

[www.aurora-group.eu](http://www.aurora-group.eu)

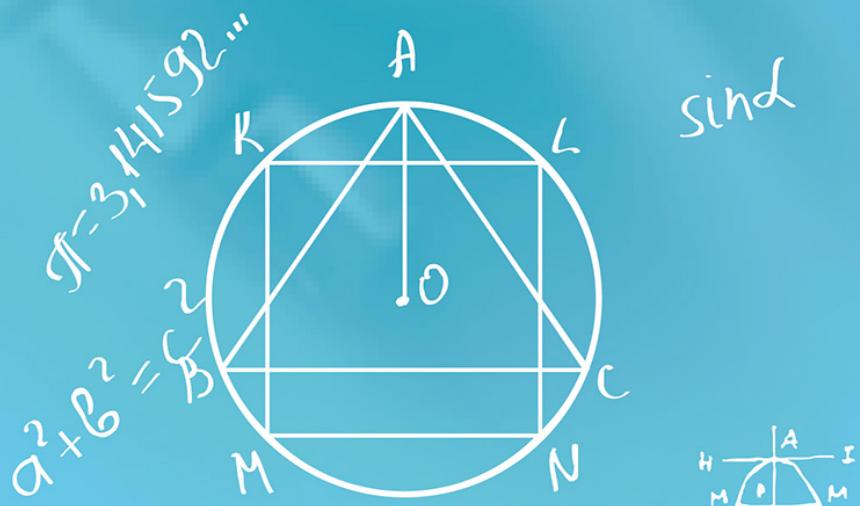
[www.nbpublish.com](http://www.nbpublish.com)

# ПРОГРАММНЫЕ СИСТЕМЫ

и

## ВЫЧИСЛИТЕЛЬНЫЕ МЕТОДЫ

научный журнал



**AURORA Group s.r.o.**

*nota bene*

## Выходные данные

Номер подписан в печать: 04-08-2023

Учредитель: Даниленко Василий Иванович, w.danilenko@nbpublish.com

Издатель: ООО <НБ-Медиа>

Главный редактор: Морозов Михаил Николаевич, кандидат технических наук,  
mikhail.n.morozov@gmail.com

ISSN: 2454-0714

Контактная информация:

Выпускающий редактор - Зубкова Светлана Вадимовна

E-mail: info@nbpublish.com

тел.+7 (966) 020-34-36

Почтовый адрес редакции: 115114, г. Москва, Павелецкая набережная, дом 6А, офис 211.

Библиотека журнала по адресу: [http://www.nbpublish.com/library\\_tariffs.php](http://www.nbpublish.com/library_tariffs.php)

## Publisher's imprint

Number of signed prints: 04-08-2023

Founder: Danilenko Vasiliy Ivanovich, w.danilenko@nbpublish.com

Publisher: NB-Media ltd

Main editor: Morozov Mikhail Nikolaevich, kandidat tekhnicheskikh nauk,  
mikhail.n.morozov@gmail.com

ISSN: 2454-0714

Contact:

Managing Editor - Zubkova Svetlana Vadimovna

E-mail: info@nbpublish.com

тел.+7 (966) 020-34-36

Address of the editorial board : 115114, Moscow, Paveletskaya nab., 6A, office 211 .

Library Journal at : [http://en.nbpublish.com/library\\_tariffs.php](http://en.nbpublish.com/library_tariffs.php)

## РЕДАКЦИОННЫЙ СОВЕТ

**Гельман Виктор Яковлевич** – доктор технических наук, профессор, профессор кафедры медицинской информатики и физики ФГБОУ ВО «Северо-Западный государственный медицинский университет им. И.И.Мечникова», 191015, Россия, г. Санкт-Петербург, ул. Кирочная, д.41, [gelm@sg2104.spb.edu](mailto:gelm@sg2104.spb.edu)

**Поляков Виктор Павлович** – доктор педагогических наук, профессор, Главный научный сотрудник лаборатории психолого-педагогического и учебно-методического обеспечения развития информатизации образования Центра информатизации образования Федерального государственного бюджетного научного учреждения «Институт управления образованием Российской академии образования», 105062, г. Москва, ул. Макаренко, д. 5/16, стр. 1Б, [polvikpal@mail.ru](mailto:polvikpal@mail.ru)

**Гармаев Баир Заятуевич** – кандидат физико-математических наук, научный сотрудник, Институт физического материаловедения Сибирского Отделения РАН, 670000, Россия, Республика Бурятия, г. Улан-Удэ, ул. Сахьяновой, 6, каб. 313

**Клименко Анна Борисовна** – кандидат технических наук, научный сотрудник Научно-исследовательского института многопроцессорных вычислительных систем имени академика А.В. Каляева Южного федерального университета (НИИ МВС ЮФУ), 347935, Россия, Ростовская область, г. Таганрог, ул. 8 Переулок, 15

**Лютикова Лариса Адольфовна** – кандидат физико-математических наук, заведующая отделом Нейроинформатики и машинного обучения, Институт прикладной математики и автоматизации Кабардино-Балкарского научного центра РАН – филиал Кабардино-Балкарского научного центра РАН (ИПМА КБНЦ РАН), 360000, Россия, Республика Кабардино-Балкария, г. Нальчик, ул. Шортанова, 89а

**Мустафаев Арслан Гасанович** – доктор технических наук, профессор, Государственное автономное образовательное учреждение высшего образования "Дагестанский государственный университет народного хозяйства", кафедра «Информационные технологии и информационная безопасность», 367015, Россия, Республика Дагестан, г. Махачкала, ул. Атаева, 5, каб. 4.5

**Шестаков Александр Валентинович** – кандидат технических наук, доцент Южный Федеральный университет, кафедра вычислительной техники, 347902, Россия, Ростовская область, г. Таганрог, ул. Свободы, 24/2

**Сидоркина Ирина Геннадьевна** - доктор технических наук, профессор, декан факультета Информатики и вычислительной техники Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: [dekan\\_fivt@mail.ru](mailto:dekan_fivt@mail.ru)

**Екатерина Прасолова-Forland** - PhD, Норвежский университет науки и технологии (NTNU), Тронхейм, Норвегия E-mail: [Ekaterina.Prasolova-Forland@idi.ntnu.no](mailto:Ekaterina.Prasolova-Forland@idi.ntnu.no)

**Голенков Владимир Васильевич** - доктор технических наук, профессор, заведующий кафедрой Интеллектуальных информационных технологий Белорусского государственного университета информатики и радиоэлектроники, г. Минск, Республика Беларусь E-mail: [golen@bsuir.by](mailto:golen@bsuir.by)

**Домошицкий Александр Исаакович** - кандидат физико-математических наук, декан естественно-научного факультета Университетского центра в г.Ариэль, Израиль, Самария E-mail: adom@ariel.ac.il Department of Mathematics and Computer Sciences, The Ariel University Center of Samaria, 44837 Ariel, ISRAEL

**Коробейников Анатолий Григорьевич** - доктор технических наук, профессор «Институт земного магнетизма, ионосферы и распространения радиоволн РАН (ИЗМИРАН)»,Санкт-Петербургский филиал E-mail: [korobeynikov\\_a\\_g@mail.ru](mailto:korobeynikov_a_g@mail.ru)

**Заболеева-Зотова Алла Викторовна**, доктор технических наук, профессор Волгоградского технического университета, Волгоград, Россия E-mail: [zabzot@gmail.com](mailto:zabzot@gmail.com)

**Бенкевич Леонид Владимирович** - кандидат физических наук и инженерной физики, научный сотрудник Массачусетского Технологического Института (MIT), обсерватория Хэйстек, Бостон, США E-mail: [lbenkev@gmail.com](mailto:lbenkev@gmail.com)

**Морозов Михаил Николаевич** - кандидат технических наук, профессор, руководитель лаборатории мультимедиа, заведующий кафедрой Информатики и системного программирования Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: [mikhail.n.morozov@gmail.com](mailto:mikhail.n.morozov@gmail.com)

**Олзоева Сэсэг Ивановна** - доктор технических наук, профессор, Восточно-Сибирский государственный университет технологий и управления (г. Улан-Удэ) E-mail: [sseseg@yandex.ru](mailto:sseseg@yandex.ru)

**Курейчик Владимир Викторович** - доктор технических наук, профессор, заведующий кафедрой Систем автоматизации проектирования Технологического института «Южного федерального университета» в г.Таганрог, Россия E-mail: [vkur@tsure.ru](mailto:vkur@tsure.ru)

**Филатова Наталья Николаевна** - доктор технических наук, профессор, Тверской государственный технический университет, Тверь, Россия E-mail: [nfilatova99@mail.ru](mailto:nfilatova99@mail.ru)

**Песошин Валерий Андреевич** - член-корреспондент Академии наук Республики Татарстан, заслуженный деятель науки Республики Татарстан и Российской Федерации, доктор технических наук, профессор, заслуженный деятель науки и техники Республики Татарстан. Заведующий кафедрой Компьютерных систем Казанского национальный исследовательский университет им. А.Н. Туполева, Казань, Россия E-mail: [pesoshin@evm.kstu-kai.ru](mailto:pesoshin@evm.kstu-kai.ru)

**Краснов Сергей Викторович** - доктор технических наук, профессор, проректор по научно-исследовательской работе, заведующий кафедрой Информатика и системы управления Волжского университета им. Татищева, Тольятти, Россия E-mail: [krasnovtlt@mail.ru](mailto:krasnovtlt@mail.ru)

**Горохов Алексей Витальевич** - доктор технических наук, профессор кафедры Прикладной математики и информационных технологий Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: [agv64@mail.ru](mailto:agv64@mail.ru)

**Галанина Наталья Андреевна** - доктор технических наук, профессор, Чувашский государственный университет им. И.Н.Ульянова, Чебоксары, Россия E-mail: [galaninacheb@mail.ru](mailto:galaninacheb@mail.ru)

**Сюзев Владимир Васильевич** - доктор технических наук, профессор, заведующий кафедрой Компьютерные системы и сети Московского государственного технического университета им.

Н. Э. Баумана, Москва, Россия E-mail: [v.suzev@bmstu.ru](mailto:v.suzev@bmstu.ru)

**Леухин Анатолий Николаевич** - доктор физико-математических наук, профессор, заведующий кафедрой Информационной безопасности Поволжского государственного технологического университета, Йошкар-Ола, Россия E-mail: [code@volgattech.net](mailto:code@volgattech.net)

**Гвинианидзе Темур Николаевич** - Доктор технических наук, профессор, Государственный университет им. Ак. Церетели Грузия, г. Кутаиси, пр. Тамар-мепе 59. П.и 4600.

[temuri1951@mail.ru](mailto:temuri1951@mail.ru)

## COUNCIL OF EDITORS

**Gelman Viktor Yakovlevich** – Doctor of Technical Sciences, Professor, Professor of the Department of Medical Informatics and Physics of the I.I.Mechnikov Northwestern State Medical University, 41 Kirochnaya Str., St. Petersburg, 191015, Russia, [gelm@sg2104.spb.edu](mailto:gelm@sg2104.spb.edu)

**Polyakov Viktor Pavlovich** – Doctor of Pedagogical Sciences, Professor, Chief Researcher of the Laboratory of Psychological, Pedagogical and Educational methodological support for the development of Informatization of Education of the Center for Informatization of Education of the Federal State Budgetary Scientific Institution "Institute of Education Management of the Russian Academy of Education", 105062, Moscow, Makarenko str., 5/16, p. 1B, [polvikpal@mail.ru](mailto:polvikpal@mail.ru)

**Garmaev Bair Zayatuevich** – Candidate of Physical and Mathematical Sciences, Researcher, Institute of Physical Materials Science of the Siberian Branch of the Russian Academy of Sciences, 670000, Russia, Republic of Buryatia, Ulan-Ude, Sakyanova str., 6, room 313

**Klimenko Anna Borisovna** – Candidate of Technical Sciences, Researcher at the Research Institute of Multiprocessor Computing Systems named after Academician A.V. Kalyaev of the Southern Federal University (Research Institute of the Ministry of Internal Affairs of the Southern Federal University), 347935, Russia, Rostov region, Taganrog, ul. 8 Lane, 15

**Lyutikova Larisa Adolfovna** – Candidate of Physical and Mathematical Sciences, Head of the Department of Neuroinformatics and Machine Learning, Institute of Applied Mathematics and Automation of the Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences - branch of the Kabardino-Balkarian Scientific Center of the Russian Academy of Sciences (IPMA KBSC RAS), 360000, Russia, Republic of Kabardino-Balkaria, Nalchik, 89a Shortanova str.

**Mustafayev Arslan Hasanovich** – Doctor of Technical Sciences, Professor, State Autonomous Educational Institution of Higher Education "Dagestan State University of National Economy", Department of "Information Technologies and Information Security", 367015, Russia, Republic of Dagestan, Makhachkala, Ataeva str., 5, office 4.5

**Alexander V. Shestakov** – Candidate of Technical Sciences, Associate Professor, Southern Federal University, Department of Computer Engineering, 24/2 Svobody str., Taganrog, Rostov Region, 347902, Russia

**Sidorkina Irina Gennadievna** - Doctor of Technical Sciences, Professor, Dean of the Faculty of Computer Science and Computer Engineering of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: [dekan\\_fivt@mail.ru](mailto:dekan_fivt@mail.ru)

**Ekaterina Prasolova-Forland** - PhD, Norwegian University of Science and Technology (NTNU), Trondheim, Norway E-mail: [Ekaterina.Prasolova-Forland@idi.ntnu.no](mailto:Ekaterina.Prasolova-Forland@idi.ntnu.no)

**Golenkov Vladimir Vasilevich** - Doctor of Technical Sciences, Professor, Head of the Department of Intelligent Information Technologies of the Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus E-mail: [golen@bsuir.by](mailto:golen@bsuir.by)

**Domoshnitsky Alexander Isaakovich** - Candidate of Physical and Mathematical Sciences, Dean of the Faculty of Natural Sciences of the University Center in Ariel, Israel, Samaria E-mail: [adom@ariel.ac.il](mailto:adom@ariel.ac.il) Department of Mathematics and Computer Sciences, The Ariel University Center

of Samaria, 44837 Ariel, ISRAEL

**Korobeynikov Anatoly Grigorievich** - Doctor of Technical Sciences, Professor, Institute of Terrestrial Magnetism, Ionosphere and Radio Wave Propagation of the Russian Academy of Sciences (IZMIRAN), St. Petersburg Branch E-mail: [korobeynikov\\_a\\_g@mail.ru](mailto:korobeynikov_a_g@mail.ru)

**Zaboleeva-Zotova Alla Viktorovna**, Doctor of Technical Sciences, Professor of Volgograd Technical University, Volgograd, Russia E-mail: [zabzot@gmail.com](mailto:zabzot@gmail.com)

**Leonid V. Benkevich** - Candidate of Physical Sciences and Engineering Physics, Researcher at the Massachusetts Institute of Technology (MIT), Haystack Observatory, Boston, USA E-mail: [ibenkev@gmail.com](mailto:ibenkev@gmail.com)

**Mikhail N. Morozov** - Candidate of Technical Sciences, Professor, Head of the Multimedia Laboratory, Head of the Department of Computer Science and System Programming of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: [mikhail.n.morozov@gmail.com](mailto:mikhail.n.morozov@gmail.com)

**Olzoeva Seseg Ivanovna** - Doctor of Technical Sciences, Professor, East Siberian State University of Technology and Management (Ulan-Ude) E-mail: [sseseg@yandex.ru](mailto:sseseg@yandex.ru)

**Kureychik Vladimir Viktorovich** - Doctor of Technical Sciences, Professor, Head of the Department of Design Automation Systems of the Technological Institute of the Southern Federal University in Taganrog, Russia E-mail: [vkur@tsure.ru](mailto:vkur@tsure.ru)

**Natalia Filatova** - Doctor of Technical Sciences, Professor, Tver State Technical University, Tver, Russia E-mail: [nfilatova99@mail.ru](mailto:nfilatova99@mail.ru)

**Pesoshin Valery Andreevich** - Corresponding member of the Academy of Sciences of the Republic of Tatarstan, Honored Scientist of the Republic of Tatarstan and the Russian Federation, Doctor of Technical Sciences, Professor, Honored Worker of Science and Technology of the Republic of Tatarstan. Head of the Department of Computer Systems of Kazan National Research University named after A.N. Tupolev, Kazan, Russia E-mail: [pesoshin@evm.kstu-kai.ru](mailto:pesoshin@evm.kstu-kai.ru)

**Krasnov Sergey Viktorovich** - Doctor of Technical Sciences, Professor, Vice-Rector for Research, Head of the Department of Computer Science and Control Systems of the Volga State University. Tatishcheva, Togliatti, Russia E-mail: [krasnovtlt@mail.ru](mailto:krasnovtlt@mail.ru)

**Gorokhov Alexey Vitalievich** - Doctor of Technical Sciences, Professor of the Department of Applied Mathematics and Information Technologies of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: [agv64@mail.ru](mailto:agv64@mail.ru)

**Galanina Natalia Andreevna** - Doctor of Technical Sciences, Professor, I.N.Ulyanov Chuvash State University, Cheboksary, Russia E-mail: [galaninacheb@mail.ru](mailto:galaninacheb@mail.ru)

**Vladimir V. Syuzev** - Doctor of Technical Sciences, Professor, Head of the Department of Computer Systems and Networks of the Bauman Moscow State Technical University, Moscow, Russia E-mail: [v.suzev@bmstu.ru](mailto:v.suzev@bmstu.ru)

**Leukhin Anatoly Nikolaevich** - Doctor of Physical and Mathematical Sciences, Professor, Head of the Department of Information Security of the Volga State Technological University, Yoshkar-Ola, Russia E-mail: [code@volgatech.net](mailto:code@volgatech.net)

**Gvinianidze Temur Nikolaevich** -Doctor of Technical Sciences, Professor, Ak. Tsereteli State University, Georgia, Kutaisi, 50 Tamarisvili Ave., 38000, [temur@tgc.edu.ge](mailto:temur@tgc.edu.ge)



## Требования к статьям

Журнал является научным. Направляемые в издательство статьи должны соответствовать тематике журнала (с его рубрикатором можно ознакомиться на сайте издательства), а также требованиям, предъявляемым к научным публикациям.

Рекомендуемый объем от 12000 знаков.

Структура статьи должна соответствовать жанру научно-исследовательской работы. В ее содержании должны обязательно присутствовать и иметь четкие смысловые разграничения такие разделы, как: предмет исследования, методы исследования, апелляция к оппонентам, выводы и научная новизна.

Не приветствуется, когда исследователь, трактуя в статье те или иные научные термины, вступает в заочную дискуссию с авторами учебников, учебных пособий или словарей, которые в узких рамках подобных изданий не могут широко излагать свое научное воззрение и заранее оказываются в проигрышном положении. Будет лучше, если для научной полемики Вы обратитесь к текстам монографий или докторских диссертаций работ оппонентов.

Не превращайте научную статью в публицистическую: не наполняйте ее цитатами из газет и популярных журналов, ссылками на высказывания по телевидению.

Ссылки на научные источники из Интернета допустимы и должны быть соответствующим образом оформлены.

Редакция отвергает материалы, напоминающие реферат. Автору нужно не только продемонстрировать хорошее знание обсуждаемого вопроса, работ ученых, исследовавших его прежде, но и привнести своей публикацией определенную научную новизну.

Не принимаются к публикации избранные части из докторских диссертаций, книг, монографий, поскольку стиль изложения подобных материалов не соответствует журнальному жанру, а также не принимаются материалы, публиковавшиеся ранее в других изданиях.

В случае отправки статьи одновременно в разные издания автор обязан известить об этом редакцию. Если он не сделал этого заблаговременно, рискует репутацией: в дальнейшем его материалы не будут приниматься к рассмотрению.

Уличенные в плагиате попадают в «черный список» издательства и не могут рассчитывать на публикацию. Информация о подобных фактах передается в другие издательства, в ВАК и по месту работы, учебы автора.

Статьи представляются в электронном виде только через сайт издательства <http://www.enotabene.ru> кнопка "Авторская зона".

Статьи без полной информации об авторе (соавторах) не принимаются к рассмотрению, поэтому автор при регистрации в авторской зоне должен ввести полную и корректную информацию о себе, а при добавлении статьи - о всех своих соавторах.

Не набирайте название статьи прописными (заглавными) буквами, например: «ИСТОРИЯ КУЛЬТУРЫ...» — неправильно, «История культуры...» — правильно.

При добавлении статьи необходимо прикрепить библиографию (минимум 10–15 источников, чем больше, тем лучше).

При добавлении списка использованной литературы, пожалуйста, придерживайтесь следующих стандартов:

- [ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления.](#)
- [ГОСТ 7.0.5-2008 Библиографическая ссылка. Общие требования и правила составления](#)

В каждой ссылке должен быть указан только один диапазон страниц. В теле статьи ссылка на источник из списка литературы должна быть указана в квадратных скобках, например, [1]. Может быть указана ссылка на источник со страницей, например, [1, с. 57], на группу источников, например, [1, 3], [5-7]. Если идет ссылка на один и тот же источник, то в теле статьи нумерация ссылок должна выглядеть так: [1, с. 35]; [2]; [3]; [1, с. 75-78]; [4]....

А в библиографии они должны отображаться так:

[1]  
[2]  
[3]  
[4]....

Постраничные ссылки и сноски запрещены. Если вы используете сноски, не содержащую ссылку на источник, например, разъяснение термина, включите сноски в текст статьи.

После процедуры регистрации необходимо прикрепить аннотацию на русском языке, которая должна состоять из трех разделов: Предмет исследования; Метод, методология исследования; Новизна исследования, выводы.

Прикрепить 10 ключевых слов.

Прикрепить саму статью.

Требования к оформлению текста:

- Кавычки даются углками (« ») и только кавычки в кавычках — лапками (" ").
- Тире между датамидается короткое (Ctrl и минус) и без отбивок.
- Тире во всех остальных случаяхдается длинное (Ctrl, Alt и минус).
- Даты в скобках даются без г.: (1932–1933).
- Даты в тексте даются так: 1920 г., 1920-е гг., 1540–1550-е гг.
- Недопустимо: 60-е гг., двадцатые годы двадцатого столетия, двадцатые годы XX столетия, 20-е годы ХХ столетия.
- Века, король такой-то и т.п. даются римскими цифрами: XIX в., Генрих IV.
- Инициалы и сокращения даются с пробелом: т. е., т. д., М. Н. Иванов. Неправильно: М.Н. Иванов, М.Н. Иванов.

## **ВСЕ СТАТЬИ ПУБЛИКУЮТСЯ В АВТОРСКОЙ РЕДАКЦИИ.**

**По вопросам публикации и финансовым вопросам** обращайтесь к администратору Зубковой Светлане Вадимовне  
E-mail: [info@nbpublish.com](mailto:info@nbpublish.com)  
или по телефону +7 (966) 020-34-36

## **Подробные требования к написанию аннотаций:**

Аннотация в периодическом издании является источником информации о содержании статьи и изложенных в ней результатах исследований.

Аннотация выполняет следующие функции: дает возможность установить основное

содержание документа, определить его релевантность и решить, следует ли обращаться к полному тексту документа; используется в информационных, в том числе автоматизированных, системах для поиска документов и информации.

Аннотация к статье должна быть:

- информативной (не содержать общих слов);
- оригинальной;
- содержательной (отражать основное содержание статьи и результаты исследований);
- структурированной (следовать логике описания результатов в статье);

Аннотация включает следующие аспекты содержания статьи:

- предмет, цель работы;
- метод или методологию проведения работы;
- результаты работы;
- область применения результатов; новизна;
- выводы.

Результаты работы описывают предельно точно и информативно. Приводятся основные теоретические и экспериментальные результаты, фактические данные, обнаруженные взаимосвязи и закономерности. При этом отдается предпочтение новым результатам и данным долгосрочного значения, важным открытиям, выводам, которые опровергают существующие теории, а также данным, которые, по мнению автора, имеют практическое значение.

Выводы могут сопровождаться рекомендациями, оценками, предложениями, гипотезами, описанными в статье.

Сведения, содержащиеся в заглавии статьи, не должны повторяться в тексте аннотации. Следует избегать лишних вводных фраз (например, «автор статьи рассматривает...», «в статье рассматривается...»).

Исторические справки, если они не составляют основное содержание документа, описание ранее опубликованных работ и общеизвестные положения в аннотации не приводятся.

В тексте аннотации следует употреблять синтаксические конструкции, свойственные языку научных и технических документов, избегать сложных грамматических конструкций.

### **Гонорары за статьи в научных журналах не начисляются.**

### **Цитирование или воспроизведение текста, созданного ChatGPT, в вашей статье**

Если вы использовали ChatGPT или другие инструменты искусственного интеллекта в своем исследовании, опишите, как вы использовали этот инструмент, в разделе «Метод» или в аналогичном разделе вашей статьи. Для обзоров литературы или других видов эссе, ответов или рефератов вы можете описать, как вы использовали этот инструмент, во введении. В своем тексте предоставьте prompt - командный вопрос, который вы использовали, а затем любую часть соответствующего текста, который был создан в ответ.

К сожалению, результаты «чата» ChatGPT не могут быть получены другими читателями, и хотя невосстановимые данные или цитаты в статьях APA Style обычно цитируются как личные сообщения, текст, сгенерированный ChatGPT, не является сообщением от человека.

Таким образом, цитирование текста ChatGPT из сеанса чата больше похоже на совместное использование результатов алгоритма; таким образом, сделайте ссылку на автора алгоритма записи в списке литературы и приведите соответствующую цитату в тексте.

Пример:

На вопрос «Является ли деление правого полушария левого полушария реальным или метафорой?» текст, сгенерированный ChatGPT, показал, что, хотя два полушария мозга в некоторой степени специализированы, «обозначение, что люди могут быть охарактеризованы как «левополушарные» или «правополушарные», считается чрезмерным упрощением и популярным мифом» (OpenAI, 2023).

### **Ссылка в списке литературы**

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].  
<https://chat.openai.com/chat>

Вы также можете поместить полный текст длинных ответов от ChatGPT в приложение к своей статье или в дополнительные онлайн-материалы, чтобы читатели имели доступ к точному тексту, который был сгенерирован. Особенno важно задокументировать созданный текст, потому что ChatGPT будет генерировать уникальный ответ в каждом сеансе чата, даже если будет предоставлен один и тот же командный вопрос. Если вы создаете приложения или дополнительные материалы, помните, что каждое из них должно быть упомянуто по крайней мере один раз в тексте вашей статьи в стиле APA.

Пример:

При получении дополнительной подсказки «Какое представление является более точным?» в тексте, сгенерированном ChatGPT, указано, что «разные области мозга работают вместе, чтобы поддерживать различные когнитивные процессы» и «функциональная специализация разных областей может меняться в зависимости от опыта и факторов окружающей среды» (OpenAI, 2023; см. Приложение А для полной расшифровки). .

### **Ссылка в списке литературы**

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].  
<https://chat.openai.com/chat> Создание ссылки на ChatGPT или другие модели и программное обеспечение ИИ

Приведенные выше цитаты и ссылки в тексте адаптированы из шаблона ссылок на программное обеспечение в разделе 10.10 Руководства по публикациям (Американская психологическая ассоциация, 2020 г., глава 10). Хотя здесь мы фокусируемся на ChatGPT, поскольку эти рекомендации основаны на шаблоне программного обеспечения, их можно адаптировать для учета использования других больших языковых моделей (например, Bard), алгоритмов и аналогичного программного обеспечения.

