ГУАП

Государственный университет
аэрокосмического приборостроения

ОБРАБОТКА, ПЕРЕДАЧА И ЗАЩИТА ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СИСТЕМАХ’22

Международная научная конференция 11–15 апреля 2022 г.

Санкт-Петербург

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное
учреждение высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ

Посвящается
Всемирному дню космонавтики и авиации

ОБРАБОТКА, ПЕРЕДАЧА И ЗАЩИТА ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СИСТЕМАХ '22

Вторая Международная научная конференция
11–15 апреля 2022 г.

Сборник докладов

Санкт-Петербург
2022

УДК 001
ББК 72
О-23

О-23 Обработка, передача и защита информации в компьютерных системах '22: Вторая Междунар. науч. конф. (СПб., 11–15 апреля 2022 г.): сб. докл. – СПб.: ГУАП, 2022. – 341 с.
ISBN 978-5-8088-1701-2
DOI: 10.31799/978-5-8088-1701-2-2022-2

В апреле 2022 г. в Санкт-Петербургском государственном университете аэрокосмического приборостроения состоится Международная научная конференция «Обработка, передача и защита информации в компьютерных системах». В работе конференции примут участие ведущие ученые и специалисты предприятий, преподаватели, научные сотрудники и аспиранты вузов со всего мира.

Статьи сборника отражают основные направления научных исследований, обсуждаемые на конференции. Представленные работы посвящены актуальным проблемам обработки, передачи, защиты информации; проблемам построения современных компьютерных систем и вопросам автоматического управления; разработке перспективных вычислительных сетей, их математическому и программному обеспечению.

Сборник предназначен для научных работников, аспирантов, докторантов и студентов старших курсов технических вузов.

УДК 001
ББК 72

ISBN 978-5-8088-1701-2
DOI: 10.31799/978-5-8088-1701-2-2022-2

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ

ПРОБЛЕМНО-ОРИЕНТИРОВАННЫЕ ВЫЧИСЛИТЕЛЬНЫЕ КОМПЛЕКСЫ

УДК 004.4 **DOI: 10.31799/978-5-8088-1701-2-2022-2-3-9**

А. В. Аграновский*
кандидат технических наук, доцент
Е. Л. Турнецкая*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ПЕРСПЕКТИВНЫЕ НАПРАВЛЕНИЯ ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ МНОГОФУНКЦИОНАЛЬНЫХ ВЕБ-СИСТЕМ

Программная реализация многофункциональной веб-системы основана на комплексе цифровых решений, выбор которых требует понимания перспективных направлений веб-разработки. В проведенном исследовании рассмотрены тенденции выбора типа архитектурного решения с позиций подходов API-First development, Mobile First, Progressive Web Apps, Serverless-технологии; возможности применения сквозных цифровых технологий искусственного интеллекта, виртуальной и дополненной реальности; актуальные программные средства реализации веб-систем с применением принципов No-Code/Low Code, WebAssembly, Accelerated mobile pages и программных фреймворков. Особое внимание уделено принципам построения пользовательского интерфейса с использованием подходов Content First и User Centered Design. Приведены примеры практической реализации компонентов веб-систем, основанные на перечисленных принципах. Обширный библиографический список подтверждает состоятельность аналитической базы исследования.
Ключевые слова: *веб-разработка, API-First development, Mobile First, Progressive Web Apps, Serverless-технологии, No-Code/Low Code, WebAssembly, Accelerated mobile pages, фрейморки, Content First, User Centered Design.*

A. V. Agranovskii*
PhD, Tech., Associate Professor
E. L. Turnetskaya*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

FUTURE DIRECTIONS IN THE DESIGN AND DEVELOPMENT OF MULTIFUNCTIONAL WEB-BASED SYSTEMS

The software implementation of a multifunctional web system is based on a set of digital solutions, the choice of which requires an understanding of the promising areas of web-development. The study examines the trends in choosing the type of architectural solution from the standpoint of API-First development, Mobile First, Progressive Web Apps, Serverless-technologies; the possibility of using end-to-end digital technologies of artificial intelligence, virtual and augmented reality; current software tools for the implementation of web systems using the principles of No-Code / Low Code, WebAssembly, Accelerated mobile pages and software frameworks. Particular attention is paid to the principles of building a user interface using Content First and User Centered Design approaches. Examples of practical implementation of components of web systems based on the above principles are given. An extensive bibliographic list confirms the consistency of the analytical base of the study.
Keywords: *web-programming, web-based systems, API-First development, Mobile First, Progressive Web Apps, Serverless-технологии, No-Code/Low Code, WebAssembly, Accelerated mobile pages, фрейморки, Content First, User Centered Design.*

Данные аналитического ресурса Internet Live Stat свидетельствуют, что в интернет-сообществе зарегистрировано около 2 млн веб-систем [1]. Создание эффективно функционирующей многофункциональной веб-системы, позволяющей решать разноплановые задачи

без привязки к конфигурации вычислительной системы и программного обеспечения пользователя, требует специальных навыков в области веб-программирования, знания перспективных подходов к веб-разработке, выбора актуальных архитектурных решений и понимания моделей поведения посетителя, основанных на пользовательских предпочтениях и ожиданиях [2, 3]. Заказчик веб-системы заинтересован в разработке ресурса, который органично связан с бизнес-процессами предприятия, ориентирован на привлечение пользователей и реализован с учетом актуальных технологий разработки. Поэтому возникает необходимость в освещении перспективных подходов к реализации веб-систем, рассматриваемых со следующих сторон:

1) особенностей архитектурных решений;

2) применения перспективных информационных технологий;

3) актуальных программных средств реализации;

4) тенденций развития дизайна интерфейсов.

Особенности архитектурных решений

1. Принцип разработки API-First development.

Идея данной технологии, точнее сказать подхода к разработке, состоит в первоочередной реализации программных интерфейсов API между компонентами реализуемого программного продукта, например между клиентской и серверной частями. Такой подход позволяет вести параллельную разработку множества компонентов веб-системы: серверной части, базы данных, контроллеров, тестовых кейсов, дизайна интерфейса пользователя, создания документации и т. д. В частности, в компании «Мегагрупп.ру» [4] в ходе выполнения заказов на веб-разработку одновременно идет работа над сервисами:

– онлайн-консультаций Onicon.ru;

– почтовых рассылок Maliver.ru;

– учета клиентов и заявок MegaCRM;

– инструментов управления рекламными кампаниями Rekmala.

2. Принцип разработки Mobile First.

Согласно отчету Digital-2020, более 5,19 млрд чел. используют мобильные телефоны [5]. Прирост пользователей, выбирающих такие устройства, вырос на 124 млн (2,4%) за последний год. Поэтому веб-разработчики в первую очередь стремятся реализовать мобильную версию веб-системы, которая может значительно отличаться по своему дизайну и функционалу от версии, которую может просматривать пользователь на персональном компьютере или планшете [6]. Согласно подходу Mobile First, изначально разрабатывается UI/UX-интерфейс для расширений, которые устанавливают на мобильные устройства, затем он может быть дополнен и масштабирован под планшетные и стационарные устройства. Применение такого подхода повышает удобство использования приложений и увеличивает конверсию.

По данным исследования Innofact и Mastercard [7], в России 84% людей совершают покупки с помощью смартфона, при этом 36% делают это минимум раз в неделю. Сравнительное исследование компании Criteo [8] показало, что по сравнению с мобильным сайтом в мобильном приложении:

– количество просмотренных продуктов на 285% выше;

– конверсия в покупки на 120% выше;

– средний чек на 11% выше.

Можно ожидать, что идеология подхода Mobile First будет востребована в ближайшие 10–15 лет [9].

3. Технология Progressive Web Apps.

Progressive Web Apps (PWA) – это прогрессивные веб-приложения, в которых сочетают основные принципы реализации мобильных приложений и классических веб-систем. С помощью этой перспективной технологии расширяют возможности классического веб-приложения функциональностью мобильного приложения, повышают производительность за счет быстрой загрузки веб-страниц, размер которых не превышает 1 Мбайт, и улучшают конверсию.

Так как PWA состоят из веб-страниц, каждая из которых имеет свой URL, то при запросе пользователя они будут отображены в любом браузере без установки мобильного приложения. В случае неустойчивого сетевого соединения или его отсутствия пользователь может загрузить PWA на свое устройство и просматривать интересующий контент в мобильной компоненте, так как данные подгружаются из кэша. Дружелюбный интерфейс приложений основан на принципах User Interface (UI) и User Experience (UX) [9]. В настоящее время бренды Twitter, Book My Show, Uber, Telegram, Starbucks, Forbes, The Washington Post, Walmart используют приложения на базе PWA как основное или в дополнение к мобильному приложению [10].

Пользовательская лояльность к сайтам PWA достигается с помощью push-уведомлений, возможности установки ярлыка на рабочий стол, автономной работы приложения и высокой скорости загрузки страниц.

Progressive Web Apps состоят из двух основных частей:

– оболочки, которая отображает структуру страницы (сетку);

– контента, который может варьироваться между различными страницами приложения.

В зависимости от технологии, используемой для разработки страницы, содержимое может быть отправлено с сервера в формате HTML (HyperText Markup Language) или с помощью текстового формата обмена данными JSON (JavaScript Object Notation) [11]. Использование технологии PWA подтверждает тенденцию распространения подхода Mobile First.

4. Serverless-технология.

Бессерверная архитектура – это технологическое решение, основанное на событиях и запросах, которое позволяет разработчикам приложений создавать в облаке эффективные рабочие среды, обладающие всеми вычислительными ресурсами, необходимыми для организации бесперебойного процесса разработки [12].

При таком подходе реализуют две взаимодополняющие сервисные модели:

– бэкенд как услуга (Backend-as-a-Service, BaaS);

– функция как услуга (Function-as-a-Service, FaaS).

Для приложений, основанных на этой технологии, характерна событийная архитектура (event-driven architecture, EDA). Преимущество бессерверной архитектуры состоит в том, что она позволяет разработчикам сосредоточиться исключительно на функционале приложения, а обслуживание инфраструктуры и настройка ее параметров, таких как масштабируемость, высокая доступность, будут в ведении компаний, которые предоставляют Serverless-услуги. Наиболее известными фреймворками для Serverless.com являются APEX (https://www.delltechnologies.com/) и Amazon Web Services (https://aws.amazon.com/ru/), Cloud Function от Google (https://cloud.google.com/functions) и Azure Function от компании Microsoft (https://azure.microsoft.com/ru-ru/services/functions/), Yandex Cloud Functions (https://cloud.yandex.ru/services).

Применение сквозных цифровых технологий при разработке веб-систем

В федеральном проекте «Цифровые технологии» программы «Цифровая экономика» определены шесть сквозных цифровых технологий: нейротехнологии и искусственный интеллект; системы распределенного реестра; квантовые технологии; новые производственные технологии; технологии беспроводной связи; технологии виртуальной и дополненной реальности. При веб-разработке чаще всего применяют нейротехнологии, технологии искусственного интеллекта, виртуальной и дополненной реальности. Рассмотрим подробнее направления, основанные на применении искусственного интеллекта.

1. Голосовые помощники.

Для автоматизации рутинных процессов в работе службы поддержки используют голосовых помощников, базирующихся на технологии искусственного интеллекта по распознаванию и синтезу речи. Они могут отвечать на простые вопросы с помощью базы знаний компании и переводить разговор на оператора, если разговор идет не по заранее составленному алгоритму. К таким помощникам относят, например, боты для записи на прием к врачу по телефону 112 в Санкт-Петербурге.

2. SMART-системы.

Чат-боты, в отличие от голосовых помощников, отправляют автоматизированные ответы пользователю при общении в мессенджерах Telegram, WhatsApp, Viber и других чатах. Основные технологии, заложенные в основу их функционирования, – обработка естественного языка и машинное обучение, что позволяет боту понимать сообщение пользователя и отвечать в соответствии с алгоритмом. Обучение помощников проходит, как правило, на открытых данных для обучения и с привлечением суперкомпьютеров, в частности чат-бот Олег «Тинькоф банка» проходит обучение с помощью компьютера «Колмогоров» [13].

В настоящее время более 3 млрд людей регулярно обращаются к голосовым ассистентам, в частности Google Assistant пользуются 500 млн чел. [14]. Международный опыт в данной субтехнологии также представлен виртуальным помощником Cortana в операционной системе Windows, голосовым помощником Siri от Apple. Российскими продуктами являются разработки ABBYY, голосовой помощник «Алиса» от «Яндекса» и библиотека диалоговых систем DeepPavlov.

Предполагалось, что к началу 2022 г. данную технологию на своих интернет-платформах внедрят более 70% компаний. В частности, в Российской Федерации Сбербанк разработал дорожную карту по развитию направления «Нейротехнологии и искусственный интеллект» в рамках федерального проекта «Цифровые технологии» нацпроекта «Цифровая эконо-

мика» [15]. В этом документе прогнозируют активное внедрение мультизадачных разговорных ассистентов до 2024 г. В мобильном приложении «Сбербанк-онлайн» реализовано семейство виртуальных ассистентов «Салют» [16]. Трудности при их внедрении обусловлены сложностью обучения смарт-систем общению с клиентами на основе живых диалогов, навыкам распознавания эмоций, намерений собеседника, с одновременным решением бизнес-задачи клиента или компании.

С помощью перечисленных технологий решают наиболее востребованные функции маркетинга:

– сегментация клиентской базы и рассылка по сегментам;

– привлечение новых клиентов;

– удержание старых клиентов;

– возврат клиентов;

– коммуникация с клиентом в синхронном и асинхронном режимах;

– адаптированная программа лояльности;

– реализация автоматизированной воронки продаж;

– онлайн-продажи через чат-бот.

Перспективным направлением считают реализацию многозадачного запроса пользователя, выполнение которого будет основано на сочетании голосового и визуального поиска.

Актуальные программные средства реализации веб-систем

1. Подходы к веб-разработке No-Code/Low Code.

Использование этих подходов при веб-разработке позволяет в короткие сроки проверить гипотезы по продвижению товаров конкретной компании, например стартапа. При Low Code подходе используют визуальные конструкторы, а для решения типовых задач – готовые программные модули, компоненты или встроенные сервисы. Для доработки программного продукта под запросы конкретного заказчика привлекают опытного программиста. Подход No-Code подразумевает реализацию веб-системы программными инструментами специализированных платформ на основе визуального моделирования без дополнительной доработки. Примером одновременного использования подходов может служить программная связка Тильды и Flex-Table или проблемно-ориентированная платформа системы Directum [17].

2. Технология WebAssembly.

WebAssembly (Wasm) – безопасный и эффективный низкоуровневый формат данных в виде текстового представления бинарного байт-кода, созданного для виртуальной вычислительной машины. С помощью этой технологии программный код, написанный, например, на C++ или Rust, компилируется в машинный код по мере загрузки. Он оборачивает строки исходного кода на LLVM-совместимом языке в специализированную оболочку для преобразования, создает бинарный файл .ism и производит автоматическое подключение к исполняемому JavaScript-коду в формате .wasm. Необходимость взаимодействия с JavaScript вызвана тем, что на сегодняшний день файлы только такого формата могут запустить сценарий в браузере пользователя с WEB API. Если рассматривать технологию Wasm в широком смысле, то она позволяет получить доступ к разработанной инфраструктуре любых браузеров для программного кода, написанного на других языках программирования [18].

3. Технология Accelerated mobile pages.

По данным аналитической компании Statista.com, мировой мобильный трафик вырос в 4 раза по сравнению с 2017 г., а к 2022 г. его доля составит 77% от общего интернет-трафика [19]. Поэтому необходимо обеспечивать быстрое получение информации из Интернета пользователями, владеющими мобильными устройствами и планшетами. Для повышения скорости загрузки веб-страниц разработана технология Accelerated mobile pages (AMP) или ускоренные мобильные страницы [20]. Она обладает открытым исходным кодом и разработана компанией Google (https://www.ampproject.org/) для оперативной загрузки страниц даже при низкой скорости сети. Ускоренные страницы состоят из HTML-разметки, в которой часть тегов запрещена или заменена на специальные AMP-теги, AMP-библиотеки JavaScript и AMP Cache [21]. Быстрой загрузке также способствует кэширование страниц в поисковой системе Google за счет использования Content Delivery Network (CDN, сети доставки контента) или автоматическое перенаправление на ускоренные страницы.

В Российской Федерации компания «Яндекс» разрабатывает аналогичную технологию под названием «турбо-страницы» (https://tech.yandex.ru/turbo/). Примерами реализации веб-систем с применением мобильных страниц служат [21]:

– интернет-магазин ebay.com, каталог товаров [22];

– новостной портал lenta.ru, новостная статья [23];

– кулинарный сайт eda.ru, страница рецепта [24].

4. Программные фреймворки.

Согласно рейтингу от компании TIOBE, лидирующие позиции занимают языки программирования Python (11,27%), C (11,16), Java (10,46), C++ (7,50), C# (5,26), Visual Basic (5,24), JavaScript (2,19%) [25]. Для уменьшения нагрузки на телекоммуникационные сети, ускорения загрузки страницы и повышения интерактивности веб-приложений осуществляют перенос большинства действий на вычислительные мощности конечных пользователей – программные средства браузеров.

В результате проведенного на Stack Overflow в 2020 г. опроса 69,7 % из 47 184 опрошенных профессиональных веб-разработчиков отдают предпочтение JavaScript [26]. Аналитический отчет State of Frontend 2020 [27] показывает, что чаще всего при разработке клиентской части веб-систем применяют JavaScript-фреймворки: React, Vue.js, Svelte, Angular.

Тенденции развития дизайна интерфейсов веб-системы

Увеличение конверсии веб-приложений и посещаемости целевых веб-страниц возможно за счет повышения их визуальной привлекательности и дружественного интуитивно-понятного интерфейса [28]. Цифровой портрет посетителя веб-приложения постоянно меняется. Поэтому важно понимать, какие факторы и модели поведения влияют на его выбор и расстановку приоритетов при выборе контента [29]. Исследование поведения посетителей, проведенное маркетологом Нилом Пателем [30], показывает, что 90% читателей веб-страницы изучают ее информационный контент около 10 с, затем клиент принимает решение о продолжении изучения материалов, совершении целевого действия или уходе на другое веб-приложение. Одним из факторов фиксации внимания является дизайн сайта, который должен быть эстетически близок и понятен посетителю. Поэтому при разработке интерфейсов веб-систем следует выбирать актуальные направления дизайна, основанные на достижения UX/UI, такие как:

– подход Content First с ориентацией на контент и стиранием грани между пользователем и интерфейсом;

– подход User Centered Design, основанный на проектировании и разработке приложений с опорой на интересы и потребности пользователя.

Результат использования этих подходов можно увидеть в следующих перспективных направлениях дизайнов интерфейсов [31].

1. Элементы 3D и дополненной реальности (WebAR).

Объемные 3D-объекты позволяют сделать изображение более живым и сфокусировать внимание пользователя на важных информационных блоках. Средства дополненной реальности WebAR превносят на веб-страницах новые возможности по продаже и продвижению товаров, меняющие культуру потребления [31]. Эта технология «переносит интерактивные 3D-модели в реальный мир, к которым можно получить доступ через QR-код или ссылку» без установки специализированных приложений [32]. По открытым данным магазина приложений Apple технология WebAR поддерживается на 94% устройств (https://developer.apple.com/support/app-store/).

2. Использование минимализма в оформлении.

Дизайнеры веб-интерфейсов минимизируют количество дополнительных элементов в оформлении, акцентируя внимание пользователей на информационных блоках, которые несут функциональную и смысловую нагрузку. В оформлении блоков и объектов проявляются четкие линии, в цветовом решении присутствуют чистота цвета и монохромность.

3. Применение видеообзоров или слайдового оформления контента.

Посетителям веб-страниц иногда бывает удобно просматривать информационный контент веб-страниц, заменяя визуальным обзором чтение текстовых материалов. Для такой категории пользователей в интерфейсе предусматривают motion-дизайн и видеоплееры, с помощью которых появляется возможность наглядного изучения материалов и просмотра практических примеров. Программная реализация таких компонент – сложная технологическая задача, с которой может справиться веб-разработчик высокой квалификации. Таким образом, дизайн веб-приложения ориентирован на пользователя, выполняет задачи бизнеса и должен быть реализован актуальными программными средствами с применением перспективных подходов разработки.

Заключение

Обзор технологических векторов развития заказного рынка веб-разработки основан на анализе публикаций и на обобщении мнений экспертов в данной области, опубликованных в профессионально-ориентированных источниках информации.

Характерной чертой публичного информационного общества является наличие веб-

систем, разработанных с применением перспективных технологий. Пропорционально развитию рынка мобильного интернета растет рынок мобильных приложений, поэтому одним из важных для веб-разработки является принцип Mobile First и сопряженная с ним технология Progressive Web Apps. Пользователи часто интуитивно отдают предпочтения многофункциональным веб-системам, информационный контент которых доступен сразу после запроса посетителя, а результат выполнения отображен средствами дружелюбного интерфейса. Именно на основе клиентоориентируемости и персонализированного контента следует проводить современную веб-разработку, базируясь на подходах User Centered Design и Content First при разработке интерфейса. Удержание клиентов и помощь при выборе требуемого функционала веб-систем осуществляют средствами SMART-систем, основанных на технологиях искусственного интеллекта. Понимание основных направлений развития в проектировании и разработке веб-систем позволит разработчикам программного обеспечения выбрать адекватные цифровые решения при выполнении требований заказчиков [33].

Библиографический список

1. Internet Live Stat: аналит. ресурс. URL: https://www.internetlivestats.com/ (дата обращения: 30.10.2021).

2. *Гольчевский Ю. В.* Подходы к проектированию и разработке современного корпоративного web-ресурса // Экономика. Информатика. 2020. № 2. URL: https://cyberleninka.ru/article/n/podhody-k-proektirovaniyu-i-razrabotke-sovremennogo-korporativnogo-web-resursa (дата обращения: 30.10.2021).

3. *Аграновский А. В., Турнецкая Е. Л.* Обзор современных инструментальных средств разработки интернет-приложений // Обработка, передача и защита информации в компьютерных системах: первая Всерос. науч. конф. (СПб., 14–22 апр. 2020 г.): сб. докл. СПб.: ГУАП, 2020. С. 3–8.

4. Блог. Университет Интернет-маркетинга. URL: https://blog.megagroup.ru/ post/megagroup-ru-popala-v-top-5-veb-studij-po-rossii-v-2020-godu (дата обращения: 31.10.2021).

5. Отчет о состоянии Digital 2020 Global Overview о цифровых трендах. URL: https://datareportal.com/reports/digital-2020-global-digital-overview (дата обращения: 31.10.2021).

6. *Купина А.* Увеличиваем прибыль приложения-магазина в 2021 году: какие фичи привлекут больше клиентов и повысят конверсии в покупку. URL: https://vc.ru/design/197220-uvelichivaem-pribyl-prilozheniya-magazina-v-2021-godu-kakie-fichi-privlekut-bolshe-klientov-i-povysyat-konversii-v-pokupku (дата обращения: 31.10.2021).

7. *Пророкова Е.* Итоги исследования мобильной коммерции. URL: https://www.mastercard.com/news/europe/rus/press-fentr/press-relizy/ru-ru/2018/dekabr/issledovanie-mastercard-8-iz-10-evropeifev-sovershayut-pokupki-cherez-mobil-nye-ustroistva (дата обращения: 30.10.2021).

8. Аналитическое исследование компании мобильной коммерции // Criteo. URL: https://www.criteo.com/resources (дата обращения: 31.10.2021).

9. *Аграновский А. В., Турнецкая Е. Л.* Оптимизация интерфейсов веб-систем с использованием принципов юзабилити // Научная сессия ГУАП: сб. докл. науч. сессии, посвящ. Всемир. дню авиации и космонавтики: в 3 ч. Ч. II. Технические науки. СПб., 2019. С. 231–233.

10. *Данилюк Е.* PWA-приложения. Что это такое и для чего бизнесу создавать приложение из сайта. URL: https://vc.ru/marketing/141463-pwa-prilozheniya-chto-eto-takoe-i-dlya-chego-biznesu-sozdavat-prilozhenie-iz-sayta (дата обращения: 31.10.2021).

11. Блог хостинг-провайдера Timeweb. Что такое Progressive Web Apps и в чем их преимущества. URL: https://timeweb.com/ru/community/articles/chto-takoe-progressive-web-apps (дата обращения: 31.10.2021).

12. Бессерверная архитектура или микросервисы – как выглядит будущее вычислительных технологий для бизнеса? URL: https://habr.com/ru/post/558990/ (дата обращения: 31.10.2021).

13. Голосовой помощник Тинькофф банка в сфере финансов и лайфстайл-услуг. URL: https://www.tinkoff.ru/about/news/13062019-tinkoff-introduces-oleg-the-worlds-first-voice-assistant-in-lifestyle-and-finance/ (дата обращения: 31.10.2021).

14. *Калинин И.* Цифровое доверие. URL: https://trends.rbc.ru/trends/ industry/5f7b5aeb9a79470fc0caa01d (дата обращения: 31.10.2021).

15. Дорожная карта развития «сквозной» цифровой технологии «Нейротехнологии и искусственный интеллект». URL: https://digital.gov.ru/ru/documents/6658/ (дата обращения: 31.10.2021).

16. Виртуальные ассистенты Салют. URL: https://www.sberbank.ru/ru/ person/dist_services/salut_bank. (дата обращения: 31.10.2021).

17. Компания Directum. URL: https://www.directum.ru/company/news-analytics/low-code/ (дата обращения: 31.10.2021).

18. Введение в WebAssembly: как устроена технология и почему она важна. URL: https://tproger.ru/translations/introduction-to-webassembly (дата обращения: 31.10.2021).

19. Statista.Com: аналит. компания. URL: https://www.statista.com/ statistics/271405/global-mobile-data-traffic-forecast/ (дата обращения: 31.10.2021).

20. Что такое AMP: подробное руководство по ускоренным мобильным страницам. URL: https://texterra.ru/blog/kak-sozdat-amp-ili-podrobnoe-rukovodstvo-po-uskorennym-mobilnym-stranitsam.html/ (дата обращения: 31.10.2021).

21. Блог компании AMG. URL: https://amdg.ru/blog/nuzhny-li-amp-vashemu-saytu-podrobnoe-rukovodstvo-po-tekhnologii/ (дата обращения: 31.10.2021).

22. Каталог товаров // Ebay.com: интернет-магазин. URL: https://by.ebay.com/b/amp/Sony-PS4-Consoles/139971/bn_339810 (дата обращения: 31.10.2021).

23. Lenta.ru: новостной портал. URL: https://m.lenta.ru/news/ 2019/02/04/gdp/amp/ (дата обращения: 31.10.2021).

24. Eda.ru: кулинар. сайт. URL: https://amp.eda.ru/recepty/vypechka-deserty/tvorozhnij-desert-bez-vipechki-33617(дата обращения: 31.10.2021).

25. Рейтинг компании TIOBE. URL: https://www.tiobe.com/tiobe-index (дата обращения: 31.10.2021).

26. Рейтинг компании Stackoverflow. URL: https://insights.stackoverflow.com/ survey/2020#technology-programming-scripting-and-markup-languages-professional-developers (дата обращения: 31.10.2021).

27. Аналитический отчет State of Frontend 2020. URL: https://tsh.io/state-of-frontend/#frameworks (дата обращения: 31.10.2021).

28. *Stojmenovic M., Biddle R., Grundy J., Farrel V.* The influence of textual and verbal word-of-mouth on website usability and visual appeal // Journal of Supercomputing. 2019. № 75 (4). P. 1783–1830.

29. *Kim I., Pant G.* Predicting web site audience demographics using content and design cues // Information and Management. 2019. № 56 (5). P. 718–730.

30. *Блог Н.* Паттеля. URL: https://neilpatel.com/blog/ (дата обращения: 31.10.2021).

31. *Чигарев И.* 60 трендов веб-дизайна в 2021 году. URL: https://ichigarev.ru/web-design/trends-web-design-2.html (дата обращения: 31.10.2021).

32. WebAR – новая дополненная реальность, меняющая культуру потребления. Особенности и проблемы технологии. URL: https://vc.ru/future/186094-webar-novaya-dopolnennaya-realnost-menyayushchaya-kulturu-potrebleniya-osobennosti-i-problemy-tehnologii (дата обращения: 31.10.2021).

33. Исследование рынка заказной веб-разработки. URL: https://cmsmagazine.ru/journal/research-market-research-for-custom-web-development (дата обращения: 31.10.2021).

УДК 681.2.001.5 **DOI: 10.31799/978-5-8088-1701-2-2022-2-10-14**

Б. К. Акопян*
ассистент
Е. П. Виноградова*
старший преподаватель
М. В. Русанов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

МОДЕЛИРОВАНИЕ СОВМЕСТНЫХ ОЦЕНОК ПАРАМЕТРОВ ГАУССОВСКОГО СИГНАЛА

Рассматривается методика получения совместных оценок параметров гауссовского сигнала на основе комплексной модели. Обсуждаются возможности современных пакетов математического моделирования по получению оценок в формальном виде, а также методология получения алгоритмов для моделирования значений интересующих параметров.
Ключевые слова: *гауссовский сигнал, комплексная модель сигнала, оценка гауссовского сигнала, моделирование, Maple.*

B. K. Akopyan*
Assistant
E. P. Vinogradova*
Senior Lecturer
M. V. Rusanov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

SIMULATION OF JOINT ESTIMATES OF THE PARAMETERS OF THE GAUSSIAN SIGNAL

The paper discusses a technique for obtaining joint estimates of the parameters of a Gaussian signal based on a complex model. The possibilities of modern mathematical modeling packages for obtaining estimates in a formal form are discussed, as well as the methodology for obtaining algorithms for modeling the values of the parameters of interest.
Keywords: *Gaussian signal, complex signal model, Gaussian signal estimation, modeling, Maple.*

Известно [1], что плотность вероятности совместного распределения в совпадающие моменты времени компонент гауссовского аналитического сигнала и его первой производной имеет вид:

$$\theta_4(u,v,du,dv,w,ww,disp) \equiv \equiv \frac{e^{-\frac{\left[w^2\cdot(u^2+v^2)+du^2+dv^2-2\cdot ww\cdot(u\cdot dv-v\cdot du)\right]}{2\cdot disp\cdot(w^2-ww^2)}}}{4\cdot\pi^2\cdot disp^2(w^2-ww^2)},$$

где u –действительный нормальный случайный процесс, v – мнимый нормальный процесс, сопряженный с действительным по Гильберту, $du=\dot{u}$ и $dv=\dot{v}$ – их первые производные по времени, ww – средняя частота спектра, w – среднеквадратическая ширина спектра, $disp$ – дисперсия нормального случайного процесса. Здесь u, v, du, dv – переменные, w, ww, $disp$ – параметры.

Задача нахождения оценок параметров гауссовского сигнала в случае, когда неизвестен только один из параметров, два из трех и все три, достаточно подробно решена вручную [2, 3]. Однако для практического осуществления исследований и моделирования оценок в различных ситуациях наибольший интерес представляет поиск удобных средств автоматизации символьных вычислений.

Расчет совместных оценок максимального правдоподобия трех параметров плотности вероятности требует совместного решения трех уравнений, получаемых путем приравнивания

к нулю первых производных по соответствующим параметрам логарифма функции правдоподобия, приведенной к безразмерному виду [2]:

$$\begin{cases} \dfrac{\partial}{\partial(disp)} L(u,v,du,dv,\alpha,ww,disp) = 0, \\ \dfrac{\partial}{\partial(ww)} L(u,v,du,dv,\alpha,ww,disp) = 0, \\ \dfrac{\partial}{\partial(\alpha^2)} L(u,v,du,dv,\alpha,ww,disp) = 0. \end{cases}$$

Здесь

$$L(u,v,du,dv,\alpha,ww,disp) = \ln\left[\frac{k}{4}\cdot\frac{e^{-\frac{\left[(\alpha^2+ww^2)\cdot(u^2+v^2)+du^2+dv^2-2\cdot ww\cdot(u\cdot dv-v\cdot du)\right]}{2\cdot disp\cdot\alpha^2}}}{\pi^2\cdot disp^2\cdot\alpha^2}\right],$$

Решая данную систему современными средствами автоматизации вычислений, можно весьма быстро получить совместные безусловные максимально правдоподобные оценки (рис. 1).

При использовании пакета Maple для выполнения символьных вычислений следует учитывать некоторые особенности данного программного продукта. Данный пакет, обладая несомненно более мощным вычислительным аппаратом, чем традиционно используемые математические пакеты (например, MathCAD), представляет собой скорее язык программирования со своим специфическим синтаксисом [4], нежели просто математический редактор. Поэтому при производстве символьных вычислений не удается напрямую воспользоваться процедурой копирования результатов вычислений, что,

```
a— альфа квадрат  w—ср. частота  d — дисперсия

f4 := exp(-((a+w^2)·(u^2+v^2) - s^2 - t^2 + 2·w·(u·s - v·t))/(2·d·a)) / (4·(π)^2·d^2·a);

                                  1/4 exp(1/2 (-(a+w^2)(u^2+v^2) - s^2 - t^2 + 2w(us - vt))/(da)) / (π^2 d^2 a)

lf4 := ln(f4);

                                  ln(1/4 exp(1/2 (-(a+w^2)(u^2+v^2) - s^2 - t^2 + 2w(us - vt))/(da)) / (π^2 d^2 a))

simplify(lf4);

                                  -2 ln(2) - 2 ln(π) + ln(exp(-1/2 (au^2 + av^2 + w^2u^2 + w^2v^2 + s^2 + t^2 - 2wus + 2wvt)/(da)) / (d^2 a))

[

dlf4d := simplify(diff(lf4, d));

                                  1/2 (au^2 + av^2 + w^2u^2 + w^2v^2 + s^2 + t^2 - 2wus + 2wvt - 4da)/(d^2 a)

dlf4w := simplify(diff(lf4, w));

                                  -(wu^2 + wv^2 - us + vt)/(da)

dlf4a := simplify(diff(lf4, a));

                                  -1/2 (-s^2 - t^2 - w^2u^2 - w^2v^2 + 2wus - 2wvt + 2da)/(da^2)

sys := {dlf4d = 0, dlf4w = 0, dlf4a = 0};
{-1/2 (-s^2 - t^2 - w^2u^2 - w^2v^2 + 2wus - 2wvt + 2da)/(da^2) = 0, 1/2 (au^2 + av^2 + w^2u^2 + w^2v^2 + s^2 + t^2 - 2wus + 2wvt - 4da)/(d^2 a) = 0
, -(wu^2 + wv^2 - us + vt)/(da) = 0}

res := solve(sys, {d, w, a});

                                  {a = (v^2 s^2 + 2usvt + t^2u^2)/(v^4 + 2u^2v^2 + u^4), d = 1/2 u^2 + 1/2 v^2, w = -(-us + vt)/(u^2 + v^2)}
```

Рис. 1. Совместные безусловные максимально правдоподобные оценки, полученные посредством автоматизации вычислений в Maple

однако, легко обходится. Другим существенным отличием пакета Maple является тенденция поиска максимально точного решения, которое не всегда оказывается самым простым, поэтому в случае символьных вычислений целесообразно использование оператора упрощения *simplify*, который автоматически производит процедуры приведения подобных слагаемых и сокращения. К несомненным преимуществам пакета можно отнести способность корректно решать достаточно сложные (в том числе нелинейные) системы уравнений, не прибегая к методу подстановки. Этот путь может быстро привести к известным точным выражениям для оцениваемых параметров α^2, *ww* и *disp*, равным соответственно:

$$\alpha^2 = \left(\frac{u\cdot du + v\cdot dv}{u^2+v^2}\right)^2,$$

$$ww = \frac{u\cdot dv - v\cdot du}{u^2+v^2} \text{ и } disp = \frac{u^2+v^2}{2}.$$

Таким образом, получены формальные выражения для совместных максимально-правдо-

Fischer

$$int\left(e^{\frac{-1}{2}\cdot\left(\frac{2\cdot w\cdot u\cdot x + x^2}{d\cdot a}\right)}, x = -\infty..\infty\right);$$

Fischer

$$\begin{cases} \dfrac{e^{\left(\frac{1}{2}\frac{w^2u^2}{da}\right)}\sqrt{2}\sqrt{\pi}}{\sqrt{\frac{1}{da}}} & \operatorname{csgn}\left(\frac{1}{da}\right) = 1 \\ \infty & otherwise \end{cases}$$

$$int\left(e^{\frac{-1}{2}\cdot\left(\frac{-2\cdot w\cdot v\cdot y + y^2}{d\cdot a}\right)}, y = -\infty..\infty\right);$$

$$\begin{cases} \dfrac{e^{\left(\frac{1}{2}\frac{w^2v^2}{da}\right)}\sqrt{2}\sqrt{\pi}}{\sqrt{\frac{1}{da}}} & \operatorname{csgn}\left(\frac{1}{da}\right) = 1 \\ \infty & otherwise \end{cases}$$

$$int\left(u\cdot u\cdot e^{\frac{-1}{2}\cdot\left(\frac{a\cdot u^2}{d\cdot a}\right)}, u = -\infty..\infty\right);$$

$$\begin{cases} \dfrac{\sqrt{2}\,d\sqrt{\pi}}{\sqrt{\frac{1}{d}}} & \operatorname{csgn}\left(\frac{1}{d}\right) = 1 \\ \infty & otherwise \end{cases}$$

Интегральная оценка по Фишеру

Интегральная оценка по Фишеру

$$2\cdot\sqrt{2}\cdot\sqrt{\pi\cdot d}\cdot a\cdot\sqrt{2}\cdot\sqrt{\pi\cdot d}\cdot d\cdot a\cdot\sqrt{2}\cdot\sqrt{\pi\cdot d}\cdot a\cdot\sqrt{2}\cdot\sqrt{\pi\cdot d}\cdot\frac{1}{4\cdot\pi^2\cdot d^3\cdot a^4};$$

$$\frac{2}{a}$$

Рис. 2. Интегральная оценка частоты по Фишеру в Maple

подобных оценок трех искомых параметров плотности распределения. На основании данных выражений легко могут быть реализованы алгоритмы моделирования параметров гауссовского случайного процесса по известным реализациям компонент исходного аналитического сигнала, его первой производной.

Также средствами пакета Maple можно получить информационную матрицу Фишера, элементами которой [1] являются взятые с противоположным знаком математические ожидания вторых производных логарифма функции правдоподобия, приведенной к безразмерному виду $L(u,v,du,dv,\alpha,ww,disp)$ по оцениваемым параметрам.

Вычислим, например, информацию по Фишеру для оценки средней частоты (рис. 2). Как показано в [2], процесс вычисления математического ожидания второй производной логарифма функции правдоподобия по каждому из параметров, определяемый как соответствующий четырехкратный интеграл [1], сводится к последовательному взятию интегралов, представленных ниже (два из этих интегралов различаются только параметром интегрирования, поэтому описано вычисление только одного из них).

Пакет Maple позволяет силами встроенных библиотек интегрирования вычислять достаточно сложные интегралы от экспоненциально-степенных функций, которые представляют сложности для других математических пакетов (в пакете MathCAD требуется создание дополнительных пользовательских библиотек). К сожалению, возможности кратного интегрирования в пакете Maple ограничены взятием трехкратных интегралов [4]. Однако эта трудность может быть обойдена применением последовательного интегрирования.

Аналогичным образом можно получить и другие элементы матрицы Фишера (процесс точного определения компонент матрицы опущен ввиду громоздкости выкладок).

Полученная матрица Фишера имеет вид:

$$\begin{pmatrix} \frac{2}{disp^2} & 0 & \frac{1}{disp\cdot\alpha^2} \\ 0 & \frac{2}{\alpha^2} & 0 \\ \frac{1}{disp\cdot\alpha^2} & 0 & \frac{1}{\alpha^4} \end{pmatrix},$$

где на главной диагонали расположены информации по Фишеру (величины, обратные дисперсиям несмещенных эффективных оценок соответствующих параметров [1]) полученных оценок. Поскольку определитель матрицы Фишера отличен от нуля, это позволяет рассчитать матрицу ошибок, обратную информационной:

$$\begin{pmatrix} disp^2 & 0 & -disp\cdot\alpha^2 \\ 0 & \frac{1}{2}\alpha^2 & 0 \\ -disp\cdot\alpha^2 & 0 & 2\alpha^4 \end{pmatrix},$$

где на главной диагонали расположены дисперсии оценок параметров, а остальные элементы представляют собой смешанные корреляционные моменты полученных совместных оценок параметров.

Таким образом, применение современных математических пакетов позволяет получать выражения для совместных оценок параметров гауссовского сигнала по его комплексной модели, что дает более высокую точность алгоритмов по сравнению с алгоритмами, основанными на оценке параметров действительного сигнала.

Полученные совместные безусловные оценки представляют собой явные алгоритмы моделирования, легко реализуемые при помощи современных вычислительных средств.

Следует с сожалением отметить, что различные математические операции более удачно реализованы в разных математических пакетах. Возможности символьного интегрирования в MathCAD в значительной степени ограничены, зато есть возможность работы непосредственно с текущими результатами вычислений, без использования всякий раз оператора присваивания. Возможности пакета Maple, несомненно, более широки, но отсутствует интеграция с другими математическими пакетами (не считая усеченной возможности совместной работы с пакетом MATLAB) и полученные результаты могут быть использованы только внутренними процедурами и функциями пакета. Визуализация результатов здесь также может быть представлена только средствами пакета (входящий в состав пакета редактор не позволяет сохранять информацию в виде, доступном для других текстовых редакторов). Отмеченные обстоятельства существенно снижают эффективность применения современных вычислительных пакетов, но это все же несравненно более быстрый путь к результатам синтеза, анализа и моделирования совместных оценок параметров гауссовского сигнала, чем ручная работа.

Библиографический список

1. *Левин Б. Р.* Теоретические основы статистической радиотехники. М.: Мир, 1989. 653 с.

2. *Фалеев С. П.* Расчет и моделирование устройств обработки сигналов систем управления: учеб. пособие / Ленингр. электротехн. ин-т. Л., 1980. 110 с.

3. *Виноградова Е. П., Шепета А. П., Фалеев С. П.* Автоматизация анализа и синтеза оценок параметров комплексной модели сигнала. СПб.: ГУАП, 2002.

4. *Дьяконов В. П.* Компьютерная математика. Maple 10/11/12/13/14 в математических расчетах. М.: ДМК Пресс, 2018. 688 с.

УДК 004.94, 616-71 **DOI: 10.31799/978-5-8088-1701-2-2022-2-15-18**

Б. К. Акопян*
ассистент
О. О. Жаринов*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАЗРАБОТКА И ИССЛЕДОВАНИЕ ПОКАЗАТЕЛЕЙ КАЧЕСТВА АЛГОРИТМА ОБНАРУЖЕНИЯ И КЛАССИФИКАЦИИ НАРУШЕНИЙ СЕРДЕЧНОГО РИТМА

Предложен алгоритм классификации аритмий на основе автоматического алгоритма обнаружения аритмических эпизодов по сигналу электрокардиограммы. Осуществлена оценка вероятностей ошибок классификации методом статистического моделирования.
Ключевые слова: *электрокардиосигнал, электрокардиограмма, автоматическая обработка, классификация, принятие решений.*

B. K. Akopyan*
Assistant
O. O. Zharinov*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

DEVELOPMENT AND QUALITY ASSESSMENT OF THE CARDIAC RHYTHM DISORDERS DETECTION AND CLASSIFICATION ALGORITHM

An algorithm for classifying arrhythmias based on an automatic arrhythmic episodes detection algorithm is proposed. The probability of classification errors is evaluated by means of statistical modelling.
Keywords: *electrocardiosignal, electrocardiogram, automatic processing, classification, decision-making rule.*

Процесс классификации нарушений сердечного ритма можно разделить на две задачи: анализ типа зарегистрированных аритмических кардиокомплексов (сортировка кардиокомплексов каждого вида по группам «норма»/«патология») и анализ количества последовательно обнаруженных аритмических *RR*-интервалов и временных соотношений между ними [1].

Обобщенный принцип работы алгоритмов классификации нарушений сердечного ритма описан в [2, 3]. Исходя из него, можно сделать вывод, что для разработки алгоритма классификации аритмий необходимо решить две задачи:

1) выбрать алгоритм обнаружения аритмических эпизодов, обеспечивающий минимально возможную ошибку обнаружения;

2) предложить решающее правило, осуществляющее классификацию нарушений ритма на основе результатов обнаружения аритмических эпизодов.

Обоснование алгоритма обнаружения аритмических эпизодов

Исходя из обобщенного принципа работы алгоритмов экспресс-диагностики ЭКГ [2], очевидно, что результат работы системы принятия решения о классификации нарушений сердечного ритма будет напрямую зависеть не только от непосредственного правила классификации обнаруженного нарушения ритма, но и от качества работы алгоритмов обработки ЭКГ на этапе обнаружения QRS-комплексов и анализа нарушений ритма. Поэтому в качестве основы для разрабатываемого решающего плавила был выбран алгоритм обнаружения аритмий на основе цифровой фильтрации *QRS*-комплексов с решающим правилом на основе измерения разности среднеквадратического отклонения (СКО) *RR*-интервалов на скользящем окне, описанный в статье [4], показавший пригодные для практического применения результаты по критерию минимальной вероятности ошибочного решения.

Также преимуществом данного алгоритма является то, что результаты его работы проще анализировать с точки зрения ЭКГ-признаков рассматриваемой аритмии. В частности, по количеству автоматически обнаруженных последовательных эпизодов нарушения ритма и временных соотношений между ними можно осуществить классификацию сердечного ритма по трем категориям:

– нормальный ритм,

– одиночные нарушения ритма (например, экстрасистолия),

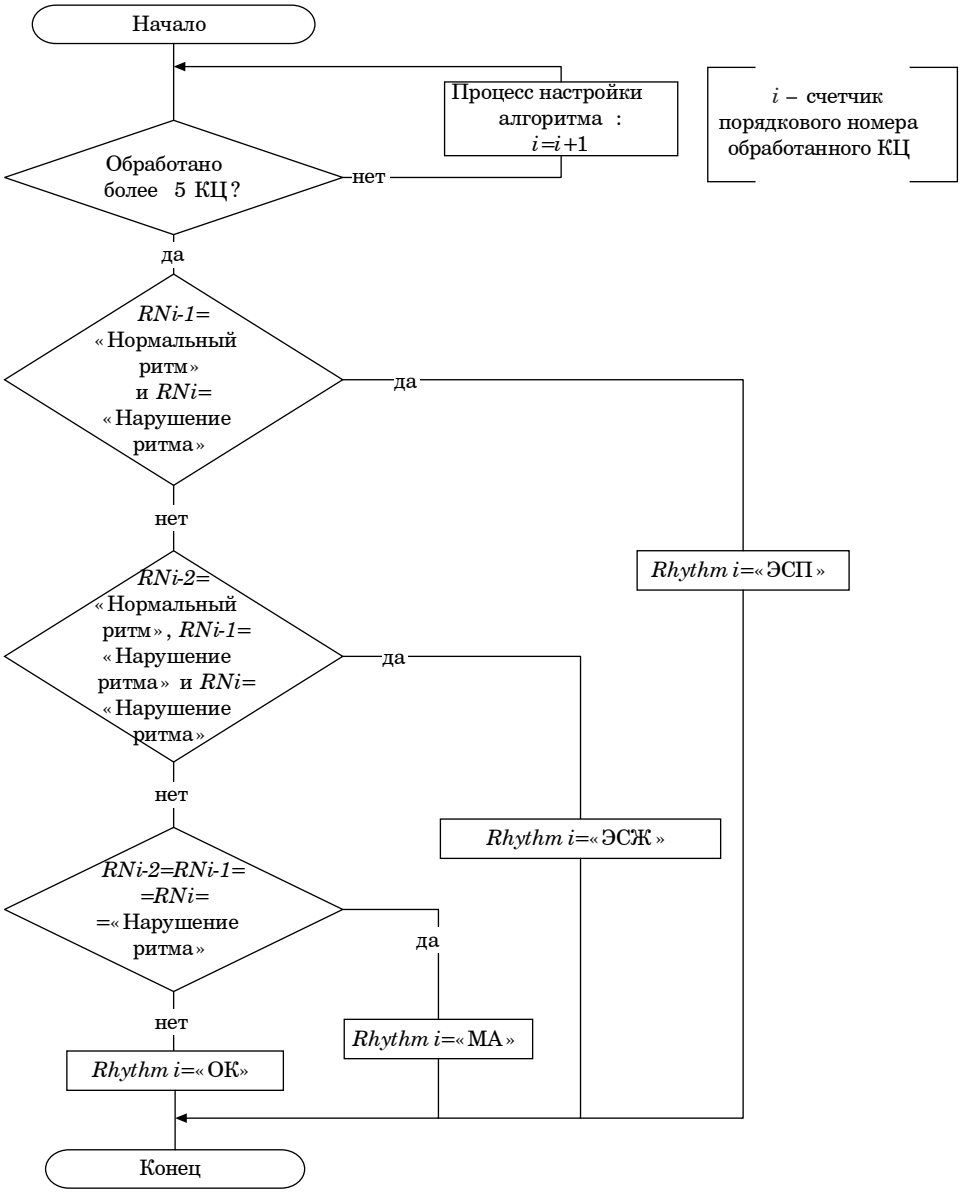
– частые нарушения ритма (например, мерцательная аритмия как один из наиболее распространенных типов аритмии [5]).

Разработка решающего правила классификации аритмических эпизодов

Исходя из анализа результатов работы алгоритма, представленных в статье [4], можно выделить следующие критерии классификации для рассматриваемых в исследовании категорий сигналов:

– одиночной предсердной экстрасистоле соответствует одно зафиксированное нарушение (интервал сцепления экстрасистолы);

– одиночной желудочковой экстрасистоле соответствуют два нарушения ритма, зафиксированных подряд (интервал сцепления экстрасистолы и компенсаторная пауза);



Блок-схема решающего правила классификации нарушения ритма записи сигнала ЭКГ

– мерцательной аритмии соответствуют три и более нарушения ритма подряд (хаотичный характер сердечного ритма).

В соответствии с данными выводами, предлагается следующее решающее правило для классификации ЭКГ в зависимости от зарегистрированных аритмических эпизодов. Блок-схема предлагаемого алгоритма классификации представлена на рисунке. Исходя из принципа работы алгоритма обнаружения аритмий [3], первые пять результатов принятия решения о состоянии ритма являются частью этапа настройки и не нуждаются в классификации.

Классификация обнаруженных нарушений ритма производится по категориям:

– одиночный эпизод нарушения ритма классифицируется как предсердная экстрасистола (состояние «ЭСП» в блок-схеме алгоритма);

– два эпизода нарушения ритма, следующие друг за другом, классифицируются как желудочковая экстрасистола (состояние «ЭСЖ»);

– три эпизода нарушения ритма, следующие друг за другом, классифицируются как продолжительный аритмический эпизод (состояние «МА»).

В противном случае, если нарушения ритма зафиксировано не было, состояние ритма определено как нормальное (состояние «ОК»).

Результаты моделирования

Для проведения исследования применены записи ЭКГ реалистичной формы, созданные с помощью модели ЭКГ [6, 7] на основе параметров реальных записей ЭКГ, представленных в базах данных (БД) MIT-BIH Normal Sinus Rhythm Database [8] и MIT-BIH Arrhythmia Database [9]. Было использовано 1200 записей длительностью 120 с с частотой дискретизации 1000 Гц, поделенных на три категории:

1) ЭКГ нормальной формы с параметром вариабельности сердечного ритма ±0–5%;

2) ЭКГ с имитацией одиночных экстрасистолических нарушений сердечного ритма (предсердных и желудочковых экстрасистол), составляющих не более 10% от общего числа кардиоциклов ЭКГ;

3) ЭКГ с мерцательной аритмией.

Любая ЭКГ из базы классифицируется однозначно и не может быть отнесена к другой категории. Исходя из этого, для моделирования этапа классификации диагностированных аритмий достаточно классифицировать непосредственно записи ЭКГ. Если количество нарушений в записи ЭКГ, относящихся к определенной категории, превышает некоторое пороговое значение, то в соответствии с данной категорией классифицируется и вся запись ЭКГ.

Записи классифицируются следующим образом:

– если преобладают одиночные нарушения ритма (состояния «ЭСП» и «ЭСЖ»), то запись соответствует одиночной экстрасистолии;

– продолжительные эпизоды аритмии (состояние «МА») – мерцательной аритмии.

Если доля классифицированных нарушений не превысила порогового значения, данная запись классифицируется как нормальная, что позволит исключить влияние незначительных нарушений ритма у здорового человека на результаты обследования. Исходя из результатов анализа работы алгоритма обнаружения аритмий [3], было решено выбрать пороговое значение общего количества нарушений для выставления диагноза, равное 3,5%. Результаты оценок вероятностей ошибок классификации представлены в таблице.

Результаты оценок вероятностей ошибок предлагаемого алгоритма классификации аритмий

Истинное состояние ЭКГ	Вероятность классифицировать как		
	нормальный ритм	одиночные нарушения ритма (экстрасистолия)	частые нарушения ритма (мерцательная аритмия)
Нормальный ритм	0,993	0,007	0
Одиночные нарушения ритма (экстрасистолия)	0,005	0,985	0,010
Частые нарушения ритма (мерцательная аритмия)	0	0,005	0,995

Полученные показатели качества классификации удовлетворительны для алгоритма обнаружения аритмий, применяемого на практике, следовательно, рассматриваемый алгоритм можно применять для анализа трех приведенных категорий ЭКГ.

Библиографический список

1. *Анциперов В. Е., Забросаев И. В., Растягаев Д. В.* Детектирование нарушений сердечного ритма с использованием техники аналитических спектров // Журнал радиоэлектроники. 2015. № 12. URL: http://jre.cplire.ru/jre/dec15/4/text.html (дата обращения 25.10.2021).

2. *Акопян Б. К.* Обзор алгоритмов обработки электрокардиограмм для систем поддержки принятия решения о классификации нарушений сердечно-

го ритма // Обработка, передача и защита информации в компьютерных системах: сб. докл. Междунар. науч. конф., СПб., 14–22 апр. 2021 г. СПб.: ГУАП, 2021. С. 7–12.

3. Кардиомониторы. Аппаратура непрерывного контроля ЭКГ: учеб. пособие для вузов / под ред. А. Л. Барановского и А. П. Немирко. М.: Радио и связь, 1993. 248 с.

4. *Akopyan B. K.* Development and research of automated arrhythmic episodes detection algorithms by electrocardiographic signal // Bulletin of the UNESCO department «Distance education in engineering» of the SUAI: collection of the papers. Iss. 6. SPb.: SUAI, 2021. P. 29–34.

5. *Рузов В. И., Гимаев Р. Х., Разин В. А.* Основы клинической электрокардиографии. Практическое руководство по внутренним болезням: учеб. пособие. Ульяновск: УлГУ, 2009. 124 с.

6. *Акопян Б. К., Жаринов О. О.* Разработка компьютерной имитационной модели электрокардиограммы // Обработка, передача и защита информации в компьютерных системах: сб. докл. I Всерос. науч. конф. (СПб., 14–22 апр. 2020 г. СПб.: ГУАП, 2020. С. 17–23.

7. *Акопян Б. К., Жаринов О. О.* Разработка программного обеспечения для имитационного моделирования сигнала электрокардиограммы // Обработка, передача и защита информации в компьютерных системах: Междунар. науч. конф.: сб. докл. (СПб., 14–22 апр. 2021 г.) СПб.: ГУАП, 2021. С. 13–19.

8. MIT-BIH Normal Sinus Rhythm Database. URL: https://physionet.org/content/nsrdb/1.0.0/ (дата обращения 15.10.2021).

9. MIT-BIH Arrhythmia Database. URL: https://physionet.org/content/mitdb/1.0.0/ (дата обращения 15.10.2021).

УДК 004.942, 616-71 **DOI: 10.31799/978-5-8088-1701-2-2022-2-19-22**

Б. К. Акопян*
ассистент
А. П. Шепета*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОСОБЕННОСТИ ОЦЕНКИ АРТЕРИАЛЬНОГО ДАВЛЕНИЯ ПРИ АВТОМАТИЗИРОВАННОМ ИЗМЕРЕНИИ ЭЛЕКТРОННЫМ ТОНОМЕТРОМ

Предложен метод оценки артериального давления как крайних значений статистики вариационного ряда: максимума для систолического и минимума для диастолического. В сравнении с традиционным методом оценки значения артериального давления как среднего арифметического трех измерений, широко применяемого при автоматизированных измерениях электронными тонометрами, предложенный метод имеет более высокую точность, поскольку основан на достаточных статистиках, используемых для оценки крайних границ финитных распределений.
Ключевые слова: *измерение артериального давления, финитное распределение, достаточная статистика, статистическое моделирование.*

B. K. Akopyan*
Assistant
A. P. Shepeta*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

FEATURES OF THE BLOOD PRESSURE ESTIMATION DURING AUTOMATED MEASUREMENT WITH AN ELECTRONIC MONITORING DEVICE

The method of blood pressure estimation as extremes of statistics of the variational series is proposed: maximum for systolic and minimum for diastolic. Compared to the traditional method of estimating the value of arterial pressure as the average of three measurements, widely used in automated measurements by electronic monitoring devices, the proposed method has a higher accuracy, because it is based on sufficient statistics used to estimate the limits of finite distributions.
Keywords: *blood pressure measurement, uniform distribution, statistics, statistical modelling.*

Введение

Принцип работы большинства электронных тонометров основан на аускультативном методе Н. С. Короткова. Для регистрации артериального давления (АД) на обследуемого надевается манжета, которая полой трубкой соединена с регистрирующим устройством. После включения устройства компрессор начинает заполнять воздухом манжету, повышая внутри нее давление p_m (рис. 1), которое постоянно измеряется датчиком. Как только значение давления в манжете достигает пороговой величины (в зависимости от устройства, она может равняться 170–180 мм рт. ст.), компрессор отключается, воздух из манжеты стравливается, вследствие чего давление в манжете плавно уменьшается. Чувствительный датчик определяет время появления t_s и затухания t_d звуковых волн тонов Короткова, частота которых соответствует частоте сердечных сокращений, и регистрирует соответствующие значения давления $\hat{p}_s$ и $\hat{p}_d$ в манжете. Сердечные сокращения при нормальном ритме возникают относительно равномерно, с учетом вариабельности сердечного ритма допускается отклонение длительности RR-интервалов не более чем на 10% [2, 3].

Метод Короткова обладает погрешностью [4], при которой результаты измерения АД всегда будут отклоняться от истинного значения артериального давления, так как тоны Короткова возникают позже того момента времени, когда давление в манжете становится равным истинному систолическому давлению p_s, и затухают

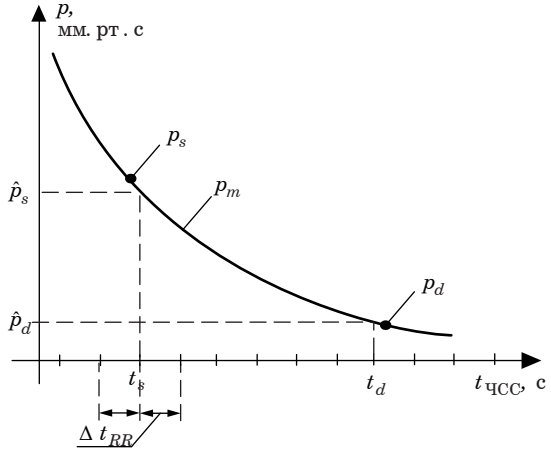


Рис. 1. Кривая зависимости давления воздуха в манжете электронного тонометра от времени

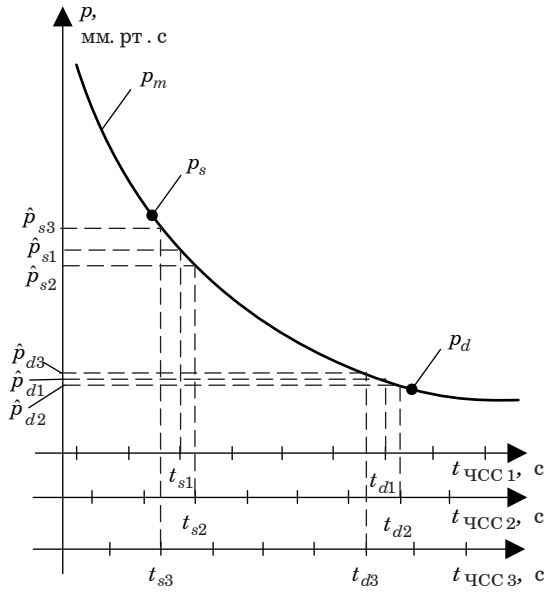


Рис. 2. Результаты трех измерений по методу Короткова на кривой зависимости давления воздуха в манжете электронного тонометра от времени

раньше, чем давление в манжете достигает значения истинного диастолического давления p_d. В результате оценка систолического давления будет меньше истинного значения, а оценка диастолического давления – больше (рис. 2).

Методики оценки артериального давления

В соответствии с клиническими рекомендациями мировых медицинских обществ по артериальной гипертонии [5], в частности Министерства здравоохранения Российской Федерации и Российского медицинского общества по артериальной гипертонии [6], для определения уровня артериального давления требуется выполнить не менее двух измерений, при разнице значений измерения, превышающей 5 мм рт. ст., требуется произвести дополнительное измерение. В качестве оценки значения артериального давления $\left[\hat{p}_{ci}; \hat{p}_{\partial i}\right]$ используется среднее арифметическое полученных измерений:

$$\begin{cases} \hat{p}_s = \dfrac{p_{s1} + p_{s2} + p_{s3}}{3}, \\ \hat{p}_d = \dfrac{p_{d1} + p_{d2} + p_{d3}}{3}, \end{cases} \quad (1)$$

где p_{si} и p_{di} – результаты i-го измерения артериального давления, $i = 1, 2, 3$.

Современные электронные тонометры осуществляют это автоматически. Технология вычисления значения артериального давления Microlife Average Mode [7] состоит в проведении серии из трех измерений с паузой в 15 с, после чего их среднее арифметическое фиксируется в качестве оценки артериального давления. Устройства, основанные на данной технологии оценки АД, широко применяются при экспресс-диагностике состояния обследуемого [8, 9]. Примеры подобных устройств приведены в работах [10–14].

Недостатком применения среднего арифметического в качестве оценки значения артериального давления является то, что в случае пропуска одного сердечного сокращения или сдвига ритма вследствие компенсаторной паузы при одиночной экстрасистолии оценка с помощью среднего арифметического может быть существенно искажена по сравнению с истинным значением, особенно в тех случаях, когда экстрасистол предшествует измерению. На кривой давления воздуха в манжете (рис. 3) представлены результаты оценки значения АД в случае нормального ритма $t_{\text{ЧСС}}$ и одиночной предсердной экстрасистолии $t'_{\text{ЧСС}}$. Предсердная экстрасистолия характеризуется неполной компенсаторной паузой, т. е. длительность суммы RR-интервалов до экстрасистолы и после нее меньше, чем длительность любых двух последовательных RR-интервалов последующего течения ритма [15–17]. По графику видно, что в случае возникновения одиночной предсердной экстрасистолы в момент времени t_{es} и последующей нормализации ритма значения систолического $\hat{p}'_s$ и диастолического $\hat{p}'_d$ давления будут измерены с отклонением от значений $\hat{p}_s$ и $\hat{p}_d$ при нормальном ритме, что приведет к искажению оценки при использовании формулы среднего арифметического.

При измерении значения артериального давления в интервале $[t_i; t_{i+1}]$ фактически нужно определять левую (систолическое давление)

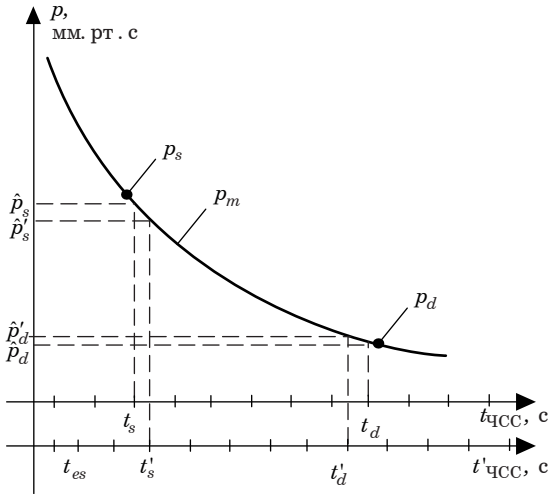


Рис. 3. Определение значения артериального давления электронным тонометром в случае нормального ритма $t_{ЧСС}$ и одиночной предсердной экстрасистолии $t'_{ЧСС}$ на графике кривой давления воздуха в манжете

и правую (диастолическое давление) границы финитных распределений. Достаточными статистиками для оценки этих границ являются минимальное и максимальное значение соответствующих вариационных рядов, поэтому оценка границ должна производиться по выражениям

$$\begin{cases} \hat{p}_s = \max\left(p_{s1}, p_{s2}, p_{s3}\right), \\ \hat{p}_d = \min\left(p_{d1}, p_{d2}, p_{d3}\right), \end{cases} \quad (2)$$

являющимся оценками систолического $\hat{p}_s$ и диастолического $\hat{p}_d$ давлений.

Поскольку оценки (2) сформированы на основе достаточных статистик, то они будут иметь меньшую погрешность по сравнению с оценками, определенными выражением (1), что, собственно, и проиллюстрировано на рис. 2.

Результаты исследования

Для численного сравнения методов оценки было проведено статистическое моделирование процесса измерения артериального давления. Модель кривой давления в манжете представлена как экспоненциальная функция вида

$$p(t) = p_0 e^{-\alpha t}, t \geq 0, \quad (3)$$

где p_0 – первоначальное значение давления воздуха в манжете на момент начала стравливания ($t = 0$), α – параметр, физически представляющий скорость стравливания воздуха, $\alpha > 0$, t – время. В рамках одного измерения осуществляются следующие действия: начиная со случайного момента времени, формируется выборка равноудаленных друг от друга моментов появления сердечных сокращений. Алгоритм определяет момент появления t_s и затухания t_d звуковых волн тонов Короткова как временны́е координаты сердечных сокращений, при которых значения АД ближе всего к истинным, и фиксирует в выборке соответствующие значения давления $\hat{p}_s$ и $\hat{p}_d$. Данные измерения повторяются три раза, после чего рассчитываются оценки АД по формулам (1) и (2).

В рамках измерения АД по исследуемым методикам получены результаты статистического моделирования, приведенные в табл. 1, 2. Оценка относительной погрешности измерений от истинного значения АД осуществляется по формуле

$$\sigma(p) = \frac{\left|\overline{p} - p_{real}\right|}{p_{real}}, \quad (4)$$

где $\overline{p}$ – оценка значения АД по выражениям (1) или (2), p_{real} – истинное значение артериального давления. Из результатов, приведенных в таблицах, видно, что выборочное среднее оценок значений артериального давления, определенных по среднему арифметическому трех измерений, отклоняется от истинного значения больше, чем выборочное среднее оценок значений АД, определенное с применением крайних значений вариационного ряда, что хорошо видно и на рис. 2.

Таблица 1

Отклонение оценки выборочного среднего от истинного значения артериального давления по результатам статистического моделирования, количество экспериментов M = 10^5

Систолическое давление, мм рт. ст.			Диастолическое давление, мм рт. ст.		
p_{real}	Отклонение оценки по среднему арифметическому от истинного значения	Отклонение оценки по крайнему значению вариац. ряда от истинного значения	p_{real}	Отклонение оценки по среднему арифметическому от истинного значения	Отклонение оценки по крайнему значению вариац. ряда от истинного значения
100	0,052	0,026	60	0,027	0,013
110	0,055	0,028	70	0,038	0,019
120	0,059	0,030	80	0,047	0,022
130	0,066	0,035	90	0,054	0,026

Таблица 2

Математическое ожидание результатов измерения артериального давления по результатам статистического моделирования при заданном истинном значении артериального давления 120/80, количество экспериментов M = 10^5

Систолическое давление, мм рт. ст.		Диастолическое давление, мм рт. ст.	
Оценка по среднему арифметическому трех измерений	Оценка по крайнему значению вариац. ряда	Оценка по среднему арифметическому трех измерений	Оценка по крайнему значению вариац. ряда
112,9	116,4	83,7	81,8

Заключение

Предложенный метод оценки систолического и диастолического давления по значениям вариационного ряда малой выборки, состоящей из трех измерений, имеет меньшую методическую погрешность в сравнении с рекомендуемым в официальных источниках методом оценки среднего арифметического трех измерений. Средняя относительная погрешность измерения давления традиционным методом находится в пределах 5,5–6,5% при измерении систолического давления и в пределах 2,5–5,5% при измерении диастолического давления, средняя же погрешность предложенного метода в два раза меньше. Следует отметить и тот факт, что предложенный метод более устойчив к измерению давления при экстрасистолах предшествующих интервалу измерений, хотя точность его при этом падает.

Библиографический список

1. *Цырлин В. А., Плисс М. Г., Кузьменко Н. В.* История измерения артериального давления: от Хейлса до наших дней // Артериальная гипертензия. 2016. Т. 22, № 2. С. 144–152.

2. *Иванов Г. Г., Дворников В. Е., Сбеитан С.* и др. Анализ показателей структуры вариабельности ритма сердца у здоровых лиц по данным PP- и RR-интервалов // Вестник Российского университета дружбы народов. Серия: Медицина. 2007. № 4. С. 26–34.

3. *Акопян Б. К., Жаринов О. О.* Разработка программного обеспечения для имитационного моделирования сигнала электрокардиограммы // Обработка, передача и защита информации в компьютерных системах: сб. докл. Междунар. науч. конф., СПб., 14–22 апр. 2021 г. СПб.: ГУАП, 2021. С. 13–19.

4. *Манвелов Л. С., Кадыков А. В.* Артериальное давление и техника его измерения // Российский медицинский журнал. 2015. № 1. С. 49–51.

5. *Wang J. G., Bu P. L., Chen L. Y.* et al. Chinese Hypertension League guidelines on home blood pressure monitoring // Journal of Clinical Hypertension. 2020. № 22. P. 378–383.

6. *Чазова И. Е., Жернакова Ю. В.* (от имени экспертов). Клинические рекомендации. Диагностика и лечение артериальной гипертонии // Системные гипертензии. 2019. Т. 16, № 1. С. 6–31.

7. *Смогунов В. В., Кузнецов Н. С.* Системный анализ проблем самоконтроля интегративной метасистемы кровообращения // Известия Самарского научного центра Российской академии наук. 2015. Т. 17, № 2 (5). С. 1141–1146.

8. *Wilton A., De Greef A., Shennan A.* Rapid Assessment of Blood Pressure in the Obstetric Day Unit Using Microlife MAM Technology // Hypertension in Pregnancy. 2007. № 26 (1). P. 31–37.

9. *Burnier M., Gasser U. E.* End-digit preference in general practice: a comparison of the conventional auscultatory and electronic oscillometric methods // Blood Pressure. 2008. Vol. 17, № 2. P. 104–109.

10. Патент на изобретение RU 2332925 C2, 10.09.2008. Заявка № 2006129231/14 от 11.08.2006. Электронный монитор артериального давления, вычисляющий среднее значение артериального давления / Кисимото Х., Саваной Ю., Танака Т., Эда К.

11. Патент на изобретение RU 2396898 C2, 20.08.2010. Заявка № 2008129706/14 от 22.11.2006. Электронное устройство измерения артериального давления, вычисляющее значение артериального давления / Сано Й., Такахаси А., Ямасита С., Саваной Ю.

12. *Palatini P., Dorigatti F., Bonso E., Ragazzo F.* Validation of the Microlife BP W200-1 wrist device for blood pressure measurement // Blood Pressure Monitoring. 2008. Vol. 13, № 5. P. 295–298.

13. *Stergiou G. S., Jaenecke B., Giovas P. P.* et al. A tool for reliable self-home blood pressure monitoring designed according to the European Society of Hypertension recommendations: the Microlife WatchBP Home monitor // Blood Pressure Monitoring. 2007. Vol. 12, № 2. P. 127–131.

14. *Stergiou G. S., Lin C. W., Lin C. M., Chang S. L.* et al. Automated device that complies with current guidelines for office blood pressure measurement: design and pilot application study of the Microlife WatchBP Office device // Blood Pressure Monitoring. 2008. Vol. 13, № 4. P. 231–235.

15. Основы клинической электрокардиографии / под ред. И. Г. Меньшиковой. Благовещенск, 2010. 112 с.

16. *Акопян Б. К., Жаринов О. О.* Разработка компьютерной имитационной модели электрокардиосигнала // Обработка, передача и защита информации в компьютерных системах: сб. докл. I Всерос. науч. конф. СПб.: ГУАП, 2020. С. 17–23.

17. *Жаринов О. О., Шепета А. П.* Методика обнаружения микропотенциалов ЭКГ // Информационно-управляющие системы. 2002. № 1 (1). С. 48–51.

УДК 004.622 **DOI: 10.31799/978-5-8088-1701-2-2022-2-23-27**

Е. А. Бакин*
кандидат технических наук, доцент
В. В. Вихров*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОЦЕНКА КАЧЕСТВА РЕЗУЛЬТАТОВ АВТОМАТИЧЕСКОГО СБОРА ОТКРЫТЫХ ДАННЫХ О ЦЕНАХ НА НЕДВИЖИМОСТЬ

Проанализированы данные о цене недвижимости в крупных городах России, полученные при помощи инструмента для сбора информации парсера с сайтов – агрегаторов недвижимости. Рассмотрены количественные и качественные характеристики полученных данных, а также оценены их характеристики и достоверность.
Ключевые слова: *цена на недвижимость, парсер, автоматизация обработки данных, оценка рынка недвижимости, предобработка данных.*

E. A. Bakin*
PhD, Tech., Associate Professor
V. V. Vikhrov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

QUALITY ESTIMATES OF AUTOMATIC COLLECTION RESULTS OF OPEN REAL ESTATES PRICES DATA

In this article a price data of real estate in large Russian cities are analyzing. The data was obtained from real estate aggregator sites using a data collection tool called a parser. The quantitative and qualitative characteristics of the obtained data are considered, as well as their characteristics and reliability are assessed.
Keywords: *real estate price, parser, data processing automation, real estate market assessment, data preprocessing.*

Предварительная обработка данных – важный шаг в процессе интеллектуального анализа данных. Фраза «мусор на входе — мусор на выходе» применима, в частности, и для проектов интеллектуального анализа данных и машинного обучения. Здесь имеется в виду, что даже самый изощренный анализ не принесет пользы, если за основу взяты сомнительные данные [1].

Построение систем предсказания стоимости недвижимости осложняется отсутствием в открытом доступе подходящих по объему данных для анализа, которые также часто неполны, что приводит к значительной ошибке предсказания даже для наиболее эффективных алгоритмов. В связи с этим был создан парсер для сбора данных о недвижимости из открытых источников. В данном исследовании будут проанализированы данные, полученные при помощи парсера.

Для последующей оценки и предобработки полученных данных важно понимать, как работает парсер. Он не берет случайные квартиры из всего списка, а применяет различные фильтры и сортировки для получения некоторого количества страниц с общим описанием квартир и ссылками на них. После получения очередной страницы с ссылками на квартиры он открывает ссылку на каждую квартиру из списка и собирает полученные данные.

В связи с тем что фильтры задаются вручную, нельзя доподлинно говорить о соотношении количества полученных квартир к общему количеству по задаваемым фильтрам. Фильтры, которые задавались для парсера в данном случае, – количество комнат и тип квартиры (новостройка или вторичное жилье). Поэтому относительные показатели – процент квартир разного типа и с разным количеством комнат к общему количеству полученных квартир анализировать не стоит.

При работе парсер обрабатывал разное количество квартир за час, в зависимости от города, скорее всего, из-за разной степени защиты серверов в данных городах. В лучшем случае он

обрабатывал около 1500 квартир в час, в худшем ему могло понадобится около 6 ч на обработку 3000 квартир, т. е. 500 квартир в час. Сложнее всего было работать с Санкт-Петербургом и Москвой, здесь приходилось задавать большие тайминги «сна» кода для того, чтобы не слать большое количество запросов серверу единовременно.

Парсер собрал списки квартир в пяти самых больших городах по населению в России, а именно в Москве, Санкт-Петербурге, Новосибирске, Екатеринбурге и Казани. Важно понимать, что люди, которые создавали карточки квартир, часто не указывали некоторые факторы, например высоту потолков. Относительные показатели по соотношению полученных факторов квартир к общему количеству представлены в табл. 1.

Из табл. 1 можно сделать вывод, что для всех квартир заполнены полностью только общая площадь, количество комнат и этаж. Также указан адрес (в таблице не представлен, так как в дальнейшем исследовании не используется), цена общая и цена за квадратный метр, цена – это оцениваемая величина, без нее не имеет смысла рассматривать квартиру. Почти всегда плохо заполнены поля с типом дома, классом квартиры, высотой потолков (за исключением Москвы), площадью комнат в Новосибирске, Екатеринбурге и Казани.

Все показатели были получены после предобработки данных, включающих удаление пустых и дублирующихся строк. Из-за того, что нередко адрес пишут без номера квартиры и в новостройках квартиры имеют одинаковые факторы, при обработке такие квартиры будут удалены, за исключением одного вхождения каждой такой повторяющейся квартиры.

После удаления дублирующихся и плохо обработанных парсером строк (пустые, в которых есть только ссылка) массивы данных в среднем «худеют» на 30–40 %, наибольшие потери наблюдались у Москвы, получено было около 27000 квартир, после предобработки осталось около 16000.

Некоторая информация, полученная парсером, представляла собой совокупность факторов, например парсер собирал информацию об этаже, в виде текста «2 из 10», где 2 – этаж, на котором находится квартира, а 10 – общее количество этажей. Такую информацию при предобработке данных было решено делить на отдельные факторы: этаж квартиры и этажность дома. Также для каждой из квартир с указанной станцией метро был создан показатель линии метро с целью укрупнения категорий квартир по типу ветки метро.

В итоге для каждого из городов были обработаны данные и получены факторы квартир, которые впоследствии будут использованы для создания модели машинного обучения. Факторы, полученные после предобработки, их тип и примеры представлены в табл. 2.

Некоторые из факторов имеют тип «Факторный», такую информацию модель машинного обучения будет воспринимать не как некое чис-

Таблица 1

Соотношение полученных факторов квартир к общему числу, %

Фактор	Москва	Санкт-Петербург	Новосибирск	Екатеринбург	Казань
Площадь общая	100,00	100,00	100,00	100,00	100,00
Жилая площадь	76,11	85,69	80,57	63,70	75,72
Площадь кухни	78,81	72,51	83,53	70,04	81,72
Площадь комнат	61,36	64,07	24,07	25,22	46,38
Высота потолков	55,25	38,46	6,04	24,78	23,42
Тип ремонта	50,95	59,24	43,98	46,02	32,19
Класс квартиры	16,30	13,41	3,65	13,50	7,70
Тип дома	16,30	13,41	3,65	13,50	7,70
Этаж	100,00	100,00	100,00	100,00	100,00
Балкон	44,14	60,72	73,48	59,85	74,37
Санузел	79,31	78,72	79,51	63,23	62,43
Метро	90,01	80,49	45,03	49,90	57,97
Количество комнат	100,00	100,00	100,00	100,00	100,00

Таблица 2

Факторы, полученные после предобработки

Фактор	Внутреннее название	Пример	Тип
Цена, руб.	price	10000000	Числовой
Цена за 1 кв. м	price_m2	124530	Числовой
Общая площадь, кв. м	square	80,3	Числовой
Жилая площадь, кв. м	square_live	60	Числовой
Площадь кухни, кв. м	kitchen	10,5	Числовой
Этаж квартиры	floor	15	Числовой
Количество этажей в доме	floor_max	25	Числовой
Тип жилья	type	Новостройка	Факторный
Площадь комнат, кв. м	square_rooms	30;10;20	Числовой
Высота потолков, м	ceiling	2,7	Числовой
Ремонт	state	Евроремнот	Факторный
Метро	subway	Чкаловская	Факторный
Класс жилья	class	Премиум	Факторный
Тип дома	type_of_building	Панельный	Факторный
Цена в долларах	price_dollars	139000	Числовой
Год окончания строительства (для новостроек)	time_end_year	2022	Факторный
Квартал окончания строительства (для новостроек)	time_end_quarter	4	Факторный
Количество комнат	rooms	3	Числовой
Вид балкона	type_balcony	Лоджия	Факторный
Количество балконов	count_balcony	2	Числовой
Вид санузла	type_bathroom	Совмещенный	Факторный
Количество санузлов	count_bathroom	1	Числовой
Линия метро	line_subway	Зелёная	Факторный
Вид времени от метро	type_time_to_subway	пешком	Факторный
Время от метро, мин	count_time_to_subway	15	Числовой

ло, которое можно сравнивать в абсолютных значениях, а как некую категорию квартир для группировки по этому признаку, для последующего сравнения по абсолютным значениям.

После предобработки данных 14 показателей квартиры были разделены на 25 для корректной работы модели машинного обучения в будущем.

Отличительная черта рынка недвижимости – особенный товар, а именно объект недвижимости. В отличие от других товаров, объект недвижимости является неподвижным и непереносим [2].

Изменение рыночной стоимости любой жилой недвижимости зависит от ряда факторов, в исследовании рассматривается уровень влияния факторов, связанных с объектом недвижимости и во многом обусловленных его характеристиками [3].

Для того чтобы понять структуру некоторых факторов квартир и оценить их реалистичность, построим графики зависимость цены от фактора (будут представлены квартиры из города Москвы).

На рис. 1 можно увидеть уверенную зависимость цены от общей площади квартиры, а наиболее встречающееся их соотношение – это 130–140 тыс. долл. за 35–40 кв. м общей площади. Размах по цене составляет от 20 тыс. долл. до 1400 тыс. долл., размах площади при этом от 8 до 172 кв. м.

На рис. 2 представлен график зависимости цены от количества комнат: студии (0 комнат) стоят меньше, чем однокомнатные квартиры. На такой очевидной зависимости можно понять, что данные достаточно корректно отражают рынок недвижимости.

