
Вертикальное (v)	98,98	99,05	98,79	98,77	98,14				
Горизонтальное (h)	98,91	99,21	98,66	98,36	97,98				
Левая диагональ (ld)	99,13	98,76	98,99	98,61	98,13				
Правая диагональ (rd)	99,23	99,44	99,14	98,9	98,35				
Кольцо (r)	87,34	83,98	82,83	78,13	73,59				
6 × 6									
Вертикальное (v)	98,51	98,97	98,51	98,27	97,97	97,84			
Горизонтальное (h)	98,85	98,58	98,84	98,27	98,11	97,46			
Левая диагональ (ld)	99,21	99,09	98,85	98,73	98,18	98,10			
Правая диагональ (rd)	98,74	99,07	99,07	98,85	98,6	97,95			
Кольцо (r)	83,42	78,62	76,06	75,12	74,92	73,42			
7 × 7									
Вертикальное (v)	98,69	98,96	98,79	98,18	98,05	97,73	97,26		
Горизонтальное (h)	99,11	98,68	98,72	98,49	98,13	98,11	97,36		
Левая диагональ (ld)	98,88	99,08	99,24	98,45	98,30	98,02	97,24		
Правая диагональ (rd)	99,14	98,99	98,72	98,85	98,29	98,32	97,20		
Кольцо (r)	81,09	78,32	74,20	74,38	73,28	73,19	73,00		
8 × 8									
Вертикальное (v)	98,53	98,69	98,60	98,41	98,13	97,54	97,22	96,95	
Горизонтальное (h)	98,38	98,77	98,81	98,63	97,94	97,55	98,36	96,49	
Левая диагональ (ld)	98,50	98,91	98,75	98,35	98,51	98,17	97,63	97,04	
Правая диагональ (rd)	98,66	99,02	99,18	98,86	98,38	97,93	97,13	96,73	
Кольцо (r)	80,59	77,67	77,48	74,21	75,44	74,65	73,59	74,32	
9 × 9									
Вертикальное (v)	98,38	99,02	98,56	98,43	98,25	97,94	97,27	96,61	95,77
Горизонтальное (h)	98,80	98,71	98,83	98,25	98,22	98,09	97,53	96,49	96,37
Левая диагональ (ld)	98,84	98,97	98,74	98,76	98,01	97,63	97,80	96,40	95,70
Правая диагональ (rd)	98,72	99,09	99,12	98,66	98,41	98,16	97,52	96,68	94,59
Кольцо (r)	79,77	77,64	74,89	74,16	74,00	75,31	73,27	74,80	75,09

Примечание: составлено авторами.

Поскольку точность результатов во всех случаях при использовании линейно заданных ядер $> 98\%$, что, вероятнее всего, связано с большой размерностью входных изображений, были проведены дополнительные расчеты для нахождения размерности ядра, на которой точность начинает снижаться в большем диапазоне с увеличением шага.

Для дальнейших расчетов было выбрано вертикальное ядро (табл. 10).

Расчет комбинаций ядер происходил на размерности 11×11 с шагом 5 (табл. 11). В табл. 10 красным цветом выделена соответствующая точность для вертикального ядра по обучающей выборке.

Таблица 10

Точность для «вертикального» ядра (PneumoniaDetection), в %

Шаг Размер	1	2	3	4	5	6	7	8	9	10	11	12
10×10	98,1	98,6	98,3	98,3	98,0	97,7	97,3	96,4	96,2	95,2		
11×11	98,0	98,5	98,0	98,3	98,4	98,0	97,0	97,2	96,4	95,1	94,5	
12×12	98,3	98,1	98,2	97,8	97,5	98,0	97,2	97,0	96,2	95,4	94,7	93,0

Примечание: составлено авторами.

Таблица 11

Точность для комбинаций ядер (Pneumonia)

Комбинация	Точность по обучающей выборке, %
v + h	98,77
v + ld	98,38
v + rd	98,76
v + h + ld	98,50
v + h + rd	98,81
v + rd + ld	98,87
v + ld + rd + h	98,63

Примечание: составлено авторами.

В данном случае комбинация из трех ядер (вертикального и двух диагональных) показала наилучшие результаты по точности распознавания.

На датасете Pneumonia достигается высокая точность на обучающей выборке и значительно меньшая на тестовой выборке (< 80 %). Было выдвинуто предположение, что кольцевой тип ядер, возможно, увеличит прирост точности на тестовой выборке в комбинации с некоторым линейным ядром. Взято вертикальное ядро и кольцевое размерности 7×7 с шагом 6 (табл. 9, выделено красным цветом) с целью сравнения результатов улучшений.

В табл. 12 сведены результаты определения необходимости использования кольцевого ядра в комбинации с линейным. Результаты представлены для исходной максимально

упрощенной структуры, состоящей из одного сверточного слоя, содержащего два ядра (кольцевое и вертикальное), и одного полносвязного. По данным из табл. 12 можно сделать вывод, что кольцевой тип сверточных ядер не дает прироста точности в комбинации с линейным.

В табл. 13 представлены результаты при дообучении матриц первого сверточного слоя при различном количестве эпох обучения. Предполагается, что большее количество эпох изменит заданные в сверточном слое ядра в большем диапазоне. Из рис. 7 видно, что наиболее сильно изменилось кольцевое ядро. Шаг изменения вертикального ядра практически одинаковый при разном количестве эпох. При этом как на кольцевом, так и на вертикальном ядре сдвиг происходит в положительную область.

Таблица 12

Влияние кольцевого ядра на точность распознавания

Тип ядра	Точность по обучающей выборке, %	Точность по тестовой выборке, %
Исходная структура ИНС		
Вертикальное	98,04	73,24
Кольцевое	74,79	62,50
Вертикальное + кольцевое	98,35	84,29
1 сверточный слой + 1 полносвязный слой		
Вертикальное	97,41	78,53
Кольцевое	75,69	62,66
Вертикальное + кольцевое	97,34	78,53

Примечание: составлено авторами.

Точности в таблицах 9 и 12 расходятся по вертикальному ядру на 0,31 % и по кольцевому на 1,6 % в связи с повторным обуче-

нием и отсутствием воспроизводимости результатов в моделях глубокого обучения.

Таблица 13

Изменения точности после дообучения

Количество эпох	Точность по обучающей выборке, %	Точность по тестовой выборке, %
Исходная структура ИНС		
5	99,71	76,76
10	99,58	76,76
20	99,92	79,81
25	99,96	79,49
1 сверточный слой + 1 полносвязный слой		
5	97,41	78,53
10	99,02	79,33
20	99,46	74,84
25	100	78,85

Примечание: составлено авторами.

-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2
-2	-1	-1	0	1	1	2

-1,95	-0,94	-0,93	0,06	1,06	1,06	2,07
-1,94	-0,94	-0,93	0,07	1,06	1,06	2,07
-1,94	-0,93	-0,93	0,07	1,06	1,06	2,06
-1,94	-0,93	-0,93	0,07	1,06	1,06	2,06
-1,94	-0,94	-0,93	0,06	1,05	1,05	2,06
-1,94	-0,94	-0,94	0,06	1,05	1,06	2,06
-1,95	-0,94	-0,94	0,06	1,05	1,06	2,07

-1,95	-0,91	-0,90	0,09	1,09	1,09	2,11
-1,95	-0,91	-0,90	0,11	1,10	1,08	2,09
-1,93	-0,90	-0,90	0,10	1,08	1,08	2,09
-1,94	-0,91	-0,91	0,08	1,06	1,07	2,10
-1,95	-0,93	-0,93	0,06	1,04	1,05	2,08
-1,97	-0,95	-0,94	0,05	1,04	1,05	2,06
-1,94	-0,92	-0,91	0,07	1,08	1,07	2,10

-2	-2	-2	-1	-2	-2	-2
-2	-1	0	0	0	-1	-2
-2	0	1	1	1	0	-2
-1	0	1	2	1	0	-1
-2	0	1	1	1	0	-2
-2	-1	0	0	0	-1	-2
-2	-2	-2	-1	-2	-2	-2

-1,81	-1,82	-1,84	-0,80	-1,88	-1,88	-1,94
-1,81	-0,84	0,14	0,17	0,16	-0,92	-1,95
-1,81	0,14	1,23	1,19	1,14	0,15	-1,96
-0,85	0,12	1,14	2,19	1,16	0,09	-0,95
-1,87	0,10	1,13	1,13	1,14	0,09	-1,99
-1,90	-0,93	0,05	0,11	0,09	-0,98	-1,94
-1,81	-1,89	-1,89	-0,88	-1,91	-1,89	-1,94

-1,36	-1,36	-1,38	-0,38	-1,61	-1,58	-1,12
-1,36	-0,17	0,56	0,66	0,62	-0,66	-1,17
-1,41	0,78	1,71	1,73	1,50	0,50	-1,18
-0,50	0,58	1,46	2,71	1,51	0,39	-0,32
-1,62	0,57	1,38	1,32	1,33	0,24	-1,30
-1,75	-0,51	0,17	0,25	0,35	-0,88	-1,47
-1,31	-1,81	-1,59	-0,77	-1,67	-1,68	-1,22

Исходные ядра

После дообучения
(5 эпох)

После дообучения
(25 эпох)

Рис. 7. Дообучение сверточных ядер при разном количестве эпох
(на примере структуры с одним сверточным + одним полносвязным слоем)

Примечание: составлено авторами.

Датасет FashionMNIST является вариацией широко используемого набора черно-белых изображений рукописных цифр. В данном случае вместо цифр представлены различные типы одежды. Структура сети FashionMNIST была подвергнута следующему сокращению:

- 1 ядро в первом сверточном слое вместо 16;
- удален второй сверточный слой;

- удалены второй maxpooling-слой и dropout-слой;

- два полносвязных слоя с общим количеством нейронов 1024 + 128 сведены в один с количеством нейронов 512.

В табл. 14 представлены результаты для задаваемых ядер.

На основании табл. 14 комбинации ядер были рассмотрены на размерности 5×5 с шагом 2 (табл. 15).

Таблица 14

Точность распознавания для задаваемых ядер (FashionMNIST), в %

Шаг Ядро	1	2	3	4	5	6	7	8	9
5 × 5									
Вертикальное (v)	98,85	97,76	93,68	88,57	86,13				
Горизонтальное (h)	99,27	98,58	95,98	91,18	85,42				
Правая диагональ (rd)	99,24	96,48	91,08	83,71	76,92				
Левая диагональ (ld)	99,20	97,96	94,39	85,73	81,04				
Кольцо (r)	80,64	59,90	44,87	30,83	29,53				
6 × 6									
Вертикальное (v)	98,08	96,86	95,42	91,33	89,69	83,72			
Горизонтальное (h)	98,37	96,31	94,75	92,47	89,35	82,18			
Правая диагональ (rd)	98,98	97,22	94,11	88,84	85,36	78,86			
Левая диагональ (ld)	98,82	97,73	94,59	88,35	83,64	71,35			
Кольцо (r)	58,97	40,21	29,37	22,79	19,11	16,67			
7 × 7									
Вертикальное (v)	98,40	96,39	94,06	91,03	85,73	81,95	80,10		
Горизонтальное (h)	98,58	97,23	94,68	92,16	87,72	81,46	77,39		
Правая диагональ (rd)	98,94	96,92	95,17	90,38	85,69	79,64	79,10		
Левая диагональ (ld)	99,06	96,00	94,22	88,78	84,90	73,81	77,61		
Кольцо (r)	31,24	19,16	16,82	14,59	12,38	11,16	15,55		
8 × 8									
Вертикальное (v)	95,45	95,55	91,60	91,66	87,20	86,85	78,90	78,93	
Горизонтальное (h)	96,89	94,98	92,73	90,33	89,18	84,23	76,74	76,53	
Правая диагональ (rd)	98,21	95,71	92,59	91,50	86,29	81,03	72,46	70,66	
Левая диагональ (ld)	98,04	96,29	91,64	90,29	84,92	76,65	55,30	57,99	
Кольцо (r)	17,20	13,72	12,02	11,73	11,06	10,59	10,14	10,24	
9 × 9									
Вертикальное (v)	95,89	93,25	91,80	88,61	83,50	82,58	76,65	73,48	70,46
Горизонтальное (h)	97,91	96,06	93,69	88,07	82,10	81,79	72,79	73,50	70,74
Правая диагональ (rd)	96,40	94,41	92,03	86,94	81,19	82,60	73,76	70,68	69,86
Левая диагональ (ld)	96,35	93,56	92,27	81,11	70,33	78,08	59,08	59,73	64,07
Кольцо (r)	11,48	10,75	10,41	10,18	10,10	10,12	10,03	10,04	10,06

Примечание: составлено авторами.

Таблица 15

Точность для комбинаций ядер (FashionMNIST)

Комбинация	Точность по обучающей выборке, %
v + h	98,85
v + ld	98,74
v + rd	98,82
h + ld	99,03
h + rd	99,05
ld + rd	98,59
v + h + ld	98,75
v + h + rd	98,80
h + ld + rd	98,73
v + ld + rd	99,01
v + h + ld + rd	98,91

Примечание: составлено авторами.

Комбинация горизонтального и праводиагонального ядра дала наибольшую точность: по обучающей выборке = 99,05 %, по тестовой = 90,11 %.

Из полученных результатов по четырем датасетам можно сделать выводы:

- предположительно, при увеличении размера сверточных ядер первого сверточного

слоя требуется установление соответствующей размерности ядер в последующих слоях нейронной сети для избегания снижения точности при установлении ядер большего размера;

- с увеличением шага сдвига сверточного ядра по изображению наблюдается уменьшение точности нейронной сети, при этом в большинстве случаев на шаге, равном половине размерности ядра, происходит существенное снижение точности выходного результата;

- использование представленных в работе кольцевых типов ядер во всех случаях давало наименьшие результаты точности в сравнении с линейно заданными (вертикальными, горизонтальными, диагональными). Возможно, авторами неудачно сформированы такие виды ядер, и необходима другая методика их генерации. На каждой из структур нейронных сетей показана возможность

использования заранее фиксированных ядер в ограниченном количестве.

ЗАКЛЮЧЕНИЕ

На различных примерах приведено обоснование возможности использования дискретных фиксированных сверточных ядер в виде заранее заданных «шаблонов», которые формируются на основании особенностей входного набора изображений. Применение таких ядер позволит сократить потребление вычислительных ресурсов как при обучении нейронных сетей, так и на этапе рабочего хода.

Таким образом, для получения требуемого результата необходимо подбирать предпочтительные «шаблоны» сверточных ядер, их размер и шаг сдвига по изображению, количество этих ядер и соответствующие комбинации.

Список источников

1. Mahajan P. Fully Connected vs Convolutional Neural Networks. URL: <https://medium.com/swlh/fully-connected-vs-convolutional-neural-networks-813ca7bc6ee5> (дата обращения: 25.07.2022).
2. Lin Z., Memisevic R., Konda K. How Far Can We Go without Convolution: Improving Fully-Connected Networks // arXiv. 2015. URL: <https://arxiv.org/abs/1511.02580> (дата обращения: 25.07.2022).
3. Vajpayee S. License Plate Recognition Using CNN. URL: <https://www.kaggle.com/code/sarthakvajpayee/license-plate-recognition-using-cnn/notebook> (дата обращения: 25.07.2022).
4. Гиниятуллин В. М., Ермолаев Е. В., Хлыбов А. В. Формализация процедуры подбора сверточных ядер // Вест. кибернетики. 2022. № 2. С. 66–74.
5. Хлыбов А. В., Ермолаев Е. В., Зеленев А. С., Кonyaeva Е. Э., Фаррахова К. А., Хузиахметов В. Э., Яковлев А. С. Преобразование весов матриц сверточных нейронных сетей // Теория и практика процессов хим. технологии (Марушкинские чтения) : материалы VI Междунар. науч. конф. Уфа, 2021. С. 269–270.
6. CNN Fruit Classification. Data. URL: <https://www.kaggle.com/etatbak/cnn-fruit-classification/> data (дата обращения: 25.07.2022).
7. Pneumonia Detection Using CNN (92.6 % Accuracy). Data. URL: <https://www.kaggle.com/madz2000/pneumonia-detection-using-cnn-92-6-accuracy/data> (дата обращения: 25.07.2022).
8. Detecting Malaria | Keras | CNN. Data. URL: <https://www.kaggle.com/miracle9to9/detecting-malaria-keras-cnn/data> (дата обращения: 25.07.2022).