Ссылки и цитаты в тексте для ChatGPT форматируются следующим образом:

OpenAI. (2023). ChatGPT (версия от 14 марта) [большая языковая модель].  
<https://chat.openai.com/chat>

Цитата в скобках: (OpenAI, 2023)

Описательная цитата: OpenAI (2023)

Давайте разберем эту ссылку и посмотрим на четыре элемента (автор, дата, название и

источник):

Автор: Автор модели OpenAI.

Дата: Дата — это год версии, которую вы использовали. Следуя шаблону из Раздела 10.10, вам нужно указать только год, а не точную дату. Номер версии предоставляет конкретную информацию о дате, которая может понадобиться читателю.

Заголовок. Название модели — «ChatGPT», поэтому оно служит заголовком и выделено курсивом в ссылке, как показано в шаблоне. Хотя OpenAI маркирует уникальные итерации (например, ChatGPT-3, ChatGPT-4), они используют «ChatGPT» в качестве общего названия модели, а обновления обозначаются номерами версий.

Номер версии указан после названия в круглых скобках. Формат номера версии в справочниках ChatGPT включает дату, поскольку именно так OpenAI маркирует версии. Различные большие языковые модели или программное обеспечение могут использовать различную нумерацию версий; используйте номер версии в формате, предоставленном автором или издателем, который может представлять собой систему нумерации (например, Версия 2.0) или другие методы.

Текст в квадратных скобках используется в ссылках для дополнительных описаний, когда они необходимы, чтобы помочь читателю понять, что цитируется. Ссылки на ряд общих источников, таких как журнальные статьи и книги, не включают описания в квадратных скобках, но часто включают в себя вещи, не входящие в типичную рецензируемую систему. В случае ссылки на ChatGPT укажите дескриптор «Большая языковая модель» в квадратных скобках. OpenAI описывает ChatGPT-4 как «большую мультимодальную модель», поэтому вместо этого может быть предоставлено это описание, если вы используете ChatGPT-4. Для более поздних версий и программного обеспечения или моделей других компаний могут потребоваться другие описания в зависимости от того, как издатели описывают модель. Цель текста в квадратных скобках — кратко описать тип модели вашему читателю.

Источник: если имя издателя и имя автора совпадают, не повторяйте имя издателя в исходном элементе ссылки и переходите непосредственно к URL-адресу. Это относится к ChatGPT. URL-адрес ChatGPT: <https://chat.openai.com/chat>. Для других моделей или продуктов, для которых вы можете создать ссылку, используйте URL-адрес, который ведет как можно более напрямую к источнику (т. е. к странице, на которой вы можете получить доступ к модели, а не к домашней странице издателя).

## **Другие вопросы о цитировании ChatGPT**

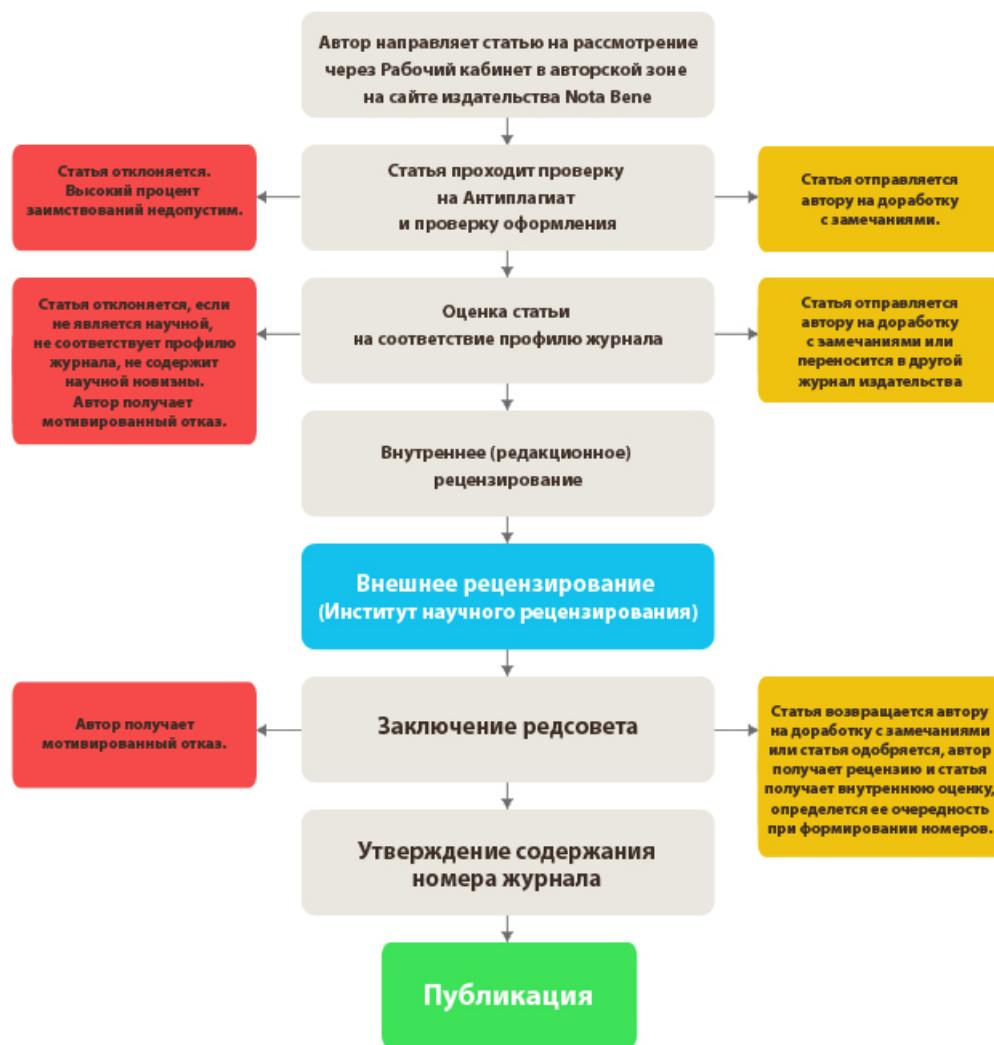
Вы могли заметить, с какой уверенностью ChatGPT описал идеи латерализации мозга и то, как работает мозг, не ссылаясь ни на какие источники. Я попросил список источников, подтверждающих эти утверждения, и ChatGPT предоставил пять ссылок, четыре из которых мне удалось найти в Интернете. Пятая, похоже, не настоящая статья; идентификатор цифрового объекта, указанный для этой ссылки, принадлежит другой статье, и мне не удалось найти ни одной статьи с указанием авторов, даты, названия и сведений об источнике, предоставленных ChatGPT. Авторам, использующим ChatGPT или аналогичные инструменты искусственного интеллекта для исследований, следует подумать о том, чтобы сделать эту проверку первоисточников стандартным процессом. Если источники являются реальными, точными и актуальными, может быть лучше прочитать эти первоисточники, чтобы извлечь уроки из этого исследования, и перефразировать или процитировать эти статьи, если применимо, чем использовать их интерпретацию модели.

Материалы журналов включены:

- в систему Российского индекса научного цитирования;
- отображаются в крупнейшей международной базе данных периодических изданий Ulrich's Periodicals Directory, что гарантирует значительное увеличение цитируемости;
- Всем статьям присваивается уникальный идентификационный номер Международного регистрационного агентства DOI Registration Agency. Мы формируем и присваиваем всем статьям и книгам, в печатном, либо электронном виде, оригинальный цифровой код. Префикс и суффикс, будучи прописанными вместе, образуют определяемый, цитируемый и индексируемый в поисковых системах, цифровой идентификатор объекта — digital object identifier (DOI).

[Отправить статью в редакцию](#)

### Этапы рассмотрения научной статьи в издательстве NOTA BENE.



## Содержание

Гибадуллин Р.Ф., Викторов И.В. Неоднозначность результатов при использовании методов класса Parallel в рамках исполняющей среды .NET Framework	1
Антонова П.В. Принципы разработки систем массового обслуживания с ограниченной очередью на платформе .NET	15
Марченко А.Г., Щемелинин Д.А. Разработка автоматизированной системы тестирования облачного сервиса развертывания виртуальных машин с использованием современных средств мониторинга	29
Юсупов Б.З. Разработка учебного стенда охранно-пожарной системы для обучения студентов	40
Лаптев М.В., Янчус В.Э., Лаптев В.В. Выявление зон интереса пользователя при визуализации данных с использованием метода ай-трекинга	49
Англоязычные метаданные	63

## Contents

Gibadullin R.F., Viktorov I.V. Ambiguous Results when Using Parallel Class Methods within the .NET Framework	1
Antonova P.V. Principles of developing a limited-queue mass service system on the .NET platform	15
Marchenko A., Shchemelinin D. Development of an automated system for testing a cloud service for deploying virtual machines using modern monitoring tools	29
Yusupov B.Z. Development of the security and fire system training stand for student training	40
Laptev M.V., Yanchus V.E., Laptev V.V. Eye-tracking detection of the area of interest in data visualization	49
Metadata in english	63

**Программные системы и вычислительные методы***Правильная ссылка на статью:*

Гибадуллин Р.Ф., Викторов И.В. — Неоднозначность результатов при использовании методов класса Parallel в рамках исполняющей среды .NET Framework // Программные системы и вычислительные методы. – 2023. – № 2. DOI: 10.7256/2454-0714.2023.2.39801 EDN: IDMOCV URL: [https://nbpublish.com/library\\_read\\_article.php?id=39801](https://nbpublish.com/library_read_article.php?id=39801)

**Неоднозначность результатов при использовании методов класса Parallel в рамках исполняющей среды .NET Framework****Гибадуллин Руслан Фаршатович**

ORCID: 0000-0001-9359-911X

кандидат технических наук

доцент кафедры компьютерных систем Казанского национального исследовательского технического университета им. А.Н. Туполева-КАИ (КНИТУ-КАИ)

420015, Россия, республика Татарстан, г. Казань, ул. Большая Красная, 55, каб. 432

**✉ rfgibadullin@kai.ru****Викторов Иван Владимирович**

аспирант кафедры компьютерных систем Казанского национального исследовательского технического университета им. А.Н. Туполева - КАИ (КНИТУ-КАИ)

420015, Россия, республика Татарстан, г. Казань, ул. Большая Красная, 55, оф. 432

**✉ victorov.i.vl@yandex.ru**[Статья из рубрики "Языки программирования"](#)**DOI:**

10.7256/2454-0714.2023.2.39801

**EDN:**

IDMOCV

**Дата направления статьи в редакцию:**

17-02-2023

**Дата публикации:**

08-03-2023

**Аннотация:** Параллельное программирование – это способ написания программ, которые могут выполняться параллельно на нескольких процессорах или ядрах. Это позволяет программам обрабатывать большие объемы данных или выполнить более сложные вычисления за приемлемое время, чем это было бы возможно на одном процессоре. Преимущества параллельного программирования: увеличение производительности, распределение нагрузки, обработка больших объемов данных,

улучшение отзывчивости, увеличение надежности. В целом, параллельное программирование имеет множество преимуществ, которые могут помочь улучшить производительность и надежность программных систем, особенно в условиях растущей сложности вычислительных задач и объемов данных. Однако параллельное программирование также может иметь свои сложности, связанные с управлением синхронизацией, гонками данных и другими аспектами, которые требуют дополнительного внимания и опыта со стороны программиста. В ходе тестирования параллельных программ можно получить неоднозначные результаты. Например, это может происходить, когда мы оптимизируем объединение данных типа float или double посредством методов For или ForEach класса Parallel. Подобное поведение программы заставляет усомниться в потокобезопасности написанного кода. Такой вывод может быть неправильным и преждевременным. Статья раскрывает возможную причину неоднозначности результатов, получаемых параллельной программой, и предлагает лаконичное решение вопроса.

### **Ключевые слова:**

Параллельное программирование, Язык программирования СиШарп, Многопоточность, Ошибки округления, Неоднозначность результатов, Потоковая безопасность, Вещественные числа, Тип decimal, Платформа NET, Класс Parallel

### **Введение**

В научных и инженерных расчетах точность является важным критерием, определяющим достоверность результатов. При проведении математических операций и вычислений на ЭВМ (электронно-вычислительной машине) возникает необходимость округления чисел, что может приводить к погрешностям. Поэтому снижение погрешности расчетов, связанной с использованием округлений, является важной задачей в научной и инженерной практике.

Округление чисел происходит в результате ограничения количества значащих цифр после запятой, что неизбежно приводит к потере точности. В зависимости от задачи и требуемой точности необходимо выбирать правильный способ округления чисел. Например, при округлении до целого числа, некоторые десятичные дроби будут потеряны, а при округлении до определенного числа знаков после запятой произойдет потеря точности из-за отбрасывания оставшихся десятичных цифр.

Снижение погрешности расчетов может быть достигнуто путем использования более точных методов вычислений, таких как метод Гаусса или методы численного интегрирования. Кроме того, для уменьшения погрешности можно применять различные алгоритмы и техники вычислений, которые позволяют уменьшить потерю точности при округлении чисел.

Снижение погрешности расчетов особенно важно в случаях, когда требуется высокая точность, например, при использовании методов решения систем линейных уравнений [1], при использовании численных методов дифференцирования и интегрирования [2], при моделировании физических явлений [3], при вычислении математических функций [4], при вычислении процентных ставок или доходности инвестиций [5,6], при использовании алгоритмов машинного обучения и искусственного интеллекта [7]. В этих

случаях даже небольшие погрешности могут привести к серьезным ошибкам и неверным результатам, которые могут негативно повлиять на принимаемые решения.

Таким образом, снижение погрешности расчетов, связанной с использованием округлений, является важным фактором в научной и инженерной практике, поскольку позволяет получать более точные результаты вычислений и уменьшить вероятность ошибок и неверных выводов.

В современных вычислительных системах все более широко используются технологии параллельного программирования для увеличения производительности. Однако при использовании параллельных алгоритмов возникают дополнительные сложности [8,9], связанные с необходимостью обмена данными между различными процессами, а также с синхронизацией операций.

В условиях параллельного программирования использование округления чисел может приводить к возникновению дополнительных погрешностей, связанных с несогласованностью округления между различными процессами. Это может приводить к ошибкам в вычислениях, что особенно важно в случаях, когда требуется высокая точность вычислений.

Поэтому снижение погрешности расчетов, связанной с использованием округлений, имеет практическую значимость в технологиях параллельного программирования. Для уменьшения погрешностей в параллельных алгоритмах необходимо применять специальные алгоритмы и техники. В частности, это актуально в ходе параллельного агрегирования локальных значений, что определяет предмет исследования данной статьи. В демонстрационных целях взята задача – вычисление квадратных корней, которое часто используется в различных областях науки и техники, где необходимы точные вычисления.

Вычисление суммы квадратных корней может быть полезным в решении многих задач, таких как:

- Определение расстояния между двумя точками в n-мерном пространстве.
- Вычисление нормы вектора в n-мерном пространстве.
- Расчет площади кривой на плоскости.
- Определение времени, за которое тело падает на землю с заданной высоты при ускорении свободного падения.
- Расчет стоимости доставки груза в зависимости от расстояния и веса груза.
- Определение времени, за которое свет проходит расстояние между двумя точками в пространстве.
- Оценка времени, необходимого для прохождения тестов и обучения в машинном обучении.
- Оценка скорости работы компьютерных программ и алгоритмов.

### **Неоднозначность результатов в ходе параллельной агрегации локальных значений**

Методы Parallel.For и Parallel.ForEach предлагают набор перегруженных версий, которые работают с аргументом обобщенного типа по имени TLocal. Такие перегруженные версии призваны помочь оптимизировать объединение данных из циклов с интенсивными итерациями. Ниже представлена перегруженная версия метода Parallel.For, которую далее мы будем использовать в предметном анализе.

```
public static ParallelLoopResult For(
    int fromInclusive,
    int toExclusive,
    Func localInit,
    Func<int, ParallelLoopState, TLocal, TLocal> body,
    Action localFinally);
```

Где:

- `fromInclusive` – начальный индекс, включительно.
- `toExclusive` – конечный индекс, не включительно.
- `localInit` – делегат функции, который возвращает начальное состояние локальных данных для каждой задачи.
- `body` – делегат, который вызывается один раз за итерацию.
- `localFinally` – делегат, который выполняет финальное действие с локальным результатом каждой задачи.
- `TLocal` – тип данных, локальных для потока.
- Возвращаемый объект – структура (`ParallelLoopResult`), в которой содержатся сведения о выполненной части цикла.

Применим данный метод на практике, чтобы просуммировать квадратные корни чисел от 1 до  $10^7$ .

```
object locker = new object();

double grandTotal = 0;

Parallel.For(1, 10000000,
    () => 0.0, // Initialize the local value.

    (i, state, localTotal) => // Body delegate. Notice that it

        localTotal + Math.Sqrt(i), // returns the new local total.

    localTotal => // Add the local

        { lock (locker) grandTotal += localTotal; } // to the master value.

);

Console.WriteLine(grandTotal);
```

Данное решение может выдавать неоднозначный результат, например:

- 21081849486,4431;
- 21081849486,4428;
- 21081849486,4429.

Таким образом, по итогам запусков программы результаты вычислений могут отличаться в 3 или 4 знаке после запятой, что является недопустимым при решении таких задач, как:

- Научные и инженерные расчеты, связанные с проектированием и тестированием новых технологий и устройств, где даже небольшие погрешности могут привести к неправильным выводам.
- Разработка и анализ финансовых моделей, где точность результатов может оказать значительное влияние на принимаемые инвестиционные решения.
- Работа с большими наборами данных в машинном обучении и анализе данных, где точность результатов может быть важна для получения правильных выводов и прогнозов.
- Вычисление физических и химических констант, таких как постоянная Планка или постоянная Больцмана, которые используются в широком диапазоне научных и инженерных расчетов.
- Исследования в области астрономии и космологии, где высокая точность результатов может быть важна для определения физических характеристик звезд, планет и галактик.

Причина неоднозначности результатов является комплексной. Во-первых, имеют место ошибки округления вещественных чисел. Во-вторых, выполнение делегата, отвечающего за формирование локального накопителя, в потоках пула носит порциональный характер. Рассмотрим и то и другое более детально.

Типы `float` и `double` внутренне представляют числа в двоичной форме. По указанной причине точно представляются только числа, которые могут быть выражены в двоичной системе счисления. На практике это означает, что большинство литералов с дробной частью (которые являются десятичными) не будут представлены точно. Например:

```
float x = 0.1f; // Не точно 0.1
Console.WriteLine (x + x + x + x + x + x + x + x + x); // 1.0000001
```

Именно потому типы `float` и `double` не подходят для финансовых вычислений. В противоположность им тип `decimal` работает в десятичной системе счисления, так что он способен точно представлять дробные числа вроде 0.1, выражимые в десятичной системе (а также в системах счисления с основаниями-множителями 10 – двоичной и пятеричной). Поскольку вещественные литералы являются десятичными, тип `decimal` может точно представлять такие числа, как 0.1. Тем не менее, ни `double`, ни `decimal` не могут точно представлять дробное число с периодическим десятичным представлением:

```
decimal m = 1M / 6M; // 0.1666666666666666666666667M
double d = 1.0 / 6.0; // 0.1666666666666666
```

Это приводит к накапливающимся ошибкам округления:

```
decimal notQuiteWholeM = m+m+m+m+m+m; // 1.0000000000000000000000000002M
```

```
double notQuiteWholeD = d+d+d+d+d+d; // 0.9999999999999999
```

которые нарушают работу операций эквивалентности и сравнения:

```
Console.WriteLine (notQuiteWholeM == 1M); // False
```

```
Console.WriteLine (notQuiteWholeD < 1.0); // True
```

Ниже в таблице 1 представлен обзор отличий между типами `double` и `decimal`.

Таблица 1. Отличия между типами double и decimal [9]

Характеристика	double	decimal
Внутреннее представление	Двоичное	Десятичное
Десятичная точность	15–16 значащих цифр	28–29 значащих цифр
Специальные значения	+0, -0, положительная (отрицательная) бесконечность и NaN	Отсутствуют
Скорость обработки	Присущая процессору	Не присущая процессору (примерно в 10 раз медленнее, чем в случае double)

Раскроем тип decimal более детально, чтобы ответить на вопрос, почему обработка данных типа decimal не является присущей процессору.

Двоичное представление decimal числа состоит из 1-битового знака, 96-битового целого числа и коэффициента масштабирования, используемого для деления целочисленного числа и указания его части десятичной дроби. Коэффициент масштабирования неявно представляет собой число 10, возведенное в степень в диапазоне от 0 до 28. Таким образом, bits – это массив из четырех элементов, состоящий из 32-разрядных целых чисел со знаком:

- bits\_0, bits\_1 и bits\_2 содержат низкие, средние и высокие 32 биты 96-разрядного целого числа.
- bits\_3:
  - 0-15 не используются;
  - 16-23 (8 бит) содержат экспоненту от 0 до 28, что указывает на степень 10 для деления целочисленного числа;
  - 24-30 не используются;
  - 31 содержит знак (0 означает положительное значение, а 1 – отрицательное).

Для вычисления суммы квадратных корней с использованием типа данных decimal некоторые параметры представления числа, которые могут быть значимыми, включают:

- Количество знаков в дробной части – чем больше знаков, тем точнее будет представление квадратных корней, и тем выше точность вычисления суммы.
- Режим округления – если результат вычисления имеет много знаков после запятой, режим округления может повлиять на точность вычисления суммы. Режим округления может влиять на округление до ближайшего четного числа или до ближайшего нечетного числа, что может повлиять на точность.
- Размер памяти – чем больше размер памяти, тем больше знаков можно хранить, и тем выше точность вычисления суммы.
- Диапазон значений – это может быть не столь значимо, так как в задаче вычисления суммы квадратных корней используются только положительные значения.

В целом, для вычисления суммы квадратных корней с использованием типа данных decimal, параметры, которые являются наиболее значимыми, это количество знаков в дробной части и режим округления. Эти параметры могут быть выбраны на основе требуемой точности и скорости выполнения вычислений.

Разбиение на основе порций работает путем предоставления каждому рабочему потоку

возможности периодически захватывать из входной последовательности небольшие "порции" элементов с целью их обработки [9]. Например (рис. 1), инфраструктура Parallel LINQ начинает с выделения очень маленьких порций (один или два элемента за раз) и затем по мере продвижения запроса увеличивает размер порции: это гарантирует, что небольшие последовательности будут эффективно распараллеливаться, а крупные последовательности не приведут к чрезмерным циклам полного обмена. Если рабочий поток получает "простые" элементы (которые обрабатываются быстро), то в конечном итоге он сможет получить больше порций. Такая система сохраняет каждый поток одинаково занятым (а процессорные ядра "сбалансированными"); единственный недостаток состоит в том, что извлечение элементов из разделяемой входной последовательности требует синхронизации – и в результате могут появиться некоторые накладные расходы и состязания.

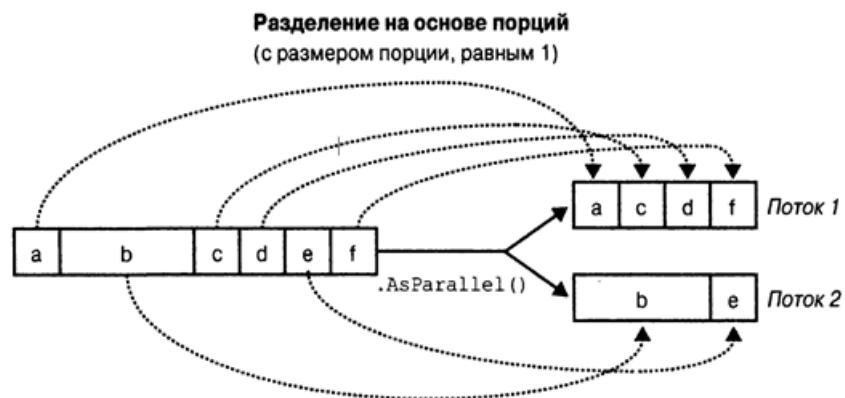


Рисунок 1. Разделение на основе порций [9]

Метод For класса Parallel работает схожим образом, разница лишь в том, что вместо элемента входной последовательности выступает номер итерации, который как правило учитывается при выполнении тела цикла (точнее делегата типа Action). Реализация разделения основана на механизме разбиения на порции, при котором размер порции потенциально увеличивается в случае положительной динамики обработки итераций. Такой подход помогает обеспечить качественную балансировку нагрузки при небольшом количестве итераций и минимизировать число монопольных блокировок (в ходе назначения диапазонов номеров итераций для рабочих потоков) при их большом количестве. При этом обеспечивается, чтобы большинство итераций потока было сосредоточено в одной и той же области итерационного пространства для достижения высокой локальности кэша.

### Исследование метода Parallel.For для детализации причины неоднозначности конечного результата

Реализация метода For сложна и требует детального рассмотрения, которое выходит за рамки данной статьи. Тем не менее отметим некоторые моменты программной реализации метода Parallel.For с аргументом обобщенного типа.

```
public static ParallelLoopResult For (int fromInclusive, int toExclusive, Func localInit, Func<int, ParallelLoopState, TLocal, TLocal> body, ...) {
    ...
    return ForWorker(fromInclusive, toExclusive, s_defaultParallelOptions,
        null, null, body, localInit, localFinally);
}
```

```
    }

    private static ParallelLoopResult ForWorker (int fromInclusive, int toExclusive,
ParallelOptions parallelOptions, Action body, ...) {

    ...

    rootTask = new ParallelForReplicatingTask(parallelOptions, delegate {

        if (rangeWorker.FindNewWork32(
            out var nFromInclusiveLocal,
            out var nToExclusiveLocal) &&
        !sharedPStateFlags.ShouldExitLoop(nFromInclusiveLocal)) {

            ...

            do {
                if (body != null) {

                    for (int i = nFromInclusiveLocal; i < nToExclusiveLocal; i++) {

                        if (sharedPStateFlags.LoopStateFlags != ParallelLoopStateFlags.PLS_NONE
                            && sharedPStateFlags.ShouldExitLoop()) {

                            break;
                        }

                        body(i);
                    }
                }
            }
        }

        while (rangeWorker.FindNewWork32(out nFromInclusiveLocal, out nToExclusiveLocal)
        ...);

        ...
    }
}, creationOptions, internalOptions);

rootTask.RunSynchronously(parallelOptions.EffectiveTaskScheduler);

rootTask.Wait();

...
}

internal ParallelForReplicatingTask(...) {

    m_replicationDownCount = parallelOptions.EffectiveMaxConcurrencyLevel;
```

...

}

Метод `rootTask.RunSynchronously` запускает исполнение задач в рабочих потоках пула, при этом число задач задается свойством `parallelOptions.EffectiveMaxConcurrencyLevel`. Метод `FindNewWork32` определяет рабочий диапазон для каждого потока пула. В представленном коде можно увидеть, что выполнение любой задачи не ограничивается выполнением первоначально определенного диапазона, потоки пула продолжают работу для вновь задаваемых диапазонов в операторе `while`.