На рис. 3 представлена зависимость цены от линии метро, для упрощения отрисовки названия линий были заменены соответствующими номерами. Можно заметить, что квартиры на 18-й линии метро (МДЦ2) стоят меньше, чем квартиры на 5-й линии (кольцевая), скорее всего, это связано с тем, что кольцевая ветка метро

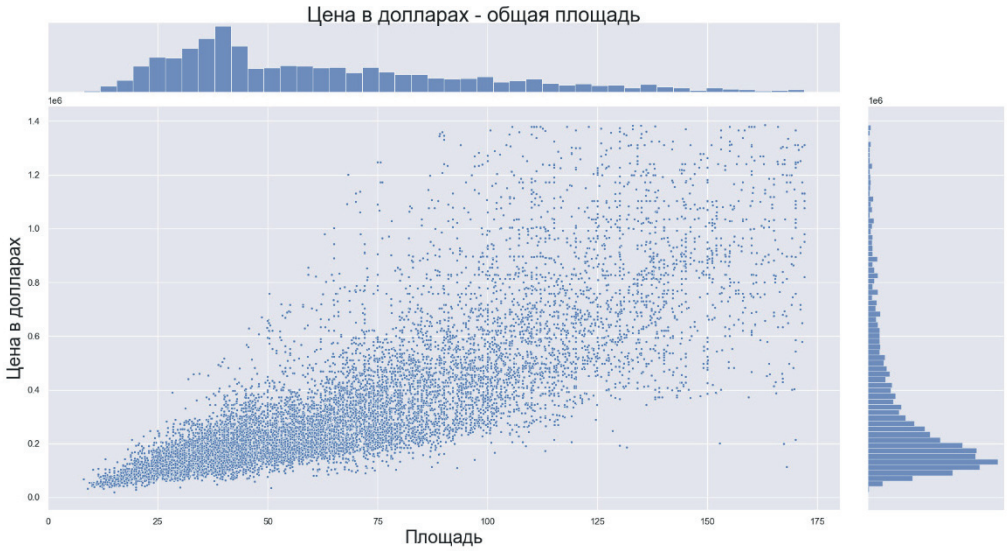


Рис. 1. Зависимость цены от общей площади квартиры

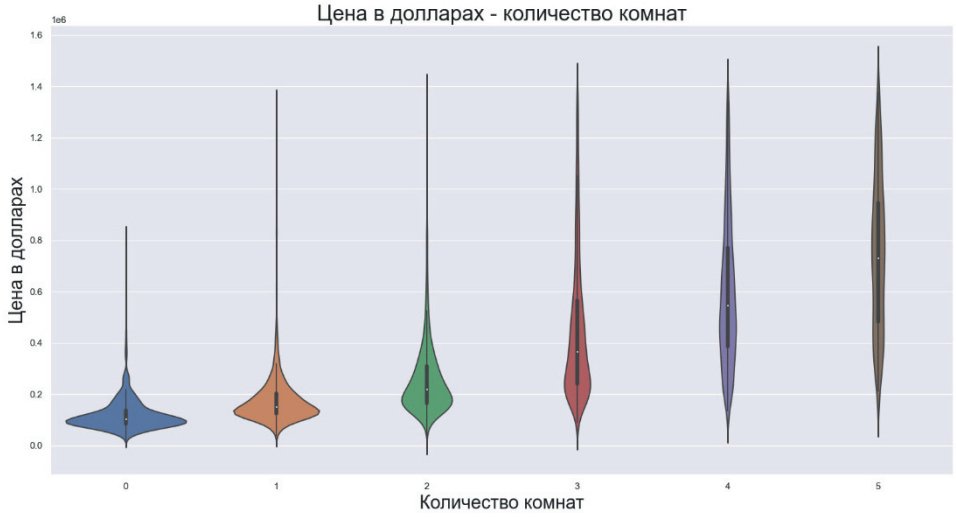


Рис. 2. Зависимость цены от количества комнат

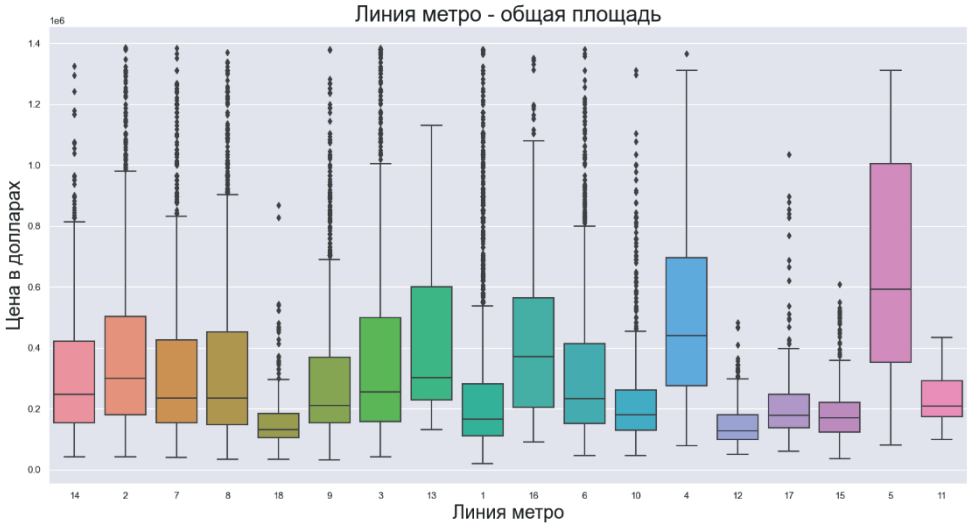


Рис. 3. Зависимость цены от линии метро

Таблица 3

Статистические показатели полученных факторов недвижимости (для г. Москвы)

Показатель	Общая площадь, кв. м	Жилая площадь, кв. м	Высота потолков, м	Время до метро, мин	Количество санузлов	Количество балконов
mean	62,933721	34,89	2,91	9,45	1,33	1,17
std	35,095948	22,25	0,32	6,03	0,58	0,42
min	8,100000	1,00	2,00	0,00	1,00	1,00
max	172,000000	151,80	4,80	60,00	4,00	3,00

находится в основном в центре Москвы, где недвижимость стоит дороже, а остальные ветки частично относятся к окраинам, где квартиры стоят меньше.

Стоит так же посмотреть и на статистические показатели некоторых факторов (табл. 3).

В табл. 3 можно увидеть максимальные и минимальные показатели факторов, а также среднее значение и стандартное отклонение. Можно заметить, что жилая площадь в среднем в 2 раза меньше, чем общая, высота потолков начинается от 2 м и заканчивается почти 5 м, а время от метро в основном примерно равно 10 мин. Все данные достаточно показательны и не противоречат здравому смыслу.

Выводы

В ходе исследования были получены следующие результаты.

1. Рассмотрен принцип работы разработанного ранее парсера.

2. Получены количественные характеристики собранной информации по квартирам в разных городах.

3. Рассмотрены факторы, полученные после предобработки собранных парсером данных.

4. Рассмотрена цена в зависимости от некоторых из факторов, а также статистические показатели числовых данных. Полученные результаты не противоречат здравому смыслу.

Библиографический список

1. *Уилан Ч.* Голая статистика. 2-е изд. Москва: Манн, Иванов и Фербер, 2017. 341 с.

2. *Стерник Г. М., Стерник С. Г.* Анализ рынка недвижимости для профессионалов. М., 2009. 605 с.

3. *Асаул А. Н., Князь И. П., Коротаева Ю. В.* Теория и практика принятия решений по выходу организаций из кризиса. СПб.: АНО «ИПЭВ», 2007. 224 с.

УДК 004.85 **DOI: 10.31799/978-5-8088-1701-2-2022-2-28-33**

В. В. Боженко*
ассистент
В. К. Клюканов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ПРИМЕНЕНИЕ АЛГОРИТМОВ МАШИННОГО ОБУЧЕНИЯ В ЗАДАЧАХ КЛАССИФИКАЦИИ И КЛАСТЕРИЗАЦИИ

Произведен обзор алгоритмов машинного обучения, применяемых в задачах классификации и кластеризации, для каждой задачи представлены метрики качества для оценки моделей, а также описаны результаты программной реализации некоторых алгоритмов на разных наборах данных и выполнено сравнение эффективности алгоритмов.
Ключевые слова: *машинное обучение, анализ данных, классификация, кластеризация, метрики качества.*

V. V. Bozhenko*
Assistant
V. K. Klukanov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

APPLICATION OF MACHINE LEARNING ALGORITHMS FOR CLASSIFICATION AND CLUSTERING

The research presents different machine learning algorithms for binary and multiclass classification and clustering, describes metrics to evaluate model to take the right decisions. The result is creation models for different datasets and comparing metrics to choose the best algorithm.
Keywords: *machine learning, data analysis, classification, clustering, metrics.*

В современном мире имеется колоссальный объем данных, который обычный человек не имеет возможности обработать и проанализировать, поэтому используется машинное обучение (*machine learning*) для анализа, обнаружения взаимосвязей и закономерностей. Машинное обучение – это отдельный класс методов искусственного интеллекта, который отличается от классического программирования тем, что правила обработки данных задаются не человеком, а машиной, которая может на основе входных данных и ответов находить статистическую структуру и строить правило обработки [1]. С помощью разных методов машинного обучения можно решить следующие задачи: прогнозирование, классификация, кластеризация.

Машинное обучение является перспективной областью искусственного интеллекта и уже сейчас используется повсеместно, например в задачах распознавания образов, предсказаниях смертности и заболеваемости в медицине, анализе предпочтений пользователей для формирования рекламы (маркетинг), обеспечения безопасности, создания финансовых прогнозов и т. д. Таким образом, применение алгоритмов машинного обучения необходимо для автоматизации процесса принятия решений в различных сферах деятельности.

Существуют разные методологии машинного обучения, обычно выделяют контролируемое обучение (*supervised learning*), или обучение с учителем, и неконтролируемое (*unsupervised learning*), или обучение без учителя [2].

В обучении с учителем предполагается, что имеется полный набор размеченных данных для обучения модели. Это означает, что каждому образцу изначально соответствует правильный ответ, который алгоритм должен получить в результате работы. В основном такое обучение применяют в прогнозировании и классификации.

Постановка задачи классификации: каждый объект характеризуется парой $<X, Y>$, где $X = x_1, x_2,..., x_m$ – набор признаков (*features*), m – количество признаков, Y – метка класса. При этом

множество меток классов в наборе данных конечно, а между X и Y существует целевая зависимость a: $X \to Y$, которую необходимо найти с помощью алгоритма. Если в задаче всего два класса – это бинарная классификация, а если классов больше – мультиклассификация [3].

Часто в данных нет явных меток на принадлежность к классу, поэтому используется обучение без учителя, например в рекомендательных системах для анализа поведения клиентов и сегментации их на группы. Неконтролируемое обучение применяется для кластеризации, в которой целью модели является нахождение в данных зависимости путем извлечения нужных признаков и их анализа, тем самым разделяя данные на схожие категории. Алгоритм кластеризации – это функция a: $X \to Y$, которая любому объекту $x \in X$ ставит в соответствие метку кластера $y \in Y$ [4].

Перед созданием модели необходимо провести предварительную обработку, потому что ошибки в данных могут привести к неправильным заключениям, а в случае сильно несбалансированных данных метрики могут быть бесполезными, так как такие данные приведут к стремлению модели в одну сторону. Также важно отобрать нужные признаки, которые необходимо использовать, при этом чем меньше признаков, тем проще обучение, но одного признака нередко недостаточно для достоверной оценки. При условии наличия хороших данных выбор алгоритма машинного обучения также будет влиять на точность итоговой модели.

Алгоритмы классификации

Логистическая регрессия – метод, в котором используется вероятность принадлежности к определенному классу, а чтобы предсказать метку, необходимо установить порог вероятности для положительного результата.

Метод k-ближайших соседей (*kNN*) заключается в том, чтобы определить точку в определенный класс, используя k ближайших точек данных.

Дерево решений (*Decision Tree*) представляет собой иерархическую древовидную структуру, состоящую из правил, которые генерируются автоматически в процессе обучения.

Случайный лес (*Random Forest*) – алгоритм, который заключается в использовании ансамбля деревьев решений, каждое из которых может давать не слишком высокое качество классификации, но из-за большого их количества итоговая модель получается эффективной.

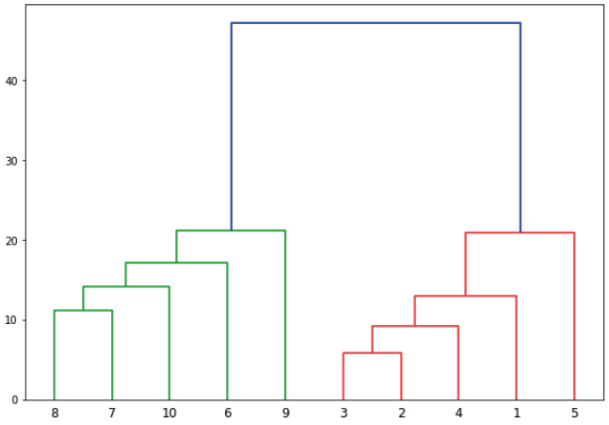


Рис. 1. Пример дендрограммы

Алгоритмы кластеризации

K-means – алгоритм, в котором выбирается k случайных центров кластеризации и близкие объекты объединяются. Чем больше k, тем более мелкие группы создаются с большей степенью детализации.

Агломеративная кластеризация выполняет иерархическую кластеризацию с использованием подхода снизу вверх, кластеры постепенно сливаются в один. Является самым простым и понятным алгоритмом и не имеет фиксированного числа кластеров.

Иерархическая кластеризация – построение иерархии кластеров в виде дендрограммы (рис. 1). Для начала каждому экземпляру данных соответствует свой индивидуальный кластер. Затем два ближайших кластера объединяются в один, это происходит до тех пор, пока не будет выявлен один большой кластер.

Аффинное распространение (*Affinity Propagation*) – метод распространения близости, который получает на вход матрицу схожести между элементами набора данных и возвращает набор меток элементов.

Спектральная кластеризация использует спектр матрицы сходства, которая состоит из количественных оценок схожести относительно каждой пары точек данных.

Оценка качества модели

Для того чтобы оценивать качество модели в контексте решения задачи и выбирать лучшую модель из нескольких, вводится понятие метрики качества, т. е. насколько хорошо происходит процесс предсказания. При этом для кластеризации и классификации меры качества различаются.

Метрики качества классификации

Эффективность модели оценивается с помощью матрицы неточностей (*Confusion Matrix*) – матрица $k \times k$ с комбинациями прогнозируемых и фактических значений, в столбцах – прогноз модели, а в строках – истинная классификация, k – количество классов [5]. Для бинарной классификации матрица состоит из четырех элементов:

True Positive (TP) – количество истинно положительных результатов классификации,

False Positive (FP) – количество ложных положительных результатов, неверный прогноз (ошибка I рода),

False Negative (FN) – количество ложно отрицательных результатов (положительный класс распознан как отрицательный), неверный прогноз (ошибка II рода),

True Negative (TN) – количество истинно отрицательных результатов.

Accuracy (правильность или аккуратность) – это отношение всех верных классификаций к общему числу прогнозов:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}.$$

Данная метрика полезна для хорошо сбалансированных классов, а в случае несбалансированных данных будет неинформативна.

Recall (полнота или чувствительность) – доля правильных положительных результатов, демонстрирует способность алгоритма обнаруживать данный класс:

$$Recall = \frac{TP}{TP + FN}.$$

Precision (точность) – количество верных классификаций к общему количеству положительных элементов. Данная метрика отражает, насколько надежна модель при классификации положительных меток:

$$Precision = \frac{TP}{TP + FP}.$$

Повышение *Precision* снижает *Recall,* и наоборот. В зависимости от задач может быть полезно повышать одну из метрик, например в медицине лучше использовать модели с более высоким значением *Recall*, так как пропущенная болезнь может привести к серьезным последствиям.

*F*1-*score* (*F*1-мера) – среднее гармоническое значение точности и полноты, обозначает, как много сделано правильных прогнозов, и сколько положительных объектов модель не пропустит:

$$F1 - score = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}.$$

ROC-кривая (*Receiver Operating Characteristics curve*) представляет кривую вероятности, которая отражает истинную положительную скорость против ложноположительной скорости при различных пороговых значениях и показывает долю ложноположительных примеров *FPR* от доли истинно положительных примеров (*TPR*). Площадь под кривой или *AUC* (*Area Under the Curve*) также является характеристикой качества классификации: чем больше *AUC*, тем лучше производительность модели.

Для мультиклассификации указанные метрики необходимо обобщить, при этом выделяют микро-, макро- и взвешенный подход для подсчета показателей качества. В основном используется макроусреднение: вычисление средних показателей по всем классам, при этом в качестве положительного берется вычисляемый класс, а остальные полагаются отрицательными [6]. Так, *Macro Average Precision* рассчитывается по следующей формуле:

$$Precision_{macro} = \frac{\sum_{i=1}^{k} Precision_i}{k},$$

где k – количество классов, а *Precision*:

$$Precision_i = \frac{TP_i}{TP_i + FP_i}.$$

Метрики качества кластеризации

Для кластеризации есть внутренние метрики, которые отражают качество кластеризации только по информации в данных, для них не нужно заранее знать истинное разделение на классы [5].

К такой метрике относится коэффициент силуэта, который показывает, насколько похож объект на другие объекты своего кластера. Силуэт объекта i вычисляется по формуле:

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)},$$

где a_i – среднее расстояние между i и другими объектами этого кластера, b_i – наименьшее среднее расстояние между образцом i и другими объектами во всех других кластерах.

Силуэт выборки – средняя величина силуэта объектов данной выборки, силуэт показывает, насколько хорошо была выполнена кластеризация, данный показатель находится в диапазоне

от –1 до +1, при этом оценка около 0 указывает на перекрывающиеся кластеры.

Внешние метрики основаны на сравнении результата кластеризации с априори известными разделениями на классы. Далее приведены некоторые внешние метрики [5, 7].

Метрика *ARI* (*Adjusted Rand Index*) измеряет сходства между полученными метками и базовыми метками, которые были установлены заранее, игнорируя перестановки:

$$ARI = \frac{RI - E[RI]}{\max(RI) - E[RI]},$$

где *RI* – индекс Рэнда (*Rand Index*), *E[RI]* – ожидаемое значение *RI*.

Метрика *AMI* (*Adjusted Mutual Information*) – оценка на основе взаимной информации, которая измеряет согласованность двух назначений, вычисляется с использованием функции энтропии:

$$AMI = \frac{MI - E[MI]}{\text{mean}(H(U), H(V)) - E[MI])},$$

где *MI* – взаимная информация между *V* и *U*, *E[MI]* – ожидаемое значение взаимной информации, *H(U)* – величина неопределенности для набора разбиения *U*, *H(V)* – энтропия для набора разбиения *V*.

Индекс Фаулкса – Мэллоуса FMI для определения сходства между кластерами:

$$FMI = \frac{TP}{\sqrt{(TP + FP)(TP + FN)}}.$$

Homogenity (однородность) описывает, насколько каждый кластер состоит из объектов одного класса, для ее вычисления необходимо определить условную энтропию разбиения и энтропию класса:

$$Homogenity = 1 - \frac{H(C \mid K)}{H(C)},$$

где *H(C|K)* – условная энтропия разбиения, *H(C)* – энтропия класса, *C* – истинное разбиение на классы, *K* – результат кластеризации.

Completeness (полнота) показывает, насколько все объекты одного класса относятся к одному кластеру:

$$Completeness = 1 - \frac{H(K \mid C)}{H(K)}.$$

V-мера – гармоническое среднее полноты и однородности:

$$V = \frac{2 \cdot Homogenity \cdot Completeness}{Homogenity + Completeness}.$$

Реализация алгоритмов классификации

В данной работе были реализованы следующие алгоритмы классификации: метод *k*-ближайших соседей (*kNN*), логистическая регрессия (*LR*), дерево решений (*DT*), случайный лес (*RF*), а также проведена оценка качества этих методов по описанным метрикам. Так как перспективной задачей является машинное обучение в медицине, были выбраны задачи, связанные с постановкой диагноза, а именно определение наличия диабета (первый набор данных) и заболеваний печени (второй набор данных). При этом наборы для анализа и обучения взяты из открытых источников, а для реализации методов использовались язык программирования *Python* и библиотека *scikit-learn*.

Если применять алгоритмы машинного обучения без предварительной обработки, то точность модели будет низкой, поэтому данные сначала были проанализированы, удалены пустые значения, выбраны наиболее значимые признаки для модели. Выборка разделена на тестовую (*test*) и тренировочную (*train*) в соотношении 75 на 25, при этом классы в этих выборках представлены в одинаковой пропорции (стратификация). Также была проведена перекрестная проверка (*K-Fold Cross Validation*, $k = 10$) для получения наиболее достоверной оценки модели. Для метода *kNN* было выбрано оптимальное количество ближайших соседей, а для *RF* определена оптимальная глубина деревьев.

Таблица 1

Метрики классификации для разных алгоритмов

Первый набор данных					Второй набор данных				
Метрика	Название алгоритма				Метрика	Название алгоритма			
	kNN	LR	DT	RF		kNN	LR	DT	RF
Accuracy	0.69	0.76	0.68	0.75	Accuracy	0.71	0.72	0.67	0.70
Precision	0.70	0.74	0.60	0.70	Precision	0.77	0.73	0.77	0.73
Recall	0.49	0.61	0.62	0.59	Recall	0.97	0.92	0.74	0.90
F1 score	0.57	0.67	0.61	0.64	F1 score	0.84	0.81	0.75	0.82

В табл. 1 приведены результаты расчета основных метрик, а на рис. 2, 3 отображены *ROC*-кривые и *AUC* для исследуемых наборов данных для всех реализованных алгоритмов.

Из итоговых показателей видно, что наиболее стабильные результаты дают методы *LR* и *RF*, но и другие алгоритмы также обеспечивают хорошие результаты классификации. Во многом выбор метода зависит от исходного набора данных, при этом невозможно сразу сказать, какой метод будет лучше, обычно используется сравнение сразу нескольких алгоритмов и выявление наиболее эффективного для конкретной задачи исследователя.

Кроме того, не слишком результативно оценивать алгоритм по одной метрике. Например, показатели *AUC* для первого набора данных выше, чем *AUC* для второго, но такие показатели, как *Precision* и *Recall*, для второго набора данных лучше, что гарантирует меньшее количество ошибок

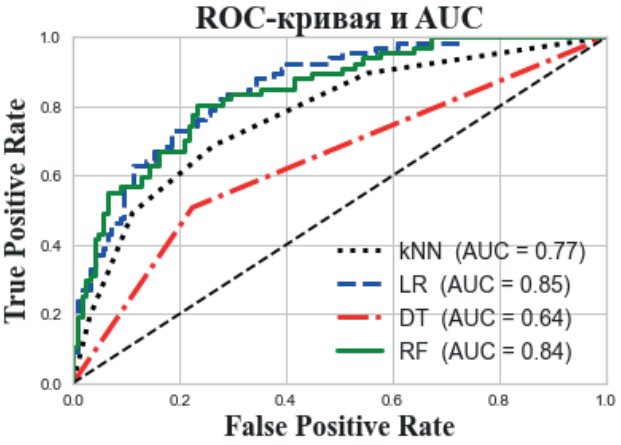


Рис. 2. Оценка качества классификации с помощью ROC-кривых для первого набора данных

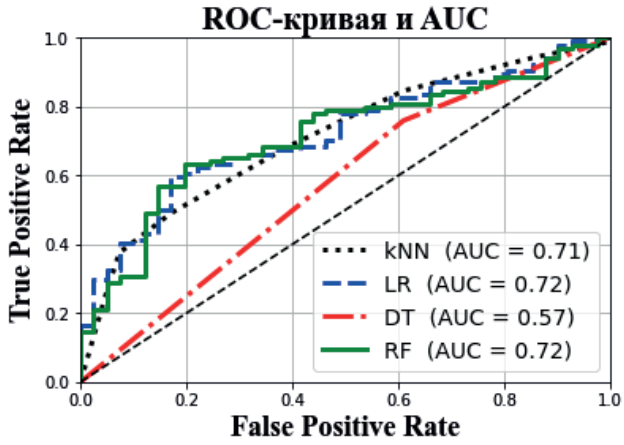


Рис. 3. Оценка качества классификации с помощью ROC-кривых для второго набора данных

классификации. Также видно, что для модели, предсказывающей диабет, *Accuracy* и *AUC* выше для методов *LR* и *RF*, а *Precision* для *LR* и *kNN*. Более того, при оценке *ROC*-кривых часто необходимо задать порог ошибки I или II рода, т. е. выбрать интересующий участок на графике, и уже на его основе делать выводы о том, какая модель лучше. Если для второго набора данных установить порог вероятности ошибки I рода в 0.1, то лучше будут алгоритмы *kNN* и *LR*, а не *RF*, как в общем случае.

Реализация алгоритмов кластеризации

Также были реализованы следующие алгоритмы кластеризации для набора данных по определению заболеваний позвоночника: метод *k-means*, агломеративная кластеризация (*AC*), спектральная кластеризация (*SPC*), Аффинное распространение (*AP*). Исходный набор данных уже был размечен и разделен на три класса: спондилолистез (смещение позвонка), грыжа и здоровые пациенты. Безусловно, в самих алгоритмах кластеризации класс не используется, он необходим только для вычисления внешних метрик. Для построения модели были взяты признаки: наклон таза, угол поясничного лордоза и степень спондилолистеза, результат представлен в табл. 2. Наиболее качественные характеристики получены для *AC* и *SPC*, а метод *AP* дает низкие результаты кластеризации. В целом качество полученной модели невысокое в связи с тем, что кластеры расположены близко друг к другу и многие значения пересекаются.

Кроме того, в ходе анализа было выявлено, что более высокий коэффициент силуэта получается для двух кластеров, а не для трех, как в исходном наборе данных, т. е. для улучшения качества модели можно объединить класс спондилолистез и грыжа в один аномальный класс с заболеванием.

Таблица 2

Метрики кластеризации для разных алгоритмов

Метрика	Название алгоритма			
	k-means	AC	SPC	AP
Коэффицент силуэта	0.60	0.59	0.55	0.41
AMI	0.38	0.71	0.61	0.38
ARI	0.27	0.69	0.50	0.19
FMI	0.59	0.81	0.69	0.41
Homogenity	0.32	0.60	0.59	0.60
Completeness	0.46	0.82	0.60	0.30
V-measure	0.37	0.69	0.59	0.40

Заключение

Таким образом, были рассмотрены некоторые алгоритмы классификации и кластеризации, произведено их сравнение по различным метрикам качества, а также изложены способы улучшения качества предсказательной модели. В результате можно сказать, что любые методы могут быть полезны и эффективны, при этом выбор алгоритма и качество модели зависят от предметной области, исходного набора данных и задач, которые ставит перед собой исследователь, а также от уровня его компетентности. Любое исследование может стать бесполезным при отсутствии навыков анализа данных, понимания метрик качества и знаний правильной интерпретации результатов применения алгоритмов машинного обучения.

Библиографический список

1. *Шолле Ф.* Глубокое обучение на Python. СПб.: Питер, 2018. 400 с.

2. *Пател А.* Прикладное машинное обучение без учителя с использованием Python. СПб.: Диалектика, 2020. 432 с.

3. *Harrington P.* Machine Learning in Action. NY: Manning Publications, 2012. 354 p.

4. *Shalev-Shwartz S.* Understanding machine learning From Theory to Algorithms. NY: Cambridge Univ. Press, 2014. 398 p.

5. *Han J.* Data Mining Concepts and Techniques. MA: Morgan Kaufmann, 2012. 703 p.

6. *Grandini M.* Metrics for multi-class classification: an Overview. URL: https://www.arxiv-vanity.com/papers/2008.05756/ (дата обращения: 15.11.2021).

7. *Rosenberg A., Hirschberg J.* V-Measure: A conditional entropy-based external cluster evaluation measure. URL: https://aclanthology.org/D07-1043 (дата обращения: 15.11.2021).

УДК 004.89 **DOI: 10.31799/978-5-8088-1701-2-2022-2-34-39**

***В. В. Боженко** **
ассистент
В. К. Клюканов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАЗРАБОТКА ПРОГРАММНОГО СРЕДСТВА ДЛЯ АНАЛИЗА МНОГОМЕРНЫХ ДАННЫХ И ПРИМЕНЕНИЯ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

Разработано программное средство с графическим интерфейсом для анализа и визуализации многомерных данных, применения методов машинного обучения для задач прогнозирования, классификации и кластеризации. Программное средство позволяет провести анализ данных, построить модели прогнозирования по различным алгоритмам, оценить их качество. Также приведены примеры результата анализа данных.
***Ключевые слова:** машинное обучение, анализ данных, программное средство, регрессия, классификация, кластеризация.*

V. V. Bozhenko*
Assistant
V. K. Klukanov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

DEVELOPMENT OF A MULTIDIMENSIONAL DATA ANALYSIS AND MACHINE LEARNING ALGORITHMS SOFTWARE TOOL

Special software tool with a graphical interface has been developed for analyzing and visualizing data, applying machine learning algorithms for regression, classification and clustering. The software allows to analyze data and evaluate a predictive model built on the dataset using different metrics. Also examples of data analysis are presented.
***Keywords:** machine learning, data analysis, software tool, regression, classification, clustering.*

В последнее время машинное обучение активно применяется во многих сферах деятельности: в научных дисциплинах, биоинформатике, медицине, продажах, маркетинге, безопасности [1]. При этом затруднительно использовать алгоритмы машинного обучения без применения информационных технологий и минимальных знаний программирования. Целесообразна разработка программного средства с графическим интерфейсом пользователя, которое будет осуществлять анализ данных, позволять строить модели прогнозирования и оценивать их качество, не обращаясь напрямую к коду.

К разрабатываемому средству выдвигались следующие требования: возможность загрузки данных или создания собственного набора данных, возможность проводить предварительный анализ, применять методы машинного обучения в задачах регрессии, классификации и кластеризации, оценить качество созданных моделей (вычисление метрик), сохранить результаты. Кроме того, программное средство должно иметь понятный интерфейс, а архитектура приложения должна обладать гибкостью и расширяемостью.

В качестве языка программирования для разработки был выбран Python, а графический интерфейс создавался с помощью библиотеки PyQt5. Python имеет множество средств для машинного обучения, например библиотеку Scikit-Learn, которая включает алгоритмы обучения с учителем и без учителя, метрики качества, позволяет выполнять кросс-валидацию, анализ и выбор признаков и т. д. А библиотека PyQt5 содержит все необходимые элементы для создания удобного интерфейса: кнопки, вкладки, поля для ввода и вывода данных [2, 3]. В результате было разработано приложение с графическим интерфейсом, которое позволяет проводить анализ данных.

Структура приложения

Приложение состоит из строкового меню «Файл», «Помощь» и вкладок «Таблица», «Анализ данных», «Регрессия», «Классификация» и «Кластеризация», каждая из которых имеет определенную структуру и разные возможности (рис. 1). Для всех кнопок имеются подсказки, которые помогают пользователю ориентироваться в интерфейсе.

Для начала работы пользователь может либо загрузить имеющийся у него файл с данными формата .csv, либо ввести собственные значения с помощью приложения. Чтобы загрузить или сохранить имеющиеся сведения необходимо обратиться к вкладке «Файл», при этом все данные отображаются в виде таблицы. На рис. 1 представлена вкладка «Таблица» разработанного приложения, в качестве примера загружен тестовый набор данных. На этой вкладке содержатся три категории: «Изменение данных» для добавления и удаления строк, столбцов, «Изменить название» для изменения названия столбца и «Работа с таблицей». Для всех функций сделаны интуитивно понятные кнопки. Если пользователь нажимает неверную кнопку, появляется подсказка, и никаких изменений с данными не происходит. Для изменения названия столбца необходимо выбрать в выпадающем списке (элемент ComboBox) интересующий столбец и нажать кнопку «Изменить» для подтверждения действия. При необходимости можно изменить имеющиеся в таблице данные, введя их в соответствующую ячейку (они поддерживают ввод данных). Для удаления пропусков и значений NaN используется кнопка «Убрать пропуски», а кнопка «Очистить данные» отвечает за очищение значений из таблицы и приведение ее к стандартному виду. Чтобы сохранить изменения сделана кнопка «Сохранить данные».

Далее происходит предварительный анализ данных загруженного датасета. На данном этапе есть возможность оценить статистическую зависимость между признаками с помощью корреляционной матрицы, которая отражает коэффициенты корреляции, а также построить парные графики – матрицу диаграмм рассеяния, чтобы выявить закономерности и аномалии для последующего анализа. Парные графики позволяют увидеть отношения между двумя переменными, а также распределение каждой переменной в отдельности, так как такой вид графика позволяет разделить данные по категориальной переменной на классы по цвету и сделать график более информативным [2]. Для построения графика в приложении необходимо выбрать нужные столбцы слева в объекте ListWidget, в котором отражены все столбцы загруженной таблицы, затем выбрать класс, по которому будут разбиты данные и нажать на кнопку «Построить графики» или «Корреляционный анализ» в зависимости от цели пользователя. На рис. 2 представлен пример матрицы диаграмм рассеяния для трех признаков, в качестве класса выбран столбец class, который отвечает за наличие заболевания у пациентов и принимает два значения: Normal и Abnomal. Если необходимо построить графики или корреляционную матрицу по другим данным, то нужно снять выделенные столбцы и выбрать новые или воспользоваться кнопками «Очистить» и «Выбрать все». Каждый результат можно сохранить в виде изображения в формате

Рис. 1. Интерфейс разработанного приложения

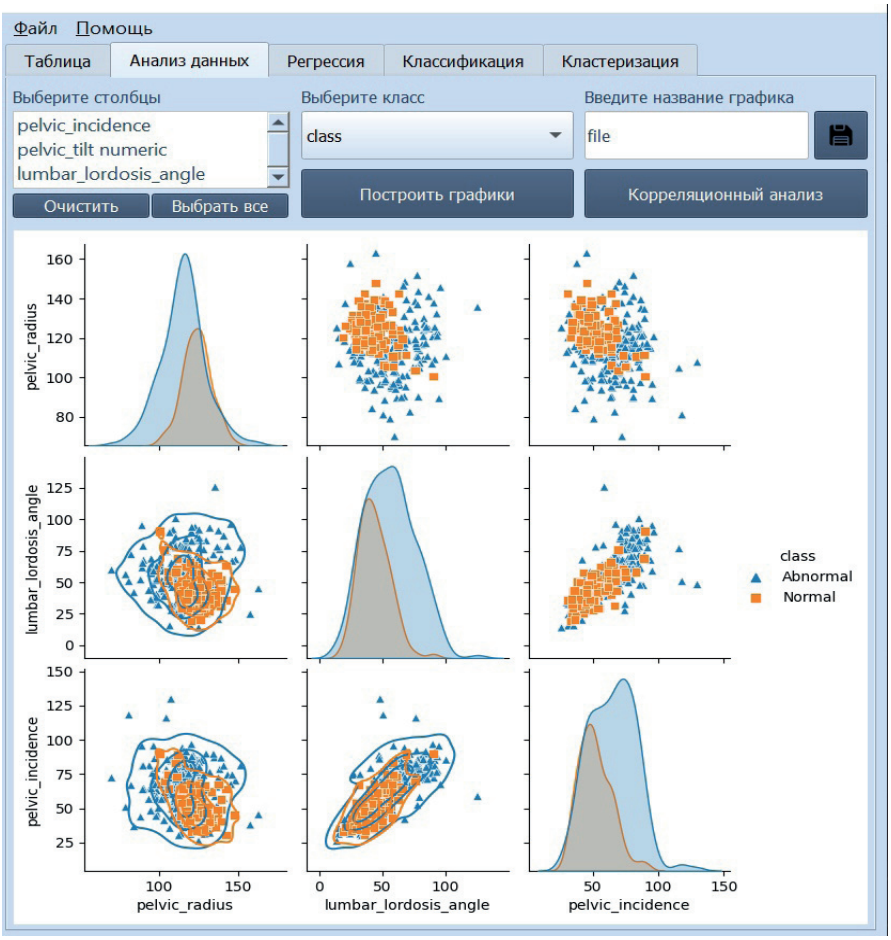


Рис. 2. Пример парных графиков

*.png, файл по умолчанию будет иметь название file.png, но его можно поменять с помощью поля ввода «Введите название графика».

После загрузки набора данных и предварительного анализа выбирается задача: регрессия, классификация, кластеризация. При этом каждая задача имеет свои особенности и параметры, которые необходимы для применения алгоритма машинного обучения. Результат – обучение модели и метрики качества для оценки эффективности.

В регрессии необходимо подобрать такие параметры a_k и степень полинома p для функции вида $Y = \sum_{k=0}^{p} a_k \cdot x^k$, чтобы полином проходил через множество исходных точек наилучшим образом [1]. Линейная регрессия ($p = 1$) отличается простотой и распространенностью применения, поэтому в созданном приложении вынесена в отдельный пункт. При построении полиномиальной регрессии степень полинома выбирается автоматически, чтобы обеспечить минимальную ошибку.

Для использования регрессии необходимо выбрать один или несколько признаков X (feature) и отклик Y (response) в списке «Выберите X и Y», и с помощью кнопок «Добавить X» или «Добавить Y» отнести столбец к выбранной группе. Вместе с тем все столбцы должны содержать числовые значения. Несмотря на то что для создания модели можно выбрать несколько признаков, при построении графика на плоскости используется только первый выбранный пользователем признак. В регрессии реализованы два метода построения модели: «Линейная» и «Полиноминальная». В примере на рис. 3 отражена зависимость между столбцами Total_Bilirubin и Direct_Bilirubin с помощью линейной регрессии. Также справа пользователю доступны метрики, по которым он может оценить качество прогноза. Предусмотрена возможность тестирования, т. е. можно ввести собственные значения признака, и модель автоматически покажет отклик, как показано на рис. 3.

Вкладка «Классификация» также содержит поля для выбора признаков X и отклика (рис. 4),

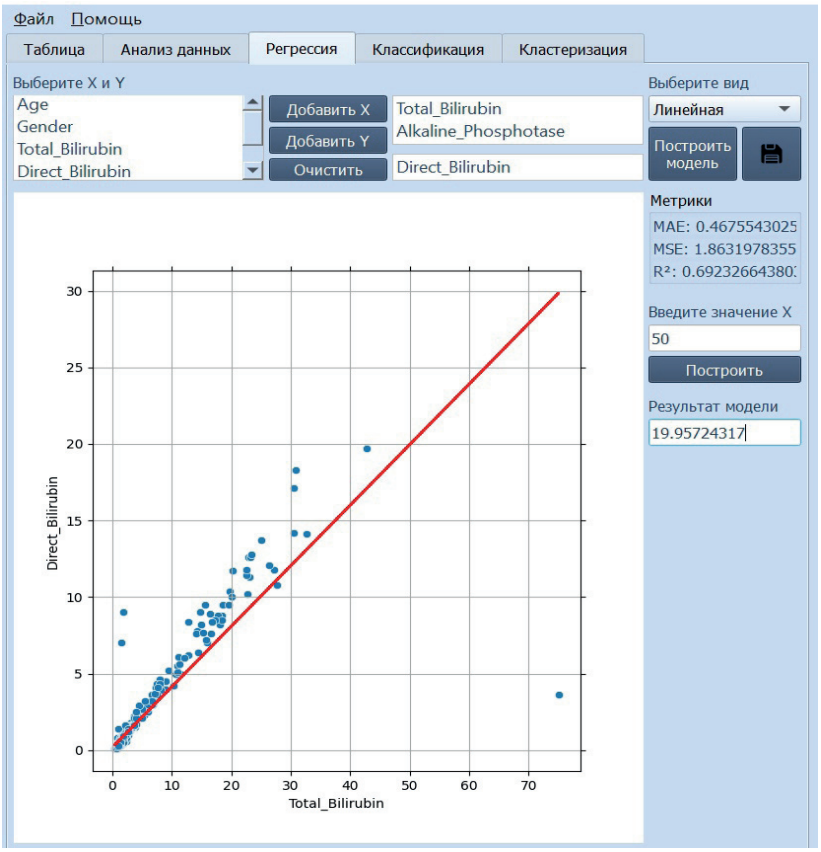


Рис. 3. Пример регрессии

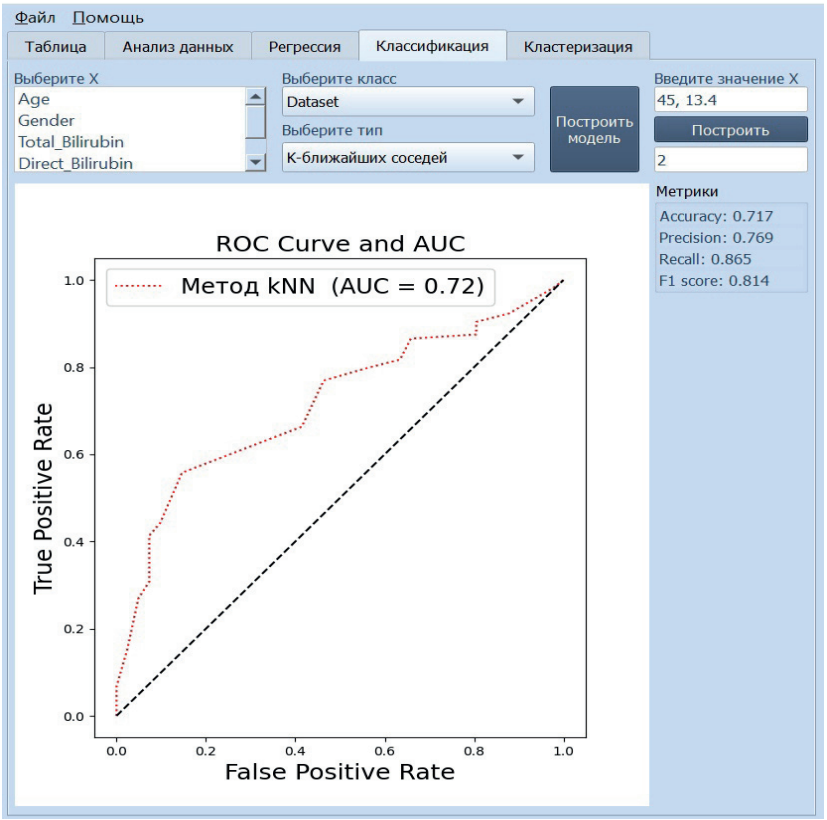


Рис. 4. Пример классификации

только в данном случае отклик является качественным параметром, т. е. классом, который модель должна предсказать. Если будет выбран столбец неподходящего типа, программа выдаст сообщение об ошибке. После выбора корректных столбцов для модели выбирается алгоритм машинного обучения из списка: k-ближайших соседей, логистическая регрессия, дерево решений, случайный лес. На рис. 4 представлен результат использования алгоритма k-ближайших соседей для тестового набора данных для классификации по столбцу Dataset, в качестве признаков использовались столбцы Age, Total_Bilirubin. В результате пользователю отображаются метрики (Accuracy, Precision, Recall, F1-score) и ROC-кривая для оценки качества классификации [2, 4].

Кластеризация отличается от классификации тем, что выбор класса не является обязательным для обучения модели, если в таблице нет столбца с явной меткой класса, модель все равно будет построена и выведена метрика коэффициент силуэта и дендрограмма. Для данной задачи можно использовать следующие алгоритмы машинного обучения: k-средних, агломеративная кластеризация, спектральная кластеризация, аффинное распространение. Если в наборе данных имеется столбец с классом, пользователь может выбрать его, чтобы рассчитать дополнительные метрики качества кластеризации: ARI, AMI, FMI, Homogenity, Completeness, V-measure [2, 5]. На рис. 5 представлена вкладка «Кластеризация» для создания модели по обнаружению пациентов с заболеванием и без него.

Таким образом, разработанное средство соответствует всем предъявляемым требованиям и позволяет существенно упростить анализ данных. На примерах было продемонстрировано, что выполнение анализа данных происходит в удобном формате, а интерфейс интуитивно понятен. При этом возможно использование машинного обучения в различных сферах, так как приложение универсально и рабо-

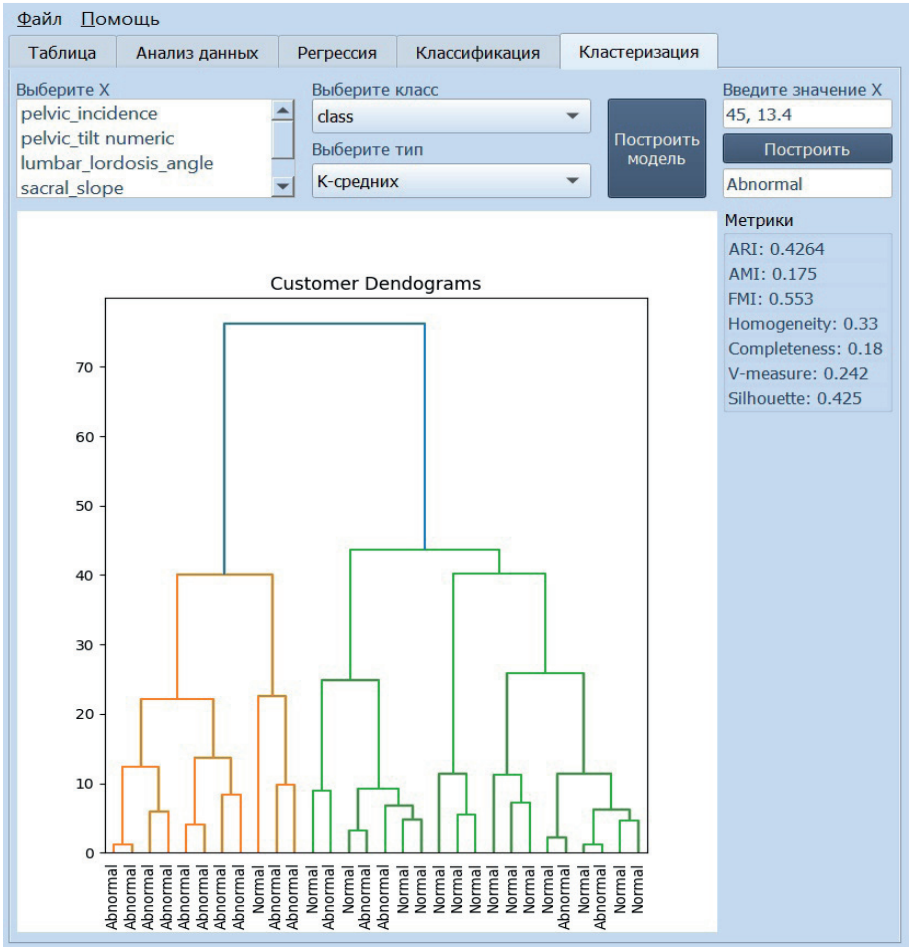


Рис. 5. Пример кластеризации

тает с набором данных, состоящим из столбцов и строк. Безусловно, пользователю необходимо иметь представление о методах машинного обучения, чтобы грамотно использовать функционал и выбирать нужные данные, но применение графического приложения помогает избежать множества ошибок, а также уменьшает время проведения исследований и является доступным вариантом для людей, которые не имеют навыков программирования (например, врачи, агенты по недвижимости и др.). При необходимости возможно расширение разработанного приложения и добавление новых алгоритмов или средств визуализации данных.

Библиографический список

1. *Шалев-Шварц Ш.* Идеи машинного обучения: от теории к алгоритмам. М.: ДМК Пресс, 2019. 436 с.

2. *Müller A.* Introduction to Machine Learning with Python. CA: O'Reilly, 2017. 378 p.

3. *Прохоренок Н. А.* Python 3 и PyQt 5. Разработка приложений. СПб.: БХВ-Петербург, 2018. 832 с.

4. *Grandini M.* Metrics for multi-class classification: an Overview. URL: https://www.arxiv-vanity.com/papers/2008.05756/ (дата обращения: 15.11.2021).

5. *Rosenberg A., Hirschberg J.* V-Measure: A conditional entropy-based external cluster evaluation measure. URL: https://aclanthology.org/D07-1043 (дата обращения: 15.11.2021).

УДК 621.396 **DOI: 10.31799/978-5-8088-1701-2-2022-2-40-43**

Н. Н. Григорьева*
старший преподаватель
В. И. Исаков*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОСОБЕННОСТИ МОДЕЛИРОВАНИЯ ЭХОСИГНАЛОВ МОРСКОЙ ПОВЕРХНОСТИ

При проектировании бортовых устройств поиска и обнаружения физических объектов на фоне отражений от морской поверхности широко используются методы имитационного моделирования. В работе приводятся математические модели и алгоритмы моделирования эхосигналов морской поверхности, учитывающие ракурс морских волн при работе бортовой РЛС, на основе моделей синтезируются алгоритмы моделирования эхосигналов, используемые при проектировании бортовых систем.
Ключевые слова: *обнаружение, корреляционная функция, зондирующий импульс, морская поверхность, ракурс, алгоритм моделирования.*

N. N. Grigorieva*
Senior Lecturer
V. I. Isakov*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

FEATURES OF MODELING OF ECHOES OF THE SEA SURFACE

When designing on-board devices for searching and detecting physical objects against the background of reflections from the sea surface, simulation modeling methods are widely used. The paper presents mathematical models and algorithms for modeling echoes of the sea surface, taking into account the angle of sea waves during the operation of the onboard radar, based on the models, algorithms for modeling echo signals used in the design of onboard systems are synthesized.
Keywords: *detection, correlation function, probing pulse, sea surface, angle, modeling algorithm.*

Введение

Поиск и обнаружение эхосигналов на фоне отражений от морской поверхности в современных бортовых РЛС осуществляется обнаружителями, накапливающими отраженные зондирующие сигналы в течение некоторого времени, зависящего от используемого в бортовой аппаратуре режима работы РЛС. Обнаружение эхосигналов надводных морских объектов происходит не только на фоне собственных шумов приемных устройств, но и на фоне интенсивных отражений от морской поверхности.

Эхосигналы морской поверхности имеют достаточно сильную корреляционную связь, описываемую корреляционными временными функциями, которую необходимо учитывать как при расчете ложных тревог, так и при расчете характеристик обнаружения. Эта корреляционная связь эхосигналов морской поверхности зависит и от ракурса морских волн, что необходимо учесть как в математических моделях, используемых для синтеза алгоритмов обработки сигналов, так и при синтезе алгоритмов моделирования флюктуаций этих мешающих воздействий.

В работе предложены простые модели корреляционных зависимостей эхосигналов морской поверхности, на основе которых синтезированы алгоритмы имитационного моделирования сигналов. В математических моделях и алгоритмах учитываются статистические различия отражений при разных ракурсах морских волн, а также требования быстродействия имитационных алгоритмов.

Математическая модель эхосигналов морской поверхности

Наиболее распространенная математическая модель локационных сигналов, отраженных от морской поверхности, – модель с лога-

рифмически нормальным распределением. Эта модель достаточно точно описывает флюктуации огибающей эхосигналов морской поверхности, для нее разработаны эффективные алгоритмы моделирования, получены аналитические выражения, связывающие параметры многомерной модели, и имеются достаточно обширные экспериментальные исследования зависимости этих параметров от условий морского волнения, условий его наблюдения и характеристик бортовой аппаратуры, предназначенной для обработки эхосигналов [1].

При исследовании влияния спектрально-корреляционных характеристик на флюктуации эхосигналов морской поверхности достаточно ограничиться двумерным законом распределения огибающей эхосигналов. Ограничиваясь рассмотрением временных флюктуаций эхосигналов морской поверхности, отраженных от одной и той же дорожки дальности, двумерный закон распределения огибающей имеет вид:

$$w(A_i, A_n) = \frac{1}{2\pi\sigma_i\sigma_n A_i A_n\sqrt{1-r_{in}^2}}\exp\left\{-\frac{1}{2\left(1-r_{in}^2\right)}\times\right.$$
$$\left.\times\left[\begin{array}{l}\dfrac{1}{\sigma_i^2}\ln^2\dfrac{A_i}{\bar{A}_i}+\dfrac{1}{\sigma_n^2}\ln^2\dfrac{A_n}{\bar{A}_n}-\\ -2r_{in}\dfrac{1}{\sigma_i}\ln\dfrac{A_i}{\bar{A}_i}\cdot\dfrac{1}{\sigma_n}\ln\dfrac{A_n}{\bar{A}_n}\end{array}\right]\right\}, \quad (1)$$

где $\bar{A}_i$, $\bar{A}_n$, σ_i и σ_n – параметры распределения, связанные со средним значением $\tilde{A}_i$ и дисперсией $\tilde{\sigma}_i^2 = \tilde{D}_i$ выражениями, приведенными в [2].

Коэффициент вариации огибающей $K_i = \tilde{\sigma}_i / \tilde{A}_i$, экспериментальные численные значения K_i приведены во многих источниках, в частности в [3], а $\tilde{A}_i$ выражается через среднюю мощность отражений $\tilde{P}_i$ с учетом площади элемента разрешения по известной формуле радиолокации [4], аналогично для индекса n. Параметр r_{in} равен:

$$r_{in} = \frac{\ln\left(1+K_iK_nR_{in}\right)}{\sqrt{\ln\left(1+K_i^2\right)\ln\left(1+K_n^2\right)}}, \quad (2)$$

где R_{in} – коэффициент корреляции между i-м и n-м отсчетами амплитуд принимаемой пачки отраженных от морской поверхности импульсов.

За время поиска параметры морского волнения практически не изменяются, что позволяет записать выражения (1) и (2) в следующем виде

$$w(A_i, A_n) = \frac{1}{2\delta\acute{o}^2 A_i A_n\sqrt{1-r_{in}^2}}\exp\left\{-\frac{1}{2\acute{o}^2\left(1-r_{in}^2\right)}\times\right.$$
$$\left.\times\left[\ln^2\frac{A_i}{\bar{A}}+\ln^2\frac{A_n}{\bar{A}}-2r_{in}\cdot\ln\frac{A_i}{\bar{A}}\cdot\ln\frac{A_n}{\bar{A}}\right]\right\}, \quad (3)$$

где $\bar{A}_i = \bar{A}_n = \bar{A}$, $\sigma_i = \sigma_n = \sigma$ и соответственно $\tilde{A}_i = \tilde{A}_n = \tilde{A}$ и $\tilde{\sigma}_i^2 = \tilde{\sigma}_n^2 = \tilde{\sigma}^2 = \tilde{D}$, $K_i = \tilde{\sigma}_i / \tilde{A}_i = K_n = \tilde{\sigma}_n / \tilde{A}_n = K$, а r_{in} равен:

$$r_{in} = \frac{\ln\left(1+K_iK_nR_{in}\right)}{\sqrt{\ln\left(1+K_i^2\right)\ln\left(1+K_n^2\right)}} = \frac{\ln\left(1+K^2\ R_{in}\right)}{\ln\left(1+K^2\right)} \quad (4)$$

В выражениях (3) и (4) остается лишь временная зависимость статистических параметров флюктуаций амплитуд, а это позволяет рассматривать процесс отражения от элемента разрешения как стационарный процесс, что очень существенно позволяет повысить эффективность, в частности быстродействие, алгоритмов моделирования флюктуаций эхосигналов.

Корреляционные характеристики эхосигналов морской поверхности

Из выражения (4) получаем зависимость временной корреляционной функции от параметров закона распределения флюктуаций огибающей эхосигналов морской поверхности:

$$R(\tau) = R\left(|i-n|\cdot T_{\text{РЛС}}\right) = R_{|i-n|} = \frac{1}{K^2}\left[\left(1+K^2\right)^{r_{|i-n|}}-1\right] = \frac{1}{K^2}\left[\left(1+K^2\right)^{r(\tau)}-1\right], \quad (5)$$

где $r(\tau) = r\left(|i-n|T_{\text{РЛС}}\right)$ – коэффициент корреляции между i-м и n-м отсчетами логарифмов амплитуд принимаемой пачки импульсов.

В выражении (5) при принятых ограничениях от ракурса ψ морских волн могут зависеть только два параметра – коэффициент вариации $K = K(\psi)$ и корреляционная функция логарифмов амплитуд $r(\tau) = r(\tau/\psi)$. Эти зависимости, полученные эмпирическим путем, приво-

дятся во многих источниках, в частности они использованы в работах [5, 6].

Коэффициент вариации K зависит в основном от угла визирования элемента разрешения в вертикальной плоскости θ (в градусах), ракурса волны ψ (в радианах) – *угла между направлением движения морских волн и направлением диаграммы направленности антенной системы бортовой РЛС*, волнения моря W (в баллах):

$$K = K(\theta,\psi,W) = 0{,}52 - \\ -\left[0{,}17 + (W - 5)\cdot 10^{-2}\right]\left(1 - \frac{\psi}{\pi}\right)\sin(9\theta). \quad (6)$$

Выражение (6) справедливо при $0° \le \theta \le 10°$, $0 \le \psi \le \pi$, $0 \le W_i \le 6$ баллов. В дальней зоне, когда θ малы, можно считать K постоянным и равным $K_A = \sqrt{(4-\pi)/\pi} \approx 0{,}52$ – коэффициенту вариации распределения Релея.

Корреляционные функции межпериодных флюктуаций – временные корреляционные функции многообразны, поэтому в источниках приводится эмпирическая зависимость лишь для длительности функции $R(\tau)$ на уровне 0,5:

$$\tau_{0,5} = \frac{4 + 0{,}3\cdot\theta/W + \cos\psi}{1 + VW/300}, \quad (7)$$

где θ и ψ – в градусах, W – в баллах, V – скорость перемещения «пятна» засветки на морской поверхности в м/с, $\tau_{0,5}$ – в миллисекундах. Этой формулой можно воспользоваться для приблизительного определения параметров аппроксимации $R(\tau)$ при отсутствии экспериментальных $R(\tau)$ для заданных условий наблюдения [7].

Алгоритмы моделирования эхосигналов морской поверхности

Алгоритмы моделирования эхосигналов морской поверхности, учитывающие ракурс морских волн, проще всего синтезируются и реализуются в виде нелинейных цифровых формирующих фильтров. При этом ограничимся простейшими, но достаточными для практического использования фильтрами второго порядка, такими, с помощью которых реализуются дифференцируемые марковские процессы.

Для реализации используем два типа фильтров – один с корреляционной функцией, представляющей весовую сумму двух экспонент, а другой – с корреляционной функцией, имеющей колебательный характер.

Первый тип фильтра

Корреляционная функция логарифмов амплитуд для этого типа фильтра определяется как

$$r(kT) = \frac{\mu_2}{\mu_2 - \mu_1}\cdot\exp(-\mu_1 Tk) + \\ + \frac{\mu_1}{\mu_1 - \mu_2}\cdot\exp(-\mu_2 Tk). \quad (8)$$

Параметры этой корреляционной функции определяются по приведенным выражениям, учитывающим ракурс волны. Корреляционная функция флюктуаций огибающей в этом случае определена выражением (5), $T_{\text{РЛС}} = T$.

Коэффициенты разностного уравнения, определяющего алгоритм моделирования, вычисляются по выражениям

$$\begin{cases} a_1 = \exp(-\mu_1 T) + \exp(-\mu_2 T), \\ a_2 = \exp\left(-(\mu_1 + \mu_2)T\right), \\ r_0 = 1 - 2\dfrac{\mu_2 + \mu_1}{\mu_2 - \mu_1}\times \\ \times\left(\exp(-2\mu_1 T) - \exp(-2\mu_2 T)\right) - \\ - \exp\left(-2(\mu_1 + \mu_2)T\right), \\ r_1 = -\left(\exp(-\mu_1 T) + \exp(-\mu_2 T)\right) + \\ + \left(1 + \exp\left(-(\mu_1 + \mu_2)T\right)\right)\times \\ \times\left(\dfrac{\mu_2}{\mu_2 - \mu_1}\cdot\exp(-\mu_1 T) + \right. \\ \left. + \dfrac{\mu_1}{\mu_1 - \mu_2}\cdot\exp(-\mu_2 T)\right), \\ b_0 = 0{,}5\left(\sqrt{r_0 + 2r_1} + \sqrt{r_0 - 2r_1}\right), \\ b_1 = 0{,}5\left(\sqrt{r_0 + 2r_1} - \sqrt{r_0 - 2r_1}\right). \end{cases} \quad (9)$$

Этот тип корреляционной функции соответствует движению вдоль волн – ракурс морских волн близок к нулю.

Второй тип фильтра

Корреляционная функция логарифмов амплитуд для этого типа фильтра определяется как

$$r(kT) = \exp(-\mu Tk)\times \\ \times\left(\cos(\gamma Tk) + \frac{\mu}{\gamma}\cdot\sin(\gamma Tk)\right). \quad (10)$$

Параметры этой корреляционной функции определяются также по приведенным выражениям, учитывающим ракурс волны. Корреляционная функция флюктуаций огибающей

в этом случае определена выражением (5), $T_{\text{РЛС}} = T$.

Коэффициенты разностного уравнения, определяющего алгоритм моделирования, вычисляются по выражениям

$$\begin{cases} a_1 = 2\exp(-\mu T)\cdot\cos(\gamma T), \\ a_2 = \exp(-2\mu T), \\ r_0 = 1 - 2\dfrac{\mu}{\gamma}\cdot\exp(-2\mu T)\cdot\sin(\gamma T) - \exp(-4\mu T), \\ r_1 = -\exp(-\mu T)\cdot(1-\exp(-2\mu T))\times \\ \times\cos(\gamma T) + \dfrac{\mu}{\gamma}\cdot\exp(-\mu T)\times \\ \times(1+\exp(-2\mu T))\cdot\sin(\gamma T), \\ b_0 = 0{,}5\left(\sqrt{r_0+2r_1}+\sqrt{r_0-2r_1}\right), \\ b_1 = 0{,}5\left(\sqrt{r_0+2r_1}-\sqrt{r_0-2r_1}\right). \end{cases} \quad (11)$$

Этот тип корреляционной функции соответствует движению поперек волн – ракурс морских волн близок к девяноста градусам.

Оба типа фильтров реализуются одним и тем же уравнением, коэффициенты которого определены выражениями (9) и (11):

$$U_k = a_1\cdot U_{k-1} + a_2\cdot U_{k-2} + b_0\cdot\eta_k + b_1\cdot\eta_{k-1}, \quad (12)$$

в котором η_k – случайные величины, распределенные по нормальному закону с нулевыми средними и единичными дисперсиями. Последовательность амплитуд формируется путем возведения экспоненты в степень U_k и соответствующим умножением на коэффициенты, учитывающие энергетические характеристики отражений [8]. Здесь эти выражения не приводятся, поскольку для их расчета необходимо задаться характеристиками бортовой РЛС, а это выходит за рамки рассматриваемого вопроса.

Заключение

В работе приведены выражения и алгоритмы моделирования флюктуаций эхосигналов морской поверхности, которые позволяют учесть ракурс направления движения морских волн, что важно при моделировании работы бортовых систем поиска и обнаружения морских объектов, а также для картографических систем, определяющих очертание береговой кромки.

Библиографический список

1. *Иванова М. С., Шепета А. П.* Алгоритмы моделирования информационных потоков систем обработки информации, синтезированные по эмпирическим данным // Обработка, передача и защита информации в компьютерных системах '21: Междунар. науч. конф.: сб. докл. СПб., 2021. С. 20–24.

2. *Блауштейн Н. Ш., Сергеев М. Б., Шепета А. П.* Прикладные аспекты электродинамики. СПб., 2016. 239 с.

3. *Шепета А. П., Подоплёкин Ю. Ф.* Корреляционные функции эхо-сигналов кораблей при перестройке несущей частоты бортовой РЛС // Морская радиоэлектроника. 2021. № 1 (75). С. 50–53.

4. *Исаков В. И., Шепета Д. А.* Плотность распределения мощности огибающей локационных сигналов, отраженных от кромки земля–море // Обработка, передача и защита информации в компьютерных системах '21: Междунар. науч. конф.: сб. докл. СПб., 2021. С. 25–28.

5. *Шепета А. П., Бажин С. А., Давидчук А. Г.* Экспериментальные характеристики эхо-сигналов кораблей, наблюдаемых локаторами бортовых систем обработки информации // Информационно-управляющие системы. 2005. № 2 (15). С. 7–12.

6. *Шепета А. П.* Определение зоны поиска надводного объекта по данным предварительного целеуказания // Информационно-управляющие системы. 2012. № 4 (59). С. 98–99.

7. *Давидчук А. Г., Подоплёкин Ю. Ф., Шепета А. П.* Выбор главной морской цели по эффективной поверхности рассеяния // Морская радиоэлектроника. 2019. № 1 (67). С. 28–31.

8. *Мауритс В. Г., Климова М. А., Иванова М. С.* Математическая модель эхо-сигналов морской поверхности, наблюдаемой бортовыми локаторами летательных аппаратов // Научная сессия ГУАП: сб. докл.: в 3 ч. Ч. 2. 2018. С. 351–357.

УДК 621.396 **DOI: 10.31799/978-5-8088-1701-2-2022-2-44-47**

Н. Н. Григорьева*
старший преподаватель
Д. А. Шепета*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАСЧЕТ ЭНЕРГЕТИЧЕСКИХ ПОТЕРЬ ОБНАРУЖИТЕЛЕЙ БОРТОВЫХ РЛС

Обнаружение физических объектов в современных бортовых РЛС осуществляется при сканировании заданной зоны электромагнитным лучом и накопления информационных эхосигналов в обнаружителях, каждый из которых соответствует своей дорожке дальности. Расчет вероятности правильного обнаружения проводится в предположении, что вся энергия отраженного от искомого объекта накапливается только в одном из обнаружителей. Это условие нарушается, если обнаруживаемый объект находится вблизи границ элементов разрешения бортовой РЛС. В этом случае энергия сигнала делится между соседними обнаружителями, что приводит к энергетическим потерям. В работе производится оценка этих потерь.
Ключевые слова: *зона поиска, закон распределения, элемент разрешения, режим обзора, подстилающая поверхность, вероятность ложной тревоги, вероятность обнаружения.*

N. N. Grigorieva*
Senior Lecturer
D. A. Shepeta *
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

CALCULATION OF ENERGY LOSSES OF ONBOARD RADAR DETECTORS

The detection of physical objects in modern airborne radars is carried out by scanning a given area with an electromagnetic beam and accumulating information echoes in detectors, each of which corresponds to its own range track. Calculation of the probability of correct detection under the assumption that the energy reflected from the target object is accumulated only in one of the detectors. This condition is violated if the detected object is near the boundaries of the resolution elements of the onboard radar. In this case, the signal energy is divided between neighboring detectors, which leads to energy losses. The work evaluates these losses.
Keywords: *search area, distribution law, resolution element, survey mode, underlying surface, false alarm probability, detection probability.*

Введение

В современных бортовых системах поиск и обнаружение физических объектов на фоне собственных шумов и отражений от подстилающих поверхностей осуществляются группой обнаружителей, каждый из которых накапливает и обнаруживает эхосигналы от искомых объектов на соответствующих дорожках дальности [1]. При этом необходимо обеспечить заданный уровень ложных тревог для всей зоны поиска и максимизировать вероятность правильного обнаружения.

При расчете вероятности правильного обнаружения сигнала исходят из того, что физический объект полностью находится в элементе разрешения бортовой РЛС, а отраженный от объекта локационный сигнал полностью попадает в элемент разрешения. В этом случае при обнаружении информационного эхосигнала используется вся энергия отраженного импульса [2].

На практике такой случай соответствует ситуации, когда радиолокационный размер физического объекта много меньше линейных размеров элементов разрешения бортовой РЛС. Однако если для угловых координат подобное условие выполняется в силу ограничений на ширину диаграммы направленности антенной системы и обнаружения объектов в дальней зоне, то по отношению к размерам элементов разрешения по координате дальности оно не выполняется [3]. Тогда сигнал, отраженный от физического объекта, делится между соседними элементами, в каждом из которых возможно его обнаружение в соответствии с той долей энергии, которая приходится на соответствующие элементы. Разделение энергии эхосигнала физического объекта приводит к энергетическим поте-

рям и, соответственно, к уменьшению вероятности правильного обнаружения [4].

В работе оцениваются потери, возникающие от деления энергии информационного сигнала между соседними элементами разрешения по дальности.

Математические модели информационного сигнала

Для оценки возникающих энергетических потерь необходимо определить математические модели информационного и мешающего сигнала. Будем считать, что обнаружение происходит на фоне собственных шумов приемного устройства бортовой РЛС, которые традиционно описываются нормальным законом распределения. Огибающая шума в этом случае распределена по закону Рэлея:

$$f_{\text{ш}}(x) = \frac{x}{\sigma_{\text{ш}}^2} \cdot \exp\left(-\frac{x^2}{2\sigma_{\text{ш}}{}^2}\right), \quad x > 0, \qquad (1)$$

где $\sigma_{\text{ш}}^2$ – средняя мощность шума.

Традиционно используются две математические модели информационного сигнала – модель быстрых флюктуаций (шумоподобный сигнал) и модель стабильного сигнала. Реальные характеристики обнаружения находятся между кривыми обнаружения, рассчитанными по этим двум моделям [5, 6].

Модель быстрых флюктуаций позволяет получить простые аналитические выражения для расчета характеристик обнаружения, а при использовании модели стабильного сигнала характеристики обнаружения выражаются через квадратуры и рассчитываются численно [6]. В нашем случае для оценки потерь удобнее использовать более простую модель шумоподобного сигнала, для которой огибающая также описывается законом Рэлея:

$$f_{\text{ш+с}}(x) = \frac{x}{\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2} \times \exp\left(-\frac{x^2}{2\left(\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2\right)}\right), \quad x > 0, \qquad (2)$$

где $\sigma_{\text{с}}^2$ – средняя мощность информационного сигнала.

В том случае когда информационный эхосигнал делится между двумя элементами разрешения по дальности, в один элемент попадает часть мощности сигнала равная $\gamma\sigma_{\text{с}}^2$, а в другой элемент – $(1-\gamma)\sigma_{\text{с}}^2$. При $\gamma = 0$ или $\gamma = 1$ получаем традиционную модель обнаружения шумоподобного сигнала, здесь же обнаружение происходит с помощью двух обнаружителей.

В качестве обнаружителей рассмотрим наиболее распространенную схему обнаружения эхосигналов с помощью линейки обнаружителей типа «k из n». В этих обнаружителях используются два порога: аналоговый порог компаратора X_0 и цифровой порог накопителя бинарных сигналов k_0. Аналоговый компаратор интегрирует сигнал по квадратурам, формируя его огибающую, которая сравнивается с порогом X_0. При превышении порога на выходе компаратора формируется сигнал «1», в противном случае сигнал «0».

Вероятность превышения порога компаратора шумом $P_{\text{ш}}$ равна

$$P_{\text{ш}} = \int_{X_0}^{+\infty} \frac{x}{\sigma_{\text{ш}}^2} \cdot \exp\left(-\frac{x^2}{2\sigma_{\text{ш}}^2}\right) \cdot dx = \exp\left(-\frac{X_0{}^2}{2\sigma_{\text{ш}}^2}\right), \qquad (3)$$

откуда при заданном $P_{\text{ш}}$ получаем значение порога $X_0 = \sigma_{\text{ш}}\sqrt{-2\ln P_{\text{ш}}}$.

Вероятность превышения порога компаратора аддитивной смесью информационного сигнала и шума $P_{\text{с}}$ равна

$$P_{\text{с}} = \int_{X_0}^{+\infty} \frac{x}{\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2} \cdot \exp\left(-\frac{x^2}{2\left(\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2\right)}\right) \cdot dx = \exp\left(-\frac{X_0{}^2}{2\left(\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2\right)}\right). \qquad (4)$$

Подставляя в выражение (4) значение порога $X_0 = \sigma_{\text{ш}}\sqrt{-2\ln P_{\text{ш}}}$, получаем

$$P_{\text{с}} = \exp\left(-\frac{X_0{}^2}{2\left(\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2\right)}\right) = \exp\left(\frac{\sigma_{\text{ш}}^2 \ln P_{\text{ш}}}{\sigma_{\text{ш}}^2 + \sigma_{\text{с}}^2}\right) = P_{\text{ш}}^{\frac{1}{1+\rho^2}}, \qquad (5)$$

где $\rho^2 = \sigma_{\text{ш}}^2 / \sigma_{\text{с}}^2$ отношение сигнал/шум, при этом если мощность сигнала делится между двумя элементами разрешения, то

$$\rho_{\gamma}^2 = \sigma_{\text{ш}}^2 \Big/ \left(\gamma\sigma_{\text{с}}^2\right) \text{ и } \rho_{1-\gamma}^2 = \sigma_{\text{ш}}^2 \Big/ \left((1-\gamma)\sigma_{\text{с}}^2\right).$$

Характеристики обнаружения сигнала

При расчете характеристик обнаружителя типа «k из n» необходимо задать: размер пачки импульсов n, по которой принимается решение об обнаружении сигнала; порог аналогового компаратора X_0; порог цифрового компаратора k_0.

Размер пачки импульсов n рассчитывается исходя из организации режима поиска объекта в заданной зоне, который зависит от характеристик бортовой аппаратуры и априорных сведений о характеристиках искомого физического объекта. Порог аналогового компаратора зависит от его структуры и от заданной вероятности ложной тревоги $P_{\text{лт}}$, которая определяется исходя из заданной вероятности ложной тревоги в течение всего режима поиска и для всей зоны. Порог цифрового компаратора определяется из максимизации вероятности правильного обнаружения $P_{\text{по}}$ при фиксации вероятности ложной тревоги $P_{\text{лт}}$.

Перечисленные характеристики обнаружителя связаны следующими соотношениями:

$$P_{\text{лт}} = \sum_{k=k_0+1}^{n} C_n^k P_{\text{ш}}^k (1-P_{\text{ш}})^{n-k}, \qquad (6)$$

где $C_n^k = \dfrac{n!}{k!(n-k)!}$, $k_0 = E(\sqrt{1.5\cdot n} + 0.5)$, $E(x)$ – функция Антье – целая часть числа x,

$$P_{\text{по}} = \sum_{k=k_0+1}^{n} C_n^k P_{\text{с}}^k (1-P_{\text{с}})^{n-k}, \qquad (7)$$

$P_{\text{ш}}$ и $P_{\text{с}}$ определены выражениями (3) и (5).

В нашем случае, когда мощность информационного сигнала делится между двумя обнаружителями, вероятность правильного обнаружения вычисляется по формуле

$$P_{\text{по}} = 1 - \left(1 - P_{\text{по}}^{(\gamma)}\right)\cdot\left(1 - P_{\text{по}}^{(1-\gamma)}\right), \qquad (8)$$

где

$$\begin{cases} P_{\text{по}}^{(\gamma)} = \sum\limits_{k=k_0+1}^{n} C_n^k P_{c,\gamma}^k (1-P_{\text{с},\gamma})^{n-k}, \\ P_{\text{по}}^{(1-\gamma)} = \sum\limits_{k=k_0+1}^{n} C_n^k P_{\text{с},1-\gamma}^k (1-P_{\text{с},1-\gamma})^{n-k}, \end{cases} \qquad (9)$$

вероятности правильного обнаружения первым и вторым обнаружителями, а

$$\begin{cases} P_{\text{с}}^{(\gamma)} = P_{\text{ш}}^{\frac{1}{1+\rho_{\text{ä}}^2}} = P_{\text{ш}}^{\frac{1}{1+\rho_\gamma^2=\sigma_{\text{ш}}^2/\left(\gamma\sigma_{\text{с}}^2\right)}}, \\ P_{\text{с}}^{(1-\gamma)} = P_{\text{ш}}^{\frac{1}{1+\rho_{1-\gamma}^2}} = P_{\text{ш}}^{\frac{1}{1+\tilde{n}_{\text{ä}}^2=\sigma_{\text{ш}}^2/\left((1-\gamma)\sigma_{\text{с}}^2\right)}}, \end{cases} \qquad (10)$$

вероятности формирования сигналов «1» на выходе аналогового компаратора.

При $\gamma = 1$ или $\gamma = 0$ выражение (8) записывается в виде

$$\begin{aligned} P_{\text{по}} &= 1 - \left(1 - P_{\text{по}}^{(1)}\right)\cdot\left(1 - P_{\text{лт}}\right) = \\ &= 1 - \left(1 - P_{\text{по}}\right)\cdot\left(1 - P_{\text{лт}}\right) = \\ &= P_{\text{по}} + P_{\text{лт}}\cdot\left(1 - P_{\text{по}}\right) \approx P_{\text{по}}, \end{aligned} \qquad (11)$$

поскольку $P_{\text{по}}$ близко к 1, а $P_{\text{лт}}$ имеет порядок 10^{-5}–10^{-7}. Традиционный метод расчета $P_{\text{по}}$ учитывает лишь один член в точном выражении (11), хотя вероятность того, что сигнал полностью попадет в элемент разрешения, практически равна нулю. Поэтому всегда сигнал обнаруживается именно двумя обнаружителями и выражение (11) дает более точное выражение для расчета $P_{\text{по}}$, хотя, с точки зрения инженерных расчетов, эта поправка несущественна.

Заключение

Получены выражения, позволяющие рассчитывать вероятности правильного обнаружения информационного сигнала при заданной ложной тревоге в том случае, когда мощность информационного эхосигнала делится между соседними элементами разрешения бортовой РЛС. Полученные выражения уточняют известную методику расчета характеристик обнаружителей и позволяют определить энергетические потери при делении информационного сигнала.

Библиографический список

1. *Шепета А. П.* Определение зоны поиска надводного объекта по данным предварительного целеуказания // Информационно-управляющие системы. 2012. № 4 (59). С. 98–99.

2. *Бажин С.* А., Бестугин А. Р., Шепета А. П. Поиск надводных объектов по данным предварительного указания // Оборонная техника. 2001. № 6–7. С. 71–72.

3. *Тверской Г. Н., Терентьев Г. К., Харченко И. П.* Имитаторы эхо-сигналов судовых радиолокационных станций. Л.: Судостроение, 1973. 228 с.

4. *Изранцев В. В., Шепета Д. А.* Моделирование внешних сигналов бортовых приборных комплексов летательных аппаратов пятого поколения // Научное приборостроение. 2000. Т. 10, № 2. С. 14–19.

5. *Подоплёкин Ю. Ф., Шепета А. П.* Выбор морских целей по энергетическим признакам // Морская радиоэлектроника. 2018. № 4 (66). С. 34–37.

6. *Давидчук А. Г., Шепета Д. А.* Математические модели эхо-сигналов кораблей, наблюдаемых локаторами бортовых систем обработки информации // Информационно-управляющие системы. 2005. № 6(19). С. 2–8.

УДК 53.072.8, 60-7 **DOI: 10.31799/978-5-8088-1701-2-2022-2-48-50**

Б. В. Давидович*
магистрант
К. Б. Гурнов*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИССЛЕДОВАНИЕ ВОЗМОЖНОСТИ ИЗМЕРЕНИЯ ПУЛЬСОКСИМЕТРИИ В ВОДЕ

Исследована возможность измерения пульса и сатурации в водной среде. Осуществлен эксперимент по измерению пульса и сатурации в воде. Произведена оценка полученных результатов эксперимента.
Ключевые слова: *пульс, сатурация, микроконтроллер, датчик, обработка, расчет.*

B. V. Davidovich *
Postgraduate Student
K. B. Gurnov*
PhD., Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

INVESTIGATION OF THE POSSIBILITY OF MEASURING PULSE OXIMETRY IN WATER

The ability of measuring pulse and saturation in an aquatic environment is investigated. An experiment of measuring and saturation in water was conducted. The results of the experiment were evaluated.
Keywords: *pulse, saturation, microcontroller, sensor, processing, calculation.*

В современном мире пульсоксиметрия применятся в разных сферах. Пульсоксиметры устанавливаются в смарт-часах, фитнес-браслетах, используются в ходе тренировок спортсменов, в медицинской диагностике. Однако подобные измерения проводятся в воздушной среде. В некоторых случаях аналогичные измерения требуется осуществлять в воде, что позволит, например, эффективнее проводить тренировки пловцов или диагностировать различные нарушения в работе организма дайверов.

Процесс измерения пульса и сатурации основывается на принципах фотоплетизмографии (от греч. plethysmos «наполнение»), позволяющих выделить артериальную составляющую абсорбции света для определения оксигенации артериальной крови. Исходя из методики фотоплетизмографии, участок, на котором исследуется кровоток, располагается между источником света и фотодиодом (рис. 1) [1].

Абсорбцию света средой возможно вычислить по закону Бугера – Ламберта – Беера (1):

$$I(l) = I_0 e^{-k_\lambda l}, \quad (1)$$

где $I(l)$ – интенсивность света, прошедшего слой вещества толщиной l, I_0 – интенсивность света на входе в вещество, k_λ – показатель поглощения [2].

Из закона Бугера – Ламберта – Беера следует, что чем меньше толщина слоя, сквозь который проходит свет, тем меньше светопоглощение. Следовательно, если обеспечить плотное прилегание светодиода и фотодиода к поверхности тела, то можно пренебречь поглощением

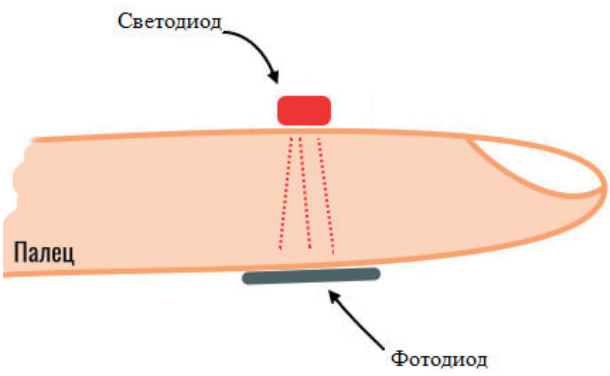


Рис. 1. Пульсоксиметр

света внешней средой. Тогда, вне зависимости от того, водная среда или воздушная, обеспечив должное прилегание датчика к поверхности, можно фиксировать пульс и сатурацию. Стоит учитывать, что если надавливание на датчик будет чрезмерным, то произойдет нарушение артериального кровотока и будет наблюдаться венозная пульсация. Пульсоксиметр не способен отличать пульсацию артерий от пульсации вен, а потому начинает включать в расчет абсорбцию света венозной кровью, тем самым занижая результат. Венозная пульсация может наблюдаться и при различных заболеваниях тканей и сердечно-сосудистой системы.

Скорость реакции пульсоксиметра на изменения сатурации и пульса определяется линейной скоростью артериального кровотока, которая зависит от физиологических процессов организма. У здорового человека кровь после очередного удара сердечной мышцы достигает пальца руки спустя 3–5 с, а уха – через 2–3 с, но при различных заболеваниях интервал может возрасти до 1,5 мин.

Расчет сатурации можно проводить по каждой из волн фотоплетизмограммы, а пульс – по каждому интервалу между волнами. При представлении информации в таком виде, точных данных получить не удастся, поэтому берут усредненное значение за период наблюдения. В основном эти периоды наблюдения варьируются от 3 до 20 с [3].

После анализа принципов пульсоксиметрии был проведен эксперимент по измерению сатурации и пульса в водной и воздушной среде на здоровом человеке. Для этого выбраны платформа Arduino UNO и пульсоксиметр KY-039. Схема датчика представлена на рис. 2 [4].

Датчик подключался к входу АЦП микроконтроллера, схема подключения представлена на рис. 3.

Для проведения эксперимента разработана программа в среде ARDUINO IDE. Программа снимает показания с датчика каждые 5 с и выводит их на монитор компьютера по последовательному порту.

Перед экспериментом все контактные площадки, smd-резисторы и разъем PLS-3 с подключенным кабелем были загерметизированы во избежание сбоев в работе датчика и нарушения его целостности. Герметизация была проведена при помощи термоклея, не проводящего электрический ток. На рис. 4 представлен подготовленный к экспериментам датчик.

Эксперимент по замеру пульса проводился на живом человеке, находящемся в состоянии покоя, при комнатной температуре.

При проведении эксперимента потребовалось учесть, что показания датчика KY-039 считываются встроенным АЦП микроконтроллера ARDUINO UNO. Точность встроенного 10-разрядного АЦП определяется формулой

$$V = (5/1024)\cdot ADC, \qquad (2)$$

где ADC – значение, поданное на вход АЦП [5].