References

1. Mahajan P. Fully Connected vs Convolutional Neural Networks. URL: <https://medium.com/swlh/fully-connected-vs-convolutional-neural-networks-813ca7bc6ee5> (accessed: 25.07.2022).
2. Lin Z., Memisevic R., Konda K. How Far Can We Go without Convolution: Improving Fully-Connected Networks // arXiv. 2015. URL: <https://arxiv.org/abs/1511.02580> (accessed: 25.07.2022).
3. Vajpayee S. License Plate Recognition Using CNN. URL: <https://www.kaggle.com/code/sarthakvajpayee/license-plate-recognition-using-cnn/notebook> (accessed: 25.07.2022).
4. Giniyatullin V. M., Ermolaev E. V., Khlybov A. V. Characterizing a Procedure for Searching Convolution Kernels // Proceedings in Cybernetics. 2022. No. 2. P. 66–74. (In Russian).
5. Khlybov A. V., Ermolaev E. V., Zelenov A. S., Konyaeva E. E., Farrakhova K. A., Khuzhiakhmetov V. E., Yakovlev A. S. Preobrazovanie vesov matrits svertochnykh neironnykh setei // Teoriia i praktika protsessov khimicheskoi tekhnologii (Marushkinskie chteniia) : Proceedings of the VI International Scientific Conference. Ufa, 2021. P. 269–270. (In Russian).
6. CNN Fruit Classification. Data. URL: <https://www.kaggle.com/etatbak/cnn-fruit-classification/data> (accessed: 25.07.2022).
7. Pneumonia Detection Using CNN (92.6 % Accuracy). Data. URL: <https://www.kaggle.com/madz2000/pneumonia-detection-using-cnn-92-6-accuracy/data> (accessed: 25.07.2022).

9. Fashion MNIST – Keras CNN. Data. URL: <https://www.kaggle.com/vishnu0399/fashion-mnist-keras-cnn/data> (дата обращения: 25.07.2022).
10. CNN Fruit Classification. Notebook. URL: <https://www.kaggle.com/etatbak/cnn-fruit-classification/notebook> (дата обращения: 25.07.2022).
11. Pneumonia Detection Using CNN (92.6 % Accuracy). Notebook. URL: <https://www.kaggle.com/madz2000/pneumonia-detection-using-cnn-92-6-accuracy/notebook> (дата обращения: 25.07.2022).
12. Detecting Malaria | Keras | CNN. Notebook. URL: <https://www.kaggle.com/miracle9to9/detecting-malaria-keras-cnn/notebook> (дата обращения: 25.07.2022).
13. Fashion MNIST – Keras CNN. Notebook. URL: <https://www.kaggle.com/vishnu0399/fashion-mnist-keras-cnn/notebook> (дата обращения: 25.07.2022).
14. Understand How Color to Gray Scale Works Using OpenCV. URL: <https://www.learnpythonwithrune.org/understand-how-color-to-gray-scale-works-using-opencv/> (дата обращения: 25.07.2022).
15. Tf.keras.layers.Dropout. URL: https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dropout (дата обращения: 25.07.2022).
8. Detecting Malaria | Keras | CNN. Data. URL: <https://www.kaggle.com/miracle9to9/detecting-malaria-keras-cnn/data> (accessed: 25.07.2022).
9. Fashion MNIST – Keras CNN. Data. URL: <https://www.kaggle.com/vishnu0399/fashion-mnist-keras-cnn/data> (accessed: 25.07.2022).
10. CNN Fruit Classification. Notebook. URL: <https://www.kaggle.com/etatbak/cnn-fruit-classification/notebook> (accessed: 25.07.2022).
11. Pneumonia Detection Using CNN (92.6 % Accuracy). Notebook. URL: <https://www.kaggle.com/madz2000/pneumonia-detection-using-cnn-92-6-accuracy/notebook> (accessed: 25.07.2022).
12. Detecting Malaria | Keras | CNN. Notebook. URL: <https://www.kaggle.com/miracle9to9/detecting-malaria-keras-cnn/notebook> (accessed: 25.07.2022).
13. Fashion MNIST – Keras CNN. Notebook. URL: <https://www.kaggle.com/vishnu0399/fashion-mnist-keras-cnn/notebook> (accessed: 25.07.2022).
14. Understand How Color to Gray Scale Works Using OpenCV. URL: <https://www.learnpythonwithrune.org/understand-how-color-to-gray-scale-works-using-opencv/> (accessed: 25.07.2022).
15. Tf.keras.layers.Dropout. URL: https://www.tensorflow.org/api_docs/python/tf/keras/layers/Dropout (accessed: 25.07.2022).

Информация об авторах

В. М. Гиниятуллин – кандидат технических наук, доцент.
А. В. Хлыбов – аспирант.
М. А. Федоров – студент.
Т. А. Асадуллин – студент.
А. С. Крутин – студент.
И. А. Осипов – студент.
Д. М. Зарипов – кандидат физико-математических наук, доцент.

Information about the authors

V. M. Giniyatullin – Candidate of Sciences (Engineering), Associate Professor.
A. V. Khlybov – Postgraduate.
M. A. Fedorov – Student.
T. A. Asadullin – Student.
A. S. Krutin – Student.
I. A. Osipov – Student.
D. M. Zaripov – Candidate of Sciences (Physics and Mathematics), Associate Professor.

Научная статья
УДК 004.925.8
doi: 10.34822/1999-7604-2022-3-99-108

ГЕОМЕТРИЧЕСКИЕ МОДЕЛИ УКЛОНЕНИЯ ЦЕЛИ ОТ ПРЕСЛЕДОВАТЕЛЯ

Александр Анатольевич Дубанов

Бурятский государственный университет им. Доржи Банзарова, Улан-Удэ, Россия
alandubanov@mail.ru, <https://orcid.org/0000-0002-1855-2562>

Аннотация. В данной статье предлагаются к рассмотрению геометрические модели уклонения цели от преследователя, движущегося прямолинейно и равномерно. В геометрических моделях предполагается, что преследователь обладает областью обнаружения. Рассматривались следующие модели области обнаружения: круговая, секторная и угловая. Цель первоначально находится на линии визирования и стремится покинуть место вероятного обнаружения по оптимальной траектории. Рассматриваются вопросы минимального расстояния от преследователя, когда еще будет возможно покинуть область обнаружения. Отметим, что адаптивное поведение не было смоделировано со стороны преследователя. По результатам исследования изготовлены анимированные изображения, на которых можно посмотреть процесс уклонения цели от преследователя.

Ключевые слова: траектория, цель, преследователь, уклонение, область обнаружения, преследование

Для цитирования: Дубанов А. А. Геометрические модели уклонения цели от преследователя // Вестник кибернетики. 2022. № 3 (47). С. 99–108. DOI 10.34822/1999-7604-2022-3-99-108.

Original article

GEOMETRIC MODELS OF TARGET EVASION FROM A PURSUER

Aleksandr A. Dubanov

Buryat State University named after D. Banzarov, Ulan-Ude, Russia
alandubanov@mail.ru, <https://orcid.org/0000-0002-1855-2562>

Abstract. The article considers geometric models of target evasion from a pursuer moving in a straight line and uniformly. In geometric models, it is supposed that a pursuer has a detection area. The study discusses the following detection models: circular, sector, and angular. Initially, the target is moving out of the line of sight along the optimal trajectory to leave the area of probable detection. The issues of the minimum distance from a pursuer allowing the target to escape the detection area are analyzed. It is noted that the adaptive behavior of the pursuer is not simulated. As a result of the study, animated images have been produced to present the process of target evasion from the pursuer.

Keywords: trajectory, target, pursuer, evasion, detection area, pursuit

For citation: Dubanov A. A. Geometric Models of Target Evasion from a Pursuer // Proceedings in Cybernetics. 2022. No. 3 (47). P. 99–108. DOI 10.34822/1999-7604-2022-3-99-108.

ВВЕДЕНИЕ

Вопросы уклонения цели от преследователя были исследованы в следующих работах: в [1] рассмотрены вопросы оптимизации траекторий цели на плоскости и в пространстве; в [2] описаны задачи планирования траектории движения подвижного объекта

на постоянной скорости на плоскости в случае, когда карта рисков-угроз формируется одиночным сенсором; в [3] найден закон управления скоростью подвижного объекта (цели): построена оптимальная траектория движения, проведено сравнение зависимости накопленного сигнала от времени при дру-

гих законах управления; в [4] представлена задача построения оптимальных траекторий подвижного наблюдателя, выполняющего угловые наблюдения за целью, движущейся на плоскости равномерно и прямолинейно; в [5] исследована задача планирования оптимальной траектории уклонения подвижного объекта от системы наблюдателей.

На основе анализа данных работ произведено моделирование уклонения цели от преследователя с круговой, секторной и угловой областью обнаружения.

Преследователь в моделях статьи не предпринимает ответных действий на движение цели и движется равномерно и прямолинейно.

В начале процесса преследования скорость преследователя направлена вдоль линии визирования, что означает направление на цель.

Предполагается, что если преследователь будет совершать ответные действия в отношении цели, то он будет придерживаться метода погони.

По результатам исследования изготовлены анимированные изображения, где показаны моменты уклонения цели от преследователя.

МАТЕРИАЛЫ И МЕТОДЫ

Круговая область обнаружения. Рассмотрим на плоскости преследователя Р и цель Т (рис. 1).

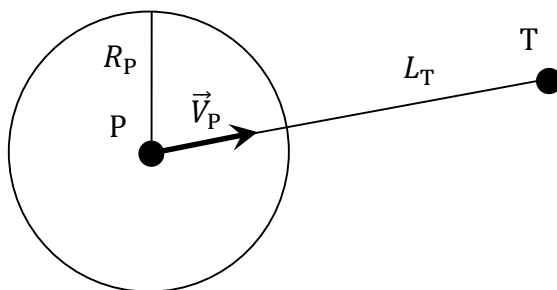


Рис. 1. Модель уклонения цели

Примечание: составлено авторами на основании данных, полученных в исследовании.

Будем считать, что преследователь Р, двигаясь с вектором скорости $\vec{V}_P$, имеет область обнаружения в виде окружности радиуса R_P . А цель Т находится от преследователя Р на расстоянии L_P в момент начала преследования так, что линия (РТ) и вектор $\vec{V}_P$ являются коллинеарными.

Требуется найти траекторию движения цели Т, при которой время движения цели

является минимальным, чтобы избежать области обнаружения.

Для решения этой задачи перейдем в локальную динамическую систему координат с центром в точке нахождения преследователя Р. Ось абсцисс E_1 этой системы расположена вдоль вектора $\vec{V}_P$, а ось ординат E_2 располагается перпендикулярно $\vec{V}_P$ (рис. 2).

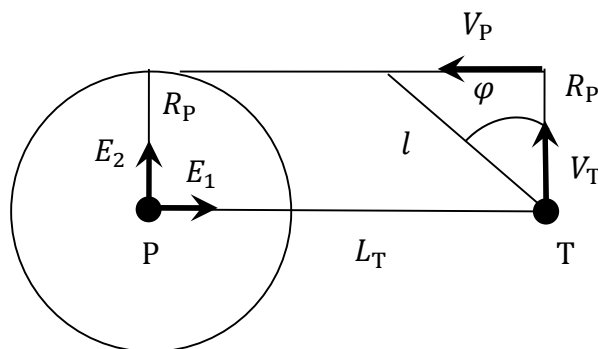


Рис. 2. Локальный динамический базис преследователя

Примечание: составлено авторами на основании данных, полученных в исследовании.

В такой постановке задача похожа на задачу, где надо переплыть реку шириной R_P со скоростью течения V_P (рис. 2). Если скорость цели будет V_T , а ее направление будет перпендикулярно берегам, то минимальное время переправы через реку будет:

$$T_{\min} = \frac{R_P}{V_T}. \quad (1)$$

Итоговая скорость будет направлена под углом φ к направлению скорости цели T (рис. 2):

$$\operatorname{tg}(\varphi) = \frac{V_P}{V_T}. \quad (2)$$

Цель T преодолит расстояние, равное l :

$$l = R_P \cdot \frac{\sqrt{V_T^2 + V_P^2}}{V_T}. \quad (3)$$

На рис. 3 показан случай, когда цель T находится на минимальном расстоянии

$L_{T\min}$ от преследователя P , при котором траектория цели не будет проходить через область обнаружения при условии перпендикулярности векторов скоростей.

Прямая траектории цели (TQ) касается окружности обнаружения в точке K , при этом имеется два конгруэнтных треугольника (TQF) и (PKT) , то есть $[TQ] = [PT]$. Откуда мы заключаем, что если $[TQ]$ вычисляется по формуле (1), то для величины $L_{T\min}$ имеется следующее выражение:

$$L_{T\min} = R_P \cdot \frac{\sqrt{V_T^2 + V_P^2}}{V_T}. \quad (4)$$

Напомним, скорость цели V_T должна быть направлена перпендикулярно линии (PT) , тогда линия (TQ) разделяет области, где траектория цели будет замеченной преследователем или нет.

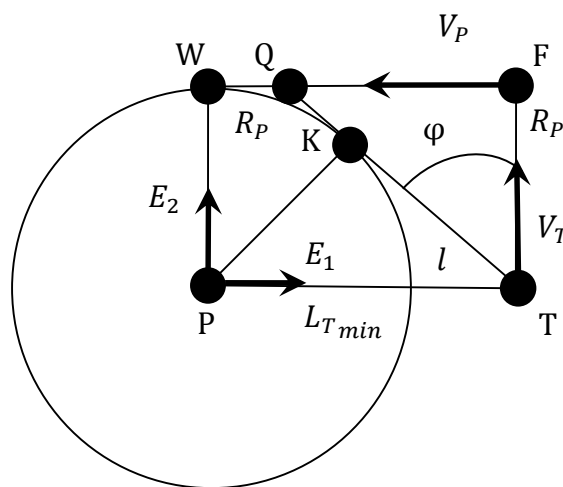


Рис. 3. Минимальное расстояние до преследователя

Примечание: составлено авторами на основании данных, полученных в исследовании.

На рис. 4 показано, как преследователь P с областью обнаружения в виде окружности радиуса R_P движется к цели T .

Цель T расположена на плоскости произвольно, в момент начала преследования она неподвижна. Расстояние между точками P и T выражается формулой (1).

Рис. 4 дополнен ссылкой на анимированное изображение [6], где показано критиче-

ское уклонение с касанием зоны обнаружения.

На рис. 5 показан случай, когда траектория цели проходит через область обнаружения преследователя.

Рис. 5 дополнен ссылкой на анимированное изображение [7], показано прохождение цели через зону обнаружения преследователя.