Проведем детализацию работы метода `Parallel.For` с аргументом обобщенного типа на ранее представленном примере по суммированию квадратных корней чисел, расширив код следующим образом.

```
object locker = new object();

double grandTotal = 0;

ConcurrentBag<(int?, double)> cb1 = new ConcurrentBag<(int?, double)>();
ConcurrentDictionary<int?, long> cd = new ConcurrentDictionary<int?, long>();
ConcurrentBag<(int?, int)> cb2 = new ConcurrentBag<(int?, int)>();

var time = Stopwatch.StartNew();

time.Start();

Parallel.For(1, 1000,
    () => { return 0.0; },
    (i, state, localTotal) =>
{
    cb1.Add((Task.CurrentId, localTotal));
    if (!cd.ContainsKey(Task.CurrentId)) cd[Task.CurrentId] = time.ElapsedTicks;
    cb2.Add((Task.CurrentId, i));
    return localTotal + Math.Sqrt(i);
},
    localTotal =>
{
    lock (locker) grandTotal += localTotal;
});

cb1.GroupBy(_ => _.Item1).Select(_ => new
{
    TaskId = _.Key,
```

```

Iterations = _.Count(),
StartTime = cd[_.Key]

}).OrderBy(_ => _.StartTime).Dump();

var query = cb2.OrderBy(_ => _.Item2).GroupBy(_ => _.Item1, _ => _.Item2);

foreach (var grouping in query)

{
    Console.WriteLine("TaskId: " + grouping.Key);

    var r = grouping.GetEnumerator();

    int? i = null;

    bool onlyOne = true;

    foreach (int iteration in grouping)

    {
        if (i == null)

            Console.Write("{ " + $"{{iteration}}");

        else

        {

            if (iteration - i != 1)

                Console.Write(",..." + i + "}, {" + iteration);

            onlyOne = false;
        }

        i = iteration;
    }

    if (onlyOne) Console.WriteLine("}");

    else Console.WriteLine(",..." + i + "}");
}

```

Программный код позволяет учесть:

- идентификатор каждой задачи TaskId;
- количество итераций выполненное в рамках каждой задачи Iterations;
- StartTime – время начала работы каждой задачи, выраженное в тиках посредством класса Stopwatch (один тик является меньше одной микросекунды);
- диапазоны номеров обработанных итераций каждой задачи.

Например, по результатам работы программы на машине, способной выполнять 8 потоков параллельно на аппаратном уровне, можно получить следующие показатели TaskId,

Iterations, StartTime (табл. 2). Диапазоны номеров обработанных итераций представлены в таблице 3.

Таблица 2. Показатели TaskId, Iterations, StartTime после завершения метода Parallel.For

<b>TaskId</b>	<b>Iterations</b>	<b>StartTime</b>
20	205	54568
21	1	54597
16	1	54709
22	159	54846
18	204	54986
24	161	55689
17	111	55689
15	1	55821
19	156	55880

Таблица 3. Диапазоны номеров обработанных итераций каждой задачи

<b>Идентификатор задачи</b>	<b>Диапазоны номеров обработанных итераций</b>
24	{1,...,31}, {40,...,55}, {88,...,103}, {120,...,124}, {142,...,173}, {206,...,221}, {266,...,281}, {330,...,345}, {484,...,496}
22	{32,...,39}, {56,...,87}, {104,...,119}, {126,...,141}, {174,...,189}, {222,...,237}, {250,...,265}, {298,...,313}, {346,...,361}, {993,...,999}
15	{125}
20	{190,...,205}, {238,...,248}, {282,...,297}, {314,...,329}, {362,...,372}, {858,...,992}
16	{249}
17	{373,...,483}
18	{497,...,620}, {716,...,731}, {746,...,761}, {778,...,793}, {810,...,825}, {842,...,857}
19	{621,...,715}, {732,...,744}, {762,...,777}, {794,...,809}, {826,...,841}
21	{745}

По результатам работы программы можно увидеть, что рабочие диапазоны различны. Некоторые задачи состоят из единственной итерации. Но это не является недостатком алгоритма по которому реализован исследуемый метод, а следствием того, что обработка одной итерации представляет собой нетрудоемкую с вычислительной точки зрения процедуры. Так, например, если в целевой метод делегата, представляющий четвертый параметр метода Parallel.For, добавить строки:

```
for (int k = 0; k < 1000000; k++)
```

```
    Math.Sqrt(k);
```

тем самым существенно усложнив обработку каждой итерации цикла, то можно получить равномерное распределение диапазонов по задачам (табл. 4).

Таблица 4. Показатели TaskId, Iterations, StartTime после усложнения обработки каждой итерации цикла

<b>TaskId</b>	<b>Iterations</b>	<b>StartTime</b>
13	79	50828
10	63	50849
12	79	51226
16	79	51698
15	79	52224
11	95	52788
19	108	53181
17	84	53640
14	79	53976
20	32	3263706
21	32	3355186
22	32	3462087
23	32	3543335
24	29	3562197
25	39	3575327
26	29	3639235
27	29	3797790

Таким образом, результаты апробации метода Parallel.For показывают, что в ходе повторных запусков программы с данным методом создается различное число задач и рабочих диапазонов, отличных друг от друга. Данное поведение программы при обработке данных типа float и double приводит к неоднозначности результата выполнения делегата localFinally, определяющего финальное действие с локальным результатом каждой задачи.

Чтобы обеспечить высокую точность проводимых вычислений, следует обеспечить переход на тип decimal:

```
object locker = new object();

decimal grandTotal = 0;

Parallel.For(1, 10000000,
    () => (decimal)0,
    (i, state, localTotal) =>
    localTotal + (decimal)Math.Sqrt(i),
    localTotal =>
    { lock (locker) grandTotal += localTotal; });
grandTotal.Dump();
```

Такой переход сопряжен с накладными расходами по быстродействию программы (при

вычислении суммы квадратных корней чисел от 1 до  $10^7$  на четырехъядерном процессоре Intel Core i5 9300H время выполнения составляет приблизительно 0,260 мсек. при использовании типа decimal, в то время как при использовании типа double это занимает лишь 0,02 мсек.) и может быть неоправданным из-за отсутствия необходимости в результатах повышенной точности. Однако взамен на выходе обеспечивается однозначный результат: 21081849486,44249240077608.

## **Заключение**

Точность вычислений имеет большое значение в параллельном программировании, особенно при решении сложных научных задач или задач, связанных с обработкой больших объемов данных. В параллельном программировании, когда вычисления выполняются на многопроцессорных системах, точность вычислений может быть нарушена из-за нескольких причин:

- Проблемы с синхронизацией данных – когда несколько процессоров обрабатывают одни и те же данные, возможно необходимо синхронизировать вычисления между процессорами, чтобы результаты были согласованы и точны.
- Проблемы с коммуникацией – когда процессы обмениваются данными, возможны проблемы с задержками и утечками данных, которые могут привести к ошибкам в вычислениях.
- Проблемы с точностью округления – некоторые операции вычислений могут приводить к потере точности, особенно при использовании типов данных с плавающей запятой. При распределенных вычислениях на многопроцессорных системах потеря точности может быть усугублена.

В статье особое вниманиеделено последней причине на примере использования метода For класса Parallel в рамках исполняющей среды .NET Framework. Проведен анализ метода Parallel.For, исследована работа параллельной программы вычисления суммы квадратных корней заданного диапазона чисел. Выявлено, что причина неоднозначности результатов вычислений является комплексной и связана с ошибками округления вещественных чисел и порциональной обработкой диапазона чисел в потоках пула. Для уменьшения этой неоднозначности целесообразно использовать более точные типы данных, также представляет интерес применения алгоритмов, которые уменьшают влияние ошибок округления и улучшают согласованность результатов между потоками [\[10\]](#).

## **Библиография**

1. X. Fan, R. -a. Wu, P. Chen, Z. Ning and J. Li, "Parallel Computing of Large Eigenvalue Problems for Engineering Structures," 2011 International Conference on Future Computer Sciences and Application, Hong Kong, China, 2011, pp. 43-46, doi: 10.1109/ICFCSA.2011.16.
2. Xuehui Chen, Liang Wei, Jizhe Sui, Xiaoliang Zhang and Liancun Zheng, "Solving fractional partial differential equations in fluid mechanics by generalized differential transform method," 2011 International Conference on Multimedia Technology, Hangzhou, 2011, pp. 2573-2576, doi: 10.1109/ICMT.2011.6002361.
3. R. Landau, "Computational Physics: A Better Model for Physics Education?," in Computing in Science & Engineering, vol. 8, no. 5, pp. 22-30, Sept.-Oct. 2006, doi: 10.1109/MCSE.2006.85.
4. H. Ali, A. Doucet and D. I. Amshah, "GSR: A New Genetic Algorithm for Improving Source and Channel Estimates," in IEEE Transactions on Circuits and Systems I:

- Regular Papers, vol. 54, no. 5, pp. 1088-1098, May 2007, doi:  
10.1109/TCSI.2007.893507.
5. D. -M. Zhu, J. -w. Gu, F. -H. Yu, W. -K. Ching and T. -K. Siu, "How correlation risk in basket credit derivatives might be priced and managed?," in IMA Journal of Management Mathematics, vol. 32, no. 2, pp. 195-219, April 2020, doi:  
10.1093/imaman/dpaa013.
  6. J. Xu and Y. Shi, "Financial Leasing, Optimal Financial Structure and Economic Growth: An Analysis Based on Financial Inclusion Perspective," 2019 International Conference on Economic Management and Model Engineering (ICEMME), Malacca, Malaysia, 2019, pp. 147-150, doi: 10.1109/ICEMME49371.2019.00038.
  7. S. N. Cherny and R. F. Gibadullin, "The Recognition of Handwritten Digits Using Neural Network Technology," 2022 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russian Federation, 2022, pp. 965-970, doi: 10.1109/ICIEAM54945.2022.9787104.
  8. Albahari, Joseph. C# 10 in a Nutshell. " O'Reilly Media, Inc.", 2022.
  9. Гибадуллин Р.Ф. Потокобезопасные вызовы элементов управления в обогащенных клиентских приложениях // Программные системы и вычислительные методы. – 2022. – № 4. – С. 1-19.
  10. Rump, S.M. Fast and Parallel Interval Arithmetic. BIT Numerical Mathematics 39, 534–554 (1999). <https://doi.org/10.1023/A:1022374804152>.

## **Результаты процедуры рецензирования статьи**

*Рецензия скрыта по просьбе автора*

Программные системы и вычислительные методы

Правильная ссылка на статью:

Антонова П.В. — Принципы разработки систем массового обслуживания с ограниченной очередью на платформе .NET // Программные системы и вычислительные методы. — 2023. — № 2. DOI: 10.7256/2454-0714.2023.2.43403 EDN: HIXNHH URL: [https://nbpublish.com/library\\_read\\_article.php?id=43403](https://nbpublish.com/library_read_article.php?id=43403)

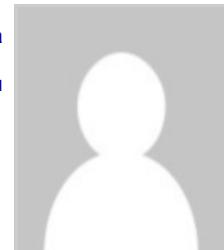
## Принципы разработки систем массового обслуживания с ограниченной очередью на платформе .NET

Антонова Полина Валерьевна

старший преподаватель кафедры интеллектуальных систем и управления информационными ресурсами Казанского национального исследовательского технологического университета

420015, Россия, Республика Татарстан, г. Казань, ул. Карла Маркса, 68

✉ valerevna.p@inbox.ru



Статья из рубрики "Автоматизированные системы управления технологическими процессами"

### DOI:

10.7256/2454-0714.2023.2.43403

### EDN:

HIXNHH

### Дата направления статьи в редакцию:

21-06-2023

### Дата публикации:

29-06-2023

**Аннотация:** Статья является результатом тщательного исследования, направленного на изучение принципов создания, моделирования и практического использования систем массового обслуживания (СМО) с ограниченным размером очереди, основанных на технологической платформе .NET и реализованных с применением языка программирования C#. В ходе изложения материала особое внимание уделяется как одноканальным, так и многоканальным системам. Во вступительной части статьи рассмотрены фундаментальные концепции теории массового обслуживания. Обсуждаются основные характеристики систем, такие как системы с фиксированной длиной очереди, одноканальные и многоканальные системы с вероятностью отказа в обслуживании, системы с неограниченным и ограниченным временем ожидания, замкнутые системы, а также многоканальные системы, в которых имеется взаимодействие между каналами. Представлены подробные примеры программного кода на языке C#, иллюстрирующие структуры классов, применяемые для моделирования как одноканальных, так и многоканальных СМО. Рассматриваются конкретные сценарии

использования представленных структур в рамках моделирования СМО. В статье освещается практическое применение систем массового обслуживания в решении реальных задач. На примерах из банковской сферы и управления трафиком в телекоммуникациях демонстрируется, как СМО могут способствовать оптимизации времени ожидания и эффективному управлению ресурсами. Предлагаются перспективы дальнейших исследований в области систем массового обслуживания. Учитывая важность СМО в различных отраслях, таких как банковское дело, телекоммуникации, логистика и многие другие, высокая актуальность темы обусловлена необходимостью поиска новых подходов и методов для повышения эффективности и оптимизации этих систем. В целом статья представляет собой ценный исследовательский материал для специалистов, занимающихся вопросами моделирования и практического применения систем массового обслуживания.

### **Ключевые слова:**

системы массового обслуживания, моделирование, ограниченная очередь, одноканальные системы, многоканальные системы, Си Шарп, платформа NET, оптимизация, программные системы, вычислительные методы

### **Введение**

Теория массового обслуживания играет ключевую роль в анализе и оптимизации процессов, связанных с обслуживанием потоков заявок в различных областях, таких как телекоммуникации, транспорт, здравоохранение и производство [1, 2]. Рассмотрим основные понятия теории массового обслуживания и типы систем обслуживания.

Системы с фиксированной очередью: в системах с фиксированной очередью количество мест в очереди ограничено. Задачи упорядочения касаются определения порядка, в котором заявки будут обслуживаться. Основные стратегии упорядочения включают FIFO (First In, First Out), LIFO (Last In, First Out) и приоритетное обслуживание.

Одноканальные и многоканальные системы с отказами: в одноканальных системах присутствует только один канал обслуживания. Если заявка приходит, когда канал занят, она может быть отклонена или поставлена в очередь. Многоканальные системы имеют несколько каналов обслуживания, что позволяет обрабатывать несколько заявок одновременно. Если все каналы заняты, заявка также может быть отклонена или поставлена в очередь.

Системы с неограниченным и ограниченным ожиданием обслуживания: в системах с неограниченным ожиданием заявки могут ожидать обслуживания неопределенно долго. В системах с ограниченным ожиданием существуют ограничения по длине очереди или времени ожидания. Если заявка не может быть обслужена в пределах этих ограничений, она отклоняется.

Замкнутые системы: в замкнутых системах заявки возвращаются в систему после обслуживания. Это часто встречается в сценариях, где ресурсы циркулируют в системе и могут быть повторно использованы.

Многоканальные системы с взаимодействием между каналами: в некоторых многоканальных системах каналы могут взаимодействовать между собой, динамически распределяя ресурсы и обмениваясь информацией для повышения эффективности

обслуживания. Это важно в сложных сетевых средах, где различные каналы могут обрабатывать разные типы заявок и необходима гибкая реакция на изменения в рабочей нагрузке.

Для анализа систем массового обслуживания и определения оптимальных стратегий обслуживания используются различные математические модели и методы, такие как марковские процессы, теория очередей и симуляционное моделирование. Марковские процессы позволяют описать случайные процессы с дискретными состояниями, теория очередей фокусируется на характеристиках потоков заявок и каналов обслуживания, а симуляционное моделирование позволяет изучать поведение системы в различных условиях.

Знание и понимание теории массового обслуживания имеют высокую практическую значимость. Например, в телекоммуникациях это помогает оптимизировать сетевые ресурсы, в здравоохранении – уменьшить время ожидания пациентов, а в производстве – повысить производительность и снизить издержки [\[3, 4\]](#).

Системы массового обслуживания с ограниченной очередью являются важным подклассом в теории массового обслуживания. Они характеризуются наличием верхнего предела на количество заявок, которые могут ожидать обслуживания в очереди. Как только этот предел достигнут, все последующие заявки будут отклонены до тех пор, пока место в очереди не освободится [\[5\]](#).

Представим аспекты систем массового обслуживания с ограниченной очередью:

- Очередь: очередь – это структура данных, которая хранит заявки, ожидающие обслуживания. В системах с ограниченной очередью количество заявок в очереди не может превышать определенный уровень, известный как вместимость очереди.
- Стратегии обслуживания: системы с ограниченной очередью могут использовать различные стратегии обслуживания, такие как FIFO (первым пришел, первым обслужен), LIFO (последним пришел, первым обслужен) или приоритетное обслуживание, где некоторые заявки могут иметь приоритет перед другими.
- Модель M/M/c/K: одна из наиболее известных моделей систем с ограниченной очередью – это модель M/M/c/K. В этой модели, интервалы времени между поступлением заявок и время обслуживания предполагается экспоненциальным (обозначается буквой 'M'), 'c' обозначает количество каналов обслуживания, а 'K' - общее количество мест в системе (количество мест в очереди плюс количество каналов).
- Анализ производительности: для анализа производительности систем с ограниченной очередью используются различные показатели, такие как вероятность блокировки (вероятность отказа новой заявке), среднее время ожидания в очереди и среднее число заявок в системе [\[6\]](#).
- Применение: системы с ограниченной очередью широко применяются в различных областях, таких как телекоммуникации (для управления вызовами), сетевые маршрутизаторы (для управления пакетами данных), здравоохранение (для управления потоком пациентов) и производство (для оптимизации производственных линий). Например, в телекоммуникациях системы с ограниченной очередью могут использоваться для управления входящими звонками в колл-центре. Когда все операторы заняты и очередь достигла своей максимальной вместимости, последующие звонки могут быть перенаправлены на автоматическую систему голосового ответа или получать сообщение

об отказе. В сфере здравоохранения системы с ограниченной очередью могут быть использованы в отделениях экстренной помощи, где пациенты могут быть приняты в соответствии с уровнем срочности их состояния. Когда очередь достигает своей максимальной вместимости, пациенты с менее срочными состояниями могут быть направлены в другие медицинские учреждения.

- Вызовы и стратегии оптимизации: одним из ключевых вызовов в системах с ограниченной очередью является балансировка между уровнем обслуживания и ресурсами. Слишком маленькая очередь может привести к частым отказам, тогда как слишком большая очередь может приводить к неэффективному использованию ресурсов. Оптимизация таких систем часто сводится к выбору подходящего размера очереди и числа каналов обслуживания, а также выбору эффективной стратегии обслуживания. Симуляционное моделирование и аналитические методы, такие как теория очередей, могут быть использованы для анализа и оптимизации производительности системы в различных сценариях.

Системы массового обслуживания с ограниченной очередью играют важную роль во многих областях, где необходимо эффективно управлять потоками заявок. Оптимизация таких систем является сложной задачей, которая требует анализа и использования специализированных инструментов и методов.

В статье представлены принципы реализации одноканальной и многоканальной СМО с ограниченной очередью на платформе .NET с применением языка программирования языка C#.

Актуальность применения платформы .NET для реализации систем массового обслуживания с ограниченной очередью на языке программирования C# можно рассмотреть с нескольких точек зрения:

- C# является высокопроизводительным языком программирования, а платформа .NET предоставляет современные инструменты и библиотеки для оптимизации работы с данными и ресурсами. Это позволяет создавать высокопроизводительные СМО, способные обрабатывать большое количество заявок с минимальными задержками.
- Платформа .NET и C# предоставляют встроенную поддержку многопоточности и параллельных вычислений [7, 8]. Это особенно актуально для СМО, где часто необходимо параллельно обрабатывать несколько заявок и координировать работу нескольких каналов обслуживания.
- .NET предоставляет широкий набор библиотек и инструментов, которые могут быть использованы при разработке СМО. Это включает в себя библиотеки для работы с математическими моделями, статистическим анализом, симуляционным моделированием и другими инструментами, необходимыми для эффективного проектирования систем массового обслуживания.
- Платформа .NET поддерживает кросс-платформенное развертывание, что позволяет разрабатывать приложения, которые могут работать на различных операционных системах без изменения кода. Это добавляет гибкости в развертывание и масштабирование СМО в различных средах.
- C# известен своей чистотой синтаксиса и строгой типизацией, что способствует написанию читаемого и поддерживаемого кода [9]. Это облегчает разработку сложных систем, таких как СМО, и сокращает затраты на их поддержку и развитие.

## Принципы разработки одноканальной СМО с ограниченной очередью

Одноканальная система массового обслуживания с ограниченной очередью является важной моделью, использующейся в различных сферах, от телекоммуникаций до здравоохранения.

Для демонстрации реализации данного типа СМО с применением платформы .NET и языка программирования C# рассмотрим простой пример одноканальной СМО с максимальной вместимостью очереди равной 5.

Сначала создадим класс SingleServerQueue, который будет инкапсулировать логику СМО.

```
public class SingleServerQueue
{
    private Queue queue;
    private int maxQueueLength;
    private bool isServerBusy;

    public SingleServerQueue(int maxQueueLength)
    {
        this.queue = new Queue();
        this.maxQueueLength = maxQueueLength;
        this.isServerBusy = false;
    }

    public bool EnqueueCustomer(Customer customer)
    {
        if (queue.Count < maxQueueLength)
        {
            queue.Enqueue(customer);
            return true;
        }
        return false;
    }

    public void DequeueCustomer()
    {
        if (queue.Count > 0)
        {
            queue.Dequeue();
            if (queue.Count == 0)
            {
                isServerBusy = false;
            }
        }
    }

    // Дополнительные методы и свойства
}
```

Теперь, когда у нас есть основной класс, создадим симуляцию.

```
class Program
{
    static void Main(string[] args)
    {
```

```

    □□□□□□□□□SingleServerQueue□singleServerQueue□=□new□SingleServerQueue(5);□□□
    □□□□□□□□for□(int□i□=□0;□i□<□10;□i++)□
    □□□□□□□□{□
        □□□□□□□□□Customer□customer□=□new□Customer();□
        □□□□□□□□□bool□isEnqueued□=□singleServerQueue.EnqueueCustomer(customer);
        □□□□□□□□□if□(isEnqueued)□
        □□□□□□□□□{□
            □□□□□□□□□Console.WriteLine($"Customer□{i}□is□enqueued.");□
        □□□□□□□□□}□
        □□□□□□□□□else□
        □□□□□□□□□{□
            □□□□□□□□□Console.WriteLine($"Customer□{i}□is□rejected.");□
        □□□□□□□□□}□
        □□□□□□□}□
    }□
}

```

В этом примере кода создаем экземпляр одноканальной СМО с максимальной длиной очереди 5 и пытаемся добавить 10 клиентов.

После создания базовой симуляции, важным этапом является анализ и оптимизация СМО. В C# можно использовать различные библиотеки и инструменты для анализа производительности, такие как профилировщики и средства для работы со статистическими данными.

```

public□void□ProcessCustomers()
{
    □□□□while□(queue.Count□>□0)
    □□□□{□
        □□□□□Customer□customer□=□queue.Peek();□
        □□□□□if□(IsServiceCompleted(customer))□
        □□□□□□□{□
            □□□□□□□□□DequeueCustomer();□
        □□□□□□□}□
        □□□□□}□
    }□
private□bool□IsServiceCompleted(Customer□customer)□
{
    □□□□//□Логика□определения,□завершено□ли□обслуживание□клиента□
}

```

В приведенном выше фрагменте кода добавляем методы для обработки клиентов в очереди и проверки завершения обслуживания. Это можно расширить с использованием статистических методов для анализа времени ожидания, эффективности обслуживания и других метрик.

Благодаря объектно-ориентированной природе C#, можно легко расширять и модифицировать СМО, добавляя новые функциональные возможности или изменяя существующие. Например, можно добавить поддержку приоритетов в очереди, внедрить алгоритмы планирования или интегрировать с внешними источниками данных.

При создании реального приложения СМО также важно предоставить интерфейс

пользователя. С использованием .NET возможно создание как консольных, так и графических пользовательских интерфейсов (GUI), что позволяет создавать удобные инструменты для мониторинга и управления СМО.

### **Принципы разработки многоканальной СМО с ограниченной очередью**

В современных условиях, когда множество процессов требуют оптимизации и параллелизации, многоканальные системы массового обслуживания становятся все более актуальными. Они находят применение в различных отраслях: от телекоммуникаций до логистики. В данном фрагменте научной статьи рассматриваются принципы разработки многоканальной СМО с ограниченной очередью на платформе .NET с использованием языка программирования C#.

Основными компонентами СМО являются каналы обслуживания и очередь. В C# можно использовать классы и коллекции для моделирования этих компонентов. Например, очередь можно представить с помощью коллекции Queue, а каналы обслуживания – с помощью списка объектов каналов.

```
public class ServiceChannel
{
    public bool IsBusy { get; set; }

    // другие свойства и методы
}

public class MultiChannelQueueSystem
{
    private Queue queue = new Queue();
    private List serviceChannels = new List();
    public MultiChannelQueueSystem(int numberOfChannels)
    {
        for (int i = 0; i < numberOfChannels; i++)
        {
            serviceChannels.Add(new ServiceChannel());
        }
    }

    // другие методы
}
```

Обработка заявок в многоканальной СМО включает добавление заявок в очередь и их последующую обработку доступными каналами.

```
public void EnqueueCustomer(Customer customer)
{
    if (queue.Count < MaxQueueLength)
    {
        queue.Enqueue(customer);
    }
}

public void ProcessQueue()
{
    foreach (ServiceChannel channel in serviceChannels)
    {
        if (!channel.IsBusy && queue.Count > 0)
```

```

    {
        Customer customer = queue.Dequeue();
        // Обработка заявки на канале
    }
}

```

После моделирования многоканальной СМО важно провести анализ ее работы. Это может включать в себя мониторинг времени ожидания в очереди, загрузки каналов и других параметров.

С использованием объектно-ориентированных подходов (ООП) подходов в C# можно создать гибкие и многократно используемые компоненты, что позволит с легкостью адаптировать систему к различным условиям и требованиям. Например, можно реализовать базовый класс для канала обслуживания и наследовать его для специальных типов каналов.

```

public class ServiceChannel
{
    public virtual bool IsBusy { get; set; }

    // другие свойства и методы
}

public class PriorityServiceChannel : ServiceChannel
{
    public override bool IsBusy
    {
        get
        {
            // Кастомная логика для приоритетного канала
        }
        set /* ... */
    }
}

```

Также можно добавить события и делегаты для обработки различных сценариев в системе, таких как оповещения при переполнении очереди.

Поскольку многоканальная СМО предполагает одновременное обслуживание нескольких заявок, важно эффективно использовать многопоточность [\[10\]](#). В C# это можно сделать с помощью Task Parallel Library (TPL).

```

public void ProcessQueue()
{
    Parallel.ForEach(serviceChannels, channel =>
    {
        if (!channel.IsBusy && queue.Count > 0)
        {
            Customer customer = queue.Dequeue();
            // Обработка заявки на канале
        }
    });
}

```

При разработке СМО важно проводить тестирование и валидацию для проверки корректности работы системы. Это можно делать с использованием модульных тестов и интеграционного тестирования.