В результате измерений пульса в воздухе был получен следующий график (рис. 5). По вертикали – значения пульса, по горизонтали – номера замеров, замеры проводились каждые

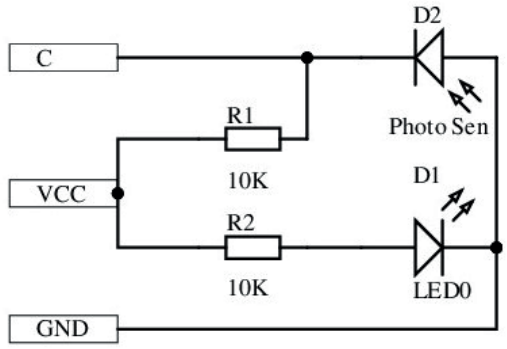


Рис. 2. Схема датчика KY-039

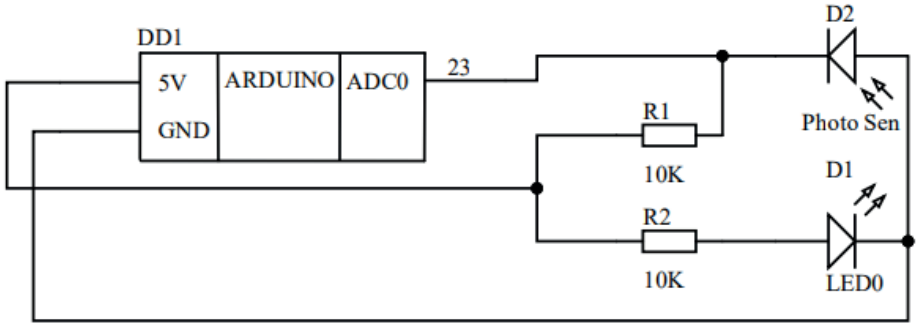


Рис. 3. Схема подключения датчика KY-039 к микроконтроллеру ARDUINO UNO

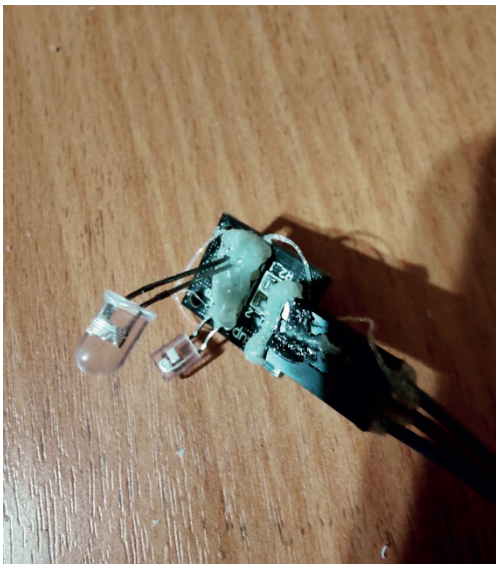


Рис. 4. Герметизация датчика

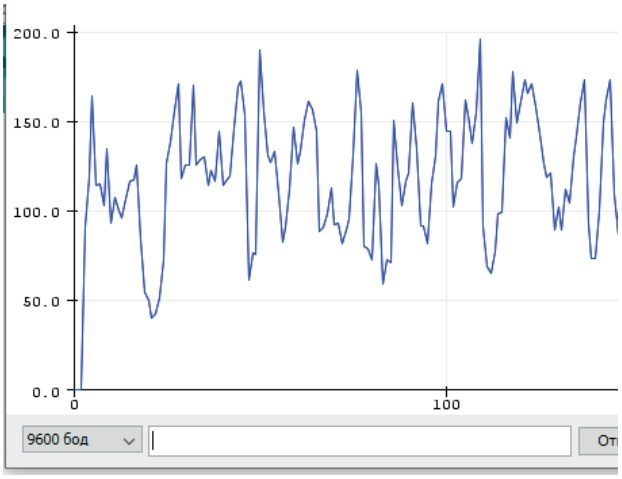


Рис. 5. Пульс в воздухе

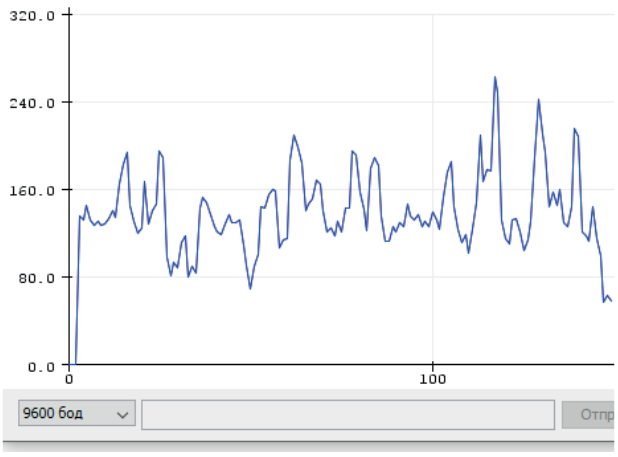


Рис. 6. Пульс в воде

5 с. Согласно принципам пульсоксиметрии, результат измерений – это среднее значение из всех полученных [6]. В ходе эксперимента было получено значение 126 уд./мин.

Для проведения эксперимента палец с датчиком был помещен в водную среду, не ниже уровня сердца. Полученный результат представлен в виде графика на рис. 6.

Согласно принципам пульсоксиметрии, слишком высокие одиночные пики в расчет не берутся. В результате эксперимента в воде таких всплесков около 9. По результатам измерений пульс оказался равным 136 уд./мин. Во время замеров производился замер пульса со смарт-часов Samsung galaxy gear s3. Во время замеров в воде пульс по показаниям смарт часов был равен 134 уд./мин.

Исходя из описанных теоретических выкладок и проведенного эксперимента, можно утверждать, что снятие пульса и сатурации в воде возможно.

Измерение сатурации и пульса в водной среде позволит повысить безопасность погружений с аквалангом, заранее определять наступление азотного наркоза у дайверов, разработать систему автоматического всплытия при потерях сознания и рассчитать минимальное время декомпрессионной паузы индивидуально для каждого ныряльщика.

Библиографический список

1. Пульсоксиметрия: физические принципы и применение в медицине: спец. практикум / Моск. гос. ун-т им. М. В. Ломоносова. М., 2008. 15 с.

2. Закон Бугера – Ламберта – Бера // Википедия. URL: https://ru.wikipedia.org/wiki/Закон_Бугера_—_Ламберта_—_Бера (дата обращения: 20.11.21).

3. *Шурыгин И. А.* Мониторинг дыхания: пульсоксиметрия, капнография, оксиметрия. М.: BINOM; СПб.: Невский Диалект, 2000. 301 с.

4. Датчик сердцебиения KY-039 Ардуино. URL: https://роботехника18.рф/ky-039-arduino/ (дата обращения: 20.11.21).

5. *Евсегнеев О.* Аналого-цифровые преобразования – АЦП. URL: https://robotclass.ru/tutorials/arduino_adc/ (дата обращения 20.11.21).

6. Sviluppo e validazione di un pulsossimetro a riflettanza indossabile al dito / Politecnico di Milano; relatore prof. A/ Aliverti; correlatore ing. P. Patete, ing. M. Torregian. Anno Accademico, 2014–2015.

УДК 004.65 **DOI: 10.31799/978-5-8088-1701-2-2022-2-51-54**

М. В. Загураева*
ассистент
Е. Л. Турнецкая*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ПРОГРАММНЫЕ ИНСТРУМЕНТЫ ДОСТУПА К РЕЛЯЦИОННЫМ БАЗАМ ДАННЫХ В МНОГОФУНКЦИОНАЛЬНЫХ ИНФОРМАЦИОННЫХ СИСТЕМАХ

Проведен краткий обзор способов организации доступа к данным в реляционных базах данных (БД). Представлены примеры программных инструментов для решения задач доступа к БД в многофункциональных информационных системах в зависимости от сферы применения, ресурсов, структуры данных и других факторов. Материалы доклада внедрены в учебный процесс при подготовке бакалавров по направлению «Прикладная информатика».
Ключевые слова: *базы данных, доступ к данным, бизнес-аналитика.*

M. V. Zaguraeva*
Assistant
E. L. Turnetskaya*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

SOFTWARE TOOLS FOR ACCESSING RELATIONAL DATABASES IN MULTIFUNCTIONAL INFORMATION SYSTEMS

A brief overview of the ways of organizing access to data in relational databases is carried out. Examples of software tools for solving problems of access to databases in multifunctional information systems are given, depending on the scope, resources, data structure and other factors. The materials of the report were introduced into the educational process in the preparation of bachelors in the direction of «Applied Informatics».
Keywords: *databases, data access, business intelligence.*

Введение

Исторически выделяют два основополагающих направления развития информационных технологий. К первому можно отнести создание вычислительных комплексов, позволяющих достаточно быстро преобразовывать значительные объемы данных с использованием сложных алгоритмов вычислений. В качестве второго направления можно рассмотреть появление информационных систем, в рамках которых предполагаются хранение и обработка больших объемов данных со сложной внутренней структурой.

На начальном этапе объем оперируемых данных был сравнительно небольшим и не требовал специальных программных инструментов и технологий для доступа к данным и их обработке. Большинство разработчиков ограничивалось использованием прямого доступа к данным с использованием функционала операционной системы [1].

Вследствие совершенствования аппаратной базы вычислительных комплексов, увеличения их быстродействия, а также в связи с появлением высокоемких электронных носителей информации возникла потребность в создании и совершенствовании новых технологий организации доступа к данным.

На настоящий момент можно выделить несколько основных и наиболее интересных задач, с которыми приходится сталкиваться разработчикам при проектировании информационных систем, предназначенных для хранения и обработки данных.

– Универсальность протоколов доступа к данным – возможность использовать различные платформы при проектировании информационных систем с использованием клиент-серверных баз данных.

– Комбинаторность и комплиментарность – возможность внедрять функционал технологий доступа к данным в различные информационные системы, сохраняя их гибкость и универсальность. Сочетать возможности современных языков программирования с технологиями доступа и управления данными.

– Многомерность массивов информации – оптимизация быстродействия системы, уменьшение чувствительности этого параметра к объему и структуре оперируемых данных.

– Доступность – использование функционала общедоступных готовых программных продуктов для решения уникальных задач конечного пользователя.

Особое место в семействе баз данных занимают реляционные базы данных.

К несомненным достоинствам реляционных БД относятся:

– четкая и логичная структура данных;

– независимость и целостность данных;

– широкая область применения.

Рассмотрим примеры решения указанных задач на примере реляционных баз данных.

Способы доступа к данным

1. Стандартизация протоколов доступа к данным.

Изначально в связи с ограниченным количеством используемой вычислительной техники и монополией первых производителей компьютеров и программных продуктов, как правило, компании использовали одну систему управления базами данных. Доступ к данным выполнялся либо через внешний интерфейс этой системы, либо через приложения, написанные для работы с ней. С развитием информационно-вычислительной техники возникла потребность в стандартизации взаимодействия между различными базами данных. Основными предпосылками для этого процесса стали следующие тенденции:

– появление персональных компьютеров;

– разнообразие операционных систем;

– рост рынка программных продуктов, поддерживающих СУБД;

– повсеместное использование клиент-серверных технологий СУБД;

– реализация многопользовательского доступа к БД.

К таблицам реляционной БД можно получить доступ с помощью технологии связи по низкоуровневому интерфейсу OLE DB, через программный интерфейс ADO (Microsoft ActiveX Data Objects) или с помощью специализированных драйверов для доступа к реляционным данным формата ODBC (Open Database Connectivity) [2].

Для подключения к реляционным БД на основе стандарта SQL-92 можно использовать решения компании CDATA (https://www.cdata.com/), специализирующейся на поставках программных решений, основанных на стандартах ODBC, JDBC, ADO.NET, ODATA и JSON.

2. Внедрение языка запросов в язык программирования на примере ESQL/C.

В случаях когда БД является частью информационной системы, реализованной средствами современного языка программирования, рационально использовать для доступа к данным встроенный язык запросов. Примером такой интеграции является ESQL/C.

Основной код программы создается на языке C. Операторы языка запросов SQL внедряются в код программы наравне с операторами C. Программные переменные также могут выступать в качестве параметров запроса. Выполнение оператора SQL по сути является обращением к серверу данных (клиент-серверные технологии) [3].

3. Применение специализированных языков программирования для организации доступа к БД.

Перед аналитиками остро встает вопрос выбора программного инструмента доступа к данным. Во многих случаях приходится ограничиваться индивидуальными разработками под конкретную БД, что, безусловно, увеличивает затраты различных ресурсов на разработку и эксплуатацию подобной информационной системы. В этом случае разрабатывают программные коды доступа к данным, а также последующей их обработки и визуализации на основе специализированных библиотек языков программирования, в частности библиотеки по обработке и анализу данных pandas и библиотеки научной графики matplotlib для Python. Также для анализа данных применяют языки программирования, специализирующиеся на аналитической обработке, например язык R, который может взаимодействовать с БД SQL Server, MySQL, PostgreSQL, SQLite и др.

В настоящее время насчитывается более 4 млн веб-систем, контент которых хранится в базах данных [4]. Реализация таких систем, как правило, основана на использования патерна MVC (Model-View-Controller), позволяющего отделить интерфейс пользователя, базы данных и контроллера. В качестве контроллера, осуществляющего связь между компонентами и доступа к БД, используют межплатформенные языки сценариев, например PHP.

4. Использование функционала пакетов прикладных программных средств компании Microsoft для обработки многомерных массивов данных.

Специфика предметной области не всегда позволяет структурировать данные таким образом, чтобы получить стандартную двумерную таблицу с сохранением целостности данных и минимизацией их избыточности. В этом случае подобную совокупность данных можно рассматривать как многомерный массив данных. С другой стороны, статистическая обработка данных из реляционной БД может привести к созданию многомерного массива данных.

Например, данные предприятия по ценам, ассортименту, покупателям в большинстве случаев можно представить в виде двумерных таблиц реляционной БД. Но в целях проведения анализа бизнес-процессов эти данные легко преобразуются в многомерный массив (куб данных или метакуб) для дальнейшей обработки в соответствии с принципами технологии OLAP (On-Line Analytical Processing).

Технология OLAP применяется, чтобы, во-первых, упростить работу с большим объемом накопленных данных о деятельности предприятия, во-вторых, превратить набор количественных показателей в качественные, в-третьих, производить анализ данных, используя быстрый, единообразный, оперативный доступ к разнообразным формам представления информации. Такие формы, полученные на основании первичных данных, позволяют пользователю сформировать полноценное представление о деятельности предприятия [5].

Обработку данных OLAP-куба можно реализовать на основе BI-линейки Microsoft и БД MS SQL. В этом случае ETL-процессы реализуют средствами службы SQL Server Integration Services, интеллектуальный анализ данных – с помощью SQL Server Analysis Services [6].

5. Применение систем бизнес-аналитики.

Цифровую основу предприятий Индустрии 4.0 составляют корпоративные интегрированные среды, включающие различные источники данных. Результаты исследования Data Age 2025 «The Evolution of Data to Life-Critical» компании IDC (https://www.idc.com/) показывают, что в настоящее время накоплено более одного триллиона гигабайт информации, большая часть которых относится к корпоративным данным [7]. При этом только 2% из них относят к структурированным ресурсам. Для перевода пассивно хранящихся данных в активные ресурсы и для комплексной многоаспектной обработки исходных данных требуется обеспечивать доступ к ним как со стороны разработчика и администратора базы данных – специалистов в области прикладного программирования, так и со стороны грамотных пользователей, не обладающих такими навыками. Поэтому широкое распространение получили системы бизнес-аналитики, обладающие интуитивно понятным пользовательским интерфейсом.

При выборе аналитической системы принимают во внимание исследование рынка платформ бизнес-аналитики. В частности, компания Gartner (https://www.gartner.com/) проводит сравнение BI-систем на основании 17 критериев, оценивающих возможности инструментальных средств рассматриваемых систем: представления информации; интеграции с другими приложениями и анализа данных [8]. По результатам исследования «Magic Quadrant for Analytics and Business Intelligence Platforms» [9] лидирующие позиции в рейтинге лучших продуктов, предназначенных для аналитической обработки и визуализации данных, занимают BI-системы от компаний Microsoft (https://powerbi.microsoft.com/ru-ru/why-power-bi/), Tableau (https://www.tableau.com/) и Qlick Sense (https://www.qlik.com/ru-ru/). Особенностью этих систем является низкий порог входа для специалистов по обработке структурированных данных. Как правило, интегрированные среды предприятия или веб-систем включают БД разных производителей. Поэтому BI-системы обладают широким спектром специализированных коннекторов – ODBC-драйверов для автоматического подключения. В частности, Tableau обладает встроенными драйверами, которые автоматически создают связь приложения с базами MySQL, Access, SQL Server, DB2 и др. Для нетривиального анализа в таких системах предусмотрены встроенные языки программирования, например DAX для Tableau [10].

6. Использование программного инструмента MS EXCEL.

Интегрированная среда предприятия может включать приложения компании Microsoft, взаимосвязь между которыми происходит автоматически на основе бесшовных технологий. Поэтому в случае ограниченного бюджета можно использовать встроенные программные BI-инструменты табличного процессора Excel [11]:

– Power Query для осуществления сбора данных из различных источников, например БД SQL, Oracle, Access и т. д.;

– Power Pivot для аналитической обработки данных и построения на их основе сводных диаграмм и таблиц;

– Power Map для связи результатов вычислений с их географическим положением;

– Power View для создания интерактивных отчетов.

Предобработку и анализ данных проводят средствами визуального редактора и с помощью внутреннего языка формул M среды Power Query.

Заключение

Современный мир называют VUCA-миром (Volatility, Uncertainty, Complexity, Ambiguity – нестабильность, неопределенность, сложность и неоднозначность). В нем экономическая успешность компании во многом зависит от полноты и своевременного предоставления информации, на которую опираются руководители при принятии решений. Цифровая трансформация предприятий, основанная на извлечении новой информации из сохраненных данных, позволяет прогнозировать и адекватно реагировать на внешние изменения. На основе обработки данных получают также метрики для измерения эффективности цифрового предприятия, бизнес-процессы которого уже модифицированы в соответствии с принципами Индустрии 4.0. К таким метрикам цифровой трансформации можно отнести:

– способность к адаптации при возникновении проблем в работе бизнеса за счет использования цифровых возможностей для восстановления бизнес-операций;

– извлечение преимуществ из изменившихся условий для экономического роста;

– принятие инновационных решений на основании информации, полученной после анализа данных;

– оценивание трендов развития партнеров по бизнесу, клиентов и сотрудников.

Обзор способов предоставления доступа к реляционным базам данных доказывает, что в зависимости от предметной области, сложности программной системы, языков программирования, которые использованы при ее построении, модели архитектуры, квалификации аналитиков можно применять различные программные решения.

Б. Йохансен говорит, что «лидеров нового поколения отличают Vision (видение), Understanding (понимание), Clarity (ясность) и Agility (быстрота)» [12]. Поэтому при подготовке бакалавров по направлению подготовки 09.03.03 «Прикладная информатика» следует акцентировать внимание на многовариантных подходах к решению поставленных задач. Поэтому в дисциплинах «Информационные системы и технологии», «Базы данных», «Программная инженерия» при выполнении практических заданий обучающимся предлагают задачи, которые позволят получить навыки организации доступа к базам данных различными способами, повышая их конкурентоспособность на рынке труда.

Библиографический список

1. *Нестеров С.* А. Базы данных: учеб. и практикум для вузов. М.: Юрайт, 2021. 230 с.

2. *Стасышин В. М., Стасышина Т. Л.* Базы данных: технологии доступа: учеб. пособие для вузов. 2-е изд., испр. и доп. М.: Юрайт, 2021. 164 с.

3. Документация по драйверам ODBC – Overview // Microsoft Docs. URL: https://docs.microsoft.com/en-us/sql/odbc/reference/odbc-overview?view=sql-server-2017 (дата обращения: 19.11.2021).

4. Отчет о состоянии цифровой сферы Digital-2020. URL: https://wearesocial.com/blog/2020/01/digital-2020/ (дата обращения: 22.11.2021).

5. *Козикова П. В.* Краткий обзор OLAP технологии // Студенческий научный форум: матер. VII Междунар. студ. науч. конф. URL: https://scienceforum.ru/2015/article/2015017686 (дата обращения: 22.11.2021).

6. Документация по службам интеллектуального анализа данных компании Microsoft. URL: https://docs.microsoft.com/ru-ru/analysis-services/data-mining/data-mining-ssas?view=asallproducts-allversions (дата обращения: 22.11.2021).

7. Результаты исследования компании Data Age 2025 «The Evolution of Data to Life-Critical». URL: https://www.import.io/wp-content/uploads/2017/04/Seagate-WP-DataAge2025-March-2017.pdf (дата обращения: 22.11.2021).

8. Тенденции мирового ИТ-рынка. URL: https://www.tadviser.ru/index.php/Статья:Тенденции_мирового_ИТ-рынка (дата обращения: 22.11.2021).

9. Блог компании Power BI Russia. URL: http://powerbirussia.ru/2021/03/07/gartner-magic-quadrant-2021/ (дата обращения: 22.11.2021).

10. Официальная документация по Tableau. URL: https://www.tableau.com/ru-ru/learn/whitepapers/tableau-visual-guidebook (дата обращения: 22.11.2021).

11. Официальная документация по Power BI. URL: https://powerbi.microsoft.com/ru-ru/desktop/ (дата обращения: 22.11.2021).

12. Совместное исследование Союза «Молодые профессионалы (Ворлдскиллс Россия)» и проекта Global Education Futures. URL: https://rda.worldskills.ru/project/future-skills-2 (дата обращения: 22.11.2021).

УДК 621.396 **DOI: 10.31799/978-5-8088-1701-2-2022-2-55-57**

М. С. Иванова*
аспирант
В. И. Исаков*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАСЧЕТ ЗОНЫ ПОИСКА ФИЗИЧЕСКИХ ОБЪЕКТОВ С УЧЕТОМ АПРИОРНОЙ ИНФОРМАЦИИ ОБ ИХ ПЕРЕДВИЖЕНИЯХ

Приводятся аналитические выражения и результаты расчета вероятности обнаружения движущихся физических объектов от размеров зоны поиска, «накрывающей» обнаруживаемый объект с заданной вероятностью.
Ключевые слова: *зона поиска, вероятность накрытия, закон распределения, элемент разрешения, режим обзора, морская поверхность, вероятность ложной тревоги, вероятность обнаружения.*

M. S. Ivanova*
PhD Student
V. I. Isakov*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

CALCULATING THE SEARCH AREA FOR PHYSICAL OBJECTS TAKING INTO ACCOUNT A PRIORI INFORMATION ABOUT THEIR MOVEMENTS

The paper presents analytical expressions and the results of calculating the probability of detecting moving physical objects from the size of the search zone, «covering» the detected object with a given probability.
Keywords: *search area, coverage probability, distribution law, resolution element, view mode, sea surface, false alarm probability, detection probability.*

В случае чрезвычайного происшествия на море немедленно организовывается спасательная операция. Одной из первых возникает задача поиска терпящих бедствие. Поиском занимаются в том числе летательные аппараты с помощью радиолокационных средств. Летательному аппарату выдается целеуказание – координаты точки, в которой с наибольшей вероятностью могут находиться терпящие бедствие. Но точные координаты чаще всего неизвестны, кроме того, все средства управления работают с некоторой ошибкой. И сами разыскиваемые могут не находиться на одном месте, а двигаться под воздействием ветра и течения. Таким образом, вместо одной точки возникает некоторая зона, в которой необходимо проводить поиски. Размеры этой зоны сильно зависят от времени подлета к ней, силы и направления ветра и скорости течения, т. е. скорости перемещения терпящих бедствие, точности работы всех систем управления спасательной операцией. Возникает задача определения оптимальных размеров зоны, в пределах которой необходимо проводить поиск [1].

На рис. 1 приведена схема построения зоны поиска обнаруживаемого объекта.

В точке 0 выдаются предполагаемые координаты объекта поиска, отрезок $0L$ – расстояние до центра зоны, $(L - \Delta L, L + \Delta L)$ – размер зоны поиска по дальности, $(-\Delta\alpha, +\Delta\alpha)$ – размер зоны по углу азимуту. В дальнейшем предполагается, что V – максимальная возможная скорость объекта, v – предполагаемая средняя скорость

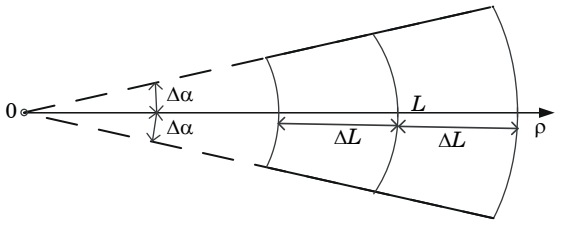


Рис. 1. Схема параметров зоны поиска терпящих бедствие

движения объекта поиска в данных условиях в неизвестном направлении φ, t – предполагаемое время подлета до зоны поиска, σ^2 – дисперсия ошибок координат объекта в зоне поиска.

Тогда в момент начала поиска плотность распределения координат объекта поиска $f(x,y)$ можно записать в виде [2]

$$f(x,y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(x-L)^2+y^2+(vt)^2}{2\sigma^2}\right)\times I_0\left(\frac{vt\sqrt{(x-L)^2+y^2}}{\sigma^2}\right), \quad (1)$$

где $I_0(.)$ – функция Бесселя нулевого порядка от мнимого аргумента, модифицированная функция Бесселя.

Так как локационная станция системы поиска работает в полярной системе координат, то и зона поиска тоже определяется в полярной системе. Значит, функция плотности распределения координат объекта принимает следующий вид [2]:

$$f(\rho,\alpha,v) = \frac{\rho}{2\pi\sigma^2}\times \exp\left(-\frac{\rho^2-2\rho L\cos\alpha+L^2+(vt)^2}{2\sigma^2}\right)\times I_0\left(\frac{vt\sqrt{\rho^2-2\rho L\cos\alpha+L^2}}{\sigma^2}\right). \quad (2)$$

Используя выражение (2), можно определить максимальный размер зоны поиска. Этот параметр наиважнейший в спасательной операции, так как определяет время поиска.

Выбирая величину зоны поиска, мы тем самым определяем и ту вероятность накрытия ($P_{\text{накр}}$), с которой объект поиска окажется в этой зоне. Чем больше зона поиска, тем более вероятно, что объект в ней окажется, и мы его найдем, но тем больше будет потрачено времени на поиски, в этом случае помощь может опоздать.

Из выражения (2) следует, что вероятность накрытия зоной сканирования обнаруживаемого объекта при заданной средней скорости его движения v [2, 3]

$$P_{\text{накр}} = 2\int_{L-\Delta L}^{L+\Delta L}\int_{0}^{\Delta\alpha/2} f(\rho,\alpha,v)\,d\rho d\alpha. \quad (3)$$

Таким образом, используя формулу (3), можно посчитать вероятность накрытия зоной сканирования обнаруживаемого объекта в зависимости от размеров зоны по дальности и углу.

На рис. 2, 3 приведены результаты расчетов вероятности накрытия зоной сканирования обнаруживаемого объекта в зависимости от размеров зоны по дальности и углу соответственно.

Расчет проводился при следующих данных: расстояние до центра зоны поиска L 150 км, время подлета до центра t = 0,5 ч, максимально возможная скорость объекта V 4 км/ч, средняя скорость 2 км/ч, σ = 0,3.

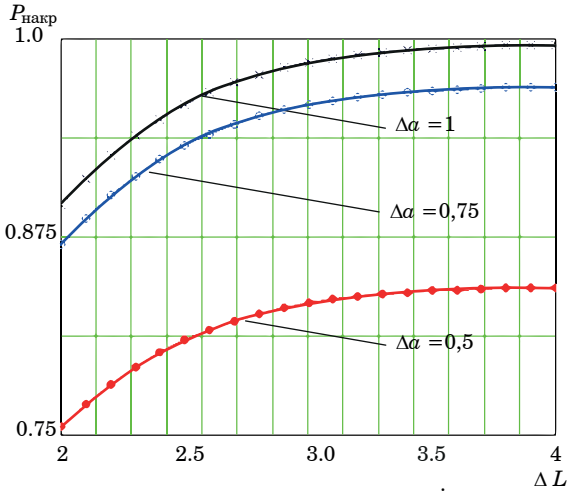


Рис. 2. Вероятность накрытия зоной сканирования обнаруживаемого объекта в зависимости от размеров зоны по дальности

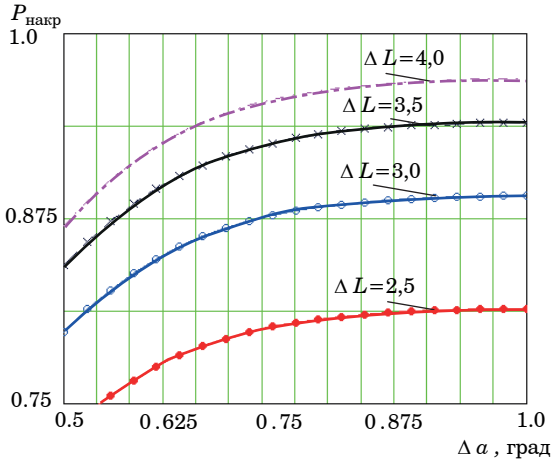


Рис. 3. Вероятность накрытия зоной сканирования обнаруживаемого объекта в зависимости от размеров зоны по углу

Графики зависимости вероятности накрытия объекта от размера зоны по дальности (см. рис. 2) рассчитывались для значений Δa, *равных* 0,5, 0,75 и 1,0 град.

Графики зависимости вероятности накрытия объекта от размера зоны по углу (см. рис. 3) рассчитывались для значений ΔL, *равных* 2,5, 3,0, 3,5 и 4,0 *км*.

Таким образом, имея графики и задавшись вероятностью накрытия объекта, можно определить размеры зоны поиска по обеим координатам.

Библиографический список

1. *Шепета А. П.* Определение зоны поиска надводного объекта по данным предварительного целеуказания // Информационно-управляющие системы. 2012. № 4 (59). С. 98–99.

2. *Shepeta A. P., Nenashev V. A.* Optimization the size of the search area for moving physical objects based on preliminary target designation data // 2021 Wave Electronics and its Application in Information and Telecommunication Systems. St. Petersburg, 2021.

3. *Isakov V. I., Shepeta D. A.* Simulation of location signals when determining a coastal edge // 2021 Wave Electronics and its Application in Information and Telecommunication Systems. St. Petersburg, 2021. P. 1–5.

4. *Подоплекин Ю. Ф., Шепета Д. А., Иванова М. С.* Расчет зоны поиска физических объектов с учетом априорной информации об их передвижении // Морская радиоэлектроника. 2021. № 3 (77).

УДК 621.38 **DOI: 10.31799/978-5-8088-1701-2-2022-2-58-61**

В. А. Килимник*
кандидат технических наук
А. А. Чекменева*
магистрант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

УСТРОЙСТВО ДЛЯ ЗАБОРА ПРОБЫ ВЫДЫХАЕМОГО ВОЗДУХА

Рассмотрены критерии отбора пробы, позволяющие выбрать необходимые сенсоры для выполнения данного условия. Представлена структура устройства для забора пробы выдыхаемого воздуха на основе датчика кислорода и управляющего микроконтроллера.
Ключевые слова: *фаза выдоха, выдыхаемый воздух, проба, датчик кислорода, датчик углекислого газа, схема, устройство.*

V. A. Kilimnik*
PhD, Tech.
A. A. Chekmeneva*
Postgraduate Student
**St. Petersburg State University of Aerospace Instrumentation*

DEVICE FOR SAMPLING EXHALED AIR

The sampling criteria allowing to select the necessary sensors to fulfill this condition are considered. The structure of a device for sampling exhaled air based on an oxygen sensor and a control microcontroller is presented.
Keywords: *exhalation phase, exhaled air, sample, oxygen sensor, carbon dioxide sensor, speed, circuit, device.*

В соответствии с международными данными по отбору проб выдыхаемого воздуха известно, что можно использовать датчики кислорода или углекислого газа [1–3]. По данным, полученным с них, принимается решение о заборе пробы выдыхаемого воздуха. В рамках данной работы был произведен анализ существующих датчиков кислорода [4] и углекислого газа [5]. Так как не существует единого мнения о целесообразности того или иного датчика, в макете устройства используется датчик кислорода Oksik-3. Основными критерием выбора были: быстродействие, возможность работы без принудительной прокачки воздушной пробы и рабочий диапазон.

Для сравнительного анализа датчиков кислорода по быстродействию была составлена электрическая принципиальная схема (рис. 1) [4].

Каждый канал оценки содержания кислорода в выдыхаемом воздухе выполнен по идентичной схеме, поэтому остановимся на описании одного канала. Датчики подключаются к усилителям через разъемы $X1$ и $X2$. Сигнал с датчика кислорода типа Oksik-3 подается на неинвертирующий вход повторителя напряжения, собранного на сдвоенном операционном усилителе (ОУ) типа LM358. По условиям производителя датчика кислорода, он должен нагружаться на резистор сопротивлением не менее 2 МОм. В данном варианте используется нагрузочный резистор 4,3 МОм, что не противоречит условиям применения датчика и не вызывает большого смещения выходного напряжения ОУ. Калибровка показаний датчика кислорода осуществляется при включении макета устройства и заключается в установке выходного напряжения повторителя напряжения на ОУ $D1.B$, с помощью переменного многооборотного резистора $R2$ с напряжением 0,21 В. Это соответствует величине концентрации кислорода в окружающей атмосфере в 21%. По такой методике калибруются почти все серийно выпускаемые датчики кислорода. При дальнейшей обработке показаний датчиков кислорода необходимо учитывать значения калибровки.

Был изготовлен макет устройства (рис. 2). Он представляет собой трубку диаметром 20 мм, внутри которой расположен датчик дыхания в виде шайбы, формирующий перепад давления в зависимости от фазы дыхания.

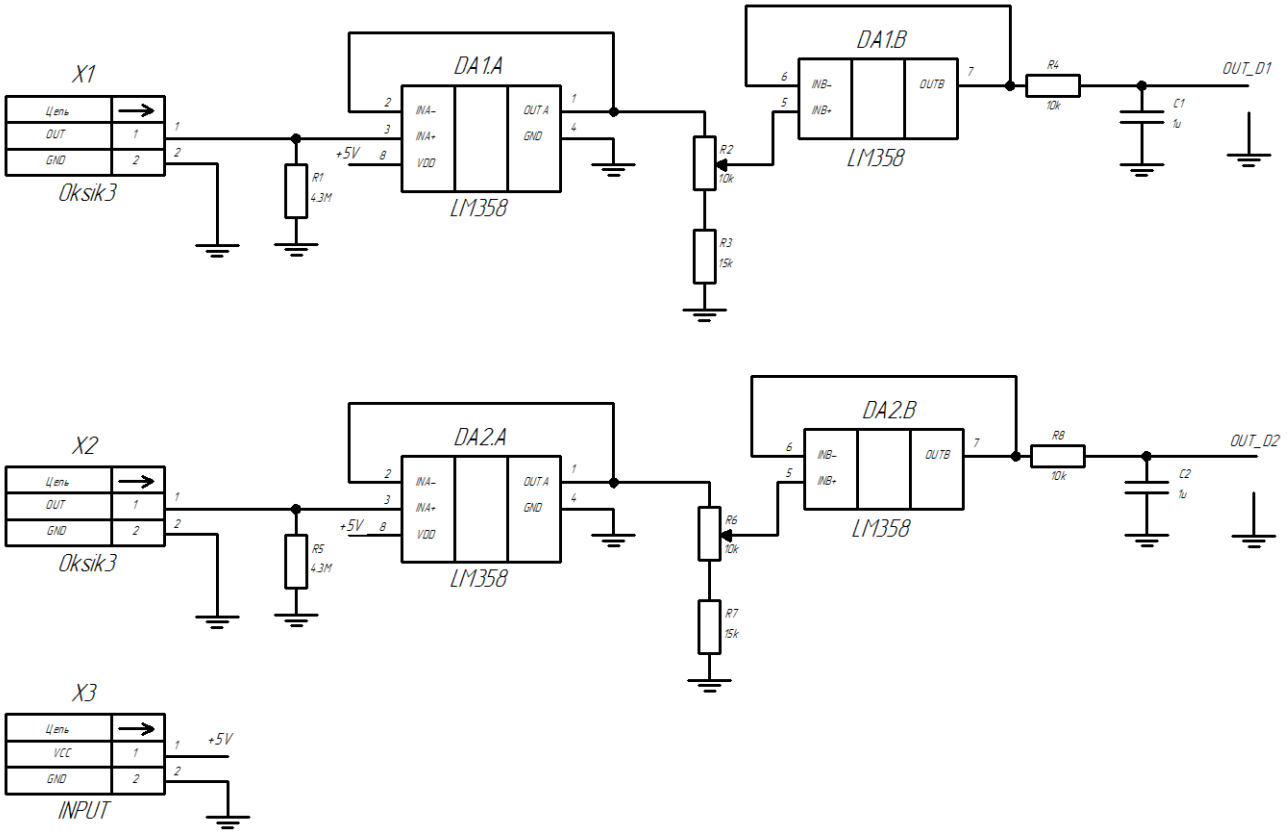


Рис. 1. Электрическая принципиальная схема подключения двух датчиков кислорода

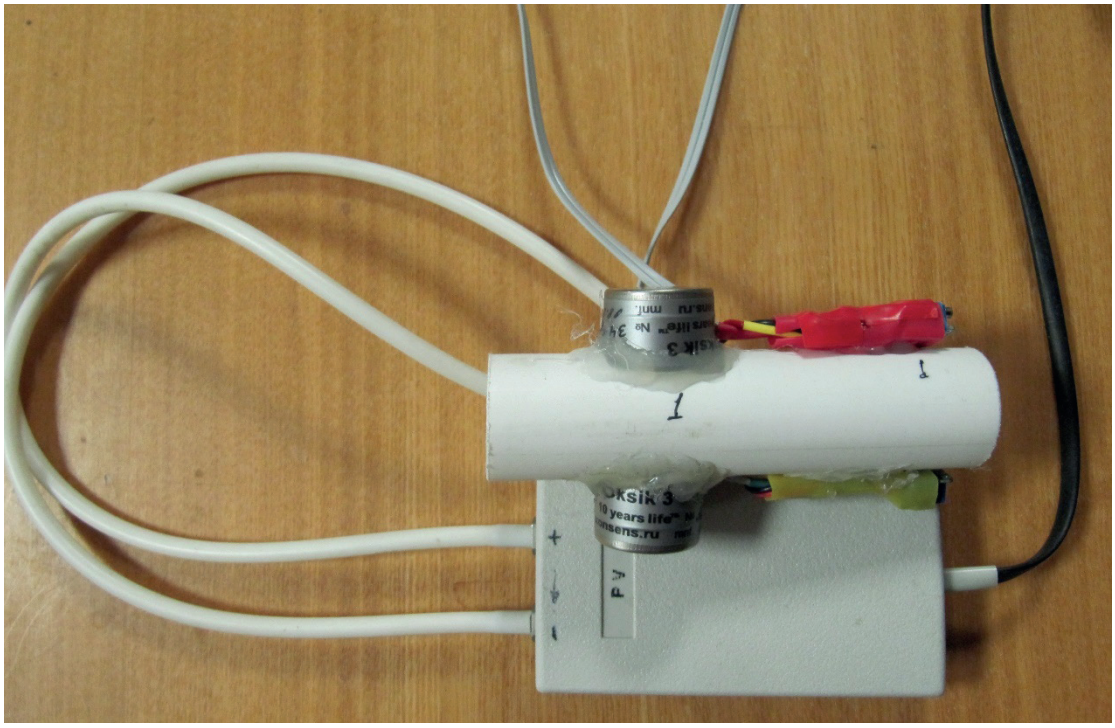


Рис. 2. Общий вид макета устройства для сравнения быстродействия датчиков кислорода и показаний датчика дыхания

С двух сторон расположены исследуемые датчики кислорода, которые встроены в канал и анализируют содержание кислорода в выдыхаемом воздухе.

С помощью дифференциального датчика давления регистрируется фаза дыхания. На рис. 3 представлена осциллограмма записи с датчика давления, характеризующая фазу выдоха и сигналы с датчиков кислорода с быстродействием 1 и 5 с.

Запись осуществлялась в программной среде LGraph2 с частотой дискретизации сигналов 1 кГц и разрешением 12 бит. На верхней части рисунка показаны фазы четырех выдохов, на среднем графике приставлен сигнал датчика кислорода с высоким быстродействием (1 с), на

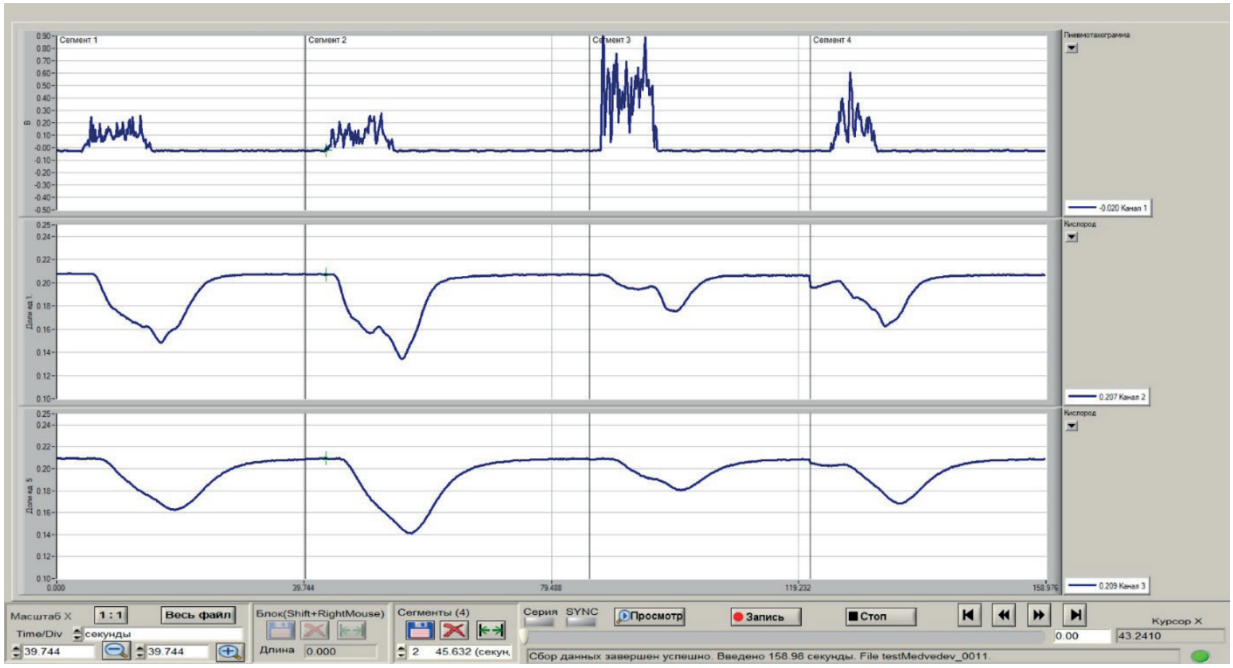


Рис. 3. Осциллограмма записи с дифференциального датчика давления и датчиков кислорода с разным быстродействием

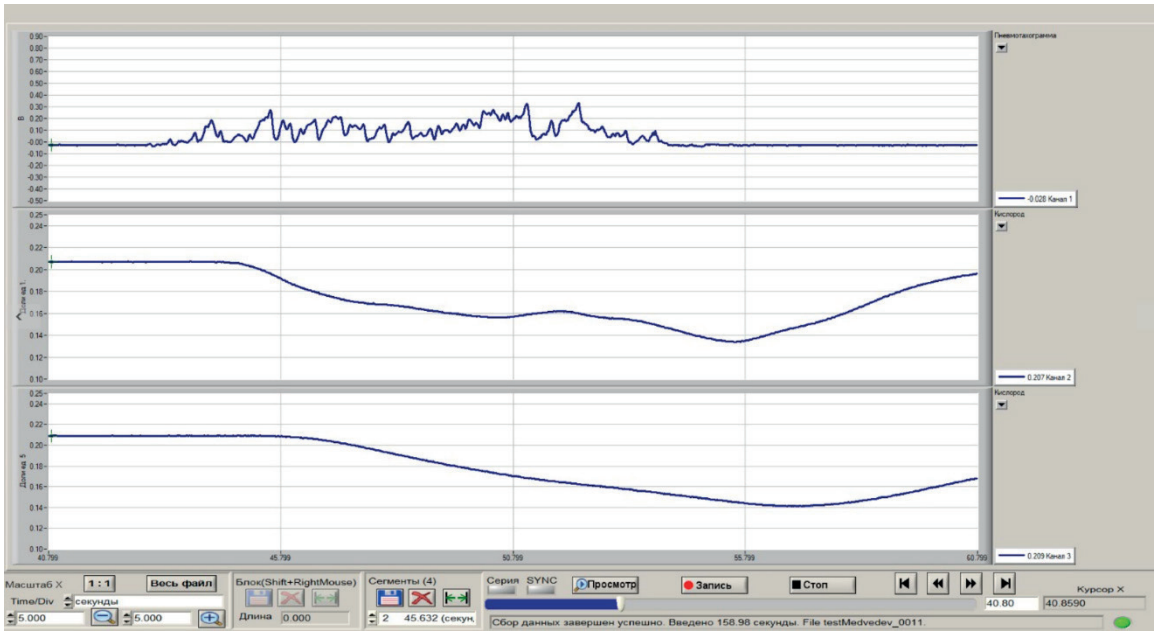


Рис. 4. Детальная осциллограмма одной фазы выдоха

нижнем – со средним быстродействием (5 с). Видно, что сигнал с датчика с высоким быстродействием более детально показывает содержание кислорода в вдыхаемом воздухе.

На рис. 4 представлена детальная осциллограмма одной фазы выдоха, позволяющая увидеть содержание кислорода с двух датчиков с разные моменты этой фазы. Видно, что более быстродействующий датчик показывает больше информации о содержании кислорода в выдыхаемом воздухе.

Таким образом, был изготовлен макет устройства для исследования датчиков кислорода. В перспективе данная работа позволит создать критерии выбора датчиков кислорода для различных устройств по анализу состава газов в выдыхаемом воздухе.

Библиографический список

1. *Lee S. M., Falconer I. H. E., Madden T., Laidler P. O.* Characteristics of oxygen concentration and the role of correction factor in realtime GI breath test // Gastrointestinal infection. 2020. P. 6.

2. *Anderson J. C., Hlastala M. P.* Breath tests and airway gas exchange // Pulm Pharmacol Ther. 2007. P. 1–7.

3. *Braden B.* Methods and functions: breath tests // Best Pract Res Clin Gastroenterol 2009. P. 337–52.

4. Датчик кислорода. URL: https://www.eksis.ru/catalog/sensors-and-mikrokompressory/product293.php (дата обращения: 09.11.2021).

5. Датчик углекислого газа SensirionSCD30 SensorModule. URL: https://docs.rs-online.com/2d49/0900766b816b6f9d.pdf (дата обращения: 09.11.2021).

УДК 621.38 **DOI: 10.31799/978-5-8088-1701-2-2022-2-62-64**

В. А. Килимник*
кандидат технических наук
А. А. Чекменева*
магистрант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОЦЕНКА СОСТАВА ВЫДЫХАЕМОГО ВОЗДУХА С ПОМОЩЬЮ ГАЗОВЫХ ДАТЧИКОВ

Представлен обзор литературы по анализу выдыхаемого воздуха. Отражена актуальность методов диагностики заболеваний с использованием датчиков газового состава.
Ключевые слова: *датчики, газовый состав, диагностирование, анализ, параметры, селективность.*

V. A. Kilimnik*
PhD, Tech.
A. A. Chekmeneva*
Postgraduate Student
**St. Petersburg State University of Aerospace Instrumentation*

ASSESSMENT OF THE COMPOSITION OF EXHALED AIR USING GAS SENSORS

The paper presents a review of the literature on the analysis of exhaled air. The paper reflects the relevance of methods for diagnosing diseases using gas composition sensors.
Keywords: *sensors, gas composition, diagnostics, analysis, parameters, selectivity.*

Один из перспективных методов ранней диагностики заболеваний человека – анализ выдыхаемого воздуха на содержание в нем специфических газов [1–3]. Анализ выдыхаемого воздуха – это способ обнаружения некоторых заболеваний путем исследования определенных соединений в выдыхаемом воздухе, которое в основном состоит из кислорода, углекислого газа, водяного пара и оксида азота, а также менее 100 ppm (частей на миллион) смеси с более 500 наименований составляющих [2], такими как окись углерода, метан, водород, ацетон и многочисленные летучие органические соединения. В табл. 1 приведены заболевания и сопутствующие им газы в выдыхаемом воздухе.

В настоящее время анализ выдыхаемого воздуха обычно выполняется газоаналитическими аппаратами: газовой хроматографией, масс-спектрометрией, электрохимическими датчиками, полупроводниковыми сенсорами и др. Например, с помощью газовой хроматографии [4] можно разделять и идентифицировать молекулы, которые ответственны за возникающие при определенных заболеваниях симптомы [2]. Но такой способ имеет ряд недостатков: техническое средство является дорогостоящими и имеет большие габариты, процессы отбора проб и анализа сложны и требуют много времени.

Определение состава выдыхаемого воздуха с помощью датчиков экономически выгодно и при решении всех проблем с датчиками становится доступным для клинического использования.

Основными критериями при выборе датчиков являются их чувствительность и селективность [5]. Высокая селективность исследуемых веществ требуется, поскольку выдыхаемый воздух – это сложная смесь газов. Применяемый метод должен быть нечувствителен прежде всего к азоту и кислороду, так как их концентрации составляют десятки процентов. Также важна селективность датчиков относительно углекислого газа и паров воды, концентрация которых в выдохе достигает до 3…6% [6], что значительно выше содержания анализируемых газов (на 6…8 порядков). Также при работе необходимо учитывать, что при анализе выдыхаемой пробы относительная влажность может достигать 90…95%.

Таблица 1

Заболевания и сопутствующие им газы в выдыхаемом воздухе

Заболевания	Газ	Концентрация
Цирроз и дисфункция печени; инфицирование бактерией Helicobacter pylori; избыточный рост бактерий; дисфункции поджелудочной железы; метаболизм глюкозы; прохождение пищи через желудочно-кишечный тракт; метаболизм желчи	Углекислый газ (CO_2)	> 4%
Анемии (гемолитическая, сидеробластическая, серповидноклеточная); инфекция дыхательных путей; астма; гематомы, гемоглобинурия, инфекции	Окись углерода (CO)	> 2%
Заболевания органов пищеварения: расстройства желудочно-кишечного тракта, расстройства пищеварения у младенцев, анаэробные бактерии в толстом кишечнике, мальабсорбция углеводородов	Водород	1…30 ppm
Функция поджелудочной железы при остром деструктивном панкреатите и диетическом разбалансе; тяжелая сердечная недостаточность; рак легкого	Ацетон (C3H6O)	1…20 ppm
Недостаточность печени при желтухе, гепатитах, циррозе печени, токсическом гепатите; рак легкого; почечная недостаточность: нефрит, гипертоническая болезнь, атеросклероз почечных артерий, токсикоз, токсические поражения почек	Аммиак (NH3)	> 1 ppm

В настоящее время для анализа газового состава выдыхаемого воздуха используются разнообразные датчики. В табл. 2 приведены методы определения некоторых газов и диапазоны измерения концентрации для использования в диагностике заболеваний.

Таблица 2

Требуемый диапазон концентраций газовых датчиков

Газ	Тип датчика	Диапазон измерения концентрации
Диоксид углерода (CO_2)	Инфракрасный	0…40 000 ppm
Окись углерода (CO)	Электрохимический, полупроводниковый	0…20 000 ppm
Ацетон (C_3H_6O)	Электрохимический, полупроводниковый	1…20 ppm
Водород	Электрохимический, полупроводниковый	0…100 ppm
Аммиак (NH_3)	Электрохимический, полупроводниковый	1…100 ppm

Целесообразно подробно рассмотреть принцип и основные технические характеристики наиболее распространенных датчиков. Представитель электрохимического метода анализа – датчик кислорода отечественного производства Oksik 3 [7] (рис. 1).

Сигнал с датчика пропорционален парциальному давлению кислорода в анализируемой газовой смеси. В основе функционирования электрохимического сенсора лежит амперометрический метод определения концентрации: при поступлении кислорода на поверхность измерительного электрода через поры мембраны, которая выполнена из фторопласта, происходит регенерация газа, подвергающегося анализу (кислорода). В результате отдачи электронов катодом и принятия их анодом – преобразователем кислорода вырабатывается выходной сигнал постоянного напряжения.

Диапазон измерения данного датчика от 0,1 до 30,0% кислорода, а выходной сигнал от 150 до 500 мВ. Зависимость выходного напряжения датчика от содержания кислорода линейна при сопротивлении нагрузки более 2 МОм. Датчик может использоваться при температуре –35...+50 °С, атмосферном давлении: 730... 800 мм рт. ст. и влажности 0...98% отн. при 25 °С (без конденсата). Главное преимущество данного датчика – срок службы более 10 лет.

К преимуществам электрохимических датчиков можно отнести линейный выходной сигнал, высокую точность и хорошую воспроизводимость результатов.

Из представителей полупроводникового метода измерения газового состава выдыхаемого воздуха рассмотрен датчик Figaro TGS8100 [8], который имеет низкое энергопотребление и длительный срок службы, малогабаритный.

Рис. 1. Датчик кислорода Oksik 3

*Рис. 2. Датчик Figaro TGS*8100

*Рис. 3. Датчик Sensirion SCD*30

Датчик состоит из чувствительного чипа и встроенного нагревателя, сформированного на кремниевой подложке с использованием технологии МЭМС, и слоя полупроводника из оксида металла, сформированного на чувствительном чипе (рис. 2). Диапазон обнаружения водорода от 1 до 30 ppm, напряжение нагревателя 1,8 В, потребляемая мощность 15 мВт.

К преимуществам полупроводниковых датчиков можно отнести чувствительность к сверхнизким концентрациям и долговременную стабильность.

Также для газового анализа используется ИК-излучение, принцип работы которого основан на поглощении инфракрасных лучей газом. Находясь в небольшой камере, анализируемый воздух подвергается облучению инфракрасным лучом. Сначала осуществляется измерение интенсивности без оптического устройства. После этого луч, пересекая смесь газов и светофильтр, доходит до датчика, который считывает и фиксирует показания интенсивности принятого луча в диапазоне от 1 до 15 мкм. После определения двух значений прибор определяет концентрацию углекислого газа в воздухе по их разнице. В данный модуль встроен датчик температуры и влажности. Благодаря двухканальному принципу измерения концентрации диоксида углерода датчик автоматически компенсирует долгосрочную девиацию.

Для обнаружения углекислого газа указанный метод широко используется. Наиболее доступным является датчик Sensirion SCD30 (рис. 3) [9]. Диапазон измерения газа составляет от 0 до 40 000 ppm, с точностью ±30 ppm.

Анализ состава выдыхаемого воздуха все активнее используется как диагностический инструмент для определения различных заболеваний. Актуальна разработка анализирующих устройств на основе датчиков, но основная сложность возникает в том, что определение конкретного заболевания часто ведется не по одному газу, а по нескольким.

Библиографический список

1. *Копылов Ф. Ю.* Перспективы диагностики различных заболеваний по составу выдыхаемого // Клиническая медицина. 2013. № 10. С. 16–21.

2. *Cao W., Duan Y.* Current Status of Methods and Techniques for Breath Analysis // Critical Reviews in Analytical Chemistry. 2007. Vol 37. P. 3–13.

3. *Risby T. H., Solga S.* Current Status of Clinical Breath Analysis // Applied Physics B. 2006. Vol 85. P. 421–426.

4. *Степанов Е. В.* Методы высокочувствительного газового анализа молекул-биомаркеров в исследованиях выдыхаемого воздуха // Труды института общей физики им. А. М. Прохорова. 2005. Т. 61. С. 1–47.

5. *Агейкин А. В., Пронин И. А.* Диагностика заболеваний желудочно-кишечного тракта человека по выдыхаемому воздуху с помощью массива полупроводниковых газовых сенсоров // Молодой ученый. 2014. № 12 (71). С. 383–384.

6. *Лукаш С. И.* Проблемы диагностики некоторых заболеваний // Комп'ютерні засоби, мережі та системи. 2010. № 9. С. 62–71.

7. Датчик кислорода oksik 3. URL: http://oxonsens.ru/oksik-3 (дата обращения: 25.11.21).

8. Датчик Figaro TGS8100. URL: https://www.sensor-test.de/ausstellerbereich/upload/mnpdf/en/tgs8100_product_infomation_14.pdf (дата обращения: 25.11.21).

9. Датчик Sensirion SCD30. URL: https://www.sensirion.com/en/environmental-sensors/carbon-dioxide-sensors/carbon-dioxide-sensors-scd30/ (дата обращения: 25.11.21).

УДК 621.38 **DOI: 10.31799/978-5-8088-1701-2-2022-2-65-70**

В. А. Килимник*
кандидат технических наук
Д. С. Шамрицкая*
магистрант
О. С. Медведев**
доктор медицинских наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*
***Московский государственный университет им. М. В. Ломоносова*

МАКЕТ СИСТЕМЫ ДЛЯ АВТОМАТИЧЕСКОГО УЧЕТА ПОВЕДЕНИЯ ЛАБОРАТОРНЫХ ЖИВОТНЫХ

Рассмотрена структура комплексного макета устройства, позволяющего вести учет поведения лабораторных животных. Разработан алгоритм работы программы устройства.
Ключевые слова: *RFID, лабораторное животное, датчик, макет.*

V. A. Kilimnik*
PhD, Tech.
D. S. Shamritskaya*
Postgraduate Student
O. S. Medvedev**
Dr. Sc. Med., Professor
**St. Petersburg State University of Aerospace Instrumentation*
***Lomonosov Moscow State University*

MODEL OF A SYSTEM FOR AUTOMATIC RECORDING OF THE BEHAVIOR OF LABORATORY ANIMALS

The structure of a complex model of a device that allows keeping records of the behavior of laboratory animals is considered. An algorithm for the operation of the device program has been developed.
Keywords: *RFID, laboratory animal, sensor, model.*

В современном мире человечество с каждым годом все больше нуждается в качественных, безопасных фармакологических и косметических препаратах, а также бытовой и промышленной химии. Ни для кого не секрет, что перед выходом на рынок любые химические вещества проходят обязательные тесты, в первую очередь на лабораторных животных.

В лабораториях и научных институтах повсеместно в качестве подопытных животных многие годы используются мыши и крысы. Ввиду своих небольших размеров, хорошо развитого интеллекта, социального поведения и небольшого срока жизни эти животные крайне легки в изучении и тестировании. Однако грызуны очень схожи между собой внешне, что приводит к затруднениям в идентификации особей между собой, особенно если исследование требуется проводить на большой группе животных.

В современной научно-исследовательской деятельности существуют различные способы маркировки лабораторных животных. В большинстве случаев эти методы заключаются в физическом нанесении отличительных знаков на тело подопытного животного.

Примерами обычной маркировки крыс и мышей могут служить следующие способы.

Татуировка.

Некоторые части тела у грызунов, такие как уши, хвост, подошва и подушечки пальцев, могут быть использованы для маркировки при помощи нанесения татуировки. Чернила для татуировок изготавливаются из минеральных пигментов на органической основе, поскольку вредные или ядовитые чернила могут вызвать у животных аллергические реакции и ухудшение здоровья [1]. Хотя татуировка и является постоянным методом маркировки, с течением времени она может поблекнуть или быть по-

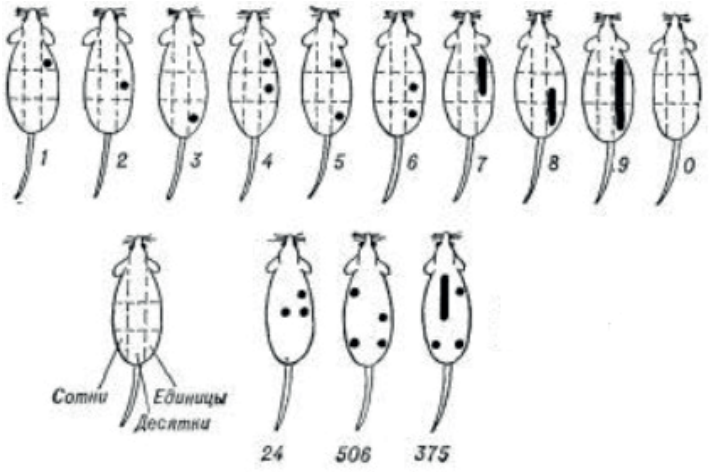


Рис. 1. Маркировка с помощью краски

вреждена. Также для ее нанесения необходим специально обученный сотрудник.

Подкожное введение чернил.

Подкожная инъекция чернил отличается от татуировки тем, что чернила вводят под кожу, а не в ее слои. Поскольку все вещества, введенные подкожно, постепенно всасываются, чернила исчезают через некоторое время (в зависимости от пигмента и места нанесения это может произойти от нескольких часов до нескольких дней). Также данный способ имеет крайне ограниченную возможность нумерации особей (разные места нанесения и цвета). Эта процедура состоит из двух болезненных компонентов – инъекция и раздражение после введения вещества [1].

Нанесение краски на шерсть.

Помимо подкожного введения чернил, краску могут наносить прямо на шерсть животных. Даже самые устойчивые краски для шерсти удерживаются не более чем в течение двух–трех месяцев. Саму краску наносят в виде точек или полосок на спине и по бокам животного. Также не все краски хорошо сказываются на здоровье животных, поскольку все равно всасываются сквозь кожу и попадают в организм. Способ нанесения меток на шерсть изображен на рис. 1 [2].

Помимо перечисленных методов, используют такие способы маркировки, как нанесение ушных бирок, выстригание и выжигание частей шерсти, ампутация фаланг пальцев, перфорация ушей и др.

Как видно из приведенных примеров, почти все эти способы маркировки имеют существенные недостатки, главные из которых:

1) недолговечность – в зависимости от метода маркировки она прослужит от нескольких дней до нескольких месяцев, ушные бирки могут быть потеряны или оторваны во время драки;

2) неточность и ненадежность – физические метки могут затираться или исчезать со временем, в результате чего неверно прочитаны;

3) нанесение вреда здоровью подопытным животным – поскольку краски легко впитываются через кожу, они могут попасть в кровоток грызуна, что может привести к ухудшению физического состояния и неточным данным исследований;

4) времязатратность – для нанесения меток требуется очень много времени, при идентификации особи необходимо поймать ее для детального рассмотрения метки, также метки необходимо регулярно обновлять, чтобы не потерять результаты исследований.

Поскольку рассмотренные недостатки классических методов могут достаточно критично отразиться на результатах проводимых научных исследований, их дальнейшее использование в научных целях малорационально в сравнении с новыми технологичными решениями рассматриваемой проблемы.

В данной научно-исследовательской работе будет описан и применен один из самых современных и точных способов маркировки лабораторных животных – RFID-микрочипирование.

В указанном методе используются электронные радиочастотные транспондеры (микрочипы). Они вводятся животному подкожно, как правило, в область холки. На рис. 2 изображено примерное место инъекции чипа [3]. Данные считываются с чипа с помощью антенны-считывателя. Каждый чип имеет собственный уникальный код, благодаря чему можно идентифицировать бесконечное число чипированных жи-

вотных. Считыватели подключаются к компьютеру, что позволяет собирать статистику и сразу обрабатывать ее или записывать в базу данных.

В отличие от классических методов маркировки, чип невозможно неправильно считать, что исключает ошибки в идентификации. Кроме того, в зависимости от мощности приемника, необязательно физически контактировать с чипированным животным, достаточно поднести считыватель на расстояние нескольких сантиметров от холки крысы, чтобы получить данные с чипа. Процедура чипирования проводится один раз, дополнительных обновлений метки не требуется. Чипы практически не ломаются, а потерять их физически невозможно.

К минусам микрочипов можно отнести их высокую стоимость относительно классических методов маркировки, а также нежелательное чипирование маленьких и новорожденных особей, поскольку это может быть крайне болезненно для них. Взрослая особь никакого дискомфорта от чипа не испытывает.

Основные задачи данной научно-исследовательской работы – создание схемы будущего устройства для автоматического учета поведения лабораторных животных и предварительный подбор компонентов для его дальнейшей реализации потенциальному потребителю.

Разрабатываемое устройство должно автоматически определять индивидуальный номер крысы, которая подошла к кормушке, а также вычислять время, которое она провела у нее во время питания, после чего эти данные должны быть переданы на компьютер.

При разработке макета устройства следует учитывать особенности крысиного поведения, которые могут привести к неточным данным или поломке устройства. Поскольку крысы относятся к грызунам, все провода и платы должны быть надежно защищены и вынесены за пределы клеток во избежание порчи компонентов животными. Для этого будет создан специальный металлический кожух, который будет защищать оборудование.

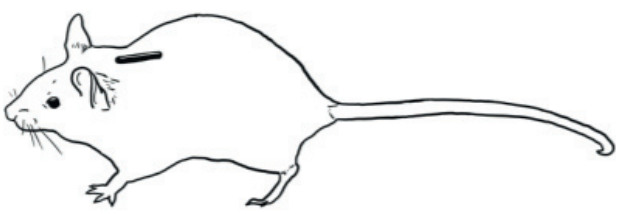


Рис. 2. Примерное расположение чипа под кожей

В клетке будет находиться большое количество крыс, поэтому антенна RFID-приемника не должна считывать данные любой близко подошедшей к ней особи. Для этого, возможно, внутри кожуха будет помещен инфракрасный датчик препятствий, который будет активировать считыватель только в момент, когда крыса забирается внутрь кожуха, чтобы попить. На рис. 3 схематично изображено устройство будущего прибора: 1 – резервуар с водой или кормом, 2 – инфракрасный датчик, 3 – обмотка RFID-антенны, 4 – трубка для подачи корма или воды, 5 – металлический кожух.

На рис. 4 показано схематическое применение устройства.

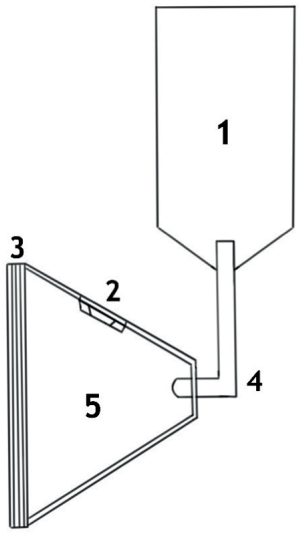


Рис. 3. Структурная схема макета устройства

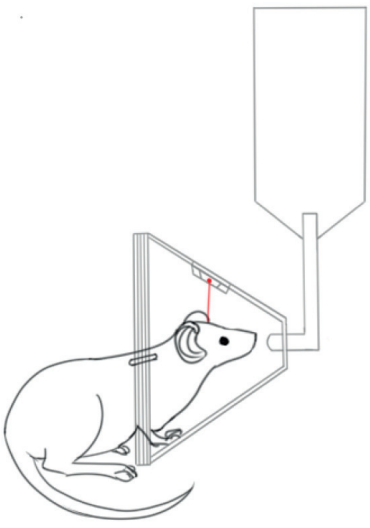


Рис. 4. Схематическое применение устройства

RFID-датчик и чип

Технология радиочастотной идентификации основана на передаче данных при помощи электромагнитных полей (по радио). Чип, называемый ответчиком или меткой, хранит на себе уникальный порядковый номер идентификации. RFID-метка состоит из антенны и чипа, содержащего индивидуальные данные объекта [4].

RFID-метки, используемые для чипирования животных, являются низкочастотными ответчиками и работают в диапазоне 120–134 кГц. Они являются пассивными ответчиками, т. е. получают питание от электромагнитного поля считывающего устройства, им не требуется собственный источник питания, что делает эти ответчики очень маленькими и экономичными. Как правило, их корпус изготавливается из стекла, что также делает их устойчивыми к коррозии. На рис. 5 представлены размеры RFID-меток в сравнении со спичкой [5].

Для реализации RFID-считывателя будет использован один из приемников EM4305 с антенной. Данный датчик имеет несколько моделей, различающихся расстоянием считывания RFID-метки, а также размером (рис. 6, 7).

Инфракрасный датчик

Для определения нахождения крысы внутри поилки предполагается использовать инфракрасный датчик препятствий. Цифровой инфракрасный датчик обхода препятствий применяется тогда, когда нужно определить наличие объекта, а точное расстояние до объекта знать необязательно [6]. Наиболее распространены на рынке на данный момент датчики KY-032 (рис. 8) и YL-63 (рис. 9).

Рис. 5. Размеры RFID-меток

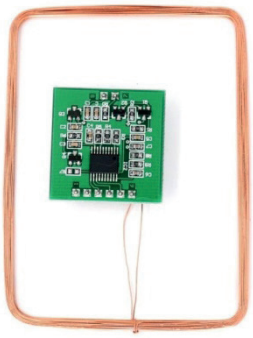


Рис. 6. Приемник EM4305 26×26 мм

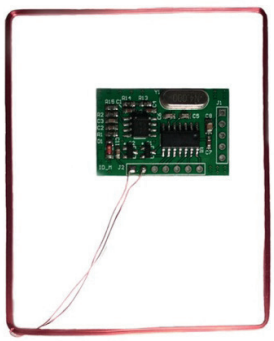


Рис. 7. Приемник EM4305 60×75 мм

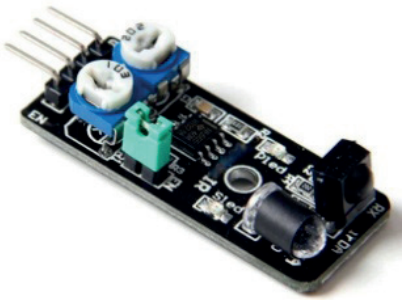


Рис. 8. Датчик KY-032

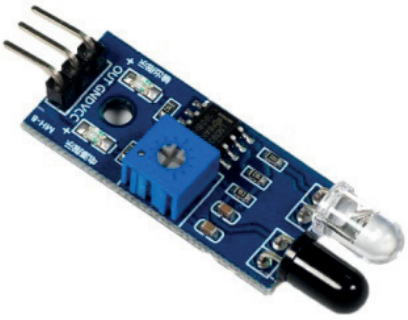


Рис. 9. Датчик YL-63

Датчик времени

Одно из наиболее оптимальных решений для отслеживания времени подхода крыс к кормушке – датчик DS3231 (рис. 10). Это модуль часов реального времени RTC со встроенной батарейкой на случай потери питания датчиком [7]. В отличие от многих других датчи-

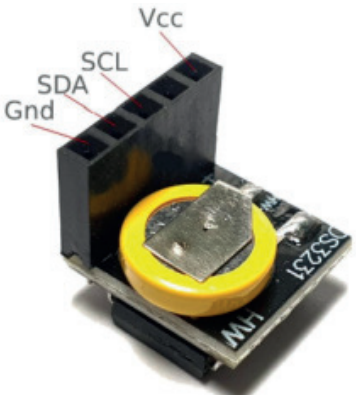


Рис. 10. Датчик DS3231

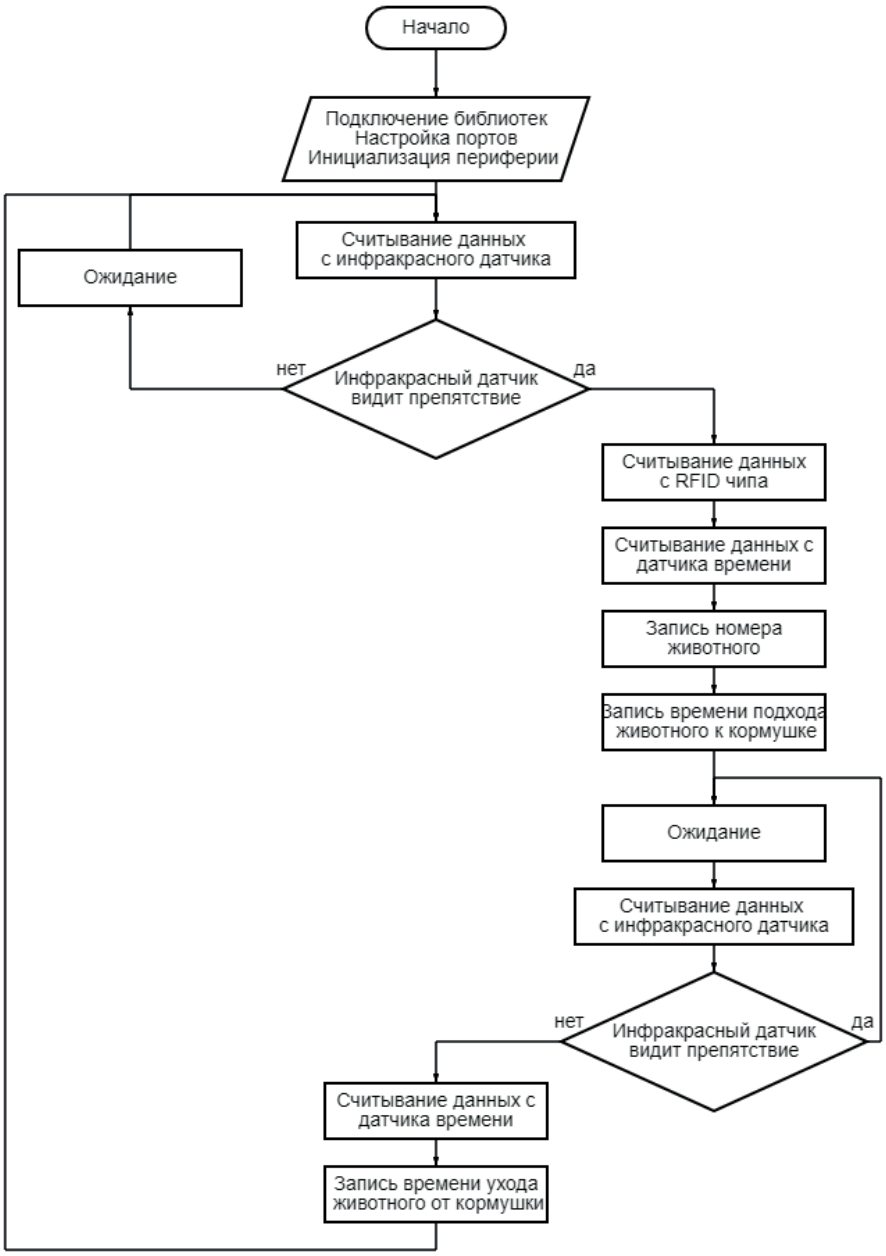


Рис. 11. Алгоритм работы программы устройства

ков времени, он имеет гораздо меньшие габариты, но не имеет встроенной памяти. Поэтому для записи времени и индивидуального кода животного придется использовать внешнюю память.

Для дальнейшей реализации основной программы устройства был составлен алгоритм (рис. 11).

В данной статье были рассмотрены основные вопросы, возникающие при реализации системы автоматического учета поведения лабораторных животных. Исходя из них, были подобраны предварительные компоненты для создания устройства, а также составлен алгоритм программы.

Библиографический список

1. Маркировка и идентификация лабораторных животных для проведения научно-исследовательских работ. URL: https://vivarium-nnz.ru/markirovka-i-identifikatsiya-laboratornyh-zhivotnyh/ (дата обращения: 15.11.21).

2. Способы мечения лабораторных животных. URL: http://handcent.ru/laboratornye-zhivotnye/261-sposoby-mecheniya-laboratornyh-zhivotnyh-chast-1.html (дата обращения: 16.11.21).

3. O mouse, where art thou? The Mouse Position Surveillance System (MoPSS)—an RFID-based tracking system. URL: file:///C:/Users/user/Downloads/2021-Habedank-OMouseWhereArtThouTheMousePosi%20(1).pdf (дата обращения: 13.11.21).

4. Радиочастотная идентификация (RFID) Руководство по выбору. URL: https://kb-raskat.ru/upload/medialibrary/8a5/8a5b563e7048911c9a89e320177c4d91.pdf (дата обращения: 20.11.21).

5. Стеклянные метки для животных. URL: https://www.hidglobal.ru/products/rfid-tags/identification-technologies/glass-tags-animals (дата обращения: 20.11.21).

6. Инфракрасный датчик препятствий YL-63. URL: https://3d-diy.ru/wiki/arduino-datchiki/infrakrasnyj-datchik-prepyatstvij-yl-63/ (дата обращения: 18.11.21).

7. Часы реального времени DS3231 mini. URL: https://www.mini-tech.com.ua/rtc-modul-ds3231-mini (дата обращения: 19.11.21).

УДК 621.396.96 **DOI: 10.31799/978-5-8088-1701-2-2022-2-71-73**

В. С. Павлов*
доктор технических наук, профессор
Е. Л. Турнецкая*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

СПЕЦИФИКА МОНОИМПУЛЬСНОЙ НОРМИРОВКИ ПРОСТРАНСТВЕННО-ТРЕХКАНАЛЬНЫХ ОЦЕНОК УГЛОВЫХ КООРДИНАТ ЛОКАЦИОННОГО ОБЪЕКТА

Рассматривается специфика нормировки оценок двух угловых координат локационного объекта, формируемых на основе пространственно-трехканального метода пеленгации. Предложена процедура комбинированной моноимпульсной нормировки, реализуемая относительно суммы мощностей выходных сигналов трех каналов приема и трех откликов перекрестного фазового детектирования данных сигналов. Технический результат предложенной нормировки состоит в значительном расширении телесного угла пеленгации локационного объекта.
Ключевые слова: *моноимпульсная нормировка, пеленгация, радиолокация, трехканальный.*

V. S. Pavlov*
Dr. Sc., Tech., Professor
E. L. Turnetskaya*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

MONOPULSE NORMALIZATION SPECIFICS OF THREE-SPATIAL-CHANNEL ESTIMATIONS OF RADAR OBJECT ANGULAR COORDINATES

Normalization specifics is studied for two angular coordinates estimations of radar object those are evaluated on the basis of three-spatial-channel method of direction-finding. The combined monopulse normalization procedure is proposed which is fulfilled relatively the power sum of output signals of three receiving channels and the triple response of cross phase detection of these signals. Technical result of suggested normalization is consisted in the significant spatial angle extension of radar object direction-finding.
Keywords: *direction-finding, monopulse normalization, radar, three-channel.*

Научно-прикладная разработка новых технических решений, направленных на повышение точности и надежности пеленгации локационного объекта (ЛО) методами моноимпульсной радиолокации [1], относится к числу актуальных задач совершенствования радиолокационных систем различного назначения.

Одним из таких новых технических решений является пеленгационное устройство [2], реализующее пространственно-трехканальный метод совместной оценки двух угловых координат ЛО в ортогональных плоскостях пеленгации [3]. Новизне данного пеленгационного устройства сопутствуют дополнительные вопросы его реализации, в частности вопросы нормировки формируемых оценок двух угловых координат ЛО.

Цель работы состоит в обосновании моноимпульсной нормировки для пространственно-трехканального метода пеленгации ЛО, при которой обеспечивается неискаженное совместное формирование оценок двух угловых координат ЛО в пределах широкой области их значений.

Следуя цели работы, опишем пространственно-временной сигнал на n-м выходе ($n = 0, 1, 2$) трехканального приемного тракта [2], характеризуя направление на ЛО в сферической системе координат через его широту θ и долготу φ:

$$y_n(\theta,\varphi,t) = s(t)G_n(\theta,\varphi)\times \\ \times\exp\left(j\mu\sin\theta\cos\left(\gamma_n-\varphi\right)\right), \quad (1)$$

где $s(t)$ – временной процесс принимаемого сигнала ЛО; $G_n(\theta, \varphi)$ – диаграмма направленности n-го канала приема; j – мнимая единица; μ =

$2\pi r/\lambda$ – показатель пеленгационной чувствительности (r – радиус окружности, проходящей через три эквивалентные точки приема [2–4]; λ – длина волны); $\gamma_n = 2\pi n/3 + \gamma_*$ – полярный угол положения n-й эквивалентной точки приема на плоскости приема, γ_* – угловое смещение совокупности трех эквивалентных точек приема.

В результате квадратичного и перекрестного фазового детектирования [2–4] трех выходных сигналов трехканального приемного тракта формируются шесть откликов, которые опишем совместно:

$$u_{mn}(\theta,\varphi) = \mathrm{M}\left\{y_n(\theta,\varphi,t)\, y^*_{(m+n)\bmod 3}(\theta,\varphi,t)\right\}, \qquad (2)$$

где $m = 0$ – при квадратичном детектировании каждого из трех принимаемых сигналов и $m = 1$ – при перекрестном фазовом детектировании данных сигналов; $\mathrm{M}\{\cdot\}$ – оператор усреднения на интервале временной обработки; $(\cdot)^*$ – знак комплексного сопряжения.

Согласно [2–4], пространственно-трехканальная оценка угла θ в отдельной плоскости пеленгации ЛО определяется следующим образом:

$$z(\theta,\varphi) = K\sum_n a_n(\nu)\,\mathrm{Im}\left\{u_{1n}(\theta,\varphi)\right\} \Big/ Q(\theta,\varphi), \qquad (3)$$

где $K = 2/\left(\sqrt{3}\mu\right)$ – коэффициент, обеспечивающий единичную крутизну преобразования угла θ в его оценку; $a_n(\nu) = \cos(2\pi n/3 - \pi/6 + \gamma_* - \nu)$ – n-й коэффициент весового суммирования [4], ν – параметр, задающий одну из двух ортогональных плоскостей пеленгации ЛО и принимающий одно из двух значений: $\nu = 0$ или $\nu = \pi/2$; $\mathrm{Im}\{\cdot\}$ – оператор выделения мнимой части комплексного аргумента; $Q(\theta, \varphi)$ – нормирующая составляющая оценки $z(\theta, \varphi)$.

Отметим, что нормирующая составляющая процедуры оптимального дискриминатора обобщенного неэнергетического параметра, вычисляемая через вторую производную корреляционного интеграла (эквивалента функции правдоподобия), обладает рядом недостатков [5]. В этой связи рекомендуется к применению модифицированный (подоптимальный) вариант нормировки относительно самого корреляционного интеграла [5]. Кроме того, возможен упрощенный способ нормировки оценок угловых координат ЛО, при котором нормирующая составляющая формируется за счет суммирования мощностей откликов трех каналов приема.

Объединяя два указанных способа нормировки оценок угловых координат ЛО, нетрудно составить комбинированную нормирующую составляющую в виде

$$Q(\theta,\varphi) = w\sum_n u_{0n}(\theta,\varphi) + (1-w)\sum_n \mathrm{Re}\left\{u_{1n}(\theta,\varphi)\right\}, \qquad (4)$$

где w — весовой коэффициент, уточняемый исходя из критериев ширины раствора и линейности пеленгационной характеристики; $\mathrm{Re}\{\cdot\}$ – оператор выделения действительной части комплексного аргумента.

Отметим, что область однозначности оценки угла θ не выходит за пределы главного лепестка диаграммы направленности антенны и можно приближенно полагать $G_0(\theta, \varphi) \cong G_1(\theta, \varphi) \cong G_2(\theta, \varphi)$. Следовательно, в нормирующей процедуре (3) сокращаются сомножители диаграмм направленности антенны и сомножитель мощности принимаемого сигнала ЛО (когда влияние внутриприемного шума пренебрежимо мало).

Учитывая изложенные допущения, нетрудно провести расчет пеленгационных характеристик по формулам (1)–(4). В силу центральной симметрии данных характеристик [4] ограничимся иллюстрацией только их правых ветвей, которые изображены на рис. 1 при $\varphi = 0$ и различных вариантах нормировки. По оси абсцисс на рис. 1 отложена нормированная угловая координата ЛО $\Theta = \mu\sin\theta \cong \mu\theta$, а по оси ординат – нормированные значения пеленгационной характеристики $\Pi(\Pi\mu\theta, 0) = \mu z(\theta, 0)$.