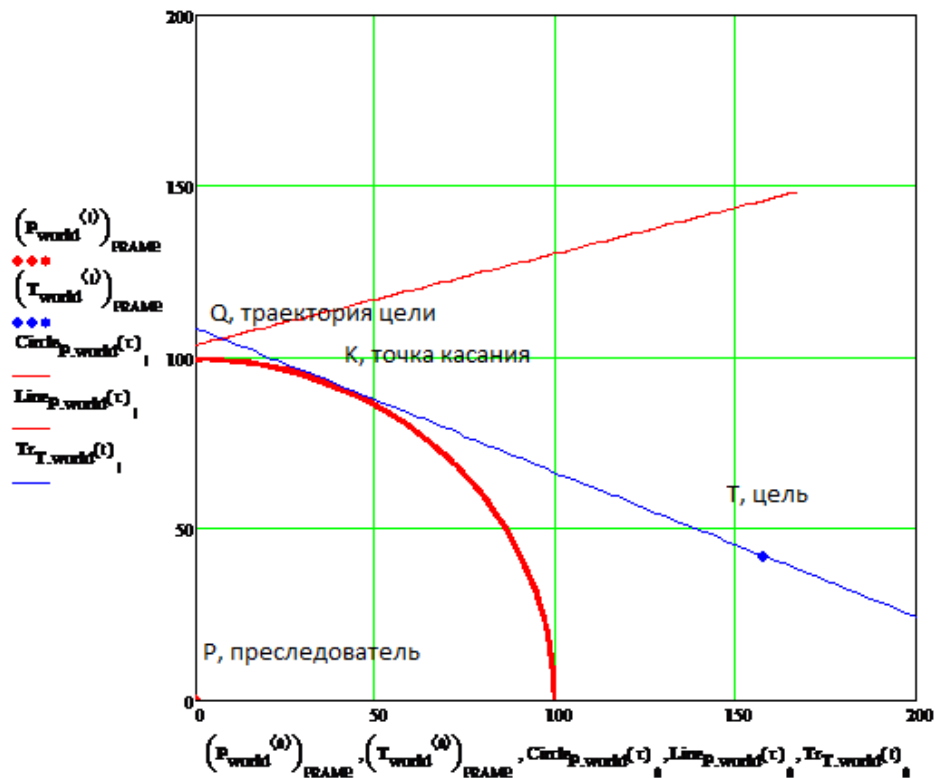


Рис. 4. Результаты моделирования преследования с областью обнаружения в виде окружности
 Примечание: составлено авторами на основании данных, полученных в исследовании.

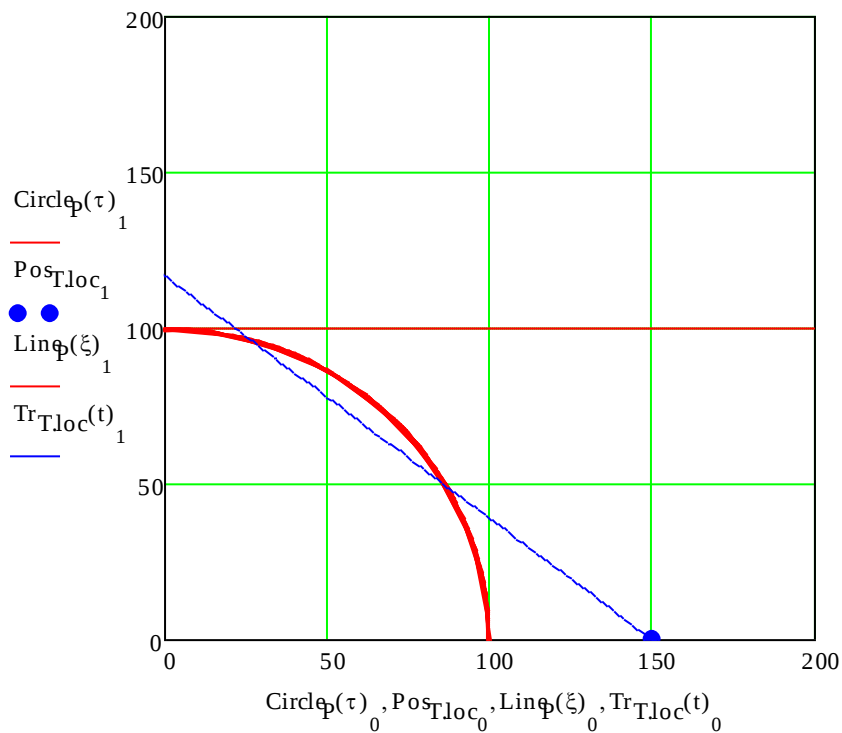


Рис. 5. Цель проходит через область обнаружения преследователя
 Примечание: составлено авторами на основании данных, полученных в исследовании.

Секторная область обнаружения. Рассмотрим задачу, где область обнаружения ограничена не только окружностью, но и углом величиной 2α (рис. 6).

В этом случае цель Т должна пересечь участок плоскости шириной $R_T = R_P \cdot \sin(\alpha)$. Тогда цель Т пройдет расстояние l , равное:

$$l = R_T \cdot \frac{\sqrt{V_T^2 + V_P^2}}{V_T} = R_P \cdot \frac{\sqrt{V_T^2 + V_P^2}}{V_T} \cdot \sin(\alpha). \quad (5)$$

На рис. 7 показано, что угол α сектора обнаружения преследователя Р больше угла φ движения цели Т.

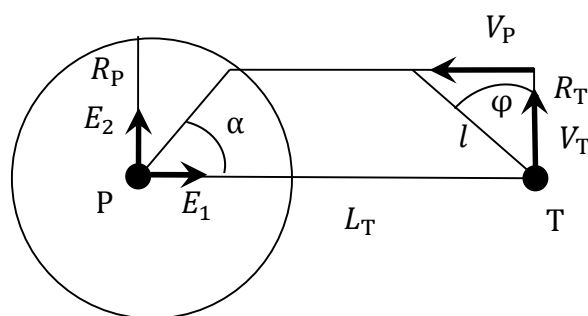


Рис. 6. Локальный динамический базис преследователя с секторной областью обнаружения
Примечание: составлено авторами на основании данных, полученных в исследовании.

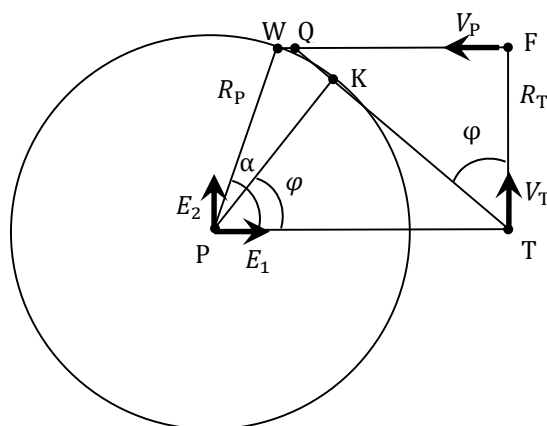


Рис. 7. Угол сектора обнаружения преследователя больше угла движения цели
Примечание: составлено авторами на основании данных, полученных в исследовании.

На этом рисунке отображена ситуация когда траектория (ТQ) цели Т касается сектора обнаружения преследователя Р. Минимальное время прохождения составляет:

$$T_{\min} = \frac{R_T}{V_P}. \quad (6)$$

Цель Т пройдет путь (ТQ) длиной l , равной:

$$l = R_T \cdot \frac{\sqrt{V_T^2 + V_P^2}}{V_T}. \quad (7)$$

Ситуация, когда минимальное расстояние [РТ] между преследователем и целью, при

котором траектория (ТQ) будет касаться сектора обнаружения преследователя (при условии перпендикулярности скоростей), также отображена на рис. 7.

Точка касания К рассчитывается по формуле:

$$K = R_P \cdot \begin{bmatrix} \cos(\varphi) \\ \sin(\varphi) \end{bmatrix} = R_P \cdot \begin{bmatrix} \frac{V_T}{\sqrt{V_T^2 + V_P^2}} \\ \frac{V_P}{\sqrt{V_T^2 + V_P^2}} \end{bmatrix}. \quad (8)$$

На рис. 8 показано как цель Т касается секторной области обнаружения преследователя Р.

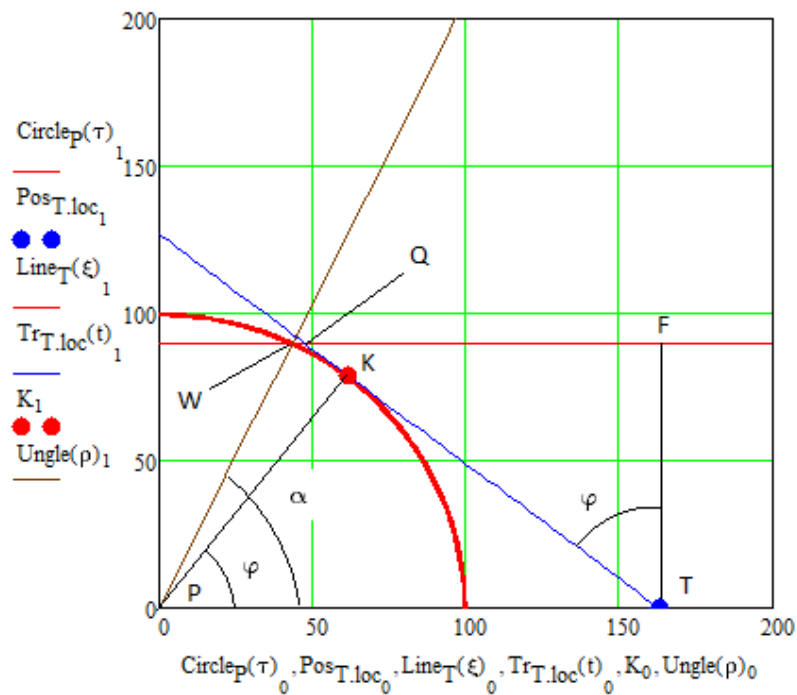


Рис. 8. Секторная область обнаружения

Примечание: составлено авторами на основании данных, полученных в исследовании.

Рис. 8 дополнен анимированным изображением [8], где изображен процесс преследования, когда угол α больше чем угол φ .

Рассмотрим случай, когда угол α сектора обнаружения преследователя меньше угла φ наклона движения цели.

На рис. 9 представлен такой случай, когда параметры угла φ определяются модулями скоростей преследователя и цели.

$$\text{tg}(\varphi) = \frac{V_P}{V_T}. \quad (9)$$

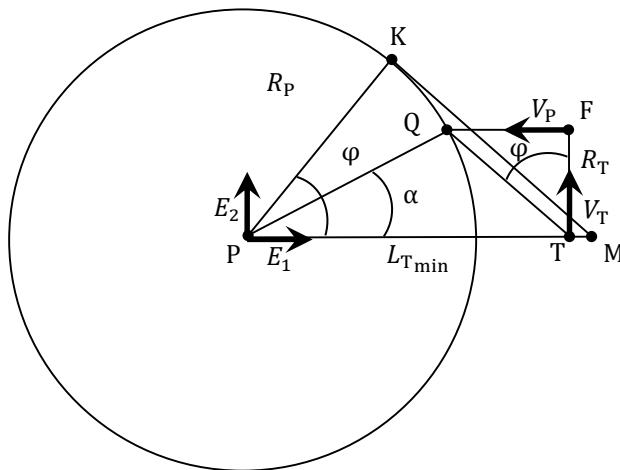


Рис. 9. Секторная область, когда угол обнаружения α меньше угла касания φ

Примечание: составлено авторами на основании данных, полученных в исследовании.

Необходимо построить касательную прямую, чтобы она с осью ординат E_2 составляла угол φ . На рис. 9 это прямая (KM). Точка Q,

ограничивающая сектор обнаружения, имеет координаты:

$$Q = R_P \cdot \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix}. \quad (10)$$

$$L_{T\min} = R_P \cdot \cos(\alpha) + R_P \cdot \sin(\alpha) \cdot \frac{V_P}{V_T}. \quad (13)$$

Ширина полосы, которую надо преодолеть, имеет величину R_T :

$$R_T = R_P \cdot \sin(\alpha). \quad (11)$$

Расстояние $|QF|$ рассчитывается следующим образом:

$$|QF| = R_T \cdot \operatorname{tg}(\varphi) = R_P \cdot \sin(\alpha) \cdot \frac{V_P}{V_T}. \quad (12)$$

Минимальное расстояние $L_{T\min} = |PT|$, при котором цель Т не попадает в зону обнаружения преследователя, при условии перпендикулярности скоростей, будет равным:

$$\operatorname{tg}(\varphi) = \frac{V_P}{V_T}. \quad (14)$$

Отрезок $[QT]$, принадлежащий траектории цели, получается параллельным смещением касательного отрезка $[KM]$ до совмещения с точкой Q.

На рис. 10 представлена компьютерная модель в системе компьютерной математики; показано, что угол сектора обнаружения α больше, чем угол φ . Напомним, величина угла φ определяется соотношением скоростей преследователя и цели:

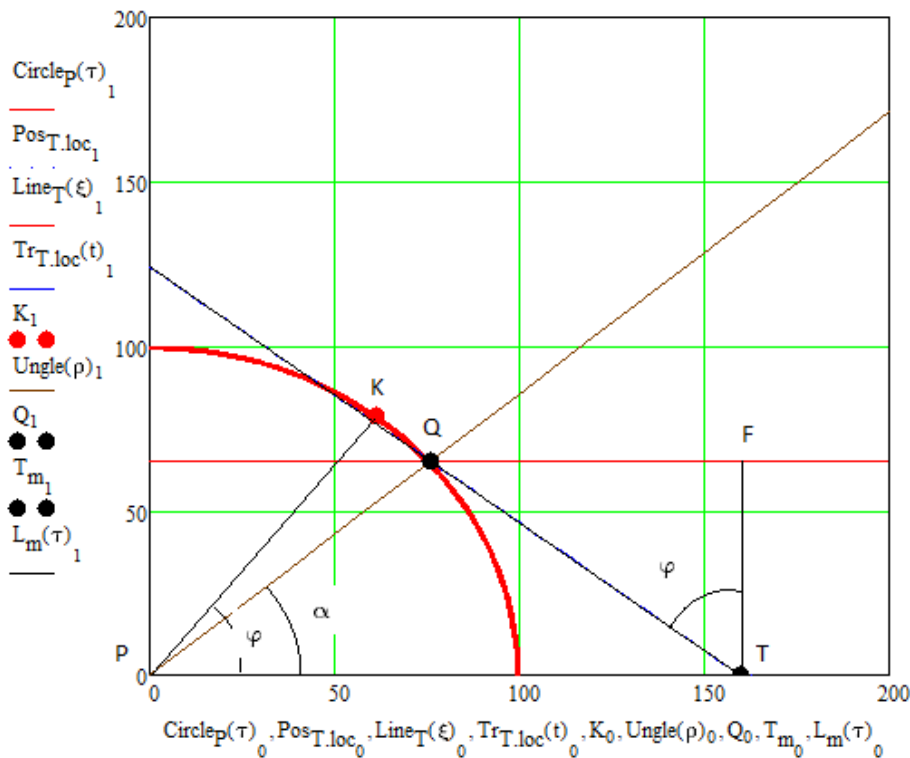


Рис. 10. Модель секторной области обнаружения в системе компьютерной математики

Примечание: составлено авторами на основании данных, полученных в исследовании.

Рис. 10 дополнен анимированным изображением [9], где можно будет посмотреть, как цель Т избегает обнаружения преследователем Р в области обнаружения в виде углового сектора 2α , ограниченного окружностью радиуса R_P . На видео показано, что цель покидает полосу вероятного обнаруже-

ния в последний момент времени, проходя через точку Q.

Угловая область обнаружения. Рассмотрим модель преследования, где область обнаружения преследователя Р представляет собой область в виде угла величиной 2α (рис. 11).

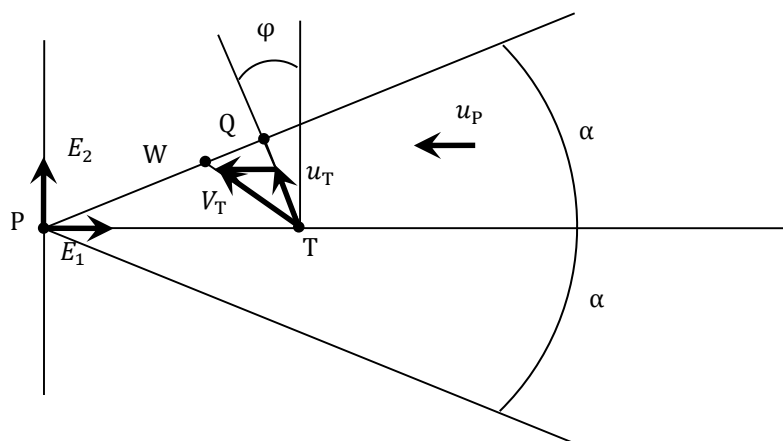


Рис. 11. Угловая область обнаружения у преследователя

Примечание: составлено авторами на основании данных, полученных в исследовании.