Разработка многоканальной системы массового обслуживания с ограниченной очередью на платформе .NET с использованием C# позволяет создавать эффективные и гибкие решения. Благодаря возможностям ООП, многопоточности и мощным библиотекам, C# является отличным выбором для реализации сложных СМО, способных адаптироваться к меняющимся требованиям и условиям.

### **Практическое использование**

Системы массового обслуживания активно применяются в различных сферах, от телекоммуникаций до финансов [11, 12]. В данной части статьи мы рассмотрим примеры практической реализации одноканальной и многоканальной СМО с ограниченной очередью на платформе .NET с использованием C#.

Для моделирования одноканальной СМО с ограниченной очередью создадим класс SingleChannelQueueSystem.

```
public class SingleChannelQueueSystem
{
    private Queue queue = new Queue();
    private bool isChannelBusy = false;
    private readonly int maxQueueLength;

    public SingleChannelQueueSystem(int maxQueueLength)
    {
        this.maxQueueLength = maxQueueLength;
    }

    public void EnqueueCustomer(Customer customer)
    {
        if (queue.Count < maxQueueLength)
        {
            queue.Enqueue(customer);
        }
    }

    public void ProcessQueue()
    {
        if (!isChannelBusy && queue.Any())
        {
            var customer = queue.Dequeue();
            isChannelBusy = true;
            // Обработка клиента
            isChannelBusy = false;
        }
    }
}
```

Пример использования:

```
SingleChannelQueueSystem queueSystem = new SingleChannelQueueSystem(5);
queueSystem.EnqueueCustomer(new Customer());
```

```
queueSystem.EnqueueCustomer(new Customer());
queueSystem.ProcessQueue();
```

Для моделирования многоканальной СМО создадим класс MultiChannelQueueSystem.

```
public class MultiChannelQueueSystem
{
    private Queue queue = new Queue();
    private List serviceChannels;
    private readonly int maxQueueLength;
    public MultiChannelQueueSystem(int numberOfChannels, int maxQueueLength)
    {
        serviceChannels = new List(new bool[numberOfChannels]);
        this.maxQueueLength = maxQueueLength;
    }
    public void EnqueueCustomer(Customer customer)
    {
        if (queue.Count < maxQueueLength)
        {
            queue.Enqueue(customer);
        }
    }
    public void ProcessQueue()
    {
        for (int i = 0; i < serviceChannels.Count; i++)
        {
            if (!serviceChannels[i] && queue.Any())
            {
                var customer = queue.Dequeue();
                serviceChannels[i] = true;
                // Обработка клиента
                serviceChannels[i] = false;
            }
        }
    }
}
```

Пример использования:

```
MultiChannelQueueSystem queueSystem = new MultiChannelQueueSystem(3, 10);
queueSystem.EnqueueCustomer(new Customer());
queueSystem.EnqueueCustomer(new Customer());
queueSystem.EnqueueCustomer(new Customer());
queueSystem.ProcessQueue();
```

После создания базовых моделей СМО, важно анализировать их производительность и корректность работы. Использование инструментов профилирования, таких как Visual Studio Profiler, поможет выявить узкие места и оптимизировать код.

Для улучшения производительности многоканальной СМО можно воспользоваться параллелизацией. Это можно сделать с использованием Task Parallel Library (TPL) в C#.

```
public void ProcessQueueParallel()
```

```

{
    Parallel.For(0, serviceChannels.Count, i =>
{
    if (!serviceChannels[i] && queue.Any())
{
    var customer = queue.Dequeue();
    serviceChannels[i] = true;
    // Обработка клиента
    serviceChannels[i] = false;
}
});
}

```

Тестирование и валидация системы необходимы для обеспечения точности и надежности моделирования. Для этого можно использовать модульные тесты и инструменты тестирования, такие как MSTest, NUnit или xUnit.

Пример модульного теста:

```

[TestClass]
public class QueueSystemTests
{
    [TestMethod]
    public void TestSingleChannelQueueSystem()
{
    var queueSystem = new SingleChannelQueueSystem(5);
    queueSystem.EnqueueCustomer(new Customer());
    queueSystem.ProcessQueue();
    // Проверка условий (например, корректное количество заявок в системе)
}
}

```

Одноканальные и многоканальные системы массового обслуживания могут быть применены в различных сферах, таких как телекоммуникации, банковское дело, логистика и т.д. В частности, они могут быть использованы для оптимизации времени ожидания клиентов, распределения ресурсов и улучшения общей производительности системы.

## **Заключение**

Разработка систем массового обслуживания с ограниченной очередью представляет собой комплексный процесс, включающий моделирование, анализ, оптимизацию и создание пользовательского интерфейса. Применение современных технологий и инструментов, доступных в рамках платформы .NET, позволяет создавать эффективные и масштабируемые решения, способные справляться с реальными задачами в различных отраслях.

В статье освещены ключевые аспекты разработки, моделирования и практической реализации СМО с использованием технологий .NET и языка C#. Проведено детальное рассмотрение одноканальных и многоканальных систем, их особенностей, принципов функционирования и методов реализации.

Кроме рассмотренных аспектов существует ряд направлений для дальнейшего

исследования и развития систем массового обслуживания:

- Интеграция с искусственным интеллектом: с применением искусственного интеллекта можно улучшить процессы принятия решений в системе, например, предсказывать поток заявок и динамически изменять параметры системы для оптимизации обслуживания.
- Использование облачных технологий: интеграция с облачными платформами может обеспечить повышение доступности, масштабируемости и надежности системы, а также снижение затрат на инфраструктуру.
- Расширенная аналитика и визуализация: разработка инструментов аналитики и визуализации данных позволит администраторам и пользователям системы более эффективно мониторить и анализировать ход обслуживания и использовать эту информацию для оптимизации процессов.
- Обеспечение безопасности: безопасность данных и транзакций является ключевым аспектом в современных системах массового обслуживания [13, 14]. Разработка и интеграция механизмов защиты от несанкционированного доступа и вредоносного программного обеспечения является важной частью создания надежной системы.

## Библиография

1. Harchol-Balter M. Performance modeling and design of computer systems: queueing theory in action / Cambridge University Press, 2013.
2. Шевцов А. Н., Щитов А. Н., Конорева Н. А. Моделирование телекоммуникационных систем с помощью СМО // Математика и ее приложения в современной науке и практике, 2015. С. 128-132.
3. Вишневский В. М., Дудин А. Н. Системы массового обслуживания с коррелированными входными потоками и их применение для моделирования телекоммуникационных сетей // Автоматика и телемеханика. 2017. № 8. С. 3-59.
4. Gross D., Harris C. Fundamentals of Queueing Theory / Wiley-Interscience, 1998.
5. Осипов Л. А. Имитационное моделирование систем массового обслуживания с ограниченной очередью // Наука и техника транспорта. 2010. № 4. С. 30-36.
6. Фурина К. О., Осечкина Т. А. Математическая модель одноканальной системы массового обслуживания с ограниченной очередью // Наука и современность. 2014. № 2. С. 103-110.
7. Гибадуллин Р.Ф. Потокобезопасные вызовы элементов управления в обогащенных клиентских приложениях // Программные системы и вычислительные методы. 2022. № 4. С. 1-19.
8. Гибадуллин Р.Ф., Викторов И.В. Неоднозначность результатов при использовании методов класса Parallel в рамках исполняющей среды .NET Framework // Программные системы и вычислительные методы. 2023. № 2. С. 1-14.
9. Albahari J. C# 10 in a Nutshell / O'Reilly Media, Inc., 2022.
10. Викторов И.В., Гибадуллин Р.Ф. Разработка синтаксического дерева для автоматизированного транслятора последовательного программного кода в параллельный код для многоядерных процессоров // Программные системы и вычислительные методы. 2023. № 1. С. 13-25.
11. Осипов Г. С. Исследование систем массового обслуживания с ожиданием в AnyLogic // Бюллетень науки и практики. 2016. № 10 (11). С. 139-151.
12. Осипов Г. С. Системы массового обслуживания с ограниченной длительностью ожидания // Бюллетень науки и практики. 2016. № 12 (13). С. 28-36.

13. Гибадуллин Р.Ф., Гашигуллин Д.А., Вершинин И.С. Разработка декоратора StegoStream для ассоциативной защиты байтового потока. Моделирование, оптимизация и информационные технологии. 2023. 11(2). URL: [moitvvt.ru/ru/journal/pdf?id=1359](http://moitvvt.ru/ru/journal/pdf?id=1359).
14. Гибадуллин Р.Ф., Вершинин И.С., Глебов Е.Е. Разработка приложения для ассоциативной защиты файлов // Инженерный вестник Дона. 2023. № 6. URL: [ivdon.ru/ru/magazine/archive/n6y2023/8462/](http://ivdon.ru/ru/magazine/archive/n6y2023/8462/).

## Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предметом исследования в рецензируемой статье выступают Принципы разработки систем массового обслуживания с ограниченной очередью.

Методология исследования базируется на изучении отечественного и зарубежного опыта разработки систем массового обслуживания на языке C#.

Актуальность работы авторы связывают с тем, что системы массового обслуживания играет ключевую роль в анализе и оптимизации процессов, связанных с обслуживанием потоков заявок в различных областях, таких как телекоммуникации, транспорт, здравоохранение и производство.

Научная новизна рецензируемого исследования, к сожалению, не сформулирована автором, и после прочтения статьи не ясно в чем состоит приращение научного знания.

Структурно в статье выделены следующие разделы: Введение, Принципы разработки одноканальной системы массового обслуживания (СМО) с ограниченной очередью, Принципы разработки многоканальной СМО с ограниченной очередью, Практическое использование, Заключение, Библиография.

Авторы в начале статьи приводят основные понятия теории массового обслуживания и рассматривают типы систем массового обслуживания. Для демонстрации реализации данного СМО с применением платформы .NET и языка программирования C# в работе приводятся фрагменты программного кода на языке C#. В публикации рассматривается, как после создания базовой симуляции провести анализ и оптимизацию СМО с использованием различных библиотек и инструментов анализа производительности (профилировщиков и средств для работы со статистическими данными). Особое внимание уделено практической реализации СМО с ограниченной очередью на платформе .NET с использованием C#. В заключении рассмотрены перспективы развития систем массового обслуживания.

Библиографический список включает 14 источников – публикации российских и зарубежных авторов по теме статьи. В тексте имеются адресные ссылки на библиографические источники, что подтверждает наличие апелляции к оппонентам.

Из недостатков, следует отметить следующие. Во-первых, в названии статьи используется аббревиатура (акроним), что является не лучшим вариантом, поскольку в научных публикациях принято после первого употребления сокращений приводить полную расшифровку слов, а в названии статьи этого по понятным причинам сделать не удастся. Автору предлагается рассмотреть следующую редакцию названия статьи: «Принципы разработки систем массового обслуживания с ограниченной очередью на современных программных платформах». Во-вторых, в тексте статьи не указан такой общепринятый в современных научных публикациях элемент как методы исследования. В-третьих, в статье не сформулирована цель и не отражены элементы научной новизны.

В-четвертых, объем статьи представляется уместным сократить за счет исключения из текста публикации общеизвестных положений теории массового обслуживания, изложенных в начале статьи и повторяющих материал учебной литературы.

Рецензируемый материал соответствует направлению журнала «Программные системы и вычислительные методы», отражает результаты проведенной авторами работы, имеет практическое значение для разработки систем массового обслуживания с ограниченной очередью на платформе .NET, может вызвать интерес у читателей, однако, статья нуждается в доработке в соответствие с высказанными замечаниями.

**Программные системы и вычислительные методы***Правильная ссылка на статью:*

Марченко А.Г., Щемелинин Д.А. — Разработка автоматизированной системы тестирования облачного сервиса развертывания виртуальных машин с использованием современных средств мониторинга // Программные системы и вычислительные методы. – 2023. – № 2. DOI: 10.7256/2454-0714.2023.2.40755 EDN: TEVFBN URL: [https://nbppublish.com/library\\_read\\_article.php?id=40755](https://nbppublish.com/library_read_article.php?id=40755)

**Разработка автоматизированной системы тестирования облачного сервиса развертывания виртуальных машин с использованием современных средств мониторинга****Марченко Андрей Геннадьевич**

ORCID: 0009-0001-9276-3907



Архитектор облачных программных систем, Компания Интел

193318, Россия, Санкт-Петербург область, г. Санкт-Петербург, ул. Бадаева, 6к1

✉ mar4enko.ag@gmail.com

**Щемелинин Дмитрий Александрович**

ORCID: 0000-0003-3032-130X

доктор технических наук



Вице-президент, Компания Интел

195251, Россия, г. Санкт-Петербург, ул. Политехническая, 29

✉ dshchmel@gmail.com

[Статья из рубрики "Телекоммуникационные системы и компьютерные сети"](#)**DOI:**

10.7256/2454-0714.2023.2.40755

**EDN:**

TEVFBN

**Дата направления статьи в редакцию:**

15-05-2023

**Аннотация:** Объектом данного исследования является сервис по управлению виртуальными машинами в облачной среде. При разработке и эксплуатации такого сервиса возникает необходимость оценки его доступности и надежности на соответствие выбранному уровню качества, на который может рассчитывать клиент. В данной работе представлена разработанная система, позволяющая тестировать доступность облачного сервиса по управлению виртуальными машинами. Рассмотрен метод интеграции с

существующей системой мониторинга на предприятии с использованием открытого программного обеспечения с целью уменьшения стоимости разработки и эксплуатации. Разработан и реализован тест-кейс по развертыванию и удалению виртуальной машины с использованием графического интерфейса пользователя, а также определены критерии срабатывания триггеров. Собраны и проанализированы требования к архитектуре и реализации системы на основе производственной статистики сервиса создания виртуальных машин с помощью системы мониторинга Prometheus. Новизна исследования заключается в разработке нового метода тестирования облачного сервиса по управлению виртуальными машинами с целью повышения его надежности и доступности. На основе данного метода описана и реализована система тестирования виртуальных машин а также метод интеграции в систему мониторинга облачного сервиса компании Intel. В процессе эксплуатации облачных сред с помощью данной системы были выявлены проблемные места в архитектуре сервиса создания виртуальных машин, что позволило своевременно оптимизировать работу системы. Описанный метод является эффективным способом тестирования облачных сервисов, и также может использоваться для анализа и повышения надежности и доступности.

### **Ключевые слова:**

информационные технологии, мониторинг, prometheus, selenium, kubernetes, python, тестирование приложений, метрики, обработка информации, пороговое значение

### **Введение**

Облачные провайдеры предоставляют различные виды услуг, такие как размещение виртуальных машин, хранение данных, базы данных, аналитику и многое другое. Как правило, доступ к этим услугам осуществляется как через веб интерфейс, так и через API (англ. Application Programming Interface) – набор готовых интерфейсов для программной интеграции с сервисами посредством различных протоколов. Мониторинг доступности и производительности облачных систем и сервисов является одним из важнейших аспектов обеспечения их стабильной работы, а также позволяет выявлять и исправлять проблемы возникающие в процессе эксплуатации. Для мониторинга доступности облачных сервисов используется специализированное программное обеспечение. Одним из популярных решений с открытым исходным кодом является сервис Prometheus [\[1\]](#). Эта система мониторинга позволяет собирать и хранить данные о производительности и доступности в виде временных рядов – последовательности значений, которые изменяются во времени. Собранные данные в дальнейшем можно анализировать с использованием различных методов, таких как визуализация, статистический анализ и машинное обучение.

В данном исследовании был предложен новый метод тестирования и мониторинга облачного сервиса, предоставляющего пользователю возможность арендовать виртуальные машины в облаке, а также выбрать один из предложенных типов виртуальной машины с заданным объемом оперативной памяти, процессоров и хранилища. Данный тип сервиса очень популярен как для предприятий, так и для индивидуальных пользователей и позволяет сэкономить значительные ресурсы предоставляя вычислительные мощности по запросу.

### **Объект научного исследования**

Объектом исследования является внутренняя облачная платформа компании Intel [\[2\]](#),

которая позволяет разработчикам и исследователям использовать ресурсы вычислительных систем для тестирования и оптимизации своих приложений и алгоритмов. Платформа также предоставляет широкие возможности по управлению инфраструктурой для вычислений, а также эффективному управлению ресурсами и используется для выполнения различных задач, таких как обработка больших данных и машинное обучение. Компания Intel является одним из ведущих мировых производителей электронных устройств и компьютерных компонентов: микропроцессоров и наборов системной логики (чипсетов) для клиентских вычислительных систем и для дата-центров, чипов для систем искусственного интеллекта и для интернета вещей, энергонезависимой памяти.

Одним из ключевых сервисов облачной платформы является VMaaS (англ. Virtual Machine as a Service), предоставляющий виртуальную инфраструктуру. Изучив статистику резервирований виртуальных машин во внутреннем облаке (рис 1), можно сделать вывод о том, что спрос на виртуальную инфраструктуру постоянно растет, и следовательно, повышаются требования к стабильности и надежности сервиса.



**Рисунок 1.** Количество резервирований виртуальных машин в системе VMaaS с 06.01.2023 по 01.04.2023

## Цель исследования

Предметом данного исследования является разработка метода и системы тестирования доступности сервиса развертывания виртуальных машин в облаке на базе Prometheus и Selenium [3], отличающиеся следующей научной новизной:

- Реализация сервиса тестирования развертывания виртуальной машины в облаке
- Интеграция инструментов Prometheus и Selenium с сервисом развертывания и определение сценариев мониторинга и тестирования облачных ресурсов
- Новый метод сбора и анализа данных о доступности с помощью Prometheus и Selenium, включая определение проблем с производительностью и оптимизация работы сервиса развертывания виртуальных машин в облаке

## Методы Исследования

Основным функциональным требованием для подсистемы тестирования является возможность создания и удаления виртуальной машины с заданными характеристиками – память, процессор, объем диска.

Для взаимодействия с системой был выбран пользовательский интерфейс. Это позволило не только обеспечить максимальное соответствие тестового сценария с поведением конечного пользователя в системе, но и организовать сбор метрик для анализа качества сервисов, обеспечивающих функционал, целью которого является выявление потенциальных проблем до того, как о них начнут сообщать пользователи.

Были сформулированы следующие функциональные требования к разработанной подсистеме:

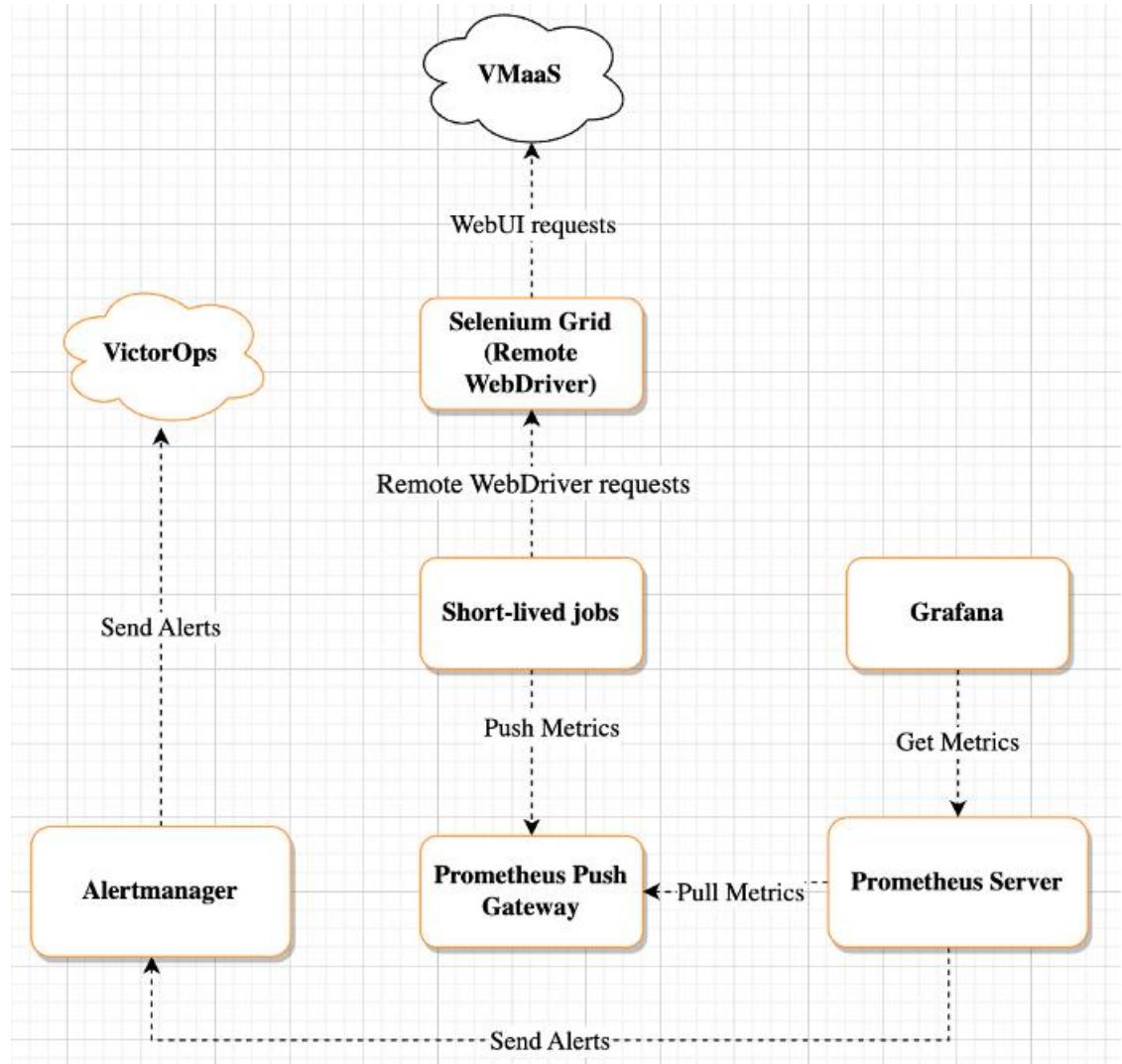
- Реализация входа в облачную систему
- Навигация по интерфейсу, локализация и выбор сервиса развертывания виртуальной машины
- Создание и удаление виртуальной машины
- Проверка статуса готовности виртуальной машины
- Вычисление ключевых метрик сервиса:
  - Отслеживание времени входа в систему
  - Отслеживание времени создания виртуальной машины
  - Отслеживание времени работы всего сценария

При выборе ключевых метрик была учтена необходимость дальнейшего прогнозирования доступности сервиса с применением математических моделей и алгоритмов прогнозирования событий для достижения целевых бизнес-показателей [\[4, 5, 6\]](#).

Также были рассмотрены нефункциональные требования. Задача создания виртуальной машины с помощью пользовательского интерфейса — это процесс с большим количеством шагов, каждый из которых необходимо рассматривать с точки зрения возникновения потенциальных ошибок и их соответствующей обработки, включая обеспечение надежного и стабильного поведения системы в случае, если задача неожиданно завершается с ошибкой. Следовательно, были сформулированы следующие нефункциональные требования:

- Запуск сценария с заданной периодичностью и по расписанию
- Автоматический повтор неуспешной операции в ходе выполнения сценария. Предусмотреть конфигурацию интервала ожидания между повторами, и количество повторов
- Глобальный лимит времени на весь сценарий. В случае если сценарий не укладывается в отведенный лимит, он должен быть завершен принудительно

Для реализации тестового сценария с учетом сформулированных функциональных и нефункциональных требований был выбран язык программирования Python и библиотека selenium-python [\[7\]](#) на основе Selenium WebDriver. Язык Python очень распространен [\[8\]](#) и позволяет в короткие сроки реализовывать приложения разного уровня сложности. Простой и понятный синтаксис, отсутствие дополнительных требований к производительности, а также простота отладки и тестирования делает его идеальным кандидатом для данного исследования. Selenium WebDriver [\[9\]](#) вместе с библиотекой selenium-python широко используется в целях тестирования веб-приложений и позволяют эмулировать взаимодействие пользователя с браузером.

**Рисунок 2.** Архитектура системы

На основе сформулированных функциональных и нефункциональных требований была разработана соответствующая архитектура (рис. 2). В качестве платформы для реализации был выбран Kubernetes – открытое программное обеспечение для автоматизации развертывания, масштабирования и управления контейнеризированными приложениями, а также специальный тип Kubernetes ресурса – периодическая задача (англ. Cronjob). Данный тип ресурса позволяет планировать и запускать задачи в определенное время, контролировать расписание и периодичность, управлять повторными запусками в случае ошибок, а также задавать глобальный лимит на выполнение задачи. Для взаимодействия с пользовательским интерфейсом облачного сервиса развертывания виртуальных машин была использована подсистема Selenium Grid [10]. Данный компонент представляет собой платформу для удаленного выполнения скриптов с использованием WebDriver. С его помощью обеспечивается параллельный запуск тестовых сценариев путем выполнения их на разных экземплярах удаленных браузеров. VMaaS – это облачный сервис, который предоставляет платформу для развертывания виртуальных машин. В качестве основной системы сбора и хранения метрик был выбран Prometheus, который также отвечает за обработку метрик и генерацию уведомлений. Alertmanager – обрабатывает уведомления, сгенерированные Prometheus и пересыпает их в VictorOps. Данный компонент отвечает за управление и обработку инцидентов в режиме реального времени и позволяет оповещать соответствующие команды в случае возникновения проблем. Prometheus Pushgateway – это промежуточный слой между короткоживущими задачами и основным сервером

**Prometheus.** Данный компонент необходим, так как особенностью работы Prometheus является периодический опрос целевых объектов для сбора метрик через короткий интервал - pull модель (англ. pull model), в то время как задачи запускаются по собственному расписанию и для эффективного сбора метрик необходимо реализовать push модель (англ. Push model). Задача в процессе выполнения отправляет HTTP (англ. Hypertext transfer protocol) запрос в Prometheus Pushgateway, который временно сохраняет метрики и является целевым объектом для Prometheus. Для создания и настройки графиков, диаграмм и других панелей визуализации метрик, совместно с Prometheus был использован сервис Grafana [11].