Из рис. 1 видно, что широкая область линейности пеленгационной характеристики обеспечивается при $w \cong 0{,}4$. Увеличение и уменьшение коэффициента w относительно данного значе-

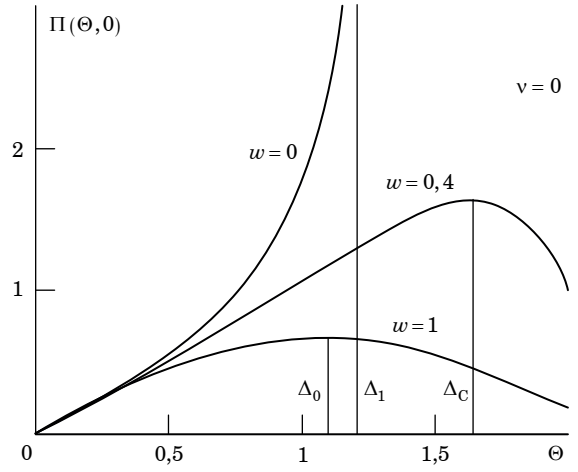


Рис. 1. Пеленгационные характеристики при различных значениях коэффициента w

ния приводит соответственно к заниженным (при $w \cong 1$) и завышенным (при $w \cong 0$) оценкам угла Θ в области $\Theta > 0{,}5$, что иллюстрируют кривые, отражающие соответственно упрощенный и модифицированный типы нормировки. Полуширина раствора пеленгационной характеристики Δ определяется параметром w (типом нормировки) и равна: $\Delta_0 \cong 1{,}08$ при $w = 1$, $\Delta_1 \cong 1{,}15$ при $w = 0$ и $\Delta_C \cong 1{,}7$ при $w = 0{,}4$.

Предложенная в работе комбинированная нормировка оценок угловых координат ЛО, формируемых на основе пространственно-трехканального метода пеленгации, позволяет значительно расширить угол однозначности результатов пеленгации в каждой из двух ортогональных плоскостей. Это расширение составляет примерно 57% относительно варианта нормировки по суммарной мощности выходных сигналов трехканального приемного тракта и примерно 48% – относительно модифицированного (подоптимального) варианта нормировки, реализуемого на основе перекрестного фазового детектирования данных сигналов. Широкая угловая область линейности пеленгационных характеристик достигается за счет возможности параметрической настройки комбинированной нормировки оценок угловых координат ЛО.

Библиографический список

1. *Леонов А. И., Фомичев К. И.* Моноимпульсная радиолокация. Фомичев. М.: Радио и связь, 1984. 312 с.

2. Патент. 2364882 Российская Федерация, МПК G01S 3/14. Моноимпульсный фазовый пеленгатор / Г. В. Анцев, В. С. Павлов, Л. С. Турнецкий, А. Д. Французов. № 2007117465/09; заявл. 10.05.07; опубл. 20.08.09. Бюл. № 23.

3. *Павлов В. С.* Точность трехотсчетной фазовой процедуры измерения направления на локационный объект // Радиотехника. 2000. № 12. С. 3–10.

4. *Павлов В. С.* Характеристики многоотсчетных чувствительных элементов локационных систем измерения угловых координат // Изв. вузов. Приборостроение. 2003. Т. 46, № 1. С. 16–21.

5. *Коростелев А. А.* Пространственно-временная теория радиосистем. М.: Радио и связь, 1987. 320 с.

УДК 621.3.087.44 **DOI: 10.31799/978-5-8088-1701-2-2022-2-74-78**

А. В. Сорокин*
старший преподаватель
А. С. Раскопина, Н. И. Мирошниченко, А. С. Волкова*
студенты
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ВЕРОЯТНОСТНЫЕ ХАРАКТЕРИСТИКИ ОПРОСНЫХ УСТРОЙСТВ В СИСТЕМАХ ДИАГНОСТИКИ НЕИСПРАВНОСТЕЙ СЕТЕЙ ЭЛЕКТРОСНАБЖЕНИЯ

В современном мире все чаще используются автоматизированные высокотехнологические производства, где большое внимание уделяют контролю качества изготавливаемых технических систем. Для того чтобы предотвратить простои из-за сбоя производства, требуются системы, которые могут автоматически передавать информацию о процессе производства в пункт управления. Реализация такой системы чаще всего происходит при помощи радиочастотных акустоэлектронных меток, которые обладают высокой устойчивостью к повышенной температуре, влажности и давлению, что позволяет надежно определять индивидуальные номера меток. Но система РЧИД (радиочастотных идентификационных) меток не имеет широкого применения из-за возникающих на производствах помех, что приводит к невозможности корректного определения кода метки, а следовательно, и к невозможности идентификации самого изделия [1–3].
Ключевые слова: *электроэнергетика, энергоэффективность, система мониторинга.*

A. V. Sorokin*
Senior Lecturer
A. S. Raskopina, N. I. Miroshnichenko, A. S. Volkova*
Students
**St. Petersburg State University of Aerospace Instrumentation*

PROBABILITY CHARACTERISTICS OF INTERROGATORS IN SYSTEMS FOR DIAGNOSING FAULTS IN POWER SUPPLY NETWORKS

In the modern world, automated high-tech industries are increasingly used, where they pay more attention to quality control of manufactured technical systems. To prevent downtime due to production disruptions, systems are required that can automatically transfer information about the production process to the control room. The implementation of such a system most often occurs with the help of radio-frequency acoustoelectronic tags, which are highly resistant to high temperature, humidity and pressure, which makes it possible to reliably determine the individual tag numbers. But the RFID (Radio Frequency Identification) tag system is not widely used due to the interference arising in production, which leads to the impossibility of correctly determining the tag code, and, consequently, to the impossibility of identifying the product itself [1–3].
Keywords: *electric power industry, energy efficiency, monitoring system.*

Введение

Электроэнергетическое оборудование – сложное техническое устройство, которое подвержено риску возникновения различных аварийных ситуаций. Одна из причин, которая приводит к таким рискам, – износ состояния контактных соединений. Чаще всего ускоряет износ контактных соединений их перегрев. Цепи токоведущих элементов электрических цепей имеют высокую нагрузку по току, вследствие чего возникает риск сильного нагрева изоляции и переходных контактов электрических цепей, а также механических частей.

Неисправность контактных соединений можно обнаружить при помощи технического осмотра шины, а также осмотра тепловизором и при помощи различных термоиндикаторов. Первые два варианта предусматривают работу специалиста, который занимается ремонтом электроэнергетического оборудования, что не позволяет автоматизировать данную часть производства и затратно финансово.

Для контроля перегрева электрооборудования вследствие автоматизации обнаружения

неисправностей предлагается внедрение системы мониторинга состояния. На данный момент существуют три вида таких систем:

– пирометрическая – позволяет производить мониторинг температуры шины;

– термосенсорная – позволяет производить мониторинг температуры контактных соединений;

– радиоканальная – позволяет производить мониторинг электрооборудования.

Наиболее перспективно и целесообразно рассматривать систему на базе пассивных акустоэлектронных датчиков на поверхностных акустических волнах, которые обладают высокой устойчивостью к повышенной температуре, влажности и давлению, что позволяет надежно определять индивидуальные номера меток, с функцией радиочастотной идентификацией [4–8].

Вероятностные характеристики опросных устройств

Проблема внедрения упомянутой системы состоит в том, что при определении уникальных идентификационных номеров РЧИД-меток опросное приемо-передающее устройство излучает опросный радиосигнал, который принимается антенной метки и преобразуется в ответный радиосигнал, излучаемый меткой, содержащий информацию о коде метки и об измеряемой физической величине. В случае когда в зону опроса попадают сильные помехи, ответные радиосигналы перекрываются во времени, что приводит к невозможности корректного определения кода метки [9].

Наиболее подходящими обнаружителями в данном случае являются цифровые обнаружители типа «k из n». Они устойчивы к помехам, на их работу не влияют различного рода выбросы, и их не сложно реализовать в цифровом виде.

Работа обнаружителя состоит в том, что на его вход подается радиосигнал с приемного устройства и после детектора огибающей видеосигнал поступает на вход аналогового компаратора, на другой вход которого подается аналоговый порог срабатывания X_0. После этого на выходе компаратора должна сформироваться последовательность единиц и нулей. Накопитель должен произвести суммирование n сигналов, после чего накопленная сумма k будет сравниваться в цифровом компараторе с цифровым порогом k_0. Сигнал обнаружения «единица» будет выдаваться, только когда накопленная сумма k будет больше цифрового порога k_0, иначе на выходе цифрового компаратора формируется сигнал, который говорит, что ничего не обнаружено [10–12].

Пусть двоичная последовательность кода метки содержит m позиций. Для достоверной идентификации позиций происходит обмен n последовательностями, т. е. n повторов одной и той же последовательности.

Тогда при идентификации каждой позиции с использованием двух обнаружителей типа «k из n» вероятность правильной идентификации $P_{\text{пи}}$ определяется выражением $P_{\text{пи}} = P_{\text{по}}(1 - P_{\text{лт}})$, где $P_{\text{по}}$ – вероятность правильного обнаружения и $P_{\text{лт}}$ – вероятность ложной тревоги. Вероятность правильной идентификации кода метки определяется как $P_{m\text{пи}} = P_{\text{по}}{}^{m}(1 - P_{\text{лт}})^{m}$. Из этого выражения следует, что высокая вероятность $P_{m\text{пи}}$ может быть обеспечена только при высокой вероятности $P_{\text{по}}$ и малой вероятности $P_{\text{лт}}$, особенно это относится к «длинным» кодам, т. е. к большим значениям m.

В свою очередь значения вероятностей $P_{\text{по}}$ и $P_{\text{лт}}$ определяются величиной принимаемой пачки n и значениями аналогового X_0 и цифрового k_0 порогов обнаружителя, а пороги зависят и выбираются в соответствии с принятыми моделями информационного сигнала и помехи. Рассмотрим два типа помех – нормальный белый шум и негауссову помеху – индустриальный шум [13–16].

Первый тип – традиционный белый шум – помеха, принимаемая по умолчанию в подавляющем большинстве случаев исследования характеристик обнаружителей. К этому типу относятся и собственные шумы приемников радиоэлектронных устройств. После детектора огибающей распределение этой помехи подчиняется распределению Рэлея:

$$f_R(x) = \frac{x}{\sigma_R^2}\exp\left(-\frac{x^2}{2\sigma_R^2}\right),\ x > 0, \qquad (1)$$

где $\sigma_R^2 = \tilde{P}_R$ – средняя мощность помехи.

Второй тип – индустриальная помеха. Распределение огибающей этой помехи чаще всего аппроксимируется логарифмически-нормальным распределением:

$$f_L(x) = \frac{1}{\sqrt{2\pi}\sigma_L x}\exp\left(-\frac{\left(\ln(x) - \ln\left(\bar{A}_L\right)\right)^2}{2\sigma_L^2}\right), \qquad (2)$$

$$x > 0,$$

где $\bar{A}_L$ и σ_L – параметры распределения, через которые средняя мощность радиоимпульсов помехи выражается как $\tilde{P}_L = 0.5\bar{A}_L^2\exp(2\sigma_L^2)$ [17].

Для информационного сигнала принимаем модель стабильного сигнала:

$$f_{\text{с}}(x) = \delta(x - A_{\text{с}}), \qquad (3)$$

где мощность радиоимпульсов информационного сигнала $\tilde{P}_{\text{с}} = 0{,}5\bar{A}_{\text{с}}^2$ [18].

Вероятности правильного обнаружения $P_{\text{по}}$ и ложной тревоги $P_{\text{лт}}$ при фиксированной длине пачки n определяются последовательностями бинарных сигналов на выходе компаратора:

$$P_{\text{лт}} = \sum_{k=k_0+1}^{n} C_n^k P_{\text{ш}}^k (1 - P_{\text{ш}})^{n-k}; \qquad (4)$$

$$P_{no} = \sum_{k=k_0+1}^{n} C_n^k P_{\text{с}}^k (1 - P_{\text{с}})^{n-k}, \qquad (5)$$

где $P_{\text{ш}}$ и $P_{\text{с}}$ – вероятности формирования сигнала «1» на выходе аналогового компаратора, т. е. вероятности превышения аналогового порога X_0 при наличии на входе компаратора (выходе детектора) только помехи и аддитивной смеси информационного сигнала и помехи, соответственно [19].

Вероятности $P_{\text{ш}}$ и $P_{\text{с}}$ определяются выражениями

$$P_{\text{ш}} = \int_{X_0}^{\infty} f_{\text{ш}}(U) \cdot dU; \qquad (6)$$

$$P_{\text{с}} = \int_{X_0}^{\infty} f_{\text{с+ш}}(U) \cdot dU, \qquad (7)$$

где $f_{\text{ш}}(U)$ и $f_{\text{с+ш}}(U)$ – плотности распределения огибающих помехи и векторной суммы информационного сигнала и помехи соответственно [20].

При вычислении $P_{\text{ш}}$ в формулу (6) подставляется $f_{\text{ш}}(U) = f_R(U)$ Рэлея, определенная выражением (1), или $f_{\text{ш}}(U) = f_L(U)$, определенная выражением (2). При практическом использовании приведенных выражений задаются значением $P_{\text{ш}}$ и по нему определяют требуемый порог аналогового компаратора X_0. Затем рассчитывают порог цифрового компаратора по приближенному выражению $k_0 = E\left(\sqrt{1{,}5n} - 0{,}5\right)$, где E – функция Антье, и по выражению (4) рассчитывают ложную тревогу $P_{\text{лт}}$ [21–22].

Обозначая пороги аналогового компаратора для помехи Рэлея и логарифмически-нормальной помехи через U_{0R} и U_{0L}, вычисляя соответствующие зависимости $X_{0R} = \varphi_R(P_{\text{ш}})$ и $X_{0L} = \varphi_L(P_{\text{ш}})$, получаем

$$X_{0R} = \varphi_R(P_{\text{ш}}) = \sigma_R\sqrt{-2\ln P_{\text{ш}}}; \qquad (8)$$

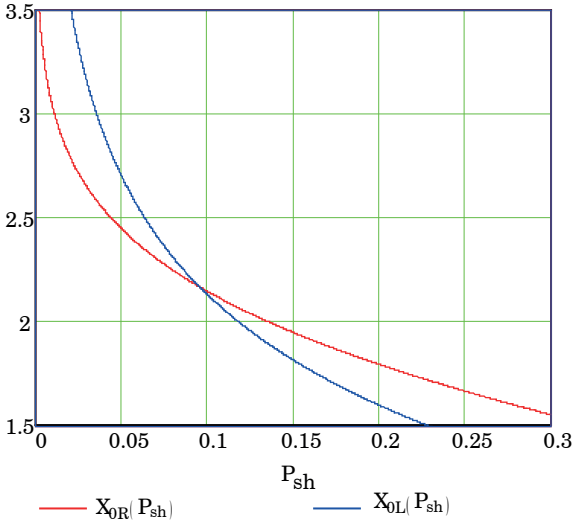


Рис. 1. Зависимость порогов аналогового компаратора от величины $P_{\text{ш}}$

$$X_{0L} = \varphi_L(P_{\text{ш}}) = \sqrt{2}\sigma_R \times$$
$$\times \exp\left(\sqrt{\frac{4-\pi}{2}} \cdot \left(\Phi^{-1}(1 - P_{\text{ш}}) - \sqrt{\frac{4-\pi}{2}}\right)\right), \qquad (9)$$

где Φ^{-1} – функция обратная интегралу Лапласа. При вычислении выражения (9) дополнительно было использовано условие равенства математических ожиданий и дисперсий рэлеевской и логарифмически-нормальной помех. Графики зависимостей $X_{0R} = \varphi_R(P_{\text{ш}})$ и $X_{0L} = \varphi_L(P_{\text{ш}})$ приведены на рис. 1.

Кривые пересекаются при $P_{\text{ш}} = \bar{P}_{\text{ш}}$, являющемся корнем уравнения:

$$\exp(2\sigma_L(\Phi^{-1}(1 - P_{\text{ш}}) - \sigma_L)) + \ln P_{\text{ш}} = 0, \qquad (10)$$

численное решение которого дает значение $\bar{P}_{\text{ш}} = 0{,}096$, при этом порог равен $\bar{X}_{0R} = \bar{X}_{0L} = \bar{X}_0 = 2{,}166$. Значения порогов X_{0R} и X_{0L} выражены в единицах σ_R, которую мы положили равной единице, поэтому при практическом расчете порогов по выражениям (8) и (9) необходимо учитывать численное значение σ_R, которое определяется средней мощностью помехи равной σ_R^2.

При выборе $\bar{P}_{\text{ш}} = 0{,}096$ вероятность ложной тревоги обнаружителей не зависит от распределения помехи. Этот же вывод относится и к аддитивной смеси Рэлеевской и логарифмически-нормальной помехи, определяемой как составное распределение модели Хьюбера:

$$f_{\text{ш}}(x) = (1 - \gamma)f_R(x) + \gamma f_L(x), \qquad (11)$$

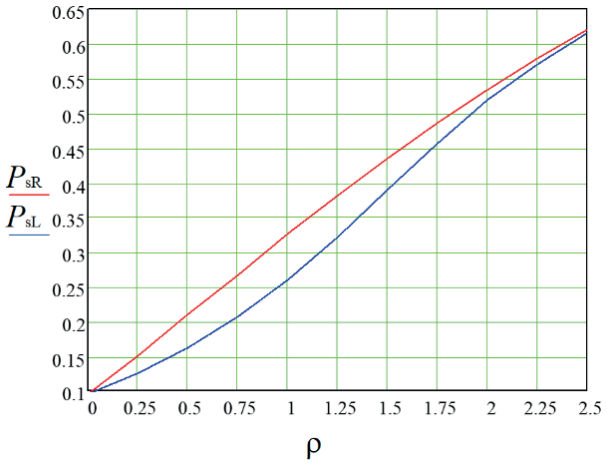


Рис. 2. Зависимость вероятности превышения смешанного полезного сигнала и помехи порога компаратора X_0 от соотношения сигнал/шум ρ

где γ – весовой коэффициент, при $\gamma = 0$ получаем Рэлеевскую помеху, а при $\gamma = 1$ – логарифмически-нормальную. При выборе порога аналогового компаратора $\bar{X}_{0R} = \bar{X}_{0L} = \bar{X}_0 = 2{,}166$ ложные тревоги обнаружителя не зависят от семейства распределений, определенных в модели Хьюбера (11). Все кривые семейства (11) при $0 < \gamma < 1$ находятся между крайними кривыми, приведенными на рис.1.

Для оценки воздействия типа помехи на вероятность правильного обнаружения необходимо вычислить вероятность превышения смесью сигнала и помехи порога X_0, т. е. найти зависимости $P_s = P_s(\rho)$, где ρ – отношение сигнал/помеха по мощности. В нашем случае при равенстве мощностей логарифмически-нормальной и рэлеевской помех $\rho = \dfrac{A_s^2}{2\sigma_R^2}$, а поскольку мы все выражаем в единицах σ, положив $\sigma = 1$, получим $\rho = A_s^2 / 2$, откуда $A_s^2 = \sqrt{2\rho}$.

После детектора амплитуда векторной суммы информационного и помехового радиоимпульсов

$$A_{s+noise} = \sqrt{A_s^2 + A_{noise}^2 - 2A_s A_{noise}\cos\varphi}, \quad (12)$$

где A_s – амплитуда информационного радиоимпульса, A_{noise} – амплитуда помехового сигнала, φ – сдвиг фаз между векторами информационного и помехового сигналов, распределенный равномерно в интервале (–π, +π). В соответствии с математической моделью стабильного сигнала (3) амплитуда A_s детерминированная величина, а амплитуда импульса помехи в соответствии с принятыми моделями распределена по закону Хьбера (11). Выражение (12) является алгоритмической записью математической модели векторной суммы информационного сигнала и помехи.

Для рэлеевской помехи известно аналитическое выражение плотности распределения A_{cR}, поэтому зависимость $P_{cR} = P_{cR}(\rho)$ может быть получена путем численного интегрирования этой плотности на интервале (–π, +π), а зависимость $P_{cL} = P_{cL}(\rho)$ получена методом имитационного моделирования по алгоритмической модели (12) [23]. Эти зависимости приведены на рис. 2.

Кривые, соответствующие модели Хьюбера для $0 < \gamma < 1$, находятся между кривыми, приведенными на рис. 2.

Из хода кривых видно, что при одной и той же мощности индустриальные помехи, распределение которых аппроксимировано логарифмически-нормальной плотностью, приводят к потерям в отношении сигнал/помеха по отношению к нормальным помехам, традиционно используемым для расчета характеристик качества функционирования опросно-считывающих устройств. Характеристики обнаружения и правильной идентификации индивидуальных кодов РЧИД-меток при использовании кривых, приведенных на рис. 2, могут быть рассчитаны аналитически по выражениям (4) и (5). Здесь эти расчеты не приводятся, поскольку результаты, приведенные на рис. 2, являются достаточными для проведения расчетов в разнообразных ситуациях считывания индивидуальных кодов РЧИД-меток.

Заключение

Система РЧИД – перспективная технология, особенно если рассматривать ее на базе пассивных акустоэлектронных датчиков на поверхностных акустических волнах. Когда в зону опроса попадают сильные помехи, ответные радиосигналы перекрываются во времени, что приводит к невозможности корректного определения кода метки. Поэтому при реализации данной системы стоит использовать частотно-временной метод с целью защиты от индустриальных помех и блок обнаружителей цифровых сигналов типа «*k* из *n*», которые обладают робастными свойствами. Использование блока обнаружителей позволяет контролировать вероятность неправильной идентификации кода метки даже при отсутствии в ответном сигнале контрольной суммы, что чрезвычайно важно при работе в сложной помеховой обстановке.

Библиографический список

1. *Sorokin A. V., Shepeta A. P., Nenashev V. A., Wattimena G. M.* Comparative characteristics of anti-collision processing of radio signal from identification tags on surface acoustic waves // Information and Control Systems. 2019. № 1. P. 49–57.

2. Патент № 2333513. Система радиочастотной идентификации против столкновений / Багдасарян С. А., Багдасарян А. С., Гуляев Ю. В., Карапетян Г. Я., Нефедов Н. А., Никитов С. А., Николаев В. И., Николаев О. В. Опубл.: 30.10.2006.

3. *Sorokin A. V., Shepeta A. P.* Time-Frequency Approach to Anti-Collision Signal Processing for RFID SAW Tags // Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). St. Petersburg, 2019. P. 1–4.

4. *Сорокин А. В, Шепета А. П., Подоплекин Ю. Ф.* Антиколлизионная пассивная радиочастотная метка с частотно-временным информационными признаками // Морская радиоэлектроника. 2018. № 2 (64). С. 24–29.

5. *Sorokin A., Shepeta A., Wattimena M.* Encoding of Passive Anticollision Radio Frequency Identification Surface Acoustic Waves Tags // Proceeding of the Electrical Engineering Computer Science and Informatics. 2017. Vol. 4. P. 605–609.

6. *Hartmann C. S., Brown P., Bellamy J.* Design of global SAW RFID tag devices // Second International Symposium on Acoustic Wave Devices for Future Mobile Communication Systems, Chiba University, Japan, 2004. P. 15–19.

7. *Plessky V. P., Reindl L. M.* Review on SAW RFID tags // Proc. IEEE Trans Ultrason Ferroelectr Freq Control. 2010. Vol. 57 (3). P. 654–668.

8. *Hartmann C. S.* A global SAW ID tag with large data capacity // Proc. IEEE International Ultrasonics Symposium, 2002. P. 65–69.

9. *Chen D. P., Haus H. A.* Analysis of metal-strip SAW gratings and transducers // IEEE Transactions on Sonics and Ultrasonics. 1985. Vol. Su-32, № 3. P. 395–408.

10. *Steindi R., Hausleitner C., Hauser H., Bulst W.* Wireless Magnetic Field Sensor Employing SAW-Transponder // Proceedings of the 12th IEEE International Symposium on Applications of Ferroelectrics (ISAF 2000); Honolulu, HI, USA. 21 July – 2 August 2000. P. 855–858.

11. *Malocha D. C., Gallagher M., Fisher B., Humphries J., Gallagher D., Kozlovski N.* Passive Wireless Multi-Sensor SAW Technology Device and System Perspective // Sensors. 2013. Vol. 13. P. 1–27.

12. *Койгеров А. С., Дмитриев В. Ф.* Радиомаркированная поверхностная акустическая волна с корректирующим ошибки частотно-манипулированным кодом // Информационно-управляющие системы. 2010. № 4. С. 22–28.

13. *Sorokin A. V., Shepeta A. P.* Anti-collision radio-frequency identification system using passive SAW tags // Proc. SPIE, Smart Sensors, Actuators, and MEMS VIII, Barcelona, Spain. 2017. Vol. 10246. P. 1024613.

14. *Sorokin A., Shepeta A., Wattimena M.* Wireless SAW passive tag temperature measurement in the collision case // Proc. EECSI 2017, Yogyakarta, Indonesia, 2017. Vol. 1008 (1). P. 012015.

15. *Harma S., Arthur W. G., Hartmann C. S., Maev R. G., Plessky V. P.* Inline SAW RFID Tag Using Time Position and Phase Encoding // Proceeding IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control. 2008. Vol. 55 (8). P. 145–158.

16. *Danicki E.* Reversing multistrip coupler // Ultrasonics. 1993. Vol. 31(6). P. 421–424.

17. *Morgan D. P.* Surface Acoustic Wave Filters 2nd Edition. With Applications to Electronic Communications and Signal Processing. 2007. 448 p.

18. *Malocha D. C., Gallagher D., Hines J.* SAW Sensors Using Orthogonal Frequency Coding // Proceedings of the 2004 IEEE International Frequency Control Symposium and Exposition, Montreal, Canada, 24–27 August 2004. P. 307–310.

19. *Puccio D., Malocha D. C., Saldanh N.* Implementation of orthogonal frequency coded SAW devices using apodized reflectors // Proc. IEEE International Frequency Control Symposium, 2005.

20. *Shepeta A. P., Makhlin A. M., Nenashev V. A., Kryachko A. F.* Performance of UWB Signal Detecting Circuits // Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2018. P. 1–4.

21. *Блаунштейн Н. Ш., Сергеев М. Б., Шепета А. П.* Прикладные аспекты электродинамики. СПб.: Аграф+, 2016. 272 с.

22. *Подоплекин Ю. Ф., Шепета А. П., Махлин А. М., Каплин А. Ю.* Цифровые обнаружители сверхширокополосных импульсных сигналов. Цифровые детекторы сверхширокополосных импульсов // Морской вестник. 2016. № 58. С. 77–81.

23. *Исаков В. И., Шепета Д. А.* Моделирование локальных сигналов, отраженных от кромки земля – море. Моделирование сигналов определения местоположения отраженный от края суши–моря // Информационные и управляющие системы. 2017. № 5. С. 89–94.

УДК 004.832.34 **DOI: 10.31799/978-5-8088-1701-2-2022-2-79-82**

С. Г. Толмачев
кандидат технических наук, доцент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ПРОЦЕДУРА КЛАССИФИКАЦИИ ОБЪЕКТОВ НА ОСНОВЕ СОГЛАСОВАНИЯ ЧАСТНЫХ РЕШЕНИЙ

Показана возможность использования процедуры комбинирования данных, полученных от различных информационных каналов, для решения задачи автоматической классификации обнаруженных объектов на фоне естественных и искусственных помех.
Ключевые слова: *автономные системы, информационные каналы, классификация, мера доверия.*

S. G. Tolmachev
PhD, Tech., Associate Professor
St. Petersburg State University of Aerospace Instrumentation

THE PROCEDURE FOR CLASSIFYING OBJECTS BASED ON THE COORDINATION OF PARTICULAR SOLUTIONS

The possibility of using the procedure of combining data obtained from various information channels to solve the problem of automatic classification of detected objects against the background of natural and artificial interference is shown.
Keywords: *autonomous systems, information channels, classification, confidence measure.*

Одна из современных тенденций развития технических систем, построенных на основе беспилотных, безэкипажных средств, – повышение уровня их автономности. Особенно ярко эта тенденция проявляется в области разработки систем вооружения. В США в ноябре 2014 г. была официально представлена третья национальная Стратегия противодействия военным угрозам. Этот документ делает основной акцент на инвестиции в развитие искусственного интеллекта (ИИ) и разработку автономных систем вооружения [1]. При этом отмечается, что если ранее разработка военных технологий финансировалась в основном Министерством обороны, то теперь ключевые технологии разрабатываются по инициативе коммерческих организаций. По сути, автономные системы – это системы, которые могут самостоятельно составлять и выбирать альтернативные варианты действий для достижения поставленных целей на основе заложенных знаний, понимания среды функционирования и динамики развития ситуации.

Разработка автономных систем сопряжена со значительными техническими трудностями. Чтобы успешно выполнять свои задачи, автономные системы должны быть способны воспринимать обстановку, обнаруживать цели, идентифицировать их, планировать свои действия, принимать решения и реагировать на разнообразные угрозы в сложных и неопределенных условиях. Данные об окружающей обстановке автономные системы получают через информационные каналы – оптические, телевизионные, радиолокационные. В настоящее время в большинстве технологически развитых стран большое внимание уделяется созданию многоканальных систем, в которых используются одновременно активные и пассивные средства освещения обстановки. При этом одной из самых больших проблем, связанных с созданием многоканальных систем, специалисты считают задачу комплексирования данных от разных информационных каналов [2] таким образом, чтобы при их использовании повышалась достоверность результатов классификации наблюдаемых объектов. Решение о принадлежности обнаруженного объекта к одному из известных классов осуществляется по совокупности наблюдаемых признаков. Каждый информационный канал имеет возможность определять свойственный ему перечень признаков, на основании которых формирует решающие правила классификации. Результи-

рующее решение выносится путем согласования частных решений.

Наряду со способами классификации объектов по частными признакам, существует подход, который заключается в использовании всей совокупности их отличительных признаков. В рамках современных интеллектуальных технологий для этого применяются нейросетевые методы. Этот подход предусматривает формирование полного вектора признаков, выбор нейросетевой модели и метода ее машинного обучения с использованием обучающих и тестовых примеров. В результате формируется интеллектуальный фильтр, на вход которого поступает сложная картина наблюдаемой обстановки, а на выходе остаются только реальные объекты с указанием их класса.

Известно [3], что в военно-воздушных силах США прогресс в развитии компьютерной архитектуры интеллектуальных бортовых систем автономных беспилотных летательных аппаратов (БПЛА) связывают с нейроморфными вычислениями на базе искусственных нейронных сетей (ИНС). При выполнении операций с ИНС такие архитектуры демонстрируют более высокую скорость получения результата, способность к самообучению и имеют исключительно малое энергопотребление. Планируется, что аппаратно-программные средства систем с искусственным интеллектом (ИИ) позволят США в среднесрочной перспективе задействовать БПЛА для организации полностью автономных разведывательно-ударных контуров, способных без участия человека выполнять задачи поиска, обнаружения и огневого поражения объектов противника. Существенным недостатком нейросетевых методов является необходимость формирования представительного массива достоверных обучающих данных, в виде образцов сигналов, полученных от реальных объектов, естественных и искусственных помех, что далеко не всегда возможно.

Поэтому актуальной остается задача классификации наблюдаемых объектов по результатам оценки отдельных признаков с последующим объединением частных решений. Частные решения о принадлежности каждого обнаруженного объекта к одному из известных классов могут быть как взаимно согласованными, так и противоречивыми. Поэтому необходима методика, позволяющая принять комплексное решение с учетом всех полученных частных оценок и условий, в которых они состоялись. Эта задача решается в рамках общего подхода, известного под названием коллективного распознавания или объединения классификаторов (classifier fusion) [4]. Под объединением классификаторов понимается задача использования множества частных классификаторов, каждый из которых принимает решение о классе одного и того же объекта, с последующим объединением и согласованием решений отдельных информационных каналов с помощью комбинирующего алгоритма.

Для объединения решений традиционно применяются вероятностные методы. Необходимое условие их использования для объединения решений состоит в том, что все информационные каналы должны иметь однородные выходные данные, несмотря на разнородность исходных обрабатываемых данных. Выходные данные должны быть векторами с мерами неопределенности одного типа – «мягкими метками» классов, характеризующими степень уверенности в принятом решении.

Введем следующие обозначения:

n – количество реальных и ложных классов объектов, различаемых информационными каналами (классификаторами);

x_i – вектор измерений, формирующий пространство признаков i-го информационного канала. Пространства признаков различных классификаторов могут быть различными, одинаковыми или пересекаться;

$D(x_i)$ – решение i-го классификатора, D – имя класса;

M – общее число информационных каналов, решения которых объединяются.

Предполагается, что каждый информационный канал формирует выходную информацию в виде вектора решений размерности n с мерой неопределенности вида:

$$\mu = <\mu_{i1}(x_i), \mu_{i2}(x_i), ..., \mu_{ij}(x_i), ..., \mu_{in}(x_i) >,$$

где i – индекс классификатора, j – порядковый номер класса объекта, $\mu_{ij} \in [0,1]$.

Каждая компонента этого вектора $\mu_{ij}(x_i)$ ($j = 1, 2,\ldots n$) отражает степень уверенности о принадлежности объекта соответствующему классу. По мере получения решений (свидетельств) от каждого i-го классификатора изменяется и степень доверия к итогам классификации в результате применения правила комбинирования свидетельств. Величины $\mu_{ij}(x_i)$ формируются в каждом информационном канале с помощью решающих правил. Все объекты среды разделены на ряд классов, которые в свою очередь образуют конечное универсальное множество E. Это множество состоит из имен этих классов $\{D_1, D_2, \ldots, D_n\}$. Степенным множеством $P(E)$ множества E называется множество, все элементы которого представляют собой подмноже-

Таблица 1

Расчет комбинированных масс

Степень доверия m	$m_2(\{БК\}) = 0{,}9$	$m_2(E) = 0{,}1$
$m_1(\{БК, ДЗ\}) = 0{,}7$	$Z = \{БК\}; m_{12} = 0{,}63$	$Z = \{БК, ДЗ\}; m_{12} = 0{,}07$
$m_1(E) = 0{,}3$	$Z = \{БК\}; m_{12} = 0{,}27$	$Z = E; m_{12} = 0{,}03$

ства множества E. Множество с количеством элементов n имеет 2^n подмножеств, включая самого себя и пустое множество ∅.

Обозначим степень доверия к свидетельству – m. В рамках методологии Демпстера – Шефера [5] степень доверия к свидетельству принято называть массой, а решение $D(x_i)$ – свидетельством. Масса присваивается по результатам частного решения i-го классификатора только тем классам, которые отвечают условиям принадлежности к этим классам.

Например, БПЛА, оснащенный оптической системой наблюдения, а также активным радаром и пассивным радиопеленгатором, получает информацию об обнаруженных объектах по трем информационным каналам. Допустим, что в результате оценки геометрических размеров объекта со степенью доверия 0,7 можно заключить, что это большой корабль (БК) или искусственная ложная цель типа дымовой завесы (ДЗ). Тогда присваивание массы подмножеству {БК, ДЗ} осуществляется по следующему правилу, в котором m_1 обозначает первое свидетельство: $m_1(\{БК, ДЗ\}) = 0{,}7$. Остальная часть степени доверия присваивается среде E как мера отсутствия знания: $m_1(E) = 1 - 0{,}7 = 0{,}3$ и не влечет за собой присваивание какого-либо значения другим подмножествам множества E. Каждое подмножество в степенном множестве среды, имеющее массу $m_i > 0$, называется фокальным элементом. Причем принимается соглашение, что масса пустого множества равна нулю: $m(\varnothing) = 0$, а сумма масс всех подмножеств степенного множества равна 1.

Допустим, что активный радиолокатор по другим информационным признакам обнаруженный объект относит к классу БК со степенью доверия 0,9. Тогда $m_2(\{БК\}) = 0{,}9$ и $m_2(E) = 0{,}1$. Полученные свидетельства комбинируются с помощью правила Демпстера для получения комбинированной массы:

$$m_{12} = m_1 \oplus m_2(Z) = \sum_{Z=X\cap Y} m_1(X)m_2(Y).$$

Это правило может применяться для комбинирования свидетельств, имеющих взаимно независимые ошибки. Операция суммирования распространяется на все элементы, для которых существует пересечение подмножеств X и Y степенного множества E. Расчет комбинированных масс для рассматриваемого примера приведен в табл. 1.

Для каждого фокального элемента вычисляется итоговая масса:

$$m_{12}(\{БК\}) = 0{,}63 + 0{,}27 = 0{,}9;$$

$$m_{12}(\{БК, ДЗ\}) = 0{,}07; m_{12}(E) = 0{,}03.$$

Значение $m_{12}(\{БК\})$ выражает доверие к тому, что рассматриваемый объект представляет собой только большой корабль. Но значения $m_{12}(\{БК, ДЗ\})$ и $m_{12}(E)$ содержат дополнительную информацию, так как эти множества также содержат объект класса БК. Таким образом, степень доверия не ограничивается одним значением, а выражается в виде ряда степеней доверия к свидетельству. В данном случае ряд степеней доверия начинается с минимального значения 0,9, согласно которому известно, что рассматриваемый объект – большой корабль, до максимального правдоподобного значения степени доверия, равного 0,9 + 0,1 = 1, что этот объект может представлять собой большой корабль. При этом предполагается, что истинная степень доверия находится где-то в диапазоне от 0,9 до 1. Нижняя граница интервала называется обоснованием и обозначается как *Bel* (*belief*). Верхнюю границу принято называть правдоподобием *Pls* (*plausibility*).

Обоснование представляет собой минимальную степень доверия, основанную на свидетельстве, а правдоподобие – максимальную степень доверия, которую можно достичь. Диапазоны, в которых изменяются *Bel* и *Pls*, выражаются соотношением $0 \le Bel \le Pls \le 1$.

Указанный подход позволяет комбинировать конфликтующие свидетельства. Допустим, что третий информационный канал – пассивный радиолокационный пеленгатор на основании энергетических признаков принимаемого сигнала относит рассматриваемый объект к классу искусственных импульсных помех (ИП) со степенью доверия $m_3(\{ИП\}) = 0{,}8$, тогда $m_3(E) = 0{,}2$. Вычисление перекрестных произведений и комбинирование масс трех свидетельств приведено в табл. 2.

В этой таблице из-за третьего конфликтующего свидетельства появляются пустые подмножества, так как {БК, ДЗ} и {ИП} не имеют

Таблица 2

Вычисление перекрестных произведений и комбинирование масс трех свидетельств

Степень доверия m	$m_{12}(\{БК\}) = 0,9$	$m_{12}(\{БК, ДЗ\}) = 0,07$	$m_{12}(E) = 0,03$
$m_3(\{ИП\}) = 0,8$	$Z = \varnothing$; $m_{123} = 0$	$Z = \varnothing$; $m_{123} = 0$	$Z = \{ИП\}$; $m_{123} = 0,024$
$m_3(E) = 0,2$	$Z = \{БК\}$; $m_{123} = 0,18$	$Z = \{БК, ДЗ\}$; $m_{123} = 0,014$	$Z = E$; $m_{123} = 0,006$

общих элементов, масса их перекрестного произведения равна 0 по определению $m(\varnothing) = 0$, из-за чего сумма всех масс меньше единицы:

$$\sum m_{123} = 0,024 + 0,18 + 0,014 + 0,006 = 0,224.$$

Этот факт говорит о наличии конфликта свидетельств. Решение заключается в нормализации масс фокальных элементов путем деления на величину $\sum m_{123}$: $m_{123}(\{ИП\}) = 0,107$; $m_{123}(\{БК\}) = 0,803$; $m_{123}(\{БК, ДЗ\}) = 0,063$; $m_{123}(E) = 0,027$. Тогда степень доверия к гипотезе о том, что наблюдаемый объект является большим кораблем, понижается до величины $Bel(\{БК\}) = 0,803$, а правдоподобие до величины $Pls(\{БК\}) = 0,803 + 0,063 + 0,027 = 0,893$. Окончательное решение о принадлежности наблюдаемого объекта к одному из известных классов осуществляется по максимальному значению обоснования *Bel*.

Правило комбинирования результатов свидетельств позволяет получать итоговый результат путем последовательного комбинирования частных свидетельств, причем порядок их обработки не влияет на конечный результат, так как правило Демпстера обладает свойствами коммутативности и ассоциативности. Таким образом, реализация процедуры не зависит от числа информационных каналов и количества комбинируемых частных решений.

Библиографический список

1. *Ilachinski A.* Artificial Intelligence & Autonomy // CNA AI Report, 2017. URL: https://www.cna.org/CNA_files/PDF/DRM-2017-U-014796-Final.pdf (дата обращения: 09.11.2021).

2. *Анцев Г. В., Сарычев В. А., Жигулин Г. П.* Попытка оценить перспективы развития высокоточного оружия // Пленарные доклады XXI Всероссийской научно-практической конференции «Актуальные проблемы защиты и безопасности». СПб., 2018. С. 114–121.

3. *Carolin Y. V.* US Air Force research lab tabs IBM to build brain-inspired AI supercomputing system. 2017. URL: https://newsroom.ibm.com/2017-06-23-U-S-Air-Force-Research-Lab-Tabs-IBM-To-Build-Brain-Inspired-AI-Supercomputing-System (дата обращения: 10.11.2021).

4. *Городецкий В. И., Серебряков С. В.* Методы и алгоритмы коллективного распознавания // Труды СПИИРАН. 2006. Вып. 3, т. 1. С. 139–171.

5. *Shafer G.* A Mathematical Theory of Evidence. NJ: Princeton Univ. Press. 1976. 297 p.

УДК 621.396 **DOI: 10.31799/978-5-8088-1701-2-2022-2-83-86**

В. А. Тюринова*
магистрант
А. П. Шепета*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОЦЕНКА K-РАВНОМЕРНОСТИ ДАТЧИКОВ ПСЕВДОСЛУЧАЙНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Один из основных методов исследования сложных систем на ЭВМ – метод математического моделирования. При его практическом применении необходим источник энтропии, в качестве которого используются генераторы псевдослучайных последовательностей. Особенность сложных систем – наличие большого числа случайных параметров, которые формируются из подпоследовательностей генератора. В силу этого необходима тщательная проверка статистических свойств подпоследовательностей, а сам генератор должен иметь K-равномерность, превосходящую число параметров сложной системы. Оценке качества генерации подпоследовательноси генераторов псевдослучайных чисел при больших K и посвящена данная работа.
Ключевые слова: *сложная система, математическое моделирование, псевдослучайная последовательность, K-равномерность, подпоследовательность.*

V. A. Tyurinova*
Postgraduate Student
A. P. Shepeta*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

ESTIMATION OF K-UNIFORMITY OF PSEUDORANDOM SEQUENCE SENSORS

One of the main methods of studying complex systems on a computer is the method of mathematical modeling. In its practical application, an entropy source is needed, as which pseudo-random sequence generators are used. A feature of complex systems is the presence of a large number of random parameters that are formed from the generator's subsequences. Because of this, a thorough check of the statistical properties of the subsequences is necessary, and the generator itself must have K-uniformity exceeding the number of parameters of a complex system. This paper is devoted to evaluating the quality of generating subsequences of pseudorandom number generators at large K.
Keywords: *complex system, mathematical modeling, pseudorandom sequence, the subsequence.*

Введение

Метод математического моделирования широко применяется при исследовании сложных систем. Часто он оказывается единственным методом анализа и проектирования сложных систем, которые не только характеризуются многочисленными нелинейными связями между подсистемами, но и являются многоканальными и нестационарными. Эти особенности не позволяют получать характеристики качества исследуемых и проектируемых систем аналитическими методами. Однако имитационные модели алгоритма функционирования системы и ее входных воздействий дают возможность не только анализировать поведение системы и получать оценки характеристик ее качества, но и прогнозировать поведение системы и будущие ее состояния. Использование метода имитационного моделирования при анализе и проектировании сложных систем имеет некоторые аспекты, которые не всегда учитываются аналитиками. В работе и рассмотрен один из таких аспектов – требование K-равномерности датчика псевдослучайных чисел, используемого для моделирования.

Особенности моделирования сложных систем

Использование метода математического моделирования сложных систем, как уже отмечено, имеет особенности, которые не всегда учитываются. В частности, при моделировании

функционирования сложной системы необходимо использовать большое количество случайных параметров как самой системы, так и ее входных воздействий. Моделируемые параметры должны иметь заданные статистические характеристики, которые в большинстве случаев формируются при использовании генератора псевдослучайных чисел, равномерно распределенных в интервале (0, 1).

Формирование параметров происходит при последовательном обращении к генератору псевдослучайных чисел, а это приводит к тому, что последовательности параметров формируются как подпоследовательности используемого генератора. В этом случае такой параметр генератора, как K-равномерность, должен быть больше количества извлекаемых подпоследовательностей, так как только при этом условии подпоследовательности будут иметь те же статистические характеристики, что и исходный «хороший» генератор.

Количество параметров M, которые необходимо моделировать при одном сценарии моделирования, у сложных систем может достигать несколько десятков, сотен и даже тысяч. Это приводит к необходимости использовать генератор, у которого $K > M$, а обычные (не специализированные) встроенные в математические пакеты генераторы могут и не удовлетворять этому условию. Проверка K-равномерности для больших K наталкивается на определенные трудности, при этом желательно проверить не только порядок K-равномерности, но и качество генерации хотя бы отдельных подпоследовательностей из числа M.

Показатели качества генераторов псевдослучайных чисел

Качество генераторов псевдослучайных последовательностей оценивается по многим показателям, следовательно, эта оценка многокритериальная. Вектор показателей качества содержит более двух десятков показателей [1]. Наиболее жесткие требования к генераторам псевдослучайных чисел предъявляют при их использовании в криптографии. В случае когда генераторы применяют как источник энтропии при исследовании сложных систем, требования не столь жесткие. Показатели качества и их численные значения зависят также от целей исследования.

При практическом моделировании можно ограничиться проверкой четырех показателей качества: достаточно большой период сгенерированных последовательностей; генерируемые числа должны быть независимы в совокупности; все сгенерированные числа должны иметь равномерное распределение на отрезке (0, 1); значение показателя K-равномерности должно быть больше числа извлекаемых подпоследовательностей.

Первые три перечисленных показателя присутствуют и в требованиях криптографии, а четвертый для криптографии несущественен, но имеет большое значение при имитационном моделировании сложных систем. При практическом использовании генераторов псевдослучайных последовательностей достаточно ограничиться проверкой этих четырех показателей качества, уделив больше внимания проверке K-равномерности при больших значениях K, так как именно этот показатель особенно важен при имитационном моделировании сложных систем.

Виртуальная матрица псевдослучайных чисел

Для удобства описания алгоритмов проверки показателей исследуемого качества генератора псевдослучайных чисел, распределенных равномерно в интервале (0, 1), введем виртуальную матрицу. Под этим понятием будем понимать матрицу

$$\mathbf{r}_{M,N} = \begin{Vmatrix} r_{1,1} & r_{1,2} & r_{1,3} & \cdots & r_{1,N} \\ r_{2,1} & r_{2,2} & r_{2,3} & \cdots & r_{2,N} \\ r_{3,1} & r_{3,2} & r_{3,3} & \cdots & r_{3,N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{M,1} & r_{M,2} & r_{M,3} & \cdots & r_{M,N} \end{Vmatrix}, \quad (1)$$

в которой M – количество подпоследовательностей, извлекаемых из исследуемого генератора, N – количество «проигруемых» сценариев моделирования системы, а элементами матрицы $\mathbf{r}_{M,N}$ являются сгенерированные числа. Матрица заполняется по столбцам: сначала меняется строчный индекс $i = 1, 2,...M$ при фиксированном индексе j, потом меняется индекс столбца j и заполняется новый столбец, $j = 1, 2,...N$.

Последовательность заполнения матрицы принципиальна. Введенная матрица необходима лишь для удобства рассмотрения используемых тестов, поэтому и названа виртуальной, так как при реальной реализации тестов на ЭВМ матрица $\mathbf{r}_{M,N}$ необязательно должна полностью находиться в памяти компьютера. Кроме того, удобство именно такого заполнения матрицы $\mathbf{r}_{M,N}$ заключается еще и в том, что столбцы матрицы являются теми векторами,

которые необходимы для формирования одного сценария работы исследуемой системы.

Перечисленные тесты будем проверять, используя элементы виртуальной матрицы $\mathbf{r}_{M,N}$, при этом естественно, что N не фиксируется и может возрастать.

Тест проверки периода повторения

При проверке периода повторения необходимо последовательно просматривать элементы матрицы $\mathbf{r}_{M,N}$ так, будто она представлена одномерным массивом, причем просмотр необходимо осуществлять последовательно по столбцам, следуя алгоритму заполнения матрицы. В нашем случае просмотр $\mathbf{r}_{M,N}$ осуществляется по алгоритму изменения индекса k:

$$r_k = r_{i+(j-1)\cdot M} = r_{i,j},\ k = i + (j-1)\cdot M,$$
$$i = 1,2,\ldots M,\ j = 1,2,\ldots N, \qquad (2)$$

причем при вычислении индекса k сначала меняется индекс i, а потом при каждом i индекс j принимает значения $j = 1,2,\ldots N$.

Такой алгоритм просмотра элементов матрицы $\mathbf{r}_{M,N}$ соответствует последовательному просмотру псевдослучайных чисел, сгенерированных датчиком, при этом N можно заранее не фиксировать. При исследовании генераторов псевдослучайных чисел в криптографии анализируется именно эта последовательность чисел [2, 3]. Наш тест проверки периода генератора является обычным стандартным тестом для определения периодичности. Однако он может быть намного проще известных тестов, используемых в криптографии. В частности, он может быть реализован следующим образом.

При первом обращении к датчику псевдослучайных чисел запоминается число $r_1 = r_{1,1}$, затем последовательно просматриваются генерируемые датчиком числа – элементы матрицы $\mathbf{r}_{M,N}$ в соответствии с выражением (2). Если мы просмотрели все элементы матрицы $\mathbf{r}_{M,N}$ и при этом не было совпадений с элементом r_1, то дальше можно не генерировать числа и не обращаться к датчику. Период мы в этом случае не определяем, но убеждаемся в том, что он достаточно большой, и генератор по этому критерию нам подходит.

Тесты проверки K-равномерности

Количество чисел до первого совпадения с числом r_1, равное индексу совпавшего числа k_0, и определяет период генератора T_0: $T_0 = k_0 - 1$, но только в том случае, если при генерации очередного числа используется только одно предыдущее число.

В более общем случае, когда при генерации очередного числа используется несколько предыдущих чисел, необязательно генерируемых подряд, мы определим не период генератора, а лишь установим факт, что истинный период T_0 больше определенного этим алгоритмом числа $k_0 - 1$, то есть что $T_0 > k_0 - 1$. В нашем случае важно лишь то, чтобы период генератора был больше $M \cdot N$.

При проверке теста K-равномерности будем следовать рекомендациям, приведенным в [4], но модифицируем их к нашей задаче. В нашем случае можно не устанавливать значение K-равномерности используемого генератора, а достаточно убедиться лишь в том, что $M < K$. Поэтому тест будет состоять в следующем:

– в каждом столбце матрицы $\mathbf{r}_{M,N}$ выбираем максимальный элемент:

$$\bar{r}_j = \max\left(r_{1,j}, r_{2,j}, \ldots r_{M,j}\right),\ j = 1,2,\ldots N; \qquad (3)$$

– формируем статистику:

$$\bar{\mathbf{r}}_N = \left(\bar{r}_1, \bar{r}_2, \ldots \bar{r}_N\right); \qquad (4)$$

– из статистики $\bar{\mathbf{r}}_N$ формируем статистику $\tilde{\mathbf{r}}_N$:

$$\tilde{\mathbf{r}}_N = \left(\bar{r}_1^M, \bar{r}_2^M, \ldots \bar{r}_N^M\right) = \left(\tilde{r}_1, \tilde{r}_2, \ldots \tilde{r}_N\right),$$
$$\tilde{r}_i = \bar{r}_i^M,\ i = 1,2,\ldots M; \qquad (5)$$

– выборка $\bar{\mathbf{r}}_N$ имеет равномерное распределение в интервале (0,1) [4]. Проверяем гипотезу. Если она подтверждается, K-равномерность генератора больше M.

Представленный тест, естественно, справедлив при условии, что используемый генератор удовлетворяет тестам на независимость в совокупности и равномерному распределению сгенерированных подряд чисел [5].

Заключение

При исследовании сложных систем на ЭВМ методом имитационного моделирования необходимо проверить используемый генератор псевдослучайных чисел на K-равномерность. Если количество обращений к генератору для моделирования одного сценария равно M, то для получения корректных результатов моделирования необходимо, чтобы $M < K$. В противном случае надо предпринять определенные меры, например использовать несколько генераторов пседослучайных последовательностей.

В работе приведена методика проверки неравенства $M < K$, позволяющая оценить возможности использования встроенного генератора псевдослучайных чисел для моделирования исследуемой системы.

Библиографический список

1. ГОСТ Р ИСО 28640–2012. Статистические методы. Генерация случайных чисел. URL: https://docs.cntd.ru/document/1200096454 (дата обращения: 12.11.2021).

2. *Буре В. М., Парилина Е. М.* Теория вероятностей и математическая статистика. СПб.: Лань, 2013. 416 с.

3. *Левитан Ю. Л., Соболь И. М.* О датчике псевдослучайных чисел для ПК // Математическое моделирование. 1990. Т. 2, № 8. С. 119–126.

4. *Ермаков С. М., Михайлов Г. А.* Статистическое моделирование. 2-е изд., доп. М.: Наука, 1982. 296 с.

5. *Иванов М. А., Чугунков И. В.* Теория, применение и оценка качества генераторов псевдослучайных последовательностей. М.: КУДИЦ-ОБРАЗ, 2003. 240 с.

УДК 004.93 **DOI: 10.31799/978-5-8088-1701-2-2022-2-87-94**

А. В. Яковлев
кандидат технических наук, доцент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ИСПОЛЬЗОВАНИЕ МНОГОСЛОЙНЫХ СЕТЕЙ-АВТОЭНКОДЕРОВ ДЛЯ РАСПОЗНАВАНИЯ УСТАЛОСТИ ЧЕЛОВЕКА НА ОСНОВЕ РЕЧЕВЫХ ДАННЫХ

Представлен подход к распознаванию усталости человека путем обработки его речевого сигнала с помощью многослойных сетей-автоэнкодеров. Подробно описаны метапараметры, используемые для обучения нейронной сети с такой архитектурой.
Ключевые слова: *утомление, когнитивная усталость, речь, автоэнкодер, auDeep, метапараметры обучения сети.*

A. V. Yakovlev
PhD, Tech., Associate Professor
St. Petersburg State University of Aerospace Instrumentation

THE USE OF MULTILAYER AUTOENCODER NETWORKS IN THE RECOGNITION OF HUMAN FATIGUE BASED ON SPEECH DATA

An approach to recognizing human fatigue by processing his speech signal with a multilayer auto-coding network is presented. The metaparameters used to train a neural network with such an architecture are described in detail.
Keywords: *fatigue, cognitive fatigue, speech, autoencoder, auDeep, network training metaparameters.*

Введение

Работа человека-оператора характеризуется напряжением внимания с необходимостью его переключения, а также высоким нервно-психическим напряжением в связи с высокой ответственностью за результаты деятельности. Высокие нагрузки способствуют развитию у человека-оператора состояния утомления (синоним: усталость), приводящего к снижению внимания, пропуску значимых сигналов и немотивированному реагированию на ложные сигналы. Поэтому контроль за развитием утомления у человека-оператора в процессе его профессиональной деятельности является актуальной проблемой.

Большая часть подходов к распознаванию усталости основана либо на самооценке человека, либо на применении контактных датчиков, снижающих подвижность человека-оператора и отвлекающих его от работы. Поэтому разработка подходов к распознаванию усталости человека-оператора с использованием бесконтактных технологий, основанных на анализе видео и речи, – актуальная научно-практическая задача.

С развитием и удешевлением видеооборудования распространение получают методы мониторинга усталости человека, основанные на видеорегистрации его лица [1, 2]. Их достоинство в том, что они не мешают человеку выполнять привычные действия, не отвлекают его. Однако качество распознавания утомления на основе анализа данных его изображения зависит от условий освещенности, угла съемки, характеристик камеры, канала передачи данных и возможностей вычислительных ресурсов.

Другой актуальный подход к оценке усталости человека – анализ его речи. По сравнению с анализом видео, речь – более удобный показатель для анализа, так как для ее регистрации требуется только микрофон и меньше зависимость от внешних факторов. Однако построение алгоритма распознавания усталости человека на основе его речи представляет алгоритмическую сложность, связанную со сложностью речевого анализа и методов его анализа [3, 4]. Кроме того, для построения такого алгоритма необходим качественный набор экспериментальных данных, содержащий надежные «внешние» оценки усталости добровольцев, принимающих участие в исследовании.

Состояние исследований

Говоря о «внешних» критериях наступления утомления, необходимо отметить, что у феномена *общей усталости* человека есть две составляющие: периферическая усталость и когнитивная усталость. *Периферическая усталость* связана с нарушением нервно-мышечной передачи, метаболическими нарушениями, дефектами мышечных мембран или недостаточностью периферического кровообращения [5]. *Когнитивная усталость,* или усталость ЦНС, – это неспособность поддерживать внимание [6].

Существуют методы самооценочной субъективной оценки усталости. Однако их недостаточно, поскольку они могут значимо отличаться от объективных оценок когнитивной или физической усталости [7].

Основные взгляды на измерение усталости основаны на оценке снижения работоспособности [7]:

1) после длительного периода времени (продолжительных усилий);

2) после тяжелого умственного напряжения;

3) после тяжелой физической нагрузки;

4) во время сильных, но мотивированных (поддерживаемых) умственных усилий.

Последний подход получил наибольшее эмпирическое подтверждение в исследованиях. Примером служит корректурная проба – метод психологии труда, используемый для выявления утомляемости, оценки концентрации и устойчивости внимания. Однако его применение ограничено отдельными видами деятельности.

Еще одна группа методов для оценки усталости – методы, основанные на объективной регистрации физиологических параметров организма человека. Наиболее информативны оценки усталости на основе анализа электроэнцефалограммы (ЭЭГ) либо электрокардиограммы (ЭКГ). Однако хорошие измерения ЭЭГ и ЭКГ труднодостижимы в эксперименте [8]. Поэтому вместо ЭКГ чаще используются данные фотоплетизмографии (ФПГ) с применением датчиков, размещенных на ухе [8], на лбу или на пальце. Такое их расположение ненавязчиво для человека-оператора и не отвлекает его от работы. При этом может быть достигнута достаточная точность измерений, в том числе путем вейвлет-анализа показателей вариабельности сердечного ритма и применения метода опорных векторов [9].

Таким образом, эффективным путем для формирования «внешней» оценки усталости можно считать сочетание подходов, основанных на моделировании элементов предметной деятельности человека-оператора с одновременной регистрацией его психофизиологических, видео- и речевых параметров.

Применение многослойных сетей-автоэнкодеров для обработки речи

Существуют два базовых подхода к построению моделей распознавания утомления человека-оператора по его речи. Первый связан с генерацией большого числа низкоуровневых дескрипторов речевого сигнала [10]. Основным средством для получения этих показателей из исходного речевого сигнала является библиотека openSmile* [11]. Второй основан на использовании многослойных сетей-автоэнкодеров (англ. multilayer autoencoder networks). Глубокие нейронные сети с такой архитектурой применяются для решения задачи сегментации изображений, когда на изображении требуется выделить основные структурные элементы. Речь же, являясь акустическим сигналом, представляет собой процесс распространения энергии акустических колебаний в упругой среде. Поэтому для задачи распознавания акустических сигналов с помощью многослойных сетей-автоэнкодеров (далее – сетей) первоначальный акустический сигнал преобразуется в спектрограмму (рис. 1). Необходимость такого преобразования связана с тем, что сверточные нейронные сети первоначально предназначались для работы с изображениями, а не с сигналами. И на их вход должно подаваться изображение.

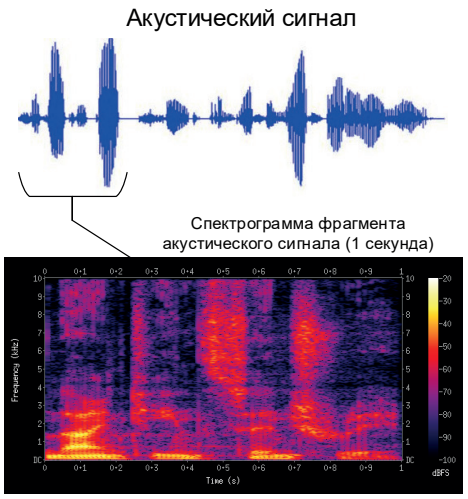


Рис. 1. Преобразование исходного акустического сигнала в спектрограмму

* www.audeering.com/research/opensmile/

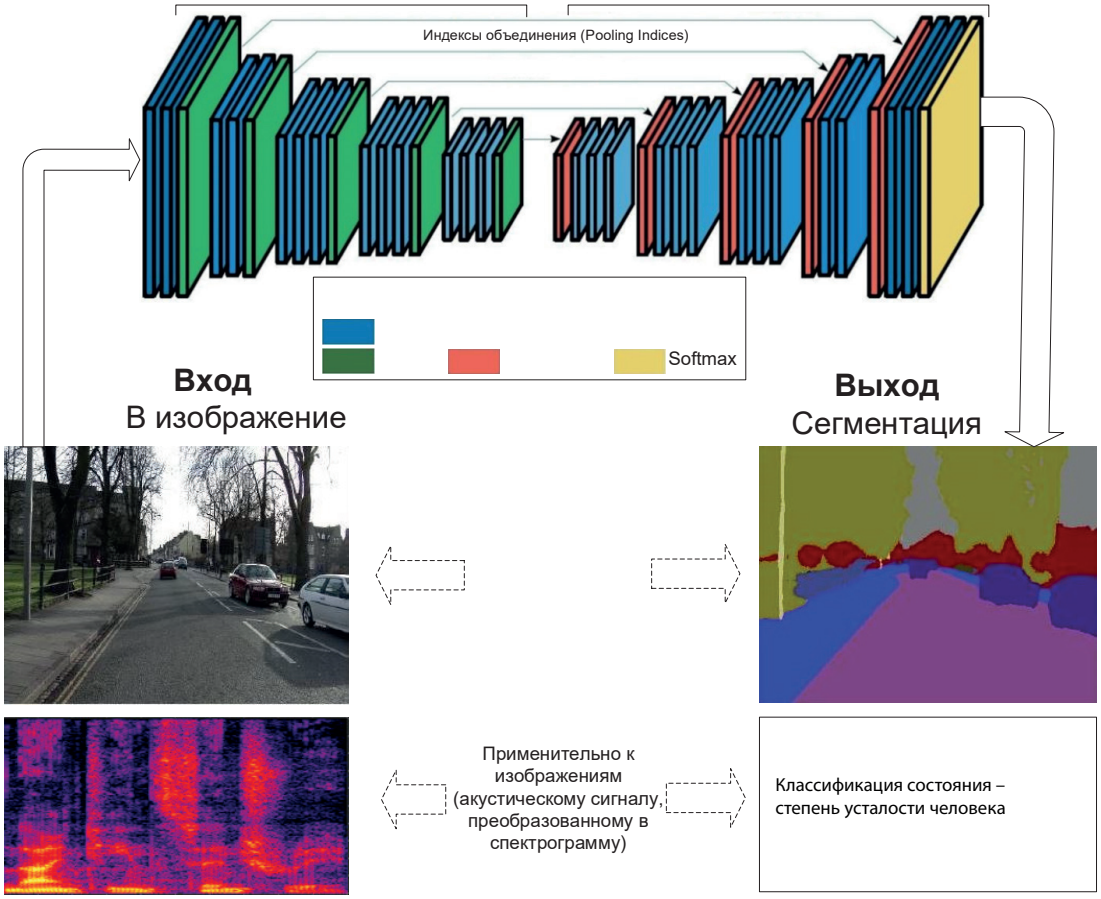


Рис. 2. Иллюстрация архитектуры многослойных сетей-автоэнкодеров (на примере сети SegNet [12])

На рис. 2 представлена иллюстрация архитектуры многослойных сетей-автоэнкодеров на примере глубокой нейронной сети SegNet [12]. Она состоит из большого числа слоев, объединенных в две подсети: кодировщика и декодировщика [13].

В результате обработки входного изображения заранее обученная сеть сегментирует и классифицирует все элементы, представленные на исходном изображении. Аналогичная операция происходит, когда на вход сети поступает спектрограмма. В процессе обучения сеть формирует сжатое представление исходной спектрограммы, отражающее ее существенные характеристики, и сопоставляет с классами состояния утомления. Поэтому результатом работы сети является не сегментация элементов изображения, а классификация состояния – степень усталости человека.

Последовательность обучения многослойных сетей-автоэнкодеров для задачи распознавании утомления человека по речи представлена на рис. 3.

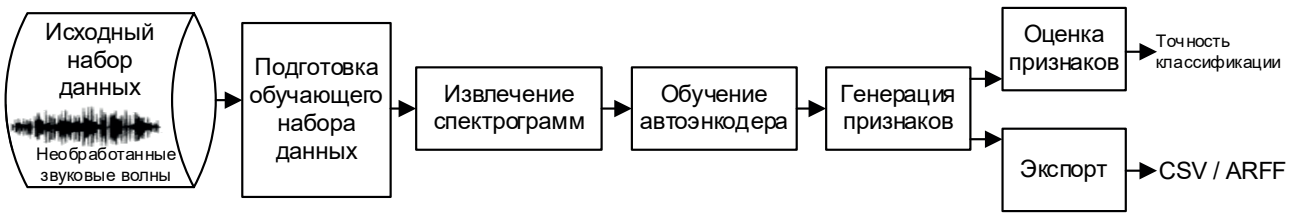


Рис. 3. Этапы обучения многослойных сетей-автоэнкодеров [4, 14]

Она состоит из шести этапов* [4, 14].

1. Подготовка обучающего набора данных (англ. dataset) – образцов речевых сигналов с метками (англ. label) классов усталости для «работы» с нейронной сетью. Обычно такая подготовка состоит в «оформлении» этого набора данных в соответствии с требованиями парсера, который будет «разбирать» его на этапе извлечения спектрограмм.

2. Извлечение спектрограмм (англ. preprocess): извлечение спектрограмм и метаданных из необработанных аудиофайлов.

3. Обучение автоэнкодера (англ. train): обучение сети на извлеченных спектрограммах.

4. Генерация признаков (англ. generate): генерация признаков обученной сетью.

5. Оценка признаков (англ. evaluate): оценка сгенерированных признаков.

6. Экспорт (англ. export): экспорт сгенерированных признаков в CSV/ARFF.

Обучение автоэнкодера управляется набором метапараметров.

Метапараметры обучения автоэнкодера условно объединяются в группу управления препроцессом извлечения спектрограмм и группу собственно обучения сети.

Метапараметры «**window-width**», «**window-overlap**» используются на втором этапе обучения при выполнении оконного преобразования Фурье (рис. 4). Оно состоит в том, что из исходного акустического сигнала во временной области вырезаются сегменты на основе заданной пользователем ширины окна (метапараметр «window-width»; синоним: англ. window length – длина окна) и размера перекрытия окна (метапараметр «window-overlap»; синоним: англ. overlap length – длина перекрытия). Размер смещения окна (англ. hop length) R рассчитывается как (window-width – window-overlap). В процессе «вырезания» сегменты умножаются на оконную функцию (например, Хэмминга) длиной window-width.

Результатом оконного преобразования Фурье исходного аудиосигнала является набор спектров выделенных сегментов исходного акустического сигнала, преобразованных оконной функцией. Получаемый набор частных спектров накладывается друг на друга и формируется спектрограмма – метод визуального представления амплитуды (громкости) сигнала, меняющейся с течением времени на разных частотах в форме изображения, показывающего зависимость спектральной плотности мощности сигнала от времени. Получаемая двумерная диаграмма имеет время по горизонтальной оси и частоту в логарифмической шкале по вертикальной оси; третье измерение с указанием амплитуды в децибелах на определенной частоте в конкретный момент времени представлено интенсивностью или цветом каждой точки изображения [15] (см. рис. 1).

Особенность формирования спектрограмм для обучения сети – возможность использования метапараметра **mel-spectrum**, позволяющего при предобработке обучающих данных вместо логарифмических спектрограмм мощности производить извлечение мел-спектрограмм (от англ. mel, melody) в логарифмическом масштабе с использованием заданного количества фильтров. Это дает возможность при обучении учитывать психофизиологические особенности слухового аппарата человека: реакция человеческого слухового аппарата является функцией уровней частоты и громкости (рис. 5, *а*). Эта взаимосвязь показывает разницу между физическими измерениями и психологическим восприятием [16]. Мел-частотный анализ представляет частоты речи с позиции психоакустического параметра слуха – высоты тона. Нелинейную связь между частотой звука и его высотой отображает мел-частотная шкала (рис. 5, *б*) [17].

Метапараметр **channels** настраивает обработку стереоаудиофайлов. Спектрограммы могут извлекаться из среднего значения двух каналов (channels = MEAN), из левого или правого каналов (channels = LEFT или RIGHT соответственно) и из разности двух каналов (channels = DIFF).

При подготовке исходного набора речевых сигналов на первом этапе обучения длительность записанных аудиофайлов может быть различной. Для «выравнивания» их длительностей используется метапараметр **fixed-length**, устанавливающий точную длину (в секундах) вырезаемого фрагмента из исходных аудиоданных. Для аудиофайлов длиннее указанной длины используются только первые N секунд. Если в обучающих данных есть аудиофайлы короче указанной длины, обучение не производится и выдается ошибка. Вместе с метапараметром **fixed-length** может использоваться метапараметр **center-fixed**, позволяющий взять фрагмент из середины обрабатываемого аудиофайла. Длина фрагмента будет определяться значением, заданным в fixed-length, поэтому метапараметр center-fixed работает только при установленном метапараметре fixed-length.

* На примере библиотеки auDeep (https://github.com/auDeep/auDeep), в основе которой лежит использование автоэнкодера, реализованного в библиотеке TensorFlow.

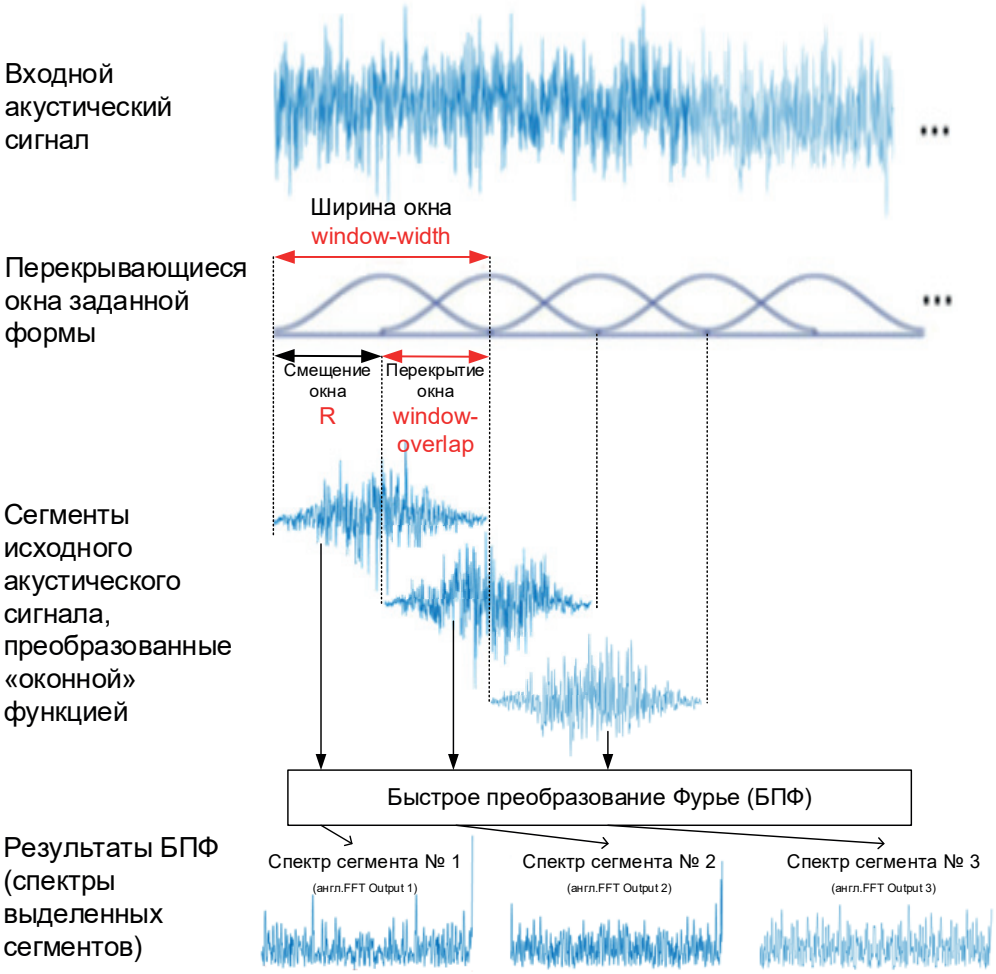


Рис. 4. Иллюстрация оконного преобразования Фурье исходного аудиосигнала. Источник англоязычного изображения: https://www.mathworks.com/help/dsp/ref/dsp.stft.html

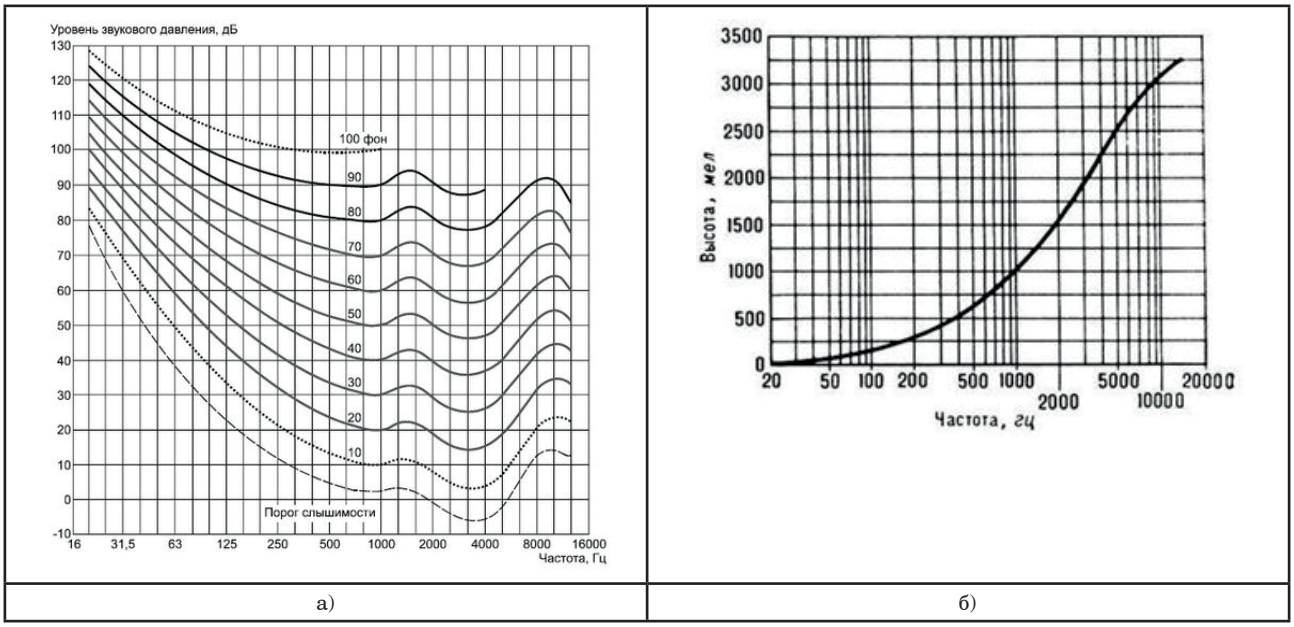


Рис. 5. Иллюстрация метапараметра mel-spectrum: а – стандартные кривые равной громкости чистых тонов согласно ГОСТ Р ИСО 226-2009 [16]; б – зависимость между высотой чистых тонов и частотой при постоянном уровне громкости (мел-шкала) [17]

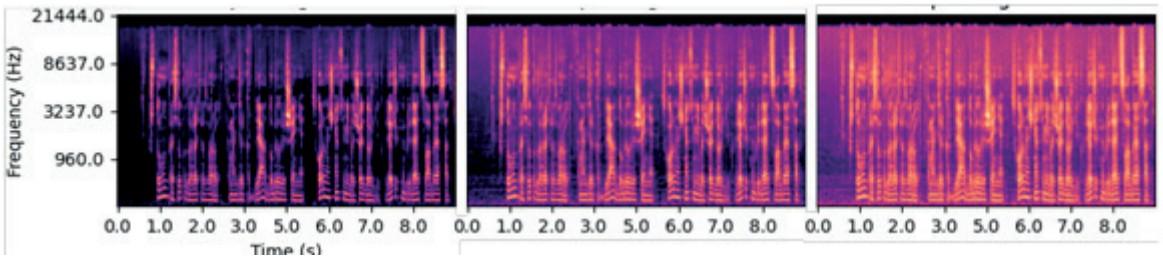


Рис. 6. Примеры спектрограмм одного аудиофайла для которых установлены разные значения метапараметра clip_below (слева направо: –45dB, –60dB, –75dB)

Спектрограммы при формировании нормируются таким образом, чтобы наибольшая амплитуда равнялась 0 dB. Можно предположить, что при анализе речевого сигнала такую наибольшую амплитуду будет формировать непосредственно говорящий. Вместе с тем в анализируемом речевом сигнале могут присутствовать помехи, посторонние шумы, дефекты речи и т. п. Естественно, все они будут присутствовать на спектрограммах, используемых в процедуре обучения сети. Один из путей улучшения изображения аудиосигнала – обрезка амплитуд ниже указанного значения в децибелах, которая управляется метапараметром **clip_below** (от англ. clipping [clip] below «обрезание, отсечение ниже»). Разница в значениях clip_below проиллюстрирована на рис. 6 для спектрограмм одного аудиофайла с разными уровнями обрезки исходной амплитуды (45dB, 60dB и 75dB).

Были рассмотрены метапараметры, относящиеся к группе управления процессом извлечения спектрограмм. Далее опишем метапараметры непосредственно обучения.

Модели глубокого обучения разбивают обучающий набор данных на небольшие пакеты – равномерные выборки из обучающего набора данных [18]. Пакет или мини-пакет (англ. batch или mini-batch) – небольшой набор образцов (обычно от 8 до 128), обрабатываемых моделью одновременно. Число образцов часто является степенью двойки для более эффективного использования памяти GPU. В процессе обучения сети один мини-пакет используется в градиентном спуске для вычисления одного изменения весов модели [18].

Размер такого мини-пакета (метапараметр **batch-size**) может составлять от одного до нескольких сотен образцов (англ. sample – образец, сэмпл, наблюдение, экземпляр) обучающе-

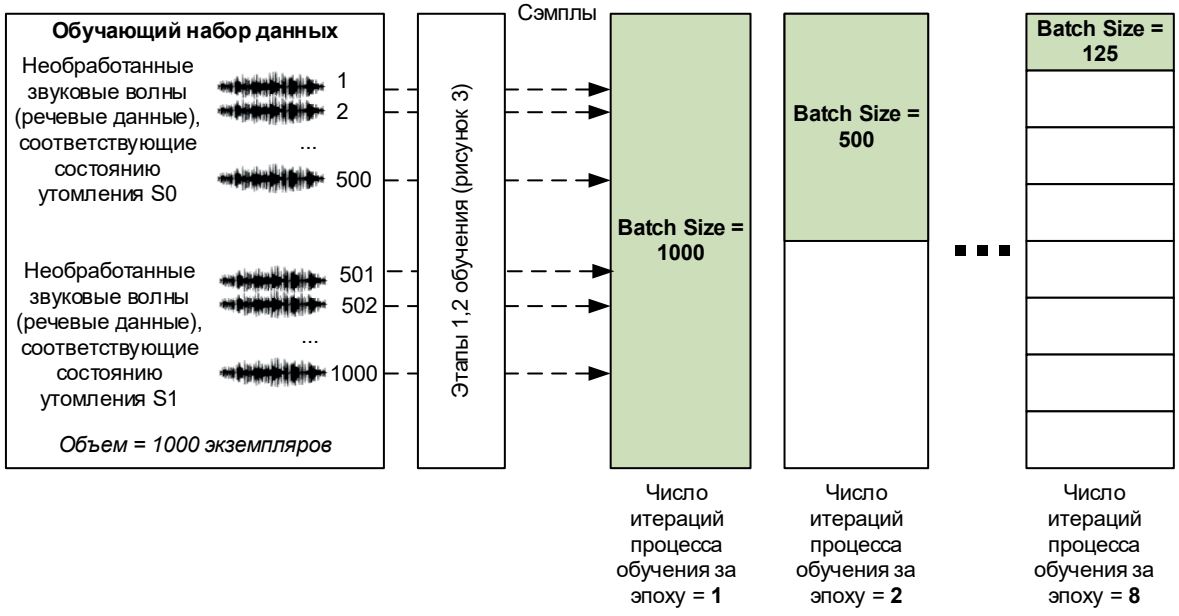


Рис. 7. Связь числа мини-пакетов и числа итераций обучения сети за эпоху

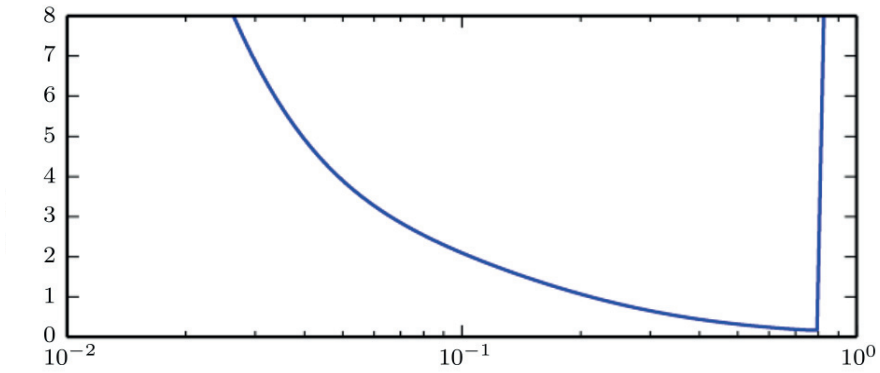


Рис. 8. Типичная зависимость ошибки обучения от скорости обучения и ошибкой обучения [19]

Состояние	Sf_0	Sf_1
Sf_0	0,6	0,4
Sf_1	0,32	0,69

Время обучения – 2m50s
Sf_0 – состояние «не утомлен»
Sf_1 – состояние «утомлен»

Рис. 9. Пример матрицы метрики ошибки

го набора данных [19, 20]. Общее число мини-пакетов равно объему обучающего набора данных, деленному на метапараметр batch-size. Размер последнего мини-пакета равняется остатку от деления. Все мини-пакеты перебираются за время одной эпохи (англ. epoch). На рис. 7 проиллюстрировано, как в зависимости от значения метапараметра batch-size и объема обучающего набора данных изменяется число итераций процесса обучения сети в течении одной эпохи.

В конце каждой итерации прогнозы сравниваются с ожидаемыми выходными предсказаниями и вычисляется ошибка, по ней алгоритм обновления улучшает модель [20]. Таким образом, обучение одной эпохи – это прогон всех образцов набора обучающих данных (упакованных в мини-пакеты) через нейронную сеть. Количество эпох (метапараметр **num-epochs**) – это параметр, который определяет, сколько раз алгоритм обучения будет работать со всем набором обучающих данных.