Сформируем локальную динамическую систему координат (E_1PE_2) , центр которой расположен в точке нахождения преследователя Р, ось ординат E_1 направлена вдоль вектора скорости, а ось E_2 направлена перпендикулярно скорости движения преследователя (рис. 11).

В системе координат (E_1PE_2) итоговая скорость V_T цели Т складывается из скорости движения среды u_r и скорости движения це-

ли u_T в мировой системе координат: $\vec{V}_T = \vec{u}_P + \vec{u}_T$.

На рис. 12 показано, что скорость u_T цели направлена под углом φ к линии перпендикуляра к линии (РТ). Если u_T это модуль скорости цели, то ее проекция на линию (QT) будет равна $u_T \cdot \cos(\alpha - \varphi)$, где (QT) является перпендикуляром к линии, ограничивающей область обнаружения преследователя (рис. 12).

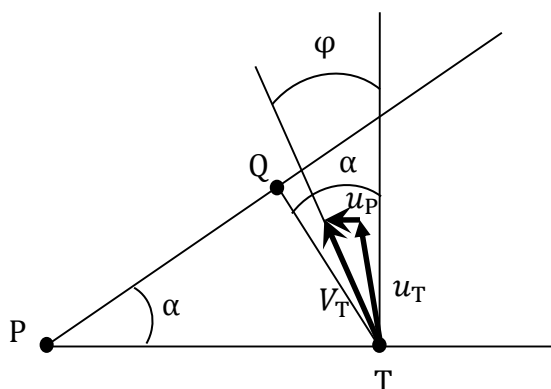


Рис. 12. Разложение вектора скорости цели

Примечание: составлено авторами на основании данных, полученных в исследовании.

Проекция скорости u_T на линию (QT) равна $u_P \cdot \sin(\alpha)$. Тогда проекция на линию (QT) результирующей скорости V_T будет $u_T \cdot \cos(\alpha - \varphi) + u_P \cdot \sin(\alpha)$.

Тогда время t_a , за которое цель Т достигнет линии угла, будет рассчитываться следующим образом:

$$t_a = \frac{|\text{QT}|}{u_T \cdot \cos(\alpha - \varphi) + u_P \cdot \sin(\alpha)}. \quad (15)$$

Время выхода из области обнаружения достигнет своего минимума при соблюдении равенства $\varphi = \alpha$, когда скорость цели u_T направлена вдоль линии (QT).

На рис. 11 изображен именно такой случай. Цель выйдет из области обнаружения в точке W . Результирующая скорость $\vec{V}_T$ направлена вдоль прямой (WT) .

На рис. 13 показан первый кадр из результата работы программы в системе компьютерной математики.

Преследователь P преследует цель T в мировой системе координат. Вектор скорости преследователя направлен на цель.

Задача состоит в построении оптимальной траектории цели, чтобы покинуть область обнаружения как можно быстрее.

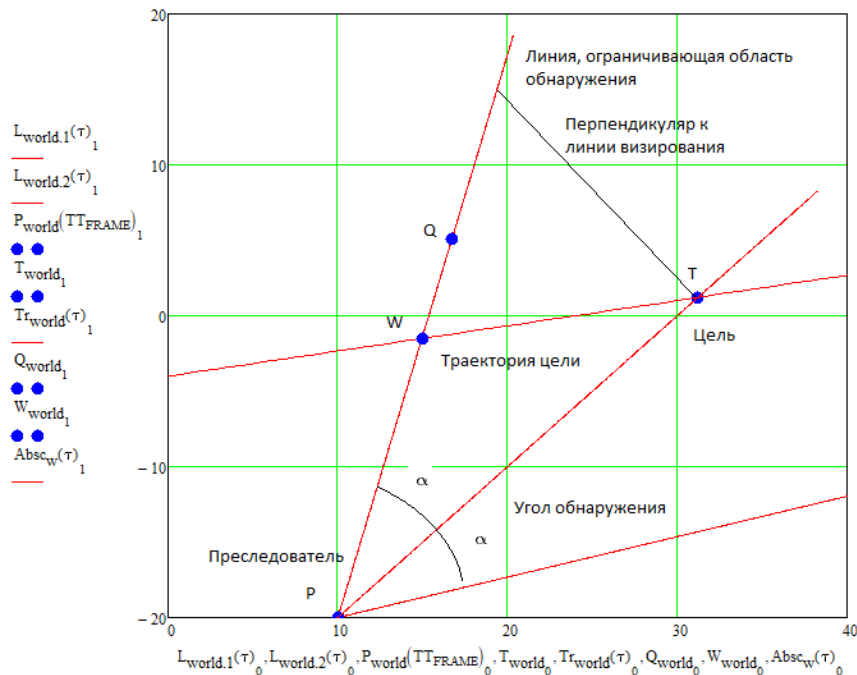


Рис. 13. Модель угловой области обнаружения у преследователя

Примечание: составлено авторами на основании данных, полученных в исследовании.

Рис. 13 дополнен ссылкой на анимированное изображение [10], где можно будет посмотреть на итерационный процесс расчета траектории цели.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

В статье представлено геометрическое моделирование процесса уклонения цели от преследователя, движущегося равномерно и прямолинейно.

Преследователь в моделях имеет круговую, секторную и угловую области обнаружения. Имелось в виду то, что при моделировании адаптивного поведения участников задачи преследования в дальнейшем преследователь будет придерживаться метода погони.

К затруднениям изложенной методики можно отнести то, что цель для маневрирования должна обладать данными о скорости

преследователя и о характеристиках области обнаружения.

Предполагается, что данную методику можно реализовать в системах виртуальной реальности, допустим, при моделировании робототехнических систем.

ЗАКЛЮЧЕНИЕ

Произведено моделирование уклонения цели от преследователя с круговой, секторной и угловой областями обнаружения. Движение участников задачи преследования происходит на плоскости с постоянной скоростью. Цель уклоняется от преследователя по оптимальной траектории.

По результатам исследований изготовлены анимированные изображения, на которых показан процесс уклонения цели от преследователя.

Список источников

1. Абрамянц Т. Г., Галяев А. А., Маслов Е. П., Яхно В. П. Уклонение подвижного объекта от обнаружения в конфликтной среде // Управление большими системами : сб. тр. 2019. Вып. 79. С. 112–184.
2. Галяев А. А., Лысенко П. В., Яхно В. П. Уклонение подвижного объекта от одиночного обнаружителя на заданной скорости // Проблемы управления. 2020. № 1. С. 83–91.
3. Галяев А. А. Задача уклонения от подвижного одиночного наблюдателя на плоскости в конфликтной среде // Автоматика и телемеханика. 2014. № 6. С. 39–48.
4. Андреев К. В., Рубинович Е. Я. Траекторное управление наблюдателем за мобильной целью по угломерной информации // Автоматика и телемеханика. 2016. № 1. С. 134–162.
5. Галяев А. А., Маслов Е. П., Абрамянц Т. Г., Яхно В. П. Уклонение подвижного объекта от обнаружения системой наблюдателей: сенсор – маневренное средство // Автоматика и телемеханика. 2017. № 8. С. 113–126.
6. Видео, результаты моделирования задачи преследования. URL: <https://www.youtube.com/watch?v=C7imOTjEgQQ> (дата обращения: 09.07.2022).
7. Видео, результаты моделирования задачи преследования. URL: <https://www.youtube.com/watch?v=0-qA8Q2fcwE> (дата обращения: 09.07.2022).
8. Видео, результаты моделирования задачи преследования. URL: <https://youtu.be/arCQYrLZGxk> (дата обращения: 09.07.2022).
9. Видео, результаты моделирования задачи преследования. URL: <https://youtu.be/dBGalSwRclg> (дата обращения: 09.07.2022).
10. Видео, результаты моделирования задачи преследования. URL: <https://www.youtube.com/watch?v=lvHvKzEjU4U> (дата обращения: 09.07.2022).

Информация об авторе

А. А. Дубанов – кандидат технических наук, доцент.

References

1. Abramyants T. G., Galyaev A. A., Maslov E. P., Yakhno V. P. A Moving Object Evasion from Detection in the Threat Environment // Large-Scale Systems Control : Collection of works. 2019. Is. 79. P. 112–184. (In Russian).
2. Galyaev A. A., Lysenko P. V., Yakhno V. P. Moving Object Evasion from Single Detector at Given Speed // Problems of Management. 2020. No. 1. P. 83–91. (In Russian).
3. Galyaev A. A. Evasion on Plane from a Single Mobile Observer in the Conflict Environment // Avtomatika i telemekhanika. 2014. No. 6. P. 39–48. (In Russian).
4. Andreev K. V., Rubinovich E. Ya. Moving Observer Trajectory Control by Angular Measurements in Tracking Problem // Avtomatika i telemekhanika. 2016. No. 1. P. 134–162. (In Russian).
5. Galyaev A. A., Maslov E. P., Abramyants T. G., Yakhno V. P. Evasion of a Moving Object from Detection by a System of Observers: Sensor–Maneuvering Search Means // Avtomatika i telemekhanika. 2017. No. 8. P. 113–126. (In Russian).
6. Video, rezultaty modelirovaniia zadachi presledovaniia. URL: <https://www.youtube.com/watch?v=C7imOTjEgQQ> (accessed: 09.07.2022). (In Russian).
7. Video, rezultaty modelirovaniia zadachi presledovaniia. URL: <https://www.youtube.com/watch?v=0-qA8Q2fcwE> (accessed: 09.07.2022). (In Russian).
8. Video, rezultaty modelirovaniia zadachi presledovaniia. URL: <https://youtu.be/arCQYrLZGxk> (accessed: 09.07.2022). (In Russian).
9. Video, rezultaty modelirovaniia zadachi presledovaniia. URL: <https://youtu.be/dBGalSwRclg> (accessed: 09.07.2022). (In Russian).
10. Video, rezultaty modelirovaniia zadachi presledovaniia. URL: <https://www.youtube.com/watch?v=lvHvKzEjU4U> (accessed: 09.07.2022). (In Russian).

Information about the author

A. A. Dubanov – Candidate of Sciences (Engineering), Associate Professor.

Научная статья

УДК 535.566.2

doi: 10.34822/1999-7604-2022-3-109-113

ТЕМПЕРАТУРНАЯ ЗАВИСИМОСТЬ УДЕЛЬНОГО ВРАЩЕНИЯ ПЛОСКОСТИ ПОЛЯРИЗАЦИИ ДЛЯ ВОДНОГО РАСТВОРА САХАРА

Александр Геннадьевич Заводовский*Сургутский государственный университет, Сургут, Россия**averin117@mail.ru, <https://orcid.org/0000-0002-5419-7115>*

Аннотация. Влияние температуры на вращательную способность оптически активных веществ при использовании источников света с разной длиной волны определяется для каждого случая опытным путем. В основном имеются данные для $\lambda = 589,3$ нм. В настоящей работе определена температурная зависимость удельного вращения водного раствора сахара при использовании He-Ne лазера с длиной волны $\lambda = 632,8$ нм. Исследования выполнялись в интервале температур 20–80 °С для различной концентрации раствора. Полученная зависимость является линейной с угловым коэффициентом $0,043 \frac{\text{град}}{^\circ\text{C}}$. С увеличением температуры значение удельного вращения слабо уменьшается.

Ключевые слова: плоскость поляризации, оптически активные вещества, удельное вращение, лазер

Для цитирования: Заводовский А. Г. Температурная зависимость удельного вращения плоскости поляризации для водного раствора сахара // Вестник кибернетики. 2022. № 3 (47). С. 109–113. DOI 10.34822/1999-7604-2022-3-109-113.

Original article

TEMPERATURE DEPENDENCE OF THE SPECIFIC ROTATION OF SUGAR SOLUTION

Aleksandr G. Zavadovsky*Surgut State University, Surgut, Russia**averin117@mail.ru, <https://orcid.org/0000-0002-5419-7115>*

Abstract. The temperature effect on the rotatory power of optically active substances when using light sources with different wavelengths is determined experimentally for each case. The commonly known data is provided for $\lambda = 589.3$ nm. The article determines the temperature dependence of the specific rotation of sugar solution when using a He-Ne laser with a wavelength of $\lambda = 632.8$ nm. The experiments are carried out at the temperature range of 20–80 °C for various solution concentrations. The recorded dependence is linear with an angular coefficient of $0.043 \frac{\text{degrees}}{^\circ\text{C}}$. As the temperature increases, the value of the specific rotation decreases slightly.

Keywords: polarization plane, optically active substances, specific rotation, laser

For citation: Zavadovsky A. G. Temperature Dependence of the Specific Rotation of Sugar Solution // Proceedings in Cybernetics. 2022. No. 3 (47). P. 109–113. DOI 10.34822/1999-7604-2022-3-109-113.

ВВЕДЕНИЕ

Естественная оптическая активность вещества была открыта Араго в 1811 г. на пластинке кварца. В 1815 г. Ж. Б. Био обнаружил вращение плоскости поляризации также и водным раствором сахара. Позже естественное вращение плоскости поляризации было найдено у многих других веществ.

В зависимости от конкретного вещества это вращение может быть правосторонним или левосторонним по отношению к наблюдателю, к которому свет приближается. Вращение вправо считается положительным, а влево – отрицательным. Большинство естественно-активных жидкостей вращает плоскость поляризации в одну и ту же сторону, независи-

мо от направления распространения света. В случае растворов речь идет о растворе оптически активного вещества в неактивном растворителе. Опытным путем Ж. Б. Био установил для растворов качественный закон, который утверждает, что угол поворота плоскости поляризации φ прямо пропорционален толщине слоя раствора d и концентрации C оптически активного вещества:

$$\varphi = [\alpha] d C. \quad (1)$$

Коэффициент пропорциональности $[\alpha]$ характеризует природу вещества и называется постоянной вращения (удельным вращением) плоскости поляризации. Значение удельного вращения для желтой линии натрия $\lambda = 589,3$ нм при температуре 20°C обычно используют для сравнения оптической активности различных веществ. Этот параметр зависит от длины волны света и температуры раствора и может довольно сложным образом меняться при изменении растворителя [1]. Ж. Б. Био эксперименталь-

но определил, что постоянная вращения примерно обратно пропорциональна квадрату длины волны. Это утверждение передает зависимость неточно и является качественным. Опытные данные позволяют утверждать, что постоянная вращения естественно-активных веществ с увеличением длины волны убывает, но существуют вещества, для которых эта зависимость является аномальной. Анализ данных показывает, что области аномалии соответствуют полосам поглощения и устанавливают связь этой зависимости с явлением дисперсии. В настоящее время влияние длины волны на вращательную способность принято определять для каждого конкретного случая, выполняя соответствующие эксперименты. На рис. 1 представлена опытная зависимость удельного вращения от длины волны света (дисперсия оптического вращения) для водного раствора сахара, полученная при средних комнатных температурах [2].

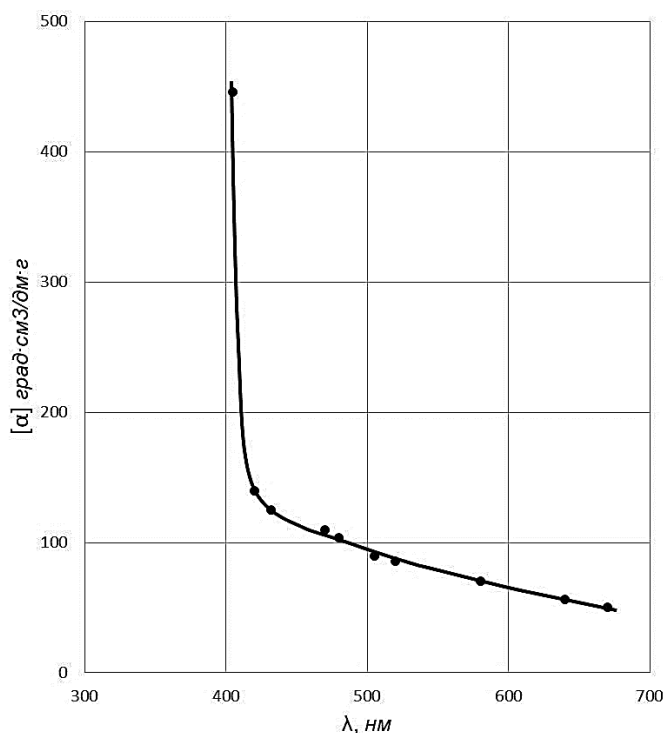


Рис. 1. Дисперсия оптического вращения для водного раствора сахара

Примечание: составлено на основании опытных данных [2].