Для оценки доступности сервиса развертывания виртуальных машин были выбраны ключевые метрики. За основную метрику было выбрано время входа пользователя в систему. Это время, в течение которого происходит аутентификация и авторизация пользователя, с использованием зарегистрированных учетных данных. Замер был осуществлен с момента появления формы ввода учетных данных, до момента, когда пользователю был показан главный экран с возможностью запуска виртуальной машины. Для подсчета временных метрик было предложено воспользоваться следующей функцией time\_monitor.

```
@contextmanager
def time_monitor(self, metric_name: str, start_time: Optional[datetime] = None):
    start_time = start_time or datetime.now()
    try:
        yield
    finally:
        self.metrics[metric_name] = (datetime.now() - start_time).seconds
```

Использование данной функции для процесса входа в систему (англ. login) выглядит следующим образом:

```
wrapped_login = retry(
    on_exception=WebDriverException,
    to_exception=LoginError,
    attempts=self.login_retry_count,
    sleep_interval=self.login_retry_interval
)(self.login)
with self.time_monitor('login'):
    wrapped_login()
```

В качестве следующей ключевой метрики было выбрано время запуска виртуальной машины (англ. launch instance). Именно оно определяет допустимый предел, в течение которого мы ожидаем готовность виртуальной машины.

```
with self.time_monitor('launch_instance', start_time):
    self.driver.find_element(
        By.XPATH, "//button[@class='btn btn-primary' and text() = 'Launch Instance']")
    .click()
    logger.info(f'Instance launched: {start_time}')
    self.wait_running_instance()
```

В случае если данный предел будет превышен, сценарий завершится с ошибкой.

```
deadline = datetime.now() + timedelta(seconds=self.wait_deadline_seconds)
while True:
    if datetime.now() > deadline:
        raise VmProvisioningTimeout(f'VM provisioning exceeds wait deadline {self.wait_deadline_seconds}')
```

Последней ключевой метрикой было обозначено полное время работы сценария создания виртуальной машины, включающее в себя обработку ожидаемых исключительных ситуаций, повторы операций, ожидание готовности машины и последующее ее удаление. Удаление машины является неотъемлемой частью завершения сценария тестирования, так как все ресурсы выделенные в процессе работы должны быть освобождены.

Была проанализирована метрика времени запуска виртуальной машины и определено пороговое значение, которое было использовано в Prometheus для создания нотификаций.

Для выбора оптимального метода определения порогового значения необходимо проверить, является ли распределение нормальным. Для этого было предложено применить тест Шапиро-Уилка [\[12\]](#). В основе проверки лежит проверка нулевой гипотезы – данные распределены нормально. Альтернативная гипотеза – данные не имеют нормального распределения. Результат критерия Шапиро-Уилка – это значение статистики, а также значение p-value. В качестве порогового значения в большинстве случаев берется 0.05. При значении p-value меньше 0.05 нулевая гипотеза отклоняется. Для реализации была использована библиотека SciPy языка python. В качестве входных данных был использован массив метрик времени запуска виртуальной машины из Prometheus за последние 7 дней.

```
from scipy.stats import shapiro
from datetime import timedelta, datetime
from prometheus import PrometheusClient

p = PrometheusClient(PROMETHEUS_ADDRESS, PROMETHEUS_USER, PROMETHEUS_PASSWORD)

query = 'vm_test_provisioning_seconds'
end = datetime.now()
start = end - timedelta(days=7)
step = 600 # 10 min

input_list = p.make_range_query(start.timestamp(), end.timestamp(), step, query)
statistic, p_value = shapiro(input_list)

print('Statistic:', statistic)
print('p-value:', p_value)

alpha = 0.05
if p_value > alpha:
    print('fail to reject H0')
else:
    print('reject H0')
```

В результате было получено низкое значение p-value  $2.098965063816978e-31$  на основе которого нулевая гипотеза была отклонена и сделан вывод о том, что данные не имеют нормального распределения. Для вычисления порогового значения (англ. Threshold) был использован следующий метод – сумма медианы и трех абсолютных отклонений [\[13\]](#)

$\text{Threshold} = \text{Median} + 3 * \text{MAD}$

С помощью библиотеки NumPy языка Python была вычислена медиана данных и рассчитано среднее абсолютное отклонение.

```

import numpy as np
median = np.median(input_list)
mad = np.median(np.abs(input_list - median))
threshold = median + 3 * mad

print("Median: ", median)
print("MAD: ", mad)
print("Threshold: ", threshold)

```

В качестве результата было рассчитана медиана 131.5, среднее абсолютное отклонение 12.5 и пороговое значение 169.

**Результаты и обсуждение.** Была разработана система позволяющая собирать различные ключевые параметры облачного сервиса развертывания виртуальных машин, на основе реального сценария имитирующего работу пользователя.



**Рисунок 3.** Время запуска виртуальной машины

Проанализировав собранные данные (рис. 3) методом суммирования медианы и трех абсолютных отклонений, было вычислено пороговое значение 169, которое можно использовать в Prometheus в выражении описывающим потенциальную проблему в сервисе развертывания виртуальных машин. Превышение данного порога сигнализирует о значительном отклонении от ожидаемых значений, и требует немедленной эскалации и технического расследования причин возникновения со стороны профильной команды поддержки сервиса [14].

## Выводы

Основным результатом работы является разработанная система тестирования доступности виртуальных машин и метод интеграции в систему мониторинга внутреннего облачного сервиса компании Intel. Был предложен и реализован основной тест-кейс – создание и удаление виртуальной машины с помощью графического интерфейса пользователя, определены критерии срабатывания триггеров. В процессе эксплуатации облачных сред с помощью данной системы были выявлены проблемные места в архитектуре сервиса создания виртуальных машин, что позволило своевременно оптимизировать работу системы. Путем анализа собранных данных были выявлены и устранены проблемы с производительностью веб-интерфейса, что позволило существенно улучшить пользовательский опыт в использовании данного интерфейса. Разработанная система позволяет регистрировать отклонения в поведении сервиса по ряду ключевых параметров, таких как время входа в систему и время запуска

виртуальной машины. Благодаря этому удалось значительно сократить время реакции на возникающие инциденты. Использование бесплатных технологий с открытым исходным кодом, таких как Prometheus, Selenium способствует повышению эффективности и снижению затрат компании на обслуживание облачных сервисов. Описанный метод является эффективным способом тестирования облачных сервисов, и также может использоваться для анализа и повышения надежности и доступности – важнейших критериев для бизнес-приложений, работающих в режиме 24/7

## Библиография

1. Prometheus – Monitoring system & time series database [Электронный ресурс]. URL: <https://prometheus.io/docs/introduction/overview/> (дата обращения 04.04.2023).
2. Официальный Интернет-сайт Intel [Электронный ресурс]. URL: <https://www.intel.com/> (дата обращения 04.04.2023).
3. The selenium browser automation project [Электронный ресурс]. URL: <https://www.selenium.dev/documentation/> (дата обращения 04.04.2023).
4. Щемелинин Д.А. Математические модели и методы мониторинга и прогнозирования состояния глобально распределенных вычислительных комплексов. Труды учебных заведений связи. 2021. Т. 7. № 3. С. 73–78.
5. Щемелинин Д.А. Метод прогнозирования событий в глобально распределенных вычислительных комплексах. Современная наука: актуальные проблемы теории и практики. Серия: Естественные технические науки. 2021. № 12–2. С. 47–54.
6. Щемелинин Д.А. Метод и алгоритм автоматического восстановления информационных сервисов на основе объективных прогностических данных мониторинга. Современная наука: актуальные проблемы теории и практики. Серия: Естественные технические науки. 2021. № 8. С. 140–144.
7. Selenium with python [Электронный ресурс] URL: <https://selenium-python.readthedocs.io/> (дата обращения 04.04.2023).
8. The TIOBE Programming Community index an indicator of the popularity of programming languages [Электронный ресурс] URL: <https://www.tiobe.com/tiobe-index/> (дата обращения 04.04.2023).
9. Sujay Raghavendra. – Python testing with selenium – Apress Berkeley CA, 2020 – 4 с. ISBN 978-1-4842-6249-8
10. When to use selenium grid [Электронный ресурс] URL: <https://www.selenium.dev/documentation/grid/applicability/> (дата обращения 04.04.2023)
11. Grafana documentation [Электронный ресурс]. URL: <https://grafana.com/docs/grafana/latest/> (дата обращения 04.04.2023)
12. Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. Journal of Statistical Modeling and Analytics, Vol.2, No. I, 21-33, 2011. URL: <https://www.nrc.gov/docs/ML1714/ML17143A100.pdf> (дата обращения 04.04.2023)
13. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. Journal of Experimental Social Psychology, 49(4), 764–766. URL: <https://www.sciencedirect.com/science/article/pii/S0022103113000668> (дата обращения 04.04.2023)
14. Щемелинин Д.А. Система критериев и алгоритм обработки информации и принятия решений для программного модуля отображения наиболее значимых событий мониторинга в информационной системе. XXI век: итоги прошлого и проблемы

настоящего плюс 2021. Т. 10. № 3 (55). С. 67–71.

## Результаты процедуры рецензирования статьи

В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.

Со списком рецензентов издательства можно ознакомиться [здесь](#).

Предмет исследования. С учётом указанного наименования научной статьи она должна быть посвящена разработке автоматизированной системы тестирования облачного сервиса развертывания виртуальных машин с использованием современных средств мониторинга. Содержание статьи соответствует заявленной теме.

Методология исследования базируется на разработке системы тестирования доступности виртуальных машин и метод интеграции в систему мониторинга внутреннего облачного сервиса компании Intel. Причём, исследование проведено с использованием Python, что формирует положительное впечатление от качества выбранной методологии. Очень ценно, что по тексту статьи автор активно использует графический метод представления результатов научных исследований, при этом при доработке статьи следует подписать наименования у всех рисунков (в т.ч. относящиеся к коду) и указать источник данных (если разработано автором, то следует так и указать).

Актуальность исследования вопросов, связанных с практическим использованием информационных технологий и достижений научно-технического прогресса, не вызывает сомнения, т.к. это позволяет обеспечить научное обоснование мероприятий по обеспечению технологического суверенитета Российской Федерации, о необходимости которого было заявлено Президентом России.

Научная новизна в представленных на рецензирование материалах содержится и крайне чётко определена: связана с разработанной системой тестирования доступности виртуальных машин и метод интеграции в систему мониторинга внутреннего облачного сервиса компании Intel.

Стиль, структура, содержание. Стиль изложения научной. Статья имеет достаточно чётко выстроенную структуру, что формирует положительное впечатление от ознакомления с нею. Рекомендуется дополнить блоком «Предложения по практическому использованию авторской разработки». Наполнение содержанием данного структурного элемента обеспечит рост интереса к данной научной публикации в геометрической прогрессии. Ценно, что статья базируется на конкретных результатах, полученных автором.

Библиография. Автором сформирован библиографический список из 14 источников. Однако не изучены научные публикации 2022-2023 гг., в т.ч. в зарубежных изданиях. Это позволило бы учесть самые последние тенденции в научной литературе по рассматриваемым вопросам.

Апелляция к оппонентам. Несмотря на сформированный список литературы и, в целом, высокое качество научной статьи, в т.ч. связанное с наличием конкретных авторских суждений и разработок, не осуществлено их обсуждение с итогами исследований, отражённых в трудах других авторов. При доработке статьи необходимо уделить этому важное внимание, т.к. это ещё больше повысит качество научной статьи, представленной на рецензирование, в т.ч. сформирует предпосылки для расширения

читательской аудитории.

Выводы, интерес читательской аудитории. С учётом всего вышеизложенного, следует отметить, что статья выполнена на актуальную тему на достаточно высоком уровне, обладает высоким уровнем потенциального интереса читательской аудитории (как в научном сообществе, так и со стороны органов государственной власти Российской Федерации, субъектов Российской Федерации, субъектов предпринимательства). После проведения указанных в тексте рецензии корректировок (носящих преимущественно формальный характер, но соблюдение правил оформления научной статьи также важно, как и её содержание) может быть одобрена к опубликованию.

Программные системы и вычислительные методы

Правильная ссылка на статью:

Юсупов Б.З. — Разработка учебного стенда охранно-пожарной системы для обучения студентов //

Программные системы и вычислительные методы. – 2023. – № 2. DOI: 10.7256/2454-0714.2023.2.43552 EDN: TFKJAS URL: [https://nbpublish.com/library\\_read\\_article.php?id=43552](https://nbpublish.com/library_read_article.php?id=43552)

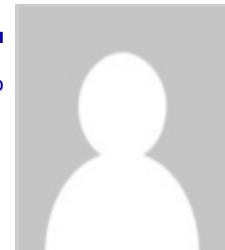
## Разработка учебного стенда охранно-пожарной системы для обучения студентов

Юсупов Булат Зуфарович

обучающийся кафедры систем информационной безопасности Казанского национального исследовательского технического университета им. А.Н. Туполева-КАИ

420015, Россия, республика Татарстан, г. Казань, ул. Большая Красная, 55

✉ Bulatusupov9@gmail.com



[Статья из рубрики "Модели и методы управления информационной безопасностью"](#)

### DOI:

10.7256/2454-0714.2023.2.43552

### EDN:

TFKJAS

### Дата направления статьи в редакцию:

10-07-2023

**Аннотация:** Статья посвящена организации учебного процесса с использованием специализированного лабораторного стенда охранной-пожарной системы. Главная цель работы заключается в формировании профессиональных компетенций у студентов в области систем безопасности и охраны. В работе детально описывается структура и компоненты лабораторного стенда, подчеркивая значение его ключевых элементов – приемно-контрольного прибора "Астра-812 Pro" и разнообразия извещателей. Это позволяет учащимся практически применять теоретические знания, изучая функционирование и взаимодействие различных составляющих охранно-пожарной системы. В статье также представлены схемы подключения извещателей к расширителю, позволяющие студентам более глубоко понять техническую сторону процесса. Шаги обучения на стенде раскрыты с целью поддержки пошагового, последовательного обучения. Исследование подчеркивает эффективность предложенной системы обучения, которая предполагает прямое взаимодействие студентов с практическим оборудованием. Выводы подтверждают значимость использования подобных обучающих стендов в современном высшем профессиональном образовании, подчеркивая их вклад в формирование необходимых профессиональных компетенций у будущих специалистов в

области охраны и безопасности. Научная новизна работы заключается в разработке и применении специализированного учебного стенда для обучения студентов работе с охранно-пожарными системами. Это позволяет учащимся получить непосредственный опыт взаимодействия с оборудованием, углубить понимание технических аспектов работы систем и улучшить свои практические навыки. Работа также вносит вклад в исследования в области образовательных подходов в области безопасности и охраны, предлагая эффективную методику обучения на основе использования лабораторного стенда. Это предоставляет платформу для дальнейших исследований и разработки в этом направлении, которые могут расширить и углубить образовательные возможности в данной сфере.

### **Ключевые слова:**

охранно-пожарная система, учебный стенд, обучение студентов, приемно-контрольный прибор, извещатели, система безопасности, профессиональные компетенции, схемы подключения, образовательные подходы, практические навыки

### **Введение**

Задача современного вуза – это подготовка будущих специалистов к осуществлению профессиональной деятельности в условиях производства и в настоящее время ориентирована на постоянно усложняющиеся требования со стороны рынка труда и услуг, которое включает обязательно достаточный уровень сформированности профессиональных компетенций и высокий уровень развития профессионально важных качеств и личностных свойств будущего выпускника [1]. В условиях цифровизации высшего профессионального образования концепция развития личности обучающихся требует пересмотр роли педагога и организацию специальной подготовки участников смешанного образовательного процесса [2]. Проведенный опрос авторами работы [3] подтвердил, что большая часть студентов готова к новому этапу развития учебных технологий, однако они обладают недостаточным объемом знаний и не до конца понимают всю сложность и многогранность цифровизации образования. Для повышения компетенции актуально использовании реальных лабораторных комплексов, на которых обучающиеся получают практический опыт работы [4,5,6]. В качестве примера рассмотрим созданный лабораторный стенд охранной-пожарной системы (далее ОПС) в рамках учебной дисциплины «Технические средства охраны».

### **Разработка стенда**

Лабораторный стенд представляет собой ОПС, построенную на основе приемно-контрольного прибора (далее ПКП) «Астра-812 Pro» (рис. 1), служащий для обработки данных с извещателей, передачи сигналов на пульт централизованного наблюдения (ПЦН) о состояние, оповещение служб охраны, а также контроля самих извещателей [7].

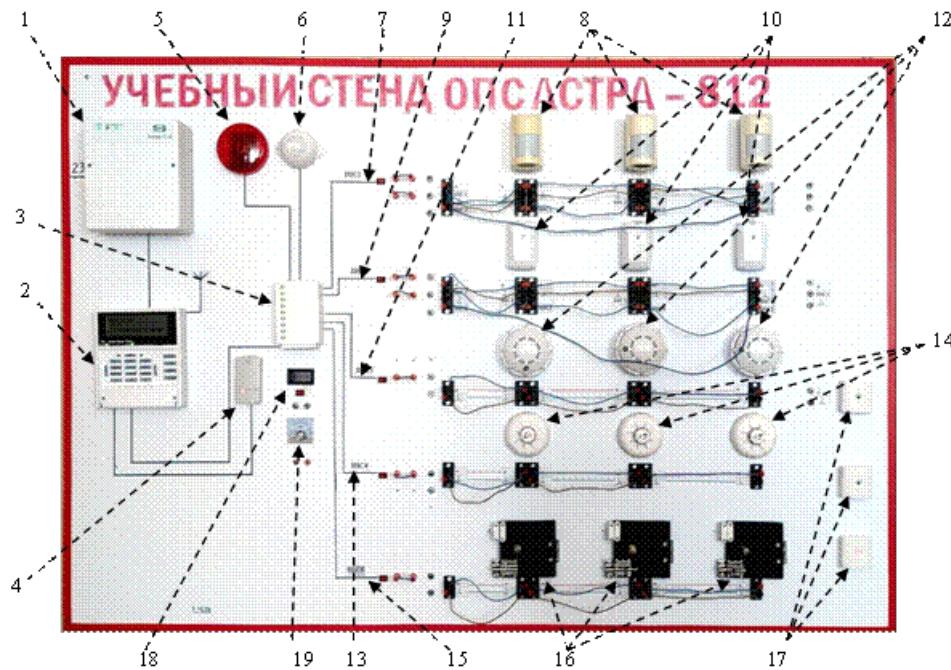


Рисунок 1. Учебный стенд ОПС АСТРА – 812

Лабораторный стенд состоит из следующих компонентов:

1. источник питания Астра-712/0, питающий все компоненты стенда;
2. приёмно-контрольный прибор Астра-812 Pro;
3. расширитель Астра-713, который подключен к ПКП;
4. считыватель ключей карт Matrix-II;
5. световой охранно-пожарный оповещатель Астра-10;
6. звуковой охранно-пожарный оповещатель Иволга;
7. шлейф сигнализации ШС1;
8. извещатели оптико-электронные ИО 409-26, подключенные к шлейфу ШС1;
9. шлейф сигнализации ШС2;
10. извещатели охранные поверхностные звуковые ИО 329-5, подключенные к шлейфу ШС2;
11. шлейф сигнализации ШС3;
12. извещатели пожарные дымовые оптико-электронные ИП 212-45, подключенные к шлейфу ШС3;
13. шлейф сигнализации ШС4;
14. извещатели пожарные тепловые ИП 103-5/4, подключенные к шлейфу ШС4;
15. шлейф сигнализации ШС5;
16. двери с извещателями охранными магнитоконтактными ИО 102-16/2, подключенные к шлейфу ШС5;

17. устройства контроля шлейфов УШК-01;
18. электронный вольтметр, для измерения напряжения шлейфов;
19. амперметр аналоговый, для измерения токов шлейфов.

На стенде есть зажимные разъёмы для подключения проводками извещателей в шлейф и выбора их количества от одного до трёх. Кроме того, на входных и выходных контактах шлейфа есть штыревые разъемы диаметром по 4 мм, для подключения различных устройств контроля и анализа, например, вольтметра и амперметра, подключив которые мы можем в реальном времени замерить вольтамперные характеристики шлейфов в различных режимах работы системы [\[8\]](#).

Обучающиеся подключают каждый из пяти шлейфов сигнализации и проверяют их работоспособность имитируя нарушения, после этого фиксируют результаты в отчётах. Рассмотрим схемы подключения шлейфов к расширителю Астра-713.

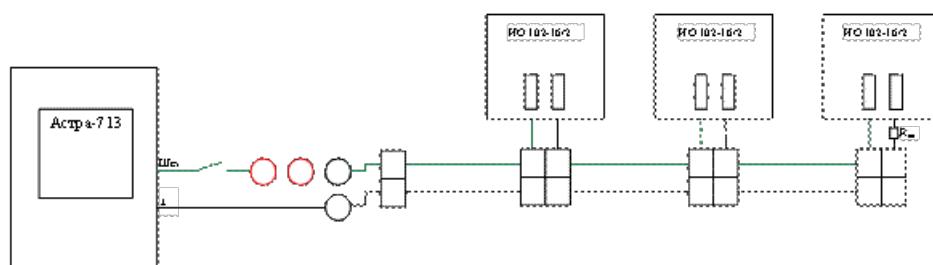


Рисунок 2. Схема подключения магнитоконтактных извещателей

Магнитоконтактные извещатели прикреплены к дверцам (рис. 2). Эти извещатели подключаются к пятому шлейфу. Для подключения используются два контакта: ШС5 и «земля», на конце шлейфа находится оконечный резистор. При открытии дверцы происходит отвод постоянного магнита от геркона и размыкание контакта [\[9,10\]](#).

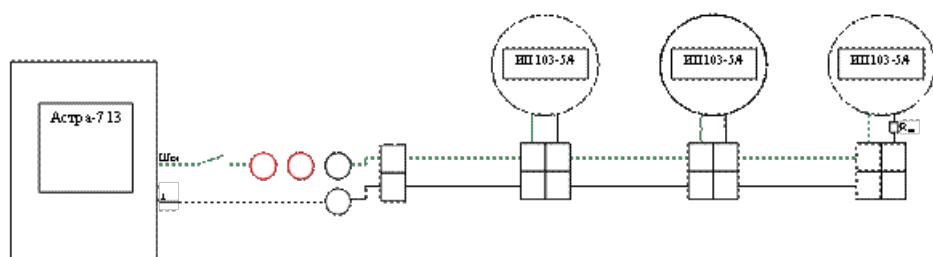


Рисунок 3. Схема подключения противопожарных тепловых извещателей

Противопожарный тепловые извещатели подключаются к четвёртому шлейфу (рис. 3). Для подключения используются два контакта: ШС4 и «земля», на конце шлейфа находится оконечный резистор. При нагреве термочувствительного элемента извещателя происходит разрыв шлейфа.

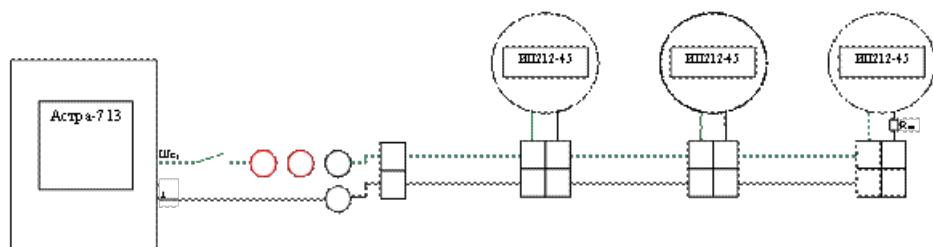


Рисунок 4. Схема подключения противопожарных дымовых извещателей

Противопожарный дымовые извещатели подключаются к третьему шлейфу (рис. 4). Для подключения используются два контакта: ШС3 и «земля», на конце шлейфа находится оконечный резистор. При попадании дыма на оптоэлектронную систему извещателя происходит изменения тока на фотоэлементе в камере извещателя. При этом извещатель замыкает шлейф, подключив свое внутреннее сопротивление.

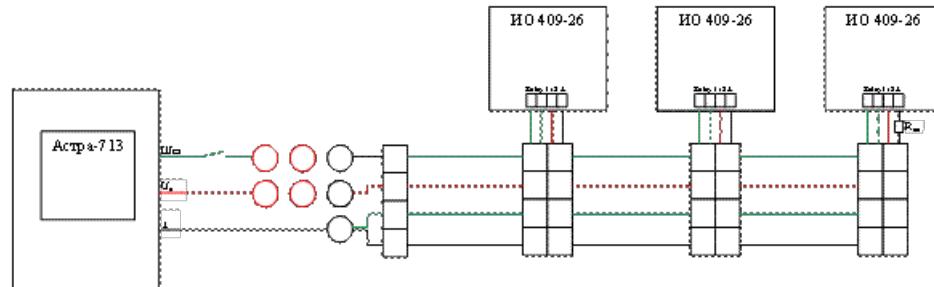


Рисунок 5. Схема подключения акустических извещателей

Акустические (пассивные) извещатели подключаются ко второму шлейфу (рис. 5). Для подключения используются уже три контакта: «питание», ШС2 и «земля» и четыре подключения между извещателями, где «земля» общая для питания извещателей и контрольного вывода ШС2. Оконечный резистор находится между выводами ШС2 с последнего извещателя и «земля». Извещатели контролируют разбитие стекла по характеристике звукового спектра в пределах 12 кГц, а также отфильтровывают другие спектры, например, речевые с целью исключить ложное срабатывание.

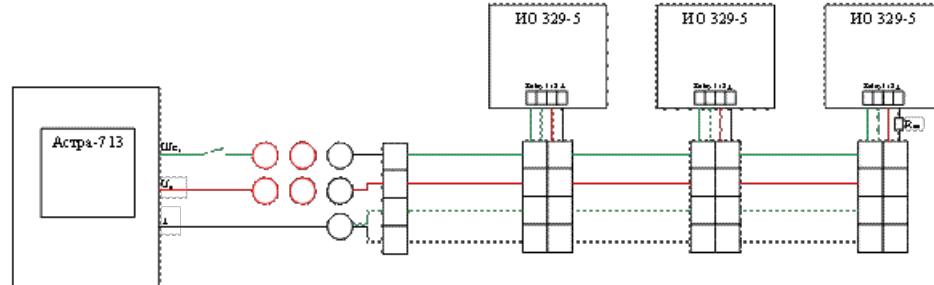


Рисунок 6. Схема подключения оптико-электронных извещателей

Оптико-электронных извещатели подключаются к первому шлейфу (рис. 6). Для подключения используются три контакта: «питание», ШС1 и «земля» и четыре подключения между извещателями, где «земля» общая для питания извещателей и контрольного вывода ШС1. Оконечный резистор находится между выводами ШС1 с последнего извещателя и «земля». Извещатели контролируют перемещение людей в зоне обзора по тепловому излучению.

### Методика проведения занятий

Для проведения обучения на данном учебном стенде были разработаны методики проведения занятий. Первое занятие направлено на изучение теоретического материала, ознакомление со стендом и оформления отчётов (рис. 7).

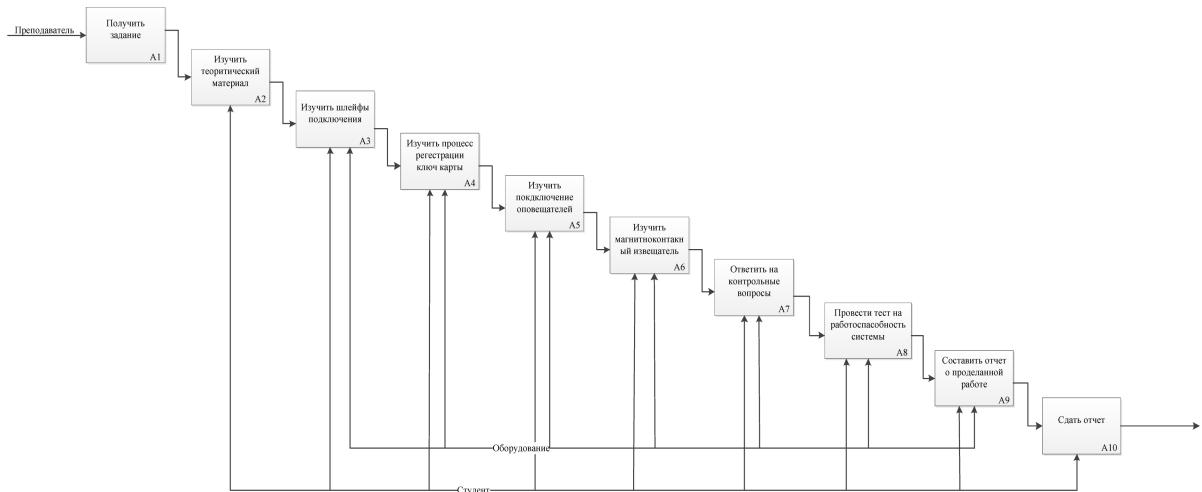


Рисунок 7. Методика проведения ознакомительного занятия

Второе занятие направлено на практическую работу, где обучающиеся работают непосредственно на стенде, подключая извещатели, шлейфы, настраивая ПКП и тестируя работу системы ОПС в целом, оформляя также отчёты (рис. 8).