Метапараметр **learning-rate** определяет скорость обучения используемого в библиотеке auDeep оптимизатора Adam [19]. В основе этого оптимизатора лежит градиентный спуск, в соответствии с которым выбор новой точки определяется соотношением [19] $x' = x - \varepsilon\nabla_x f(x)$, где ε – рассматриваемая скорость обучения, положительный скаляр, определяющий длину шага. Обычно в качестве ε выбирается малая константа. Этот параметр имеет нелинейную связь с ошибкой обучения (рис. 8). Когда скорость обучения слишком велика, градиентный спуск может непреднамеренно увеличить, а не уменьшить ошибку обучения. В идеализированном квадратичном случае это происходит, если скорость обучения как минимум вдвое превышает ее оптимальное значение [21]. Когда скорость обучения слишком мала, обучение не только замедляется, но и может навсегда застрять с высокой ошибкой обучения.

Метапараметр **num-layers** определяет количество слоев нейронной сети в кодере и декодере, метапараметр **num-units** – количество нейронов в каждом слое нейронной сети в кодере и декодере.

Один из важных результатов обучения сети – матрица метрики ошибки (англ. confusion matrix), показывающая насколько хорошо обученная сеть может определять состояния утомления человека по его речи (рис. 9).

В случае если полученные характеристики устраивают исследователя, производится выгрузка параметров обученной сети. В противном случае производится изменение метапараметров обучения и переобучение сети.

Заключение

Применение многослойных сетей-автоэнкодеров (в частности, библиотеки auDeep) для оценки уровня утомления человека по его речи, несомненно, перспективно и показывает хорошие прогностические возможности. Этот подход не требует значительных теоретических ис-

следований и «ручного» труда по изучению данных. Вместе с тем предложенная методика подразумевает внимательное отношение к формированию обучающих данных и настраивание метапараметров обучения сети. В целом перспективность предложенного подхода связана с относительной простотой регистрации речевого сигнала без какого-либо стеснения деятельности человека.

Библиографический список

1. *Golz M.* et al. Evaluation of fatigue monitoring technologies // Somnologie – Schlafforschung und Schlafmedizin. 2010. Vol. 14. P. 187–199.

2. *Яковлев А. В., Матыцин В. О.* Комплексная диагностика голосовой утомляемости у операторов на основе применения методик машинного обучения и компьютерного зрения // Профилактическая медицина-2018. 2018. Т. 3. С. 238–242.

3. *Яковлев А. В., Велюга В. А.* Контроль речевого сигнала как средство выявления утомления человека // Метрологическое обеспечение инновационных технологий: матер. III Междунар. форума / под ред. В. В. Окрепилова. СПб., 2021. С. 269–270.

4. *Яковлев А. В.* Использование методов Deep Learning для оценки уровня когнитивной усталости человека по его речи // Обработка, передача и защита информации в компьютерных системах'21: Междунар. науч. конф.: сб. докл. СПб., 2021. С. 48–53.

5. *Chaudhuri A., Behan P. O.* Fatigue and basal ganglia // J. Neurol. Sci. 2000. Vol. 179, № S 1–2. P. 34–42.

6. *Chaudhuri A., Behan P. O.* Fatigue in neurological disorders // Lancet. 2004. Vol. 363, № 9413. P. 978–988.

7. Fatigue as a window to the brain. Cambridge, MA, US: MIT Press, 2005. 336 p.

8. PPG Heart Beat for Cognitive Fatigue Prediction // Kaggle. URL: https://www.kaggle.com/canaria/5-gamers (дата обращения: 27.04.2020).

9. *Li G., Chung W.*-Y. Detection of Driver Drowsiness Using Wavelet Analysis of Heart Rate Variability and a Support Vector Machine Classifier // Sensors (Basel, Switzerland). 2013. Vol. 13. P. 16494–16511.

10. *Кипяткова И. С., Ронжин А. Л., Карпов А. А.* Автоматическая обработка разговорной русской речи: монография. СПб.: ГУАП, 2013. 314 с.

11. *Eyben F.* et al. Recent Developments in openSMILE, the Munich Open-source Multimedia Feature Extractor // Proceedings of the 21st ACM International Conference on Multimedia. New York, NY, USA: ACM, 2013. P. 835–838.

12. *Badrinarayanan V., Kendall A., Cipolla R.* SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation // IEEE Transactions on Pattern Analysis and Machine Intelligence. 2017. Vol. 39, № 12. P. 2481–2495.

13. *Яковлев А. В.* Современные направления развития прикладной информатики: учеб. пособие. СПб.: ГУАП, 2021. 87 с.

14. *Freitag M.* et al. auDeep: Unsupervised Learning of Representations from Audio with Deep Recurrent Neural Networks // Journal of Machine Learning Research. 2017. Vol. 18.

15. *Roberts L.* Understanding the Mel Spectrogram // Analytics Vidhya. 2020. URL: https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53 (дата обращения: 01.11.2021).

16. ГОСТ Р ИСО 226–2009. Акустика. Стандартные кривые равной громкости. М.: Стандартинформ, 2019.

17. *Скучек Е.* Основы акустики: пер. с нем.: в 2 т. Т. 2. М., 1959.

18. *Шолле Ф.* Глубокое обучение на Python. СПб.: Питер, 2018. 400 с.

19. *Гудфеллоу Я., Бенджио И., Курвилль А.* Глубокое обучение / пер. с англ. А. А. Слинкина. 2-е изд., испр. М.: ДМК Пресс, 2018. 652 с.

20. *Brownlee J.* Difference Between a Batch and an Epoch in a Neural Network // Machine Learning Mastery. 2018. URL: http://machinelearningmastery.com/difference-between-a-batch-and-an-epoch/ (дата обращения: 01.11.2021).

21. *LeCun Y.* et al. Efficient BackProp // Neural Networks: Tricks of the Trade / ed. Orr G.B., Müller K.-R. Berlin, Heidelberg: Springer, 1998. P. 9–50.

УДК 004.93 **DOI: 10.31799/978-5-8088-1701-2-2022-2-95-100**

А. В. Яковлев
кандидат технических наук, доцент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

РАЗРАБОТКА РАСПРЕДЕЛЕННОЙ ПРОГРАММНОЙ СИСТЕМЫ ДЛЯ СИНХРОНИЗИРОВАННОГО СБОРА РЕЧЕВЫХ, ВИДЕО- И ПСИХОФИЗИОЛОГИЧЕСКИХ ДАННЫХ О ДОБРОВОЛЬЦЕ В ПРОЦЕССЕ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

Состояние организма человека может быть описано значительным числом параметров. В большинстве своем это результаты обработки психофизиологических данных. Развитие технологий дает возможность получать новые параметры о человеке с использованием средств видеорегистрации и записи его речи. Рассматриваются вопросы создания специализированной распределенной программной системы, позволяющей синхронизированно регистрировать речевые, видео- и психофизиологические данные о добровольце в процессе экспериментального исследования с целью создания качественных наборов экспериментальных данных.
Ключевые слова: *система сбора данных; состояние организма человека; регистрация видеоданных, речи, психофизиологических данных; экспериментальное исследование.*

A. V. Yakovlev
PhD, Tech., Associate Professor
St. Petersburg State University of Aerospace Instrumentation

DEVELOPMENT OF A DISTRIBUTED SOFTWARE SYSTEM FOR SYNCHRONIZED COLLECTION OF SPEECH, VIDEO AND PSYCHO-PHYSIOLOGICAL DATA ABOUT A VOLUNTEER IN THE PROCESS OF EXPERIMENTAL RESEARCH

The state of the human body can be described by a significant number of parameters. For the most part, these are the results of processing of psychophysiological data. The development of technology makes it possible to obtain new parameters about a person using video recording and recording of his speech. The issues of creating a specialized distributed software system that allows synchronized recording of speech, video and psychophysiological data about a volunteer in the process of experimental research are considered in order to create a qualitative sets of experimental data.
Keywords: *data collection system; the state of the human body; registration of video data, speech, psychophysiological data; experimental study.*

Актуальность разработки

Современные системы автоматизированной оценки состояния человека позволят идентифицировать личность, дать экспресс-информацию о работоспособности человека-оператора, зафиксировать изменение психоэмоционального статуса обследуемого. В настоящее время подобные системы разрабатываются специалистами многих стран. В основе их работы лежат многочисленные экспериментальные исследования, так как для создания надежных алгоритмов распознавания состояния человека необходимо наличие качественных наборов данных (англ. dataset).

Значительное число признаков, характеризующих состояние человека, может быть рассчитано на основе сбора и обработки данных физиологической, видео и акустической модальностей [1]. Под термином «модальность» понимается объединение потоков данных о состоянии человека на основании общности подходов к регистрации, сохранению и математической обработки этих данных (рис. 1).

Проблема состоит в том, что в области экспериментальных исследований с участием добровольцев для качественной регистрации признаков каждой модальности обычно используется специфическое оборудование. Причем большая

Рис. 1. Потоки данных, подлежащие регистрации и сбору для дальнейшего анализа

его часть представляет собой аппаратно-программные комплексы с проприетарным программным обеспечением, которое обладает «закрытыми» интерфейсами и не предполагает использования в составе единого измерительного комплекса.

Архитектура системы

Так как создание качественного dataset связано с необходимостью синхронного сбора данных с различного оборудования, то возникла необходимость разработки средства автоматизации этого процесса.

В качестве альтернативных вариантов рассматривались:

1) установка/подключение требуемого оборудования и развертывание поставляемого с ним программного обеспечения на одной вычислительной машине;

2) установка оборудования на нескольких вычислительных машинах, объединенных в локальной компьютерной сети.

Первый вариант, несмотря на кажущуюся простоту, не обеспечивал решение поставленной задачи, так как одновременная регистрация потоков видеоданных с остальными может привести к потере части данных. Кроме того, стоимость компьютера также будет высока.

Второй вариант – разработка распределенной программной системы на базе локальной компьютерной сети – был менее требователен к характеристикам вычислительной техники и позволял использовать обычное оборудование. Однако для него нужна более высокая квалификация разработчиков.

Был выбран второй вариант. В основу разработанной распределенной программной системы для синхронизированного сбора речевых, видео- и психофизиологических данных о добровольце в процессе экспериментального исследования (далее – РПС) была положена локальная компьютерная сеть, в которую было объединено оборудование для регистрации данных разных модальностей.

Структура РПС построена по традиционной схеме и содержит компоненты сбора данных, хранения данных и обработки данных [2] (рис. 2).

Компоненты получения данных включают средства управления данными, определяющие

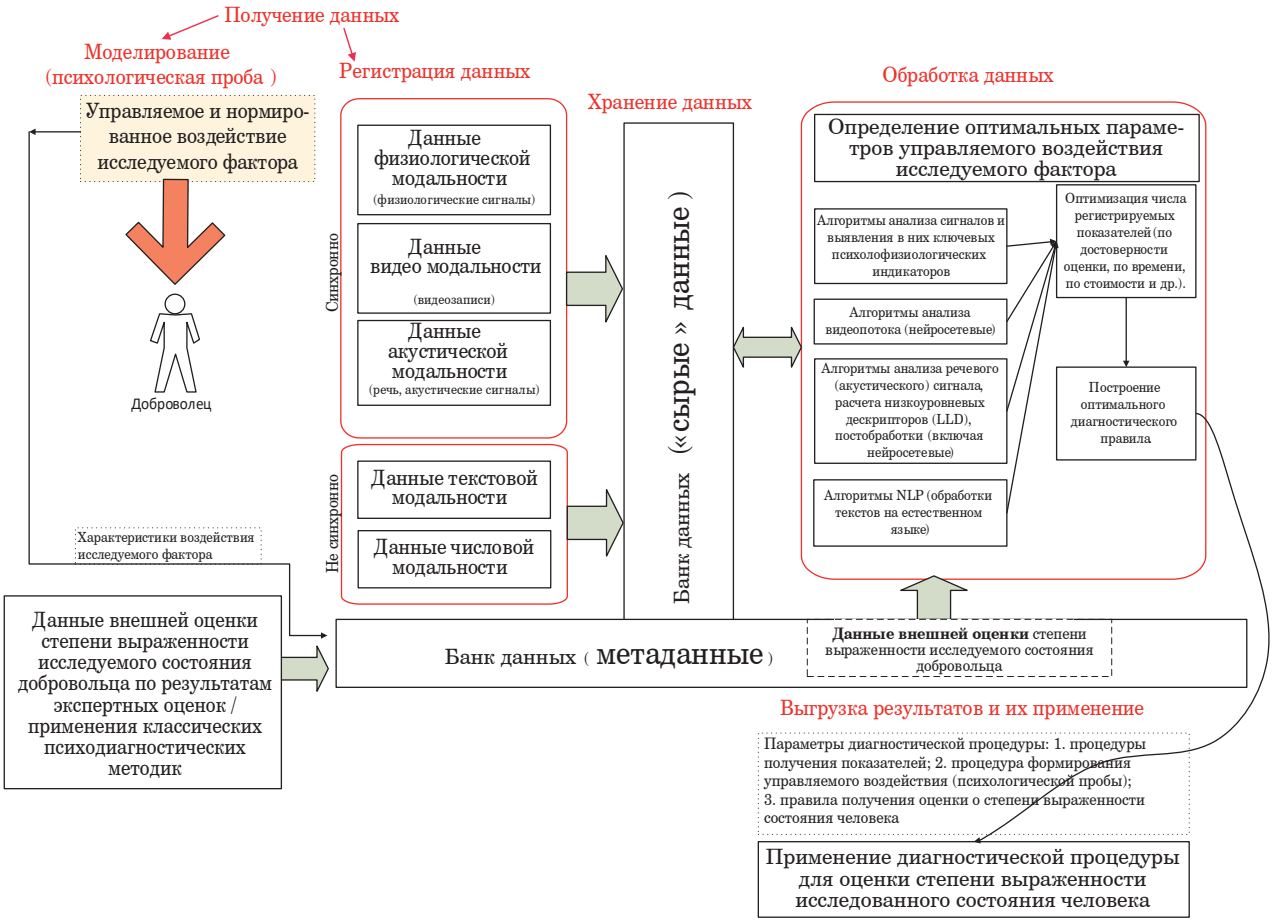


Рис. 2. Структура распределенной программной системы для синхронизированного сбора речевых, видео- и психофизиологических данных о добровольце в процессе экспериментального исследования

порядок регистрации данных и помещения их в базу данных. Этот компонент также включает средства моделирования, которые в соответствии с дизайном экспериментального исследования позволяют предъявить добровольцу, принимающему участие в исследовании, определенные стимульные материалы (например, тексты, изображения, видеофрагменты, музыкальные произведения и т. п.). Таким образом, производятся синхронизированный сбор «сырых», т. е. необработанных данных всех модальностей из различных источников, и их помещение в специализированную базу данных. Синхронизированным способом производится сбор только данных видео-, аудио- и физиологической модальностей. Данные текстовой и числовой модальности (например, пол, возраст, профессия, рост, вес, имя добровольца) вносятся перед началом экспериментального исследования.

В качестве источников данных физиологической, видео- и аудиомодальностей было использовано следующее оборудование:

– для регистрации данных физиологической модальности:

- аппаратная часть профессионального компьютерного полиграфа (ПКП) «Диана-7М» со специализированной библиотекой для получения данных, установленная на ноутбуке с операционной системой MS Windows 10;

– для регистрации данных видеомодальности:

- две IP камеры Axis, разрешением 1280×720 пикселей из состава видеокомплекса «Диана-Видео»;

– для регистрации данных акустической модальности:

- внешняя звуковая карта M-Track Plus;
- профессиональный петличный микрофон;
- измерительный микрофон 378A14 (производитель PCB Piezotronics, Inc., США, № 61720–15 в Госреестре средств измерений) вместе с усилителем.

Для предъявления стимульного материала использовался мультимедиапроектор.

В банке данных организуется и хранится для дальнейшего пользования вся собираемая информация, включая сведения об используемых средствах регистрации, а также о дизайне исследования. Это связано с тем, что разрабатываемая система позволяет проводить исследования с использованием разного регистрирующего и измерительного оборудования и предполагает легкую модификацию дизайна проводимого исследования.

Обработка данных состоит в применении современных методов анализа с целью извлечения знаний из собранных данных. Цель такой обработки собранных данных – выявление информативных признаков, свидетельствующих о наличии различных состояний у добровольцев, принимающих участие в исследованиях.

Реализация системы

Рассматривалось несколько архитектур и способов реализации РПС. В итоге предпочтение было отдано сетевой, клиент-серверной архитектуре. Соответственно, РПС состоит из двух частей – серверной и клиентской.

Клиентская часть – это графический интерфейс, с которым работает пользователь. Было рассмотрено несколько основных инструментов для создания графического интерфейса:

– библиотека Qt – кроссплатформенная библиотека для проектов на C/C++;

– Windows C# Forms – фреймворк для языка Visual C# или Visual C++, ориентированный исключительно под операционную систему Windows;

– Web-технологии – кроссплатформенное решение, использующее веб-браузеры.

В связи с тем что в локальной сети были устройства с операционными системами Windows и Linux, в качестве инструмента были выбран стек web-технологий (табл. 1), поскольку с их помощью можно было реализовать требуемый функционал в минимальные сроки с минимальным количеством кода.

Таблица 1

Используемый при разработке РПС стек web-технологий

Классический	Использованный
HTML (язык гипертекстовой разметки) CSS (язык каскадных стилей) JavaScript (скриптовой язык)	Обычно применяют системы сборки проектов и надстройки для компиляции в эти языки. Для разрабатываемого АПК были использованы: webpack для сборки проекта; TypeScript вместо JavaScript, так как TypeScript привносит в JavaScript строгую типизацию, что позволяет избегать ошибки еще во время написания кода. Во время сборки код на TS компилируется в обычный JS; SCSS вместо CSS, т.к. SCSS упрощает написание кода с помощью синтаксического анализатора – во время сборки код на SCSS компилируется в обычный CSS код

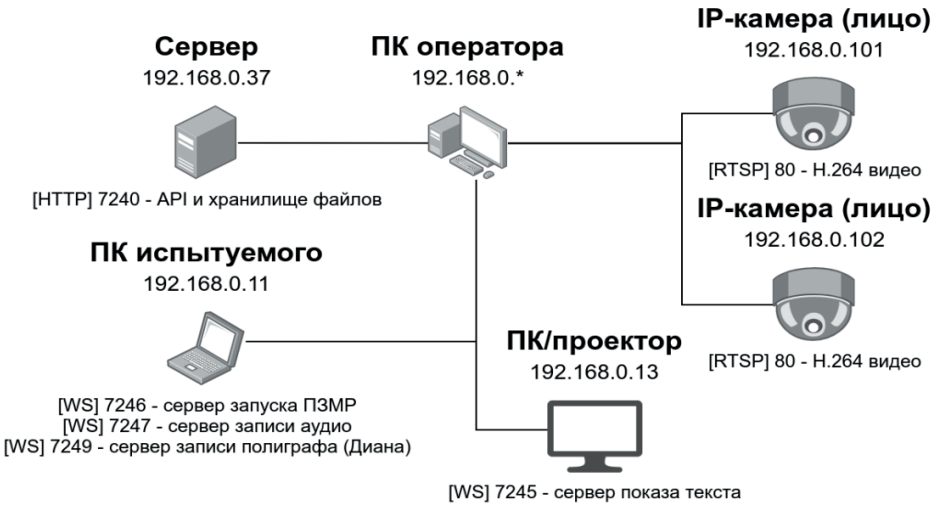


Рис. 3. Топология экспериментальной локальной компьютерной сети РПС

Таблица 2

Устройства и элементы РПС, взаимодействующие посредством экспериментальной локальной сети

IP-адрес	Порт	Назначение устройства / элемента РПС
192.168.0.37	7240	Сервер RESTful API, а также хранилище файлов банка данных (СУБД MySQL 5.8.)
192168.0.20	–	Компьютер оператора РПС
192.168.0.101	80	IP-камера для съемки лица добровольца
192.168.0.102	80	IP-камера для съемки общего плана эксперимента
192.168.0.11	7246	Сервер записи и запуска ПЗМР
	7247	Сервер записи аудио
	7249	Сервер записи физиологических показателей (полиграф «Диана-7М»)
192.168.0.13	7245	Сервер предъявления стимульного материала (через мультимедиапроектор)

Для серверной части разрабатываемой РПС также рассматривалось несколько вариантов реализации: RESTful API на NodeJS; RESTful API на Python/Django; RESTful API на PHP. Было решено остановиться на NodeJS.

Программный интерфейс приложения (API) разрабатываемой РПС представлен в виде HTTP-сервера, который «слушает» определенный порт и ожидает запросы от клиента(ов), которые будут запрашивать определенные методы.

Как было отмечено, все устройства регистрации данных, управления и предъявления стимульного материала собраны в одной экспериментальной локальной компьютерной сети. Топология сети представлена на рис. 3. Адреса устройств внутри экспериментальной локальной сети, занятые порты и их назначение представлены в табл. 2.

При работе с РПС существуют две системные роли: оператор и руководитель исследования. Руководитель формирует описание эксперимента, а после проведения эксперимента следит, чтобы все данные были корректно занесены в БД. Кроме того, руководитель может выгружать данные из БД в соответствии с критериями формирования требуемого ему набора экспериментальных данных.

Оператор со своего рабочего места удаленно запускает через терминал оператора процедуры и следит за успешностью проведения исследования. Для взаимодействия внутри локальной сети использовался удаленный вызов про-

Таблица 3

Сравнительный анализ способов удаленного вызова процедур

Название	Краткая характеристика	Особенности применительно к РПС
XML RPC	Реализовано на запросах HTTP-POST. Тело запроса находится в формате XML. Процедура выполняется на сервере, и возвращаемое значение также форматируется в XML	Избыточен из-за описания своей структуры (повторных наименований тегов). Также увеличивает размер передаваемых бинарных данных приблизительно на 25%: поскольку XML – текстовый формат, то бинарные данные можно передавать только в кодировке base64 (текстовое представление бинарных данных)
SSH	Протокол для безопасного удаленного входа в систему и других безопасных сетевых служб в незащищенной сети. Может использоваться как основа для ряда безопасных сетевых сервисов. Обеспечивает надежное шифрование, аутентификацию сервера и защиту целостности. Может обеспечивать сжатие	SSH удобен, не имеет отрицательных сторон XML, но не имеет нативного решения для MS Windows, а при разработке РПС использовалась часть компьютеров с операционной системой MS Windows
WebSocket	Описан в спецификации RFC 6455, обеспечивает возможность обмена данными между браузером и сервером через постоянное соединение. Данные передаются по нему в обоих направлениях в виде «пакетов», без разрыва соединения и дополнительных HTTP-запросов. Не имеет избыточности, не зависит от платформы и имеет нативную поддержку передачи бинарных данных без увеличения их размера. Имеет возможность компрессии данных «из коробки» для более быстрой передачи	Протокол WebSocket предназначен не для удаленного вызова процедур, а для абстрактного обмена данными. Соответственно, для его использования в РПС требуется написать клиент и сервер, которые общаются по определенному соглашению/схеме

цедур по локальной сети. Существует множество различных способов для удаленного вызова процедур. Большинство из них работают над протоколом TCP. В табл. 3 дан сравнительный анализ способов удаленного вызова процедур с позиции их применимости в разрабатываемой РПС.

Учитывая, что на всех удаленных устройствах, включенных в экспериментальную локальную сеть, требовалась разработка дополнительного программного обеспечения, было решено использовать протокол WebSocket для общения терминала оператора с остальными сервисами.

Для большего удобства был сформулирован и описан стандарт, по которому определяется формат общения терминала оператора с другими устройствами для записи по WebSocket – используются текстовые сообщения, сериализованные в JSON. В соответствии с этим стандартом, при необходимости выполнения какого-либо действия, терминал оператора отправляет сообщение в формате JSON с ключом action и строковым значением, которое умеет обрабатывать удаленное устройство. Также вместе могут быть переданы дополнительные параметры.

При возникновении на устройстве записи (сервере) какого-либо события, на терминал оператора передается объект с полем event и строковым значением, обозначающим, что именно произошло, а также другими деталями события.

Например, оператор выбирает «запись ПЗМР» в терминале оператора – терминал формирует подобную JSON строку:

*{«action»:»start»,»interval_min»:*1000*,»interval_max»:*5000*,»count»:*20}

Эта строка обозначает, что мы хотим начать запись (action = start), а также передаем дополнительные параметры, которые ожидает сервер записи ПЗМР: interval_min. interval_max (минимальное и максимальное время интервала в миллисекундах соответственно) и count (количество итераций показа симульного материала).

При окончании прохождения ПЗМР сервер отправляет на терминал оператора событие об окончании записи result (готовы результаты) и в поле content возвращает отчет о проделанной испытуемым работе (т. е. сами результаты):

*{“event”:”result”,”content”:”{\”left\”:{\”timings\”:[*230*,*289*,*354*,...],\”errors\”:*0*},\”right\”:{\”timings\”:[*230*,*289*,*354*,...],\”errors\”:*0*},\”timestamp\”:*123456*,\”settings\”:{\”interval_min\”:*1000*,\”inteval_max\”:*5000*,\”count\”:*20*}}”}*

Таким образом осуществляется информационный обмен между элементами РПС.

Заключение

Предложенный подход реализует парадигму, в соответствии с которой сбор данных отделен от их обработки. Благодаря этому можно собирать данные удаленно и с помощью оптимальных средств их сбора, а обработку можно выполнять с помощью других специализированных аппаратных и программных средств, нацеленных на высокопроизводительную обработку данных.

Библиографический список

1. *Yakovlev A.* et al. Automated system for obtaining and mining data for machine learning in evaluation of the specialists' functional state // Machine learning in analysis of biomedical and socio-economic data: collection of scientific papers / edited by X. A. Naidenova, K. V. Shvetsov, A. V. Yakovlev, V. A. Parkhomenko. St. Petersburg: POLYTECH-PRESS, 2020. P. 204–235.

2. *Яковлев А. В., Найденова К. А.* Концепция использования технологии больших данных в современной медицине // Известия Российской Военно-медицинской академии. 2018. Т. 37, № 1. С. 17–23.

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ И ПРОГРАММНАЯ ИНЖЕНЕРИЯ

УДК 004.93 **DOI: 10.31799/978-5-8088-1701-2-2022-2-101-105**

Р. С. Зулкашев*
студент
М. Д. Поляк*
старший преподаватель
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОПРЕДЕЛЕНИЕ НАЛИЧИЯ ВЫСШЕГО ОБРАЗОВАНИЯ ПО ФОТОГРАФИИ ЛИЦА

Рассматривается возможность определения уровня образования человека с помощью информативных признаков, полученных из изображения его лица.
Ключевые слова: *FaceNet, машина опорных векторов, распознавание лиц.*

R. S. Zulkashev*
Student
M. D. Polyak*
Senior Lecturer
**St. Petersburg State University of Aerospace Instrumentation*

DETECTION OF EDUCATION LEVEL FROM FACE IMAGES

In this article we review an algorithm for detection of education level of a person by analysis of face images.
Keywords: *FaceNet, support vector machine, face recognition.*

Введение

Технологии распознавания лиц стремительно развиваются, и относительно недавно появились хорошие методики выделения информативных признаков из изображений [1, 2]. По похожему принципу можно идентифицировать и классифицировать фотографии. Например, уже существуют программы, определяющие человека по его походке [3] или диагностирующие наследственные заболевания по снимку лица [4]. В связи с этими достижениями была выдвинута гипотеза о возможности извлечения различного рода информации по фотографии человека. Исследования по распознаванию преступников по фото лица проводились китайскими учеными в 2016 г. [5] и по определению политических пристрастий американским исследователем в 2021 г. [6]. Перечисленные работы давали хорошие результаты: 89,5% и 72,0% точности соответственно. В данной работе рассматривается другой частный случай, а именно возможность определить наличие высшего образования (ВО) по фотографии лица человека.

Используемая модель

Для извлечения информативных признаков из фотографии были использованы результаты, представленные в оригинальной статье по модели FaceNet 2015 г. [1], а именно ее готовая реализация Facenet by David Sandberg [8]. Используемая предобученная нейронная сеть на наборе данных (датасете) CASIA-WebFace для задачи идентификации (поиск нужного человека среди множества изображений) дает точность (accuracy) 0,9905.

Принцип работы указанной сверточной нейронной сети заключается в том, что на вход подается изображение 160×160 пикселей, по которому строится вектор признаков, состоящий из 512 чисел. Данный вектор, называемый эмбеддингом, представляет собой сжатую версию исходного изображения, где под сжатием понимается уменьшение размерности с 160×160×3 до 512 чисел с сохранением информативности полученных признаков. Получившийся вектор подается на вход классификатора, который выдает вероятности отнесения к одному из двух

классов: «человек с высшим образованием» и «человек без высшего образования».

Подготовка данных

Для обучения классификатора собран набор данных, основанный на статистике наличия высшего образования среди различных слоев населения. В него вошли следующие множества фотографий лиц людей.

1. Преступники. По статистике только 9% совершивших кражу, насилие или убийство имеют высшее образование [7].

2. Актрисы взрослого кино.

3. Медийные личности (каждая персона проверялась отдельно).

4. Ученые, профессора, лауреаты Нобелевской премии.

Данные для обучения собирались по принципу улучшения репрезентативности выборки, т. е. каждый новый пласт фотографий вводился для исправления недостатков предыдущих. Например, вначале среди фотографий преступников оказалось мало женщин, и, чтобы сделать выборку независимой по половому признаку, потребовалось добавить категорию девушек без высшего образования. Для этого были выбраны актрисы взрослого кино. Таким же образом датасет был дополнен медийными личностями (с высшим образованием и без) с целью уменьшения смещения в сторону старшего возраста, который дал пласт профессоров.

Собранный набор данных имеет две версии. В первую вошли 230 фотографий без категории «медийные личности». Во вторую версию вошли все категории, а размер датасета расширился до 430 снимков. Такое разделение понадобилось для оценки влияния расширения набора данных на качество работы модели.

Собранные данные имеют одинаковое количество экземпляров как первого, так и второго классов. Также соблюдено одинаковое соотношение мужчин и женщин, а предпочтение отдавалось фотографиям с наименьшим количеством косметических украшений и с наибольшим разрешением. Но у датасета есть и недостатки, при сборе не учитывались эмоции лица, наличие бороды или усов, очков.

Построение классификатора

В качестве классификатора используется «машина опорных векторов», или коротко SVM [9]. Для подбора оптимальных гиперпараметров (значения модели, которые устанавливаются вручную перед запуском) использовался поиск по сетке с кросс-валидацией с оценкой по F1-мере [10]. Получились следующие оптимальные гиперпараметры модели:

$$SVC\left(kernel = 'rbf',\ C = 0.3,\ gamma = 'scale'\right),$$

где *Kernel* – ядерная функция, *rbf* – ядро радиальной базисной функции, *C* – параметр регуляризации, *gamma* – коэффициент ядра равный $1 / (n_{features} * X.var())$.

В качестве метрик используются accuracy, F1 и AUC ROC. Так как датасет является сбалансированным, то accuracy можно использовать в качестве оценки работы модели.

Далее классификатор был обучен на разных версиях набора данных. Для этого из датасета была выделена тестовая выборка, равная 20% от всего объема датасета, что в итоге дало 46 экземпляров изображений (23 на каждый из классов) для первой версии набора данных и 86 (43 на каждый из классов) для второй.

Для первой версии датасета (табл. 1) были получены следующие значения метрик на тестовой выборке: accuracy 0,804, F1-score 0,800, AUC ROC 0,858.

Таблица 1

Матрица ошибок для первого варианта датасета

Исходный класс	Распознанный класс	
	Есть ВО	Нет ВО
Есть ВО	19	4
Нет ВО	5	18

Для второй версии (табл. 2) результаты оказались следующими: accuracy 0,779, F1-score 0,782, AUC ROC 0,854.

Таблица 2

Матрица ошибок для второго варианта датасета

Исходный класс	Распознанный класс	
	Есть ВО	Нет ВО
Есть ВО	33	10
Нет ВО	9	34

Анализ результатов

Рассмотрим результаты визуализации датасета с помощью алгоритмов уменьшения размерности PCA и T-SNE [11, 12]. Данные алгоритмы уменьшают количество информативных признаков с 512 до 2, позволяя отобразить данные на плоскости.

Представленные на рис. 1, 2 графики позволяют сделать выводы о нелинейном характере разделяющей классы поверхности, как мини-

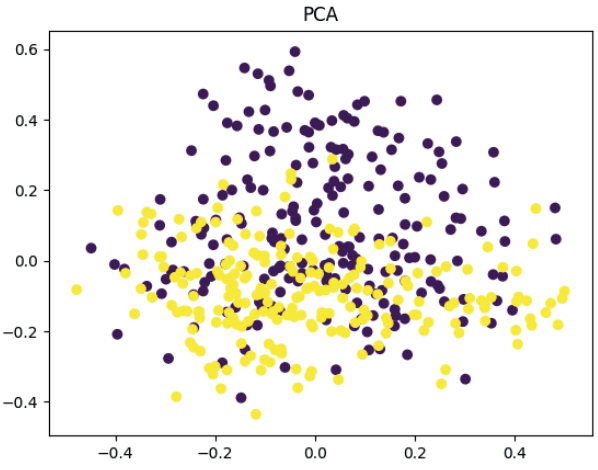


Рис. 1. Результаты PCA

мум при использовании методов уменьшения размерности PCA и T-SNE. Это может свидетельствовать о том, что среди векторов эмбеддингов отсутствуют неинформативные признаки или их количество незначительно.

Как можно было видеть по результатам метрик, классификатор научился находить некоторые закономерности в данных. Чтобы оценить результаты классификации был проведен ручной анализ подмножеств фотографий датасета. Для этого снимки были разделены на три категории (рис. 3–5): неправильно классифицированные, вызывающие сомнение у классификатора (вероятность отнесения к конкретному классу < 0,7) и правильно классифицированные без сомнений со стороны классификатора.

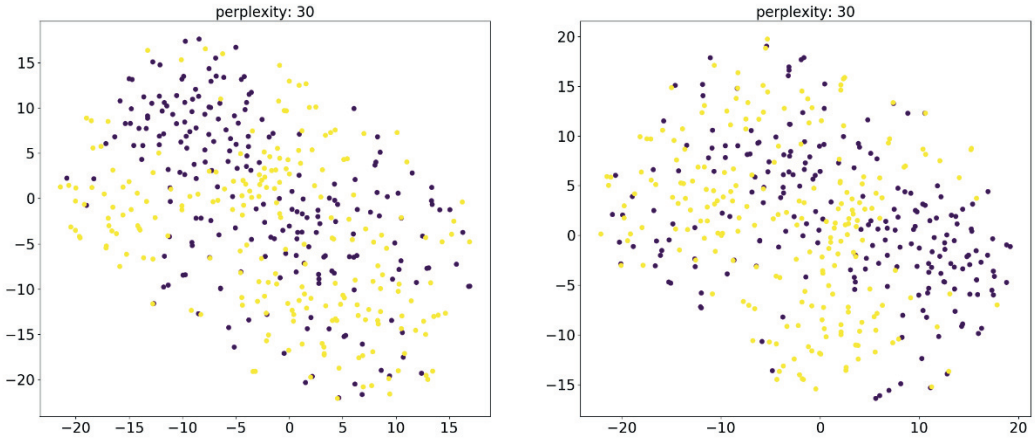


Рис. 2. Результаты T-SNE

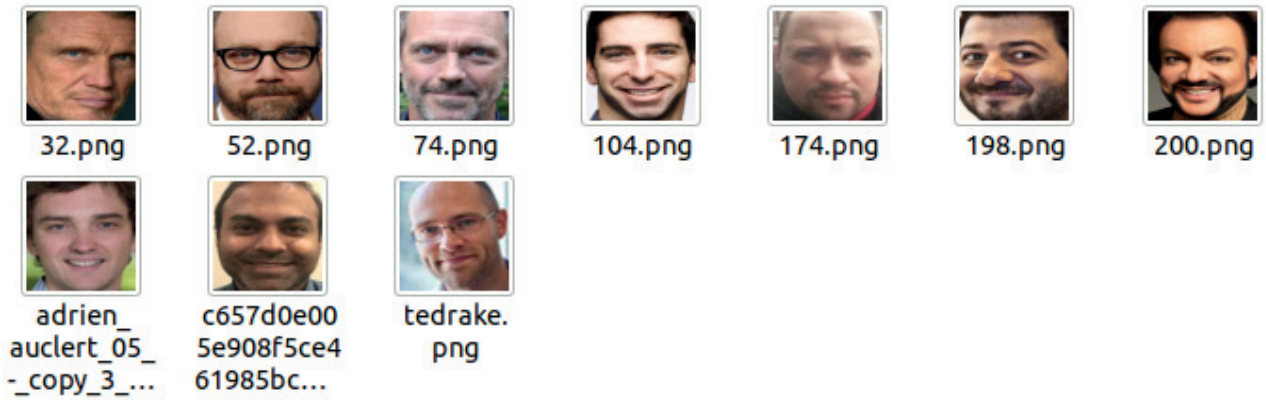


Рис. 3. Неправильно классифицированные: с высшим образованием

Третья категория изображений из ручного анализа исключалась.

Анализ фотографий из тестовой выборки, на которых классификатор сомневается, показывает наличие закономерности: людей с бородой модель относит к людям без высшего образования (7 из 7 людей, которые имеют высшее образование, неправильно классифицированы), но

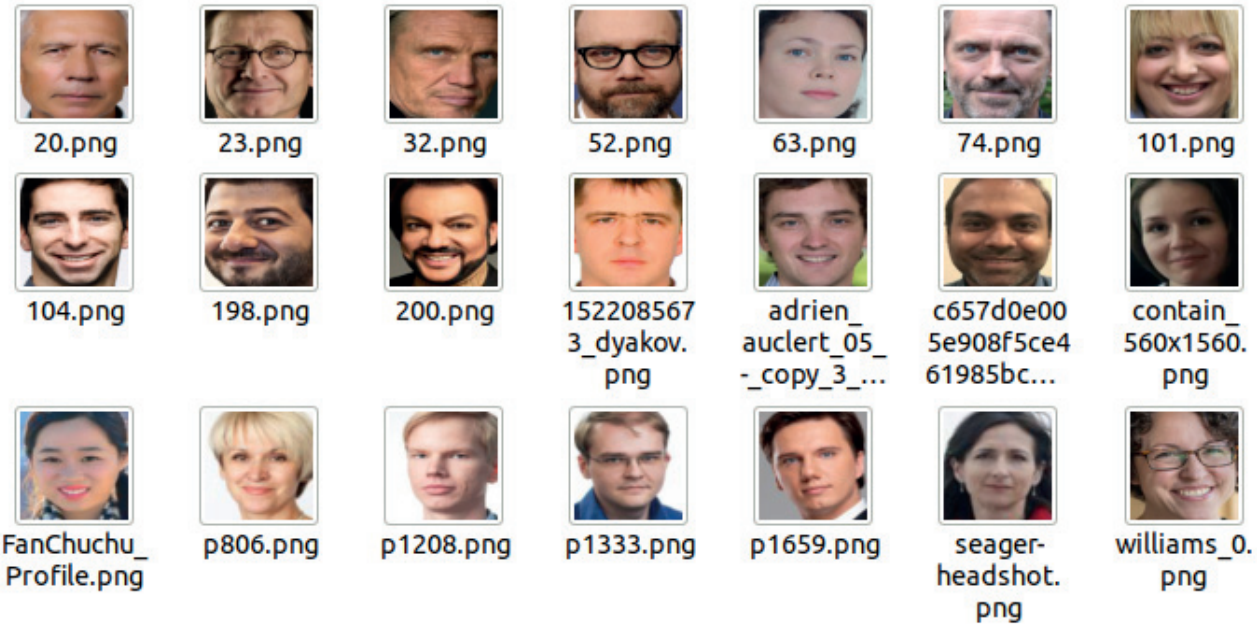


Рис. 4. Вызывают сомнение: с высшим образованием

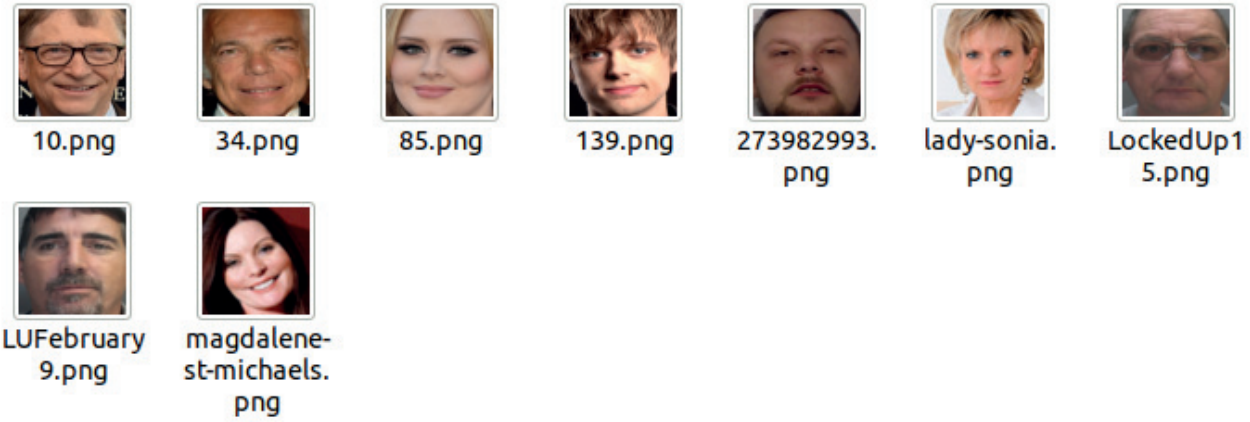


Рис. 5. Неправильно классифицированные: без высшего образования

не наоборот, на безбородых классификатор хоть и сомневается, но не относит их к людям с высшим образованием (4 из 12 людей, которые не имеют высшего образования неправильно классифицированы и на 10 из 12 модель сомневалась). Из этого можно сделать вывод, что выборка является смещенной, и имеет смысл расширить датасет с учетом фактора наличия/отсутствия бороды.

Результаты и выводы

В ходе работы с помощью модели машинного обучения получилось добиться 78% точности при классификации людей на две группы – с высшим образованием и без высшего образования. Можно сделать вывод, что, несмотря на смещенную выборку, классификатор обучился находить некоторые закономерности между фотографией человеческого лица и наличием высшего образования. Следовательно, гипотезу о возможности определения наличия высшего образования по фотографии лица с помощью методов машинного обучения можно считать подтвержденной. Выборка была небольшой, и для уверенности в полученных результатах стоит расширить набор данных с учетом найденных недостатков и провести повторное обучение модели.

Модель сомневается на людях, у которых нет диплома, но которых тем не менее нельзя с полной уверенностью отнести к классу «без высшего образования», например Стив Джобс, Билл Гейтс, Квентин Тарантино. Данный факт ско-

рее подтверждает работоспособность модели и может свидетельствовать о том, что модель способна улавливать закономерности, неочевидные для человека.

Библиографический список

1. *Schroff F., Kalenichenko D., Philbin J.* FaceNet: A unified embedding for face recognition and clustering. 2015 IEEE Conference on Computer Vision and Pattern Recognition.

2. *Simonyan K., Zisserman A.* Very Deep Convolutional Networks for Large-Scale Image Recognition. 2015.

3. *Соколова А. И., Конушин А. С.* Методы идентификации человека по походке в видео // Труды ИСП РАН. 2019. № 1. URL: https://cyberleninka.ru/article/n/metody-identifikatsii-cheloveka-po-pohodke-v-video (дата обращения: 25.11.2021).

4. *Gurovich Y., Hanani Y., Bar O.* et al. Identifying facial phenotypes of genetic disorders using deep learning // Nat. Med. 2019. Vol. 25. P. 60–64. URL: https://doi.org/10.1038/s41591-018-0279-0 (дата обращения: 25.11.2021).

5. *Wu X., Zhang X.* Responses to Critiques on Machine Learning of Criminality Perceptions. 2017.

6. *Kosinski M.* Facial recognition technology can expose political orientation from naturalistic facial images // Sci Rep. 2021. Vol. 11. P. 100. URL: https://doi.org/10.1038/s41598-020-79310-1 (дата обращения: 25.11.2021).

7. Российские преступники стали старше и образованнее // Известия. URL: https://iz.ru/news/693290 (дата обращения: 25.11.2021).

8. Face Recognition using Tensorflow // Github. URL: https://github.com/davidsandberg/facenet (дата обращения: 25.11.2021).

9. Support Vector Machines // Scikit-learn. URL: https://scikit-learn.org/stable/modules/svm.html (дата обращения: 25.11.2021).

10. GridSearchCV // Scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (дата обращения: 25.11.2021).

11. PCA // Scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html (дата обращения: 25.11.2021).

12. TSNE // Scikit-learn. URL: https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html (дата обращения: 28.11.2021).

УДК 621.398(075.8) **DOI: 10.31799/978-5-8088-1701-2-2022-2-106-108**

А. А. Ключарёв*
кандидат технических наук, доцент
А. А. Фоменкова*
аспирант
А. Д. Ельцова*
магистрант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

СИСТЕМА КОНТРОЛЯ ТЕХНИЧЕСКОГО СОСТОЯНИЯ АНАЭРОБНОГО БИОРЕАКТОРА

Предлагается система контроля технического состояния 5-секционного анаэробного биореактора для очистки сточных вод. На основе результатов численного моделирования выбраны контрольные точки для измерения диагностических признаков в каждой из секций аппарата, позволяющие выявить причины деградации биомассы в процессе анаэробного брожения и обоснованно принимать решения по недопущению аварийных ситуаций.
Ключевые слова: *анализ технического состояния, диагностические признаки, работоспособность, система биологической очистки сточных вод, анаэробная биомасса.*

A. A. Klucharev*
PhD, Tech., Associate Professor
A. A. Fomenkova*
PhD Student
A. D. Yeltsova*
Postgraduate Student
**St. Petersburg State University of Aerospace Instrumentation*

ANAEROBIC BIOREACTOR TECHNICAL STATE MONITORING SYSTEM

A system for monitoring the technical condition of a 5-section anaerobic bioreactor for wastewater treatment is proposed. Based on the results of numerical modeling, control points for measuring diagnostic signs in each of the sections of the apparatus were selected, that allows to identify the causes of biomass degradation in the process of anaerobic digestion and to justify making decisions to prevent emergency situations.
Keywords: *analysis of technical state, diagnostic signs, performance, biological wastewater treatment system, anaerobic biomass.*

Одна из важных тенденций развития инфраструктуры промышленных предприятий, направленная на повышение их экологической безопасности, – использование локальных систем очистки сточных вод. В последнее время разрабатываются и усовершенствуются методы очистки сточных вод, внедряется и модернизируется технологическое оборудование для проведения очистки. Функционирование этого оборудования требует непрерывного контроля его технического состояния с целью как соблюдения необходимых параметров технологического процесса, так и обеспечения экологической безопасности производства, требования к которой постоянно ужесточаются. В связи с этим задача непрерывного анализа технического состояния системы очистки сточных вод предприятия актуальна и имеет важное практическое значение как в процессе эксплуатации, так и при принятии решений по предотвращению аварийных ситуаций.

Для предприятий пищевой промышленности важным и наименее исследованным элементом системы очистки сточных вод является анаэробный биореактор, функционирование которого во многом зависит от состояния биомассы, выполняющей разложение органических отходов. Однако состояние микроорганизмов не может быть непосредственно оценено инструментальными методами. Мы разработали алгоритмы непрерывного контроля работоспособности системы анаэробной очистки, которые на основе доступных к измерению параметров позволяют оценить ее техническое состояние с учетом состояния биомассы [1]. В общем слу-

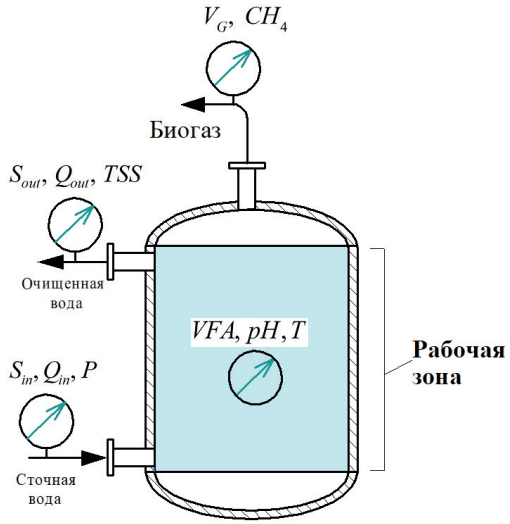


Рис. 1. Измеряемые параметры при анализе ТС САБО

чае для оценки технического состояния системы анаэробной очистки наблюдению подлежат следующие параметры (рис. 1):

– входные параметры: концентрация органических загрязнений в сточной воде S_{in}, расход Q_{in} сточных вод, поступающих на очистку, давление P насосного оборудования;

– выходные параметры: концентрация загрязнений в очищенной воде S_{out}, расход Q_{out} сточных вод на выходе из биореактора, производительность системы по биогазу V_G и содержание метана в выработанном биогазе CH_4, концентрация взвешенных веществ в очищенной воде TSS;

– параметры в рабочей зоне анаэробного биореактора: рабочая температура процесса T, концентрация летучих жирных кислот VFA, кислотность среды в биореакторе pH.

Применительно к конкретной конструкции анаэробного биореактора алгоритмы оценки ТС должны быть уточнены, в частности необходимо определить точки, в которых измеряются выделенные диагностические признаки.

Для очистки сточных вод предприятия молочной промышленности, содержащих липофильные вещества и молочную сыворотку, предлагается использовать гибридный секционный биореактор с прикрепленной биомассой [2]. Особенность этой конструкции состоит в пространственном распределении различных этапов анаэробного брожения и необходимости раздельного анализа технического состояния каждой из секций аппарата, на основе которого принимается решение о техническом состоянии всего биореактора. Особенности подобной конструкции накладывают ограничения на измерение некоторых из диагностических признаков. Ввиду преобладания того или иного этапа анаэробного брожения в секции информативными могут оказаться различные параметры, а допустимые для работоспособного состояния диапазоны их значений по секциям будут различаться.

При расчете основных конструктивных параметров анаэробного биореактора для очистки сточных вод предприятия молочной промышленности с заданным среднесуточным расходом сточных вод Q, спроектирован 5-секционный аппарат. Ввиду особенностей организации рабочей зоны секции, а именно наличия загрузки из плоскостного материала, на которой нарастает биопленка, измерение параметров VFA и pH должно реализовываться на выходе из секции. Рабочая температура в секции измеряется в середине канала между носителями биомассы, удаленно от стенок аппарата. Расход сточной воды Q_{out} и концентрацию взвешенных веществ TSS нецелесообразно измерять для каждой секции, эти параметры контролируются на выходе из биореактора.

С целью уточнения набора диагностических признаков при анализе технического состояния рассматриваемого анаэробного биореактора был проведен имитационный эксперимент, на основе которого сформированы диапазоны значений диагностических признаков. Для каждой i-й секции, $i = 1…5$, оценивались значения S_{in_i}, S_{out_i}, $VFA_{\_i}$, T_i, $pH_{\_i}$, V_{G_i}, CH_{4_i}. Ввиду того, что выход одной секции является входом последующей секции, принято, что $S_{in\_1} = S_{in}$, $S_{out\_5} = S_{out}$, для $i = 2…5$, $S_{in\_i} = S_{out\_i\text{-}1}$. Дальнейший выбор минимального достаточного набора диагностических признаков для оценки технического состояния секционного анаэробного биореактора проводился в соответствии с методологией технической диагностики [3, 4]. Для 5-секционного анаэробного биореактора получено, что концентрация летучих жирных кислот VFA является информативным признаком при ее измерении в 2–4-й секциях. Концентрацию органических загрязнений достаточно измерять на входе в биореактор (параметр S_{in}) и на выходе из него (параметр S_{out}). Для первой секции содержание метана в биогазе не позволяет судить о работоспособности секции.

В результате предложена общая структура контроля технического состояния рассматриваемого 5-секционного анаэробного биореактора, учитывающая состояние анаэробной биомассы (рис. 2).

В предложенной структуре каждая секция рассматривается как отдельный биореактор. Сигнализация о сбое в работе секции происходит изменением цвета ее заливки. На экран при

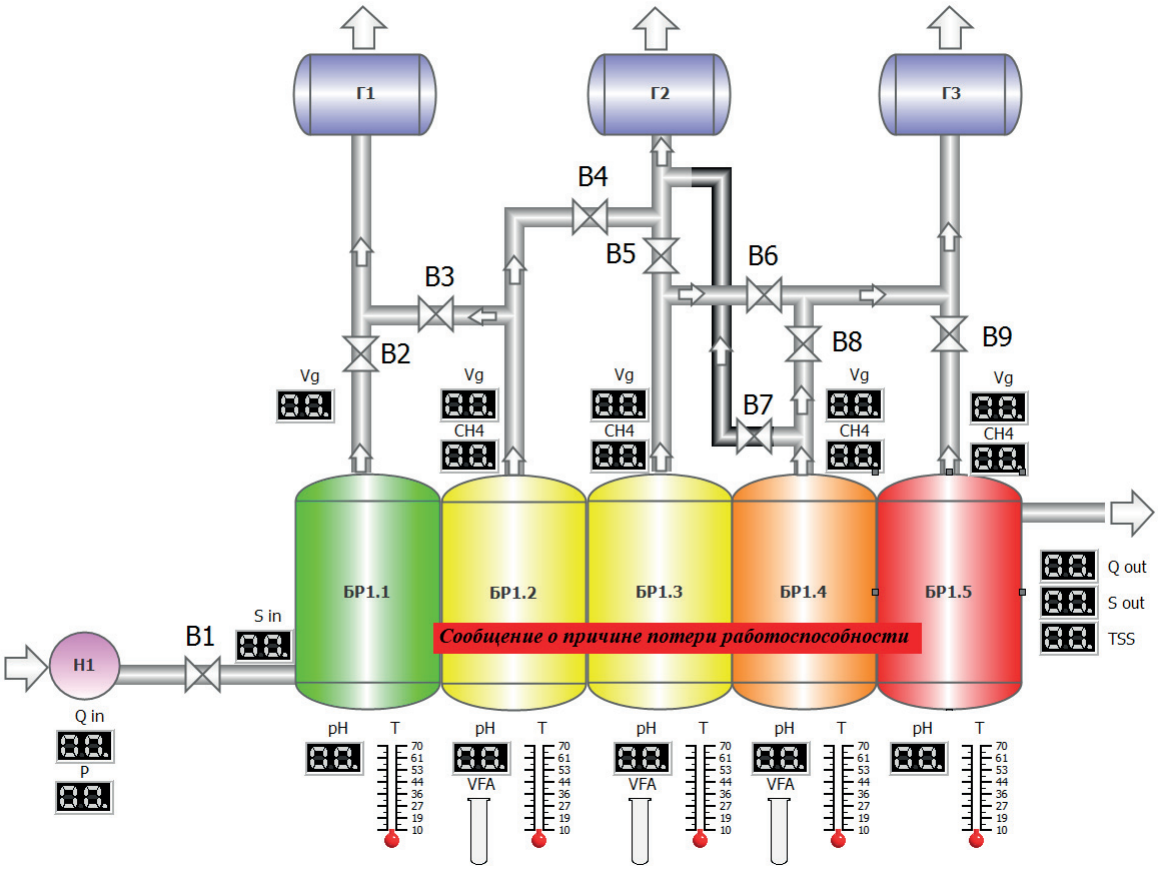


Рис. 2. Структура системы контроля технического состояния 5-секционного анаэробного биореактора

этом выводится сообщение о причине потери работоспособности. Измеренные автоматизированными средствами контроля значения диагностических признаков выводятся в соответствующих окнах. Параметр *VFA*, для контроля за которым необходимо проведение лабораторных исследований, в соответствии с разработанными алгоритмами [1] анализа технического состояния биореактора подлежит измерению в ограниченном количестве случаев. При необходимости проведения этого анализа на схеме выводится соответствующее сообщение. В соответствии с рис. 2, значение объемной концентрации метана CH_4 в образованном биогазе служит не только информативным признаком при оценке технического состояния биореактора, но и позволяет организовать раздельный сбор биогаза из секций для дальнейшего его обогащения, что повышает экономическую эффективность системы анаэробной биологической очистки.

Таким образом, алгоритмы оценивания технического состояния системы анаэробной очистки сточных вод применимы для выявления причин угнетения биомассы в различных конструкциях биореактора. Для конкретной конструкции уточнение этих алгоритмов сводится к определению количества и расположения контрольных точек для измерения выделенных диагностических признаков.

Библиографический список

1. *Фоменкова А. А.* Выбор минимального набора диагностических признаков при анализе технического состояния биомассы в системе биологической очистки сточных вод // Обработка, передача и защита информации в компьютерных системах: Междунар. науч. конф. (СПб., 14–22 апр. 2021 г.): сб. докл. СПб.: ГУАП. 2021. С. 82–86.

2. *Ключарёв А. А., Фоменкова А. А.* Проектирование секционного анаэробного биореактора // Известия Санкт-Петербургского государственного технологического института (технического университета). 2018. № 34 (60). С. 95–100.

3. *Дмитриев А. К., Юсупов Р. М.* Идентификация и техническая диагностика. М.: М-во обороны СССР, 1987. 521 с.

4. *Копкин Е. В., Кравцов А. Н., Мышко В. В.* Анализ технического состояния космических средств. СПб.: ВКА им. А. Ф. Можайского, 2016. 190 с.

УДК 378.1:004.65 **DOI: 10.31799/978-5-8088-1701-2-2022-2-109-112**

Н. В. Путилова
старший преподаватель
Санкт-Петербургский государственный университет аэрокосмического приборостроения

АВТОМАТИЗИРОВАННАЯ РАЗРАБОТКА ОБРАЗОВАТЕЛЬНЫХ ПРОГРАММ В УСЛОВИЯХ ЦИФРОВОЙ ТРАНСФОРМАЦИИ ВУЗА

Рассматриваются принципы цифровой трансформации вуза применительно к образовательному процессу. Предложены подход к цифровой трансформации автоматизированной разработки образовательных программ и модель темпоральной базы данных для хранения цифрового следа при актуализации образовательных программ и изменении данных сотрудников, ответственных за эти программы.
Ключевые слова: *разработка образовательных программ, высшее образование, цифровая трансформация, системы управления базами данных, СУБД, темпоральные базы данных, автоматизация.*

N. V. Putilova
Senior Lecturer
St. Petersburg State University of Aerospace Instrumentation

AUTOMATED CURRICULA DESIGN IN THE CONTEXT OF DIGITAL TRANSFORMATION OF THE UNIVERSITY

The article discusses the principles of digital transformation of the university in relation to the educational process. An approach to digital transformation of automated curricula design and a model of a temporal database for storing a digital footprint of curricula updates including involved employees' data changes are proposed.
Keywords: *Curriculum design, higher education, digital transformation, database management system, DBMS, temporal databases, automation.*

В условиях пандемии вузы, как и многие организации, вынуждены переводить большую часть работы в дистанционный режим. Однако многие из составляющих образовательного процесса на текущий момент плохо подходят для удаленной реализации. Понятие цифровой трансформации достаточно расплывчато, и чаще всего под ним имеют в виду сочетание цифровизации и автоматизации различных процессов какой-либо организации, позволяющее перевести максимально возможную часть работы в электронный формат для получения конкурентных преимуществ. В таких условиях задача цифровой трансформации вуза становится все более актуальной. Часто цифровую трансформацию вуза рассматривают только с точки зрения реализации обучения с использованием дистанционных образовательных технологий и электронных средств обучения [1, 2] или понимают как автоматизацию отдельных процессов [3]. Даже если рассматривать только образовательную составляющую, которая является предметом данной статьи, задача цифровой трансформации более глобальная. Ее можно сформулировать как автоматизацию и цифровизацию управления образовательным процессом вуза на всех его этапах с возможностью анализа эффективности. Можно выделить следующие этапы организации процесса обучения: анализ реализации существующих образовательных программ (ОП), разработка новых ОП, реализация ОП, включающая их регулярную актуализацию, формирование расписания и расчет нагрузки на текущий учебный год или семестр и непосредственно обучение. Завершающее действие – анализ реализации ОП с целью определения необходимых изменений в образовательном процессе для достижения максимальной эффективности.

На текущий момент многие этапы обучения имеют цифровую составляющую, однако ни один процесс не переведен в цифровой формат полностью. Это связано с внешними нормативными требованиями, с особенностями организации обучения в вузах, а также с высокой степенью изменчивости в современном мире. Для полной цифровой трансформации образования необходимо, чтобы составляющие, автоматизация или цифровизация которых невозможна, получили свой цифровой след, а особенности

современного высшего образования были отражены в специализированных математических и программных моделях.

Из обозначенных задач разработка и анализ ОП требуют наиболее серьезной проработки для цифровой трансформации, так как являются основополагающими для других действий и при этом не имеют готовых универсальных решений. Несмотря на использование автоматизированных систем при разработке ОП, имеются трудности в приведении ее к полностью цифровому виду. Разработка ОП – многоступенчатый комплексный процесс, автоматизированный не на всех этапах, за различные составляющие которого отвечают разные исполнители, что в результате осложняет передачу информации между этапами и исполнителями. Это в свою очередь ведет к потере информации о состоянии разработки и трудностям в анализе информации в целом. Существует и другая проблема: системы, связанные с разработкой ОП, хранят только текущее состояние элементов ОП и теряется цифровой след актуализации ОП. Исходя из этого, можно сделать вывод о необходимости создания новой системы для работы с ОП. Для решения обозначенных проблем система должна удовлетворять следующим требованиям:

– преемственность с уже существующими системами,

– модульная архитектура для поддержки различных этапов разработки ОП и различных исполнителей,

– ведение истории изменений учебных планов, рабочих программ дисциплин и практик,

– сохранение причин изменений составляющих образовательной программы,

– возможность получения аналитических данных для оперативного управления образовательным процессом,

– удобство сквозного поиска данных во всех ОП,

– гибкость при определении и изменении структуры учебного плана,

– информационная и управляющая связь между всеми участниками организации процесса обучения,

– общие правила и форматы представления данных на всех этапах,

– управление основными данными образовательного процесса.

Обязательность цифрового следа требует хорошо выстроенной системы управления основными данными и отслеживания их изменений во времени. При разработке ОП целесообразно рассматривать учебный план и рабочие программы дисциплин и практик как отдельные, но целостные объекты, так как изменения одних параметров или составляющих при неизменных требованиях приведет к изменению других. В таком случае удобно использовать систему контроля версий (СКВ). Однако это противоречит требованию сквозной доступности информации, так как системы контроля версий работают с хранением состояния отдельного файла, а сквозной поиск легко реализуем в базе данных, но затруднен при работе с отдельными файлами. Таким образом, необходима гибридная система, сочетающая работу с базой данных и СКВ. Возможны два подхода к реализации такой системы: разработка СКВ, работающей с данными, хранящимися в базе данных, и отдельное существование баз данных и СКВ с дополнительным модулем синхронизации и контроля.

Оба подхода имеют достоинства и недостатки и требуют реинжиниринга процессов управления ОП. Разработка собственной СКВ сопряжена с большими трудозатратами и длительным временем разработки, однако позволяет за счет более специализированного подхода создать единую систему, удовлетворяющую всем требованиям и не имеющую проблем при передаче данных. Подход, использующий существующую систему управления базами данных и СКВ, существенно снижает трудозатраты, но требует более проработанной архитектуры и модуля контроля с большим количеством диагностических функций. Также при этом подходе осложнена синхронизация данных из-за дублирования информации в базе данных и СКВ.

В данный момент трудозатраты на разработку полноценной СКВ и необходимое для этого время не позволяют осуществить цифровую трансформацию ГУАП в рамках сроков Приоритета 2030, а также для сохранения преемственности с существующими в ГУАП системами целесообразно применить решение с модулем синхронизации между базой данных и СКВ. При этом необходимо определить, какие данные будут контролироваться СКВ и направление синхронизации данных. Одна из основных причин использования СКВ применительно к образовательному процессу в вузе – необходимость своевременной актуализации ОП и ее составляющих для удовлетворения требованиям нормативной документации и соответствия современному состоянию науки и техники. ОП содержит учебный план, календарный график обучения, рабочие программы дисциплин, практик и программу государственной итоговой аттестации. Каждый из этих объектов является цельным документом с отдельным разработчиком и может быть представлен в виде отдельного файла. Та-

ким образом, удобно представлять ОП как проект в СКВ. Также временным изменениям подвержены данные о сотрудниках вуза, включая информацию о сотрудниках, связанных с разработкой и утверждением ОП. Однако выделение файла для одного сотрудника нецелесообразно из-за возникающего дублирования информации в различных проектах ОП. Изменения этих данных удобнее отслеживать через темпоральную базу данных. Существуют различные модели представления временных данных в зависимости от того, строке или атрибуту сопоставляется

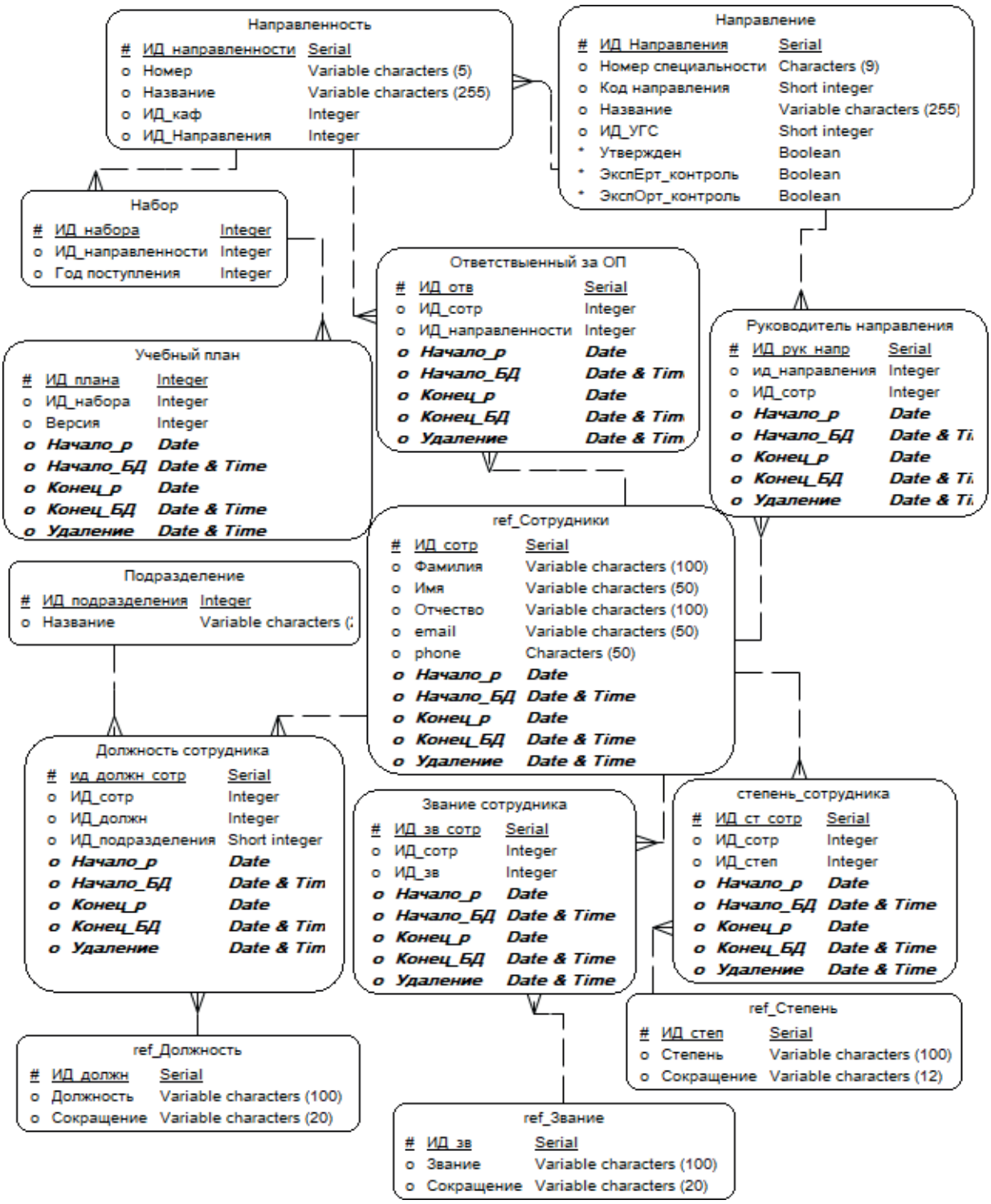


Рис. 1. Схема темпоральной базы данных сотрудников, ответственных за ОП

период актуальности, и форм сохранения периода актуальности временных данных. В данном случае наиболее подходящей является модель Дж. Бен-Зви (рис. 1).

В модели Бен-Зви каждой строке в таблице, содержащей темпоральные данные, добавляются атрибуты, содержащие реальное и транзакционное (записи информации в базу данных) время начала и окончания периода актуальности строки, а также время логического, еще называемого «мягким», удаления данных из базы [4]. Требование наличия цифрового следа для всех изменений исключает физическое удаление, а сами параметры сотрудников или свойства версии учебного плана не меняются отдельно друг от друга.

Возможность получения аналитических данных – вторая из важнейших составляющих цифровой трансформации. Аналитика при разработке ОП с учетом индивидуальных образовательных траекторий включает следующие группы:

– проверка ОП на соответствие внешним и внутренним нормативным документам,

– определение логической целостности и непротиворечивости структуры ОП,

– анализ предпочитаемых студентами элективных дисциплин,

– анализ рентабельности ОП,

– оценка изменений нагрузки вуза по годам и семестрам,

– оценка загрузки кафедр по семестрам,

– оценка удобства совместной реализации ОП,

– определение возможностей академической мобильности между ОП.

Часть аналитики предназначена для принятия стратегических для вуза решений, производится по истечении некоторого периода и не имеет жестких требований к времени выполнения, однако другая часть, связанная с определением характеристик ОП и оперативным управлением образовательным процессом, требует получения результатов в реальном времени. Если для первой части аналитики можно создать классическую аналитическую базу данных (OLAP), то для второй части она не подходит из-за задержки в обработке данных. Поскольку для разработки ОП и оперативного управления образовательным процессом используются транзакционные базы данных (OLTP), слабо подходящие для аналитики, возникает необходимость применения для оперативного управления либо резидентных транзакционных СУБД, обеспечивающих высокую скорость обработки данных за счет реализации всей работы в оперативной памяти, либо систем гибридной транзакционной/аналитической обработки (hybrid transactional/ analytical processing, HTAP) различной архитектуры [5]. Выбор конкретной СУБД зависит от архитектуры реализации остальных модулей системы для обеспечения простоты взаимодействия с ними.

Модернизация системы работы с ОП в соответствии с предложенными изменениями позволит с наименьшими затратами труда и времени получить систему, сохраняющую цифровой след актуализации ОП и позволяющую анализировать изменение различных параметров и элементов ОП во времени. Следующим этапом полной цифровой трансформации образовательного процесса будет создание систем, автоматизирующих работу над составляющими ОП (например, очень незначительно автоматизирована разработка рабочих программ дисциплин, практик и государственной итоговой аттестации), на текущий момент выполняемую в ручном режиме, а также модулей, осуществляющих аналитику и контроль передачи информации между различными этапами реализации образовательного процесса.

Библиографический список

1. *Павличева Е. Н., Сосенушкин С. Е., Куприяненко И. А.* Технологические аспекты цифровой трансформации образовательной деятельности вуза в условиях пандемии // XXI век: итоги прошлого и проблемы настоящего плюс. 2021. Т. 10, № 1 (53). С. 40–44.

2. Цифровая трансформация российских вузов: первый опыт / Н. Х. Савельева, Е. А. Гнатышина, Н. В. Уварина [и др.] // Азимут научных исследований: педагогика и психология. 2021. Т. 10, № 1 (34). С. 226–229.

3. *Петрова Е. С., Правосудов Р. Н., Правосудов А. Р.* Автоматизация разработки ОПОП ВО как фактор цифровой трансформации ВУЗа // Новые информационные технологии в образовании: сб. науч. тр. 21-й Междунар. науч.-практ. конф., М., 2–3 февр. 2021 г. / под общ. ред. Д. В. Чистова. М.: 1С-Паблишинг, 2021. С. 26–31.

4. *Тоноян С. А., Сараев Д. В.* Темпоральные модели базы данных и их свойства // Инженерный журнал: наука и инновации. 2014. № 12 (36). С. 15.

5. *Кузнецов С. Д., Велихов П. Е., Фу Ц.* Аналитика в реальном времени, гибридная транзакционная/аналитическая обработка, управление данными в основной памяти и энергонезависимая память // Труды Института системного программирования РАН. 2021. Т. 33, № 3. С. 171–198.

УДК 004.4 **DOI: 10.31799/978-5-8088-1701-2-2022-2-113-116**

С. А. Рогачев*
старший преподаватель
Д. А. Кочин*
ассистент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ГЕОИНФОРМАЦИОННЫЙ ПОРТАЛ ДЛЯ ПРЕДСТАВЛЕНИЯ РАЗНОРОДНОЙ ПРОСТРАНСТВЕННОЙ ИНФОРМАЦИИ

Рассмотрены современные сервисы для представления и передачи пространственной информации. Приведены концептуальная схема разработанного геоинформационного портала, образцы экранных форм пользовательского интерфейса геоинформационного портала.
Ключевые слова: *геопортал, аэрокосмические изображения, публикация векторных данных, публикация растровых данных.*

S. A. Rogachev*
Senior Lecturer
D. A. Kochin*
Assistant
**St. Petersburg State University of Aerospace Instrumentation*

GEOINFORMATION PORTAL FOR THE REPRESENTATION OF HETEROGENEOUS SPATIAL INFORMATION

The modern services for the representation and transmission of spatial information are considered. The conceptual diagram of the developed geoinformation portal is presented. Samples of screen forms of the user interface of the geographic information portal are presented.
Keywords: *geoportal, aerospace images, publication of vector data, publication of raster data.*

В настоящее время появилось большое количество новых методов сбора данных. Пространственные данные могут быть получены с помощью методов дистанционного зондирования Земли из космоса, наземных исследований, аэросъемки, использования навигационных систем и т. д. С развитием информационных технологий объем данных, скорость их получения и обработки постоянно растет. Данные, полученные перечисленными способами, могут храниться и обрабатываться в цифровом формате, что в свою очередь создает предпосылки для создания сервисов удаленного представления данных потенциальным конечным пользователям [1].

Для удобного, оперативного и наглядного представления пространственных данных конечному потребителю большинство современных компаний и учреждений активно используют геоинформационные технологии [2]. Технология для поиска, передачи и представления пространственной информации имеет название геоинформационный портал (геопортал). Использование геопортальных технологий:

– обеспечивает простоту визуализации разнородных пространственных данных на стороне пользователя;

– упрощает поиск данных, представляющих интерес, без использования специального программного обеспечения;

– предоставляет возможность осуществлять обработку данных без использования вычислительных мощностей конечного пользователя (обработка производится на стороне сервера).

На рис. 1 представлена обобщенная функциональная схема геопортала [1].

Пользователь при работе с геопорталом имеет доступ только к интерфейсу, который позволяет осуществлять подбор и отображение интересующей информации. Сам геопортал хранит информацию о пространственных метаданных, к которым имеется доступ. При таком подходе данные могу храниться распределено на различных геоинформационных серверах, а геопортал в свою очередь объединяет получен-

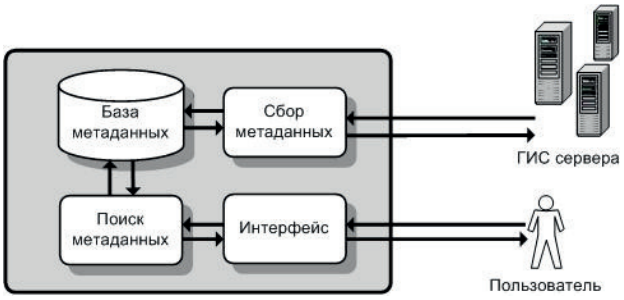


Рис. 1. Обобщенная функциональная схема геопортала

ную разнородную информацию в единый геоинформационный проект, который и представляет конечному пользователю [2].

Практически все пространственные данные можно разделить на две группы по типу хранимой информации: растровые и векторные. В зависимости от типа данных, который нужно предоставить конечному пользователю, изменяется методика его хранения, обработки и передачи.

Разработкой стандартов в области геопространственных данных и сервисов занимается международная некоммерческая организация Open Geospatial Consortium (OGC). Основные ее цели – разработка и внедрение открытых стандартов по работе с геопространственными данными в геоинформационных системах, а также стандартов совместного использования данных [3].

В базовый набор OGC входит более 30 стандартов. Один из самых распространенных – OGC Web Service (OWS), который скрывает различные сервисы доставки геопространственных данных, такие как:

– Web Feature Service (WFS), предназначенный для доставки векторных данных в различных форматах [3, 4]. При этом при передаче информации могут использоваться различные форматы получения данных, а клиентская программа просто получает поток векторных данных, не различая места хранения, которыми могут выступать как базы данных, так и отдельные файлы;

– Web Map Service (WMS), предназначенный для создания карт с заданными стилями оформления [3, 4]. Наиболее часто данный сервис используют для создания топографических карт, которые должны объединить растровые и векторные данные. Таким образом, WMS создает единую карту из различных геопространственных данных и передает результат конечному пользователю;

– Web Coverage Service (WCS), работающий с растровыми данными, которые могут быть описаны как покрытие. Самым распространенный пример – аэрокосмические изображения, модели рельефа и подобные данные, которые имеют значения в регулярной сетке [4];

– Web Processing Service (WPS), представляющий собой интерфейс пользователя к возможностям полноценных геоинформационных систем. Позволяет обрабатывать существующие данные и создавать новые [4].

Все перечисленные сервисы могут быть использованы при проектировании геопортала. Концептуальная схема разрабатываемого геопортала изображена на рис. 2. На схеме присутствует веб-сервер, который хранит проекты в виде конфигурационных файлов в формате JSON. Сервер данных содержит: картографический сервер, сервер данных и сервер системы управления базами данных (СУБД). Картографический сервер в данном случае реализован с помощью открытого программного обеспечения GeoSever, которое соответетствует стандартам OGC [5]. На файловом сервере хранятся разнородные пространственные данные, которые могут быть отображены с помощью картографического сервера. Для полноценного функционирования картографического сервера необходимо использование базы данных под управлением СУБД PostgreSQL с установленным пространственным разрешением PostGIS (обеспечивает возможность обработки географической информации и пространственных запросов) [6].

Для работы с геопорталом конечный клиент может использовать веб-приложение, которое будет обращаться или к веб-серверу (если для доступа к данным необходима авторизация), или напрямую к серверу данных. Для создания и редактирования геоинформационных проектов и данных может быть использована настольная геоинформационная система QuantumGIS (QGIS).

В результате работы был получен картографический веб-сайт (геоинформационный портал) для визуализации пространственной информации. Геопортал предназначен для совместной удаленной работы с геоинформационными проектами. Он позволяет пользователям получать доступ к интерактивной карте и инструментам для работы с картой. На рис. 3 представлен интерфейс пользователя.

Основные возможности для пользователей:

– просмотр карты, навигация, масштабирование;

– управление слоями карты (отображение/скрытие);

– приближение к экстенту слоя;

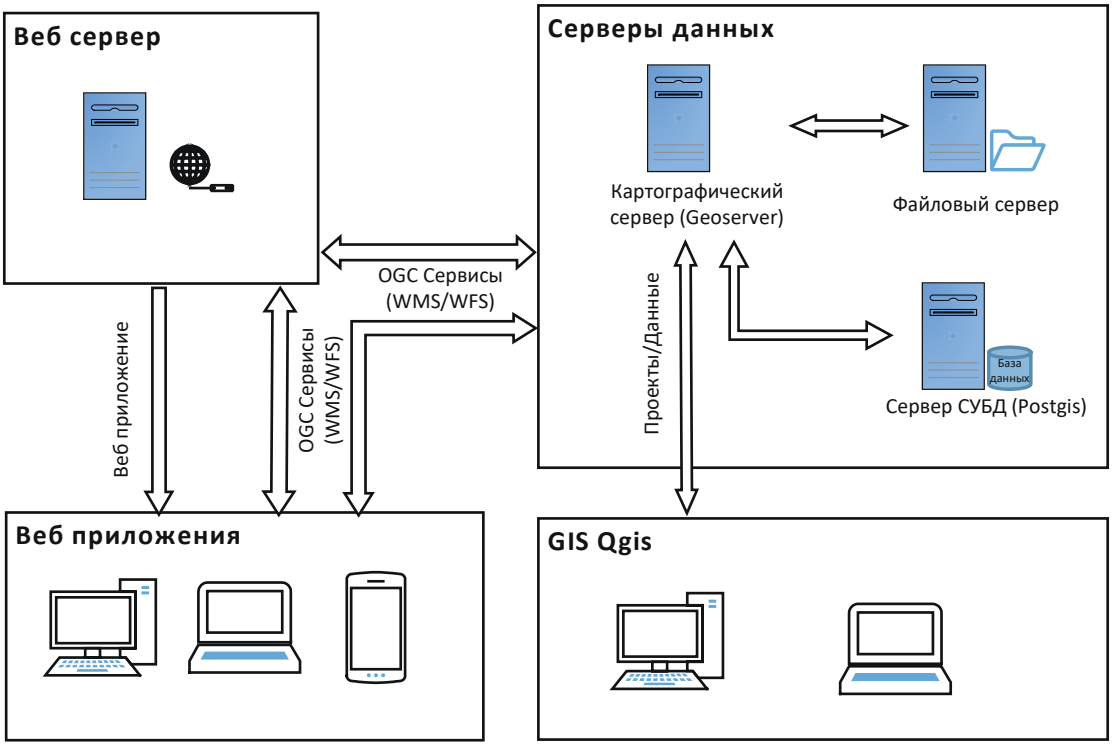


Рис. 2. Концептуальная схема геоинформационного портала

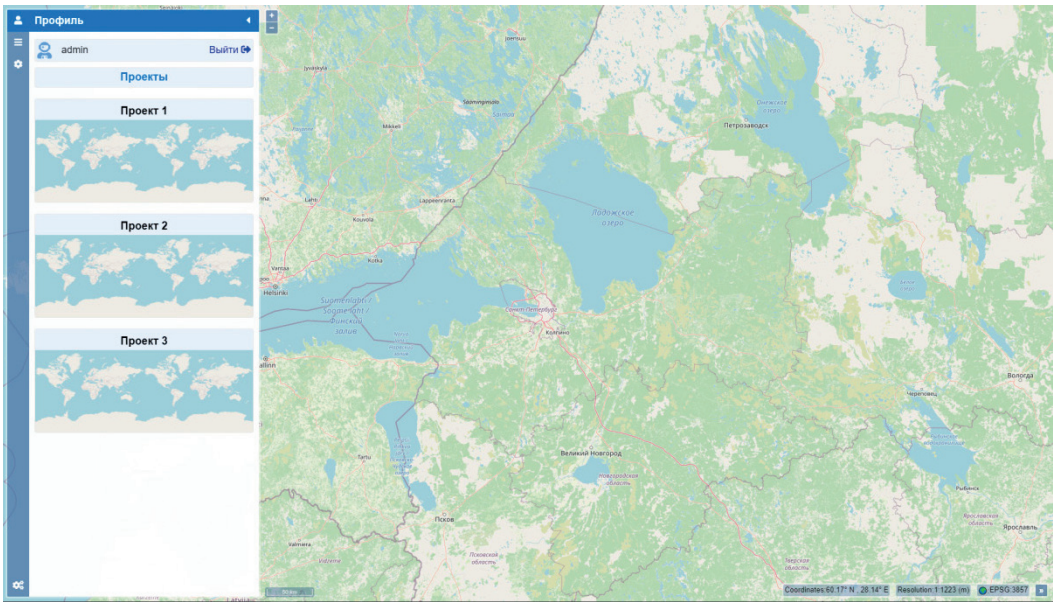


Рис. 3. Интерфейс пользователя геопортала

– поиск проекции и изменение проекции карты;

– отображение текущего масштаба и координат указателя.