Анализ экспериментальных результатов показывает, что постоянная вращения очень сильно уменьшается с увеличением длины

волны и для значения $\lambda = 632,8$ нм составляет примерно $57,3 \frac{\text{град} \cdot \text{см}^3}{\text{дм} \cdot \text{г}}$.

Опыт также показывает, что зависимость постоянной вращения от температуры очень слабая, и для большинства веществ она уменьшается незначительно при повышении температуры. Однако наблюдается для некоторых веществ и увеличение постоянной вращения с увеличением температуры. Поэтому влияние температуры на вращательную способность оптически активных веществ может быть определено лишь в общих чертах и для каждого случая также должно быть изучено опытным путем [3].

В литературе имеются результаты опытного исследования температурной зависимости удельного вращения для известных оптически активных веществ при использовании источников света с разной длиной волны. В основном данные получены для $\lambda = 589,3$ нм [4]. В настоящее время в качестве источников света широко используют

лазеры. Одним из них является He-Ne лазер с длиной волны 632,8 нм. На этой длине волны исследования температурной зависимости для водного раствора сахара не выполнялись. В работе [2] приведено значение удельного вращения на длине волны $\lambda = 636,2$ нм, и только при температуре 20 °С, оно составляет $56,51 \frac{\text{град} \cdot \text{см}^3}{\text{дм} \cdot \text{г}}$. В настоящей работе была экспериментально определена температурная зависимость удельного вращения плоскости поляризации для различных концентраций водного раствора сахара при использовании излучения He-Ne лазера с длиной волны 632,8 нм.

МАТЕРИАЛЫ И МЕТОДЫ

Для проведения опытных исследований была создана экспериментальная установка, схема которой представлена на рис. 2.

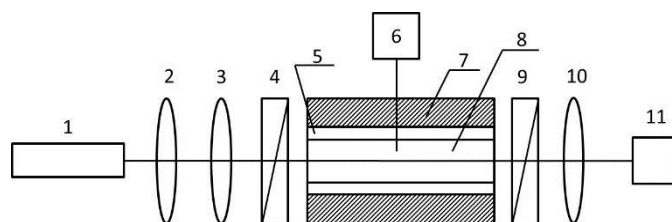


Рис. 2. Схема экспериментальной установки

1 – излучение лазера; 2, 3 – линзы; 4 – поляризатор; 5 – кювета; 6 – датчик температуры; 7 – нагреватель; 8 – водный раствор сахара; 9 – анализатор

Примечание: составлено автором.

Излучение лазера 1 с помощью линз 2, 3 и 10 фокусируется на приемную поверхность фотоприемника 11. Между поляризатором 4 и анализатором 9 располагается кювета 5 ($d = 12,5$ см), которая содержит исследуемый водный раствор сахара 8. Нагревание раствора осуществляется нагревателем 7, температура раствора контролируется датчиком температуры 6.

Перед началом эксперимента на оптической скамье между поляризатором и анализатором размещается незаполненная раствором кювета. Поворотом анализатора достигается минимальное значение интенсивности света, регистрируемое приемником, и находится величина угла анализатора φ_0 . Затем кювета заполняется водным раствором сахара определенной концентрации и определя-

ется его температура. Поворотом анализатора вновь достигается минимальное значение интенсивности света и регистрируется величина угла анализатора φ_p . На основании полученных данных определяется угол поворота плоскости поляризации $\varphi = \varphi_p - \varphi_0$. Затем включается нагреватель, который подогревает раствор до определенной температуры, контролируемой датчиком. Для каждого значения температуры определяется величина угла φ . Точность измерения температуры составляла $\pm 0,5$ °С, а измерения угла поворота плоскости поляризации $\varphi = \pm 0,5$ градуса.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

На рис. 3 представлены результаты экспериментальных исследований.

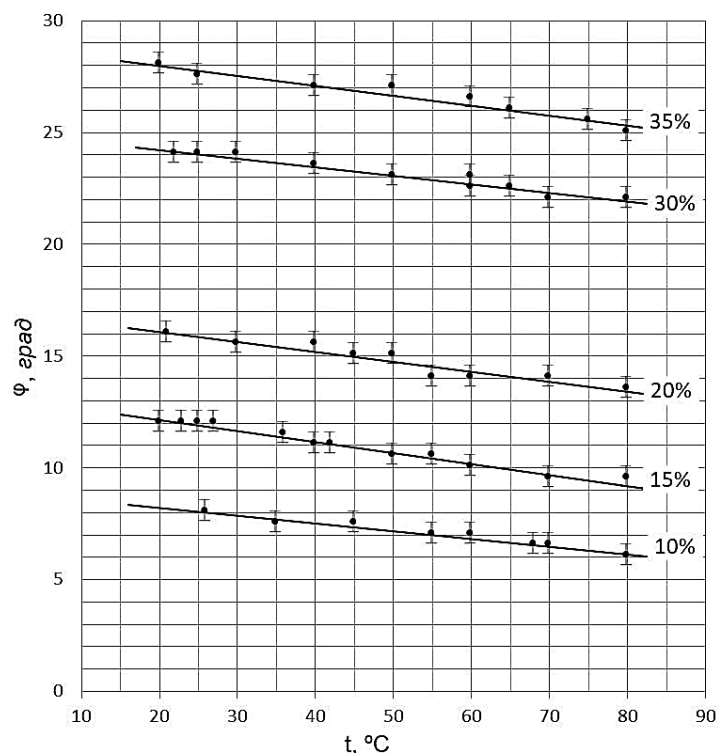


Рис. 3. Температурная зависимость удельного вращения плоскости поляризации для различных концентраций водного раствора сахара

Примечание: составлено автором.

В ходе выполнения работы были использованы водные растворы сахара различной концентрации (10 %, 15 %, 20 %, 30 % и 35 %). Температура раствора изменялась в интервале от 20 до 80 °С. При температуре 20 °С было определено среднее значение удельного вращения. Оно оказалось равным $[\alpha] = (60 \pm 3) \frac{\text{град} \cdot \text{см}^3}{\text{дм} \cdot \text{г}}$ и в пределах погрешности совпадает со значением, полученным на основании анализа данных работы [2].

Для всех концентраций раствора температурная зависимость удельного вращения была линейной в пределах погрешности измерений, и с увеличением температуры значение удельного вращения слабо уменьшалось, что согласуется с опытными данными. Угловой коэффициент наклона прямых в среднем составляет $0,043 \frac{\text{град}}{^\circ\text{C}}$, т. е. удельное вращение от концентрации не зависит.

ЗАКЛЮЧЕНИЕ

Метод поляриметрии широко используется в медицине и биохимических исследованиях [5, 6], а также в пищевой промышленности [7]. Так как влияние температуры на удельное вращение незначительно, то его обычно учитывают с помощью специальной методики [8]. Полученные в настоящей работе опытные данные позволяют более точно контролировать изучаемые процессы. Результаты данного экспериментального исследования могут быть использованы для создания оптических датчиков температуры [9]. Температурная зависимость в данном случае является линейной, что удобно для организации работы датчика, который бы измерял температуру в интервале от 10 до 80 °С. В качестве чувствительного элемента датчика можно использовать другие оптически активные вещества, например кварц.

Список источников

1. Смирнова Н. Г., Гильдеева Г. Н., Кулес В. Г. Оптическая изометрия и биологическая активность

References

1. Smirnova N. G., Gildееva G. N., Kukes V. G. Optical Isomerism and Biological Activity of Pharma-

- лекарственных средств // Вестн. Москов. ун-та. Сер. 2. Химия. 2012. Т. 53, № 3. С. 147–156.
2. Краткий физико-технический справочник / под ред. К. П. Яковлева. М. : Физматгиз, 1960. Т. 1. 446 с.
 3. Орлова А. В., Кондаков Н. Н., Зуев Ю. Ф., Кононов Л. О. Зависимость удельного оптического вращения водного раствора левоглюконата от температуры // Изв. АН. Сер. Химия 2018. № 11. С. 2155–2156.
 4. Справочник химика. 2-е изд., Т. 4 / под ред. Б. П. Никольского. Л. : Химия, 1967. 910 с.
 5. Государственная фармакопея РФ. Т. 1. М. : Науч. центр экспертизы средств мед. применения, 2015. 1470 с.
 6. Орлова А. В., Кононов Л. О. Поляриметрия как метод изучения структуры водных растворов углеводов: корреляция с другими методами // Радиоэлектроника. Наносистемы. Информ. технологии. 2020. Т. 12, № 1. С. 96–107.
 7. Косиченко Н. А. Новый подход к осветлителям при полиметрическом методе определения сахарозы // Сахар. пр-во. 2018. № 10. С. 22–24.
 8. ГОСТ 12517 – 2013. Сахар. Метод определения сахарозы. М. : Стандартинформ, 2016. 10 с.
 9. Датчики / под общ. ред. В. М. Шарапова, Е. С. Полищука. М. : Техносфера, 2012. 624 с.
 - ceutical Preparations // Moscow University Chemistry Bulletin. 2012. Vol. 53, No. 3. P. 147–156. (In Russian).
 2. Kratkii fiziko-tekhnicheskii spravochnik / Ed. K. P. Yakovlev. Moscow : Fizmatgiz, 1960. Vol. 1. 446 p. (In Russian).
 3. Orlova A. V., Kondakov N. N., Zuev Yu. F., Kononov L. O. Temperature Dependence of Specific Optical Rotation of an Aqueous Levoglucosan Solution // Russian Chemistry Bulletin. 2018. No. 11. P. 2155–2156. (In Russian).
 4. Spravochnik khimika. 2nd Ed., Vol. 4 / Ed. B. P. Nikolsky. Leningrad : Khimiia, 1967. 910 p. (In Russian).
 5. Gosudarstvennaia farmakopeia RF. Vol. 1. Moscow : Nauch. tsentr ekspertizy sredstv med. primeneniia, 2015. 1470 p. (In Russian).
 6. Orlova A. V., Kononov L. O. Polarimetry as a Method for Studying the Structure of Aqueous Carbohydrate Solutions: Correlation with Other Methods // Radioelectronics. Nanosystems. Information Technologies. 2020. Vol. 12, No. 1. P. 96–107. (In Russian).
 7. Kosichenko N. A. Novyi podkhod k osvetliteliam pri polimetrichekom metode opredeleniia sakharozy // Sakhar. pr-vo. 2018. No. 10. P. 22–24. (In Russian).
 8. State Standard 12517 – 2013. Sugar. Method for Determining Sucrose. Moscow : Standartinform, 2016. 10 p. (In Russian).
 9. Datchiki / Eds. V. M. Sharapov, E. S. Polishchuk. Moscow : Tekhnosfera, 2012. 624 p. (In Russian).

Информация об авторе

А. Г. Заводовский – кандидат физико-математических наук, доцент.

Information about the author

A. G. Zavodovsky – Candidate of Sciences (Physics and Mathematics), Associate Professor.

Научная статья

УДК 519.68

doi: 10.34822/1999-7604-2022-3-114-125

МОДЕЛЬНЫЙ КОМПЛЕКС ПОИСКА ОСНОВАНИЙ МОДУЛЯРНОЙ АРИФМЕТИКИ ДЛЯ ПРЕДОТВРАЩЕНИЯ БИВАЛЕНТНОГО ЭФФЕКТА

Сергей Арнольдович Инютин¹, Виктор Олегович Иванов²✉

^{1, 2} Московский авиационный институт (национальный исследовательский университет), Москва, Россия

¹ inyutin_int@mail.ru, <https://orcid.org/0000-0001-6879-8643>

² viktorivanov95@gmail.com ✉, <https://orcid.org/0000-0001-8597-1284>

Аннотация. Описана работа программного комплекса для решения задачи поиска пар простых чисел, равноудаленных от некоторого произвольного числа. Получены результаты работы с выделенными числами 2^{31} и 2^{32} на пределе возможностей серийных персональных компьютеров. Разработаны и описаны два разных алгоритма решения данной задачи с использованием языка программирования Python. Совпадение результатов, полученных двумя независимыми алгоритмами, можно трактовать как верификацию результатов решения задачи, визуальная идентификация которых затруднена из-за больших значений числовых величин. Полученные пары простых чисел можно использовать в качестве оснований модулярной арифметики, предназначенной для работы с большими числами, в которой устранен бивалентный эффект.

Ключевые слова: модулярная арифметика, бивалентный эффект цифрового регистра, модельный программный комплекс

Для цитирования: Инютин С. А., Иванов В. О. Модельный комплекс поиска оснований модулярной арифметики для предотвращения бивалентного эффекта // Вестник кибернетики. 2022. № 3 (47). С. 114–125. DOI 10.34822/1999-7604-2022-3-114-125.

Original article

MODEL COMPLEX FOR SEARCHING FOR MODULAR ARITHMETIC BASES TO PREVENT THE BIVALENT EFFECT

Sergey A. Inyutin¹, Viktor O. Ivanov²✉

^{1, 2} Moscow Aviation Institute (National Research University), Moscow, Russia

¹ inyutin_int@mail.ru, <https://orcid.org/0000-0001-6879-8643>

² viktorivanov95@gmail.com ✉, <https://orcid.org/0000-0001-8597-1284>

Abstract. The article describes a software package aimed at solving the problem of searching for pairs of prime numbers equidistant from a certain arbitrary number. The results of work with the selected numbers 2^{31} and 2^{32} are obtained at the limit of the capabilities of mass-produced personal computers. Two different algorithms for solving the problem have been developed using the Python programming language and are described herein. The results' agreement obtained via two independent algorithms can be interpreted as verification of the problem solution results, which are difficult to identify visually due to large numerical values. The obtained pairs of prime numbers can be used as bases in modular arithmetic designed to work with large numbers, with bivalent effect eliminated.

Keywords: modular arithmetic, bivalent effect of digital register, model software package

For citation: Inyutin S. A., Ivanov V. O. Model Complex for Searching for Modular Arithmetic Bases to Prevent the Bivalent Effect // Proceedings in Cybernetics. 2022. No. 3 (47). P. 114–125. DOI 10.34822/1999-7604-2022-3-114-125.

ВВЕДЕНИЕ

Для записи машинных кодов в компьютерных системах в настоящее время широко используется позиционная двоичная система счисления. Позиционная система счисления опирается на способ векторной записи числовой величины, в котором компоненты вектора (цифры) имеют различные веса, зависящие от ее положения в векторной записи. Зависимость веса цифр от их положения требует учета переносов из одних разрядов в соседние. Это замедляет выполнение компьютерных операций и усложняет арифметическое устройство вычислителя [1–2].

На современном этапе развития вычислительных систем особый интерес стали представлять компьютерные форматы для чисел, представленных в непозиционных системах счисления с параллельной структурой, которые позволяют сводить вычисления в основном к параллельным кольцевым операциям. Такой параллелизм позволяет делать модулярная система счисления (система остаточных классов).

Модулярная система счисления является непозиционной системой счисления, элементы которой описал китайский математик Сунь Цзы предположительно в III веке н. э. следующей формулировке:

- существует группа натуральных чисел $m_1, m_2, \dots, m_n$, называемых модулями (основаниями), и наибольший общий делитель каждой наугад выбранной пары чисел равен 1;

- существует группа натуральных чисел $r_1, r_2, \dots, r_n$, называемых остатками (вычетами), для которых соблюдается следующее неравенство $0 \leq r_i < m_i$, где $i \in \{1, 2, 3 \dots n\}$;

- натуральное число N можно представить в виде группы чисел r_i , в которой $r_i \equiv N \pmod{m_i}$;

- если найдутся два натуральных числа N_1 и N_2 , у которых для каждого r будет выполняться равенство $r_i(N_1) = r_i(N_2)$, то $|N_1|_M = |N_2|_M$, где $M_n = m_1 \cdot m_2 \cdot \dots \cdot m_n$.