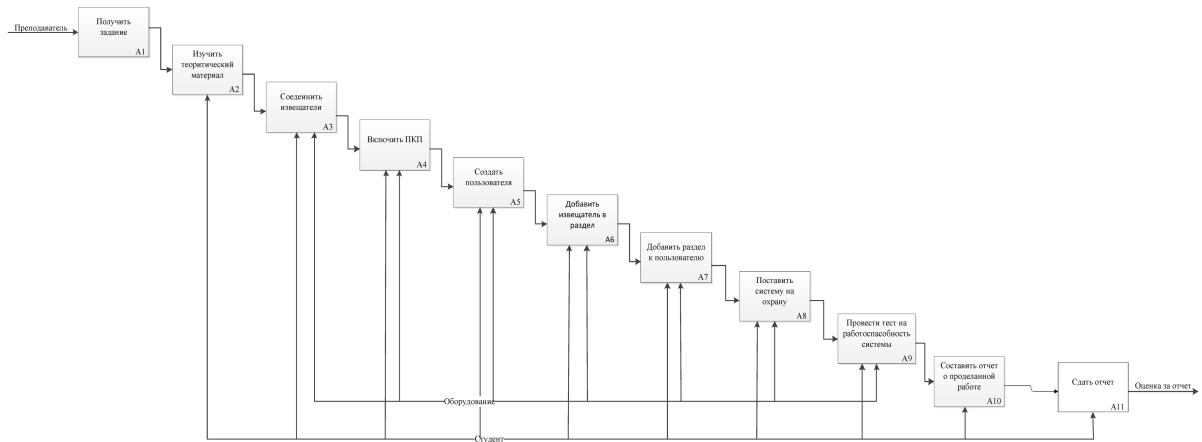


Рисунок 8. Методика проведения практического занятия

## Заключение

В данной статье продемонстрирована актуальность и эффективность разработанного учебного стенда охранно-пожарной системы в контексте формирования профессиональных компетенций студентов. Практические навыки, полученные в результате обучения на стенде, включают в себя знание схем подключения модулей, коммутацию извещателей в шлейфах, настройку системы, контроль состояния шлейфов, понимание принципов работы извещателей, а также особенности проектирования систем. Представленная разработка предлагает возможность подключения нескольких извещателей, что углубляет практическое взаимодействие студентов с системой и расширяет их компетенции. Этот подход усиливает понимание студентами реального функционирования и обслуживания систем охраны и пожарной безопасности, чем укрепляет их подготовку к будущей профессиональной деятельности. Эффективность разработанной системы подтверждена исследованиями, опубликованными в работе [6]. В этой работе авторы использовали стенд с одними извещателями на шлейфах, что также подтверждает значимость и практическую пригодность подобных подходов.

Таким образом, предложенный в статье учебный стенд для охранно-пожарной системы представляет собой эффективный инструмент для формирования профессиональных компетенций студентов, расширения их практических навыков и подготовки к будущей

профессиональной деятельности. Результаты исследования подтверждают значимость и практическую пригодность такого подхода. Однако, дальнейшие исследования в этом направлении могут быть продолжены в контексте интеграции современных информационных и вычислительных технологий в обучающие стенды. Как показали исследования [11,12,13], существует потенциал для улучшения процесса обучения с использованием распределенных вычислительных систем и методов параллельного программирования. Это может обеспечить еще более глубокое понимание студентами функционирования охранно-пожарных систем и обеспечить большую практическую пригодность обучения, особенно в контексте обработки больших данных и использования сложных алгоритмов. Мы считаем, что интеграция таких подходов в обучение на стенде может быть предметом дальнейших исследований в области профессионального образования.

## **Библиография**

1. Алексеенко О. И., Даниленко Т. В., Кирий Е. В. Проблемы обучения и воспитания студентов в современном вузе // Современное педагогическое образование. 2018. № 3. С. 3-5.
2. Фортова Л. К., Юдина А. М. Проблемы и перспективы развития цифровизации высшего профессионального образования // Вестник Владимирского государственного университета им. Александра Григорьевича и Николая Григорьевича Столетовых. Серия: Педагогические и психологические науки. 2022. № 51(70). С. 85-89.
3. Иванова О. В., Мороз И. Н. Возможности и проблемы цифровизации высшего образования // Высшее образование сегодня. 2022. № 5. С. 30-35.
4. Юсупов Б. З., Мартынов А. М. Разработка лабораторного стенда охранно-пожарной сигнализации по дисциплине технические средства охраны // Актуальные проблемы науки и образования в условиях современных вызовов : Сборник материалов XIX Международной научно-практической конференции, Москва, 21 марта 2023 года. С. 80-91.
5. Юсупов Б. З. Разработка лабораторного стенда охранно-пожарной сигнализации по дисциплине технические средства охраны // XXV Туполовские чтения (школа молодых ученых) : Международная молодёжная научная конференция, посвященная 60-летию со дня осуществления Первого полета человека в космическое пространство и 90-летию Казанского национального исследовательского технического университета им. А.Н. Туполева-КАИ. Материалы конференции. Сборник докладов. В 6-ти томах, Казань, 10-11 ноября 2021 года. Том V. Казань: Индивидуальный предприниматель Сагиева А.Р., 2021. С. 758-763.
6. Юсупов Б. З. Разработка методики проведения лабораторных работы на стенде «ОПС Астра-713» по дисциплине технические средства охраны // XXV Туполовские чтения (школа молодых ученых) : Международная молодёжная научная конференция, посвященная 60-летию со дня осуществления Первого полета человека в космическое пространство и 90-летию Казанского национального исследовательского технического университета им. А.Н. Туполева-КАИ. Материалы конференции. Сборник докладов. В 6-ти томах, Казань, 10–11 ноября 2021 года. Том V. Казань: Индивидуальный предприниматель Сагиева А.Р., 2021. С. 764-767.
7. Петик Н. С. Проектирование системы охранно-пожарной сигнализации // Молодость. Интеллект. Инициатива : Материалы VI Международной научно-практической конференции студентов и магистрантов, Витебск, 19 апреля 2018 года. Витебск: Витебский государственный университет им. П.М. Машерова, 2018. С. 33-34.

8. Бонч-Бруевич А. М., Кашпур Е. И. Исследование перспективных технологий цифровой модуляции в системах охранно-пожарной сигнализации // Спецтехника и связь. 2015. № 3. С. 24-28.
9. Sheikh S. M., Neiso M. K., Ellouze F. Design and implementation of a raspberry pi based home security and fire safety system // Computer Science & Information Technology (CS & IT). 2019. No. 3(3), P. 13.
10. Jakubowski K. Operational Analysis of Fire Alarm Systems with a Focused, Dispersed and Mixed Structure in Critical Infrastructure Buildings // Energies. 2021. No. 14(23), P. 7893.
11. Гибадуллин Р.Ф. Потокобезопасные вызовы элементов управления в обогащенных клиентских приложениях // Программные системы и вычислительные методы. 2022. № 4. С. 1-19.
12. Гибадуллин Р.Ф., Викторов И.В. Неоднозначность результатов при использовании методов класса Parallel в рамках исполняющей среды .NET Framework // Программные системы и вычислительные методы. 2023. № 2. С. 1-14.
13. Викторов И.В., Гибадуллин Р.Ф. Разработка синтаксического дерева для автоматизированного транслятора последовательного программного кода в параллельный код для многоядерных процессоров // Программные системы и вычислительные методы. 2023. № 1. С. 13-25.

## **Результаты процедуры рецензирования статьи**

*В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.*

*Со списком рецензентов издательства можно ознакомиться [здесь](#).*

Представленная на рассмотрение статья «Разработка учебного стенда охранно-пожарной системы для обучения студентов», предлагаемая к публикации в журнале «Программные системы и вычислительные методы», несомненно, является актуальной, ввиду обращения автора к особенностям профессиональной подготовки будущих специалистов и проблематике цифровизации в сфере обучения.

В условиях цифровизации высшего профессионального образования концепция развития личности обучающихся требует пересмотр роли педагога и организацию специальной подготовки участников смешанного образовательного процесса. В работе автор обращался к методу интервьюирования и последующей обработки полученных данных, что позволило сделать вывод о том, что большая часть студентов готова к новому этапу развития учебных технологий, однако они обладают недостаточным объемом знаний и не до конца понимают всю сложность и многосторонность цифровизации образования.

Исходя из данного высказывания автора и с учетом наименования журнала «Программные системы и вычислительные методы» в дальнейшем повествовании ожидается презентация некой цифровой или компьютерной модели, которая может использоваться в подготовке специалистов. Но, под цифровизацией автор понимает не компьютерное моделирование, а модель электронного устройства ОПС как элемента отработки теоретических знаний и практических навыков.

Методика проведения занятий с использованием данного пособия прописана очень кратко, никаких убедительных репрезентативных данных о повышении качества усвоемости материала по сравнению с контрольной группой автор не приводит. Собственно, поэтому вывод автора считаем не обоснованным: «предложенный в статье учебный стенд для охранно-пожарной системы представляет собой эффективный

инструмент для формирования профессиональных компетенций студентов, расширения их практических навыков и подготовки к будущей профессиональной деятельности».

Статья является новаторской, одной из первых в российской педагогике, посвященной исследованию подобной тематики в 21 веке. Исследование выполнено в русле современных научных подходов, работа состоит из введения, основной части и заключения. Отметим, что во введении не представлена степень разработанности проблематики, что важно для выявления новизны исследования и авторского вклада. Кроме того, во введении не обозначены цель и задачи исследования, что не позволяет сопоставить выводы по итогам работы с поставленными задачами. Выводы, представленные автором, не отражают проведенной работы и не подводят итога исследования и его дальнейших перспектив.

В основной части исследования отсутствует научный эксперимент и его описание как такового, методологическая основа использования предлагаемой разработки представлена слабо.

Библиография статьи насчитывает 13 источников, среди которых теоретические работы как на русском, так и на иностранном языках.

Из данных 13 источников 3 работы Гибатуллина, 3 работы Юсупова, что можно рассматривать как сужение обзора работ различных исследований и искусственное нагнетание цитирования.

К сожалению, в статье отсутствуют ссылки на фундаментальные работы, такие как монографии, кандидатские и докторские диссертации. Технически при оформлении библиографического списка нарушены общепринятые требования ГОСТа, а именно несоблюдение алфавитного принципа оформления источников, смешение работ на иностранном и русском языках.

В общем и целом, следует отметить, что статья написана простым, понятным для читателя языком. Часть материала представлены в виде диаграмм, схем и рисунков, что облегчает понимание текста читателем. Опечатки, орфографические и синтаксические ошибки, неточности в тексте работы не обнаружены. Работа является новаторской, представляющей авторское видение решения рассматриваемого вопроса и может иметь логическое продолжение в дальнейших исследованиях. Практическая значимость определяется возможностью использовать представленные наработки в дальнейших тематических исследованиях. Результаты работы могут быть использованы в ходе преподавания на специализированных факультетах. Статья, несомненно, будет полезна широкому кругу лиц, педагогам, магистрантам и аспирантам профильных вузов. Статья «Разработка учебного стенда охранно-пожарной системы для обучения студентов» может быть рекомендована к публикации в научном журнале после внесения изменений, а именно: 1) разведение понятия «цифровизация» и «электронное техническое устройство обучения», 2) усиление теоретической части исследования, 3) описание методологии и хода исследования в основной части, 4) актуализация библиографического списка и приведение его в соответствие с общепринятыми требованиями.

**Программные системы и вычислительные методы***Правильная ссылка на статью:*

Лаптев М.В., Янчус В.Э., Лаптев В.В. — Выявление зон интереса пользователя при визуализации данных с использованием метода ай-трекинга // Программные системы и вычислительные методы. – 2023. – № 2. DOI: 10.7256/2454-0714.2023.2.43502 EDN: TFRYGR URL: [https://nbpublish.com/library\\_read\\_article.php?id=43502](https://nbpublish.com/library_read_article.php?id=43502)

## **Выявление зон интереса пользователя при визуализации данных с использованием метода ай-трекинга**

**Лаптев Максим Владимирович**

инженер-проектировщик, ООО "ПСК "ФРАМ"

196128, Россия, Санкт-Петербург область, г. Санкт-Петербург, ул. Варшавская, 5а, оф. 4-Н/38

✉ lapt.maxim@gmail.com

**Янчус Виктор Эдмундасович**

кандидат технических наук

доцент Высшей школы дизайна и архитектуры, Санкт-Петербургский политехнический университет Петра Великого

195251, Россия, г. Санкт-Петербург, ул. Политехническая, 29

✉ victorimop@mail.ru

**Лаптев Владимир Владимирович**

ORCID: 0000-0002-7840-377X

доктор искусствоведения

профессор Высшей школы дизайна и архитектуры, Санкт-Петербургский политехнический университет Петра Великого

195251, Россия, Санкт-Петербург, г. Санкт-Петербург, ул. Политехническая, 29



✉ laptevsee@gmail.com

[Статья из рубрики "Компьютерная графика, обработка изображений и распознавание образов"](#)**DOI:**

10.7256/2454-0714.2023.2.43502

**EDN:**

TFRYGR

**Дата направления статьи в редакцию:**

04-07-2023

**Аннотация:** Предметом данного исследования является особенность формообразования

в визуализации данных. Для этого авторы выдвигают гипотезу о существовании особых зон интереса в диаграммах. На них пользователь обращает внимание с целью декодирования данных, зашифрованных графикой. Наличие таких площадей, а в некоторых случаях и точек, интуитивно определены в ходе формирования проектных правил информационной графики. Для их верификации в исследовании был использован метод ай-трекинга и метод кластерного анализа. Применение междисциплинарных принципов и правил проектирования инфографики изучено на примере горизонтальных и вертикальных брусковых диаграмм, секторных, фигурных и потоковых диаграмм. Результат экспериментов с различными типами диаграмм показал наличие дополнительных зон интереса, ранее не указанных специалистами визуализации данных. Это позволяет уточнить особенности графических форм и формообразования диаграмм, позволяет верифицировать использование проектных правил, сформулированных усилиями отечественных и зарубежных специалистов в конце XIX – начале XX вв. Такая верификация может быть сделана по методике, включающей в себя статистические методы и метод ай-трекинга, что позволяет учесть ценный опыт прошлого в современном дизайне информации. Представленная методика может быть распространена на другие виды диаграмм и тематических карт, а также иметь практическое применение при анализе визуализации больших массивов данных.

#### **Ключевые слова:**

визуализация данных, инфографика, метод ай-трекинга, зона интереса, потоковая диаграмма, фигурная диаграмма, брусковая диаграмма, секторная диаграмма, кольцевая диаграмма, формообразование

#### **Введение**

Визуализация данных — это часто встречающаяся задача, которую решает современный дизайнер. В его распоряжении имеется широкий спектр графических форм, в которые он превращает числа, их ряды и массивы. Преобразование данных в графику сталкивается с рядом проблем, связанных с выбором формы визуализации и ее корректным отображением. В подавляющем большинстве случаев возникает вариативность формообразования из-за имеющегося в наличии широкого спектра диаграмм. Окончательный выбор формы такой визуализации носит ярко выраженный творческий характер и относится к дизайну как проектно-художественной деятельности. При этом форма должна вписываться в выбранный стилистический концепт, раскрывать архитектуру инфографики как содержательного образа и потому должна поддерживать ее функциональность. Несмотря на это выбор формы визуализации и ее трансформация до сих пор вызывают трудности у дизайнера, что иногда приводит к неудовлетворительным результатам. По этой причине данное исследование имеет актуальность.

Выявление особенностей восприятия различных форм визуализации данных с определением особых зон интереса пользователя является целью настоящего исследования. Для ее достижения необходимо решить ряд задач: рассмотреть роль формы и формообразования в визуализации данных; изучить возможности метода окулографии (ай-трекинга) для выявления особых кластеров — зон интереса пользователя; выявить возможное влияние таких зон на проектные правила формирования диаграмм различного типа. Объекты исследования — графики и диаграммы — относятся к инфографике как к направлению коммуникативного дизайна.

Предмет исследования — особенность формообразования в визуализации данных — рассматривается и изучается с позиции информационных технологий, примененных к компьютерной графике, которая наглядно представляет числовые данные. Поэтому, данная работа носит комплексный междисциплинарный характер.

## **1. Форма и формообразование в визуализации данных**

Выраженная посредством диаграммы (или карты, или схемы) идея является одним из приоритетных принципов построения информационной графики. Для визуализации данных в системе координат одну и ту же числовую последовательность можно графически выразить несколькими способами. Для этого существуют линейная, брусковая, секторная, плоскостная или фигурная диаграмма. Если существует географическое распределение данных, то вместе с перечисленными способами в число изобразительных средств необходимо включить картограмму и картодиаграмму. «Важно определить точные, соизмеримые критерии, которые можно использовать для классификации, выбрать лучший для данного случая и объяснить, почему читатели предпочитают разную форму графика» [\[1, с. 139\]](#). Поэтому основной проблемой, стоящей перед информационным дизайнером в начале проекта, является правильный выбор формы визуализации, которая максимально соответствует идеи графика. Учет функциональности, композиционного окружения графика и семантической значимости изображения также входит в ряд решаемых им задач.

Вариативность выбора типа диаграмм может представлена на примере визуализации структуры данных. В качестве решения могут быть использованы и брусковые, и секторные, и фигурные, и потоковые диаграммы, и древовидные карты (tree maps). Во всех случаях пользователь сравнивает целое (100%) и его доли, а также эти доли между собой. Смысловая нагрузка может определить приоритеты выбора. В случае динамического контента уместно применение потоковых диаграмм, визуально реализующих принцип «движения» данных. Для статических (по своей сути) данных в зависимости от их количества и пропорциональных соотношений выбираются секторные, кольцевые или брусковые диаграммы. При преобладании дихотомного построения структура данных визуализируется при помощи древовидных карт. А если имеется возможность дискретного представления долей, то может быть осуществлен выбор в сторону фигурных количественных диаграмм. У каждого типа визуализации структуры данных имеются собственные ограничения и проектные правила, сформированные в процессе становления информационной графики.

Для анализа полученного результата необходимо использовать качественные и количественные методы оценки с учетом специфики инфографики. Качественные методы традиционны для исследования предметов дизайна. Это экспертное изучение и выявление методом опроса наглядности, функциональности, эстетической привлекательности плаката или рекламного сообщения, диаграммы или тематической карты. Связанные с проектированием инфографики вопросы тесным образом переплетаются с анализом готовых проектов на предмет их эффективности и релевантности. Эффективность инфографики, следуя Бертену, определяется временем, затраченным на визуальное восприятие и понимание графика: «Если для того, чтобы получить правильный и полный ответ на поставленный вопрос, при прочих равных условиях, один график требует более короткого времени наблюдения, чем другой, то можно сказать, что для этого случая он является более эффективным» [\[1, с. 139\]](#). Следовательно, в отношении информационной графики для оценки ее эффективности могут включаться статистические методы: количественной оценке подлежат такие

показатели как скорость решения задачи и точность анализа представляемых данных.

В непосредственной близости от количественной оценки эффективности визуализации данных находится проблема эргономики формы диаграмм. Простота восприятия, учет контекста и семантических связей между числовым массивом и его графическим образом влияют на удобочитаемости инфографики. В результате проводимых ранее исследований [2, 3] были получены результаты, что элементы визуальной структуры могут в определенной степени предсказуемо влиять на семантическую интерпретацию данных, которая выходит за пределы простого считывания данных. Поэтому применение каждого из имеющихся типов диаграмм сопряжено с определенными ограничениями, связанными с особенностями визуального восприятия, и семиотической спецификой.

Однако методы, в основе которых лежит только анализ точности и скорости восприятия данных или качественная (но приблизительная) оценка, не рассматривают набор данных в целом как паттерн, состоящий из отдельных элементов коммуникативного дизайна. Для выяснения того, как пользователь оценивает общую композицию визуализации данных или ее отдельные формы, должна быть принята во внимание глазодвигательная активность человека: направление взгляда, количество саккад и длительность его фиксаций.

## **2. Изучение вопросов формообразования в инфографике с использованием метода ай-трекинга**

Известно, что методы отслеживания глаз применялись для понимания взаимодействия частей графа [4]; для выявления различий в чтении линейных и брусковых диаграмм [5], секторных диаграмм и древовидных карт [6]; для оценки влияния макета на понимание дизайна диаграмм [7]. Использование метода ай-трекинга позволило ввести рекомендации по ограничению поля и числа линейных диаграмм в информационных конsoleях-дашбордах управления технологическими процессами [8]. Исследование транспортных схем как объекта анализа глазодвигательной активности дало возможность оценить приемы дополнительных визуализаций для обнаружения закономерностей или тенденций [9].

Здесь можно заметить, что изучению подлежат определенные фазы глазодвигательной активности: дрейфы, фиксации, варианты саккад, нистагмы. Интерес представляют, во-первых, зрительные фиксации (плавное перемещение глаза в небольшой зоне), во-вторых, саккады (скачкообразные движения высокой скорости, при которых резко изменяется позиция глаза). Считается, что зрительная информация обрабатывается именно в момент фиксации [10]. Окуломоторной активностью принято называть то, как человек рассматривает графическое изображение, читает и решает зрительные задачи. Движения глаз показывают, как формы просматриваются и интерпретируются. Определяемые текущей задачей намерения зрителя также влияют на движения и фиксации глаз. Значит глазодвигательная активность детерминирована текущей задачей или общей целью [11]. Таким образом длительность фиксации и амплитуды саккады могут использоваться в качестве индикаторов интерпретации формы во время задач поиска [12].

Общая композиция изображения воспринимается через рассматривание отдельных пространственных площадей, форм, узлов и фигур. Сложный объект визуально разделяется на более простые части, согласно композиционной структуре и

пространственных отношений между частями [13, 14]. Существует анализ паттернов рассматривания на предмет обнаружения значимых точек (Point of Interest, POI), ограниченных зон или площадей (Area of Interest, AOI) или частей изображения (Region of Interest, ROI), вызывающих интерес у зрителя. Это позволяет определить критически важные формы (или их части) для интерпретации данных.

Визуализация, основанная на выявление точек фиксации взгляда (point-based visualizations), служит для непосредственного показа деталей глазодвигательной активности. Методы, основанные на анализе зон интереса пользователя (AOI-based methods), обращаются к семантической информации стимула [15]. Совмещение двух способов анализа движения глаз, представленное в исследованиях [16], снижает субъективность определение зон интереса (AOI) за счет использования точек интереса (POI) при сохранении возможности интерпретации. Таким образом движения глаз можно использовать для анализа визуализации данных, оценки влияния типа диаграмм и их формы на эффективность решения задачи (правильность и скорость) и на поведение пользователя при решении задачи.

В ранее произведенных исследованиях формообразования инфографики [2, 17, 18, 19], рассматривались следующие индикаторы восприятия визуализации данных: (1) общее время решения задач; (2) средняя длительность фиксации; (3) среднее количество фиксаций; (4) количество правильных ответов при решении задач; (5) паттерны рассматривания. С их помощью решались задачи верификации применения междисциплинарных принципов и правил проектирования инфографики, определенных ранее отечественными и зарубежными специалистами.

Полученные результаты с использованием метода ай-трекинга подтвердили функциональность формы фигурных диаграмм при изменении изобразительности знака в фигурных диаграммах [21]. Здесь было определено отсутствие влияния изобразительности символов на длительность и производительность при решении задач на восприятие фигурных диаграмм, а также на паттерны восприятия (модели движения глаз). Однако, дополнительно было выявлено, что композиция фигурных диаграмм (дизайн-визуализации) оказывает существенное влияние на паттерны восприятия. При этом формируются устойчивые зоны интереса респондента.

В других исследованиях [17, 18] подтверждены правила построения потоковых диаграмм. Эти правила были сформулированы еще в XIX веке [20] в части поддержания стабильной толщины непрерывного потока, единообразного формирования корней потоков, пространственной компоновки узлов, целостности композиции при взаимодействии потока с внешними ориентирами. Здесь использовался метод ай-трекинга совместно с кластерным анализом, что позволило детектировать площади интереса при рассматривании потоковых диаграмм.

На основании полученных результатов стало возможным выполнить проектирование потоковой диаграммы путей пользователей, которая является прикладным инструментом веб-аналитики, альтернативной существующей системы Google Analytics. Были сделаны выводы о существенном влиянии вида диаграммы на среднее количество правильных ответов в результате проанализированных данных на воздействие разных факторов (в том числе субъективной общей нагрузки) на эффективность решения задачи и на поведение при ее решении. Следует отметить, что по таким параметрам, как скорость решения задач и количество правильных ответов, разработанный вариант также эффективнее, чем оригинальная диаграмма Google Analytics. Полученные данные можно

использовать для повышения качества работы специалистов по веб-аналитике, а также в таких областях, как разработка интерфейсов веб-приложений и исследование юзабилити.

Исследование различных видов брусковых и секторных диаграмм [19] посвящено сравнению эффективности восприятия различных типов диаграмм визуализации структуры данных, изучению особенности формирования паттерна рассмотривания с использованием метода ай-трекинга. В итоге оказалось, что брусковые и кольцевые диаграммы проще оцениваются при быстром рассмотрении, тогда как применение секторных диаграмм приводит к их более детальному изучению. Для визуализации одиночной статической структуры данных стоит отдавать предпочтение брусковому типу диаграмм. Секторные диаграммы следует выбирать только в том случае, если идет обращение к ассоциативным формам. Все эти тезисы были подтверждены не только благодаря анализу эффективности восприятия визуализации данных структурного характера, но и при использовании кластерного метода анализа для формирования зон интереса.

Результаты приведенных исследований говорят о необходимости обнаружения зон (или точек) интереса и интерпретации их значения для визуализации данных. Это способствует определению эффективности выбора формы визуального образа. Методика такого обнаружения базируется на кластерном анализе глазодвигательной активности человека с помощью системы окулографии (ай-трекинга). Это позволяет включить в исходные параметры не только точность и скорость определения параметров, но и зарегистрировать направление взгляда наблюдателя, длительность его фиксации, определить зоны интереса для последующей интерпретации.