Для администрирования геопортала используется стандартный административный интерфейс фреймворка Django, который позволяет управлять проектами и правами доступа для пользователей.

На рис. 4 приведен пример отображения геопортального проекта. Проект карты геопортала представляет собой древовидную структуру данных с описанием параметров геопространственных данных. Данная структура хранится в виде конфигурационного файла в формате JSON.

Актуальность использования описанного программного решения обусловлена тем что

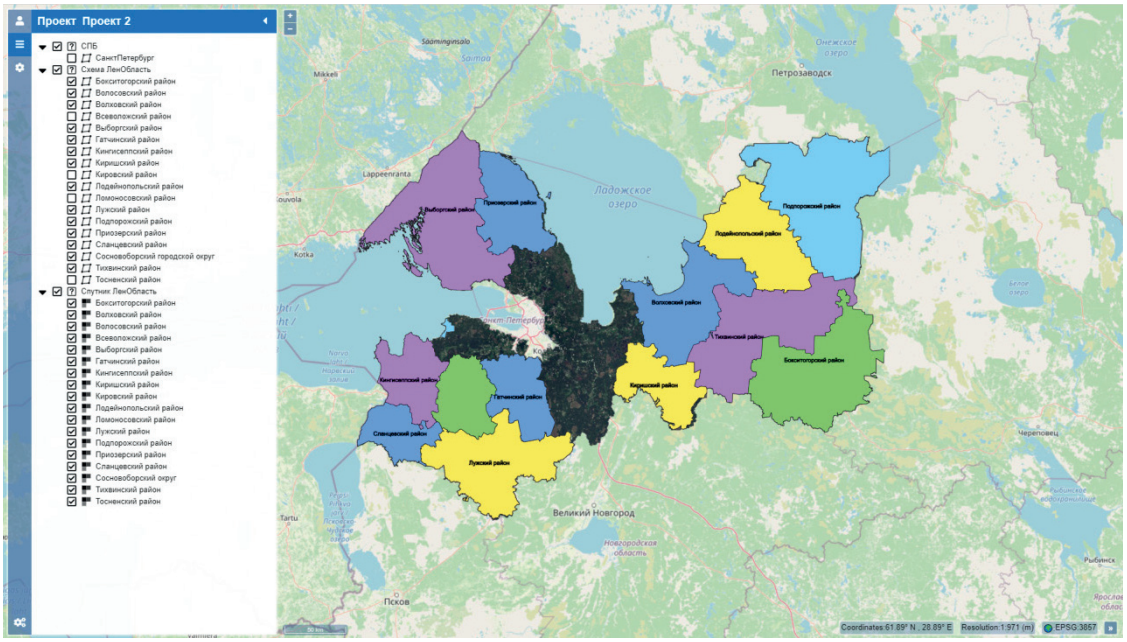


Рис. 4. Пример отображения геоинформационного проекта

пространственные данные могут храниться в распределенном виде, и для доступа к ним не требуется специализированное программное обеспечение. Данное программное решение позволит предоставить доступ пользователю к геопространственной информации с любого устройства поддерживающего работу современного веб-браузера.

Библиографический список

1. *Рогачев С. А.* Веб-картография. Представление разнородной пространственной информации // Труды СПИИРАН, 2013. № 29. С. 132–143.

2. *Ехлаков Ю. П., Жуковский О. И., Рыбалов Н. Б.* Принципы построения Web-ориентированной ГИС промышленного предприятия // Известия Томского политехнического университета. Инжиниринг георесурсов. 2006. № 309 (7). С. 146–151.

3. The Open Geospatial Consortium. URL: https://www.ogc.org/ (дата обращения: 30.11.2021).

4. *Michaelis C., Ames D.* Web Feature Service (WFS) and Web Map Service (WMS) // Encyclopedia of GIS. Springer. Boston, MA, 2008.

5. GeoServer is an open source server for sharing geospatial data. URL: http://geoserver.org/ (дата обращения: 30.11.2021).

6. *Kepka M., Jezek J.* Web client for PostGIS – the concept and implementation // Geoinformatics FCE CTU. 2013. № 11. P. 63–76.

УДК 519.6 **DOI: 10.31799/978-5-8088-1701-2-2022-2-117-119**

Ю. А. Скобцов
доктор технических наук, профессор
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ОТ ЭВОЛЮЦИОННЫХ АЛГОРИТМОВ К ВЫЧИСЛИТЕЛЬНОМУ ИНТЕЛЛЕКТУ

Представлен опыт преподавания курсов по генетическим алгоритмам, эволюционным вычислениям, вычислительному интеллекту в ряде университетов на уровне бакалавриата и магистратуры. Курсы охватывают пять основных парадигм: искусственные нейронные сети, эволюционные вычисления, нечеткие системы, роевые алгоритмы и искусственные иммунные системы. Все пять парадигм успешно применяются при решении многих научно-технических проблем по отдельности и с использованием гибридных систем.
Ключевые слова: *генетические алгоритмы, эволюционные алгоритмы, вычислительный интеллект.*

Yu. A. Skobtsov
Dr. Sc., Tech., Professor
St. Petersburg State University of Aerospace Instrumentation

FROM EVOLUTIONARY ALGORITHMS TO COMPUTATIONAL INTELLIGENCE

The experience of teaching courses on genetic algorithms, evolutionary computing, computational intelligence in a number of universities at the undergraduate and graduate levels is presented. The courses cover five main paradigms: artificial neural networks, evolutionary computing, fuzzy systems, swarm algorithms, and artificial immune systems. All five paradigms are successfully applied in solving many scientific and technical problems separately and using hybrid systems.
Keywords: *genetic algorithms, evolutionary algorithms, computational intelligence.*

В настоящее время активно развивается новое направление в теории и практике искусственного интеллекта – биоинспирированные (вдохновленные природой) алгоритмы [1]. Этот термин обычно используется для общего описания алгоритмов поиска, оптимизации или обучения, основанных на моделях различных биологических или социальных систем. Вероятно, самой известной и популярной из этих парадигм являются искусственные нейронные сети [2], которые основаны на различных моделях нервной системы. Очень популярны также эволюционные алгоритмы, где для поиска решений используется модель искусственной эволюции [3–5]. В последнее время эти парадигмы (и некоторые другие) часто объединяются на основе концепции «Вычислительный интеллект» [6]. Преподаванию подобных курсов в США и Европе отведено значительное время, что активно обсуждается в печати [6]. В университетах США все большее внимание уделяется междисциплинарному образованию и исследованиям, особенно с участием науки, техники, инженерии и математики (STEM) (science, technology, engineering and mathematics) – образования, где новое направление искусственного интеллекта входит в основные компоненты. Это объясняется тем, что, как правило, курсы по вычислительному интеллекту, в который естественно «переросли» эволюционные вычисления, носят междисциплинарный характер. К сожалению, на территории СНГ это направление (и особенно преподавание) развивается не так быстро.

Генетические алгоритмы и эволюционные вычисления

Курс «Генетические алгоритмы» (ГА) разработан автором в конце 1990-х гг. и апробирован в ряде университетов Донецка. Первоначально он включал следующие разделы: основы ГА, модификации ГА, ГА комбинаторной оптимизации, генетическое программирование, ГА машинного обучения, вероятностные ГА, программная и аппаратная реализация ГА. Кроме лекций, были поставлены лабораторные работы, которые включали поиск экстремумов функций одной и нескольких переменных и комбинаторную оптимизацию с помощью ГА. В процессе развития курс совершенствовался и

расширялся и в результате сформировался новый курс «Эволюционные вычисления», который был опубликован в качестве учебного пособия с грифом МОН Украины [3].

С 2015 г. по настоящее время автор преподавал подобные курсы в ряде университетов Санкт-Петербурга: Санкт-Петербургском политехническом университете Петра Великого, Санкт-Петербургском национальном исследовательском университете информационных технологий, механики и оптики и Санкт-Петербургском государственном университете аэрокосмического приборостроения. Содержание курсов в основном соответствует изданным учебным пособиям [3–5]. В процессе развития курс совершенствовался и в результате полученного сложилась двухуровневая схема преподавания: 1) в рамках бакалавриата на IV курсе излагаются основные разделы ЭВ и некоторые метаэвристики; 2) в магистратуре на I курсе преподаются более «продвинутые» и сложные разделы ЭВ и метаэвристики.

Вычислительный интеллект

Американское научно-техническое общество «Нейронные сети» IEEE (IEEE Neural Network Society) изменило свое название в 2004 г. на IEEE Computational Intelligence Society (общество «Вычислительный интеллект») [6]. Вычислительный интеллект (ВИ) – одно из наиболее активно развивающихся и направлений в области искусственного интеллекта и информатики, и его методы широко используются при решении многих научно-технических проблем [7, 8]. Первоначально вычислительный интеллект определяется как комбинация искусственных нейронных сетей, генетических алгоритмов и нечеткой логики. В более широком смысле он изучает адаптивные механизмы, обеспечивающие интеллектуальное поведение в сложных, неопределенных и меняющихся условиях. Такие механизмы включают те парадигмы искусственного интеллекта, которые могут учиться или адаптироваться к новым ситуациям, обобщать, открывать и ассоциировать. При этом основные парадигмы ВИ имитируют биологические системы при решении сложных задач.

В настоящее время ВИ включает 5 основных парадигм: 1) искусственные нейронные сети (ИНС), 2) эволюционные вычисления (ЭВ), 3) роевой интеллект (РИ), 4) нечеткие (фаззи) системы (ФС), 5) искусственные иммунные системы (ИИС). Все они созданы на базе моделей естественных биологических систем и процессов соответственно: 1) нервной системы, 2) эволюции, 3) поведения роя (стаи), 4) человеческих рассуждений, 5) иммунной системы млекопитающих.

Модели ИНС разработаны в виде параллельных распределенных сетевых моделей на основе моделирования процессов обучения мозга человека [2]. Разработаны десятки видов разных ИНС, среди которых наиболее популярны: многослойные сети прямого распространения, рекуррентные НС, сети с радиальными базисными функциями и вероятностные нейронные сети [2]. Как правило, ИНС состоят из входного слоя, выходного слоя и одного или нескольких скрытых слоев. Количество слоев, число нейронов в каждом слое, активационные функции нейронов и методы обучения являются определяющими при реализации ИНС для решения конкретной проблемы.

Парадигмы ЭВ созданы на основе модели естественной эволюции выживания наиболее приспособленных особей [3–5]. Здесь чаще всего используются генетические алгоритмы (ГА) и генетическое программирование (ГП). ГА представляют класс стохастических процедур поиска решений, которые используют модели законов естественной генетики (закон Дарвина, закон Менделя, концепция мутации де Вре). При этом в процессе поиска решения используется модель искусственной эволюции, где каждая особь популяции представляет потенциальное решение проблемы с помощью строки, известной как геном [3–5]. При использовании ГА для поиска решения конкретной задачи необходимо: выбрать (или разработать) представление генома – потенциального решения, определить фитнесс-функцию, позволяющую оценивать качество потенциальных решений, выбрать (или разработать) генетические операторы отбора, кроссовера (скрещивания) и мутации. Парадигма ГП имеет много общего с ГА, но отличается кодированием потенциального решения. Для ГА решение часто кодируется строкой чисел. С другой стороны, ГП использует кодирование компьютерной программы в виде древовидной структуры, которая связывает различные входы (листья дерева) через математические операторы (узлы) на выходе (корневом узле) [3, 4].

Роевые алгоритмы (РА) используют простейшие модели социального поведения стаи (роя) птиц, рыб (насекомых) и также базируются на стохастической оптимизации [9]. РА ищет (суб)оптимальное значение путем совместного использования когнитивной и социальной информации среди частиц, каждая из которых

представляет потенциальное решение проблемы. РА применяются чаще при решении задач численной оптимизации и имеют некоторые преимущества перед эволюционными алгоритмами с точки зрения более простой реализации, лучшей сходимости и меньшего количества параметров для настройки алгоритмов. С другой стороны, для комбинаторной оптимизации обычно используются муравьиные алгоритмы (МА), которые симулируют социальное поведение муравьев и основаны на моделировании явления стигметрии (локального изменения окружающей среды с помощью феромона).

Нечеткие системы (ФС) имитируют человеческие рассуждения и имеют дело с неточной и неопределенной информацией [7]. Сочетание неполной, неточной информации и неточной природы в процессе принятия решений делает нечеткую логику очень эффективной при моделировании сложной инженерии, бизнеса, финансов и систем управления, которые иначе сложно моделировать. Здесь важны выбор вида нечетких функций принадлежности, разработка базы правил, имитирующей процесс принятия решений, а также коэффициенты масштабирования, используемые на стадиях фаззификации и дефаззификации. Эти параметры и структуры в общем случае находятся на основе проб и ошибок и экспертных знаний.

Парадигмы ИИС разработаны на базе моделирования защитных механизмов биологической иммунной системы млекопитающих [10], поскольку все живые организмы обладают способностью оказывать сопротивление и развивать (частичный или полный) иммунитет к возбудителям болезней или инфекциям. Методы ИИС используют различные аспекты иммунной системы, такие как паттерн, сопоставление, извлечение признаков, обучение и память, разнообразие, распределенная обработка, самоорганизация и самозащита [11, 12]. Разработка ИИС состоит из трех основных этапов: кодирование решения, оценка взаимодействий и процедура адаптации. Различные алгоритмы ИИС предложены на основе различных моделей естественных иммунных систем, а именно метода отрицательного отбора, клонального отбора и моделей непрерывной и дискретной иммунной сети [11, 12].

Заключение

Растущий интерес к междисциплинарному образованию и исследованиям в университетах с использованием вычислительного интеллекта очевиден. ВИ – важнейшая компонента искусственного интеллекта, и его необходимо вводить в образовательные программы, по крайней мере для компьютерных специальностей [13]. Существует также интерес к широким областям кибер-инфраструктуры, кибер-систем для понимания сложных междисциплинарных систем и их развития, как видно из различных программ, принятых на федеральном уровне. Существует необходимость в обучении ВИ будущего поколения в таких «кибер»-областях, где он может стать важным инструментом для междисциплинарного образования и исследований.

Библиографический список

1. *Карпенко А. П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учеб. пособие. М.: Изд-во МГТУ им. Н. Э. Баумана, 2014. 446 с.

2. *Haykin S.* Neural Networks and Learning Machines. 3rd ed. New Jersey: Prentice Hall, 2009.

3. *Скобцов Ю. А.* Основы эволюционных вычислений. Донецк: ДонНТУ, 2008. 326 с.

4. *Скобцов Ю. А., Сперанский Д. В.* Эволюционные вычисления. М.: ИНТУИТ, 2015. 326 с.

5. *Скобцов Ю. А.* Эволюционные методы в программной инженерии: учеб. пособие. СПб.: ГУАП, 2020. 128 с.

6. *Venayagamoorthy G. K.* A Successful Interdisciplinary Course on Computational Intelligence // IEEE Computational Intellience Magazine, Institute of Electrical and Electronics Engineers (IEEE). 2009. Feb. P. 14–23.

7. *Engelbrecht A. P.* Computional intelligence: introduction. John Wiley&Sons Ltd, 2007.

8. *Скобцов Ю. А.* Междисциплинарный курс «Вычислительный интеллект» // Математические методы в технике и технологиях: сб. тр. Междунар. конф. Т. 7-1. СПб., 2020. С. 66–69.

9. *Родзин С. И., Скобцов Ю. А., Эль-Хатиб С. А.* Биоэвристики: теория, алгоритмы и приложения: монография. Чебоксары: Среда, 2019. 221 с.

10. *Dasgupta D., Luis F. N.* Immunological Computation – Theory and Applications. CRC Press, Boca Raton, FL, 2009. 298 p.

11. *Скобцов Ю. А.* Искусственные иммунные системы – основные модели // Математические методы в технологиях и технике. 2021. № 1. С. 107–110.

12. Скобцов. Ю. А. Введение в искусственные иммунные системы: учеб. пособие. СПб.: ГУАП, 2022.

13. *Skobtsov Yu.* Prospects of the Interdisciplinary Course «Computational Intelligence» in Engineering Education // Studies in Systems, Decision and Control. 2021. Vol. 342. P. 431–442.

УДК 51-76:517.9 **DOI: 10.31799/978-5-8088-1701-2-2022-2-120-126**

А. А. Щеголева*
студент
М. Д. Поляк*
старший преподаватель
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

МОДЕЛЬ «ХИЩНИК – ЖЕРТВА С ВНУТРИВИДОВОЙ КОНКУРЕНЦИЕЙ»

Описана модель «хищник – жертва с внутривидовой конкуренцией», основанная на системе уравнений Лотки – Вольтерры, с синтезом управления по методу АКАР. С помощью управления гарантируется достижение целевого значения по численности жертв. Рассмотрены основные особенности модели и приведены результаты компьютерного моделирования.
Ключевые слова: *система уравнений Лотки – Вольтерры, метод АКАР, теория управления.*

A. A. Shchegoleva*
Student
M. D. Polyak*
Senior Lecturer
**St. Petersburg State University of Aerospace Instrumentation*

PREDATOR-PREY MODEL WITH INTRASPECIFIC COMPETITION

In this article we review a predator-prey model with intraspecific competition, based on the Lotka – Volterra equations with addition of adaptive control based on the ADAR method. Adaptive control guaranties that the target value for prey is achieved by the system. Behavior of the proposed model is studied and simulation results are provided.
Keywords: *Lotka – Volterra equations, analytical design of aggregate regulators, control theory.*

Введение

В работе рассматриваются две последовательные модификации модели Лотки – Вольтерры: первая с внутривидовой конкуренцией; вторая включает внутривидовую конкуренцию и управление. Цель исследования – анализ динамики систем в зависимости от их параметров, поиск ограничений, накладываемых системами, их биологическая интерпретация и возможность их применения в экологических и биологических системах мониторинга. На основе метода аналитического конструирования агрегированных регуляторов (АКАР) выводятся закон управления для применения в системе и оценка новых ограничений. Управление позволяет использовать систему для моделирования заданных состояний и оценивать реалистичность происходящего переходного процесса.

Система Лотки – Вольтерры с внутривидовой конкуренцией

В работах [1–3] рассматривалась модель Лотки – Вольтерры, где на изменение популяции влияли рождаемость и естественная смертность. В экологии существует понятие внутривидовой конкуренции, при которой особи одного вида конкурируют друг с другом за питание из общего источника, также максимальный размер популяции может ограничиваться из-за малого ареала обитания. Например, более старые особи имеют меньше шансов выжить в борьбе за питание. Такой вид конкуренции позволяется учесть неравенство между особями, ограниченность территории проживания и делает невозможным экспоненциальный рост популяции жертв. Введем в оба уравнения модели Лотки – Вольтерры [5–6] внутривидовую конкуренцию с помощью коэффициентов γ_1 и γ_2:

$$\begin{cases} \dfrac{dx_1}{dt} = f_1 = \alpha_1 x_1 - \beta_1 x_1 x_2 - \gamma_1 x_1^2 \\ \dfrac{dx_2}{dt} = f_2 = -\alpha_2 x_2 + \beta_2 x_1 x_2 - \gamma_2 x_2^2 \end{cases}. \quad (1)$$

Система (1) описывает два вида конкуренции: межвидовую (отношения типа «хищник – жертва») и внутривидовую у обеих популяций. Внутривидовая конкуренция исключает возмож-

ность неконтролируемого экспоненциального роста жертв при отсутствии хищников (модель Мальтуса [7]). В данном виде конкуренции хищники при отсутствии жертв (единственного питания хищников в модели) вымирают, как и в системах без внутривидовой конкуренции. В отличие от систем Лотки – Вольтерры с внутривидовой конкуренций, введенной как логистическое уравнение [7], система (1) позволяет моделировать популяции без указания их максимальной емкости, что в будущем позволит осуществить алгоритм подбора коэффициентов по данным гидробиологического мониторинга. Принцип емкости экологической системы, полученный Ферхюльстом, имеет отдаленное применение в системе (1): коэффициенты γ_1 и γ_2 контролируют максимальный объем популяции жертв и хищников. Далее будет показано, как это происходит при численном решении системы (1).

Поведение системы (1) отличается от классических периодических колебаний хищника и жертвы [5–6]: на рис. 1 представлено численное решение (1), где наблюдаются затухающие колебания, приводящие обе популяции к постоянным значениям. На рис. 1, *б* показан фазовый портрет системы (1), можно определить, что фазовая траектория стремится к стационарной точке системы, что говорит об устойчивом поведении системы. Это же подтверждает рис. 1, *а*: колебания обеих популяций затухают через определенный промежуток времени, размеры популяций становятся постоянным. Для дальнейшего анализа поведения системы в зависимости от ее параметров были найдены стационарные точки.

Тривиальный случай $x_{1s} = 0, x_{2s} = 0$ не рассматривается, поскольку в биологической интерпретации он соответствует нулевым размерам обеих популяций, что не представляет интереса для исследования. Случаи равновесия, когда одна из популяций вымирает в течение определенного времени, в то время как другая становится постоянной, наиболее интересны для изучения и анализа. Как было указано ранее, система (1) позволяет исследовать такие режимы работы модели.

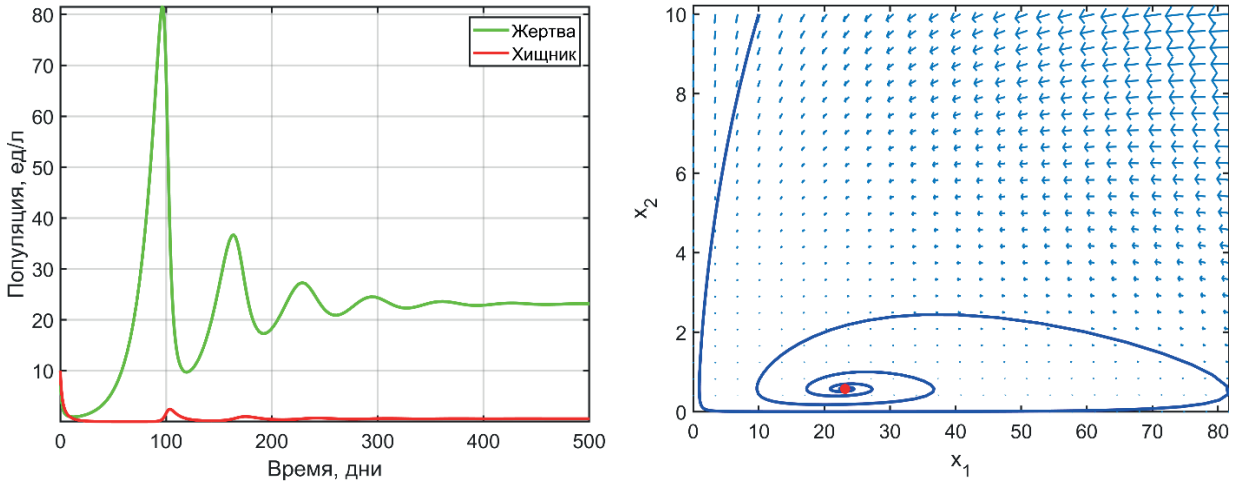
Стационарным точками системы (1) являются следующие уравнения:

$$x_{1s} = \frac{\alpha_2\beta_1 + \alpha_1\gamma_2}{\beta_1\beta_2 + \gamma_1\gamma_2}, x_{2s} = \frac{-\alpha_2\gamma_1 + \alpha_1\beta_2}{\beta_1\beta_2 + \gamma_1\gamma_2}. \quad (2)$$

Как видно из уравнения для x_{1s}, популяция жертв при условии неотрицательности всех коэффициентов системы всегда остается положительной, что является хорошим результатом. Для второго уравнения требуется вывести диапазоны значения коэффициентов, при которых хищники не будут вымирать. Предположим, что $\frac{-\alpha_2\gamma_1 + \alpha_1\beta_2}{\beta_1\beta_2 + \gamma_1\gamma_2} > 0$, тогда $-\alpha_2\gamma_1 + \alpha_1\beta_2 > 0$.

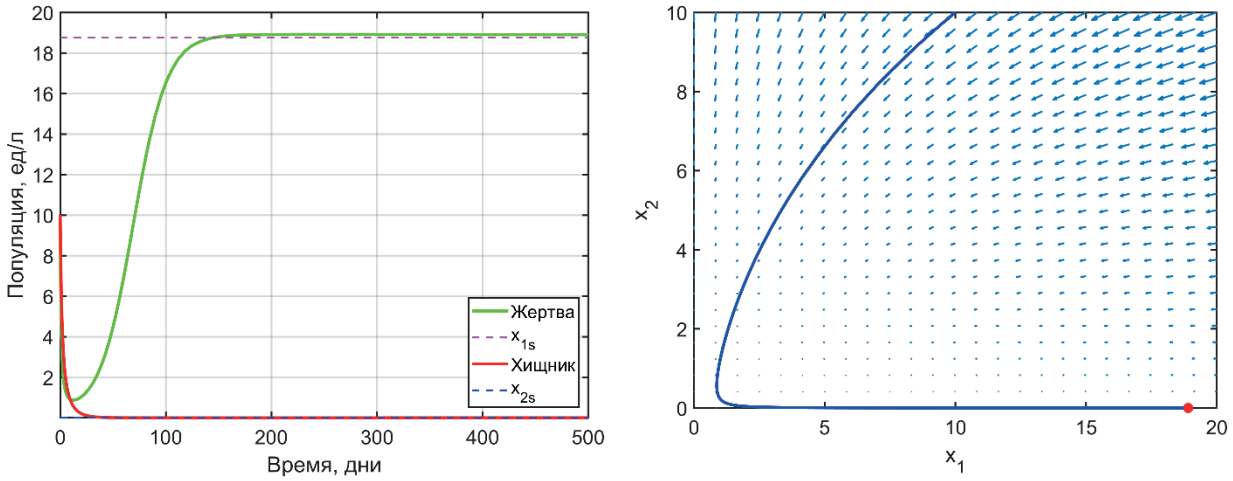
Учитывая, что коэффициенты $\alpha_1, \alpha_2, \beta_1, \beta_2$ подбираются методами оптимизации или перебора [2–4], выведем диапазон возможных значений γ_1:

$$\gamma_1 < \frac{\alpha_1\beta_2}{\alpha_2}. \quad (3)$$



а) б)

Рис. 1. Пример системы: а) хищник и жертва; б) фазовый портрет



а) б)

Рис. 2. Система при γ_1=0,0033: а) хищник и жертва; б) фазовый портрет

Можно сделать теоретическое предположение, что устоявшееся значение популяции хищников сильнее всего зависит от γ_1, остальные коэффициенты либо подбираются с учетом неотрицательности, либо являются частью суммы (как γ_2).

Найдем дополнительные стационарные точки системы (1) с учетом (3). Если не выполняется условие (3), то популяция хищников вымирает (рис. 2, *а*, 3, *а*): $x_2(t) \to 0$ при $t \to \infty$. В этом случае стационарные точки будут иметь вид:

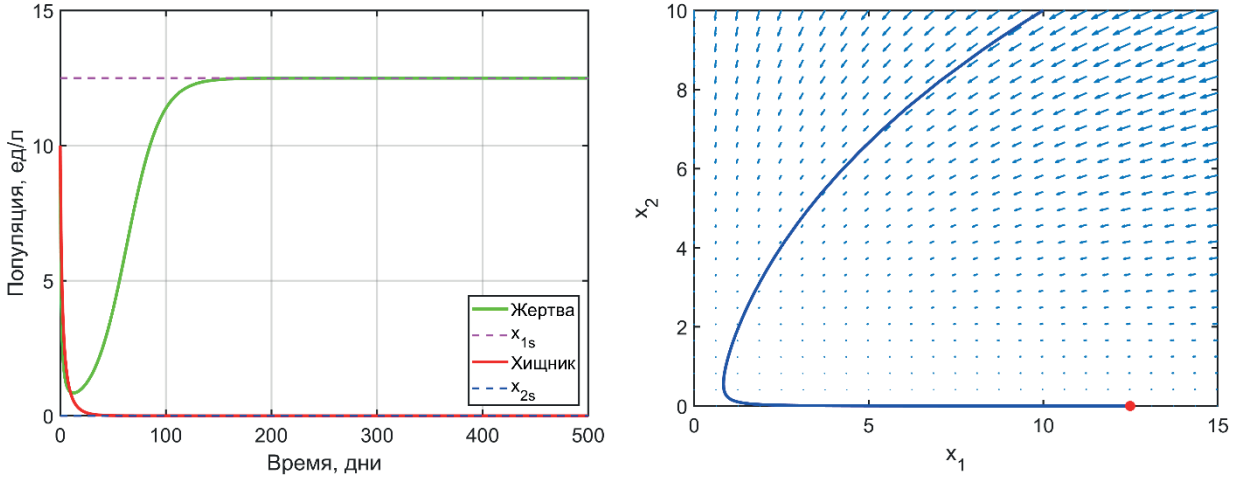
$$x_{1s} = \frac{\alpha_1}{\gamma_1}, x_{2s} = 0. \quad (4)$$

Проверим на практике выводы (3) и (4). Параметры системы:

$$\alpha_1 = 0{,}0625; \alpha_2 = 0{,}1223; \beta_2 = 0{,}0065,$$

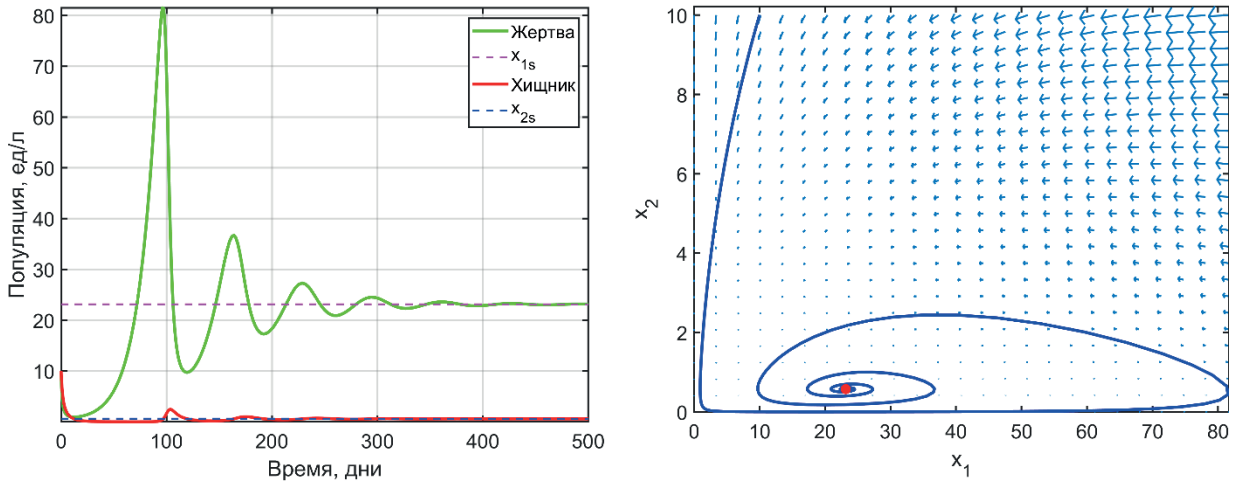
тогда $\gamma_1 < 0{,}0033$.

На рис. 2, *а* и 3, *а* не выполняется условие (3), поэтому популяция хищников становится близкой к нулю, как видно по фазовым портретам 2, *б* и 3, *б*. Жертвы остаются единственным видом, поэтому динамика их изменения отличается от исходной системы. Это выражается в отсутствии затухающих колебаний, как на рис. 4, *а*. Устоявшееся значения для жертв в таком случае соответствует стационарной точке (4). В то же



а) б)

Рис. 3. Система при γ_1=0,005: а) хищник и жертва; б) фазовый портрет



а) б)

Рис. 4. Система при γ_1=0,0001: а) хищник и жертва; б) фазовый портрет

время если γ_1 находится в диапазоне, заданном неравенством (3), как на рис. 4, *а*, то система является устойчивой по фазовому портрету (рис. 4, *б*). Можно сделать вывод, что с помощью неравенства (3) действительно можно определить диапазон возможных значений для γ_1, при которых хищники не вымирают, формулы (2) и (4) верны для расчета устоявшихся значений популяции жертв и хищников. Также это подтверждает, что система (1) остается устойчивой при разных параметрах.

Система с управлением

Для того чтобы систему (1) можно было использовать для моделирования различных биологических явлений (например, цветение сине-зеленых водорослей), необходимо добавить управление. Управление позволит контролировать переход системы из начального состояния в заданное целевое, при этом управляющее воздействие можно интерпретировать как естественные биологические процессы (миграция, приливы и отливы), явно влияющие на изменение популяции и не зависящие от размера популяций. Управление проектируется по методу аналитического конструирования агрегированных регуляторов (АКАР), описанного в работах А. А. Колесникова и С. И. Колесниковой [8–10]. В работах [1–3] показано его применение для классической модели Лотки – Вольтерры. Введем макропеременную $\psi(t) \to 0$, задающую цель управления. Целью является достижение фитопланктоном заданного значения x_1^*, т. е. $\psi(t) = x_1(t) - x_1^* \to 0$. Макропеременная должна доставлять глобальный минимум функционалу качества $\Phi = \int_0^\infty \left(\psi^2(t) + T^2\dot{\psi}^2(t)\right)dt \to \min$, для этого она должна быть решением уравнения Эйлера – Лагранжа $T\frac{d\psi(t)}{dt} + \psi(t) = 0$. Тогда система с аддитивным управлением

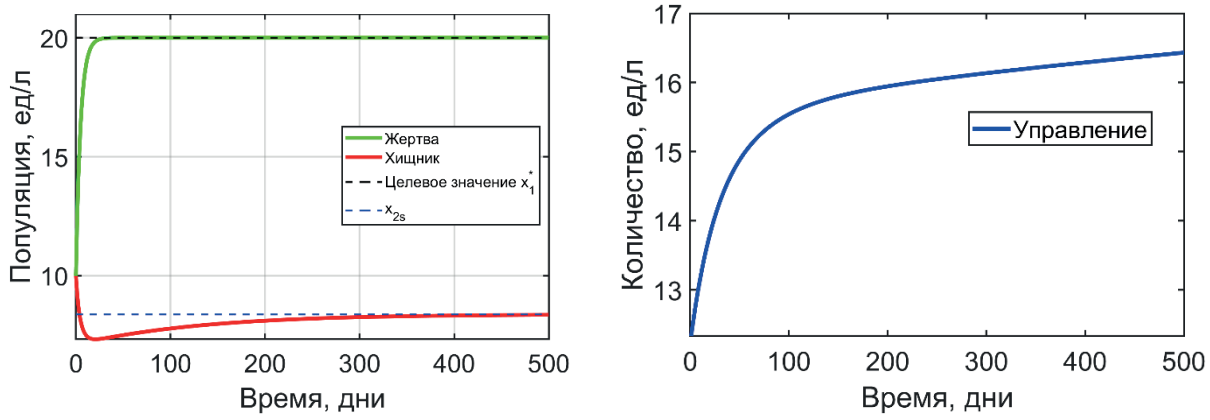
$$\begin{cases} \dfrac{dx_1}{dt} = f_1 = \alpha_1 x_1 - \beta_1 x_1 x_2 - \gamma_1 x_1^2 + u \\ \dfrac{dx_2}{dt} = f_2 = -\alpha_2 x_2 + \beta_2 x_1 x_2 - \gamma_2 x_2^2 \\ \psi = x_1 - x_1^* \\ u = -\dfrac{\psi}{T} - \alpha_1 x_1 + \beta_1 x_1 x_2 + \gamma_1 x_1^2 \end{cases}. \quad (5)$$

Устоявшееся значение для популяции хищников можно вывести при условии достижения цели $\psi = 0$, в этом случае $x_1(t) = x_1^*$ при $t \to \infty$. Достижение цели подразумевает перевод системы в конечное состояние, заданное целью управления. Выведем уравнение для расчета устоявшегося значения хищников:

$$x_{2s} = -\frac{\alpha_2 - \beta_2 x_1^*}{\gamma_2}. \quad (6)$$

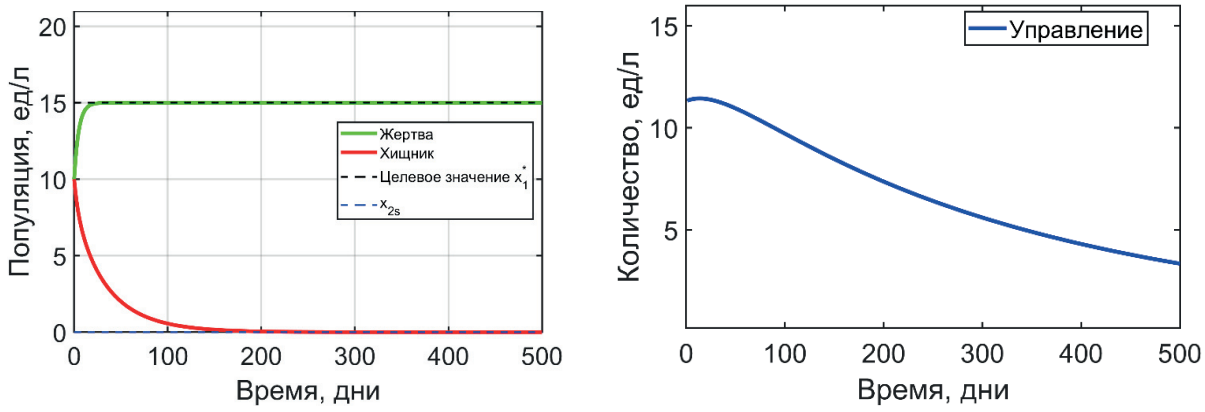
Исследуем возможные режимы сосуществования двух популяций.

При $x_1^* > \frac{\alpha_2}{\beta_2}$ обе популяции существуют, их значения положительные.



а) б)

Рис. 5. Система при $x_1^* = 20$ *: а) хищник и жертва; б) управление*



а) б)

Рис. 6. Система при $x_1^* = 15$ *: а) хищник и жертва; б) управление*

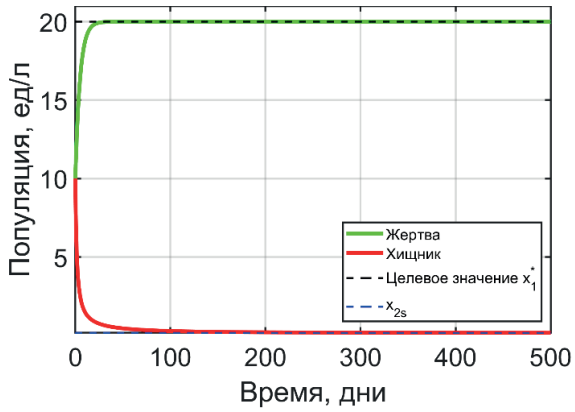
При $x_1^* \le \frac{\alpha_2}{\beta_2}$ популяция хищников вымирает, жертвы продолжают существовать.

Проверим выражение (6). Возьмем те же значения коэффициентов α_1, α_2, β_1, β_2, $T_1 = 5$, $\gamma_1 = 0{,}005$, $\gamma_{2_*} = 0{,}001$. Тогда неравенство (6) имеет вид: $x_1^* > 18{,}82$ для случая сосуществования обеих популяций. Для смены режимов будет меняться значение x_1^*.

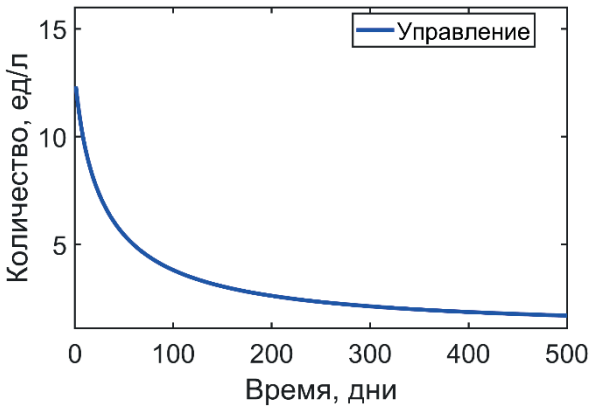
Рис. 5, *а* и 6, *а* показывают, что изменение целевого значения управления x_1^* влияет на поведение системы. Уменьшение его значения до значения ниже граничного значения, согласно неравенству, приводит к вымиранию хищников (рис. 6, *а*). Это может говорить о некоторой емкости среды, которая задается коэффициентами α_2, β_2, с их помощью задается минимальный целевой размер популяции, при котором возможно сосуществование обеих популяций. Данный факт имеет биологическую интерпретацию: в природе при уменьшении количества доступного питания снижается рождаемость среди хищников для выживания популяции. В противном случае это может привести к полному вымиранию всей популяции. Управление также отличается на рис. 5, *б* и 6, *б*: при существовании обеих популяций жертве требуется больше ресурсов для поддержания заданного значения. При отсутствии хищников управление постепенно стремится к нулю.

Исследуем влияние внутренней конкуренции γ_2 на изменение популяции хищников. В уравнении (6) явно выражена обратная пропорциональная зависимость устоявшегося значения от внутренней конкуренции. Посмотрим решения системы (5) при разных параметрах.

Из рис. 7, *а* и 8, *а* можно сделать вывод, что при уменьшении коэффициента внутривидовой конкуренции устоявшиеся значения популяции хищников больше, чем с бóльшим значе-

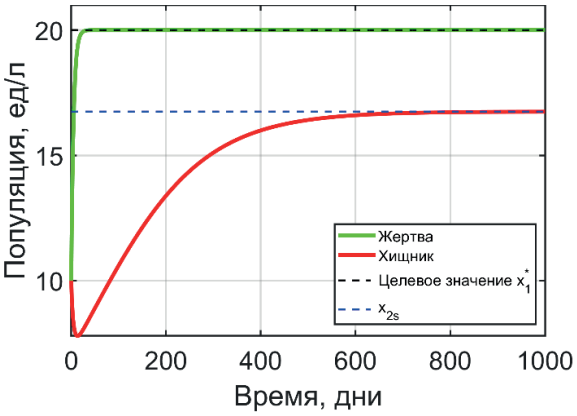


а)

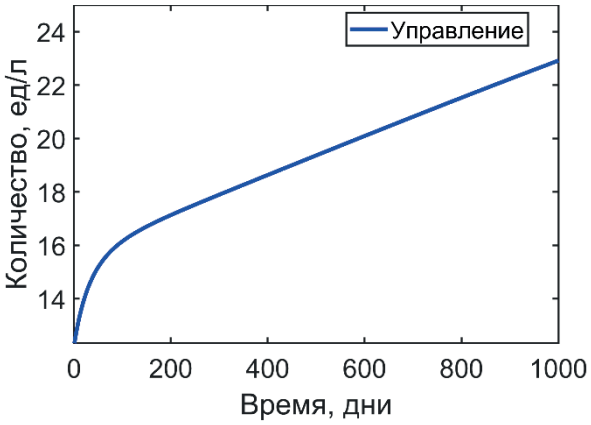


б)

Рис. 7. Система при γ_2=0,05 : а) хищник и жертва; б) управление



а)



б)

Рис. 8. Система при γ_2=0,0005 : а) хищник и жертва; б) управление

нием коэффициента внутривидовой конкуренции. Это закономерно: ввод внутривидовой конкуренции в системы подразумевал подобное поведение и свидетельствует о правильности работы модели. Также можно сделать вывод, что формула для расчета устоявшегося значения хищников верна, на рис. 5, *а* и 8, *а* размер популяции соответствует рассчитанным значениям.

Ввод управления сделал систему более гибкой. Популяция хищников зависит от собственных параметров смертности, рождаемости и внутривидовой конкуренции, а не от внутривидовой конкуренции жертв, как в системе (3). Это позволяет рассчитать минимально возможную популяцию жертв, для которой возможно сосуществование хищников. Также система является более устойчивой к изменению коэффициентов, и ее поведение более соответствует реальности.

Заключение

В работе представлена модификация классической модели «хищник – жертва» с введением внутривидовой конкуренции. Для модифицированной системы найдены стационарные точки и возможные виды поведения в зависимости от заданных параметров. Отмечены недостатки системы: сильная зависимость хищников от параметров жертв, что имеет слабое биологическое объяснение. К системе с внутривидовой конкуренцией применен метод АКАР для введения управления. Введение управления позволило вывести правило для определения минимально возможной популяции жертв, при котором возможно их сосуществование с хищниками. Полученные математические системы возможно использовать для моделирования различных экологических систем.

Библиографический список

1. *Щеголева А. А., Поляк М. Д.* Модель «хищник – жертва с питанием» // Обработка, передача и защита информации в компьютерных системах ‹21: Междунар. науч. конф.: сб. докл., СПб., 14–22 апр. 2021 г. СПб., 2021. С. 86–91.

2. *Shchegoleva A.* Investigation of phytoplankton dynamics in the Neva bay with the «predator – prey with food» model // Bulletin of the UNESCO department «Distance education in engineering» of the SUAI: Collection of the papers. St. Petersburg, 2021. P. 129–136.

3. *Щеголева А. А.* Апробация модели «хищник – жертва с питанием» на данных гидробиологического мониторинга Балтийского моря // Семьдесят четвертая Международная студенческая научная конференция ГУАП (19–23 апр. 2021 г.): сб. докл.: в 4 ч. Ч. 2: Технические науки. СПб.: ГУАП, 2021. С. 125–129.

4. *Shchegoleva A., Polyak M.* Application of ADAR method to phytoplankton population analysis // The Gulf of Finland Science Days 2021 «New start for the Gulf of Finland co-operation» Estonian Academy of Sciences, Tallinn, 29–30 Nov. 2021: abstracts. Tallin, 2021. P. 73–75.

5. *Lotka A. J.* Contribution to the Theory of Periodic Reaction // The Journal of Physical Chemistry A. 1910. № 3. P. 271–274.

6. *Volterra V.* Variations and Fluctuations of the Number of Individuals in Animal Species living together // ICES Journal of Marine Science. 1928. № 3. P. 3–51.

7. *Братусь А. С., Новожилов А. С., Платонов А. П.* Динамические системы и модели биологии. М.: ФИЗМАЛИТ, 2010. 400 с.

8. *Колесников А. А.* Синергетика и проблемы теории управления. М.: ФИЗМАТЛИТ, 2004. 504 с.

9. *Колесников А. А., Колесников Ал. А., Кузьменко А. А.* Методы АКАР и АКОР в задачах синтеза нелинейных систем управления // Мехатроника, автоматизация, управление. 2016. № 10. С. 657–669.

10. *Колесникова С. И.* Синтез управления нелинейным объектом второго порядка с неполным описанием // Автоматика и телемеханика. 2018. № 9. С. 18–30.

УДК 004.413.5 **DOI: 10.31799/978-5-8088-1701-2-2022-2-127-132**

С. В. Щёкин*
кандидат технических наук, доцент
М. В. Фаттахова*
кандидат физико-математических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОСОБЕННОСТИ ЭВОЛЮЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ РАБОТЫ С ТРЕХМЕРНОЙ ГРАФИКОЙ НА ОСНОВЕ ОТКРЫТЫХ ИСХОДНЫХ ТЕКСТОВ

Рассматриваются особенности развития программного обеспечения с открытыми исходными текстами для работы с трехмерной графикой на основе ретроспективы ряда версий исходных текстов за длительный промежуток времени.
Ключевые слова: *программное обеспечение; трехмерная графика, открытые исходные тексты; средства разработки; жизненный цикл.*

S. V. Schyokin*
PhD, Tech., Associate Professor
M. V. Fattahova*
PhD, Phys.-Math., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

FEATURES OF THE EVOLUTION OF SOFTWARE FOR WORKING WITH THREE-DIMENSIONAL GRAPHICS BASED ON OPEN SOURCE

The article discusses the features of the development of open source software for working with three-dimensional graphics based on a retrospective of a number of versions over a long period of time.
Keywords: *software; three-dimensional graphics; open source; software development tools; life cycle.*

В последнее время достаточно актуальной стала разработка программных систем с использованием открытого программного обеспечения. Доступность исходных текстов и объем существующих наработок в этой области позволяют серьезно сократить время и затраты на разработку новых систем. При этом очень важно сделать грамотный выбор того, что планируется взять за основу или использовать в качестве инструментов разработки [1]. В дальнейшем во многом от этого будут зависеть сложность и возможность длительной поддержки разрабатываемых программных решений.

Проанализируем, каким образом развивались свободно распространяемые средства трехмерной графики на примере двух графических библиотек и одного трехмерного редактора. В табл. 1 и 2 приведены объемы исходных текстов библиотеки Mesa 3D за период с 1996 по 2019 г. Мы опускаем период с 1999 по 2014 г., рассмотренный ранее [2].

Библиотека Mesa 3D – вариант реализации стандарта Open GL с открытыми исходными текстами. Основная часть кода представлена языками C и C++. Значительная часть кода во всех версиях, начиная с 1998 г. является аппаратно-зависимой и написана на ассемблере. Интересными моментами на начальном этапе развития библиотеки были фрагменты кода на языках Objective C и Fortran. Использование языка Fortran было связано с возможностью подключения библиотеки к программам, написанным на этом языке. Достаточно быстро эта возможность исчезла. Аналогичная ситуация в дальнейшем отмечалась и с языком Objective C.

Язык GLSL позволяет в программы, работающие с вызовами Open GL, включать фрагменты кода на языке программирования шейдеров, предназначенные для компиляции, загрузки и выполнения непосредственно на ядрах видеокарты. Включение кода на языке GLSL выглядит вполне логичным. Язык Python в современных версиях используется для конфигурирова-

Таблица 1

Объемы кода Mesa 3D (C/ C++, Ассемблер, GLSL)

Версия	Год выхода	Всего строк	C	C++	H, hpp	Ассемблер	GLSL
mesa-2.1	1996	144436	121979	623	16527	0	0
mesa-2.2	1997	178118	146503	8624	17097	0	0
mesa-2.6	1998	170529	134193	7221	20859	1748	0
mesa-11.0.0	2015	1842435	1011609	216400	468899	68103	0
mesa-11.2.0	2016	1931998	1066757	225849	491886	68304	0
mesa-11.2.2	2016	1932243	1066963	225874	491900	68304	0
mesa-12.0.0	2016	2099626	1149357	259432	531582	68296	642
mesa-13.0.0	2016	2297046	1212197	261237	661850	68607	667
mesa-12.0.6	2017	2137440	1149720	259743	568592	68296	642
mesa-13.0.6	2017	2301161	1214249	262745	662370	68607	667
mesa-17.0.0	2017	2355335	1248360	267727	677563	68607	667
mesa-17.0.3	2017	2353820	1246662	267790	677597	68607	667
mesa-17.3.0	2017	2902089	1509946	297353	924528	72332	667
mesa-17.3.9	2018	2901762	1509241	297485	924671	72332	667
mesa-18.0.0	2018	3000734	1592770	298322	937910	72390	667
mesa-18.3.0	2018	3148198	1620889	336991	1013062	72765	667
mesa-18.3.6	2019	3149837	1621708	337518	1013333	72765	667
mesa-19.0.0	2019	3176814	1647973	338440	1008804	73483	2453

Таблица 2

Объемы кода Mesa 3D (Fortran, Objective C, Python)

Версия	Год выхода	Всего строк	Fortran	Objective C	Python
mesa-2.1	1996	144436	75	1840	0
mesa-2.2	1997	178118	75	1840	0
mesa-2.6	1998	170529	0	1840	0
mesa-11.0.0	2015	1842435	0	0	23077
mesa-11.2.0	2016	1931998	0	0	23603
mesa-11.2.2	2016	1932243	0	0	23603
mesa-12.0.0	2016	2099626	0	0	33996
mesa-13.0.0	2016	2297046	0	0	35839
mesa-12.0.6	2017	2137440	0	0	33978
mesa-13.0.6	2017	2301161	0	0	35839
mesa-17.0.0	2017	2355335	0	0	35909
mesa-17.0.3	2017	2353820	0	0	35922
mesa-17.3.0	2017	2902089	0	0	33821
mesa-17.3.9	2018	2901762	0	0	33840
mesa-18.0.0	2018	3000734	0	0	34421
mesa-18.3.0	2018	3148198	0	0	37186
mesa-18.3.6	2019	3149837	0	0	37207
mesa-19.0.0	2019	3176814	0	0	38819

ния библиотеки в процессе сборки и компиляции из исходных текстов.

Библиотека Open Scene Graph является надстройкой над Open GL (Mesa 3D) и позволяет описывать выводимые графические объекты в виде графа сцены. Объемы исходных текстов на использованных при ее создании языках программирования в период с 2007 по 2019 г. приведены в табл. 3, 4. Основная часть кода (см. табл. 2) написана на языке C++, значительно меньшее количество – на языке C. Кроме того, представлены языки Objective C, Objective C++ и Java.

Фрагменты Objective C связаны с возможностью использования Open Scene Graph в опера-

Таблица 3

Объемы кода Open Scene Graph (C, C++)

Версия	Год выхода	Всего строк	C	C++	h, hpp
OpenSceneGraph-2.0.0	2007	370249	2814	316281	39880
OpenSceneGraph-2.2.0	2007	388333	2814	332724	40768
OpenSceneGraph-2.4.0	2008	428595	2814	366734	44640
OpenSceneGraph-2.6.0	2008	449371	2814	386284	45212
OpenSceneGraph-2.8.1	2009	504821	2814	435143	50362
OpenSceneGraph-3.0.0	2011	527882	10753	438064	56048
OpenSceneGraph-3.2.0	2013	557267	11628	461713	58278
OpenSceneGraph-3.4.0	2015	610230	28201	488616	63794
OpenSceneGraph-3.4.1	2017	611300	28201	489516	63771
OpenSceneGraph-3.6.0	2018	609216	28231	488988	63650
OpenSceneGraph-3.6.1	2018	609172	28231	488950	63652
OpenSceneGraph-3.6.3	2018	606082	25418	488880	63451
OpenSceneGraph-3.6.4	2019	604324	25418	487227	63297

Таблица 4

Объемы кода Open Scene Graph (Objective C, Objective C++, Java)

Версия	Год выхода	Всего строк	Objective C	Objective C++	Java, JS
OpenSceneGraph-2.0.0	2007	370249	438	1395	135
OpenSceneGraph-2.2.0	2007	388333	438	1395	135
OpenSceneGraph-2.4.0	2008	428595	438	1421	102
OpenSceneGraph-2.6.0	2008	449371	438	1421	102
OpenSceneGraph-2.8.1	2009	504821	438	1421	102
OpenSceneGraph-3.0.0	2011	527882	455	5899	2150
OpenSceneGraph-3.2.0	2013	557267	455	7699	2150
OpenSceneGraph-3.4.0	2015	610230	423	8107	2150
OpenSceneGraph-3.4.1	2017	611300	423	8119	2150
OpenSceneGraph-3.6.0	2018	609216	423	8228	2150
OpenSceneGraph-3.6.1	2018	609172	423	8228	2150
OpenSceneGraph-3.6.3	2018	606082	423	8228	2150
OpenSceneGraph-3.6.4	2019	604324	423	8228	2150

ционной системе MacOSX Объемы исходных текстов постепенно увеличиваются от версии к версии, в библиотеке появляются новые возможности и новые конфигурационные файлы для интеграции с современными интегрированными средами разработки и различными платформами. В версиях, начиная с OpenSceneGraph-3.0.0, поддерживается возможность создания приложений с использованием Open Scene Graph для операционной системы Android.

За последние 12 лет радикальных изменений в составе исходных текстов не происходило, что позволяет с некоторой долей уверенности полагать, что и в ближайшей перспективе развитие библиотеки будет происходить аналогичным образом.

В качестве последнего примера мы выбрали трехмерный графический редактор Blender 3D. Попытаемся проследить за изменениями его исходных текстов на протяжении 18-летнего периода, с 2003 г. по настоящее время. Он интересен как составом, так и назначением языков программирования, которые были использованы в процессе разработки. В отличие от рассмотренных ранее систем, скриптовый язык Python здесь используется не исключительно как средство конфигурирования и управления процессом сборки из исходных текстов, а в качестве одного из основных языков программирования редактора практически наравне с языком C++. Несколько большее количество кода традиционно написано на языке C (табл. 5).

Таблица 5

Объемы кода Blender 3D (C, C++, Java, Python)

Версия	Год выхода	Всего строк	C	C++	h, hpp	Java, JS	Python
Blender-2.26	2003	508147	229487	123563	95277	26	17144
Blender-2.40	2005	864552	411581	193794	149874	26	70132
Blender-2.44	2007	1329374	773794	198074	207836	26	101548
Blender-2.47	2008	1712534	998098	231012	255994	26	123137
Blender-2.48	2008	2078705	1223627	233364	282731	26	127485
Blender-2.48a	2008	2082544	1224871	233432	282732	26	130011
Blender-2.49	2009	2267131	1290455	254191	301075	26	189558
Blender-2.49a	2009	2268641	1291220	253901	301051	26	190660
Blender-2.49b	2009	2266223	1293407	254275	301165	26	191787
Blender-2.50a1	2010	1354339	740619	220772	248255	172	119180
Blender-2.60	2011	1634879	828851	305374	279960	172	198534
Blender-2.60a	2011	1634875	828823	305375	279960	172	198557
Blender-2.61	2011	1913704	884883	402454	348252	172	253404
Blender-2.62	2012	1999695	899589	422337	372594	11719	268259
blender-2.78	2016	2759974	1185840	557568	642723	11719	316672
blender-2.79	2017	2947032	1202723	565637	655186	11719	464576
blender-2.80	2019	3208653	1371877	503485	675491	0	581967
blender-2.81	2019	3463392	1427673	539441	802414	0	612259
blender-2.82	2020	3568209	1447444	570282	825107	0	641030
blender-2.83.0	2020	3635979	1482911	580734	834420	0	651852
blender-2.90.0	2020	3675412	1516000	601066	819047	0	653440
blender-2.91.0	2020	3741804	1515060	645953	832585	0	659058
blender-2.92.0	2021	3791174	1532171	658990	837521	0	669618
blender-2.93.0	2021	3859303	1550113	692375	846873	0	673435

На протяжении жизненного цикла видны периоды значительных изменений в составе языков программирования и объемах исходных текстов. Это может свидетельствовать о серьезных доработках и чистках исходных текстов редактора. В версии Blender-2.50a1 в 2010 г. объемы кода на языке C сократились почти в два раза, к прежнему уровню они вернулись только к 2019 г. вместе с полным исчезновением языка Java, на котором очень незначительное количество кода фрагментарно встречалось ранее. На языке Python также можно заметить резкие колебания объемов исходных текстов.

Еще более любопытны появление, резкий рост и полное исчезновение аппаратно зависимых участков кода на языке ассемблера, которые происходили в период с 2007 по 2010 г. В 2009 г на языке ассемблера было написано уже не менее 100 тыс. строк кода.

Параллельно с языком ассемблера в исходных текстах стали появляться и другие аппаратно-зависимые элементы на языках программирования, позволяющих запускать фрагменты программ на ядрах современных видеокарт. Так постепенно в исходных текстах трехмерного редактора появились язык программирования шейдеров GLSL, языки для запуска фрагментов приложений на ядрах видеокарт Open CL и Cuda C (табл. 6).

Появление аппаратно зависимых фрагментов, очевидно, было обусловлено попыткой ускорения сложных вычислений в процессе рендеринга реалистичных изображений для анимации. Среда Blender 3D поддерживает достаточно затратные с точки зрения вычислительных ресурсов методы и алгоритмы реалистичной компьютерной графики. С распространением языков программирования высокого уровня, позволяющих переносить часть вычислений непосредственно на свободные ядра современных видеокарт, использование большой по объему аппаратно зависимой ассемблерной части ис-

Таблица 6

Объемы кода Blender 3D (Ассемблер, GLSL, Open CL, Cuda C)

Версия	Год выхода	Всего строк	Ассемблер	Open CL, GLSL, Cuda C
blender-2.26	2003	508147	0	0
blender-2.40	2005	864552	0	0
blender-2.44	2007	1329374	2031	0
blender-2.47	2008	1712534	51079	0
blender-2.48	2008	2078705	88253	1567
blender-2.48a	2008	2082544	88253	1567
blender-2.49	2009	2267131	107309	1567
blender-2.49a	2009	2268641	107309	1567
blender-2.49b	2009	2266223	100748	1567
blender-2.50a1	2010	1354339	0	1607
blender-2.60	2011	1634879	0	1816
blender-2.60a	2011	1634875	0	1816
blender-2.61	2011	1913704	0	2303
blender-2.62	2012	1999695	0	2326
blender-2.78	2016	2759974	0	7450
blender-2.79	2017	2947032	0	7930
blender-2.80	2019	3208653	0	26683
blender-2.81	2019	3463392	0	29336
blender-2.82	2020	3568209	0	31305
blender-2.83.0	2020	3635979	0	31572
blender-2.90.0	2020	3675412	0	31632
blender-2.91.0	2020	3741804	0	32502
blender-2.92.0	2021	3791174	0	32402
blender-2.93.0	2021	3859303	0	34508

Таблица 7

Объемы кода Blender 3D (Objective C, Objective C++, Fortran)

Версия	Год выхода	Всего строк	Fortran	Objective C	Objective C++
blender-2.26	2003	508147	0	26	0
blender-2.40	2005	864552	0	26	0
blender-2.44	2007	1329374	0	26	0
blender-2.47	2008	1712534	0	26	0
blender-2.48	2008	2078705	149	26	0
blender-2.48a	2008	2082544	149	26	0
blender-2.49	2009	2267131	149	26	0
blender-2.49a	2009	2268641	149	26	0
blender-2.49b	2009	2266223	149	26	0
blender-2.50a1	2010	1354339	0	1244	3193
blender-2.60	2011	1634879	0	1746	3670
blender-2.60a	2011	1634875	0	1746	3670
blender-2.61	2011	1913704	0	1738	3788
blender-2.62	2012	1999695	0	1755	3804
blender-2.78	2016	2759974	0	1284	4149
blender-2.79	2017	2947032	0	1284	4176
blender-2.80	2019	3208653	0	0	4262
blender-2.81	2019	3463392	0	0	4362
blender-2.82	2020	3568209	0	0	4372
blender-2.83.0	2020	3635979	0	0	4503
blender-2.90.0	2020	3675412	0	0	4499
blender-2.91.0	2020	3741804	0	0	4503
blender-2.92.0	2021	3791174	0	0	4576
blender-2.93.0	2021	3859303	0	0	4527

ходного кода стало терять смысл, тат как ее преимущества в смысле возможностей по ускорению вычислений становились уже не такими очевидными. Исключение ассемблерной части в пользу развития ее альтернатив выглядело вполне разумным шагом.

Появление и исчезновение фрагментов кода на языке Fortran связаны с заимствованием набора программ для реализации быстрого преобразования Фурье и чисткой кода (табл. 7).

Компоненты на языках Objective C и Objective C++ в исходных текстах связаны с интеграцией редактора со средами выполнения для платформы Apple. Очевидно, что ее поддержка в ближайшем будущем будет продолжена.

Приведенные примеры показывают, каким образом на основе анализа исходного кода ряда версий за длительный период можно проследить особенности эволюции и оценить перспективы развития программного обеспечения с открытыми исходными текстами для работы с трехмерной графикой.

Библиографический список

1. *Щекин С. В.* Особенности жизненного цикла средств разработки пользовательских интерфейсов с открытыми исходными текстами // Обработка, передача и защита информации в компьютерных системах: первая Всерос. науч. конф., СПб, 2020. С. 113–120.

2. *Щекин С. В.* Особенности жизненного цикла программных библиотек с открытыми исходными текстами // Научная сессия ГУАП: сб. докл.: в 3 ч. Ч. 2. СПб, 2016. С. 322–326.

УДК 004.4 **DOI: 10.31799/978-5-8088-1701-2-2022-2-133-137**

А. Е. Юрченко*
студент
С. А. Рогачев*
старший преподаватель
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАЗРАБОТКА ПРОГРАММНОГО МОДУЛЯ МОНИТОРИНГА ТЕМПЕРАТУРЫ ВОДНОЙ ПОВЕРХНОСТИ ПО ДАННЫМ ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ ЗЕМЛИ

Приведена методика расчета температуры земной поверхности с использованием данных космической съемки аппаратом Landsat-8. Представлен прототип программного модуля, осуществляющего в автоматизированном режиме расчет температуры поверхности Земли и визуализацию полученных результатов.
Ключевые слова: *температура земной поверхности, мониторинг температуры, дистанционные методы.*

A. E. Yurchenko*
Student
S. A. Rogachev*
Senior Lecturer
**St. Petersburg State University of Aerospace Instrumentation*

DEVELOPMENT OF SOFTWARE FOR WATER SURFACE TEMPERATURE MONITORING BASED ON EARTH REMOTE SENSING DATA

The paper presents a methodology for calculating the temperature of the earth's surface using data from space imagery from Landsat-8 satellites. A prototype of a software module in an automated mode for calculating the surface temperature and visualizing the ground of the results obtained is presented.
Keywords: *earth surface temperature, temperature monitoring, remote sensing methods.*

В наше время на территории России существует 11 атомных электростанций (АЭС). В данной работе проводится исследование на примере Ленинградской АЭС, которая находится в городе Сосновый бор (Ленинградская область). Для охлаждения тепломеханического оборудования АЭС берет воду из Копорской губы Финского залива Балтийского моря. Из залива вода напрямую направляется в конденсаторы турбин для охлаждения пара. На следующем этапе эта вода возвращается обратно в водоем. Температура воды, сбрасываемая в водоем, становится значительно выше, что влияет на микроклимат территории и приводит к повышению температуры воды.

Основная цель данной работы – отслеживание температуры водной поверхности в области Ленинградской АЭС и выявление среднего повышения температуры воды в окрестностях после сбрасывания ее в водоем.

Задачи, выполняемые в данной работе:

1) получение и обработка космических снимков с аппарата Landsat-8;

2) разработка программного модуля, который позволит проводить мониторинг температуры водной поверхности в заданной области;

3) разработка графического интерфейса для удобной визуализации результата проделанной работы.

В данной работе используются снимки с космического аппарата Landsat-8, с помощью которых производятся расчеты для определения температуры в конкретной области поверхности Земли [1].

Landsat – проект, который обеспечивает получение спутниковых снимков поверхности Земли. Данные, полученные при помощи космического аппарата Landsat-8, можно использовать для решения большого количества различных задач. Снимок с космического аппарата Landsat-8 представляет собой набор из спектральных каналов, которые определяются интервалом электромагнитного спектра, в котором сенсор аппарата Landsat-8 ведет съемку. В подобном снимке будет представлено 11 спек-

тральных каналов, каждый из который несет в себе определенные данные (табл. 1) [1].

Таблица 1

Характеристика каналов

Спектральный канал	Длины волн (мкм)
Канал 1 – аэрозоли и береговые линии	0,433–0,453
Канал 2 – синий	0,450–0,515
Канал 3 – зеленый	0,525–0,600
Канал 4 – красный	0,630–0,680
Канал 5 – ближний инфракрасный	0,845–0,885
Канал 6 – ближний инфракрасный	1,560–1,660
Канал 7 – ближний инфракрасный	2,100–2,300
Канал 8 – панхроматический	0,500–0,680
Канал 9 – перистые облака	1,360–1,390
Канал 10 – дальний инфракрасный	10,300–11,300
Канал 11 – дальний инфракрасный	11,500–12,500

Общая схема и основные этапы расчета температуры поверхности Земли по данным космического снимка Landsat-8 представлены на рис. 1.

Для реализации расчетов в соответствии со схемой рис. 1 будет использован дальний инфракрасный канал (канал 10), а также каналы красного и ближнего инфракрасного электромагнитного спектра [2].

Переход к спектральной яркости верхних слоев атмосферы

Используя мультипликативный и аддитивный коэффициенты (которые хранятся и поставляются вместе с полученными космическими снимками), значения пикселей, содержащиеся в дальнем инфракрасном канале, можно преобразовать в спектральное излучение по формуле [3]

$$L_\lambda = ML * Q_{cal} + AL,$$

где L_λ – спектральная яркость верхних слоев атмосферы, ML – мультипликативный коэффициент из метаданных (скачиваются с вместе со слоями) типа RADIANCE_MULT_BAND_x, где x – номер канала), Q_{cal} – квантованные и откалиброванные значения пикселя канала, AL – аддитивный коэффициент из метаданных типа RADIANCE_ADD_BAND_x, где x – номер канала (в данном случае используется десятый канал).

Переход к температурной яркости верхней границы атмосферы

Данные спектральной яркости (полученные на предыдущем этапе) могут быть преобразованы в температурную яркость верхней границы атмосферы с использованием значений тепловой постоянной в файле метаданных [3]:

$$\mathrm{BT} = \frac{K_2}{\ln\left(\frac{K_1}{L_\lambda} + 1\right)} - 273{,}15,$$

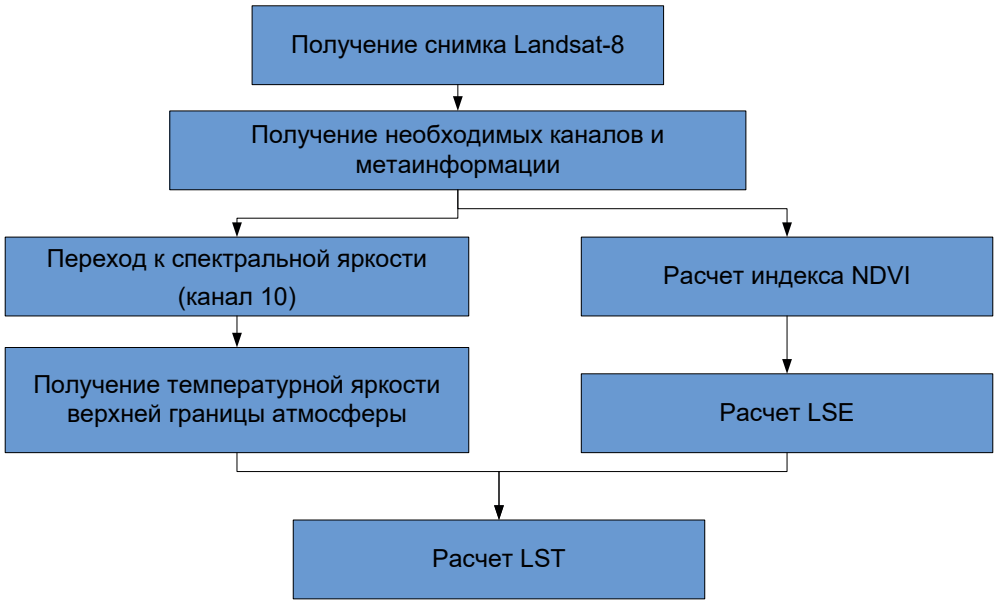


Рис. 1. Основные этапы расчета температуры поверхности Земли по данным Landsat-8

где BT – температура в верхней части атмосферы (градусы Цельсия), L_λ – рассчитан на предыдущем шаге, а K_1 и K_2 – тепловые константы, значения которых хранятся в метаданных космического снимка. Так как рассчитанные значения температуры измеряются в кельвинах, для перехода к градусам Цельсия из формулы вычтено значение абсолютного нуля.

Нормализованный разностный вегетационный индекс (NDVI)

NDVI – простой количественный показатель качества и количества растительности на заданной области. Он показывает, как растения отражают и поглощают световые волны разной длины. Рассчитывается с использованием ближнего инфракрасного и красного каналов по формуле [2]

$$NDVI = \frac{NIR - RED}{NIR + RED},$$

где $NDVI$ – значение нормализованного разностного вегетационного индекса, NIR – значения пикселей в ближнем инфракрасном диапазоне (пятый канал), RED – значения пикселей в видимом красном диапазоне электромагнитного спектра (четвертый канал).

Коэффициент отражательной способности земной поверхности

Под коэффициентом отражательной способности поверхности Земли (land surface emissivity, *LSE*) понимается средняя излучательная способность поверхности Земли, которая в данном случае рассчитывается по значениям данных *NDVI* с использованием формулы [2]

$$LSE = 0{,}04 \cdot \left(\frac{NDVI - NDVI_{\min}}{NDVI_{\max} + NDVI_{\min}} \right)^2 + 0{,}986,$$

где LSE – коэффициент отражательной способности земной поверхности, $NDVI$ – значения нормализованного разностного вегетационного индекса, $NDVI_{\min}$ – минимальное рассчитанное значение $NDVI$, $NDVI_{\max}$ – максимальное рассчитанное значение $NDVI$.

Температура поверхности земли

Температура поверхности земли (land surface temperature, *LST*) – это радиационная температура, которая рассчитывается с использованием температуры верхней части атмосферы, длины волны излучения и коэффициента отражательной способности земной поверхности [3]:

$$LST = \mathrm{BT} + W \cdot \left(\frac{\mathrm{BT}}{14380} \right) \cdot \ln(LSE),$$

где LST – температура поверхности земли, BT – температура верхней границы атмосферы, W – дина волны излучения регистрируемого на сенсоре (в данном случае десятого канала снимка Landsat-8), LSE – коэффициент отражательной способности земной поверхности.

Для разработки модуля мониторинга температуры водной поверхности был выбран язык программирования Python. Выбор языка обусловлен кроссплатформенностью его интерпретатора, что позволяет запустить программный модуль на персональном компьютере под управлением различных операционных систем.

Поставляемые данные с комического аппарата Landsat-8 хранятся в GeoTIFF, который представляет собой классический формат хранения TIFF, но при этом имеет атрибутивные поля, содержащие географическую привязку изображения. Для работы с геопривязанными растровыми данными использовался модуль GDAL (Geospatial Data Abstraction Library), который является кросплатформенным бесплатным программным обеспечением с открытым исходным кодом [4]. В данный модуль входят инструменты, которые позволяют получать информацию, объединять, перепроецировать растровые данные и т. д. Также модуль имеет инструменты для работы с векторными пространственными данными.

Для отображения результата используется библиотека Matplotlib. Данный кроссплатформенный модуль распространяется на основе BSD-подобной лицензии и предназначен для визуализации дву- и трехмерной графики. Модуль позволяет выводить графики и диаграммы разных видов, указывать оси координат, решетку, добавлять надписи и пояснения, использовать логарифмическую шкалу или полярные координаты, добавлять цветовые палитры для значений данных и т. д. [5].

На рис. 2 представлен интерфейс разработанного прототипа программного модуля для мониторинга температуры земной поверхности. Для начала работы программного модуля необходмо указать путь к директории, в которой хранятся отдельные файлы спектральных каналов космического снимка. Также указывается путь для сохранения результата, который будет представлять собой файл формата TIF

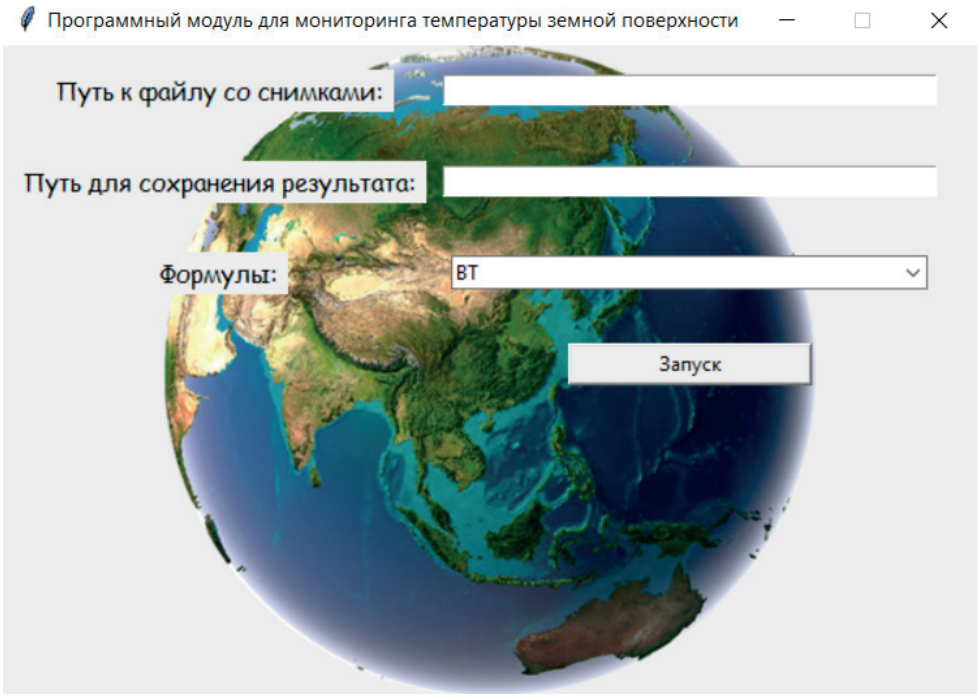


Рис. 2. Интерфейс прототипа программного модуля для мониторинга температуры земной поверхности

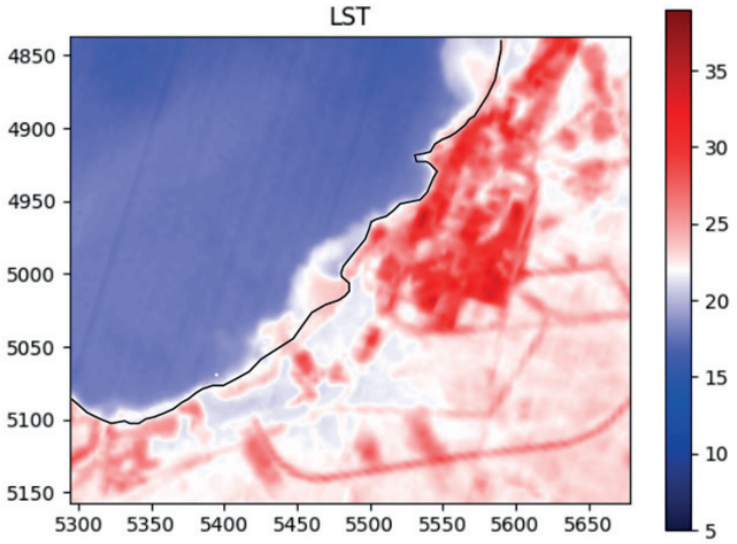


Рис. 3. Результат визуализации рассчитанной температуры поверхности

с результатом работы программного модуля. При использовании программного модуля имеется возможность выбора формул (описаны ранее) для поэтапных расчетов. При необходимости имеется возможность запустить программный модуль для последовательного расчета всех необходимых значений. После ввода всех необходимых данных при нажатии на кнопку «Запуск» начнется работа программного модуля по расчету необходимых показателей. При этом все операции происходят в автоматическом режиме, программный модуль самостоятельно осуществляет выбор каналов, а также получает нужную для расчетов метаинформацию. После окончания работы все рассчитанные показатели сохранятся в указанную дирек-

торию, а на экране отобразится окно, визуализирующее рассчитанную температуру поверхности с использованием легенды.

На рис. 3 представлен фрагмент результата расчетов температуры поверхности. Черной линией обозначены границы береговой линии Копорской губы. На данном этапе по осям изображения обозначены координаты пикселя полученного изображения, в дальнейшем имеет смысл осуществить перерасчет координат пикселей в географические координаты. Значения температурной легенды измеряются в градусах Цельсия.

На основе проделанной работы можно сделать вывод, что температура воды в районе сброса с Ленинградской АЭС выше всей водной поверхности (на период июня 2020 г.), значит, вода из охлаждающего контура может влиять на биосистему как Копорской губы, так и всего Финского залива.

В качестве дальнейших этапов работы имеет смысл верифицировать полученные результаты и рассчитать зависимость температуры поверхности воды от температуры воды, измеренной на гидропостах. Относительно разработки программного модуля стоит автоматизировать процесс получения исходных данных космического снимка, а также реализовать возможность обработки серий снимков, для осуществления мониторинга изменения температуры во времени.

Библиографический список

1. United States Geological Survey. Landsat Missions. URL: https://www.usgs.gov/core-science-systems/nli/landsat/landsat-8?qt-science_support_page_related_con=0#qt-science_support_page_related_con (дата обращения: 25.11.2021).

2. *Гостева А. А., Матузко А. К., Якубайлик О. Э.* Дистанционные методы в изучении температуры поверхности земли в городах (на примере г. Красноярска, Россия) // ИнтерКарто. ИнтерГИС: матер. Междунар. конф. 2018. Т. 24 (ч. 2). С. 195–205.

3. *Weng Q., Lu D., Schubring J.* Estimation of land surface temperature–vegetation abundance relationship for urban heat island studies // Remote Sensing of Environment. 2004. № 89. P. 467–483.

4. GDAL documentation. URL: https://gdal.org/ (дата обращения: 25.11.2021).

5. Matplotlib: Visualization with Python. URL: https://matplotlib.org/ (дата обращения: 25.11.2021).

ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ И СЕТИ

УДК 004.052.2 **DOI: 10.31799/978-5-8088-1701-2-2022-2-138-142**

А. А. Андреев*
студент
Н. А. Балонин*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ДОМЕННАЯ АРИФМЕТИКА, АРИФМЕТИКА ПОЛЕЙ ГАЛУА

Рассматривается случай парного поля, формирующегося из двух полей Галуа разных размеров при помощи Доменной арифметики, предложенной Р. Г. Стэнтоном и Д. А. Спротом. На конкретных примерах поясняется реализация простых и сложных полей Галуа, опирающаяся на полиномиальную арифметику с использованием таблицы «неприводимых многочленов» (нередуцируемых). Приводится простой в реализации алгоритм вычисления номера элемента поля. Показывается приложение теории к решению проблемы вычисления матриц Адамара: разность индексов циклически смещаемых элементов строк заменяется разностью элементов Доменной арифметики, номер элемента результата используется для указания места размещения смещаемого элемента строки.
Ключевые слова: *поле Галуа, Доменная арифметика, циклические матрицы, матрицы Адамара.*

Работа выполнена при поддержке Министерства науки и высшего образования Российской Федерации, соглашение № FSRF-2020-0004.

A. A. Andreev*
Student
N. A. Balonin*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

DOMAIN ARITHMETIC, GALOIS FIELD ARITHMETIC

The case of a pair field formed from two Galois fields of different sizes with the help of Domain Arithmetic proposed by R. G. Stanton and D.A. Sprott is considered. The realization of simple and complex Galois fields based on polynomial arithmetic using the table of «additional polynomials» (irreducible) is explained by concrete examples. The algorithm for calculating the field element number, simple to implement, is given. The application of the theory to the solution of the problem of calculating Hadamard matrices is shown: the difference of indices of cyclically shifted elements of strings is replaced by the difference of elements of Domain Arithmetic, the number of the result element is used to indicate the location of the string entry.
Keywords: *Galois field, Domain arithmetic, circulant matrices, Hadamard matrices.*

Для того чтобы понять смысл Доменной арифметики [1], можно представить арифметику комплексных чисел где вещественная и мнимая составляющая определены в конечных полях, к тому же разной размерности. Для операций сложения и вычитания правила те же, что и в комплексной арифметике, только здесь в качестве действительного числа используется элемент первого поля, а в качестве мнимого числа – элемент второго. При умножении логика комплексной арифметики нарушается, умножение происходит по составляющим, как если бы действительную часть одного комплексного числа умножали на действительную часть другого, а мнимую бы умножали на мнимую.