Числовое значение натурального числа N в модулярной системе счисления представлено в виде вектора, компонентами которого являются остатки (вычеты) от деления натурального числа N на выбранные модули (ос-

нования) системы. Оно имеет следующий вид [3–5]:

$$N = (r_1, r_2, \dots, r_n).$$

Разряды числовых величин при вычислениях в модулярной системе счисления меняются независимо друг от друга, что позволяет распараллелить и ускорить процесс вычисления, а также устранить эффект размножения ошибок. Оставаясь в рамках позиционной системы счисления, значительного ускорения выполнения операций добиться практически невозможно [6–7].

При отображении в цифровых регистрах компонента вектора модулярного представления числовой величины возникает бивалентный эффект, заключающийся в том, что не все возможные двоичные комбинации в цифровом двоичном регистре используются для отображения этих компонент. Доказано, что для уменьшения бивалентного эффекта при использовании форматов модулярной арифметики в качестве оснований должны быть взяты пары простых чисел, равноудаленные от некоторого срединного основания (например, 2^{31} или 2^{32}). Выбор этих значений для срединных оснований удобен при эмуляции модулярной арифметики на 32-разрядных серийных компьютерах [8].

Для поиска таких чисел было разработано два разных алгоритма на языке программирования Python, выбранном по причине наличия библиотек, сильно упрощающих написание кода для научных вычислений (SciPy, NumPy), а также из-за возможности легкого создания графической визуализации результатов расчета (с помощью библиотеки Matplotlib).

Поиск простых чисел такой величины является сложной задачей почти на пределе возможностей серийных персональных компьютеров. Итерационный процесс в алгоритмах содержит много шагов. Для повышения скорости работы алгоритмов используются ресурсы библиотеки Numba со встроенным в нее компилятором LLVM, которые ускоряют выполнение кода на алгоритмическом языке Python [9].

Алгоритмы дают одинаковый результат, что доказывает их взаимную правильность при решении поставленной задачи.

МАТЕРИАЛЫ И МЕТОДЫ

Для поиска пар простых чисел, равноудаленных от 2^{31} и 2^{32} было разработано два различных алгоритма, запрограммированных на алгоритмическом языке Python. Каждый блок программного кода в обоих алгоритмах сопровождается комментарием, и в отладочном режиме для него выполняется трассировка алгоритма, а также выводятся результаты выполнения.

Скрипт первого алгоритма размещен на проекте GitHub [10].

Алгоритм содержит следующие этапы:

1. Подключаются две библиотеки (NumPy и Time):

- библиотека NumPy включает в себя функции для обработки массивов, что позволило сильно упростить процесс разработки алгоритма;

- библиотека Time позволила подсчитать время выполнения алгоритма.

2. Предлагается введение срединного числа (используется функция input), для которого необходимо найти пары равноудаленных простых чисел.

3. В алгоритме для проверки на простоту используется решето Эратосфена с делителями до целой части арифметического квадратного корня из удвоенного срединного числа.

Для иллюстрации работы алгоритма введем срединное число 25 и рассмотрим его выполнение ($[\sqrt{50}] = 7$). Проверке подлежат пары чисел: 24, 26; 23, 27 и т. д.

Если одно число из пары делится на число из множества [2, 3, 4, 5, 6, 7] без остатка, то цикл переходит к проверке следующей пары равноудаленных чисел.

Результат выполнения кода для значения 25 показан на рис. 1.

В начале трассировки были отброшены пары чисел 24 и 26, 23 и 27, 22 и 28, 21 и 29, 20 и 30. Далее началась проверка пар чисел 19 и 31, последовательным нахождением остатка от деления каждого из чисел пары на 2, 3, 4, 5, 6 и 7. Каждый раз при делении остаток был больше 0, из-за чего программа

в итоге выдала сообщение «Все возможные делители закончились, следовательно, 19 и 31 – простые числа!». Далее, таким же образом были найдены пары простых чисел 13 и 27.

Имеется особенность в случае проверки на простоту пар чисел, где одно число меньше или равно самому большому элементу во множестве делителей (в примере число равно $[\sqrt{50}] = 7$), например, пара чисел 7 и 43. По алгоритму число 43 проверяется на простоту делением на 2, 3, 4, 5, 6, 7. Первое число этой пары (7) таким образом проверить не можем, стандартная проверка показала бы, что число не является простым и пара простых чисел 7 и 43 не была бы зарегистрирована (т. к. $7 \% 7 = 0$). Поэтому был настроен фильтр, что числа, меньшие или равные самому большому числу во множестве проверочных делителей, проверяются на простоту делением только до числа меньшего проверяемого на единицу.

Фильтр также проверяет, равно ли меньшее число из пары делителю из массива. Если число не равно делителю, и если остаток при делении на него не равен нулю, то проверяется большее число из пары делением на это же число.

В примере проверяем, что 7 не равно 2, и что остаток от деления 7 на 2 не равен 0, а затем проверяем, не равен ли нулю остаток от деления 43 на 2 (рис. 2).

В конце трассировки появляются строки:

« $7 \leq 7$ и $43 \% 7 \neq 0$ поэтому проводим проверку этой пары дальше все возможные делители закончились, следовательно, 7 и 43 – простые числа! 3 найденная пара».

$7 \leq 7$, поэтому деление 7 на 7 для проверки не выполняется, выполняется только проверка $43 \% 7$.

Аналогичная процедура выполняется для пары чисел 3 и 47, также являющихся простыми и равноудаленными от 25.

Дополненный вариант первого алгоритма для работы с большими числами представлен в [11].

В работе этого алгоритма подключается библиотека Numba, которая позволяет ускорить выполнение алгоритма благодаря встроенному компилятору LLVM. Его использование способствует сокращению времени вычислений примерно в 40–50 раз.

```

IDLE Shell 3.9.2
File Edit Shell Debug Options Window Help
Введите число 25
Целая часть арифметического квадратного корня из умноженного на два введенного числа ( 50 ) равна
7
24 % 2 = 0 или 26 % 2 = 0, поэтому переходим к следующей паре чисел
23 % 2 != 0 и 27 % 2 != 0, проводим проверку этой пары дальше
23 % 3 = 0 или 27 % 3 = 0, поэтому переходим к следующей паре чисел
22 % 2 = 0 или 28 % 2 = 0, поэтому переходим к следующей паре чисел
21 % 2 != 0 и 29 % 2 != 0, проводим проверку этой пары дальше
21 % 3 = 0 или 29 % 3 = 0, поэтому переходим к следующей паре чисел
20 % 2 = 0 или 30 % 2 = 0, поэтому переходим к следующей паре чисел
19 % 2 != 0 и 31 % 2 != 0, проводим проверку этой пары дальше
19 % 3 != 0 и 31 % 3 != 0, проводим проверку этой пары дальше
19 % 4 != 0 и 31 % 4 != 0, проводим проверку этой пары дальше
19 % 5 != 0 и 31 % 5 != 0, проводим проверку этой пары дальше
19 % 6 != 0 и 31 % 6 != 0, проводим проверку этой пары дальше
19 % 7 != 0 и 31 % 7 != 0, проводим проверку этой пары дальше
все возможные делители закончились, следовательно 19 и 31 - простые числа! 1 найденная пара
18 % 2 = 0 или 32 % 2 = 0, поэтому переходим к следующей паре чисел
17 % 2 != 0 и 33 % 2 != 0, проводим проверку этой пары дальше
17 % 3 = 0 или 33 % 3 = 0, поэтому переходим к следующей паре чисел
16 % 2 = 0 или 34 % 2 = 0, поэтому переходим к следующей паре чисел
15 % 2 != 0 и 35 % 2 != 0, проводим проверку этой пары дальше
15 % 3 = 0 или 35 % 3 = 0, поэтому переходим к следующей паре чисел
14 % 2 = 0 или 36 % 2 = 0, поэтому переходим к следующей паре чисел
13 % 2 != 0 и 37 % 2 != 0, проводим проверку этой пары дальше
13 % 3 != 0 и 37 % 3 != 0, проводим проверку этой пары дальше
13 % 4 != 0 и 37 % 4 != 0, проводим проверку этой пары дальше
13 % 5 != 0 и 37 % 5 != 0, проводим проверку этой пары дальше
13 % 6 != 0 и 37 % 6 != 0, проводим проверку этой пары дальше
13 % 7 != 0 и 37 % 7 != 0, проводим проверку этой пары дальше
все возможные делители закончились, следовательно 13 и 37 - простые числа! 2 найденная пара
12 % 2 = 0 или 38 % 2 = 0, поэтому переходим к следующей паре чисел
11 % 2 != 0 и 39 % 2 != 0, проводим проверку этой пары дальше
11 % 3 = 0 или 39 % 3 = 0, поэтому переходим к следующей паре чисел
10 % 2 = 0 или 40 % 2 = 0, поэтому переходим к следующей паре чисел
9 % 2 != 0 и 41 % 2 != 0, проводим проверку этой пары дальше
9 % 3 = 0 или 41 % 3 = 0, поэтому переходим к следующей паре чисел
8 % 2 = 0 или 42 % 2 = 0, поэтому переходим к следующей паре чисел
7 != 2 и 7 % 2 != 0 поэтому проводим проверку этой пары дальше
7 != 2 и 43 % 2 != 0 поэтому проводим проверку этой пары дальше
7 != 3 и 7 % 3 != 0 поэтому проводим проверку этой пары дальше
7 != 3 и 43 % 3 != 0 поэтому проводим проверку этой пары дальше
7 != 4 и 7 % 4 != 0 поэтому проводим проверку этой пары дальше
7 != 4 и 43 % 4 != 0 поэтому проводим проверку этой пары дальше
7 != 5 и 7 % 5 != 0 поэтому проводим проверку этой пары дальше
7 != 5 и 43 % 5 != 0 поэтому проводим проверку этой пары дальше
7 != 6 и 7 % 6 != 0 поэтому проводим проверку этой пары дальше
7 != 6 и 43 % 6 != 0 поэтому проводим проверку этой пары дальше
7 <= 7 и 43 % 7 != 0 поэтому проводим проверку этой пары дальше
все возможные делители закончились, следовательно 7 и 43 - простые числа! 3 найденная пара
6 != 2 и 6 % 2 = 0 поэтому проводим проверку этой пары дальше
5 != 2 и 5 % 2 != 0 поэтому проводим проверку этой пары дальше
5 != 2 и 45 % 2 != 0 поэтому проводим проверку этой пары дальше
5 != 3 и 45 % 3 = 0 поэтому проводим проверку этой пары дальше
4 != 2 и 4 % 2 = 0 поэтому проводим проверку этой пары дальше
3 != 2 и 3 % 2 != 0 поэтому проводим проверку этой пары дальше
3 != 2 и 47 % 2 != 0 поэтому проводим проверку этой пары дальше
3 <= 3 и 47 % 3 != 0 поэтому проводим проверку этой пары дальше
3 <= 4 и 47 % 4 != 0 поэтому проводим проверку этой пары дальше
3 <= 5 и 47 % 5 != 0 поэтому проводим проверку этой пары дальше
3 <= 6 и 47 % 6 != 0 поэтому проводим проверку этой пары дальше
3 <= 7 и 47 % 7 != 0 поэтому проводим проверку этой пары дальше
все возможные делители закончились, следовательно 3 и 47 - простые числа! 4 найденная пара
Все возможные пары чисел для проверки закончились, проверка окончена
Количество найденных пар = 4
--- 2.690601110458374 seconds ---
Программа завершена
Press ENTER to exit
Ln:71 Col:19
  
```

Рис. 1. Результат выполнения первого алгоритма для введенного числа 25

Примечание: составлено авторами.

```

7 != 2 и 7 % 2 != 0 поэтому проводим проверку этой пары дальше
7 != 2 и 43 % 2 != 0 поэтому проводим проверку этой пары дальше
7 != 3 и 7 % 3 != 0 поэтому проводим проверку этой пары дальше
7 != 3 и 43 % 3 != 0 поэтому проводим проверку этой пары дальше
7 != 4 и 7 % 4 != 0 поэтому проводим проверку этой пары дальше
7 != 4 и 43 % 4 != 0 поэтому проводим проверку этой пары дальше
7 != 5 и 7 % 5 != 0 поэтому проводим проверку этой пары дальше
7 != 5 и 43 % 5 != 0 поэтому проводим проверку этой пары дальше
7 != 6 и 7 % 6 != 0 поэтому проводим проверку этой пары дальше
7 != 6 и 43 % 6 != 0 поэтому проводим проверку этой пары дальше
7 <= 7 и 43 % 7 != 0 поэтому проводим проверку этой пары дальше
все возможные делители закончились, следовательно 7 и 43 - простые числа! 3 найденная пара
  
```

Рис. 2. Результат выполнения первого алгоритма для проверки чисел 7 и 43

Примечание: составлено авторами.

Также в этом алгоритме добавлено значение узнать сколько пар простых чисел, «шага подсчета», благодаря которому можно равноудаленных от введенного числа,

находится на заданном от него расстоянии. Для нашего примера с числом 25, введем в качестве шага подсчета значение «5». Увидим результат, представленный на рис. 3.

Графическое изображение результатов выполнения алгоритма поиска пар простых чисел представлено на рис. 4.

```

*IDLE Shell 3.9.2*
File Edit Shell Debug Options Window Help
Введите число 25
Квадратный корень из двойного введенного числа ( 50 ) равен 7
Введите шаг 5
5 пар чисел просмотрено. Кол-во пар простых чисел = 0
19 и 31 - простые числа!
10 пар чисел просмотрено. Кол-во пар простых чисел = 1
13 и 37 - простые числа!
15 пар чисел просмотрено. Кол-во пар простых чисел = 1
7 и 43 - простые числа!
20 пар чисел просмотрено. Кол-во пар простых чисел = 1
3 и 47 - простые числа!
25 пар чисел просмотрено. Кол-во пар простых чисел = 1
Итоговый массив: [0 1 1 1 1]
--- 3.309372901916504 seconds ---
Программа завершена
Press ENTER to exit
Ln: 20 Col: 19
    
```

Рис. 3. Результат выполнения первого алгоритма для работы с большими числами
 Примечание: составлено авторами.

Введенное число: 25	Расстояние от 25:	Текст, выведенный на экран программой:
24 26	— расстояние от 25 равно 1	5 пар чисел просмотрено. Кол-во пар простых чисел = 0
23 27	— расстояние от 25 равно 2	
22 28	— расстояние от 25 равно 3	
21 29	— расстояние от 25 равно 4	
20 30	— расстояние от 25 равно 5	
19 31	— расстояние от 25 равно 6	19 и 31 - простые числа! 10 пар чисел просмотрено. Кол-во пар простых чисел = 1
18 32	— расстояние от 25 равно 7	
17 33	— расстояние от 25 равно 8	
16 34	— расстояние от 25 равно 9	
15 35	— расстояние от 25 равно 10	
14 36	— расстояние от 25 равно 11	
13 37	— расстояние от 25 равно 12	13 и 37 - простые числа! 15 пар чисел просмотрено. Кол-во пар простых чисел = 1
12 38	— расстояние от 25 равно 13	
11 39	— расстояние от 25 равно 14	
10 40	— расстояние от 25 равно 15	
9 41	— расстояние от 25 равно 16	
8 42	— расстояние от 25 равно 17	
7 43	— расстояние от 25 равно 18	7 и 43 - простые числа! 20 пар чисел просмотрено. Кол-во пар простых чисел = 1
6 44	— расстояние от 25 равно 19	
5 45	— расстояние от 25 равно 20	
4 46	— расстояние от 25 равно 21	
3 47	— расстояние от 25 равно 22	3 и 47 - простые числа! 25 пар чисел просмотрено. Кол-во пар простых чисел = 1
2 48	— расстояние от 25 равно 23	
1 49	— расстояние от 25 равно 24	
0 50	— расстояние от 25 равно 25	

Рис. 4. Графическое изображение результатов выполнения алгоритма поиска пар простых чисел, равноудаленных от 25, с шагом подсчета 5
 Примечание: составлено авторами.