### **3. Определение АОI пользователя на основе кластерного анализа**

Анализ глазодвигательной активности является сложной и нетривиальной задачей, в ходе решения которой требуется учесть сразу несколько факторов одновременно. Это координатное положение точки взора и их группировка, протяженность саккады, последовательность движения взгляда, длительность и количество фиксации, и другие параметры. Такие виды задач как временная или координатная группировка при анализе данных ай-трекинга можно решать с помощью кластерного анализа [17, 21, 22]. При этом основной трудностью группирования объектов в ранее неизвестные группы является отсутствие заранее известного количества кластеров и их характеристик.

В ай-трекинговом исследовании потоковых диаграмм [17] проблема определения числа кластеров решалась посредством иерархических дивизимных алгоритмов. Они работали с универсальным кластером, состоящим из всех зрительных фиксаций, с последующим пошаговым разбиением его на меньшие части по назначенным признакам. В качестве аппарата кластеризации был выбран метод k-средних с многомерным вектором, основанным на координатном построении. Визуализация результата при 4-кластерном разбиении представлена на рисунке 1 с демонстрацией стимульного материала одного из заданий и точек фиксации взгляда испытуемых. Учет семантических характеристик диаграммы-стимула дает возможность уточнить наличие точек интереса как центров формирования зон интереса пользователя.

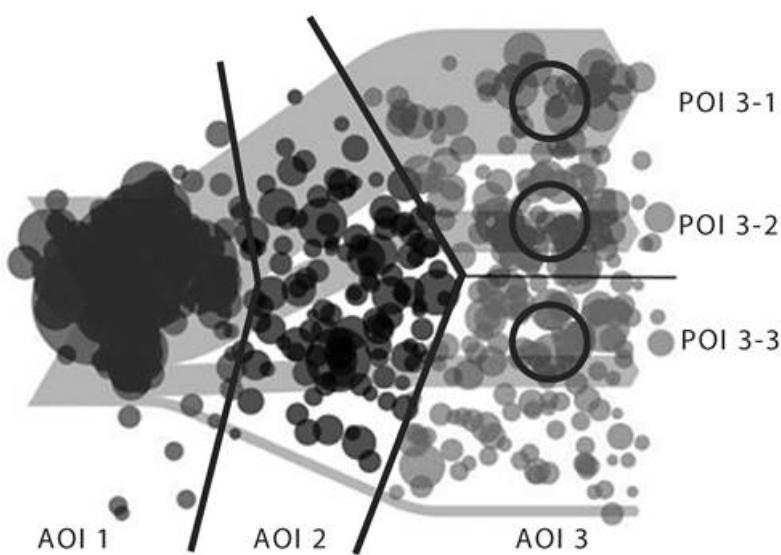


Рис. 1. Визуализация кластерной модели, соотнесенной с фиксациями взора. Выделены зоны интереса пользователя (AOI), полученные на основании кластерного анализа с учетом семантики стимула

Fig. 1. Visualization of a cluster model correlated with eye fixations. The areas of user interest (AOI) obtained on the basis of cluster analysis considering the semantics of the stimulus are highlighted

Алгоритмы кластеризации [15, 17] и семантическая группировка точек интереса позволяют выявить три основные площади интереса, основанные на координатах взгляда и длительности фиксации. Одна группа фиксаций сконцентрирована во входящей части стимула (потоковой диаграммы) с левой стороны. Две выходящие группы располагаются в его правой части. Они объединяются в единую площадь интереса по семантической причине. Обнаруживается устойчивый кластер, расположенный в центральной части стимула.

Каждая часть структуры сравнивается с представленными вариантами данных последовательно от верхней правой части и далее вниз. Этим обусловлена правая зона интереса пользователя (АОИ 3). Сравнение также идет каждой части с целым, представляемым «входным» кластером (АОИ 1). Центральная зона интереса (АОИ 2) интерпретирует сравнение частей между собой в динамической части, когда разделение на части уже есть, но оно еще незначительно. В этом случае потоки композиционно объединяются в единый блок. Очевидно, прослеживается нарратив при рассматривании потоковой диаграммы, выявляющий последовательность движения потока, от целого к его структурным частям.

В результате анализа визуальной оценки кластерной модели можно сделать заключение, что кластеры располагаются в соответствии со структурой стимула в виде 4-потоковой диаграммы. То есть, глазодвигательная активность детерминирована текущей задачей или общей целью [11]. Выходной кластер имеет явно выраженные три точки интереса (POI 3-1, 3-2, 3-3), а на меньшем (четвертом) потоке диаграммы такая точка интереса отсутствует. Это обусловлено задачей и предлагаемыми наборами ответов: пользователь не имеет затруднений в ответе на вопрос о данных, визуально представляемых этим явно наименьшим потоком.

Интерпретация этих результатов позволила дать рекомендацию по форме потоковой диаграммы. Во-первых, существует индифферентность визуального восприятия к

разделению входного потока на его составные части в корневой начальной области интереса (АОИ 1). Это позволяет безболезненно использовать выбор цвета отдельных потоков в статических потоковых диаграммах или в интерактивных версиях потоковых диаграмм. Во-вторых, деформация формы потока в средней части имеет решающее значение для визуального восприятия, поскольку происходит изменение поведенческих реакций, связанных с потерей смысловых связей между процессом и методом визуализации. Например, предварительный анализ потоковых диаграмм пользовательских путей в Google Analytics (рис. 2) показал, что одним из недостатков визуализации является нестабильная толщина формы потоков [18].

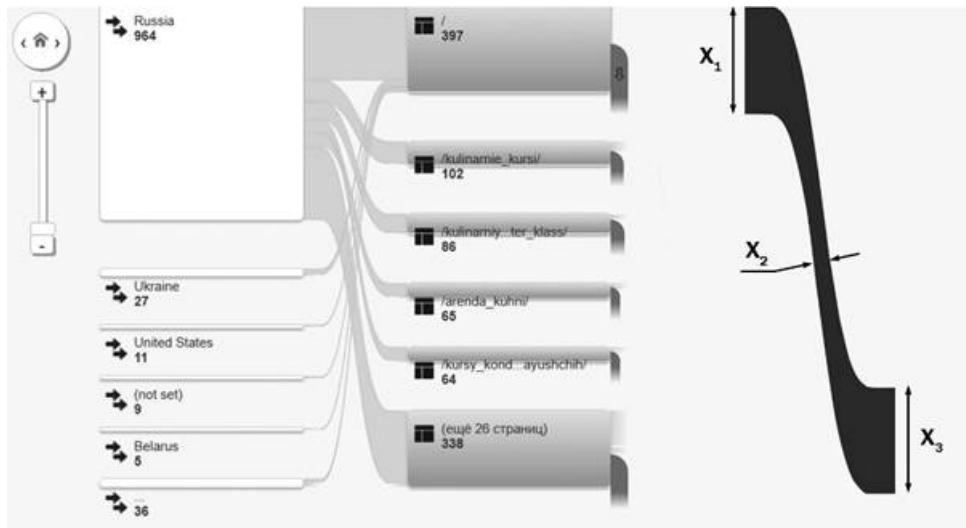


Рис. 2. Потоковая диаграмма пользовательских путей Google Analytics (фрагмент показывает изменение толщины потока в средней части в несколько раз)

Fig. 2. Flow chart of Google Analytics user paths (the fragment shows a change in the thickness of the flow in the middle part several times)

Следующий пример анализа формы визуализации данных продемонстрирован в исследовании [19], целью которого является сравнение эффективности восприятия различных типов диаграмм структуры данных. Дополнительно изучаются особенности формирования паттерна рассматривания при помощи метода ай-трекинга. В эксперименте рассматривались 4-частные структуры данных с доминирующей долей (65–75%) и тремя относительно небольшими долями. В качестве инструмента визуализации структуры данных используются вертикальные и горизонтальные брусковые диаграммы, секторные и кольцевые диаграммы.

Интересные результаты получены для секторных и кольцевых диаграмм. Используя кластерный анализ паттерна рассматривания, мы получили точки интереса пользователя ROI 1–3 в области дуг малых секторов (рис. 3). Это справедливо и для секторных, и для кольцевых диаграмм. Кроме этого, в центре круга имеется точка интереса ROI 4, центр которой совпадает с центром диаграммы. В итоге можно увидеть зону интереса АОИ в виде вытянутого эллипса для секторной диаграммы и ее концентрическую форму для кольцевой диаграммы.

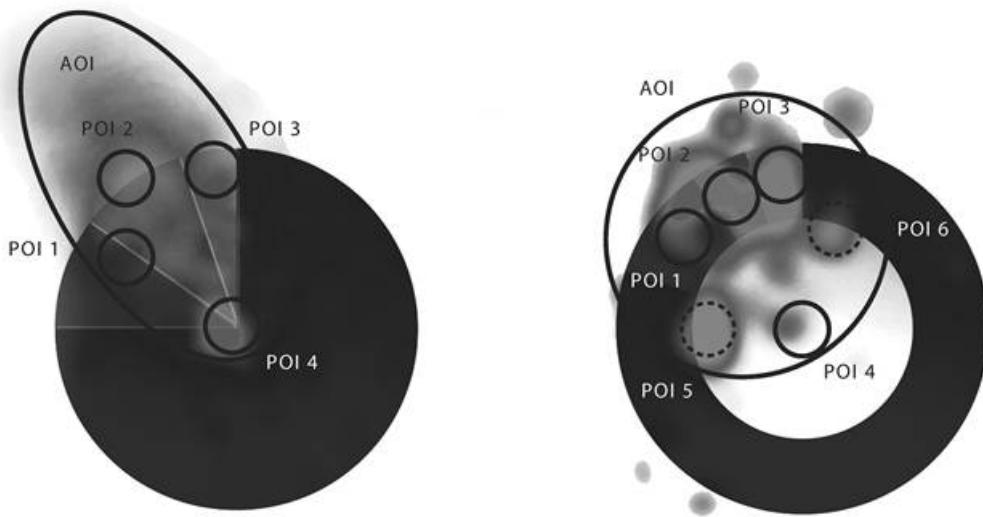


Рис. 3. Формирование точек и зоны интереса пользователя (POI, AOI), полученные на основании кластерного анализа с учетом семантики стимула для секторной диаграммы и для кольцевой диаграммы

Fig. 3. Formation of the user's points and area of interest (POI, AOI), obtained on the basis of cluster analysis, considering the semantics of the stimulus for the pie chart and for the doughnut chart

Если для секторной диаграммы это свидетельствует о возможном сравнении угловых величин секторов или об опорной точке отсчета, то для кольцевой диаграммы угловое сравнение отсутствует. Значит человеческий глаз ищет опорную точку в центре круговой диаграммы для визуального сравнения дуг секторов. Это подтверждается исследованиями восприятия различных вариантов формы кольцевых диаграмм [23]. При добавлении опорной точки в центре диаграммы эффективность ее визуального восприятия повышалась. Кстати, этим можно объяснить изменение формы паттерна рассматривания кольцевой диаграммы относительно секторной диаграммы. Отсутствие центральной точки приводит к тому, что появляются дополнительные точки интереса POI 5 и POI 6, которые соответствуют границам большого сектора. А эллипсовидная зона интереса превращается в концентрическую зону интереса.

Изучение формы брусковой диаграммы, которая визуализирует структуру данных — это следующий пример использования метода окулографии. Мы можем видеть различную стратегию рассматривания вертикальных и горизонтальных брусковых диаграмм (рис. 4). Так, паттерн рассматривания вертикальной диаграммы показывает, что пользователь изучает только малые доли структуры в ее верхней части. При этом не обращает внимания на доминирующий сегмент структуры, расположенный в нижней части бруска. Напротив, рассматривание горизонтальной диаграммы структуры данных вызывает появление двух зон интереса. Первая зона интереса (AOI 1) формируется за счет изучения зрителем доминирующей доли, а вторая возникает при рассматривании малых долей. Очевидно, что в случае горизонтальных брусковых диаграмм имеется явно выраженное сравнение частей структуры между собой. А в вертикальном бруске основная доля структуры фиксируется фовиальным зрением.

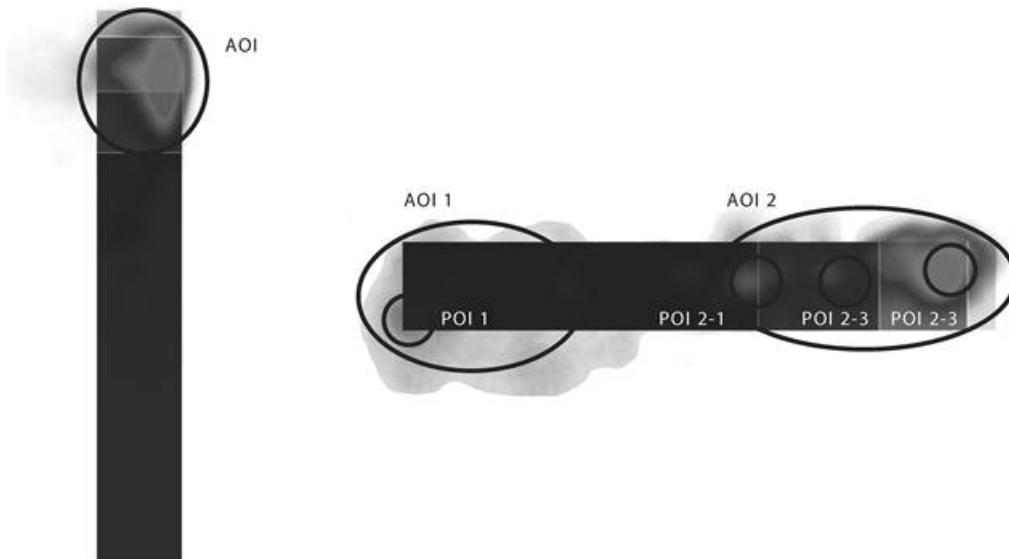


Рис. 4. Формирование точек и зон интереса пользователя (POI, AOI), полученные на основании кластерного анализа с учетом семантики стимула для вертикальной и горизонтальной брусковой диаграммы

Fig. 4. Formation of points and areas of interest of the user (POI, AOI), obtained on the basis of cluster analysis, considering the semantics of the stimulus for the vertical and horizontal bar chart

Дополнительно, сделанный анализ подтверждает ранее выполненное эмпирическое исследование, согласно которому даже небольшие изменения, как округления вершины прямоугольного бруска, приводят к повышению коэффициента ошибок [24]. Это утверждение можно экстраполировать на значащие границы частей в самих брусковых диаграммах. Горизонтальные границы для вертикальной брусковой диаграммы и вертикальные границы для горизонтальной брусковой диаграммы должны быть контрастно визуализированы и не иметь искажений. Нижняя граница вертикального бруска становится исключением. Она может быть визуализирована с меньшей тщательностью, например, размыта или деформирована. Но это предположение в свою очередь требует дополнительных сравнений и исследований.

#### **4. Обсуждение и выводы**

Итоги исследований, представленные здесь, основаны на методе окулографии и методе кластеризации, обусловленными детерминированным поведением. В результате мы ожидаем, что они станут полезными инструментами изучения форм визуализации данных для обнаружения и анализа областей, представляющих интерес для зрителя. Это справедливо как для использования в экспериментальном анализе, так и для автоматизации решений по контенту в интерактивных системах с использованием айтреクинга. Этот подход наиболее применим в ситуациях, связанных с просмотром изображения или сцены, неограниченным по времени и в пространстве, например, по примеру ранее выполненных исследований композиции кинокадра [25]. Метод айтреクинга может применяться для исследования эргономики визуального восприятия плоскостных моделей трехмерных объектов, например, рельефов поверхностей в картографии и гидрологии, в ультразвуковом и лазерном зондировании, в других прикладных инженерных задачах.

За рамками настоящего исследования остались вопросы цветового кодирования при

визуализации массивов информации, формирования рекомендаций по колористике диаграмм. Визуальное восприятие отдельных цветов, групп контрастов и колористических схем остается актуальной задачей. Это требует дальнейшего изучения, в том числе с использованием метода ай-трекинга и кластерного метода анализа полученных результатов, в основу которого может быть положено определение точек интереса и зон интереса пользователя.

Нам необходимо сказать, что кластерный анализ проводился на основе одного из возможных признаков: координатного расположения фиксаций глазодвигательной активности. В настоящей статье не рассматривалась кластеризация, основанная на принципах оценки близости или порядка следования саккад, продолжительности фиксаций и так далее. Использование различных принципов кластерного анализа для обработки данных глазодвигательной активности требует дальнейшего отдельного исследования.

Возможно, интересными являются новые экспериментальные протоколы, алгоритмы кластеризации и интерактивные системы по определению зон интереса. Их местоположение и свойства можно использовать для изучения особенностей визуального восприятия различных групп зрителей, сходства и различий решаемых задач или использования разных версий изображений. Это можно было бы сделать как в координатном пространстве, сравнив центроиды кластера, созданные разными пользователями или их группами, так и по временной шкале. Относительное количество точек данных внутри кластера указывает на его важность для определения зон интереса. Это может сделать кластерный метод полезным для количественной оценки визуального интереса. Алгоритмизация анализа кластерных моделей позволяет перевести визуальную интерпретацию структур числовых данных в круг задач поддержки принятия решений, решаемых с помощью программных средств.

Применение кластерного метода анализа результатов ай-трекинговых исследований в отношении средств визуализации имеет непосредственное значение для информационного дизайна. Это позволяет уточнить особенности графических форм и формообразования диаграмм, позволяет верифицировать использование проектных правил, сформулированных усилиями отечественных и зарубежных специалистов в конце XIX — начале XX вв. Такая верификация может быть сделана по методике, включающей в себя статистические методы и метод ай-трекинга, что позволяет учесть ценный опыт прошлого в современном дизайне информации.

## **Библиография**

1. Bertin J. *Semiology of Graphics. Diagrams. Networks. Maps.* Redlans: Esri Press, 2011. 438 p.
2. Laptev V. V., Orlov P. A. The Eye-Tracking Study of Effects of the Stylisation Level in Pictorial Charts // *Humanities and Science University Journal.* 2016. № 19. P. 44–56.
3. Ziemkiewicz C., Kosara R. Implied Dynamics in Information Visualization // *Proceedings Advanced Visual Interfaces (AVI).* 2010. P. 215–222.
4. Huang W. Using eye tracking to investigate graph layout effects // *Proceedings of the 6th Asia-Pacific Symposium on Visualisation.* 2007. P. 97–100.
5. Goldberg J. H., Helfman J. I. Comparing information graphics: a critical look at eye tracking // *Proceedings of the 3rd BELIV'10 Workshop: Beyond time and errors: Novel evaluation methods for information visualization.* 2010. P. 71–78.
6. Huestegge L, Pötzsch T. H. Integration processes during frequency graph

- comprehension: Performance and eye movements while processing tree maps versus pie charts // Applied Cognitive Psychology. 2018. 32 (2). P. 1–17. DOI: 10.1002/acp.3396
7. Sharif B., Maletic J. I. An empirical study on the comprehension of stereotyped UML class diagram layouts // Proceedings of the IEEE 17th International Conference on Program Comprehension. 2009. P. 268–272.
  8. Orlov P., Ermolova T., Laptev V., Mitrofanov A., Ivanov V. The Eye-tracking Study of the Line Charts in Dashboards Design // VISIGRAPP 2016. Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. Volume 2: IVAPP. Rome – Italy, February 27–29, 2016. P. 205–213.
  9. Burch M., Wallner G., Broeks N., Piree L., Boonstra N. The Power of Linked Eye Movement Data Visualizations // ETRA '21. ACM Symposium on Eye Tracking Research and Applications. 2021. P. 1–11. DOI: 10.1145/3448017.3457377
  10. Величковский В. М. Когнитивная наука: основы психологии познания. М.: Смысл : Академия, 2006. Т. 2. 432 с.
  11. Yarbus A. L. Eye Movements and Vision. Plenum Press, 1967. DOI: 10.1007/978-1-4899-5379-7
  12. Prats M., Garner S., Jowers I., McKay A., Pedreira N. Interpretation of geometric shapes – an eye movement study // Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications. 2010. P. 243–250.
  13. Biederman I. Recognition by components: A theory of human image understanding // Psychological Review. 1987. Vol. 94 (2). P. 115–147.
  14. Hoffman D. D., Richards W. A. Parts of recognition // Cognition. 1984. Vol. 18 (1–3). P. 65–96.
  15. Rim N. W., Choe K. W., Scrivner C., Berman M. G. Introducing Point-of-Interest as an alternative to Area-of-Interest for fixation duration analysis // PLoS One. 2021. No. 16 (5). URL: <https://pubmed.ncbi.nlm.nih.gov/33970920/> (data access: 24.02.2022). DOI: 10.1371/journal.pone.0250170
  16. Blascheck T. et al. AOI hierarchies for visual exploration of fixation sequences // Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications. 2016. P. 111–118.
  17. Лаптев В. В., Орлов П. А. Кластерный анализ визуального восприятия структуры данных // Бизнес-информатика. 2015. № 3 (33). С. 34–43.
  18. Лаптев В. В., Орлов П. А., Драгунова О. В. Визуализация динамических структур данных с помощью потоковых диаграмм в веб-аналитике // Научно-технические ведомости Санкт-Петербургского государственного политехнического университета. Информатика. Телекоммуникации. Управление. 2017. Т. 10. № 4. С. 7–16. DOI: 10.18721/JCSTCS.10401
  19. Ермолова Т. К., Иващенко П. В., Лаптев В. В. Изучение эффективности визуализации статических структур данных с помощью брусковых и секторных диаграмм методом ай-трекинга // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление. 2019. Т. 12. № 2. С. 16–27. DOI: 10.18721/JCSTCS.12202
  20. Янсон Ю. Э. Теория статистики : лекции проф. Ю. Э. Янсона 1886/87. СПб. : тип. Шредера, 1891. 561 с.
  21. Haass M. J. et al. A new method for categorizing scanpaths from eye tracking data // Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications. 2016. С. 35–38.

22. Murray N. et al. An examination of the oculomotor behavior metrics within a suite of digitized eye tracking tests // IEEE J Transl Eng Health Med. 2019. Т. 5. №. 4. С. 15–.
23. Cai X., Efstatiou K., Xie. X., Wu Y. A Study of the Effect of Doughnut Chart Parameters on Proportion Estimation Accuracy // Computer Graphics Forum 2018. Vol. 37 (3). Р. 1–13. DOI: 10.1111/cgf.13325
24. Skau D., Harrison L., Kosara R. An Evaluation of the Impact of Visual Embellishments in Bar Charts // Eurographics Conference on Visualization (EuroVis). 2015. Vol. 34 (3). Р. 221–230. DOI: 10.1111/cgf.1263
25. Боревич Е. В. Ай-трекинговое исследование влияния композиции на восприятие кинокадра // Программные системы и вычислительные методы. 2023. № 1. С. 51–60. DOI: 10.7256/2454-0714.2023.1.39634 EDN: IWYBNX URL: [https://nbpublish.com/library\\_read\\_article.php?id=39634](https://nbpublish.com/library_read_article.php?id=39634)

## **Результаты процедуры рецензирования статьи**

*В связи с политикой двойного слепого рецензирования личность рецензента не раскрывается.*

*Со списком рецензентов издательства можно ознакомиться [здесь](#).*

Рецензируемая статья посвящена актуальной задаче в области компьютерного дизайна – фокусировке внимания пользователя на определенных элементах. Авторы используют метод ай-трекинга для анализа внимания пользователя к различным формам представления данных, решая задачу визуализации комплексно с учетом семантической значимости данных.

Авторы используют качественные и количественные методы анализа, что повышает достоверность полученных результатов. Положительной стороной работы является формулировка критериев эффективности, учет эргономики, акцент на композиционных характеристиках. Авторы подробно останавливаются на особенностях чтения пользователем различных типов визуальных представлений, учитывая все фазы глазодвигательной активности, приводят результаты обзора исследований в данном направлении. Сильной стороной исследования является использование инструментов аналитики, позволивших выявить корреляцию между оцениваемыми параметрами.

Структура статьи отвечает требованиям к публикации, результаты выполненного авторами исследования количественных оценок не содержат.

Стиль изложения соответствует требованиям. Имеются иллюстрации.

Приводятся возможности использования результатов исследования в различных областях, авторы отмечают основные направления использования рассмотренных методов.

Библиография содержит 25 источников, преимущественно в зарубежных индексируемых источниках, а также отечественных рецензируемых журналах. Указаны doi. Ссылки по тексту имеются.

**Замечания.**

Авторам предлагается возможность привести представленные иллюстрации в цветном виде, что улучшит восприятие её содержания читателем.

Следует отметить, что некоторые формы визуального представления (графики, брусковые и секторные диаграммы) более привычны и, следовательно, понятны пользователю в случае небольшого числа переменных. Древовидные карты и потоковые диаграммы характерны для более узких областей; понимание их потребует большего времени от пользователя, что будет зафиксировано ай-трекером.

В экспериментальной части необходимо добавить описание эксперимента

(анализируемых данных, возможно привести скрин одного из заданий). Что собой представляют описываемые точки интереса, вопросы пользователю, предлагаемые варианты ответа (необходимо конкретизировать). Желательно добавить обоснование, почему разбиение проводится именно на 4 кластера. Подрисуточные надписи на 2 языках? Целесообразно привести схему разметки анализируемой страницы или графической информации задания, что позволит конкретизировать его.

Не хватает конкретики в описании эксперимента.

Потоковая диаграмма рис. 2 трудна для восприятия из-за низкого качества. Необходимо увеличить шрифт и/или сделать более контрастным текст относительно фона. Переменные X не пояснены.

Сравнения восприятия диаграмм приводятся по предыдущим исследованиям, не являющимися предметом исследований в данной статье.

Желательно отметить что являлось опорной точкой в приведенных исследованиях и упомянуть описания стимулов. Результаты анализа брусковых диаграмм (с накоплением?) сжаты.

Статья будет интересна широкому кругу специалистов, имеет практическую ценность для всех областей, где результат исследования представляется в графическом виде.

Статья соответствует тематике Журнала и может быть опубликована без повторного рецензирования

## Англоязычные метаданные

# Ambiguous Results when Using Parallel Class Methods within the .NET Framework

Gibadullin Ruslan Farshatovich

PhD in Technical Science

Associate Professor of the Computer Systems Department of Kazan National Research Technical University named after A.N. Tupolev-KAI (KNRTU-KAI)

420015, Russia, Republic of Tatarstan, Kazan, Bolshaya Krasnaya str., 55, office 432

 rfgibadullin@kai.ru



Viktorov Ivan Vladimirovich

Postgraduate Student of the Computer Systems Department of Kazan National Research Technical University named after A.N. Tupolev-KAI (KNITU-KAI)

420015, Russia, Republic of Tatarstan, Kazan, Bolshaya Krasnaya str., 55, office 432

 victorov.i.vl@yandex.ru



**Abstract.** Parallel programming is a way of writing programs that can run in parallel on multiple processors or cores. This allows programs to process large amounts of data or perform more complex calculations in a reasonable amount of time than would be possible on a single processor. The advantages of parallel programming: increased performance, load sharing, processing large amounts of data, improved responsiveness, increased reliability. In general, parallel programming has many advantages that can help improve the performance and reliability of software systems, especially with the increasing complexity of computational tasks and data volumes. However, parallel programming can also have its own complexities related to synchronization management, data races, and other aspects that require additional attention and experience on the part of the programmer. When testing parallel programs, it is possible to get ambiguous results. For example, this can happen when we optimize concatenation of float- or double-type data by means of For or ForEach methods of the Parallel class. Such behavior of a program makes you doubt about the thread safety of the written code. Such a conclusion can be incorrect and premature. The article reveals a possible reason for ambiguity of the results received by a parallel program and offers a concise solution of the question.