Таким образом, имеем комплексное поле Galois Domain GD(v), $v = p^n \times q^m$ (с ограничением $q^m = p^n + 2$), т. е. набор элементов (α, β), где $\alpha \in GF(p^n)$, а $\beta \in GF(p^m)$. Сложение и умножение для них тогда будут определены следующими отношениями:

$$(\alpha_1, \beta_1) + (\alpha_2, \beta_2) = (\alpha_1 + \alpha_2, \beta_1 + \beta_2),$$

$$(\alpha_1, \beta_1)(\alpha_2, \beta_2) = (\alpha_1\alpha_2, \beta_1\beta_2).$$

Поля такая арифметика не образует, но мы работаем с группами, а потому деление элементов

не нужно. Тогда приходится решать примеры, в которых размер матриц v=a×b не равен простому числу или его степени, из-за чего матрицы будут наследовать сложной структуре полей Галуа с составным показателем GD($p^n(p^n + 2)$).

Теперь рассмотрим подробнее поле Галуа GF(p^m). В конечном поле показательная функция перекрывает собой все пространство и играет роль спирали, пронизывающей каждую его точку без пересечения, порождая тем самым циклическую группу. Иными словами, в конечном поле так себя ведет тривиальная линейная система первого порядка, описываемая простым разностным уравнением или показательной функцией (экспонентой, в общем ее толковании). Нелинейность и обратная связь здесь незримо присутствуют в формулах реализации полиномиальной арифметики.

Операция умножения в полиномиальной арифметике основана на свертке, которая дает примерно вдвое (менее на 1, чем вдвое) больше коэффициентов, чем нужно. Так, при умножении полинома A с коэффициентами [1, 2, 3, 4] на полином B с коэффициентами [5, 6, 7, 8] получим полином с коэффициентами [5, 16, 34, 60, 61, 52, 32]:

$$(4 + 3x + 2x^2 + x^3)(8 + 7x + 6x^2 + 5x^3) =$$
$$=32 + 52x + 61x^2 + 60x^3 + 34x^4 + 16x^5 + 5x^6.$$

Произведение двух полиномов увеличивает степень полинома, а нам нужно сохранять размер кода результата (т. е. под вектор коэффициентов полинома). Значит, итог произведения должен быть пересчитан в фиксированный по длине вектор. Полином не должен расти сверх порога при умножении. Ни в амплитудах коэффициентов < p, ни в числе параметров < m. Точно так же результат выражения (2·3) mod 4 = 2 цикличен относительно mod 4, принимая значения < 4, только мы должны сделать это с полиномом, а не числами.

Для решения этой проблемы используется вспомогательный полином (полином обратной связи), с помощью которого к коэффициентам младшей части полинома произведения прибавляются старшие. Зависимость несложна для некоторых форм вспомогательных (нередуцируемых, неприводимых) полиномов, схема полиномиального умножения напоминает нейронную сеть (коррекция с весами). Вспомогательный полином дает веса обратных связей в операции умножения, желательно, чтобы большую часть ветвей назад ампутировали нулевые коэффициенты этих ветвей (рис. 1).

При обсуждении полиномиальной арифметики термин «простое число» заменяется термином «неприводимый многочлен» (нередуцируемый). Полином называется неприводимым, если его нельзя представить в виде двух других полиномов (кроме представления в виде единицы и самого полинома). Например, полином $x^2 + 1$ неприводим над целыми числами, это касается также любого конечного поля. Для поля Галуа GF(p^m) в качестве «модулей» (полиномов обратных связей) используются трехчлены, они содержат много нулевых коэффициентов. Чаще всего это $x^m = sx^d + r$, при d = 1 ко всем m младшим коэффициентам свертки прибавляются все старшие с множителями s, r за исключением крайних членов, к младшему коэффициенту произведения не дотягивается ветвь с весом s, к старшему коэффициенту – ветвь с весом r. При d > 1 в этом простом алгоритме возникают поправки [2, 3].

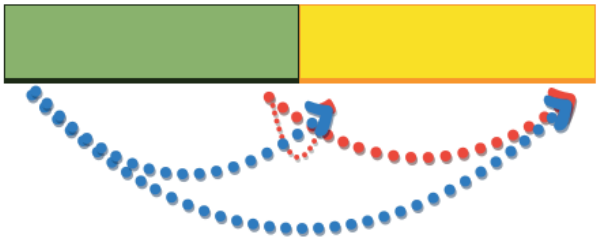


Рис. 1. Неприводимый полином дает веса обратных связей

```
C = conv(A, B); //Результат произведения полиномов
for (i=0; i<m; i++){S=C[i];
if (i<m–1){S+ =r*C[m+i]; if (i<d–1) S+=r*s*C[2*m–d+i];}
if (i>d–1){S+ =s*C[m+i–d]; if (i<2*d–1)S+=s*s*C[2*(m-d)+i];}
C[i]=S%p;}
```

Потребность в s-ветви отпадает совсем для GF(p^2), p – нечетное простое, произведение пары двухкомпонентных чисел (a, b)(c, d) = (ac–rbd), (ad + bc). Это поле сходно с полем комплексных чисел, где –r – аналог –1 (квадрат невычета). Параметр r = 2 помимо случаев r = 1, p + 1 кратно 4, и r = 3, p + 1 кратно 6. Для более сложных полей параметры заданы в табл. 1.

Пример. Возьмем поле GF(p^m) = GF(3^3) = GF(27). Ищем в таблице по первым цифрам p=3 и m=3 вспомогательный многочлен [p,m,d,s,r]. Находим [3,3,1,1,1], а значит d = s = r = 1. Подставляем полученные значения в выражение $x^m = sx^d + r$ и получаем вспомогательный многочлен $x^3 = x^1 + 1$.

Тут m = 3, длина кода элемента A = [*, *, *], где * имеет значения от 0 до 2 < p = 3. Зададим пару элементов и умножим. Конволюция – это умножение пары полиномов по правилам средней школы.

Таблица 1

Степень и весовые коэффициенты

p	m	d	s	r	p	m	d	s	r	p	m	d	s	r
2	2	1	1	1	7	3	1	0	2	19	3	1	0	2
2	3	1	1	1	7	4	1	1	3	19	4	1	1	1
2	4	1	1	1	7	5	1	1	3	19	5	1	1	3
2	5	2	1	1	7	6	1	0	3	23	3	1	1	4
2	6	1	1	1	7	7	1	1	1	23	4	1	1	3
2	7	1	1	1	7	8	1	1	1	23	5	1	1	2
2	8	5	1	1	7	9	1	0	2	29	3	1	1	1
3	3	1	1	1	7	10	1	2	1	29	4	1	1	1
3	4	1	1	1	11	3	1	1	3	31	3	1	0	3
3	5	1	1	1	11	4	1	1	6	31	4	1	1	1
3	6	1	1	1	11	5	1	1	1	37	3	1	0	2
3	7	2	1	2	11	6	1	1	1	37	4	1	1	1
3	8	2	1	1	13	3	1	0	2	41	3	1	0	2
3	9	5	1	1	13	4	1	0	2	41	4	1	1	7
5	3	1	1	1	13	5	1	1	1	47	3	1	1	1
5	4	1	1	1	13	6	1	0	2	47	4	1	1	1
5	5	1	1	1	17	3	1	1	2	53	3	1	1	4
5	6	1	1	3	17	4	1	0	3	53	4	1	1	2
5	7	1	1	2	17	5	1	1	6	59	3	1	1	8
5	8	1	0	2	17	6	1	1	4	59	4	1	1	5

Умножим A = 16 на B = 23. Сперва преобразуем эти числа в полиномы. Тогда

A = [1, 2, 1], так как $16 = 1 + 2x + x^2$, при x = p = 3,

B = [2, 1, 2], так как $23 = 2 + 1x + 2x^2$, при x = p = 3

$$(x^2 + 2x + 1)(2x^2 + x + 2)$$

$$2x^4 + 5x^3 + 6x^2 + 5x + 2.$$

Преобразуем коэффициенты по mod 3 (так как p = 3):

$$2x^4 + 2x^3 + 0 + 2x + 2.$$

Данное выражение нам нужно преобразовать, используя вспомогательный полином $x^3 = x^1 + 1$. Произведем замену:

$$2xx^3 + 2x^3 + 2x + 2,$$

$$2x(x + 1) + 2(x + 1) + 2x + 2,$$

$$2x^2 + 2x + 2x + 2 + 2x + 2,$$

$$2x^2 + 6x + 4.$$

Снова преобразуем коэффициенты по mod 3:

$$2x^2 + 0 + 1,$$

$$1 + 0 + 2x^2,$$

C = [1, 0, 2].

Подставим в полином x = 3 и получим C = 19. Тем самым установили, что 16×23 = 19 в поле GF(27).

Вариант реализации умножения в коде будет иметь вид:

```
function gfmul(A,B) {
var i, j, n, mi, ni, i1, i2, c, C, S;
C=conv(A,B); // Результат произведения полиномов
n=m–d; i1=d–1; i2=i1+d;
for (i=0; i<m; i++) {S=C[i]; j=i+1;
if (j<m) {mi=m+i; c=C[mi]; if (i<i1) c+=s*C[mi+n]; S+=r*c;}
if (j>d) {ni=n+i; c=C[ni]; if (i<i2) c+=s*C[ni+n]; S+=s*c;}
C[i]=S%p;}
```

Если же говорить о сложении и вычитании, то здесь все просто. Для сложения мы покоординатно складываем m координаты элемента поля по модулю p. Для вычитания мы покоординатно вычитаем m координаты элемента поля по модулю p. Чтобы элементы остались положительными, нужно добавить к разности максимальное положительное число, решающее проблему:

```
function gfadd(A,B) {
var i, C; C=one(m); // вектор единиц размера m x 1.
for (i=0; i<m; i++) C[i]=(A[i]+B[i])%p;
return C;}

function gfsub(A,B) {
var i, p2, C; p2=p*p; C=one(m);
for (i=0; i<m; i++) C[i]=(p+A[i]–B[i])%p;
return C;}
```

Тем же правилам будет подчинятся и Доменная арифметика, только формироваться она будет иначе (из двух Галуа).

При формировании комплексного поля Galois Domain GD(v) вместо дифференциального набора x формируется для заданных начального w и начального z дифференциальный набор по

$$\left(z^0, z, z^2, ..., z^{\frac{1}{2}\left(s^2-1\right)-1}, 0, w^0, w, ..., w^{s-2} \right),$$

где $s = p^n$, $s + 2 = q^m$, v = s(s + 2).

Алгоритм, реализующий Доменную арифметику Стэнтона – Спрота разработал профессор Санкт-Петербургского государственного университета аэрокосмического приборостроения Н. А. Балонин [4]. Пример сформированной этим алгоритмом матрицы Адамара порядка 100 приведен ниже. Основа (core) – это блок размера 99 = 9 × 11, т. е. n = 3, m = 1. Элементы дифференциального набора, начиная с 0, образованы степенями элементов w = [5, 0], z = [5, 2]

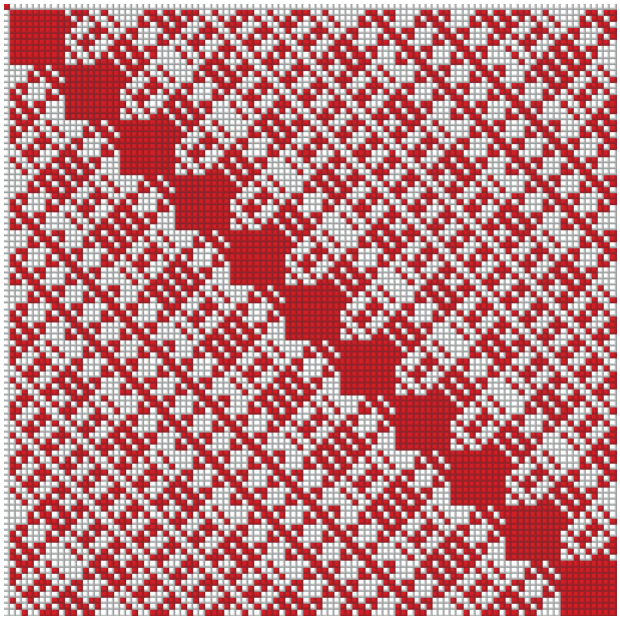


Рис. 2. Stanton – Spott – Balonin algorithm: 1B1M order 100

из GD(99) с 99 элементами вида [0, 0], [1, 0], …, [9, 0], [0, 1], [1, 1], …, [9, 1], …, [0, 11], [1, 11], … , [9, 11], их порядковые номера дадут адреса –1, выделенных цветом в первой строке основы (рис. 2).

Элементы строки i получены сдвигом элементов стартовой строки путем перемещения на места с номерами i–j, вычитание производится над элементами Доменной арифметики с номерами $i, j < 99$. Иными словами, мы пользуемся i, j как указателями на ячейки памяти, в которых расположены вычитаемые бинарные элементы, а номер элемента результата вычитания укажет на то место, куда переместился элемент первой строки с образованием узора основы матрицы Адамара (матрицы Мерсенна [2]).

Заключение

Последовательное развитие теории матриц после выхода статьи Стэнтона и Спротта отражено в работах [5, 6]. В работах [7, 8] обсуждаются новые семейства матриц Адамара, а в работе [9] состав семейств расширен рассмотрением продуктов ортогонализации основ (cores) матриц Адамара. В работах [10, 11] отражен рост интереса к численным методам анализа матриц для порядков, для которых нет ни одного, ни двух полей Галуа. Также обширна практика программирования, организация программных комплексов [12–14] и программ клиент-серверного поиска [15, 16] в сети Интернета. Комплексы доступны для скачивания на сайте облачных вычислений [4]. Все это свидетельствует об успешном включении ГУАП в работы международных коллективов с ведущими учеными мира из этой области.

Библиографический список

1. *Stanton R. G., Sprott D. A.* A family of difference sets // Canad. J. Math. 1958. № 10. p. 73–77.

2. *Балонин Н. А., Сергеев М. Б.* Специальные матрицы: псевдообратные, ортогональные, адамаровы и критские. СПб.: Политехника, 2019. 196 с.

3. *Балонин Н. А., Сергеев М. Б.* Ортогональные преобразования: учеб. пособие. СПб.: ГУАП, 2018. 57 с.

4. *Балонин Н. А.* Облачные вычисления «MATHSCINET.RU». URL: http://www.mathscinet.ru (дата обращения: 06.11.2021).

5. *Seberry J., Yamada M.* Hadamard Matrices: Constructions using number theory and linear algebra. Wiley, 2020. 384 p.

6. *Kharaghani H., Tayfeh-Rezaie B. A.* Hadamard matrix of order 428 // Journal of Combinatorial Designs. 2005. Vol. 13. P. 435–440.

7. *Acevedo S., Dietrich H.* New infinite families of Williamson Hadamard matrices // Australian Journal of Combinatorics. 2019. Vol. 73 (1). P. 207–219.

8. *Seberry J., Balonin N. A.* Two infinite families of symmetric Hadamard matrices // Australian Journal of Combinatorics. 2017. Vol. 69 (3). P. 349–357.

9. *Mohan M. T.* p-almost Hadamard matrices and λ-planes // Journal of Algebraic Combinatorics. 2020. 20 p.

10. *Востриков А. А., Дехканбаев Д. С., Куртяник Д. В., Сергеев А. М.* О стратегиях вычисления матриц Адамара симметричных структур блочной конструкции Балонина-Себерри // Телекоммуникации. 2020. № 5. С. 20–27.

11. *Абузин Л. В., Балонин Ю. Н., Куртяник Д. В., Сергеев А. М.* Генерация, фильтрация и поиск экстремума в сверхбольшом каталоге бинарных последовательностей // Обработка, передача и защита информации в компьютерных системах: Первая Всерос. науч. конф. (СПб, 14–22 апр. 2020 г.): сб. докл. СПб.: ГУАП, 2020. С. 121–124.

12. Свидетельство о государственной регистрации программы для ЭВМ № 2018616389 от 01.06.2018 г. Программный комплекс поиска матриц локального максимума детерминанта с самомасштабированием / Балонин Ю. Н., Сергеев А. М., Синицына О. И.

13. Свидетельство о государственной регистрации программы для ЭВМ № 2019615126 от 18.04.2019 г. Программа генерации матричных рап-

портов «Калейдоскоп» / Сергеев М. Б., Сергеев А. М., Балонин Н. А., Балонин Ю. Н.

14. Свидетельство о государственной регистрации программы для ЭВМ № 2018616390 от 01.06.2018 г. Программный комплекс поиска бициклических матриц на основе таблицы перекрестных ссылок / Балонин Ю. Н., Сергеев А. М.

15. Свидетельство о государственной регистрации программы для ЭВМ № 2018617112 от 19.06.2018 г. Программный комплекс клиент-серверного поиска бициклических матриц Адамара в реальном масштабе времени / Балонин Ю. Н., Сергеев А. М.

16. Свидетельство о государственной регистрации программы для ЭВМ № 2020662384 от 13.10.2020 г. Накопление пар ортогональных последовательностей для поиска симметричных ортогональных матриц Адамара с тремя блоками (Пропусов) / Балонин Ю. Н., Сергеев А. М.

УДК 004.946 **DOI: 10.31799/978-5-8088-1701-2-2022-2-143-147**

А. А. Антипова*
магистрант
А. В. Никитин*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РАЗРАБОТКА НА ОСНОВЕ ВИРТУАЛЬНОЙ РЕАЛЬНОСТИ ОБУЧЕНИЯ КОММУНИКАТИВНОМУ НАВЫКУ

Рассмотрены понятие и виды коммуникативных навыков, способы их изучения, основные проблемы реализации и решение их на основе виртуальной реальности.
Ключевые слова: *коммуникативные навыки, виртуальная реальность, методика.*

A. A. Antipova*
Postgraduate Student
A. V. Nikitin*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

DEVELOPMENT BASED ON VIRTUAL REALITY FOR TEACHING COMMUNICATION SKILLS

The concept and types of communication skills, methods of studying them, the main problems of implementation and their solution based on virtual reality are considered.
Keywords: *communication skills, virtual reality, methodology.*

В современном мире все чаще встречается понятие soft skills, или гибкие навыки, означающее умение общаться с людьми, эффективно организовывать свое время, творчески мыслить, принимать решения и брать на себя ответственность. Эти навыки играют одну из самых важных ролей при трудоустройстве: владение ими обеспечивает быстрое построение карьерной лестницы и высокий заработок. Коммуникативный навык с приходом виртуализации во многие области жизни людей – один из самых востребованных [1]. Переход к дистанционной работе привел к увеличению числа онлайн-взаимодействий взамен живому общению, которое, однако, по-прежнему является неотъемлемой частью человеческих потребностей.

Коммуникация – это процесс взаимодействия между различными субъектами коммуникации, при котором осуществляется обмен информацией. Коммуникативный процесс носит двусторонний характер и предполагает последовательность этапов формирования, приема, декодирования и использования информации при ее обмене коммуникантами [2].

Принято выделять следующие виды коммуникативных навыков:

— «информационно-коммуникативные, под которыми подразумевается способность вступать в общение, ориентироваться в собеседниках и ситуациях, соотносить разные способы общения (вербальные/невербальные)» [3];

— «регуляционно-коммуникативные, которые включают в себя способность согласовывать собственные действия и слова с потребностями собеседника, доверять, оказывать поддержку, помощь, использовать свои личные навыки при коллективном решении проблем, оценивать результат совместного общения» [3];

— «аффективно-коммуникативные, которые определяют способность делиться собственными чувствами и интересами с собеседником, оценивать его эмоциональное поведение, проявлять эмпатию (быть отзывчивым, сопереживать, заботиться)» [3].

Базовым можно назвать информационно-коммуникативный навык, на основе которого строятся все остальные, который задействует как вербальное (язык, речь), так и невербальное (мимика, жесты) поведение коммуникантов, например:

– выражать просьбу;

– приветствовать другого человека, приглашать, вежливо общаться;

– придерживаться правил поведения с коллегами, руководством и т. д.;

– понимать ситуацию общения, намерения и мотивы партнеров;

– умение слушать другого человека;

– использовать как слова вежливости, так и язык тела – различные жесты, мимику;

– адекватно выражать свои мысли и эмоции [3].

Есть несколько самых известных вариантов изучения soft skills, к которым относится и коммуникативный навык.

1. Физический (реальный) представляет собой чаще всего обычные занятия, на которых ведутся беседы с участниками процесса и им преподносится различный лекционный материал, данный сценарий реализуется путем подготовки материала лектором/преподавателем, который входит в состав планировки занятий. Преимущества: общение напрямую с людьми входит в само понятие коммуникативного навыка, если подход индивидуальный, вероятно быстрое освоение недостающего материала. Недостатки: soft skills как таковые не являются дисциплиной в учебных заведениях или отдельно не подаются как курс, что затрудняет поиск специалистов в этой области, предполагается, что это те навыки, которые уже заложены у человека и обучаться им можно в процессе всей жизни методом проб и ошибок.

2. Интерактивный двумерный (с использованием компьютера/смартфона). Предполагается наличие интерактивного приложения с возможностью взаимодействия по сценарию с объектами внутри двумерной сцены, включают незамысловатый интерфейс для удобства работы, бывают в виде сферических панорам или приложений с тестовой частью. Преимущества: мобильность доступа к материалу, возможность разделения объема изучаемой информации, интерактивное взаимодействие напрямую с объектами позволяет удерживать внимание пользователя. Недостатки: по большей части информация подается сжато, сами сцены статичны, недостаток интерактивности и правдоподобности, что может быстро наскучить пользователю, а также сам процесс отличен от живого общения, которое протекает более естественным образом.

3. Интерактивный трехмерный (с использованием обычного компьютера). Иинформация подается посредством организации трехмерной сцены с определенным сценарием, а пользователь участвует в освоении материала, используя такие инструменты, как клавиатура, мышь и монитор (для получения графической информации). Преимущества: доступнее, чем специализированные курсы, так как большинство современных людей имеют компьютер на работе или дома, при этом отличие подачи материала от интернет-уроков или видеоматериалов в том, что пользователь интерактивно и напрямую взаимодействует с трехмерными реалистичными объектами на сцене, что, можно сказать, приближает его опыт к реальному; в любой момент можно начать изучать все заново или уйти без последствий. Недостатки: получаемый опыт отличается от живого общения с людьми, а взаимодействие со сценой монитор–клавиатура–мышь не всегда обеспечивает нужное погружение.

4. Интерактивный трехмерный (с использованием специального оборудования). Информация, как и в предыдущем варианте, подается с помощью трехмерной сцены с отработанным сценарием, но теперь интерактивность и погружение достигаются в большей степени за счет использования контроллеров и шлема. Внутри виртуальной реальности, надевая шлем, человек буквально переносится в различные места и отдается происходящему. Преимущества: погружение близкое к реальному, с помощью VR можно достичь максимальной вовлеченности пользователя в процесс. Недостатки: получаемый опыт все еще проигрывает реальному, но в виртуальном мире возможно отработать большее количество возможных вариантов без влияния на репутацию, вероятно ухудшение состояния здоровья при долгом пребывании внутри виртуальной среды.

В XXI в. существует возможность изучать материал в удобном месте и в удобное время, используя последние технологические решения, что позволит компаниям сэкономить средства на организации обучения внутри предприятия или же обычным людям не проходить курсы, в которых информация может быть тяжела для восприятия. В соответствии с этим предлагается представить необходимый материал посредством виртуальной реальности, которая уже используется в различных обучающих проектах, в данном случае благодаря возможностям виртуальной среды возможно достичь близкого к реальному опыту. Таким образом, возникает проблема: каким образом организовать переход от очного обучения коммуникативного навыка к обучению с помощью виртуальной среды? Какие свойства коммуникативного навыка можно промоделировать на основе средств виртуаль-

ной реальности и тем самым повысить эффективность обучения? Для решения поставленной проблемы необходима разработка научно обоснованной методики перехода от очного обучения гибким навыкам к получению их с использованием VR-технологий, что определяет актуальность темы исследования.

Назначение НИР – создание посредством виртуальной среды метода обучения людей социальным взаимодействиям, которые составляют часть гибких навыков, основанных на социальных компетентностях, большинство из которых связано непосредственно с грамотным изложением своих мыслей и умением взаимодействовать в обществе. Предполагаемые технологии обучения представлены в виде схем на рис. 1.

Рассмотрим цикличную модель коммуникации Г. Д. Лассуэлла (1948 г.) (рис. 2).

На основе информационно-коммуникативного навыка можно представить простейшую сценографию классической модели начала диалога: подойти к отправителю, но не вплотную, поприветствовать (предать сообщение через один из каналов), получить на него ответ и поддержать беседу.

Таким образом, а качестве аналогов использовались приложения, содержащие взаимодействия пользователя с искусственным интеллектом (аватаром). Анализ подходящих аналогов представлен в табл. 1.

Аналог № 1. В диссертации «Design of Immersive Virtual Reality System to Improve Communication Skills in Individuals with Autism» ученых из Катара представлено приложение по освоению коммуникативного навыка детьми с аутизмом. В игре организовано распознавание речи, наличие аватара пользователя (вид

Рис. 1. Взаимодействие неигровых персонажей и пользователя в форме диалога и схема его построения

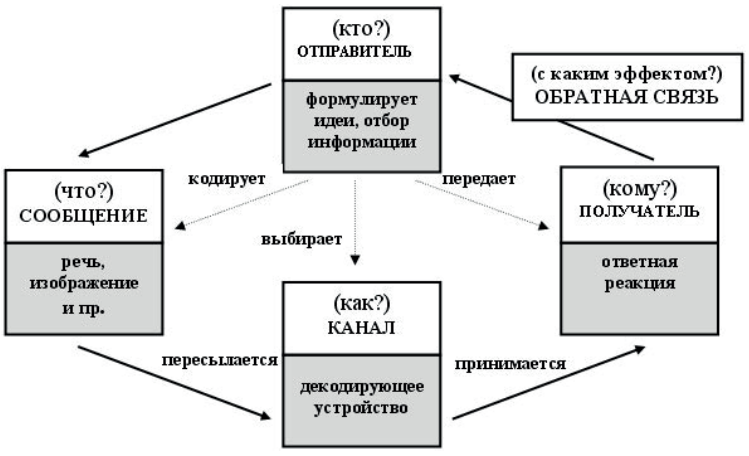


Рис. 2. Модель коммуникации Г. Д. Лассуэлла

Таблица 1

Сравнение аналогов

Аналог № 1	Аналог № 2	Заявляемый объект
Использование иммерсивной виртуальной среды	+	+
Манипуляция объектами внутри сцены	–	+
Реалистичность объектов виртуальной среды	+	+
Распознавание ключевых выражений	+	+
Контроль за действиями пользователя	+	+
Указание ошибок	+	+
Навигация в среде	–	+

от 3-го лица), аватаров учителей и учеников и организация виртуальной площадки для тренировок навыка, которая оснащена автонавигацией [4]. В качестве недостатков можно отметить отсутствие самостоятельного передвижения пользователя (часть навигации с исследованием и поиском), что ограничивает приближение к реальным условиям, недостаточно проработанные диалоги, ограниченные лишь приветствием и представлением себя, малое количество заданий и недостаточный уровень интерактивности.

Аналог № 2. Приложение, разработанное Институтом творческих технологий Университета Южной Калифорнии, нацелено на тренировку навыка общения на интервью для лиц, побывавших на местах боевых действий. Оно ориентировано на ПК, пользователь выбирает разные уровни сложности интервьюверов и по их реакции может следить за правильностью выбранных им ответов [5]. К недостаткам можно отнести ограниченный функционал, недостаточный уровень интерактивности и ограничение по аудитории.

Для рассмотрения результатов анализа свойств коммуникативного навыка средствами виртуальной реальности необходимо дать ей общее определение: «Виртуальная реальность – разновидность человеко-компьютерного интерфейса, обеспечивающего присутствие человека в моделируемой среде посредством поддержки его погружения и взаимодействия с объектами этой среды в реальном времени» [6].

В данной научной работе используется как погружение, так и интерактивность. Задача иммерсивности (погружения) – полностью вытеснить из сознания пользователя реальность, создать иллюзию пребывания «там» (в модели), погрузить его в созданный компьютером мир [6]. Данный подход позволит создать условия, приближенные к реальным, и в полной мере отрабатывать необходимый навык.

За счет использования свойств виртуальной реальности возможно приблизиться к реализации реалистичных ситуаций и достичь наилучшей продуктивности в изучении материала. Во взаимодействии пользователя с системой делается упор на сенсорно-моторные навыки с помощью свойств виртуальной среды.

1. Погружение, за счет которого достигается реалистичность пребывания в среде:

– визуальное – пользователь наблюдает за происходящим в сцене, замечает изменения и ориентируется в пространстве, например в ПК-версии это осмотр с помощью перемещений мыши;

– аудиальное – пользователь слушает игровые звуки и другую аудиоинформацию приложения через наушники/динамики, например звук стука в дверь, шагов;

– форма (позиция восприятия) – от первого лица, для достижения реалистичности диалога между пользователем и интеллектуальными аватарами.

2. Вовлеченность – пользователь за счет сквозного сюжета сосредоточивается на происходящем, выполняя задания и участвуя в событиях внутри обучающего приложения.

3. Присутствие:

– личностное – пользователь представляется в виде собственного аватара и способен принимать решения;

– средовое – интеллектуальные аватары реагируют на пользователя и его реплики.

4. Иинтерактивность:

– навигация – с помощью джойстика/контроллеров/клавиатуры пользователь способен осматриваться (поворот на 180 градусов вверх-вниз и влево-вправо) и перемещаться по сцене и локациям, также внутри игры по сюжету могут ставиться задачи исследования и поиска, что может улучшить ориентацию в пространстве для выполнения заданий;

– взаимодействие со средой посредством селекции – пользователь с помощью контроллера/мыши способен выбирать реплики и объекты для взаимодействия;

– система управления и манипуляция – с помощью джойстика/контроллеров/мыши и клавиатуры пользователь может выполнять взаимодействия с объектами внутри приложения, например постучать в дверь, прежде чем зайти в аудиторию, взять со стола документ и отнести его аватару. что необходимо для выполнения

некоторых заданий по усвоению информационной части.

На рынке недостаточно приложений с использованием VR, направленных на изучение коммуникативных навыков с возможным влиянием пользователя на сюжет, интерактивностью, погружением и вовлеченностью. Реализация итогового геймифицированного приложения обусловлена тем, в игровом формате подача информации будет более интересна и легкоусваиваема, а влияние на сюжет позволит удерживать внимание пользователя.

Реализация прототипа организации взаимодействия пользователя с интеллектуальными аватарами предполагается через создание триггер-зон, при пересечении которых интеллектуальный аватар реагирует на пользователя и вступает с ним в диалог. Сами диалоги реализуются путем системы управления, состоящей из панелей на экране, которые пользователь периодически выбирает.

Согласно приведенной модели (см. рис. 2), предлагается два варианта взаимодействий пользователя и аватаров.

1. Отправитель (пользователь) передает сообщение в виде изображения с текстом по визуальному каналу неигровому персонажу (например, интеллектуальному аватару Куратора).

2. Отправитель (Куратор) передает сообщение в виде изображения с текстом по визуальному каналу пользователю.

Пример: пользователь задет вопрос, как добраться до аудитории, и адресует его Куратору (выбирает нужную реплику на экране во время диалога с Куратором). Куратор получает вопрос и выдает ответ о месторасположении нужного объекта пользователю (перед пользователем появляется ответ куратора на экране). Куратор также может задавать вопросы, на которые пользователь может давать ответы путем выбора реплик из нескольких предложенных. Таким образом строится диалог.

В настоящее время на основе анализа свойств коммуникативного навыка разработан прототип на основе виртуальной реальности, позволяющий предварительно оценить принятые решения с подготовкой экспериментально проверить его на пользователях.

Библиографический список

1. Какие soft skills нужны в IT и как их развивать. URL: https://issoft.by/blog/kakie-soft-skills-nuzhny-v-it-i-kak-ikh-razvivat/ (дата обращения: 18.09.2021).

2. *Викулова Л. Г., Шарунов А. И.* Основы теории коммуникации: практикум. М.: АСТ: АСТ МОСКВА: Восток – Запад, 2008. 322 с.

3. Коммуникативные навыки // Живое дело: пресс-центр. URL: https://zhyvoedelo.com/ru/news/kommunikativnye-navyki#:~:text=основные %20разновидности20%коммуникативных20%навыков,-коммуникативные20%и20%аффективно-коммуникативные (дата обращения: 16.11.2021).

4. International Journal of Emerging Technologies in Learning (iJET). Design of Immersive Virtual Reality System to Improve Communication Skills in Individuals with Autism. URL: https://www.researchgate.net/publication/317266196_Design_of_Immersive_Virtual_Reality_System_to_Improve_Communication_Skills_in_Individuals_with_Autism (дата обращения: 27.11.2021).

5. 5 лучших способов использования виртуальной реальности для обучения мягким навыкам. URL: https://www.immersivelearning.news/2020/04/09/5-best-uses-of-vr-for-soft-skills-training/ (дата обращения: 19.10.2021).

6. *Булгаков Д. А., Никитин А. В., Решетникова Н. Н., Ситников И. А.* Разработка виртуальной и дополненной реальности: учеб. пособие / под ред. проф., д. т. н. М. Б. Сергеева. СПб., ГУАП, 2021. 194 с.

УДК 004.921 **DOI: 10.31799/978-5-8088-1701-2-2022-2-148-154**

И. С. Артемьев*
студент
Н. Н. Решетникова*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РЕТОПОЛОГИЯ 3D-МОДЕЛИ ПЕРСОНАЖА ДЛЯ ИНТЕРАКТИВНЫХ СЦЕН

Рассматриваются технология 3D-моделирования персонажа и процесс ретопологии модели. Ретопология помогает оптимизировать 3D-модель за счет снижения количества полигонов с сохранением высокой степени реалистичности внешнего вида при визуализации в интерактивных сценах.
Ключевые слова: *технология 3D-моделирования персонажа, ретопология, редактор Blender 3D, интерактивность.*

I. S. Artemiev*
Student
N. N. Reshetnikova
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

RETOPOLOGY 3D CHARACTER MODEL FOR INTERACTIVE SCENES

The technology of 3D character modeling and the description of the retopology process of the model are considered. Retopology helps to optimize the 3D model by reducing the number of polygons while maintaining a high degree of realistic appearance when visualizing in interactive scenes.
Keywords: *3D character modeling technology, retopology, Blender3D software, interaction.*

Сегодня моделирование и анимация 3D-персонажей занимают одно из значимых мест как в создании компьютерных игр, так и в научной сфере, в обучающих курсах и компьютерных симуляторах, где от внешнего облика персонажа зависит уровень восприятия виртуального пространства пользователем. При использовании в интерактивных 3D-приложениях эти модели должны быть оптимизированы, т. е. иметь небольшой полигонаж, но при этом достаточно высокое качество визуализации для реалистичного восприятия, поскольку несоответствие даже малых деталей может испортить впечатление обо всей модели.

Сложившийся процесс создания анимированного персонажа в пакетах 3D-моделирования можно условно разделить на следующие основные этапы [1]:

– проектирование и разработка модели персонажа;

– создание скелета, т. е. иерархической системы костей;

– связывание модели и скелета (скиннинг/риггинг);

– создание контроллеров движения и анимационных клипов;

– подготовка модели и анимационных клипов к экспорту;

– последующее воспроизведение 3D-модели персонажа и его анимаций.

В свою очередь каждый из перечисленных этапов разделяется на ряд нетривиальных, а часто трудоемких и затратных по времени процедур, которые требуют профессиональных навыков при их выполнении.

Рассмотрим подробнее этап разработки 3D-модели персонажа, который включает:

– создание заготовки 3D-модели на основе концепта (эскиза или фотографии);

– создание высокополигональной 3D-модели;

– ретопологию высокополигональной 3D-модели;

– создание UV-развертки полигональной модели;

– запекание карт нормалей;

– создание текстурных карт.

При создании высокополигональной 3D-модели персонажа редко обращается внимание на то, как будут располагаться полигоны готового объекта. Часто дизайнеры и разработчики 3D-моделей в процессе проработки объектов не ограничивают себя строгими рамками и

случается, что результатом их творчества становится проработанный до мельчайших деталей, узнаваемый образ персонажа, который сложно преобразовать в интерактивный вид из-за огромного количества полигонов. Поэтому за созданием высокополигональной 3D-модели персонажа следует ретопология – создание новой сетки полигонов с меньшим разрешением. Прежде чем перейти к описанию процедуры ретопологии, приведем необходимые определения и пояснения.

Представление 3D-моделей

Модели 3D-объектов состоят из вершин, ребер, граней или полигонов. Оптимальным решением считается, когда модель состоит в большей степени из полигонов. Полигон – прямоугольник, у которого есть перспективное искажение (рис. 1). Из таких элементов можно составить многогранник практически любой формы.

Если полигонов будет много, ими легко покрыть, например, модель человеческого лица.

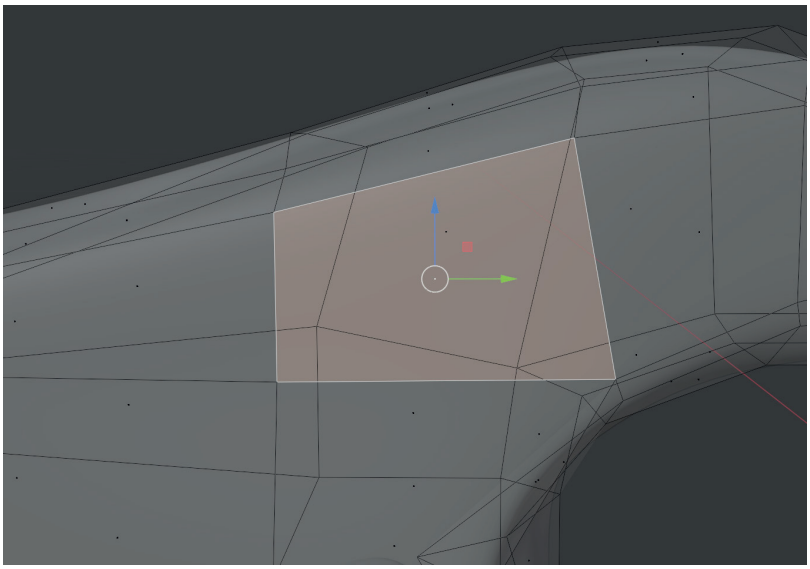


Рис. 1. Пример полигона

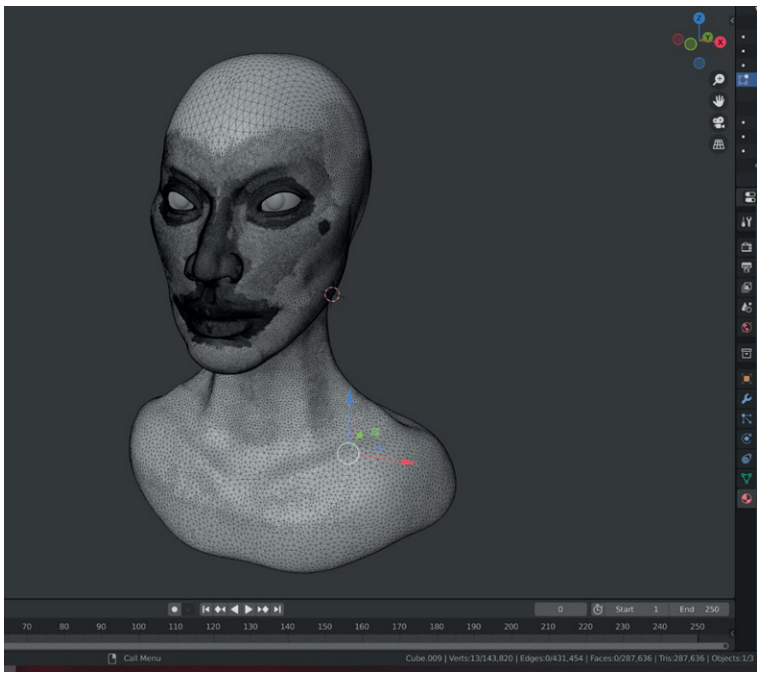


Рис. 2. Пример меша

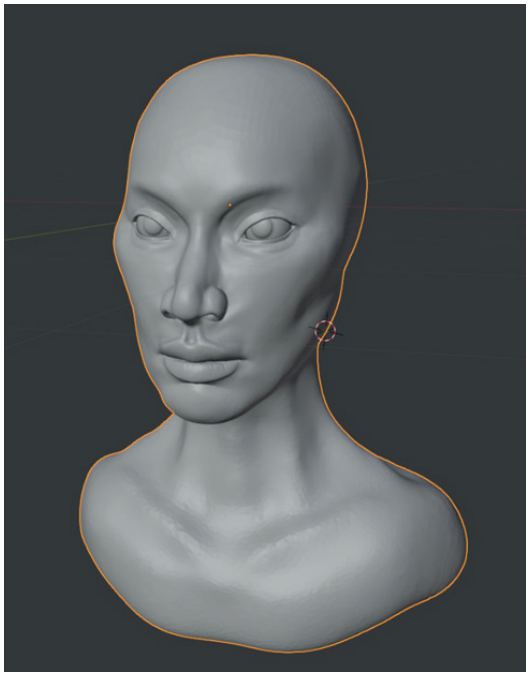


Рис. 3. Высокополигональная модель

На более плоских поверхностях будет использовано минимальное количество полигонов, а выпуклые, рельефные детали (глаза, нос, губы) окажутся покрыты десятками примитивных фигур (рис. 2). Чем больше полигонов задействовано, тем более реалистичным, ровным и проработанным выглядит результат (рис. 3).

Из определенного множества взаимосвязанных полигонов можно составить меш, сетку любой модели (см. рис. 2). Меш или сетка – совокупность вершин, ребер и полигонов, которые составляют 3D-объект. Слово «меш» происходит от английского mesh «ячейка сети», «сетка» – от английского wireframe «каркас / проволочный каркас».

Модели 3D-объектов можно условно разделить на 3 категории [2] в зависимости от характера построения меша:

1) Hi-poly (высокополигональные модели). Наиболее детализированные объекты, которые могут содержать 1–3 млн полигонов. Отличаются детальной прорисовкой и большим весом.

2) Midl-poly (среднеполигональные модели). Объекты средней детализации. Оптимальный вариант для консольных игр и проектов для персонального компьютера.

3) Low-poly (низкополигональные). Содержат всего 5–10 тыс. полигонов. Их использование экономит ресурсы, поэтому игры с такими моделями в большинстве своем предназначены для мобильных устройств, а также дополненной и виртуальной реальности.

Топология

Моделирование – сложный и кропотливый процесс, во время которого следует всегда обращать внимание на то, как располагаются вершины, грани и полигоны 3D-модели. Часто случается так, что топология объекта неэффективна, т. е. имеются многоугольники, очень плотная сетка полигонов и т. д. Неэффективная топология занимает много памяти из-за большого количества вершин модели и создает проблемы при связывании модели и скелета персонажа (рига) и в дальнейшем его анимации.

Топология как раз и описывает, как именно полигоны формируют 3D-модель [3]. Одну и ту же модель можно описать разной топологией (рис. 4).

В 3D-моделировании всегда нужно следить за топологией и ее правильностью. Правильная топология служит двум целям [3]:

– правильные деформации во время анимации;

– использование минимального количества полигонов для описания нужной формы.

Иногда 3D-модель может содержать большое количество полигонов и иметь высокую степень детализации, но что делать, если необходимо сохранить качество, а требуется уменьшить количество полигонов в разы? Здесь на помощь приходит ретопология.

Ретопология

Ретопологию применяют для сохранения формы и высокого качества 3D-модели при ее визуализации с уменьшением детализации полигональной сетки. Ретопология буквально означает перестройка топологии. Обычно, ретопологию делают на основе высокополигональной 3D-модели.

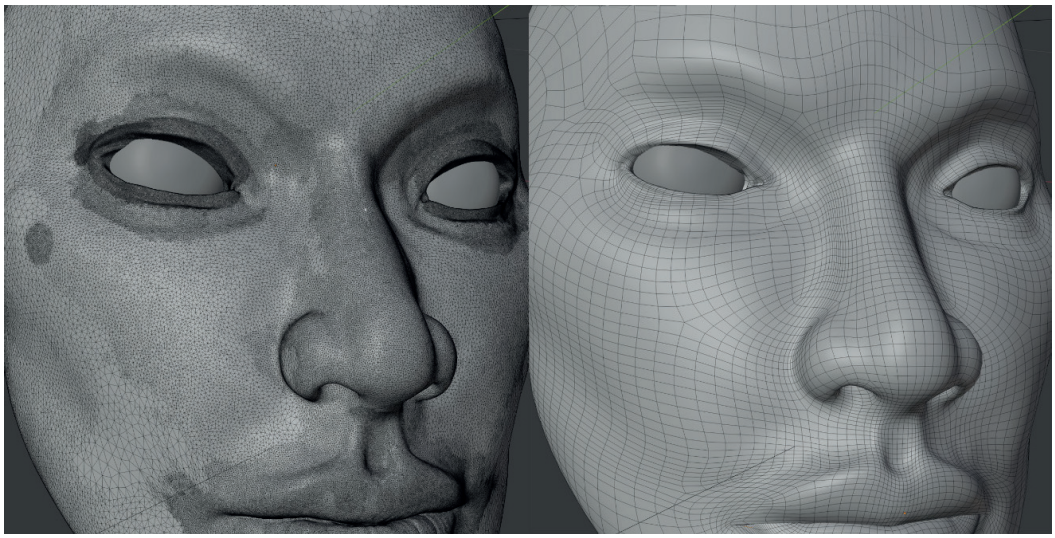


Рис. 4. Разные типы топологии

На рис. 2 и 3 приведен пример высокополигональной модели головы персонажа, созданной с помощью инструментов для скульптинга в пакете Blender 3D [4]. Процесс моделирования при этом напоминает лепку глиняной скульптуры с использованием всевозможных кистей, каждая из которых обладает особенным поведением при взаимодействии с моделью персонажа. В ходе процесса детализации у модели увеличивается количество полигонов, что позволяет наносить более мелкие детали.

Модель имеет большое количество полигонов – 287 636 и неправильный меш (см. рис. 2) из-за применения модификатора DynoTopo для создания динамической топологии (непостоянного размера полигонов в процессе скульптинга).

Создадим ретопологию, начав с добавления объекта Plane на сцену. И применим к нему обязательные модификаторы (рис. 5):

– Shrinkwarp, чтобы низкополигональная модель лучше «обволакивала» высокополигональную;

– Mirror, чтобы сохранить симметрию и автоматизировать создания зеркальной части;

– Multires, чтобы увеличить плотность пикселей модели и в дальнейшем запечь из нее карты нормалей.

Затем необходимо создать низкополигональную версию скульпта, т. е. как бы восстановить новую Low-poly 3D-модель на основе Hi-poly, как показано на рис. 6, 7. Также во время создания (если речь идет о создании модели человека) полигоны лучше разбивать на цветовые группы, которые будут означать различные зоны лица. В рассматриваемом примере красный – зона губ, розовый – область вокруг губ, зеленый – область глаз и век.

Когда результат будет удовлетворять исходным требованиям, можно остановиться, применить все нужные модификаторы и выставить нужное значение для разбиения полигонов у модификатора Multires. В цикле разработки Multires является ключевым модификатором, потому что в пакете Blender 3D предусмотрено оптимальное создание карты нормалей из высокополигональной версии модели после создания ретопологии.

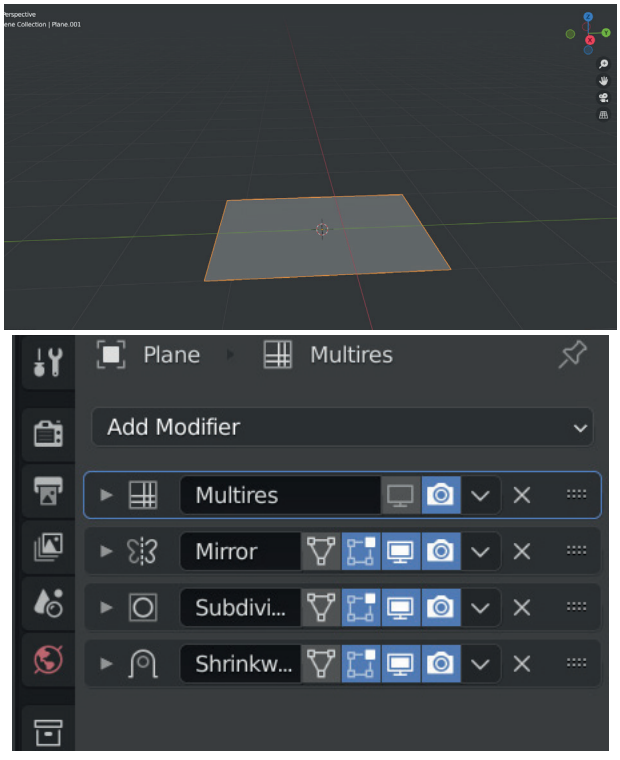


Рис. 5. Объект Plane и список модификаторов

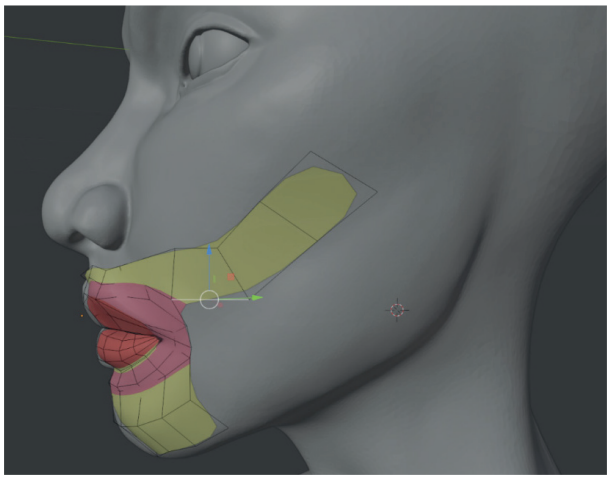


Рис. 6. Пример создания ретопологии

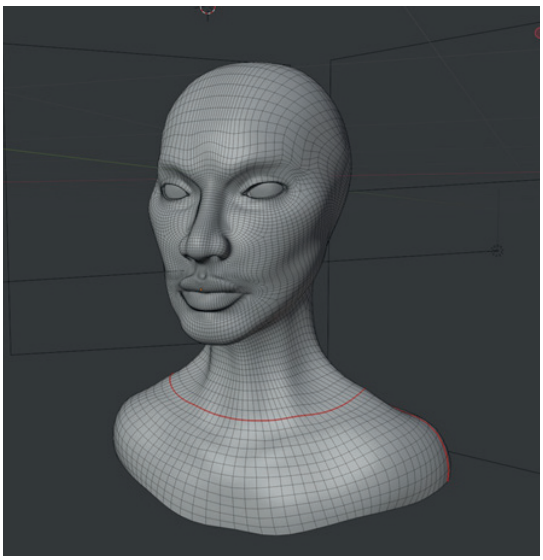


Рис. 7. Готовая ретоп-модель

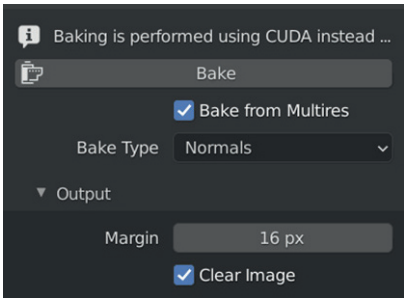


Рис. 8. Запекание карт нормалей

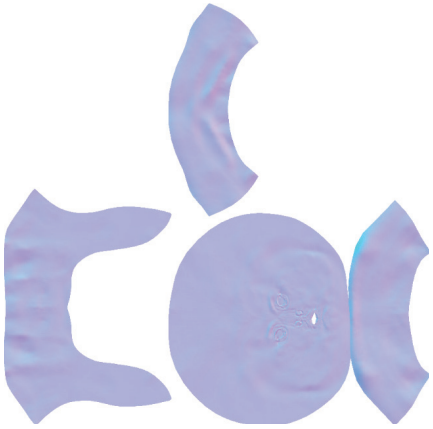


Рис. 9. Карта нормалей

Таким образом, ретопология 3D-модели выполняется для того, чтобы:

– создать правильную полигональную сетку;

– запечь карту нормалей и карту высот (Bump).

После создания правильной ретопологии можно повысить количество ее полигонов, тем самым за счет модификатора Shrimpwarp повысится и качество новой модели, поскольку она подгоняет полигоны под форму созданного ранее скульпта.

Теперь можно запечь карты нормалей и карты высот (если в пункте Bake Type (рис. 8) выбрать Bump) из высокополигональной версии модели.

На выходе получим карту нормалей (рис. 9), которая при визуализации вернет модели прежний уровень детализации, но не будет требовать большое количество вычислительных ресурсов за счет оптимизации геометрической формы полигональной сетки, построенной в результате ретопологии.

Применив карту нормалей к 3D-модели головы персонажа (рис. 10), получим практически идентичный уровень детализации, что и в скульпте.

Оптимизированная топология модели, содержащая 20 тыс. полигонов (на порядок меньше, чем высокополигональная модель), показана на рис. 11.

После переработки топологии гораздо проще развернуть модель на 2D-плоскость для создания UV-развертки [5] и текстурирования. Пакет Blender включает редактор материалов, что позволяет создать необходимые текстуры для

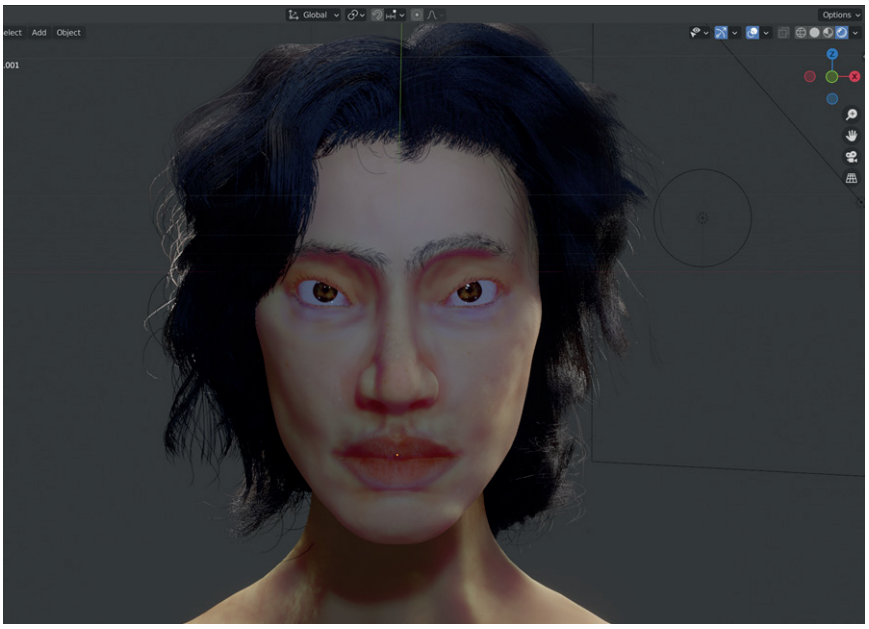


Рис. 10. Низкополигональная модель с картой нормалей

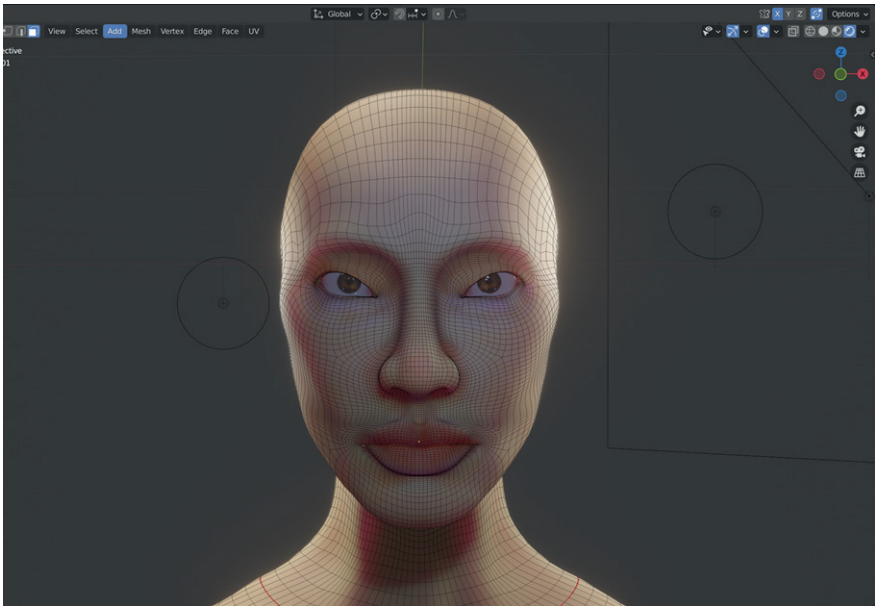


Рис. 11. Топология низкополигональной модели

Рис. 12. Итоговая 3D-модель головы персонажа

дальнейшего запекания и экспорта в игровой движок при разработке интерактивного 3D-приложения [6].

Далее можно экспериментировать с различными комбинациями силы карт нормалей и высот, добавить дополнительные карты, расставить освещение и камеру, а затем визуализировать построенную 3D-модель (рис. 12).

Ретопология помогает оптимизировать 3D-модель за счет снижения количества полигонов и построения правильной формы полигональной сетки для последующего рига (оснастки модели скелетом), чтобы в процессе анимации модель персонажа деформировалась без искажений. Кроме того, использование уже запеченных текстур и карт нормалей, существенно снижает время на обработку модели при анимации и в интерактивном режиме.

Поскольку 3D-модель персонажа предназначена для применения в интерактивных сценах, при ретопологии установлены рамки максимального числа в 20 тыс. полигонов. При таком количестве полигонов анимированная 3D-модель будет использовать небольшое количество системных ресурсов с сохранением высокой степени реалистичности внешнего вида и поведения модели при визуализации.

Библиографический список

1. Основы разработки анимированных 3D-персонажей для интерактивных приложений: учеб. пособие / А. В. Никитин, Н. Н. Решетникова, С. И. Собашников, Д. С. Потехин. СПб.: ГУАП, 2019. 111 с.

2. Компьютерная 3d-графика. За кулисами. URL: https://3dyuriki.com/2015/03/07/topologiya-retopologiya-mesh-setka-3d-slovar-spravochnik/ (дата обращения: 06.12.2021).

3. Look in AR. URL: https://lookinar.com/ru/education-ru/chto-takoe-retopologyya/ (дата обращения: 06.12.2021).

4. Blender: офиц. сайт URL: https://docs.blender.org/manual/en/dev/sculpt_paint/html (дата обращения: 06.12.2021).

5. *Филенко Р.* UV-развертка. URL: https://blender.filenko.ru/visualization/uv-map.html (дата обращения: 06.12.2021).

6. Секреты экспорта из Blender в Unity. URL: https://habr.com/ru/post/254937/ (дата обращения: 06.12.2021).

УДК 621.396.967 **DOI: 10.31799/978-5-8088-1701-2-2022-2-155-158**

В. И. Афанасьева*
студент
В. А. Ненашев*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИССЛЕДОВАНИЕ АЛГОРИТМА ОБНАРУЖЕНИЯ ДВИЖУЩИХСЯ ОБЪЕКТОВ В ВИДЕОПОТОКЕ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ОЦУ

Реализуется метод обнаружения и классификации движущихся объектов по кадрам видеопоследовательности, сформированных статичной оптико-электронной системой наблюдения. Использованы метод цифровой обработки изображений и метод обнаружения связных областей. Результатом работы является классифицированный набор кадров видеопотока, классифицированный по следующим видам движущихся объектов (с подсчетом количества объектов): люди, автомобили.
Ключевые слова: *обнаружение объектов, подсчет движущихся объектов, бинаризация изображений, метод Оцу.*

V. I. Afanas'eva*
Student
V. A. Nenashev*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

INVESTIGATION OF THE ALGORITHM FOR DETECTING MOVING OBJECTS IN A VIDEO STREAM USING OTSU'S METHOD

In this paper, a method for detecting and classifying moving objects based on the captured frames of a video stream formed by a static optical-electronic surveillance system is implemented. The method of digital image processing and the method of detecting connected areas were used. The result of the work is a classified set of video stream frames, classified by the following types of moving objects: people, cars.
Keywords: *object detection, counting of moving objects, binarization of images, Otsu's method.*

Введение

Обнаружение объектов по кадрам видеопотока, сформированных статичной оптико-электронной системой наблюдения, стало в последнее время одной из важнейших задач анализа сцен в приложениях автоматизированного мониторинга за движущимися объектами. Обработка кадров видеопоследовательности с помощью приложений автоматизированного мониторинга позволяет выделить на обрабатываемой сцене необходимые объекты, классифицировать их и подсчитать.

Обнаружение и подсчет движущихся объектов применяются в различных сферах [1–12], в частности в автономных системах управления воздушным транспортным средством, в том числе беспилотным, а также в продвинутых способах взаимодействия человека с компьютером, навигации бортовых беспилотных систем и т. д. Следовательно, можно сделать вывод об актуальности темы данной работы.

Задача данного исследования – разработка метода автоматизированного обнаружения и выделения движущихся объектов с последующей их классификацией по выбранным признакам.

Метод межкадровой разности

Обнаружение движущихся объектов часто реализуется с применением разностного метода. Он основывается на вычитании из текущего кадра видеопоследовательности предыдущего кадра, получившаяся разница будет отражать движущиеся объекты. При обработке те места на кадре, которые относятся к движению объекта, отмечаются двоичной единицей, а остальные – как двоичные нули. В результате выделяются пиксели изображения движущегося объ-

а) б) в)

Рис. 1. Результат бинаризации изображения с помощью метода Оцу: а – исходный кадр; б – кадр в оттенках серого; в – бинаризированный кадр

екта. По полученному изображению можно определить местонахождение и параметры движения объекта. Но для того чтобы использовать данный метод, кадры видеопотока должны быть бинаризованы.

Бинаризация кадров видеопотока с помощью метода Оцу

Задача бинаризации изображения заключается в установлении наличия на изображении объектов, обладающих некоторыми характеристиками, например яркостью. Один из наиболее простых и естественных способов обнаружения объекта (или объектов) – выбор порога по яркости, или пороговая классификация. Смысл такого порога заключается в том, чтобы разделить изображение на светлый объект (foreground) и темный фон (background). То есть объект – это совокупность тех пикселей, яркость которых превышает порог ($I > T$), а фон – совокупность остальных пикселей, яркость которых ниже порога ($I < T$). Существуют десятки методов выбор порога. Быстротой и эффективностью отличается метод, придуманный японским ученым Нобуюки Оцу [1].

Применим метод Оцу на практике. Пусть имеется кадр видео, для которого требуется вычислить порог и бинаризовать его. Результат бинаризации изображения с помощью данного метода в среде MATLAB показан на рис. 1.

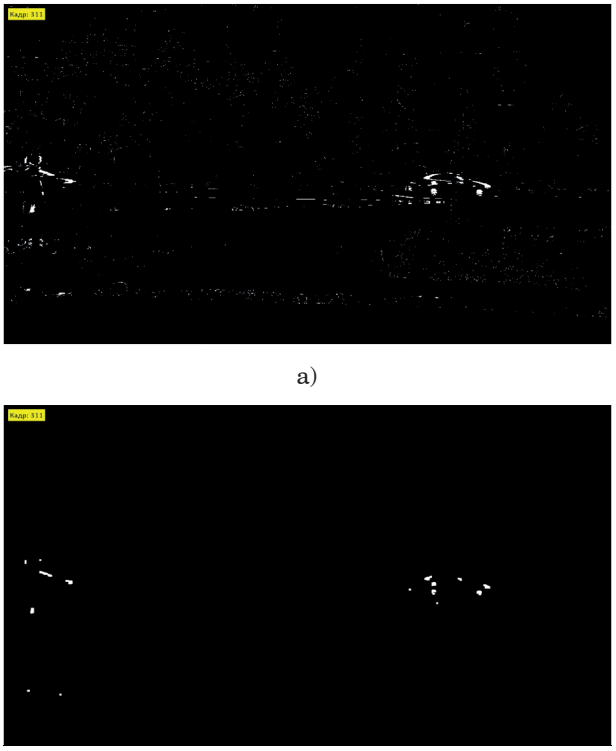
Эксперимент по обнаружению движущихся объектов в видеопотоке с использованием метода Оцу

На рис. 1 получено бинарное изображение одного кадра. Необходимо получить маску движущихся объектов на видеопоследовательность, для этого нужно применить данное преобразование к каждому кадру и воспользоваться методом межкадровой разности.

На рис. 2 можно увидеть результат обработки кадра видеопоследовательности, переведенного в бинарный вид с помощью метода Оцу с лишним шумом и без него. Для удаления лишнего шума с кадра был использован метод морфологического открытия изображения, с помощью него убраны нежелательные пиксели заданного размера и формы.

Затем, используя маску, можно выделить в рамку объекты, занимающие некоторое количество пикселей на кадре (рис. 3).

Параметры объекта, который требуется выделить, настраиваются с помощью функции задания связных областей. Таких параметров может быть множество, если нам нужно классифицировать несколько типов объектов.



а)

б)

Рис. 2. Результат обработки видео: а – маска с шумом; б – маска без шума

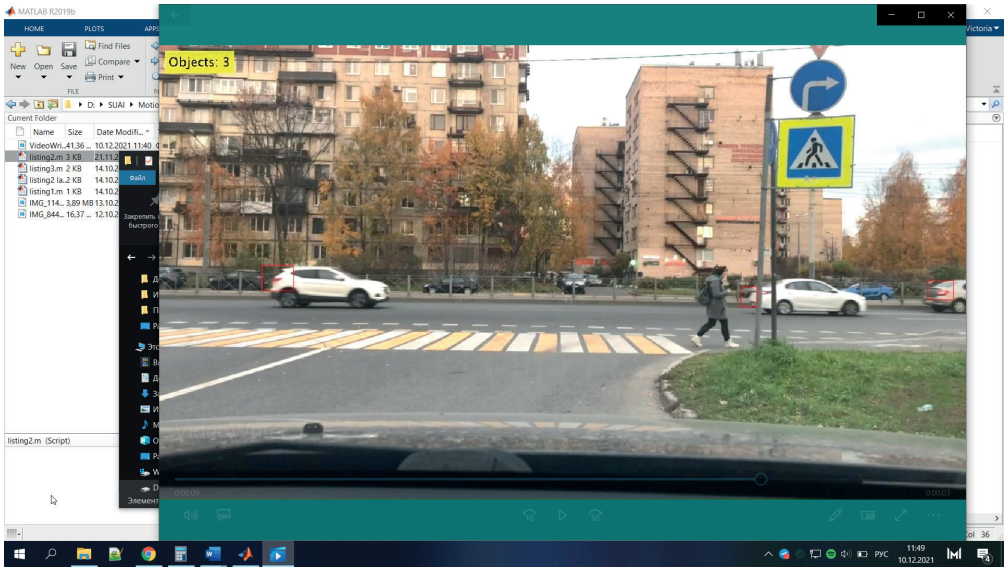


Рис. 3. Выделенные объекты

Заключение

В результате исследования рассмотрен и реализован метод обнаружения движущихся объектов в видеопоследовательности, сформированной статичной оптико-электронной системой наблюдения с использованием метода Оцу.

Был проведен эксперимент по обнаружению автомобилей путем использования реальных видеоданных, полученных с оптико-электронной системы наблюдения. Таким образом, была показана возможность обнаружения объектов на основе использования разностного метода обнаружения объектов с использованием метода Оцу.

Библиографический список

1. Обнаружение объектов методом Оцу. URL: https://habr.com/ru/post/112079/ (дата обращения: 12.11.2021).

2. *Nenashev V. A., Shepeta A. P., Kryachko A. F.* Fusion radar and optical information in multiposition on-board location systems // 2020 Wave Electronics and its Application in Information and Telecommunication Systems, WECONF 2020, St. Petersburg, 01–05 июня 2020 г. St. Petersburg, 2020. P. 9131451.

3. *Шепета А. П., Ненашев В. А., Григорьев Е. К., Ненашев С. А.* Обнаружение и оценка скорости подвижных объектов в оптико-локационных системах // Научная сессия ГУАП: сб. докл. науч. сессии, посвящ. Всемир. дню авиации и космонавтики: в 3 ч., СПб., 08–12 апр. 2019 г. Ч. II. СПб.: ГУАП, 2019. С. 282–287.

4. *Ненашев В. А., Сергеев А. М., Васильев И. А.* Моделирование сложных кодо-модулированных сигналов для современных систем обнаружения и передачи информации // Научная сессия ГУАП: сб. докл. науч. сессии, посвящ. Всемир. дню авиации и космонавтики: в 3 ч., СПб., 08–12 апр. 2019 г. Ч. II. СПб.: ГУАП, 2019. С. 413–417.

5. *Махлин А. М., Ненашев В. А., Шепета А. П.* Сравнительные характеристики квазиоптимальных цифровых обнаружителей сверхширокополосных сигналов // Волновая электроника и ее применения в информационных и телекоммуникационных системах: XXI Междунар. молодеж. конф., СПб., 01–05 окт. 2018 г. СПб.: ГУАП, 2018. С. 257–264.

6. *Shepeta A. P., Nenashev V. A.* Accuracy characteristics of object location in a two-position system of small onboard radars // Information and Control Systems. 2020. № 2 (105). P. 31–36.

7. *Sorokin A. B., Shepeta A. P., Nenashev V. A., Wattimena G. M.* Comparative characteristics of anti-collision processing of radio signal from identification tags on surface acoustic waves // Information and Control Systems. 2019. № 1 (98). P. 48–56.

8. Патент № 2703996 C2 Российская Федерация, МПК G01S 13/90. Способ локации целей в передних зонах обзора бортовых радиолокационных станций двухпозиционной радиолокационной системы: № 2019108828: заявл. 26.03.2019: опубл. 23.10.2019 / Г. А. Коржавин, В. А. Ненашев, А. П. Шепета [и др.]; заявитель АО «Концерн „Гранит-Электрон“».

9. Свидетельство о государственной регистрации программы для ЭВМ № 2016618938 Российская Федерация. Моделирование способа сжатия ФМ сигнала при влиянии активной помехи для решения задач помехоустойчивости: № 2016616140: заявл.

14.06.2016: опубл. 10.08.2016 / А. П. Шепета, В. А. Ненашев, И. А. Юдин, А. Ю. Каплин; заявитель ГУАП.

10. Свидетельство о государственной регистрации программы для ЭВМ № 2018661851 Российская Федерация. Программа для расчета взаимного расположения двухпозиционной РЛС и наблюдаемых объектов в полярной и декартовой системах координат: № 2018614572: заявл. 08.05.2018: опубл. 20.09.2018 / В. А. Ненашев, А. П. Шепета, Е. К. Григорьев [и др.]; заявитель ГУАП.

11. *Grigoriev E. K., Nenashev V. A., Sergeev A. M., Nenashev S. A.* Research and analysis of methods for generating and processing new code structures for the problems of detection, synchronization and noise-resistant coding // Proceedings of SPIE – The International Society for Optical Engineering: 26, Virtual, Online, 21–25 сент. 2020 г. Virtual, Online, 2020. P. 115331.

12. Свидетельство о государственной регистрации программы для ЭВМ № 2019612775 Российская Федерация. Программа вычисления структурированных квазиортогональных матриц Мерсенна: № 2019611351: заявл. 14.02.2019: опубл. 27.02.2019 / А. А. Востриков, А. М. Сергеев, Д. В. Куртяник [и др.]; заявитель ГУАП.

УДК 681.3.06: 519.68 **DOI: 10.31799/978-5-8088-1701-2-2022-2-159-162**

А. В. Гордеев
доктор технических наук, профессор
Санкт-Петербургский государственный университет аэрокосмического приборостроения

PVM, MPI И MOSIX КАК ТЕХНОЛОГИИ И СРЕДСТВА ОРГАНИЗАЦИИ ПАРАЛЛЕЛЬНЫХ РАСПРЕДЕЛЕННЫХ ВЫЧИСЛЕНИЙ

Кратко описываются наиболее известные технологии обмена сообщениями между параллельными вычислительными процессами, выполняющимися на разных узлах в распределенной вычислительной системе, которые позволяют организовать как параллельное выполнение какой-нибудь параллельной программы, так и параллельное выполнение множества таких программ. PVM и MPI используются при создании параллельных распределенных программ. MOSIX позволяет выполнять множество параллельных программ и осуществлять балансировку нагрузки на узлы кластера.
Ключевые слова: *распределенные системы, параллельные вычисления, кластерные системы, обмен сообщениями.*

A. V. Gordeev
Dr. Sc., Tech., Professor
St. Petersburg State University of Aerospace Instrumentation

PVM, MPI AND MOSIX AS TECHNOLOGIES AND TOOLS FOR ORGANIZING PARALLEL DISTRIBUTED COMPUTING

The most well-known technologies for exchanging messages between parallel computing processes running on different nodes in a distributed computing system are briefly described. Thay allow both to organize the parallel execution of a parallel program and the parallel execution of a set of such programs. PVM and MPI are used to create parallel distributed programs. MOSIX allows many parallel programs to run in parallel and to balance the load on the cluster nodes.
Keywords: *distributed systems, parallel computing, cluster systems, messaging.*

Одна из первых, наиболее известных (по крайней мере в нашей стране) и фундаментальных монографий, посвященных распределенным вычислительным системам, принадлежит перу профессора Э. Таненбаума [1]. В издательстве «Питер» в серии «Классика Computer Science» эта книга в переводе вышла в 2003 г. В ней описывается несколько технологий организации взаимодействия между параллельными процессами, которые могут выполняться в распределенных системах и есть краткое изложение основных идей интерфейса передачи сообщений (Message Passing Interface – MPI). Но такие технологии, как PVM и MOSIX, в этой работе не рассмотрены, хотя появились они достаточно давно.

Организовать эффективные синхронные параллельные вычисления крайне сложно, так как для этого необходимо специальное аппаратное обеспечение, поэтому преимущественно используются механизмы для асинхронного взаимодействия параллельных процессов и/или потоков, которые предоставляют существенно больше возможностей. И одним из основных механизмов для реализации взаимодействия между вычислительными процессами являются сообщения, которыми они могут обмениваться. Механизмы обмена сообщениями не очень трудно реализовать, они не требуют больших ресурсов и обладают достаточно большим быстродействием. Однако для того чтобы один процесс мог запустить другой процесс на другом компьютере, необходимо переслать соответствующий запрос к другой операционной системе. Для этого операционные системы компьютеров, объединенных в вычислительный кластер, должны поддерживать механизмы удаленного запуска программ на указанном компьютере. Причем эти механизмы должны обеспечивать полный контроль по их использованию и не позволять любому процессу самостоятельно и произвольно потреблять чужие ресурсы и влиять на чужие процессы. Другими словами, нужно иметь строгую авторизацию

для таких операций. Таким образом, это несколько сложнее, чем послать запрос на запуск процесса в той же операционной системе и, само собой, выполняется дольше из-за сетевых коммуникаций.

UNIX-подобные операционные системы, которые часто называют POSIX-системами, обладают всеми механизмами, с помощью которых можно организовать выполнение некоторого множества параллельных взаимодействующих процессов. Такие параллельные взаимодействующие вычисления легко могут быть запущены из одной программы, для этого имеется несколько механизмов [2]. Среди них на самом низком уровне находятся удаленный вызов процедур и сокеты. А поскольку с использованием упомянутых механизмов достаточно сложно создавать параллельные взаимодействующие программы, которые могли бы выполняться в распределенных вычислительных системах, то на их основе были разработаны специальные технологии и созданы более высокоуровневые средства.

Важно заметить, что запуск процесса или пересылка ему сообщений могут быть сделаны как на том же компьютере, так и на другом, но связанном с первым посредством сетевых технологий. В случае создания кластерной вычислительной системы компьютеры называют узлами. Со временем узлы кластера стали многопроцессорными. В общем случае запущенный процесс будет выполняться либо на другом процессоре того же узла (если он есть и это возможно, так как он в настоящий момент свободным или может быть освобожден), либо на другом узле. Для унификации можно использовать одни и те же вызовы функций, которые будут работать как в мультипроцессорных, так и в многомашинных системах, хотя и с разной эффективностью.

Наиболее известными первыми средствами организации параллельных распределенных вычислений, скорее всего, стали те, что были созданы в рамках технологии параллельных виртуальных машин [3]. Пакет PVM (Parallel Virtual Machine) устанавливается на операционные системы тех компьютеров, которые входят в кластерную вычислительную систему, и обеспечивает возможность использовать технологию передачи сообщений между вычислительными процессами, которые выполняются параллельно на узлах кластера. Эта технология была разработана в 1989 г. [4], и она поддерживает программирование на языках Fortran, C и C++ посредством использования специальных библиотек. Функции библиотеки позволяют достаточно динамично влиять на параллельное выполнение программы на множестве реально имеющихся узлов, поскольку дают возможность узнать их состав и количество и отправить вычисления на реально существующий узел, а не на некий гипотетический. PVM можно использовать не только в гомогенном кластере, но и в гетерогенном. Правда, это приводит к большим трудозатратам и тщательному планированию вычислений для получения реальных ускорений, поскольку необходимо заранее представлять, какая часть вычислений будет выполняться на каком компьютере, и иметь для него соответствующий двоичный код. Другими словами, PVM – это не просто средство организации параллельных вычислений, а средство создания распределенных вычислительных систем, в том числе гетерогенных. Главным механизмом в нем выступают сообщения, которые позволяют синхронизировать вычислительные процессы и передавать данные между ними. Важно отметить, что PVM поддерживает особые обмены сообщениями, которые могут осуществляться в мультипроцессорных узлах между процессами, которые выполняются в пределах узла и при этом не затрагивают межузловые коммуникации. А для обмена сообщениями между узлами можно использовать другие вызовы.

Через несколько лет после разработки PVM в 1994 г. появилась технология, названная MPI. Судя по всему, программные средства, реализующие эту технологию, создавались без оглядки на PVM. При этом технология MPI получила достаточно хорошие известность и распространение. Имеются как проприетарные, так и открытые реализации, но главное – это большое количество параллельных распределенных программ в разных предметных областях, которые созданы на базе этой технологии. При этом последние версии таких средств поддерживают как технологии обмена сообщениями, так и многопоточность, которую имеет смысл применять в случае мультипроцессорных узлов и запуска параллельных процессов на таких узлах.

В настоящее время можно легко воспользоваться открытыми пакетами MPI, среди которых хорошо известны MPICH. Имеется много публикаций на тему использования технологии MPI и соответствующих программных средств, в том числе учебных [5]. Технология MPI и библиотека MPICH настолько популярны, что их даже портировали на платформу «Эльбрус» и они входят в состав дистрибутива штатной операционной системы. Однако в случае разработки параллельных распределенных приложений

не стоит забывать, что при распараллеливании вычислений можно и не получить ожидаемых ускорений либо получать очень незначительные ускорения [6, 7].

Еще один подход к созданию эффективных параллельных приложений заключается в том, что решение о том, на каком процессоре или узле стоит запускать новый поток или процесс, можно принимать не заранее (перед компиляцией программы), а динамически, непосредственно по ходу выполнения программы. В качестве примера такого подхода можно назвать публикацию [8], причем по этой тематике имеется достаточно большое количество работ, выполненных под руководством проф. В. А. Васенина.

PVM и MPI являются средствами для создания параллельных программ, которые могут выполняться в распределенной вычислительной системе. Но каждая операционная система каждого узла работает в целом независимо друг от друга, хотя и помогает взаимодействию параллельных взаимодействующих процессов за счет пересылки сообщений между узлами. При проектировании приложений для их выполнения на конкретном вычислительном кластере можно добиться приемлемых ускорений, но в действительности ожидаемые результаты можно гарантировать только в случае выполнения только одной такой параллельной распределенной программы. Если же на таком кластере запустить несколько параллельных распределенных программ, то они начнут конфликтовать из-за ресурсов и теоретически возможные ускорения окажутся недостижимыми. Другими словами, в кластерных системах мы, как правило, имеем дело с множеством независимых операционных систем, каждая из которых отвечает только за свои ресурсы, но имеет средства для передачи заданий и/или сообщений на другую операционную систему с целью организовать выполнение распределенных параллельных программ. При этом Э. Таненбаум говорил о необходимости иметь не только мультипроцессорные операционные системы, а многомашинные, т. е. распределенные операционные системы, которые могли бы учитывать все имеющиеся ресурсы и осуществлять такую диспетчеризацию задач, чтобы можно было получить максимальную загрузку ресурсов. Помимо этого, желательно иметь методы и средства перераспределения ресурсов между задачами с целью обеспечить учет приоритетов, что в свою очередь даст возможность для тех вычислений, которым необходимы высокие значения ускорения, максимально распараллелиться.

Наиболее интересной представляется технология MOSIX, первая разработка которой состоялась в 1997 г., т. е. она появилась позже PVM и MPI. Она была разработана под руководством профессора А. Барака в Hebrew University (Иерусалим, Израиль). Самое важное в этой технологии, что она не имеет отношения к разработке параллельных распределенных программ (в отличие от технологий PVM и MPI), но зато позволяет создать и получить виртуальную POSIX-систему, которая будет работать в вычислительном кластере и обеспечит выполнение параллельных программ. В обычной операционной POSIX-системе мы имеем общую разделяемую оперативную память, в которой выполняется множество вычислительных процессов и потоков. В обычной кластерной системе мы имеем множество взаимодействующих POSIX-систем с выполняющимися в них параллельными процессами и потоками, при этом эти системы обеспечивают параллельные процессы механизмами обмена сообщениями. А общей оперативной памяти в таких системах нет, что приводит к существенному замедлению вычислений и трудностям к оперативному управлению всеми вычислениями. Технология MOSIX создает такую виртуальную память, как бы объединяет основные ресурсы узлов вычислительного кластера, что позволяет операционным системам, которые управляют каждая своим узлом, за счет обмена сообщениями выступать по отношению к выполняющимся вычислениям как единая распределенная операционная система. Именно поэтому ее часто называют Cluster Management System (CLM), т. е. система управления кластером. Перераспределение ресурсов достигается в том числе миграцией уже выполняющихся вычислений с одного узла на другой, менее загруженный.

В кластерных системах имеются методы и средства балансировки нагрузки. Однако, как правило, эти средства ориентированы на распределение нагрузки в рамках конкретного сервиса, который кластер предоставляет своим клиентам. Даже в системах Microsoft Windows имеются такие средства, и в качестве примера можно назвать Network Load Balancing (NLB). Принципиальное различие между NLB и MOSIX в том, что последняя может обеспечить балансировку нагрузки с учетом всех вычислительных процессов, которые выполняются в этой системе. Отметим, что среди процессов могут быть как простые последовательные, так и параллельные распределенные программы, причем их может выполняться сразу достаточно большое количество.