На рис. 4 в первом столбце показаны пары чисел, равноудаленные от 25. Красным выделены те пары чисел, которые являются составными, зеленым — пары чисел,

являющиеся простыми. Весь ряд чисел разделен на 5 блоков (по параметру «шаг подсчета»), в каждом из блоков находится 5 пар чисел (т. к. $25/5 = 5$). В последнем блоке

последние две пары (1, 49) и (0, 50) полностью выделены красным, т. к. проверка этих пар чисел на простоту не осуществляется.

РЕЗУЛЬТАТЫ И ИХ ОБСУЖДЕНИЕ

Протестируем этот алгоритм для числа $2^{31} = 2147483648$. Выберем шаг подсчета

$10^8 = 100000000$. Для вычислений использовался компьютер, имеющий двухъядерный процессор и оперативную память 4 ГБ. Запустим алгоритм без вывода на экран самих простых чисел, а только итоговый результат (рис. 5).

```

*IDLE Shell 3.9.2*
File Edit Shell Debug Options Window Help
Введите число 2147483648
Квадратный корень из двойного введенного числа ( 4294967296 ) равен 65536
Введите шаг 100000000
22
[ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21]
100000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 285760
номер ячейки массива равен 0 значение ячейки 285760
200000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 285702
номер ячейки массива равен 1 значение ячейки 285702
300000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 285816
номер ячейки массива равен 2 значение ячейки 285816
400000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 285866
номер ячейки массива равен 3 значение ячейки 285866
500000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 286382
номер ячейки массива равен 4 значение ячейки 286382
600000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 287670
номер ячейки массива равен 5 значение ячейки 287670
700000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 288296
номер ячейки массива равен 6 значение ячейки 288296
800000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 287703
номер ячейки массива равен 7 значение ячейки 287703
900000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 288177
номер ячейки массива равен 8 значение ячейки 288177
1000000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 288293
номер ячейки массива равен 9 значение ячейки 288293
1100000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 290057
номер ячейки массива равен 10 значение ячейки 290057
1200000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 290123
номер ячейки массива равен 11 значение ячейки 290123
1300000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 292692
номер ячейки массива равен 12 значение ячейки 292692
1400000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 293237
номер ячейки массива равен 13 значение ячейки 293237
1500000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 294377
номер ячейки массива равен 14 значение ячейки 294377
1600000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 296790
номер ячейки массива равен 15 значение ячейки 296790
1700000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 299175
номер ячейки массива равен 16 значение ячейки 299175
1800000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 302428
номер ячейки массива равен 17 значение ячейки 302428
1900000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 306096
номер ячейки массива равен 18 значение ячейки 306096
2000000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 312302
номер ячейки массива равен 19 значение ячейки 312302
2100000000 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 324569
номер ячейки массива равен 20 значение ячейки 324569
2147483648 пар чисел просмотрено. Кол-во пар взаимно простых чисел = 169913
номер ячейки массива равен 21 значение ячейки 169913
[285760 285702 285816 285866 286382 287670 288296 287703 288177 288293
290057 290123 292692 293237 294377 296790 299175 302428 306096 312302
324569 169913]
--- 6727.976837396622 seconds ---
Программа завершена
Press ENTER to exit
Ln: 59 Col: 19

```

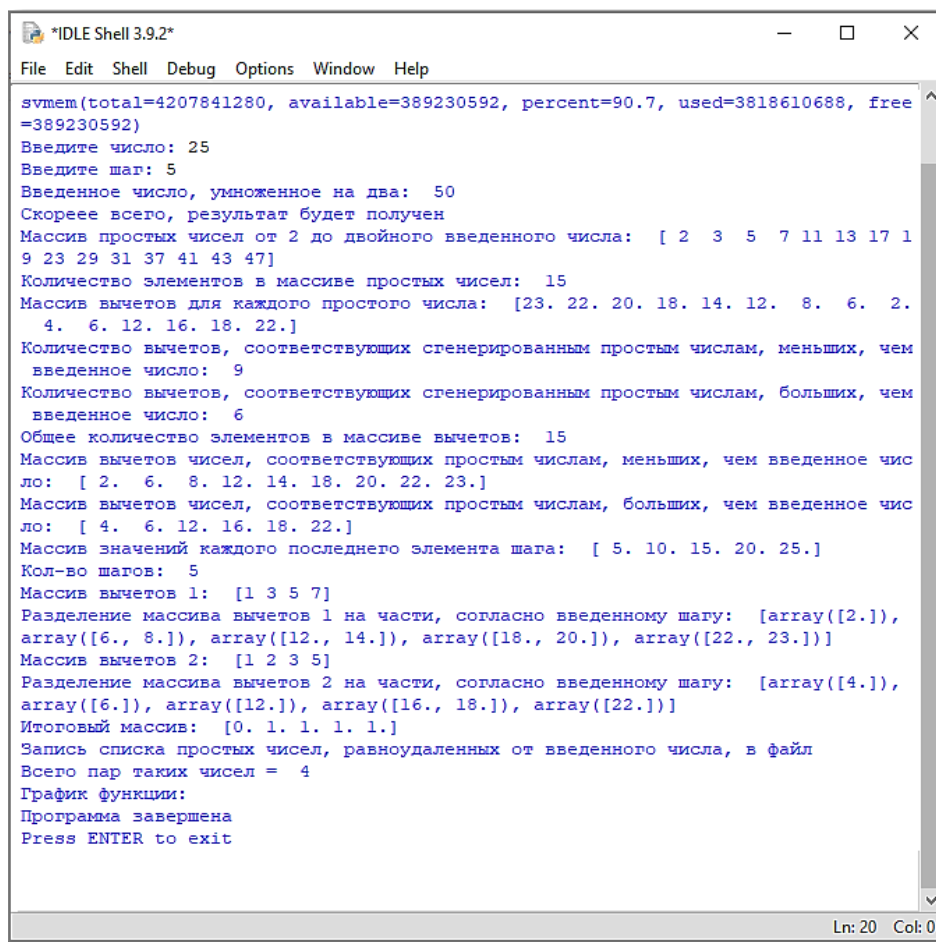
Рис. 5. Графическое изображение результата выполнения первого алгоритма для числа 2147483648 (2^{31}) с шагом подсчета 10^8
Примечание: составлено авторами.

В результате получено количество пар простых чисел, равноудаленных от 2^{31} . Вычисление удалось выполнить за 6196 секунд (около 103 минут).

Для проверки правильности результатов разработан второй алгоритм выполнения задачи, использующий принципиально другой способ вычислений. Попробуем оптимизировать его таким образом, чтобы вычисление происходило значительно быстрее.

Скрипт второго алгоритма представлен в [12].

Программа второго алгоритма содержит 13 блоков. Выполним его для числа 25 с шагом подсчета 5. Получим результат, представленный на рис. 6.



```
*IDLE Shell 3.9.2*
File Edit Shell Debug Options Window Help

svmem(total=4207841280, available=389230592, percent=90.7, used=3818610688, free=389230592)
Введите число: 25
Введите шаг: 5
Введенное число, умноженное на два: 50
Скорее всего, результат будет получен
Массив простых чисел от 2 до двойного введенного числа: [ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47]
Количество элементов в массиве простых чисел: 15
Массив вычетов для каждого простого числа: [23. 22. 20. 18. 14. 12. 8. 6. 2. 4. 6. 12. 16. 18. 22.]
Количество вычетов, соответствующих сгенерированным простым числам, меньших, чем введенное число: 9
Количество вычетов, соответствующих сгенерированным простым числам, больших, чем введенное число: 6
Общее количество элементов в массиве вычетов: 15
Массив вычетов чисел, соответствующих простым числам, меньших, чем введенное число: [ 2. 6. 8. 12. 14. 18. 20. 22. 23.]
Массив вычетов чисел, соответствующих простым числам, больших, чем введенное число: [ 4. 6. 12. 16. 18. 22.]
Массив значений каждого последнего элемента шага: [ 5. 10. 15. 20. 25.]
Кол-во шагов: 5
Массив вычетов 1: [1 3 5 7]
Разделение массива вычетов 1 на части, согласно введенному шагу: [array([2.]), array([6., 8.]), array([12., 14.]), array([18., 20.]), array([22., 23.])]
Массив вычетов 2: [1 2 3 5]
Разделение массива вычетов 2 на части, согласно введенному шагу: [array([4.]), array([6.]), array([12.]), array([16., 18.]), array([22.])]
Итоговый массив: [0. 1. 1. 1. 1.]
Запись списка простых чисел, равноудаленных от введенного числа, в файл
Всего пар таких чисел = 4
График функции:
Программа завершена
Press ENTER to exit

Ln: 20 Col: 0
```

Рис. 6. Графическое изображение результата выполнения второго алгоритма для числа 25 с шагом подсчета 5

Примечание: составлено авторами.

В начале его работы подключаются библиотеки Numba, Numpy, Time, Math, Psutil и модуль pyplot из библиотеки Matplotlib.

Описание трех библиотек (Numba, Numpy и Time) дано ранее, дадим описание других:

- Math содержит много математических функций, которые позволили упростить процесс разработки алгоритма [13];

- Psutil позволяет оперативно получать информацию о системных параметрах компьютера [14];

- Matplotlib позволяет визуализировать полученные данные с помощью графиков,

а ее модуль pyplot содержит ряд функций в стиле команд MATLAB [15].

Далее, в переменную mem вносится информация о системных параметрах компьютера (с помощью функции virtual_memory библиотеки psutil), которая выводится на экран:

- первый элемент «total» показывает общую доступную физическую память в байтах (в примере доступно 4207841280 байт (4,21 ГБ));

- второй элемент «available» показывает общее количество физической памяти, которое может быть отдано процессу без свопин-

га (в нашем случае доступно 497000448 байт, 0,5 ГБ);

- третий элемент «percent» вычисляет общее количество незанятой оперативной памяти в процентах (в примере 88,2 %);

- четвертый параметр «used» равен разности объемов памяти:

$$4207841280 - 497000448 = 3710840832.$$

Первый блок программы называется «Основные переменные». В нем с помощью функции input вводится серединное число, относительно которого нужно найти пары равноудаленных от него простых чисел, а также «шаг подсчета».

Как и в первом алгоритме, в качестве примера введем число 25 и «шаг подсчета» 5.

Как показала практика, результат с большей вероятностью будет вычислен, если введенное число примерно вдвое меньше доступной физической памяти в байтах, об этом говорит текст «Результат будет получен». В нашем случае общий объем доступной памяти равен 4207841280 байт, следовательно, максимальное число, для которого с большей вероятностью можно получить результат вычисления, равно 2103920640, что чуть меньше 2^{31} . Также практическим путем было обнаружено, что если число немного превышает деленный на два объем физической памяти, то результат все равно будет вычислен. В случае, если введенное число значительно превышает объем физической памяти, деленный на два, то с большей вероятностью возникнет ошибка «Out of memory».

В блоке «Генератор простых чисел» используется генератор простых чисел, опубликованный автором Robert William Hanks на портале Stackoverflow [16]. Автор утверждает, что данный генератор «Faster & more memory-wise pure Python code» («Наиболее быстрый и ресурсоемкий генератор простых чисел, разработанный на чистом Python»). Алгоритм генерирует простые числа, начиная с числа 3.

Третий этап алгоритма – преобразование полученного массива простых чисел в массив формата numpy, это необходимо для дальнейшей работы с ним с применением библиотеки numpy.

На четвертом этапе к массиву формата numpy добавляется число 2. Массив numpy с простыми числами выводится на экран.

Пятый этап алгоритма содержит вычисление вычетов для каждого простого числа (из каждого сгенерированного простого числа находится величина модуля разности с серединным числом 25). То есть, вычисляются значения $|2 - 25|$, $|3 - 25|$, $|5 - 25| \dots |43 - 25|$, $|47 - 25|$.

На шестом этапе алгоритма вычисляется каждый последний элемент шага и общее количество шагов.

На следующих четырех этапах (7–10) массивы вычетов (соответствующих простым числам меньшим, чем введенное число и большим, чем введенное число) делятся в соответствии с шагами.

На 11-м этапе алгоритма записывается итоговый массив, который равен $[0. 1. 1. 1. 1.]$.

Результат совпадает с тем результатом, который получен с помощью первого алгоритма.

На 12-м этапе полученный результат записывается в файл формата txt, затем результат отображается на графике (13-й этап).

Теперь для второго алгоритма введем серединное число $2^{31} = 2147483648$, с шагом $10^8 = 100000000$. Получим результат, представленный на рис. 7. Программа была запущена без вывода на экран пошаговых результатов.

Полученный результат совпадает с результатом работы первого алгоритма.

Графическое изображение, которое выводится на экран после выполнения алгоритма, представлено на рис. 8.

Изображение окна текстового файла с параметрами чисел, равноудаленных от введенного числа (в данном случае 2^{31}), которое создается в той же папке, где находится сам запускаемый алгоритм, представлено на рис. 9.

```
"IDLE Shell 3.9.2"
File Edit Shell Debug Options Window Help
svmem(total=17121157120, available=13319716864, percent=22.2, used=3801440256, free
=13319716864)
Введите число: 2147483648
Введите шаг: 100000000
Введенное число, умноженное на два: 4294967296
Скорее всего, результат будет получен
Итоговый массив: [285760. 285702. 285816. 285866. 286382. 287670. 288296. 287703.
288177.
288293. 290057. 290123. 292692. 293237. 294377. 296790. 299175. 302428.
306096. 312302. 324569. 169913.]
Запись списка простых чисел, равноудаленных от введенного числа, в файл
Всего пар таких чисел = 6341424
График функции:
Ln: 71 Cok: 19
```

Рис. 7. Графическое изображение результата выполнения второго алгоритма для числа 2^{31} с шагом подсчета 10^8
Примечание: составлено авторами.

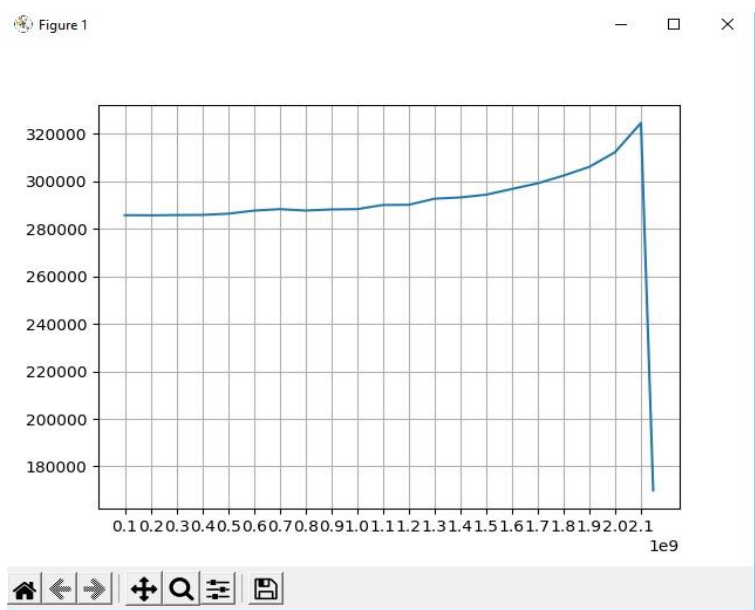


Рис. 8. График количества пар простых чисел, равноудаленных от 2^{31} .
По оси Y – количество пар чисел, по оси X – значение шага подсчета
Примечание: составлено авторами на основании данных, полученных в исследовании.