**Keywords:** Type decimal, Real numbers, Thread safety, Variability of results, Rounding errors, Multithreading, Programming language CSharp, Parallel programming, NET platform, Class Parallel

## References (transliterated)

1. X. Fan, R. -a. Wu, P. Chen, Z. Ning and J. Li, "Parallel Computing of Large Eigenvalue Problems for Engineering Structures," 2011 International Conference on Future Computer Sciences and Application, Hong Kong, China, 2011, pp. 43-46, doi: 10.1109/ICFCSA.2011.16.
2. Xuehui Chen, Liang Wei, Jizhe Sui, Xiaoliang Zhang and Liancun Zheng, "Solving fractional partial differential equations in fluid mechanics by generalized differential transform method," 2011 International Conference on Multimedia Technology,

- Hangzhou, 2011, pp. 2573-2576, doi: 10.1109/ICMT.2011.6002361.
3. R. Landau, "Computational Physics: A Better Model for Physics Education?," in Computing in Science & Engineering, vol. 8, no. 5, pp. 22-30, Sept.-Oct. 2006, doi: 10.1109/MCSE.2006.85.
  4. H. Ali, A. Doucet and D. I. Amshah, "GSR: A New Genetic Algorithm for Improving Source and Channel Estimates," in IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 54, no. 5, pp. 1088-1098, May 2007, doi: 10.1109/TCSI.2007.893507.
  5. D. -M. Zhu, J. -w. Gu, F. -H. Yu, W. -K. Ching and T. -K. Siu, "How correlation risk in basket credit derivatives might be priced and managed?," in IMA Journal of Management Mathematics, vol. 32, no. 2, pp. 195-219, April 2020, doi: 10.1093/imaman/dpaa013.
  6. J. Xu and Y. Shi, "Financial Leasing, Optimal Financial Structure and Economic Growth: An Analysis Based on Financial Inclusion Perspective," 2019 International Conference on Economic Management and Model Engineering (ICEMME), Malacca, Malaysia, 2019, pp. 147-150, doi: 10.1109/ICEMME49371.2019.00038.
  7. S. N. Cherny and R. F. Gibadullin, "The Recognition of Handwritten Digits Using Neural Network Technology," 2022 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), Sochi, Russian Federation, 2022, pp. 965-970, doi: 10.1109/ICIEAM54945.2022.9787104.
  8. Albahari, Joseph. C# 10 in a Nutshell. " O'Reilly Media, Inc.", 2022.
  9. Gibadullin R.F. Potokobezopasnye vyzovy elementov upravleniya v obogashchennykh klientskikh prilozheniyakh // Programmnye sistemy i vychislitel'nye metody. – 2022. – № 4. – S. 1-19.
  10. Rump, S.M. Fast and Parallel Interval Arithmetic. BIT Numerical Mathematics 39, 534–554 (1999). <https://doi.org/10.1023/A:1022374804152>.

## **Principles of developing a limited-queue mass service system on the .NET platform**

Antonova Polina Valerevna

Senior Lecturer of the Department of Intelligent Systems and Information Resource Management of Kazan National Research Technological University

420015, Russia, Republic of Tatarstan, Kazan, Karl Marx str., 68

✉ valerevna.p@inbox.ru



**Abstract.** The paper is the result of the thorough research aimed at studying the principles of creation, modeling and practical use of queue-limited mass service systems (MSS) based on the .NET technology platform and implemented using the C# programming language. In the course of the paper, special attention is paid to both single-channel and multi-channel systems. In the introductory part of the article the fundamental concepts of the theory of mass service are considered. The basic characteristics of systems, such as fixed queue length systems, single-channel and multi-channel systems with probability of service failure, systems with unlimited and limited wait times, closed systems, and multi-channel systems with channel-to-channel interaction, are discussed. Detailed examples of C# code are presented to illustrate the class structures used to model both single-channel and multi-channel MSS. Specific scenarios for the use of the presented structures within the framework of MSS modeling are considered. The paper highlights the practical application of mass service

systems in solving real-world problems. Using examples from banking and traffic management in telecommunications, it is demonstrated how MSS can contribute to optimizing waiting times and efficient resource management. Prospects for further research in the field of mass service systems are suggested. Given the importance of MSS in various industries such as banking, telecommunications, logistics and many others, the high relevance of the topic is due to the need to find new approaches and methods to improve the efficiency and optimization of these systems. In general, the article is a valuable research material for specialists involved in modeling and practical application of mass service systems.

**Keywords:** computational techniques, software systems, optimization, NET platform, C Sharp, multichannel systems, single-channel systems, limited queue, modeling, mass service systems

## References (transliterated)

1. Harchol-Balter M. Performance modeling and design of computer systems: queueing theory in action / Cambridge University Press, 2013.
2. Shevtsov A. N., Shchitov A. N., Konoreva N. A. Modelirovanie telekommunikatsionnykh sistem s pomoshch'yu SMO // Matematika i ee prilozheniya v sovremennoi nauke i praktike, 2015. S. 128-132.
3. Vishnevskii V. M., Dudin A. N. Sistemy massovogo obsluzhivaniya s korrelirovannymi vkhodnymi potokami i ikh primenie dlya modelirovaniya telekommunikatsionnykh setei // Avtomatika i telemekhanika. 2017. № 8. S. 3-59.
4. Gross D., Harris C. Fundamentals of Queueing Theory / Wiley-Interscience, 1998.
5. Osipov L. A. Imitatsionnoe modelirovanie sistem massovogo obsluzhivaniya s ogranicennoi ochered'yu // Nauka i tekhnika transporta. 2010. № 4. S. 30-36.
6. Furina K. O., Osechkina T. A. Matematicheskaya model' odnokanal'noi sistemy massovogo obsluzhivaniya s ogranicennoi ochered'yu // Nauka i sovremennost'. 2014. № 2. S. 103-110.
7. Gibadullin R.F. Potokobezopasnye vyzovy elementov upravleniya v obogashchennykh klientskikh prilozheniyakh // Programmnye sistemy i vychislitel'nye metody. 2022. № 4. S. 1-19.
8. Gibadullin R.F., Viktorov I.V. Neodnoznachnost' rezul'tatov pri ispol'zovanii metodov klassa Parallel v ramkakh ispolnyayushchei sredy .NET Framework // Programmnye sistemy i vychislitel'nye metody. 2023. № 2. S. 1-14.
9. Albahari J. C# 10 in a Nutshell / O'Reilly Media, Inc., 2022.
10. Viktorov I.V., Gibadullin R.F. Razrabotka sintaksicheskogo dereva dlya avtomatizirovannogo translyatora posledovatel'nogo programmного koda v parallel'nyi kod dlya mnogoyadernykh protsessorov // Programmnye sistemy i vychislitel'nye metody. 2023. № 1. S. 13-25.
11. Osipov G. S. Issledovanie sistem massovogo obsluzhivaniya s ozhidaniem v AnyLogic // Byulleten' nauki i praktiki. 2016. № 10 (11). S. 139-151.
12. Osipov G. S. Sistemy massovogo obsluzhivaniya s ogranicennoi dlitel'nost'yu ozhidaniya // Byulleten' nauki i praktiki. 2016. № 12 (13). S. 28-36.
13. Gibadullin R.F., Gashigullin D.A., Vershinin I.S. Razrabotka dekoratora StegoStream dlya assotsiativnoi zashchity baitovogo potoka. Modelirovanie, optimizatsiya i informatsionnye tekhnologii. 2023. 11(2). URL: [moitvvt.ru/rus/journal/pdf?id=1359](http://moitvvt.ru/rus/journal/pdf?id=1359).
14. Gibadullin R.F., Vershinin I.S., Glebov E.E. Razrabotka prilozheniya dlya assotsiativnoi

zashchity failov // Inzhenernyi vestnik Dona. 2023. № 6. URL:  
[ivdon.ru/ru/magazine/archive/n6y2023/8462/](http://ivdon.ru/ru/magazine/archive/n6y2023/8462/).

## **Development of an automated system for testing a cloud service for deploying virtual machines using modern monitoring tools**

**Marchenko Andrei** 

Cloud Software Architect, Intel Corporation

193318, Russia, Saint Petersburg region, Saint Petersburg, Badaeva str., 6k1

 mar4enko.ag@gmail.com

**Shchemelinin Dmitry** 

Doctor of Technical Science

Vice President, Intel Corporation

195251, Russia, Saint Petersburg, Politehnicheskaya str., 29

 dshchmel@gmail.com

---

**Abstract.** The object of this study is a service for managing virtual machines in a cloud environment. When developing and operating such a service, it becomes necessary to assess its availability and reliability for compliance with the selected quality level that the client can count on. This paper presents a developed system that allows testing the availability of a cloud service for managing virtual machines. The method of integration with the existing monitoring system at the enterprise using open source software in order to reduce the cost of development and operation is considered. A test case for deploying and removing a virtual machine using a graphical user interface has been developed and implemented, and triggering criteria have been defined. The requirements for the architecture and implementation of the system based on the production statistics of the virtual machine creation service using the Prometheus monitoring system are collected and analyzed. The novelty of the research lies in the development of a new method of testing a cloud service for managing virtual machines in order to increase its reliability and availability. Based on this method, a system for testing virtual machines is described and implemented, as well as a method for integration into the monitoring system of the Intel cloud service. During the operation of cloud environments with the help of this system, problem areas were identified in the architecture of the virtual machine creation service, which made it possible to optimize the system operation in a timely manner. The described method is an effective way to test cloud services, and can also be used to analyze and improve reliability and availability.

**Keywords:** application testing, python, kubernetes, selenium, prometheus, monitoring, information technology, metrics, data processing, threshold

### **References (transliterated)**

1. Prometheus – Monitoring system & time series database [Elektronnyi resurs]. URL: <https://prometheus.io/docs/introduction/overview/> (data obrashcheniya 04.04.2023).
2. Ofitsial'nyi Internet-sait Intel [Elektronnyi resurs]. URL: <https://www.intel.com/> (data obrashcheniya 04.04.2023).
3. The selenium browser automation project [Elektronnyi resurs]. URL:

- <https://www.selenium.dev/documentation/> (data obrashcheniya 04.04.2023).
4. Shchemelinin D.A. Matematicheskie modeli i metody monitoringa i prognozirovaniya sostoyaniya global'no raspredelennykh vychislitel'nykh kompleksov. Trudy uchebnykh zavedenii svyazi. 2021. T. 7. № 3. S. 73–78.
  5. Shchemelinin D.A. Metod prognozirovaniya sobytii v global'no raspredelennykh vychislitel'nykh kompleksakh. Sovremennaya nauka: aktual'nye problemy teorii i praktiki. Seriya: Estestvennye tekhnicheskie nauki. 2021. № 12–2. S. 47–54.
  6. Shchemelinin D.A. Metod i algoritm avtomaticheskogo vosstanovleniya informatsionnykh servisov na osnove ob"ektivnykh prognosticheskikh dannykh monitoringa. Sovremennaya nauka: aktual'nye problemy teorii i praktiki. Seriya: Estestvennye tekhnicheskie nauki. 2021. № 8. S. 140–144.
  7. Selenium with python [Elektronnyi resurs] URL: <https://selenium-python.readthedocs.io/> (data obrashcheniya 04.04.2023).
  8. The TIOBE Programming Community index an indicator of the popularity of programming languages [Elektronnyi resurs] URL: <https://www.tiobe.com/tiobe-index/> (data obrashcheniya 04.04.2023).
  9. Sujay Raghavendra. – Python testing with selenium – Apress Berkeley CA, 2020 – 4 c. ISBN 978-1-4842-6249-8
  10. When to use selenium grid [Elektronnyi resurs] URL: <https://www.selenium.dev/documentation/grid/applicability/> (data obrashcheniya 04.04.2023)
  11. Grafana documentation [Elektronnyi resurs]. URL: <https://grafana.com/docs/grafana/latest/> (data obrashcheniya 04.04.2023)
  12. Power comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling tests. Journal of Statistical Modeling and Analytics, Vol.2, No. I, 21-33, 2011. URL: <https://www.nrc.gov/docs/ML1714/ML17143A100.pdf> (data obrashcheniya 04.04.2023)
  13. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. Journal of Experimental Social Psychology, 49(4), 764–766. URL: <https://www.sciencedirect.com/science/article/pii/S0022103113000668> (data obrashcheniya 04.04.2023)
  14. Shchemelinin D.A. Sistema kriteriev i algoritm obrabotki informatsii i prinyatiya reshenii dlya programmnogo modulya otobrazheniya naibolee znachimykh sobytii monitoringa v informatsionnoi sisteme. XXI vek: itogi proshlogo i problemy nastoyashchego plus 2021. T. 10. № 3 (55). S. 67–71.

## Development of the security and fire system training stand for student training

**Yusupov Bulat Zufarovich**

Student of the Information Security Systems Department of Kazan National Research Technical University named after AN. Tupolev-KAI

420015, Russia, Republic of Tatarstan, Kazan, Bolshaya Krasnaya str., 55

✉ Bulatusupov9@gmail.com



**Abstract.** The article is devoted to the organization of the educational process using a specialized laboratory stand of security and fire systems. The main purpose of the paper is to

form professional competencies of students in the field of security and safety systems. The paper describes in detail the structure and components of the laboratory stand, emphasizing the importance of its key elements - the control panel "Astra-812 Pro" and a variety of detectors. This allows students to practically apply theoretical knowledge, studying the functioning and interaction of various components of the security and fire system. The article also presents schemes of connection of detectors to the expander, allowing students to more deeply understand the technical side of the process. The steps of the bench training are disclosed to support step-by-step, sequential learning. The study emphasizes the effectiveness of the proposed learning system, which involves direct student interaction with hands-on equipment. The findings confirm the significance of using such training stands in modern higher professional education, emphasizing their contribution to the formation of the necessary professional competencies in future specialists in the field of safety and security. Scientific novelty of the work consists in the development and application of a specialized training stand for teaching students to work with security and fire systems. This allows students to get direct experience of interaction with the equipment, deepen their understanding of the technical aspects of the systems operation and improve their practical skills. The work also contributes to research on educational approaches in the field of safety and security by proposing an effective teaching methodology through the use of a laboratory bench. This provides a platform for further research and development in this area that can broaden and deepen the educational opportunities in the field.

**Keywords:** student training, educational approaches, wiring diagrams, professional competencies, security system, detectors, monitoring device, fire protection system, training stand, practical skills

## References (transliterated)

1. Alekseenko O. I., Danilenko T. V., Kirii E. V. Problemy obucheniya i vospitaniya studentov v sovremenном vuze // Sovremennoe pedagogicheskoe obrazovanie. 2018. № 3. S. 3-5.
2. Fortova L. K., Yudina A. M. Problemy i perspektivy razvitiya tsifrovizatsii vysshego professional'nogo obrazovaniya // Vestnik Vladimirskogo gosudarstvennogo universiteta im. Aleksandra Grigor'evicha i Nikolaya Grigor'evicha Stoletovykh. Seriya: Pedagogicheskie i psichologicheskie nauki. 2022. № 51(70). S. 85-89.
3. Ivanova O. V., Moroz I. N. Vozmozhnosti i problemy tsifrovizatsii vysshego obrazovaniya // Vysshee obrazovanie segodnya. 2022. № 5. S. 30-35.
4. Yusupov B. Z., Martynov A. M. Razrabotka laboratornogo stenda okhranno-pozharnoi signalizatsii po distsipline tekhnicheskie sredstva okhrany // Aktual'nye problemy nauki i obrazovaniya v usloviyakh sovremennoy vyzovov : Sbornik materialov XIX Mezhdunarodnoi nauchno-prakticheskoi konferentsii, Moskva, 21 marta 2023 goda. S. 80-91.
5. Yusupov B. Z. Razrabotka laboratornogo stenda okhranno-pozharnoi signalizatsii po distsipline tekhnicheskie sredstva okhrany // XXV Tupolevskie chteniya (shkola molodykh uchenykh) : Mezhdunarodnaya molodezhnaya nauchnaya konferentsiya, posvyashchennaya 60-letiyu so dnya osushchestvleniya Pervogo poleta cheloveka v kosmicheskoe prostranstvo i 90-letiyu Kazanskogo natsional'nogo issledovatel'skogo tekhnicheskogo universiteta im. A.N. Tupoleva-KAI. Materialy konferentsii. Sbornik dokladov. V 6-ti tomakh, Kazan', 10-11 noyabrya 2021 goda. Tom V. Kazan': Individual'nyi preprintimatel' Sagieva A.R., 2021. S. 758-763.
6. Yusupov B. Z. Razrabotka metodiki provedeniya laboratornykh rabot na stende «OPS

- Astra-713» po distsipline tekhnicheskie sredstva okhrany // XXV Tupolevskie chteniya (shkola molodykh uchenykh) : Mezhdunarodnaya molodezhnaya nauchnaya konferentsiya, posvyashchennaya 60-letiyu so dnya osushchestvleniya Pervogo poleta cheloveka v kosmicheskoe prostranstvo i 90-letiyu Kazanskogo natsional'nogo issledovatel'skogo tekhnicheskogo universiteta im. A.N. Tupoleva-KAI. Materialy konferentsii. Sbornik dokladov. V 6-ti tomakh, Kazan', 10-11 noyabrya 2021 goda. Tom V. Kazan': Individual'nyi predprinimatel' Sagieva A.R., 2021. S. 764-767.
7. Petik N. S. Proektirovanie sistemy okhranno-pozharnoi signalizatsii // Molodost'. Intellect. Initsiativa : Materialy VI Mezhdunarodnoi nauchno-prakticheskoi konferentsii studentov i magistrantov, Vitebsk, 19 aprelya 2018 goda. Vitebsk: Vitebskii gosudarstvennyi universitet im. P.M. Masherova, 2018. S. 33-34.
  8. Bonch-Bruevich A. M., Kashpur E. I. Issledovanie perspektivnykh tekhnologii tsifrovoi modulyatsii v sistemakh okhranno-pozharnoi signalizatsii // SpetsTekhnika i svyaz'. 2015. № 3. S. 24-28.
  9. Sheikh S. M., Neiso M. K., Ellouze F. Design and implementation of a raspberry pi based home security and fire safety system // Computer Science & Information Technology (CS & IT). 2019. No. 3(3), P. 13.
  10. Jakubowski K. Operational Analysis of Fire Alarm Systems with a Focused, Dispersed and Mixed Structure in Critical Infrastructure Buildings // Energies. 2021. No. 14(23), P. 7893.
  11. Gibadullin R.F. Potokobezopasnye vyzovy elementov upravleniya v obogashchennykh klientskikh prilozheniyakh // Programmnye sistemy i vychislitel'nye metody. 2022. № 4. S. 1-19.
  12. Gibadullin R.F., Viktorov I.V. Neodnoznachnost' rezul'tatov pri ispol'zovanii metodov klassa Parallel v ramkakh ispolnyayushchei .NET Framework // Programmnye sistemy i vychislitel'nye metody. 2023. № 2. S. 1-14.
  13. Viktorov I.V., Gibadullin R.F. Razrabotka sintaksicheskogo dereva dlya avtomatizirovannogo translyatora posledovatel'nogo programmnogo koda v parallel'nyi kod dlya mnogoyadernykh protsessorov // Programmnye sistemy i vychislitel'nye metody. 2023. № 1. S. 13-25.

## **Eye-tracking detection of the area of interest in data visualization**

Laptev Maksim Vladimirovich

Design Engineer, PSK FRAM

196128, Russia, Saint Petersburg region, Saint Petersburg, Varshavskaya str., 5a, office 4-N/38

✉ lapt.maxim@gmail.com

Yanchus Victor Edmundasovich

PhD in Technical Science

Associate Professor, Higher School of Design and Architecture, Peter the Great St. Petersburg Polytechnic University

195251, Russia, Saint Petersburg, Politehnicheskaya str., 29

✉ victorimop@mail.ru

Laptev Vladimir Vladimirovich

Doctor of Art History

Professor of the Higher School of Design and Architecture of Peter the Great St. Petersburg Polytechnic University

195251, Russia, Saint Petersburg, Saint Petersburg, Politehnicheskaya str., 29

 laptevsee@gmail.com

**Abstract.** This study examines the features of forming in data visualization. To do this, the authors hypothesize that there are special areas of interest on the charts. The user pays attention to them in order to decode data encrypted with graphics. The presence of such areas, and in some cases, points, are intuitively determined during the formation of the design rules of information graphics. To verify them, the study used the eye-tracking method and the cluster analysis method. The application of interdisciplinary principles and rules of infographics design has been studied on the example of horizontal and vertical bar charts, pie, pictorial and flow charts.

The result of experiments with various types of charts showed the presence of additional areas of interest not previously indicated by data visualization specialists. This makes it possible to clarify the features of graphic forms and the formation of diagrams, allows you to verify the use of design rules formulated by the efforts of domestic and foreign specialists in the late XIX — early XX centuries. Such verification can be done using a methodology that includes statistical methods and the tracking method, which allows us to take into account the valuable experience of the past in modern information design. The presented procedure can be extended to other types of charts, diagrams and thematic maps, and have practical application in the analysis of big data visualization.

**Keywords:** pie chart, bar chart, pictorial chart, flow chart, area of interest, eye-tracking method, information graphics, data visualization, doughnut chart, formation

## References (transliterated)

1. Bertin J. Semiology of Graphics. Diagrams. Networks. Maps. Redlans: Esri Press, 2011. 438 p.
2. Laptev V. V., Orlov P. A. The Eye-Tracking Study of Effects of the Stylisation Level in Pictorial Charts // Humanities and Science University Journal. 2016. № 19. P. 44–56.
3. Ziemkiewicz C., Kosara R. Implied Dynamics in Information Visualization // Proceedings Advanced Visual Interfaces (AVI). 2010. P. 215–222.
4. Huang W. Using eye tracking to investigate graph layout effects // Proceedings of the 6th Asia-Pacific Symposium on Visualisation. 2007. P. 97–100.
5. Goldberg J. H., Helfman J. I. Comparing information graphics: a critical look at eye tracking // Proceedings of the 3rd BELIV'10 Workshop: Beyond time and errors: Novel evaluation methods for information visualization. 2010. P. 71–78.
6. Huestegge L, Pötzsch T. H. Integration processes during frequency graph comprehension: Performance and eye movements while processing tree maps versus pie charts // Applied Cognitive Psychology. 2018. 32 (2). P. 1–17. DOI: 10.1002/acp.3396
7. Sharif B., Maletic J. I. An empirical study on the comprehension of stereotyped UML class diagram layouts // Proceedings of the IEEE 17th International Conference on Program Comprehension. 2009. P. 268–272.
8. Orlov P., Ermolova T., Laptev V., Mitrofanov A., Ivanov V. The Eye-tracking Study of the Line Charts in Dashboards Design // VISIGRAPP 2016. Proceedings of the 11th Joint

- Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications. Volume 2: IVAPP. Rome – Italy, February 27–29, 2016. P. 205–213.
9. Burch M., Wallner G., Broeks N., Piree L., Boonstra N. The Power of Linked Eye Movement Data Visualizations // ETRA '21. ACM Symposium on Eye Tracking Research and Applications. 2021. P. 1–11. DOI: 10.1145/3448017.3457377
  10. Velichkovskii B. M. Kognitivnaya nauka: osnovy psikhologii poznaniya. M.: Smysl : Akademiya, 2006. T. 2. 432 s.
  11. Yarbus A. L. Eye Movements and Vision. Plenum Press, 1967. DOI: 10.1007/978-1-4899-5379-7
  12. Prats M., Garner S., Jowers I., McKay A., Pedreira N. Interpretation of geometric shapes – an eye movement study // Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications. 2010. P. 243–250.
  13. Biederman I. Recognition by components: A theory of human image understanding // Psychological Review. 1987. Vol. 94 (2). P. 115–147.
  14. Hoffman D. D., Richards W. A. Parts of recognition // Cognition. 1984. Vol. 18 (1–3). P. 65–96.
  15. Rim N. W., Choe K. W., Scrivner C., Berman M. G. Introducing Point-of-Interest as an alternative to Area-of-Interest for fixation duration analysis // PLoS One. 2021. No. 16 (5). URL: <https://pubmed.ncbi.nlm.nih.gov/33970920/> (data access: 24.02.2022). DOI: 10.1371/journal.pone.0250170
  16. Blascheck T. et al. AOI hierarchies for visual exploration of fixation sequences // Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications. 2016. P. 111–118.
  17. Laptev V. V., Orlov P. A. Klasternyi analiz vizual'nogo vospriyatiya struktury dannykh // Biznes-informatika. 2015. № 3 (33). S. 34–43.
  18. Laptev V. V., Orlov P. A., Dragunova O. V. Vizualizatsiya dinamicheskikh struktur dannykh c pomoshch'yu potokovykh diagramm v veb-analitike // Nauchno-tehnicheskie vedomosti Sankt-Peterburgskogo gosudarstvennogo politekhnicheskogo universiteta. Informatika. Telekommunikatsii. Upravlenie. 2017. T. 10. № 4. S. 7–16. DOI: 10.18721/JCSTCS.10401
  19. Ermolova T. K., Ivashchenko P. V., Laptev V. V. Izuchenie effektivnosti vizualizatsii staticheskikh struktur dannykh s pomoshch'yu bruskovykh i sektornykh diagramm metodom ai-trekinga // Nauchno-tehnicheskie vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie. 2019. T. 12. № 2. S. 16–27. DOI: 10.18721/JCSTCS.12202
  20. Yanson Yu. E. Teoriya statistiki : lektsii prof. Yu. E. Yansona 1886/87. SPb. : tip. Shredera, 1891. 561 s.
  21. Haass M. J. et al. A new method for categorizing scanpaths from eye tracking data // Proceedings of the Ninth Biennial ACM Symposium on Eye Tracking Research & Applications. 2016. S. 35–38.
  22. Murray N. et al. An examination of the oculomotor behavior metrics within a suite of digitized eye tracking tests // IEEE J Transl Eng Health Med. 2019. T. 5. №. 4. S. 15–.
  23. Cai X., Efsthathiou K., Xie. X., Wu Y. A Study of the Effect of Doughnut Chart Parameters on Proportion Estimation Accuracy // Computer Graphics Forum 2018. Vol. 37 (3). P. 1–13. DOI: 10.1111/cgf.13325
  24. Skau D., Harrison L., Kosara R. An Evaluation of the Impact of Visual Embellishments in Bar Charts // Eurographics Conference on Visualization (EuroVis). 2015. Vol. 34 (3). P. 221–230. DOI: 10.1111/cgf.1263

25. Borevich E. V. Ai-trekingovoe issledovanie vliyaniya kompozitsii na vospriyatie kinokadra // Programmnye sistemy i vychislitel'nye metody. 2023. № 1. S. 51–60. DOI: 10.7256/2454-0714.2023.1.39634 EDN: IWYBNX URL: [https://nbpublish.com/library\\_read\\_article.php?id=39634](https://nbpublish.com/library_read_article.php?id=39634)