Prostye_chisla_ravnoudalennye_ot_vvedennogo_chisla – Блокнот			
Файл	Правка	Формат	Вид
1491923219, 2803044077	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655560429
1491923093, 2803044203	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655560555
1491922937, 2803044359	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655560711
1491922847, 2803044449	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655560801
1491922673, 2803044623	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655560975
1491922529, 2803044767	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655561119
1491922379, 2803044917	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655561269
1491922307, 2803044989	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655561341
1491922013, 2803045283	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655561635
1491921707, 2803045589	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655561941
1491921647, 2803045649	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655562001
1491921569, 2803045727	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655562079
1491921419, 2803045877	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655562229
1491920999, 2803046297	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655562649
1491920909, 2803046387	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655562739
1491920813, 2803046783	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655563135
1491920393, 2803046903	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655563255
1491919343, 2803047953	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655564305
1491919157, 2803048139	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655564491
1491918689, 2803048607	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655564959
1491918503, 2803048793	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655565145
1491918353, 2803048943	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655565295
1491918017, 2803049279	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655565631
1491917597, 2803049699	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655566051
1491917477, 2803049819	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655566171
1491917363, 2803049933	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655566285
1491917177, 2803050119	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655566471
1491916883, 2803050413	-	простые числа, равноудаленные от введенного числа	2147483648, расстояние между числами = 655566765

Рис. 9. Графическое изображение результатов выполнения алгоритма поиска пар простых чисел, равноудаленных от 2^{31}
Примечание: составлено авторами на основании данных, полученных в исследовании.

Выполним второй алгоритм для серединного числа $2^{32} = 4294967296$ с шагом $10^8 = 100000000$ на компьютере с оперативной памятью 16 ГБ (рис. 10).

Графическое изображение, которое выводится на экран после выполнения алгоритма, представлено на рис. 11.

```

*IDLE Shell 3.9.5*
File Edit Shell Debug Options Window Help
Скореее всего, результат будет получен
Массив простых чисел от 2 до двойного введенного числа: [      2      3
5 ... 8589934543 8589934567 8589934583]
Количество элементов в массиве простых чисел: 393615806
Массив вычетов для каждого простого числа: [4.29496729e+09 4.29496729e+09 4.29496729e
+09 ... 4.29496725e+09
4.29496727e+09 4.29496729e+09]
Количество вычетов, соответствующих сгенерированным простым числам, меньших, чем введе
нное число: 203280221
Количество вычетов, соответствующих сгенерированным простым числам, больших, чем введе
нное число: 190335585
Общее количество элементов в массиве вычетов: 393615806
Массив вычетов чисел, соответствующих простым числам, меньших, чем введенное число: [
5.00000000e+00 1.70000000e+01 6.50000000e+01 ... 4.29496729e+09
4.29496729e+09 4.29496729e+09]
Массив вычетов чисел, соответствующих простым числам, больших, чем введенное число: [
1.50000000e+01 6.10000000e+01 7.50000000e+01 ... 4.29496725e+09
4.29496727e+09 4.29496729e+09]
Массив значений каждого последнего элемента шага: [1.00000000e+08 2.00000000e+08 3.0000
000e+08 4.00000000e+08 5.00000000e+08
6.00000000e+08 7.00000000e+08 8.00000000e+08 9.00000000e+08 1.00000000e+09
1.10000000e+09 1.20000000e+09 1.30000000e+09 1.40000000e+09 1.50000000e+09
1.60000000e+09 1.70000000e+09 1.80000000e+09 1.90000000e+09 2.00000000e+09
2.10000000e+09 2.20000000e+09 2.30000000e+09 2.40000000e+09 2.50000000e+09
2.60000000e+09 2.70000000e+09 2.80000000e+09 2.90000000e+09 3.00000000e+09
3.10000000e+09 3.20000000e+09 3.30000000e+09 3.40000000e+09 3.50000000e+09
3.60000000e+09 3.70000000e+09 3.80000000e+09 3.90000000e+09 4.00000000e+09
4.10000000e+09 4.20000000e+09 4.2949673e+09]
Кол-во шагов: 43
Массив вычетов 1: [ 4511193  9026447 13546294 18071408 22602096 27139105 31681
923
36229868 40784377 45343807 49909426 54481785 59060791 63647320
68241384 72841914 77450316 82069110 86694683 91328790 95973934
100628003 105292521 109968180 114654277 119353801 124065968 128791863
133533660 138291329 143066396 147859606 152675688 157514798 162379718
167274357 172204610 177175786 182198079 187285674 192464358 197792265]
Разделение массива вычетов 1 на части, согласно введенному шагу: Squeezed text (86 lines).
Массив вычетов 2: [ 4506732  9007111 13504823 17997847 22485901 26970061 31449
211
35925025 40397266 44865718 49329353 53790201 58247307 62702435
67151451 71598092 76041571 80481077 84919096 89352848 93782510
98208324 102634084 107056411 111475103 115888844 120302159 124712871
129119249 133523125 137925238 142325252 146720283 151114036 155505470
159894812 164282641 168667919 173049000 177430081 181807055 186181981]
Разделение массива вычетов 2 на части, согласно введенному шагу: Squeezed text (86 lines).
Итоговый массив: [268783. 268187. 268537. 268184. 268401. 268635. 268729. 268644. 268
168.
268828. 268879. 269520. 269088. 270703. 270387. 270116. 269896. 270877.
270854. 271103. 271421. 272551. 272180. 272915. 273745. 273779. 274369.
275119. 275838. 276815. 277955. 278705. 279062. 280336. 282560. 283803.
285889. 287641. 290813. 294606. 299326. 308170. 317537.]
Запись списка простых чисел, равноудаленных от введенного числа, в файл
Всего пар таких чисел = 11891654
Ln: 10 Col: 0

```

Рис. 10. Графическое изображение результата выполнения второго алгоритма для числа 2^{32} с шагом подсчета 10^8
Примечание: составлено авторами.

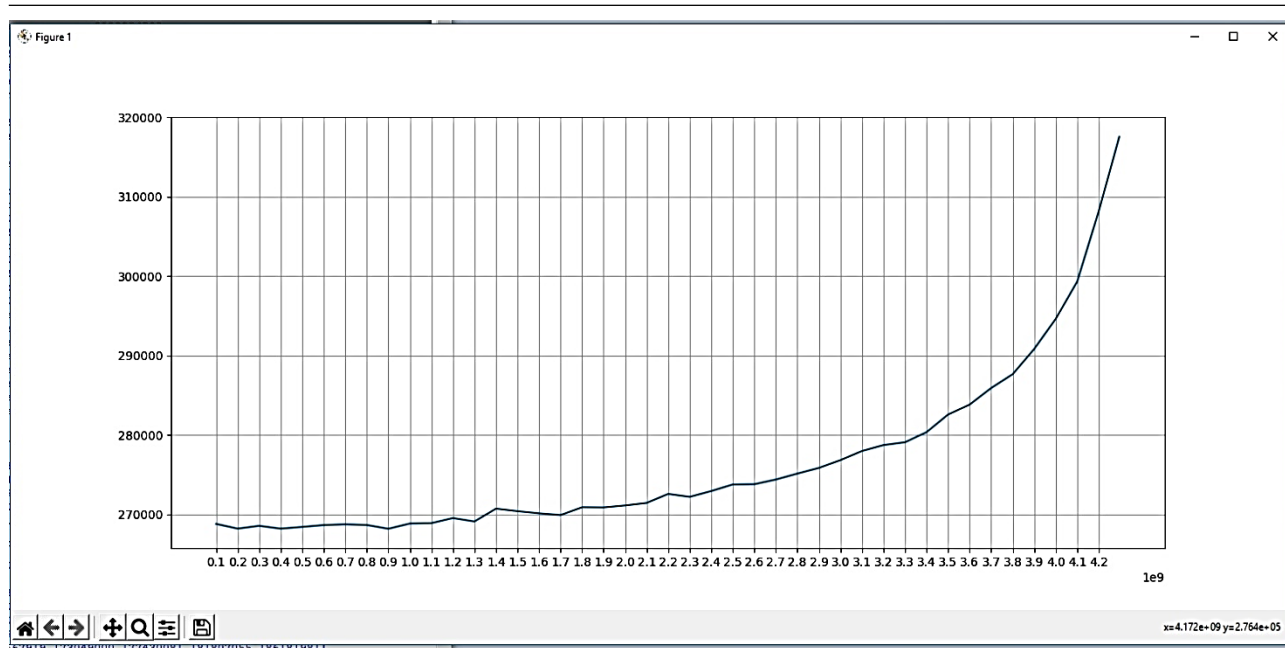


Рис. 11. График количества пар простых чисел, равноудаленных от 2^{32} .

По оси Y – количество пар чисел, по оси X – значение шага подсчета

Примечание: составлено авторами на основании данных, полученных в исследовании.

ЗАКЛЮЧЕНИЕ

При выборе срединного числа 2^{31} получено 6341424 пар простых чисел. Файл, содержащий их, имеет объем 755,7 МБ. При выборе срединного числа 2^{32} получено 11891654 пар простых чисел. Файл, содержащий их, имеет объем 1,39 ГБ. Полученные пары простых чисел можно использовать в качестве оснований модулярной арифметики, предназначенной для работы с большими числами и в которой устранен бивалентный эффект. Результаты работы двух различных алгоритмов полностью совпадают. Совпаде-

ние результатов работы свидетельствует о верификации программных реализаций алгоритмов. Сравнительный анализ алгоритмов свидетельствует, что первый алгоритм работает медленнее второго, с его использованием поиск пар простых чисел для заданного числа 2^{31} выполнен за 6196 секунд, а с использованием второго алгоритма аналогичный поиск выполнен за 51 секунду. Поиск пар простых чисел, равноудаленный от срединного числа 2^{32} , вторым алгоритмом был выполнен за 112 секунд.

Список источников

1. Инютин С. А. Модулярная алгоритмика много-разрядных вычислений. М. : Изд-во МАИ, 2020. 160 с.
2. Амербаев В. М., Балака Е. С., Тельпухов Д. В., Соловьев Р. А. Применение информационной избыточности для повышения надежности арифметического узла вычислительного элемента бимодульной арифметики // Параллельная компьютерная алгебра и ее приложения в новых инфокоммуникационных системах – 2014 : сб. науч. тр. I Междунар. конф. Ставрополь, 2014. С. 347–358.
3. Червяков Н. И., Бабенко М. Г., Кучеров Н. Н. Применение корректирующих кодов СОК для диагностики работы модулярных процессоров //

References

1. Inyutin S. A. Moduliarnaia algoritmiika mnogorazriadnykh vychislenii. Moscow : Publishing House Moscow Aviation Institute, 2020. 160 p. (In Russian).
2. Amerbaev V. M., Balaka E. S., Telpukhov D. V., Solovyev R. A. Application Information Redundancy to Improve Reliability the Arithmetic Unit Computing Elements Bimodule Arithmetic // Parallelnaia kompiuternaia algebra i ee prilozheniia v novykh infokommunikatsionnykh sistemakh – 2014 : Proceedings of the 1st International Conference. Stavropol, 2014. P. 347–358. (In Russian).
3. Chervyakov N. I., Babenko M. G., Kucherov N. N. The Use of Error-Correcting Codes for the Diagnosis of RNS Modular Processors // Science. Innovations. Technologies. 2014. No. 3. P. 24–39. (In Russian).

- Наука. Инновации. Технологии. 2014. № 3. С. 24–39.
4. Лавриненко А. В. Метод преобразования кода системы остаточных классов в позиционный с коррекцией ошибок на основе искусственных нейронных сетей // Наука. Инновации. Технологии. 2015. № 3. С. 7–36.
5. Эрдниева Н. С. Использование специальных модулей системы остаточных классов для избыточного представления // Вестн. АГТУ. Сер. Упр., вычислительная техника и информатика. 2013. № 2. С. 75–84.
6. Золотарева Н. С., Инютин С. А. Методы выбора оснований, понижающих бивалентный дефект в системе остаточных классов // Вестник кибернетики. 2020. С. 6–7.
7. Магомедов Ш. Г. Преобразование представлений чисел в модулярной арифметике в системах остаточных классов с разными основаниями // Вестн. АГТУ. Сер. Управление, вычислительная техника и информатика. 2014. № 4. С. 32–39.
8. Инютин С. А. Модулярные процессоры – оценки, история борьбы и победы над бивалентным дефектом // Развитие вычислительной техники в России и странах бывшего СССР: история и перспективы. SoRuCom-2017 : сб. тр. IV Междунар. конф., Зеленоград, 3–5 октября 2017 г. М., 2017. С. 72–77.
9. A ~5 Minute Guide to Numba. URL: <https://numba.readthedocs.io/en/stable/user/5minguide.html> (дата обращения: 01.09.2022).
10. Primesequidistants1.1. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants1.1> (дата обращения: 19.09.2022).
11. Primesequidistants1.2. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants1.2> (дата обращения: 19.09.2022).
12. Primesequidistants1.2. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants2.1> (дата обращения: 19.09.2022).
13. Math – Mathematical Functions. URL: <https://docs.python.org/3/library/math.html> (дата обращения: 01.09.2022).
14. Psutil 5.9.2. URL: <https://pypi.org/project/psutil/> (дата обращения: 01.09.2022).
15. Matplotlib: Visualization with Python. URL: <https://matplotlib.org> (дата обращения: 01.09.2022).
16. Fastest Way to List All Primes below N. URL: <https://stackoverflow.com/a/3035188/14513464> (дата обращения: 19.09.2022).
4. Lavrinenko A. V. Method of Conversion from Residue Number System to Positional with Error Correctness Based on Artificial Neural Networks // Science. Innovations. Technologies. 2015. No. 3. P. 7–36. (In Russian).
5. Erdnieva N. S. Use of Special Modules of the Residue Number System for Redundant Representation // Vestnik of Astrakhan State Technical University. Series: Management, Computer Science and Informatics. 2013. No. 2. P. 75–84. (In Russian).
6. Zolotareva N. S., Inyutin S. A. Methods for Selection of Bases Reducing Bivalent Defect in Residue Number System // Proceedings in Cybernetics. 2020. P. 6–7. (In Russian).
7. Magomedov Sh. G. Conversion of Numeration in Modular Arithmetic in the Systems of Residual Classes with Different Bases // Vestnik of Astrakhan State Technical University. Series: Management, Computer Science and Informatics. 2014. No. 4. P. 32–39. (In Russian).
8. Inyutin S. A. Modulniarnye protsessory – otsenki, istoriia borby i pobedy nad bivalentnym defektom // Computer Technology in Russia and in the Former Soviet Union. SoRuCom-2017 : Proceedings of the 4th International Conference, Zelenograd, October 3–5, 2017. Moscow, 2017. P. 72–77. (In Russian).
9. A ~5 Minute Guide to Numba. URL: <https://numba.readthedocs.io/en/stable/user/5minguide.html> (accessed: 01.09.2022).
10. Primesequidistants1.1. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants1.1> (accessed: 19.09.2022).
11. Primesequidistants1.2. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants1.2> (accessed: 19.09.2022).
12. Primesequidistants1.2. URL: <https://github.com/viktorivanov95/math/blob/main/primesequidistants2.1> (accessed: 19.09.2022).
13. Math – Mathematical Functions. URL: <https://docs.python.org/3/library/math.html> (accessed: 01.09.2022).
14. Psutil 5.9.2. URL: <https://pypi.org/project/psutil/> (accessed: 01.09.2022).
15. Matplotlib: Visualization with Python. URL: <https://matplotlib.org> (accessed: 01.09.2022).
16. Fastest Way to List All Primes below N. URL: <https://stackoverflow.com/a/3035188/14513464> (accessed: 19.09.2022).

Информация об авторах

С. А. Инютин – доктор технических наук, профессор.

В. О. Иванов – аспирант.

Information about the authors

S. A. Inyutin – Doctor of Sciences (Engineering), Professor.

V. O. Ivanov – Postgraduate.

Сетевое издание. Полные тексты статей размещаются
на официальном сайте издания <https://www.vestcyber.ru> и в базе данных
Научной электронной библиотеки на сайте elibrary.ru, сведения о публикуемых материалах включаются
в Российский индекс научного цитирования (РИНЦ)

Адрес учредителя и издателя:
бюджетное учреждение высшего образования Ханты-Мансийского автономного округа – Югры
«Сургутский государственный университет»,
628412, Ханты-Мансийский автономный округ – Югра, Сургут, пр. Ленина, 1. Тел. (3462) 76-30-67

Дата опубликования 11.11.2022. Формат 60 × 84/8
Уч.-изд. л. 12,7. Цена свободная