Существует несколько проектов под общим названием MOSIX, как проприетарных, так и открытых [9]. Поскольку имеются исходные коды библиотеки, которые после установки на узлы кластера делают из него распределенную мультипроцессорную систему с виртуальной общей памятью, то эту технологию можно перенести в том числе на отечественные вычислительные системы «Эльбрус» [10]. На компьютерах «Эльбрус» можно создать достаточно мощные кластерные мультипроцессорные вычислительные системы. Однако получить увеличение скорости вычислений, осуществляемых в таком кластере, скорее всего, не получится. Эта технология и средства, созданные на ее основе, прежде всего ориентированы на увеличение нагрузки на процессоры и узлы кластера, а не на эффективное распараллеливание, которое может привести к желаемым ускорениям. Поэтому считаю целесообразным заняться не только портированием и последующим исследованием MOSIX на платформу «Эльбрус», но и портированием PVM на эту же платформу. Технология PVM позволит создавать информационно-вычислительные системы, которые смогут работать в гетерогенной вычислительной среде, в которой будут присутствовать как классические x86-64 компьютеры, так и «Эльбрус»-системы. Наличие первых позволит наиболее эффективно выполнять программные модули, созданные именно для первой платформы, в то время как платформа «Эльбрус» может обеспечить выполнение кода, который создан для этой архитектуры и выполняется на ней более быстро. Это актуально, поскольку, как показывает практика, выполнение на компьютерах «Эльбрус» программ, созданных для компьютеров с архитектурой x86-64, хоть и возможно в режиме двоичной трансляции, но происходит гораздо медленнее из-за эмуляции этой архитектуры. Возможно портирование программного обеспечения из исходных кодов, но это трудоемко и иногда не приводит к эффективному коду для «Эльбрус». Поэтому разумным представляется часть алгоритмов готовить в виде нативного для «Эльбрус» кода, а ту часть программного обеспечения больших и сложных информационных систем, которая теряет в скорости своего выполнения на компьютерах «Эльбрус», оставить без изменения и выполнять на компьютерах с архитектурой x86-64. Технология и средства PVM это позволяют осуществлять.

Библиографический список

1. *Таненбаум Э.* Распределенные системы. Принципы и парадигмы. М.: Ван Стеен; СПб.: Питер, 2003. 877 с.

2. *Лацис А.* Как построить и использовать суперкомпьютер. М.: Бестселлер, 2003. 240 с.

3. PVM-параллельная виртуальная машина. URL: https://docstore.mik.ua/manuals/ru/linux_parallel/node209.html (дата обращения: 25.11.1921).

4. Parallel Virtual Machine // Википедия. URL: https://ru.wikipedia.org/wiki/Parallel_Virtual_Machine (дата обращения: 25.11.1921).

5. *Воеводин Вл. В., Жуматий С. А.* Вычислительное дело и кластерные системы. М.: Изд-во МГУ, 2007. 150 с.

6. *Гордеев А. В., Горелик Д. В.* Об ускорении распределенных вычислений в мультипроцессорных кластерных системах // Научная сессия ГУАП: сб. докл.: в 4 ч. Ч. 2. Технические науки. СПб.: ГУАП., 2019. С. 391–395.

7. *Гордеев А. В., Горелик Д. В.* Уточнение закона Густавсона для мультипроцессорных кластерных систем // Technical science. Maguar Tudomanuos Jornal. 2020. № 42. P. 46–52.

8. *Степанов Е. А.* Методы и средства планирования вычислений в системах автоматизированного динамического распараллеливания программ: автореф. дис. ... канд. физ.-мат. наук. М., 2008. 25 с.

9. *Лубанец А.* MOSIX-кластер в СПбГТУ. URL: https://www.opennet.ru/docs/RUS/mosix_spb/index.html (дата обращения: 25.11.1921).

10. *Горелик Д. В.* MOSIX как система динамического распараллеливания вычислений для «Эльбрус» // Обработка, передача и защита информации в компьютерных системах: Междунар. науч. конф. (СПб., 14–22 апр. 2021 г.): сб. докл. СПб.: ГУАП, 2021. С. 97–99.

УДК 681.3.06:519.68 **DOI: 10.31799/978-5-8088-1701-2-2022-2-163-165**

Д. В. Горелик
аспирант
Санкт-Петербургский государственный университет аэрокосмического приборостроения

LORA – ИНТЕГРАЦИЯ В ПОВСЕДНЕВНУЮ ЖИЗНЬ

Система, построенная на технологии LoRa, предоставляет возможность наблюдения и управления объектами, соответствующими реальным объектам слежения, а не отдельным устройствам. В состав объекта может входить произвольное количество устройств, при этом каждое из них может использоваться несколькими объектами. Это позволит получить систему с динамической миграцией сенсоров между объектами и повышает эффективность наблюдения.
Ключевые слова: *сенсоры, шлюзы, объекты, LoRa, LoraWAN, мониторинг.*

D. V. Gorelik
PhD Student
St. Petersburg State University of Aerospace Instrumentation

LORA – INTEGRATION INTO EVERYDAY LIFE

The system based on LoRa technology provides the ability to monitor and control objects that correspond to real tracking objects, and not to individual devices. An object can include an arbitrary number of devices, and each device can be used by several objects. This will provide a system with dynamic migration of sensors between objects and increase the efficiency of observation.
Keywords: *sensors, gateways, objects, LoRa, LoraWAN, monitoring.*

В нашем мире очень быстро набирают популярность IoT-устройства (Internet of Things «Интернет вещей»). Это устройства автоматизации простых процессов, для работы которых требуется выход в Интернет, например датчики протечки, температуры в помещении, открывания/закрывания дверей, движения, умный чайник и пр. И у нас много готовых решений, в частности большое разнообразие стандартов: Wi-Fi, 3g-, 4g-, 5g-сотовые сети.

Вообразим, что у нас есть большое офисное здание в 20 этажей, в котором более ста офисных помещений и работает более тысячи сотрудников. На каждом этаже существует по 6 туалетов, а также по 30 межофисных дверей. Каждый туалет оборудован современным счетчиком, способным передавать показания на сервер. Итого: 240 датчиков протечки (чтобы в туалетах не было потопов), 240 счетчиков воды, 600 датчиков открывания/закрывания дверей. Общее количество устройств около 1080+, и это мы не берем дополнительные. Предположим, в подвале здания многоуровневый паркинг на 500 автомобилей, это еще 500 сенсоров. Итого 1580. Пусть у этих устройств будет очень маленький траффик, но если мы говорим про мобильную сеть, то сотовую вышку они займут полностью, и у людей не будет мобильной связи. Мы рассматриваем одно офисное здание, а если их не одно, а несколько… Они стоят рядом, плюс жилые дома поблизости, получается, мы лишаем микрорайон сотовой связи.

Теперь рассмотрим энергопотребление. Если мы говорим про сотовую или беспроводную связь от роутера, конечному устройству необходимо большое количество электроэнергии, даже хороший элемент питания разрядится за несколько дней, а теперь представьте расход элементов питания на одно здание. Немыслимо большие траты с финансовой точки зрения и с точки зрения человеко-часов на замену этих батареек. Значит, нужна высокая энергоэффективность.

Для передачи информации от конечного устройства (сенсора) нам не нужен высокоскоростной канал связи. Приоритет в том, чтобы при передаче мы потратили минимальное количество энергии. Необходимо и достаточно, чтобы передача данных шла с минимальной скоростью и минимальной мощностью, но сигнал гарантированно проходил необходимое расстояние.

LoRA (Long Range) – это технология модуляции сигнала [1]. Она дает возможность значительно увеличить дальность связи. Сам прото-

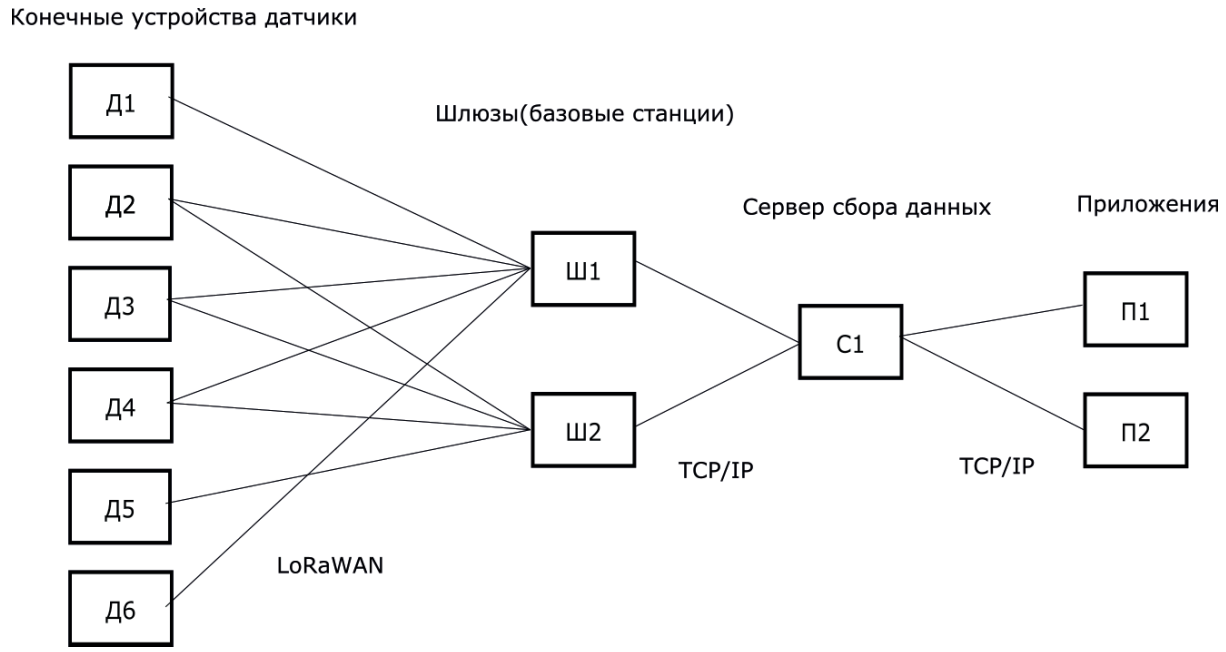


Рис. 1. Модель работы технологии LoRa: Д – датчики, конечные устройства; Ш – шлюзы; С – сервер; П – приложения

кол передачи данных носит название LoRa WAN и является открытым. Подробнее модель работы можно рассмотреть на рис. 1.

Разберемся, как же это работает. У нас есть шлюз, он же роутер, у него есть заданный диапазон частот, в котором он слушает конечные устройства. В момент когда от конечного устройства приходит кадр (запрос), станция отвечает на частоте вещания устройства. В таких условиях канал получается 125 кГц, скорость передачи данных доходит до 5 кбит в секунду [2].

Если информацию от конечных устройств принимает шлюз, то сбором и хранением полученных данных занимается сетевой сервер. Он может управлять шлюзами, сообщать датчику, с каким конкретно шлюзом вести обмен данными, если сенсор находится в зоне действия нескольких шлюзов. Может собирать и показывать статистику передачи данных. По большому счету, сервер – это «сборщик» данных с конечных устройств [3].

Сетевой сервер – это не последнее звено, ведь полученную информацию нужно обработать. Этим занимается сервер приложений. На самом деле это может быть множество приложений, работающих на разных серверах, – сервисы. Если первые ступени этой системы хорошо развиты, то работа с конечным пользователем далека от идеала [4].

Опишем конкретную реализацию конечного сервиса. Большая часть систем мониторинга и непромышленной автоматизации (различные варианты умных зданий, парковок и офисов, домов) неоднородны, часто в одной системе работают устройства не только разных производителей, но и использующие разные протоколы. Например, для каких-то задач могут использоваться LoRa-датчики, для каких-то – BLE (Bluetooth-метки, недавно на презентации компании Apple были представлены устройства, работающие по этой технологии). Отсюда вытекает задача приведения разнородных данных к единому формату, пригодному для дальнейшей обработки.

Вторая проблема – конечными пользователями подобных систем являются не технические специалисты, а обслуживающий персонал здания: охранники, администраторы парковок, уборщики и т. д., которых волнует состояние объектов наблюдения, а не установленных на них датчиков. То есть возникает задача представления данных не в контексте устройств, а в контексте объектов управления и мониторинга, с которыми ведет работы конкретный пользователь (например, пользователя интересует состояние парковочного места, а не датчика или датчиков, если их несколько).

При разработке такой системы мы предлагаем решение обеих задач. В первую очередь был определен внутренний стандарт хранения данных, основанный на JavaScript-формате обмена данными. Мониторинг работает на СУБД MongoDB, но в нее сохраняются записи в определенных форматах, принятых в системах хранения и передачи измерений. За преобразование входных данных к внутреннему формату

отвечают драйверы – специальные сервисы, которые разрабатываются специально под каждую поддерживаемую модель датчика. Драйвер интерпретирует приходящие к нему данные и изменяет в соответствии с ними информацию об устройстве в базе данных.

Решение второй проблемы осуществляется при помощи специального приложения Monitoring (Мониторинг). Это картографическое веб-приложение, позволяющее следить за состоянием специальных абстрактных элементов – объектов. Объект является представлением некого объекта реального мира и обладает такими атрибутами, как имя, уникальный номер и координаты, набором атрибутов, отображающих его состояние, а также иконкой и цветом, которые могут меняться в зависимости от состояния. Объект может включать как одно, так и несколько устройств, и система будет изменять его состояние в зависимости от состояний этих устройств и дополнительных настроек.

В интерфейсе объект имеет два представления: он отображается на карте и в списке, при выборе объекта (не важно где) открывается карточка с подробной информацией: свойствами самого объекта, его фотографией, описанием. В отдельных вкладках доступны для просмотра список устройств, привязанных к объекту, и протокол (log) его работы.

Объект обладает следующими важными характеристиками.

1. Ему можно добавлять новые свойства, используя в качестве источников данных для них добавленные в объект устройства. Это позволяет настроить объект таким образом, чтобы следить (или управлять) значениями наиболее важных свойств, абстрагируясь от уровня датчиков. Это удобно для конечных пользователей, поскольку такой пользователь сможет работать с понятными ему понятиями, например открывать объект «холодильник» и сразу видеть, какая в нем температура, горят ли лампочки, стабильное ли питание и т. д., даже если за получение всех этих данных отвечает набор из разнотипных устройств.

2. Для объектов в Мониторинге реализована система критических событий: для каждого свойства каждого объекта можно задать условие, при выполнении которого система будет формировать уведомление и, опционально, отправлять его по почте и/или используя СМС и мессенджеры.

3. Мониторинг формирует историю и журналы, которые доступны для загрузки в формате CSV (Comma-Separate Values), что позволяет анализировать изменение объектов наблюдения во времени.

4. Поддержка трекинга: в случае если в состав объекта входит устройство с GPS-датчиком, Мониторинг может изменять положение этого объекта на карте в соответствии с данными геолокации и отображать историю его перемещений. Также поддерживается отслеживание вхождения или выхода объектов из определенных зон: администратор портала может задать границы, в случае пересечения которых тем или иным объектом будет отправлять соотвествующее уведомление.

Разрабатываемый сервис является многопользовательским, поддерживает систему разделения прав доступа, систему фильтрации объектов и инструменты для работы с картой. Поверх карты могут размещаться планы, сама карта настраиваться (например, может использоваться другой тип карты в зависимости от нужд пользователей). Технология LoRa позволяет упростить жизнь и автоматизировать множество процессов и систем.

Библиографический список

1. Что такое LoRaWAN. URL: https://habr.com/ru/company/nag/blog/371067/ (дата обращения: 27.11.2021).

2. Руководство разработчика устройств LoRaWAN. URL: https://lar.tech/images/pdf/lorawan-lartech.pdf (дата обращения: 10.11.2021).

3. LoRa Сети передачи данных на большие расстояния LoRa WAN. URL: https://www.gamma.spb.ru/media/pdf/masters2015/LORA.pdf (дата обращения: 29.10.2021).

4. *Гусев О.* Технология LoraWAN. URL: https://www.citycom.ru/demo/LoraWAN_07-12-17_gusev.pdf (дата обращения: 12.11.2021).

УДК 004.946 **DOI: 10.31799/978-5-8088-1701-2-2022-2-166-168**

Е. Е. Майн*
аспирант
А. В. Никитин*
кандидат технических наук, доцент
М. Б. Сергеев*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

АНАЛИЗ МОДЕЛЕЙ МУЛЬТИМОДАЛЬНОГО ИНТЕРФЕЙСА

Рассмотрены концептуальные модели мультимодального взаимодействия человека с цифровыми реальностями, их свойства и связи между ними для определения ведущей модальности.
Ключевые слова: *мультимодальное взаимодействие, цифровые реальности, концептуальные модели.*

E. E. Main*
Phd Student
A. V. Nikitin*
PhD, Tech., Associate Professor
M. B. Sergeev*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

ANALYSIS OF MULTIMODAL INTERFACE MODELS

The article discusses conceptual models of multimodal human interaction with digital realities, their properties and connections between them to determine the leading modality.
Keywords: *multimodal interaction, digital realities, conceptual models.*

В психологии «модальность» означает принадлежность отражаемого раздражителя к определенной сенсорной системе человека (слух, осязание, зрение и т. п.). В 1997 г. Боржегони предложил следующее определение данного термина: «модальность – это механизм кодирования информации для представления людям или машинам в физически реализованной форме» [1, 2].

Термин «модальность», тесно связан с понятиями «мультимедиа» и «мультимодальность». М. Вилсон различал мультимедийную систему как систему, которая просто использует разные средства представления, а мультимодальную систему считал особым типом мультимедийной системы, ориентированный на единый внутренний язык представления информации [2, 3].

В настоящее время человек может работать в различных реальностях: виртуальной и дополненной, дополненной виртуальности, виртуальных мирах, которые объединяются термином «цифровые реальности». В контексте цифровых реальностей «модальность» следует понимать как тип канала коммуникации, используемого для взаимодействия, а «мультимодальный» означает использование более чем одного канала коммуникации для взаимодействия.

В Институте интерактивных и цифровых медиа Национального университета Сингапура была разработана программа, связывающая низкоуровневые функции в разных носителях (изображение, музыка, текст) в семантическую информацию более высокого порядка с использованием социальной семиотической теории и компьютерные методы анализа. Однако сложность мультимодального анализа ограничила тип аналитических, а в связи с этим и теоретические разработки, которые были сделаны для динамических средств массовой информации (видео-, цифровые средства информации) [4].

В человеко-машинных системах мультимодальное взаимодействие означает взаимодействие пользователя с приложением при использовании более чем одного способа взаимодействия [5]. Чтобы описать в человеко-машинных системах мультимодальное взаимодействие, были введены такие понятия, как «некоторое

Таблица 1

Сравнение концептуальных моделей мультимодального интерфейса

Категория	TYCOON	CASE	CARE
Концептуальный подход	Описание и оценивание типов связи взаимодействия модальностей	Классификация связи модальностей по трем критериям	Описание и оценивание свойств взаимодействия модальностей
Количество типов взаимодействия	5	4	4
Теоретическое обоснование	Психология, искусственный интеллект [9, 12], человеко-машинное взаимодействие	Время	Искусственный интеллект, человеко-машинное взаимодействие, время, желание достичь какого-либо результата
Формализация в математическом виде	Нет	Частично представлена в модели CARE	Да

Таблица 2

Сравнение моделей по типам взаимодействия

Категория	TYCOON	CASE	CARE
Трансформация	Устанавливает, какая часть информации, порожденная одной модальностью, будет использоваться другой модальностью. Использование возможно осуществлять как между двумя входными или выходными модальностями, так и между входной и выходной модальностью [6, 8]	Слияние, уровень абстракции	–
Специализация	Указывает на определенный тип информации, который всегда обрабатывается одной и той же модальностью	Exclusive (особенность) – одна задача за другой использует одну модальность в одно и тоже время, без обращения друг к другу	Assignment (назначенный) – модальность m называют определенной для достижения состояния s' из состояния s, если ни одна из других модальностей не может быть использована для достижения состояния s' из состояния s. Другие модальности не эквивалентны, поэтому не могут достичь состояния s'
Эквивалентность	Две модальности считаются эквивалентными, если часть информации может быть обработана, как альтернатива любой из модальностей	–	Equivalence (равнозначность) – модальности набора M эквивалентны для достижения состояния s' из состояния s, если необходимо и достаточно использовать любую из модальностей. Определяется возможность выбора между модальностями без обременения в виде ограничений времени
Избыточность	Некоторые модальности взаимодействуют избыточно, когда они обрабатывают одну и ту же информацию	Concurrent (согласованность) – две различные задачи выполняются параллельно без обращения друг к другу. Alternate (очередность) – задачи с временным чередованием модальностей, используется взаимосвязь	Redundancy (чрезмерность) – модальности из набора M используются чрезмерно для достижения состояния s' из состояния s, если они равны (эквивалентны) и если все они используются в рамках одного временного окна tw. Может быть последовательной и параллельной
Комплементарность	Рассматривается несколько модальностей, каждая из которых обрабатывает различные части информации, которые впоследствии объединяются [12]	Synergistic (совмещенность) – параллельные задачи, использующие несколько связанных модальностей	Complementarity (взаимодополняемость) – модальности из набора M используются комплементарно для достижения состояния s' из состояния s в рамках одного окна, если все они должны быть использованы для достижения состояния s' из состояния s. Отдельно взятые модальности не могут быть использована для достижения состояния s'

физическое устройство» d и «язык взаимодействия» L, т. е. $<d, L>$ [6–8].

Рассмотрим концептуальные модели мультимодального взаимодействия человека с цифровыми реальностями [9–10], их свойств и связей между ними с целью определения ведущей модальности [11]:

– TYCOON;

– CASE;

– CARE.

В табл. 1 представлен обобщенный подход для сравнения некоторых концептуальных моделей.

В табл. 2 приведено сравнение свойств и связей концептуальных моделей по типам взаимодействия.

Такие свойства концептуальной модели TYCOON, как специализация, эквивалентность, избыточность и комплементарность, включены в следующие свойства модели CARE [13]: assignment (назначенный), equivalence (равнозначность), redundancy (чрезмерность), complementarity (взаимодополняемость) [14].

Свойства модели CASE отражаются в модели CARE следующим образом: exclusive (особенность) включена в assignment, concurrent (согласованность) и altenate (очередность) отражаются в redundancy, a synergistic (совмещенность) – complementarity.

Исходя из сказанного, ведущей концептуальной моделью является CARE.

Библиографический список

1. *Bordegoni M., Faconti G., Feiner S.* et al. A standard reference model for intelligent multimedia presentation systems // Computer standards and interfaces. 1997. P. 477–496.

2. Centre for Multimodal Communikation. URL: https://multimodalkeyterms.wordpress.com (дата обращения: 08.11.2021).

3. *Wilson M., Sedlock D., Binot J., Falzon P.* An architecture for multimodal dialogue // Proceedings Second Vencona Workshop for Multimodal Dialogue. Vencona, Italy. 1992.

4. *O'Halloran K. L., Podlasov A., Chua A., Tiss C.-L., Lim F. V.* Challenges and Solutions to Multimodal Analysis: Technology, Theory and Practice // Developing Systemic Functional Linguistics: Theory and Application / eds. Y. Fang and J. Webster. L.: Equinox, 2013.

5. Multimodal Interaction Working Group. URL: https://www.w3.org/2002/mmi/Overview/html (дата обращения: 10.11 2021).

6. *Wilson M. D., Falzon P.* Multimodal and multimedia systems: architectures for advanced dialogue. ERCIM Multimedia Workshop. Lisbon, 1991.

7. *Baldry A. P., Thibault P. J.* Multimodal Corpus Linguistics // System and Corpus: Exploring Connections / eds. G. Thompson, S. Hunston. L.: Equinox. 2006. P. 164–183.

8. Архитектура виртуальных миров: монография / А. Е. Войскунский, М. Б. Игнатьев, С. А. Козловский [и др.]; под ред. М. Б. Игнатьева, А. В. Никитина, А. Е. Войскунского. 2-е изд., перераб. и доп. СПб.: ГУАП, 2009. 288 с.

9. *Morie J. F., McCallum K.* Handbook of Research on the Global Impacts and Roles of Immersive Media. USA, 2019.

10. *Turk M.* Multimodal interaction: A review. Pattern Recognition Lett. 2013. URL: http://dx.doi.org/10.1016j.patrec.2013.07.003 (дата обращения: 01.12.2021).

11. *Dumas B., Lalannc D., Oviat S.* Multimodal Interfaces: A Survey Principles, Models and Frameworks if Human Machine Interaction // Lecture Notes in Computer Science. 2009. Vol. 5440. P. 3–26.

12. *Grifoni P.* Multimodal Human Computer Interaction and Pervasive Services. Italy, 2009.

13. *LaViola J. J. Jr., Buchanan S., Pittman C.* Multimodal Input for Perceptual User Interfaces. University of Central Florida First published, 2014.

14. Multimodal Interfaces and Sensory Fusion in VR for Social Interactions / eds. E. Bekelel., J. W. Wadel, D. Bian, L. Zhang, Z. Zheng, A. Swanson, M. Sarkar, Z. Warren, N. Sarkar, R. Shumaker, S. Lackey.VAMR, 2014. Part I.

УДК 519.614 **DOI: 10.31799/978-5-8088-1701-2-2022-2-169-173**

А. М. Сергеев*
кандидат технических наук, доцент
Ю. Н. Балонин*
инженер
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

МАЙНИНГ МАТРИЦ

Рассматривается процесс поиска матриц Адамара как майнинг, включающий не только задание начальных условий, выбор метода реализации, но и фильтрацию как «обогащение» набора последовательностей для построения матриц Адамара на примере конструкции Балонина – Себерри. Анализируются трудности майнинга матриц и рассматривается способ их преодоления за счет фиксации конструкций для поиска. Предлагается предварительная фильтрация сгенерированных последовательностей для ускорения получения результата.
Ключевые слова: *майнинг матриц, матрицы Адамара, конструкция Пропус, фильтрация последовательностей.*

Работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации, соглашение № FSRF-2020-0004.

A. M. Sergeev*
PhD, Tech., Associate Professor
Y. N. Balonin*
Engineer
**St. Petersburg State University of Aerospace Instrumentation*

MINIG OF MATRICTS

The process of searching for Hadamard matrices is considered as mining, which includes not only setting initial conditions, choosing an implementation method, but also filtering as «enrichment» of a set of sequencesfor constructing Hadamard matrices using the Balonin-Seberri construction as an example. The difficulties of mining of matrices are analyzed and a way to overcome them by fixing structures for search is considered. Preliminary filtering of the generated sequences is proposed to speed up the result.
Keywords: *mining of matrices, Hadamard matrices, Propus conctruction, filtering sequences.*

Введение

Английское слово «mining» имеет устоявшееся на протяжении долгого времени (до наступления цифровой эры) значение, соответствующее процессу «добычи» или «разработки» полезных ископаемых, связанное с большими трудозатратами. Сегодня оно неразрывно связано с деятельностью по созданию новых цифровых структур (блоков в блокчейне) для обеспечения функционирования криптовалютных платформ. Процесс этот тоже является трудоемким, требующим огромного количества компьютерных вычислений и энергозатрат.

Поиск матриц Адамара [1] характеризуется аналогичными трудностями и вычислительными затратами. Каждая новая матрица с учетом структуры и порядка, будучи результатом разработки специального алгоритма, длительных вычислений, проверки ортогональности, имеет не меньшую стоимость, выраженную в трудозатратах, чем биткоин или каменный уголь из шахты.

Цель настоящей работы – показать пути повышения эффективности процесса майнинга матриц Адамара, как и прочих похожих на нее матриц с малым числом значений элементов [2, 3].

Области использования матриц Адамара

Матрицы Адамара, являясь ортогональными (квазиортогональными), широко используются в системах передачи и хранения данных, для которых характерны симметричные ортогональные преобразования. Задачи, в которых применяются матрицы Адамара и подобные им, – обработка сигналов и изображений [4–6], помехо-

устойчивое кодирование изображений [7], получение кодов для защиты данных [8] и др.

Почему труден майнинг матриц Адамара?

Во-первых, не существует универсального метода, способного одинаково эффективно искать матрицы Адамара возможных структур на всех порядках 4**t**, на которых они существуют. Здесь **t** – натуральное число. Классическими уже стали методы Сильвестра, Пэли, Вильямсона, Скарпи и их многочисленные модификации. Однако их возможности не позволяют покрыть все возможные порядки матриц Адамара – имеется большое количество пропусков порядков, характерных для указанных методов.

Во-вторых, для широкого круга поисковиков не сложилась практика использования для «добычи» матриц мощных вычислительных средств. До сих пор отсутствуют общедоступные ресурсы суперкомпьютеров – они все еще не стали массовым инструментарием со свободным доступом. При таком положении важную роль играет эффективность разрабатываемых алгоритмов как результатов использования новых подходов и приемов программирования.

В-третьих, при бесконечности количества матриц Адамара с ростом их порядка «добыча» становится все сложнее. Каждая вновь найденная матрица Адамара порядка выше 428 в известном проблемном сэте порядков является предметом обсуждения научной общественности, занимающейся данной тематикой [9]. Ведь часто новая матрица Адамара – результат многомесячных настроек алгоритма и его реализации на высокопроизводительных системах. Вновь «добытые» матрицы Адамара появляются значительно реже очередной единицы криптовалюты.

Основные подходы к разработке эффективных методов майнинга матриц

За счет чего можно повысить эффективность майнинга матриц Адамара?

Во-первых, следует предпринимать усилия к поиску альтернативы известным методам и подходам. В этом смысле пионерскими являются методы, основанные на процедурах оптимизации, не свойственных рассматриваемой области компьютерных вычислений.

Матрицы Адамара всегда считались матрицами, которые нужно находить генерацией последовательностей элементов 1 и –1, после чего выполнять перестановки. В связи с этим область поиска в целом отнесли к развитой в западной литературе ветви комбинаторики. Однако еще Адамар отмечал, что это матрицы максимальные по детерминанту. Оптимизация детерминанта матрицы никак не может быть отнесена к перестановочным алгоритмам, она меняет абсолютные значения всех элементов матрицы от 0 до 1. Есть, хотя и очень малоизвестные, итерационные алгоритмы повышения детерминанта [10]. Разумеется, им нужно стартовое начальное приближение, чтобы оптимизатор не остановился в точке локального максимума – это известная болезнь итерационных процедур. Тем не менее это серьезное предложение для майнинга редких матриц. У комбинаторных методов нет возможности исправления даже незначительного дефекта начального приближения. Совсем иное дело – майнинг, построенный на оптимизации. Мы отмечаем это как очевидный, но еще малоизученный инструмент развития темы.

Собственно, структура матрицы и есть начальное приближение, необходимое оптимизатору. В качестве подсказки ему может навязываться структура, которую он разрушает, оптимизируя, но к ней можно возвращать оптимизатор, и тогда он становится очень мощным средством поиска.

Во-вторых, разработка эффективных методов поиска матриц Адамара связана с фиксацией ограничений, например на структуры: циклические, бициклические, трициклические, симметричные и др. и возможные для них порядки. Такая фиксация ограничений, несмотря на, казалось бы, усложнение задачи, тем не менее позволяет значительно повысить эффективность поиска матриц за счет упрощения или сокращения вычислительных затрат.

В целом результаты наших поисков показали способность новых подходов при разработке алгоритмов давать значительный результат.

Пример работы со структурами матриц

Рассмотрим повышение эффективности процесса майнинга матриц Адамара за счет указанной ранее фиксации их структуры.

Матрицы Адамара **H** на высоких порядках могут представлять собой разновидность четырехблочного массива Вильямсона [11] вида

$$\mathbf{H} = \begin{pmatrix} \mathbf{A} & \mathbf{B} & \mathbf{C} & \mathbf{D} \\ \mathbf{C} & \mathbf{D} & -\mathbf{A} & -\mathbf{B} \\ \mathbf{B} & -\mathbf{A} & -\mathbf{D} & \mathbf{C} \\ \mathbf{D} & -\mathbf{C} & \mathbf{B} & -\mathbf{A} \end{pmatrix},$$

в котором блоки **A**, **B**, **C** и **D** называются матрицами Вильямсона. Эти блоки являются, как правило, циклическими и обязательно симметричными. В этом ранее видели ключ к упрощению поиска. Поясним кратко, почему это не так. Симметричность влияет, скорее, на размер памяти компьютера, на котором эти матрицы ищутся, на техническую составляющую поиска.

Первые матрицы Адамара, найденные программно на компьютере, искались программами, удерживаемыми в памяти по кускам. То же относится и к самой матрице. На время поиска принудительная симметрия матриц Вильямсона может повлиять негативно, если несимметричных по блокам конструкций гораздо больше. То есть, улучшая алгоритмизуемость, если ее оценивать в размерах искомых матриц, поисковики ухудшали другой важный показатель майнинга – компьютерное время.

В результате предложения, сформулированного в работе [12], матрица **H** может быть построена в виде симметричной конструкции Пропус [13] на основе трех блоков **A**, **B** (**C = B**) и **D,** где только блок **A** симметричен, остальные несимметричны. При этом не теряется количество возможных решений, так как есть доказательства, что все матрицы Адамара либо симметричны, либо кососимметричны в целом, а не поблочно. Это очень сильное предложение способствовало получению большого количества новых матриц.

Конструкция Пропус [13, 14] и сходные с ним кососимметричные массивы гарантируют получение матрицы Адамара независимо от порядка матрицы. Массив Вильямсона таких гарантий дать не может. Решений и вовсе нет уже при размере блока 35. Далее такие негативные последствия принуждения к структуре, не свойственной матрицам Адамара, будут встречаться все чаще. Что доказали работы, призванные первоначально навести порядок и дать сводную таблицу матриц Вильямсона? Вместо таблицы получилось решето с дырками порядков нарастающей интенсивности.

Сверхбольшие каталоги последовательностей поиска, рассматриваемые в работе [15], содержат потенциальные первые строки циклических блоков (циркулянтов) **A**, **B** и **D** матрицы Адамара конструкции Пропус. Данные последовательности образуются и накапливаются в ходе работы различных алгоритмов их генерации [16].

Общий алгоритм работы с каталогами можно разделить на три этапа:

– генерация последовательностей, необходимых для получения циркулятнов **A**, **B** и **D**;

– фильтрация последовательностей;

– поиск совместимости последовательностей между возможными вариантами реализации блоков для формирования матрицы Пропус искомого порядка.

С ростом порядка искомой матрицы **H** скорость поиска резко падает из-за увеличения объема каталога. Он не удерживает вместе все три пары, нужные для **A**, **B** и **D**. Его полезно прореживать, что имеет аналогию – «обогащение» в майнинге подземных ископаемых. Это процедура является неотъемлемой в добывающей промышленности.

В случае простейшей реализации генератора [15, 16] количество случайно сгенерированных комбинаций последовательностей из 1 и –1 растет настолько быстро, что компьютер добирается до нужной комбинации неделями. Часто все эти сверхбольшие данные теряются ввиду крайне малой вероятности встретить все три последовательности. Ведь достаточно не быть одной из них, и огромная таблица сравнений будет перекрестно проверена зря.

Обогащение исходных последовательностей

К разновидности полезной фильтрации относятся признаки совместимости последовательностей для построения блоков матрицы Адамара по Вильямсону и близких к ней. Когда одна последовательность из трех у матрицы конструкции Пропус сравнивается с двумя другими, она не может быть уже вполне произвольной. Фильтр совместимости укорачивает поле поиска, как обогатительная фабрика на шахте отделяет породу от полезного ископаемого.

Итак, не все последовательности являются источниками блоков-циркулянтов, пригодных для построения матрицы Адамара конструкции Пропус. Оказывается, фильтрацию можно осуществить такой простой и хорошо известной инженерам процедурой, как дискретное преобразование Фурье (ДПФ) или его быстрой версией – БДПФ. Ненужная последовательность имеет всплеск спектра, что свидетельствует о наличии в ней гармонической составляющей, входящей в противоречие с потенциальным решением. Можно доказать наличие порога, выше которого гармоника делает последовательность непригодной.

Простейший пороговый фильтр по выбросам спектра убирает до 99% ненужных последовательностей. Тогда, например, при синтезе матрицы из 100 парных оснований, вместо $100 \times 100 = 10000$ их перекрестных сравнений

следует выполнить всего одно. Приятная особенность фильтрации состоит в том, что количество перекрестных проверок растет квадратично, тогда как количество фильтраций (и их аналогов, фильтров совместимости) – линейно. Что оправдывает такое «обогащение» сверхбольших каталогов поиска.

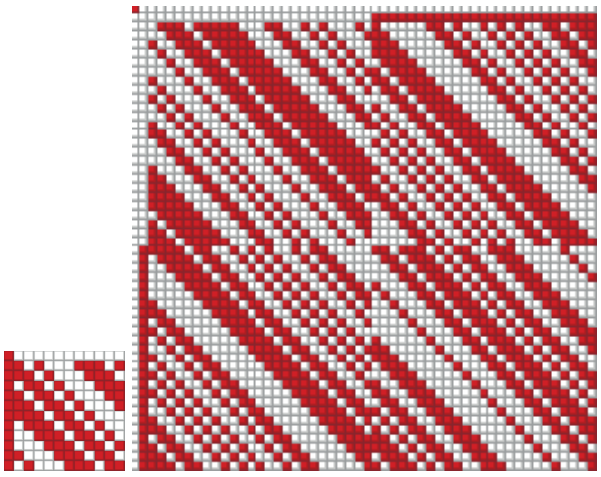
Разрешимость задачи поиска для высоких порядков матриц прямо зависит от удачного накопления сгенерированных последовательностей. Поиск таких последовательностей можно ускорить, не меняя алгоритм в целом, а лишь заменив простейший генератор на более сложный с фильтром. Разумеется, матрица Фурье не единственная ортогональная матрица, используемая для построения фильтров. Парадоксально, но для поиска новых матриц Адамара могут использоваться уже найденные матрицы Адамара в процедурах фильтрации, устроенных сходно с БДПФ. Матрицы могут быть того же порядка или усеченные.

Основные усилия следует направить на решение проблемы «повторяемости» выходов генераторов бессмысленными последовательностями при формировании больших каталогов, из-за которой растут их объемы. Некачественное заполнение ведет к расходованию ресурсов майнинга на обработку пустой «породы» – шлаков этого процесса в виде спектрально неустойчивых и несовместимых образцов.

Значение новых преобразований алгоритма велико, за этим стоит накопление опыта отечественной школы поиска, майнинга редких матриц. В каком-то смысле хорошо налаженный майнинг характеризует поисковые команды. Наивная стадия «промывочных корзин» характерна для команд поисковиков на стадии их становления.

В качестве примера можно рассмотреть поиск матриц Мерсенна **M** [3, 17]. Это разновидность матриц Адамара, взятая без каймы (рисунок).

В данном случае поиск может быть сосредоточен на поиске подходящей последовательности длины 11 вида [–1,1,–1,1,1,1,–1,–1,–1,1,–1], которая дает сразу две ортогональные матрицы – Адамара порядка 12 и Мерсенна порядка 11. Матрица Мерсенна порядка 11, являясь ядром матрицы Адамара порядка 12, ортогональна при уменьшении амплитуды отрицательного элемента до восстановления свойства ортогональности. Если есть поле Галуа GF(11), то поиск элементарен, он сводится к выяснению квадратичных вычетов или к вычислению показательной функции, дающей адреса (место в матрице) отрицательных элементов. Но ведь поля может и не быть, и тогда этот пример легко превращается в тестовый,



Две матрицы Адамара с основанием Мерсенна

где генерируемые последовательности длины 11 укладываются в каталог.

Приведенный модельный пример хорош для первых экспериментов по майнингу, мы его описываем только потому, что он прост в реализации. Перекрестных сравнений одна последовательность не вызывает, но она есть при иной конструкции матриц Мерсенна, состоящей из бицикла и бинарной каймы (см. рисунок). В таком случае каталог складывается из двух половинок матрицы, и прореживание уже более существенно скажется на времени майнинга. Но ничто не мешает условно разбивать на половинки и первую начальную последовательность, выбирая для каталога ее четные и нечетные элементы, кроме стартового. Как видно, к использованию фильтра ведут обе постановки задачи.

Эти характерные задачи осуществления майнинга предлагаются для исследований в учебном процессе кафедры вычислительных систем и сетей ГУАП. В качестве базовых программных решений для развития методов майнинга матриц Адамара используются программы, ориентированные на работу с детерминантом матриц [18, 19], со структурой матриц [20–22], с бинарными последовательностями [23, 24].

Заключение

В работе рассматривается становление техники майнинга, крайне важной в своем развитии для поиска матриц Адамара. Ранее над обогащением выборок работа почти не велась. Все отмеченные нами поисковые процедуры выполнялись без использования суперкомпьютеров по «голой» и не обогащенной выборке. Развитие майнинга матриц в этом направлении показывает качественное решение тех же задач более

высокого порядка с возрастанием культуры поиска, когда он от редких находок и «рекордов» переходит в стадию регулярных (гарантированных) находок в приемлемые временные сроки.

Авторы выражают благодарность обладателю премии Пирси почетному профессору University of Wollongong (Австралия) Дженнифер Себерри за творческие рекомендации, способствовавшие становлению современного майнинга матриц Адамара.

Библиографический список

1. *Jennifer S., Yamada M.* Hadamard Matrices: Constructions using number theory and linear algebra. Wiley, 2020. 384 p.

2. *Балонин Н. А., Сергеев М. Б.* Нормы обобщенных матриц Адамара // Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. 2014. № 2. С. 5–11.

3. *Mohan M. T.* p-almost Hadamard matrices and λ-planes // Journal of Algebraic Combinatorics. 2020. 20 p.

4. *Wang R.* Introduction to Orthogonal Transforms with Applications in Data Processing and Analysis. Cambridge University Press, 2010. 504 p.

5. *Seberry J., Wysocki B., Wysockiet T.* On some applications of Hadamard matrices // Metrika. 2005. № 62 (2–3). P. 221–239.

6. *Vostrikov A., Sergeev M., Balonin N., Sergeev A.* Use of symmetric Hadamard and Mersenne matrices in digital image processing. URL: http://10.1016/j.procS.2018.08.042

7. *Mironovsky L. A., Slaev V. A.* Strip-Method for Image and Signal Transformation. Berlin: Boston: De Gruyter, 2011.

8. *Evangelaras H., Koukouvinos C., Seberry J.* Applications of Hadamard matrices // Journal of telecommunications and information Technology. 2003. № 2. P. 3–10.

9. *Kharaghani H., Tayfeh-Rezaie B. A.* Hadamard matrix of order 428 // Journal of Combinatorial Designs. 2005. Vol. 13. P. 435–440.

10. *Балонин Н. А., Сергеев М. Б., Суздаль В. С.* Динамические генераторы квазиортогональных матриц семейства Адамара // Труды СПИИРАН. 2017. Вып. 5 (54). С. 224–243.

11. *Acevedo S., Dietrich H.* New infinite families of Williamson Hadamard matrices // Australian Journal of Combinatorics. 2019. Vol. 73 (1). P. 207–219.

12. *Seberry J., Balonin N. A.* Two infinite families of symmetric Hadamard matrices // Australian Journal of Combinatorics. 2017. Vol. 69 (3). P. 349–357.

13. *Балонин Н. А., Сергеев М. Б.* Пропус 92 и 116 // Информационно-управляющие системы. 2016. № 2(81). С. 101–103.

14. *Встриков А. А., Декханбаев Д. С., Куртяник Д. В., Сергеев А. М.* О стратегиях вычисления матриц Адамара симметричных структур блочной конструкции Балонина-Себерри // Телекоммуникации. 2020. № 5. С. 20–27.

15. *Абузин Л. В., Балонин Ю. Н., Куртяник Д. В., Сергеев А. М.* Генерация, фильтрация и поиск экстремума в сверхбольшом каталоге бинарных последовательностей // Обработка, передача и защита информации в компьютерных системах: первая Всеросс. науч. конф. (СПб., 14–22 апр. 2020 г.): сб. докл. СПб.: ГУАП, 2020. С. 121–124.

16. *Balonin Y., Abuzin L., Sergeev A., Nenashev V.* The Study of Generators of Orthogonal Pseudo-Random Sequences // Smart Innovation, Systems and Technologies. 2019. Vol. 143. P.125–133.

17. *Балонин Н. А., Сергеев М. Б.* Матрицы Мерсенна и Адамара // Информационно-управляющие системы. 2016. № 1 (80). С. 2–15.

18. Свидетельство о государственной регистрации программы для ЭВМ № 2018616389 от 01.06.2018 г. Программный комплекс поиска матриц локального максимума детерминанта с самомасштабированием / Балонин Ю. Н., Сергеев А. М., Синицына О. И.

19. Свидетельство о государственной регистрации программы для ЭВМ № 2019615126 от 18.04.2019 г. Программа генерации матричных раппортов «Калейдоскоп» / Сергеев М. Б., Сергеев А. М., Балонин Н. А., Балонин Ю. Н.

20. Свидетельство о государственной регистрации программы для ЭВМ № 2018616390 от 01.06.2018 г. Программный комплекс поиска бициклических матриц на основе таблицы перекрестных ссылок / Балонин Ю. Н., Сергеев А. М.

21. *Балонин Ю. Н., Клюковкин В. Р., Сергеев А. М.* Численный алгоритм эффективного поиска бициклических матриц на основе таблицы перекрестных ссылок // Научная сессия ГУАП: сб. докл.: в 3 ч. Ч. 2. СПб., 2017. С. 179–184.

22. Свидетельство о государственной регистрации программы для ЭВМ № 2018617112 от 19.06.2018 г. Программный комплекс клиент-серверного поиска бициклических матриц Адамара в реальном масштабе времени / Балонин Ю. Н., Сергеев А. М.

23. Свидетельство о государственной регистрации программы для ЭВМ № 2020662384 от 13.10.2020 г. Накопление пар ортогональных последовательностей для поиска симметричных ортогональных матриц Адамара с тремя блоками (Пропусов) / Балонин Ю. Н., Сергеев А. М.

24. Свидетельство о регистрации программы для ЭВМ № 2021660188 от 23.06.2021 г. Программа генерации бинарных последовательностей с заданным количеством отрицательных и положительных элементов / Сергеев М. Б., Ненашев В. А., Григорьев Е. К., Сенцов А. А.

ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ЗАЩИТА ИНФОРМАЦИИ

БЕЗОПАСНОСТЬ ИНФОРМАЦИОННЫХ СИСТЕМ И ИНФОКОММУНИКАЦИОННЫЕ СИСТЕМЫ

УДК 007 **DOI: 10.31799/978-5-8088-1701-2-2022-2-174-177**

А. В. Борисовская*
ассистент
А. М. Тюрликов*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ВЫЧИСЛЕНИЕ ВЕРХНЕЙ ОЦЕНКИ СРЕДНЕЙ ЗАДЕРЖКИ ДЛЯ СИСТЕМЫ СО СЛУЧАЙНЫМ ДОСТУПОМ И МНОЖЕСТВЕННЫМ ВЫХОДОМ

Рассматривается система со случайным доступом и множественным выходом. В случае успешной передачи систему покидает пользователь, который передал сообщение, и близкорасположенные к нему пользователи. Для этой системы предлагается численный метод вычисления верхней оценки средней задержки.
Ключевые слова: *системы связи, бесконечное число пользователей, множественный выход, среднее число пользователей в системе, средняя задержка, марковские цепи.*

A. V. Borisovskaya*
Assistant
A. M. Turlikov*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

UPPER BOUND OF AVERAGE DELAY FOR SYSTEM WITH RANDOM ACCESS AND MULTIPLE DEPARTURE CALCULATION

In this paper, we consider a system with random access and multiple departure. In case of successful transmission, the user who transmitted the message and the users close to him leave the system. For this system, a numerical method for calculating the upper bound of the average delay is proposed.
Keywords: *communication systems, unlimited number of users, multiple departure, average number of users, average delay, Markov chains.*

Системы связи с бесконечным числом пользователей исследуются давно [1]. С появлением систем интернета вещей эта задача становится еще актуальнее. В большинстве исследований систем с бесконечным числом пользователей рассматривались модели, которые стабильны для интенсивности входного потока меньше, чем e^{-1}. То есть эффективность использования канала в таких моделях составляет не более 36%. В системах интернета вещей [2–6] интенсивность входного потока может быть во много раз больше. В работе [7] впервые была предложена модель системы, которая остается стабильной при любой интенсивности входного потока. В этой модели при успешной передаче сообщения систему покидает пользователь, который передал сообщение, и близкорасположенные к нему пользователи. Она получила название «модель с множественным выходом». Эта модель была предложена недавно, поэтому ее характеристики малоизучены. В связи с этим анализ характеристик системы с множественным выходом – актуальная задача.

В работе [8] рассматривалась система с множественным выходом, предложенная в работе [7], и исследовался вопрос оценки средней за-

держки для нее. Для описания системы с множественным выходом была введена модель, которая для удобства изложения была названа моделью М0 [8]. Модель М0 отражает основные особенности системы, предложенной в работе [7], но для упрощения анализа в ней пользователи находятся на окружности. Оценка средней задержки в системе М0 – это сложная задача, так как модель М0 описывается многомерной марковской цепью. Поэтому для анализа средней задержки была введена упрощенная модель системы с множественным выходом, которую авторы обозначили М1 [8]. В модели М1, в отличие от модели М0, в случае «успеха» пользователи, оставшиеся в системе, равномерно перераспределяются по окружности. Модель М1 можно описать одномерной марковской цепью, что позволяет оценить среднюю задержку. В работе [8] приведен расчет средней задержки для модели М1 и сформулирована следующая гипотеза: «При любой интенсивности входного потока среднее число пользователей в системе М0 меньше, чем среднее число пользователей в системе М1». Если эта гипотеза верна, то оценка средней задержки в системе М1 является верхней для системы М0. Однако невозможно определить, насколько точной является оценка средней задержки, предложенная в работе [8]. В текущей статье будет предложен метод вычисления нижней оценки средней задержки с заданной точностью для модели М1, которая будет являться верхней оценкой средней задержки для системы М0.

Рассмотрим упрощенную модель системы с множественным выходом М1, которая предложена в работе [8]. Эту модель можно описать следующими допущениями.

– *Допущение* 1. В системе имеются базовая станция и неограниченное количество пользователей. Пользователи равномерно распределены по окружности, в центре которой находится базовая станция. Процесс появления пользователей в системе описывается пуассоновским потоком с интенсивностью λ. Другими словами, интервалы времени между появлениями пользователей в системе независимы и распределены по экспоненциальному закону. При этом координаты появившихся в системе абонентов распределены равномерно по окружности.

– *Допущение* 2. Время работы системы разделено на окна. Все окна имеют одинаковую длину, равную времени передачи одного сообщения. Моменты разделения окон известны всем пользователям. Абонент может начинать передачу только в начале окна.

– *Допущение* 3. В каждом окне может произойти одно из трех событий «Успех» (передает один абонент), «Пусто» (никто не передает) или «Конфликт» (передают два или более абонентов). В конце каждого окна базовая станция передает по широковещательному каналу всем пользователям информацию о событии в окне.

– *Допущение* 4. Если в окне с номером t произошло событие «Успех», то базовая станция передает по широковещательному каналу всем пользователям координаты пользователя, который успешно передал сообщение. Получив эту информацию, данный пользователь и пользователи, которые находятся на расстоянии меньше чем δ от него, покидают систему.

– *Допущение* 5. Предполагается, что в начале каждого окна базовой станции известно число активных пользователей. Активными будем называть пользователей, имеющих готовое для передачи сообщение. В начале каждого окна базовая станция передает по широковещательному каналу всем пользователям вероятность для передачи сообщения: $p_t = 1 / N_t$, где t – номер текущего окна, N_t – количество активных пользователей. Все пользователи независимо друг от друга с вероятностью p_t принимают решение о передаче сообщения в окне с номером t.

– *Допущение* 6. Если в окне с номером t произошло событие «Успех», то все пользователи, оставшиеся в системе, равномерно перераспределяются по окружности.

Длину окружности примем за единицу. Длину дуги окружности, на которой находятся пользователи, покидающие систему в окне t, будем обозначать ε. Тогда $\varepsilon = 2\delta$.

Модель М1 можно описать одномерной марковской цепью. Следовательно, функционирование модели можно описать следующим рекуррентным выражением:

$$N_{t+1} = N_t - L_t + V_t,$$

где N_t – количество активных пользователей в системе в окне t, N_{t+1} – количество активных пользователей в системе в окне $t+1$, V_t – количество пользователей, появившихся в системе в окне t, L_t – количество пользователей, покинувших систему в окне t. V_t – это случайная величина, распределенная по закону Пуассона с параметром λ. L_t – это случайная величина, математическое ожидание которой вычисляется следующим образом:

$$E[L_t] = \Pr\{\theta_t = У\}(1 + (E[N_t] - 1)\varepsilon).$$

Здесь θ_t – событие в окне t, которое может принимать одно из трех значений: У – «Успех», П – «Пусто», К – «Конфликт». Вероятность, что в окне t произошло событие «Успех», вычисляется по следующей формуле:

$$\Pr\{\theta_t = У\} = \left(1 - \frac{1}{N_t}\right)^{N_t - 1}.$$

Поиск стационарного распределения данной Марковской цепи – сложная задача, так как число состояний бесконечно. В работе [9] предлагается подход, который основан на изменении системы таким образом, что можно найти среднее число пользователей в системе и среднюю задержку. Он позволяет вычислить характеристики системы с различной точностью, т. е. построить верхние границы для средней задержки и среднего числа пользователей в системе. Рассматривается система с бесконечным числом пользователей и «оптимальным» алгоритмом ALOHA [9]. Однако этот подход неприменим для системы с множественным выходом. При изменении системы с множественным выходом, в соответствии с данным подходом, система становится нестабильной для интенсивности входного потока больше чем e^{-1}.

Ограничим число состояний марковской цепи, которая описывает функционирование модели M1, и найдем ее стационарное распределение. Будем предполагать, что имеется только $K + 1$ состояний, т. е. число пользователей в системе не может быть больше K.

Обозначим через p_{ij} вероятность перехода из состояния i в состояние j. Тогда переходные вероятности данной марковской цепи можно вычислить по следующим формулам:

– если $i = 0$:

$$p_{ij} = \frac{\lambda^j}{j!} e^{-\lambda};$$

– если $i > 0, j \geq i$:

$$p_{ij} = \left(1 - \left(1 - \frac{1}{i}\right)^{i-1}\right) \frac{\lambda^{j-i}}{(j-i)!} e^{-\lambda} + \left(1 - \frac{1}{i}\right)^{i-1} \times$$

$$\times \sum_{m=1}^{i} \left(C_{i-1}^{m-1} \varepsilon^{m-1} (1-\varepsilon)^{i-m} \frac{\lambda^{j-i+m}}{(j-i+m)!} e^{-\lambda} \right);$$

– если $i > 0, j < i$:

$$p_{ij} = \left(1 - \frac{1}{i}\right)^{i-1} \times$$

$$\times \sum_{m=0}^{j} \left(C_{i-1}^{i-j+m-1} \varepsilon^{i-j+m-1} (1-\varepsilon)^{j-m} \frac{\lambda^m}{m!} e^{-\lambda} \right);$$

Обозначим через $\pi_i = Pr\{N_t = i\}$ вероятность нахождения системы в состоянии i, т. е. вероятность, что в системе находится i активных абонентов. Выпишем систему уравнений для стационарных вероятностей:

$$\begin{cases} \pi_0 = \pi_0 p_{00} + \pi_1 p_{10} + \pi_2 p_{20} + \ldots + \pi_K p_{K0} \\ \pi_1 = \pi_0 p_{01} + \pi_1 p_{11} + \pi_2 p_{21} + \ldots + \pi_K p_{K1} \\ \pi_2 = \pi_0 p_{02} + \pi_1 p_{12} + \pi_2 p_{22} + \ldots + \pi_K p_{K2} \\ \ldots \\ \pi_{K-1} = \pi_0 p_{0K-1} + \pi_1 p_{1K-1} + \\ + \pi_2 p_{2K-1} + \ldots + \pi_K p_{KK-1} \\ \pi_0 + \pi_1 + \pi_2 + \ldots + \pi_K = 1 \end{cases}. \quad (1)$$

Вычислительная сложность решения данной системы уравнений зависит от K. Получив решение системы уравнений (1), можно найти среднее число пользователей в системе по следующей формуле:

$$\bar{N} = \sum_{i=0}^{K} i\pi_i.$$

Зная среднее число пользователей в системе при определенном значении интенсивности входного потока, можно найти среднюю задержку по формуле Литтла:

$$\bar{D} = \frac{\bar{N}}{\lambda}.$$

На рис. 1 приведена зависимость средней задержки от K для $\varepsilon = 0{,}1$ и различных значений λ. Чтобы получить оценку средней задержки при $\lambda = 2$ с точностью меньше 0,01 достаточно выбрать $K \approx 100$, при $\lambda = 5{,}2$ – $K \approx 300$ и при $\lambda = 10$ – $K \approx 500$.

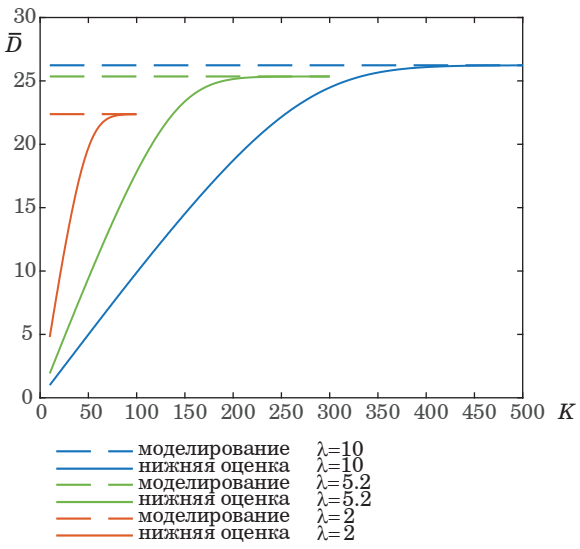


Рис. 1. Нижние границы средней задержки для модели M1 при $\varepsilon = 0{,}1$ и различных значениях λ

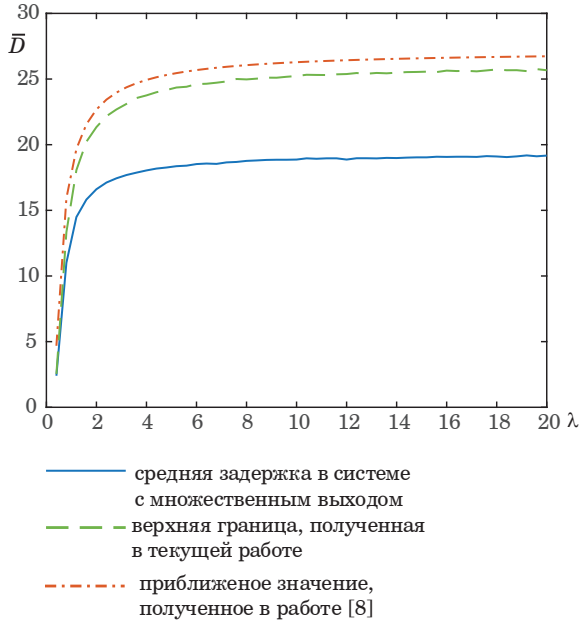


Рис. 2. Верхняя граница и приближенное значение средней задержки для системы с множественным выходом при ε = 0,1

Таким образом, ограничив число состояний марковской цепи, можно найти ее стационарное распределение и вычислить среднюю задержку в системе. В данной работе этот подход был применен к упрощенной системе с множественным выходом M1, которую можно описать одномерной марковской цепью. Благодаря этому подходу получена нижняя граница средней задержки для M1. При больших K нижняя граница средней задержки для системы M1 является верхней границей для системы с множественным выходом M0.

В работе [8] приведен приближенный расчет средней задержки для системы с множественным выходом, а в данной работе представлен метод, который позволяет, увеличивая K, получить сколь угодно точную оценку.

На рис. 2 приведены следующие зависимости: средняя задержка для системы с множественным выходом от λ, полученная моделированием модели M0; приближенное значение средней задержки от λ, полученное в работе [8]; верхняя граница для средней задержки от λ, полученная в текущей работе. Значения верхней границы средней задержки для каждого значения λ были получены при своем значении K.

В данной работе предложен метод численного вычисления средней задержки для системы со случайным доступом и множественным выходом. Данный метод уточняет подход, предложенный в работе [8], т. е. приближенное значение границы средней задержки из работы [8].

Библиографический список

1. *Цыбаков Б. С., Михайлов В. А.* Свободный синхронный доступ пакетов в широковещательный канал с обратной связью // Проблемы передачи информации. 1978. № 14:4. С. 32–59.

2. *Polyanskiy Y.* A perspective on massive random-access // 2017 IEEE International Symposium on Information Theory (ISIT). 2017. P. 2523–2527.

3. *Fengler A., Caire G., Jung P., Haghighatshoar S.* Massive MIMO unsourced random access. 2019. URL: https://arXiv preprint arXiv:1901.00828 (дата обращения: 20.11.2021).

4. Massive access for future wireless communication systems / Y. Wu, X. Gao, S. Zhou, W. Yang et al. // IEEE Wireless Communications. 2020. T. 27, № 4. P. 148–156.

5. *Shahab M. B., Abbas R., Shirvanimoghaddam M., Johnson S. J.* Grant-free non-orthogonal multiple access for IoT: A survey // IEEE Communications Surveys & Tutorials. 2020. 22(3). P. 1805–1838.

6. How to Identify and Authenticate Users in Massive Unsourced Random Access / R. Kotaba, A. E. Kalør, P. Popovski, I. Leyva-Mayorga et al. 2021. URL: https://arXiv preprint arXiv:2104.10576. (дата обращения: 20.11.2021).

7. *Foss S., Turlikov A., Grankin M.* Spatial random multiple access with multiple departure // 2017 IEEE International Symposium on Information Theory (ISIT). 2017. P. 2728–2731.

8. *Borisovskaya A., Glebov A., Turlikov A.* Estimation of average delay in systems with unsourced random access and multiple departure // 2021 XVII International Symposium «Problems of Redundancy in Information and Control Systems» (REDUNDANCY) – IEEE. 2021. P. 28–33.

9. *Burkov A. A., Sheer S., Turlikov A. M.* Arbitrarily Accurate Approximation of Numerical Characteristics of Stationary ALOHA // IEEE Wave Electronics and its Applications in Information and Telecommunication Systems (WECONF). 2021. P. 1–8.

УДК 004.7 **DOI: 10.31799/978-5-8088-1701-2-2022-2-178-182**

В. К. Витвинов*
магистрант
Н. А. Янковский*
ассистент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС ДЛЯ ИССЛЕДОВАНИЯ ЭФФЕКТИВНОСТИ ИСПОЛЬЗОВАНИЯ QUIC-ПРОТОКОЛА В СИСТЕМАХ ИНТЕРНЕТА ВЕЩЕЙ

В связи с постоянно растущим спросом на высокоскоростное подключение к Интернету, которое может удовлетворить постоянную потребность приложений в более высокой скорости передачи, компания Google, являющаяся лидером во многих веб-сервисах, запустила новый протокол на основе UDP под названием quick UDP internet connections (QUIC), который направлен на обеспечение более быстрой доставки данных без необходимости обновления или модификации сетевой инфраструктуры. На данный момент большинство исследований посвящено сценариям для передачи данных на большой скорости. Именно поэтому необходимо рассмотреть и изучить данную технологию для возможности ее интегрирования в существующие и разрабатываемые системы интернета вещей, так как в таких системах важна передача данных не с большей скоростью, а с низкой задержкой и малым объемом данных.
Ключевые слова: *протокол QUIC, системы интернета вещей, оценка сетевых ресурсов, передача малых данных, Google, TCP, UDP.*

V. K. Vitvinov*
Postgraduate Student
N. A. Jankovsky*
Assistant
**St. Petersburg State University of Aerospace Instrumentation*

HARDWARE-SOFTWARE SYSTEM FOR INVESTIGATING QUIC PROTOCOL EFFICIENCY IN THE INTERNET OF THINGS SYSTEMS

With the ever-increasing demand for high-speed Internet connections, which can meet the constant need of applications for higher transmission speeds, Google, the leader in many web services, has launched a new protocol based on UDP called QUIC, which aims to provide faster data delivery without the need to upgrade or modify the network infrastructure. At the moment, most of the research is focused on high-speed data transfer scenarios. That is why it is necessary to consider and study this technology for the possibility of integrating it into existing and emerging Internet of Things systems, as in such systems it is important to transmit data not at higher speeds, but with low latency and small amount of data.
Keywords: *QUIC protocol, Internet of Things systems, network resource assessment, small data transfer, Google, TCP, UDP.*

Введение

Цель статьи –произвести краткий обзор протокола QUIC на основе работ по данной тематике, представить возможную будущую реализацию испытательного стенда, который экспериментально используется для оценки протокола QUIC в различных сетевых условиях и сценариях в системах интернета вещей. В частности, преимущество QUIC в производительности с точки зрения задержки и пропускной способности рассматривается с учетом различных условий сети, которые напоминают реальную интернет-среду. Для достижения цели предлагается рассмотреть испытательный стенд, который контролируемым образом имитирует сетевые нарушения, возникающие в реальной сети, такие как потеря пакетов, ограничение пропускной способности. После этого протестировать работу протокола QUIC в условиях реальных проводных и беспроводных сетей.

В настоящее время миллиарды устройств и компьютеров подключены к сети Интернета. Среди всех таких устройств можно выделить область систем интернета вещей. В связи с увеличением их количества использующиеся алго-

ритмы передачи данных не всегда могут удовлетворить требуемые потребности таких устройств и систем. Поэтому проводится множество исследований, направленных на поиск новых алгоритмов и способов ускорения обработки информации при передаче без изменения основных принципов работы уже используемых протоколов. Однако для получения значительных результатов в процессе передачи данных необходимы дополнительные исследования о возможности модернизации существующих протоколов связи.

Описание и принцип работы протокола QUIC

Протокол QUIC был предложен Google в 2013 г. [1]. Его основная цель – уменьшить задержку, работая поверх UDP. На рис. 1 показаны верхние сетевые уровни: SPDY (протокол прикладного уровня, заменяет части протокола HTTP, позволяет снижать время загрузки данных) [2] и QUIC. Как показано на рисунке, уровень QUIC заменяет TCP и безопасность транспортного уровня (TLS), который используется для защиты всех коммуникаций между сервером и веб-браузерами в SPDY/HTTP2. Уровень QUIC также включает безопасность, эквивалентную TLS в SPDY. Далее кратко описываются наиболее важные характеристики протокола QUIC.

HTTP через TCP использует TLS для обеспечения безопасности при передаче данных, в то время как в QUICK TLS заменяется криптографией. Основное преимущество криптографии, использующейся для сокращения задержки QUIC, заключается в том, что, когда клиент кэширует информацию с сервера, он может восстановить зашифрованное соединение без обратной связи, тем самым уменьшая задержку (рис. 2). Кроме того, криптография имеет более высокий уровень безопасности, чем TLS, так как она всегда зашифрована, поэтому основной единицей передачи является стандартный пакет UDP.

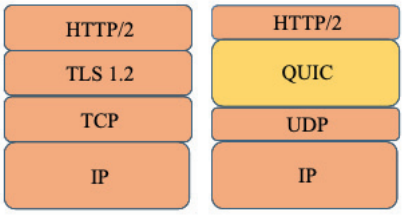


Рис. 1. Верхние сетевые уровни: а) SPDY, б) QUIC

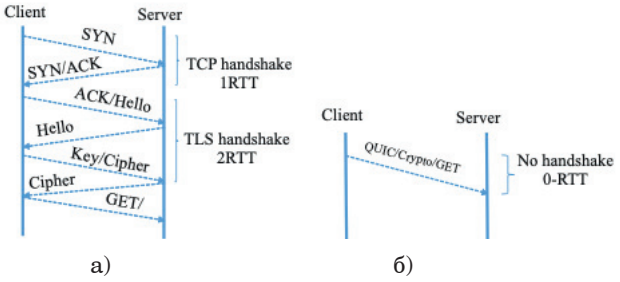


Рис. 2. Установка соединения: а) 3-RRT TCP и «рукопожатие» в TLS, b) 0-RRT в QUIC

В SPDY мультиплексирование приводит к блокировке головной линии Head of Line Blocking (HOLB), поскольку оно мультиплексирует множество сеансов потоков по одному соединению, и любая потеря пакета, которая произойдет, заблокирует все потоки до тех пор, пока не поступит повторная передача потерянного пакета. Как показано на рис. 3, *a*, поток, состоящий из пакетов 2–4, и поток, состоящий из пакетов 5 и 6, блокируются, потому что произошла потеря пакета 1. Блокировка будет происходить до повторной передачи пакета номер 1. Однако, с другой стороны, предотвращение HOLB происходит из-за того, что он построен по протоколу UDP, поэтому он может поддерживать доставку не по порядку и имеет несколько байтовых потоков, в результате потери пакетов относятся к отдельному потоку, и поток без потерь все еще может быть повторно собран и отправлен в ходе передачи. Рис. 3, *b* показывает, что, несмотря на отбрасывание пакета номер 1, другие потоки пакетов принимаются и повторно собираются клиентом без какой-либо задержки, которая может возникнуть в случае ожидания повторной передачи потерянного пакета.

Основное изменение в идентификации соединения в QUIC – добавление 64-битного идентификатора соединения (CID) на прикладном уровне, который генерируется клиентом случайным образом. В TCP соединение идентифицируется четырьмя идентификаторами: адрес источника, порта источника, адрес назначения и порт назначения, поэтому, если клиент изменяет свой IP-адрес или номер порта, любое активное TCP-соединение прерывается. Для восстановления соединения требуется 3-стороннее рукопожатие. С другой стороны, когда QUIC клиент меняет IP-адрес или номер порта, он может продолжить соединение, используя старый идентификатор соединения (CID), без повторного подтверждения или прерывания соедине-

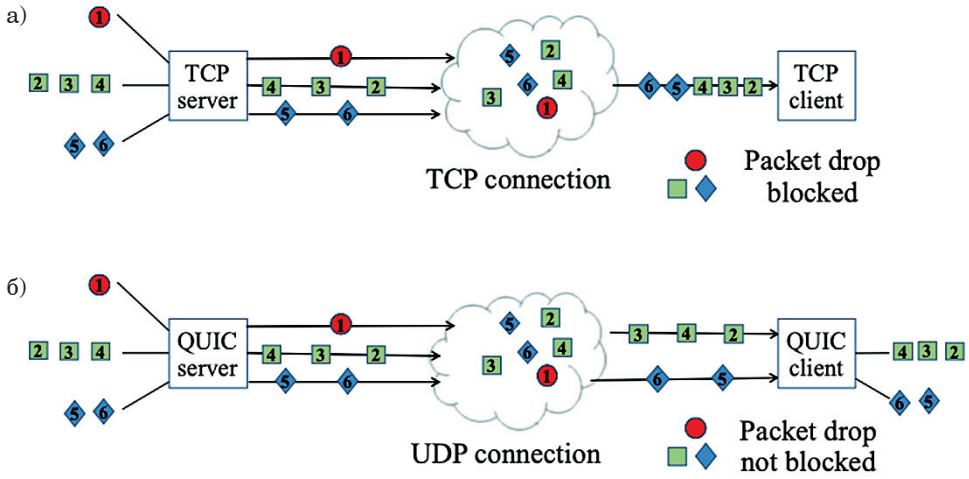


Рис. 3. Блокировка головной линии: а) HOLB в TCP, б) игнорирование HOLB в QUIC

ния. Эта функция будет полезна на мобильных устройствах во время роуминга.

Обзор схожих работ и оценка полученных результатов

Ранее было смоделировано [3] сравнение протоколов QUIC и TCP. В данной работе сравнение происходит путем тестирования двух протоколов на испытательном стенде, который разработан для демонстрации в двух сценариях: первый исследует работу протоколов в условиях контролируемой сети, которая никак не реагирует на возможные случайные задержки и внешние факторы. Также в этом сценарии предусмотрен сетевой эмулятор, который имитирует нарушения в сети, например потерю пакетов, ограничение пропускной способности и частоту битовых ошибок. Второй сценарий рассматривает работу в реальных сетях передачи данных. Это делается для того, чтобы продемонстрировать влияние реальных внешних сетевых факторов. На основе работы этих сценариев был сделан вывод, что протокол QUICK превосходит TCP в условиях реальных сетей. Таким образом, потенциал QUIC достаточно велик, чтобы заменить TCP. В ходе работы авторы наглядно проиллюстрировали работу обоих протоколов. Лучший результат сравнения был получен в условиях передачи файла размером 9 Мб при постороннем трафике. Многопоточность протокола QUIC позволила не отвлекаться на постороннюю передачу за счет параллельной обработки отправки и получения данных. Результат продемонстрирован на рис. 4.

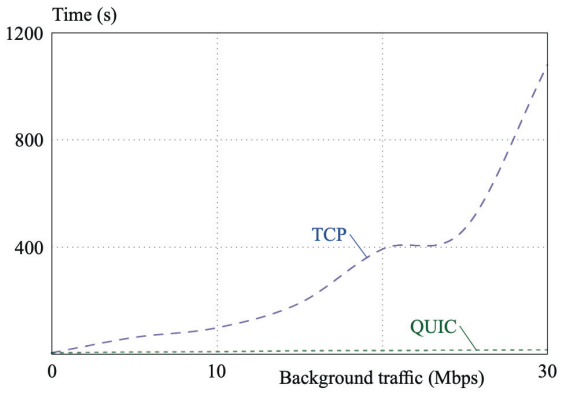


Рис. 4. Работа протоколов при постороннем трафике

Планирование будущей разработки для оценки и внедрения протокола QUIC

В ходе дальнейших исследований изучим применение протокола QUIC в системах интернета вещей. Полученные результаты в ранее рассмотренной работе не могут применяться для исследования ввиду того, что коллеги брали за основу передачу файла большого размера. В системах интернета вещей основными единицами являются датчики, и передаваемые с них данные и получаемые ими команды имеют очень маленький размер. Иногда такие данные даже не требуют разбития на пакеты. Значит, для возможности применения протокола QUIC необходимо оценить его работу при передаче данных малого объема в сценариях передачи данных от большого количества датчиков и при передаче данных через интернет.

Для оценки применимости протокола QUIC к системе интернета вещей планируется реали-

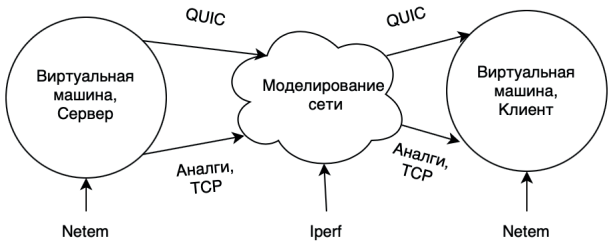


Рис. 5. Работа первого стенда на основе виртуальных машин

зовать несколько идентичных стендов. Первый будет реализован при использовании двух виртуальных машин со следующими характеристиками: 64-разрядная операционная система под управлением Ubuntu 14.04 с ядром Linux 3.14.2 (рис. 5). На первой виртуальной машине будет развернута серверная часть, на второй – клиентская, а также установлены инструменты Netem и Iperf для ручной настройки параметров передачи и регулировки искусственно созданных помех. Работа в данном стенде будет реализована с помощью двух вариантов построения системы. В первом варианте потери пакетов будут имитироваться с помощью независимой модели Бернули [4], во втором потеря пакетов будет описываться моделью Гильберта – Эллиота [5]. Таким образом, потеря пакетов будет появляться независимо, имитируя условия реальной передачи. Для каждого из сценариев будут передаваться одинаковые сообщения размером от нескольких байт до нескольких десятков килобайт, что позволит оценить влияние маленьких размеров данных на работу протокола.

Второй стенд будет реализован на компьютерах, характеристики которых будут аналогичны виртуальным машинам. При испытаниях это позволит учесть даже собственные шумы электронных компонентов, так как один из рассматриваемых сценариев включает передачу данных на основе аппаратных средств, по технологии LoRa, где передача и прием происходят ниже уровня собственных шумов. Также будет произведено тестирование передачи файлов маленького объема уже в условиях работы реальных сетей с посторонним трафиком, внешними шумами, случайными задержками. Демонстрация схемы работы стенда представлена на рис. 6.

Также будет рассмотрены условия комбинированной работы протокола. Данное испытание подразумевает совместное тестирование передачи данных по протоколу QUIC между датчиками, компьютерами и мобильными устройствами. Здесь будет возможность сравнить скорость работы мобильных и стационарных платформ с протоколом QUIC, оценить преимущества и недостатки, понять, как лучше строить архитектуру сети при работе в системах интернета вещей. На рис. 5 продемонстрирована схема работы стендов. Здесь на стороне клиента и сервера устанавливаются инструменты для эмуляции сети. Кроме того, применяется генератор трафика, роль которого выполняет инструмент Iperf. В процессе передачи будет происходить сравнение работы протокола QUIC с работой его конкурентов при различных условиях и ограничениях. При выполнении сценариев будут рассматриваться ограничения, связанные с изменением фонового трафика, битовыми ошибками, задержками сети, пропускной способностью и различными средами передачи – по воздуху и проводному соединению.

Благодаря описанным программно-аппаратным стендам можно получить реальные

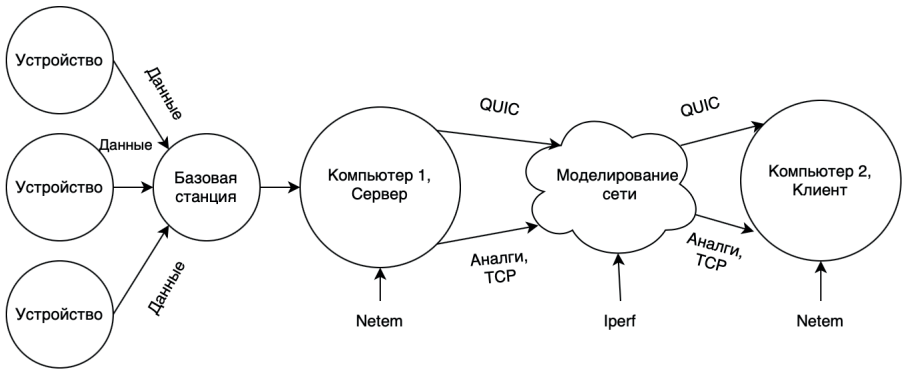


Рис. 6. Схема работы второго стенда в условиях передачи данных в реальных условиях с применением системы интернета вещей

практические данные о работе протокола QUIC, на основе которых можно сделать вывод о возможном применении протокола QUICK для систем интернета вещей и о возможной замене TCP и других протоколов. На основе результатов работы, можно будет продемонстрировать реальную практическую значимость в условиях стенда умных парковок. Алгоритм работы будет заключаться в следующем: при приезде нового автомобиля будет срабатывать ультразвуковой датчик-дальномер и сообщать об изменении расстояния от потолка до пола, что символизирует занятость парковочного места. Данная информация будет отправляться на сервер и с сервера поступать на мобильное устройство пользователя. Именно в этом момент при передаче данных конечному пользователю будет возможность применить протокол QUIC.

Заключение

В настоящей работе был сделан обзор работы протокола QUIC, сравнивалась работа данного протокола с протоколом TCP, дана оценочная характеристика полученным результатам. Были выявлены недостатки, в результате чего сделан вывод о необходимости проведения дополнительных экспериментов на программно-аппаратных стендах для оценки работы технологии в системах интернета вещей. Также был продемонстрирован план будущей работы, нацеленной на подтверждение или опровержение о возможности внедрения протокола QUIC для работы с данными малого размера.

Библиографический список

1. QUIC:Design Document and Specification Rationale. 2021. URL: https://docs.google.com/document/d/1RNHkxVvKWyWg6Lr8SZ-saqsQx7rFV-ev2jRFUoVD34/mobilebasic (дата обращения: 05.12.2021).

2. Что такое SPDY? Как развернуть SPDY? 2012. URL: https://www.oschina.net/news/29099/what-is-spdy (дата обращения: 05.12.2021).

3. *Khalifeh A., Mansour M., Alouneh S.* QUIC transmission protocol: Test-bed design, implementation and experimental evaluation // Journal of Electrical Engineering. 2021. Vol. 72, № 1. P. 20–28.

4. *McCullagh P., Nelder J.* Generalized linear models. Boca Raton: Chapman & Hall, 1999.

5. *Hablinger G., Hohlfeld O.* The Gilbert-Elliott model for packet loss in real time services on the Internet, Measuring // Modelling and Evaluation of Computer and Communication Systems (MMB): Proceedings 14th GI/ITG Conference on Measurement, March 31 – April 2, 2008, Dortmund, Germany. URL: https://www.researchgate.net/publication/221440836_The_Gilbert-Elliott_Model_for_Packet_Loss_in_Real_Time_Services_on_the_Internet (дата обращения: 05.12.2021).

УДК 004.056.53 **DOI: 10.31799/978-5-8088-1701-2-2022-2-183-190**

А. В. Воронов*
кандидат технических наук, доцент
В. Г. Ерышов*
кандидат технических наук, доцент
В. С. Коржук*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

РЕЗУЛЬТАТЫ ПРИМЕНЕНИЯ НЕЙРОННЫХ СЕТЕЙ ПРИ ПРОВЕДЕНИИ СПЕЦИАЛЬНЫХ ПРОВЕРОК ТЕХНИЧЕСКИХ СРЕДСТВ ОБРАБОТКИ ИНФОРМАЦИИ

Рассмотрены результаты применения двух разных моделей нейронных сетей при проведении специальных проверок технических средств обработки информации, проведено их сравнение.
Ключевые слова: *информационная безопасность, специальные проверки технических средств обработки информации, нейронные сети, практические результаты.*

A. V. Voronov*
PhD, Tech., Associate Professor
V. G. Eryshov*
PhD, Tech., Associate Professor
V. S. Korzhuk*
Student
**St. Petersburg State University of Aerospace Instrumentation*

RESULTS OF THE APPLICATION OF NEURAL NETWORKS IN AD HOC INSPECTIONS OF TECHNICAL MEANS OF INFORMATION PROCESSING

The article presents the application results of two different neural network models for ad hoc inspections of technical means of information processing and compares these results.
Keywords: *information security, ad hoc inspections of technical means of information processing, neural networks, practical results.*

Введение

Обладатель информации, оператор информационной системы (ИС), в целях обеспечения защиты информации (ЗИ) обязан защищать ее от несанкционированного доступа, уничтожения, модификации, блокирования, копирования в соответствии с нормативными правовыми актами Российской Федерации. Это достигается комплексом организационно-технических мероприятий, в том числе по защите ее по техническим каналы утечки информации (ТКУИ).

Один из основных способов обнаружения, идентификации и устранения ТКУИ в технических средствах обработки информации (ТСОИ) ИС – специальная проверка (спецпроверка, СП), которая проводится путем исследования ТСОИ при помощи специального контрольно-измерительного оборудования.

Одни из важнейших этапов проведения СП – рентгенография составных частей ТСОИ и последующий анализ полученного снимка с целью обнаружения закладных устройств внутри элементов исследуемого ТСОИ [1]. При этом возможны ошибки экспертов, приводящие к серьезным рискам необнаружения потенциально установленных закладных устройств в элементы ТСОИ. В связи с этим возникает задача разработки дополнительного программного обеспечения, позволяющего повысить эффективность как работы экспертов, так и СП в целом. Для решения такого рода задачи может использоваться аналитическая система анализа изображений на базе нейронной сети (нейросети).

В ранее проведенных теоретических исследованиях [2] было выявлено и обосновано, что для решения указанных задач наиболее подходят сверточная и генеративно-адаптивная мо-

дели нейронных сетей с применением алгоритмов обучения с учителем и частичным его привлечением соответственно. На основе полученных выводов было решено создать два программных модуля, реализующих работу как самих нейросетей, так и алгоритмов их обучения, схемы работы которых изображены на рис. 1, 2.

В рамках проведения СП в соответствии с ранее разработанным предложением нейронная сеть будет работать с изображениями сним-

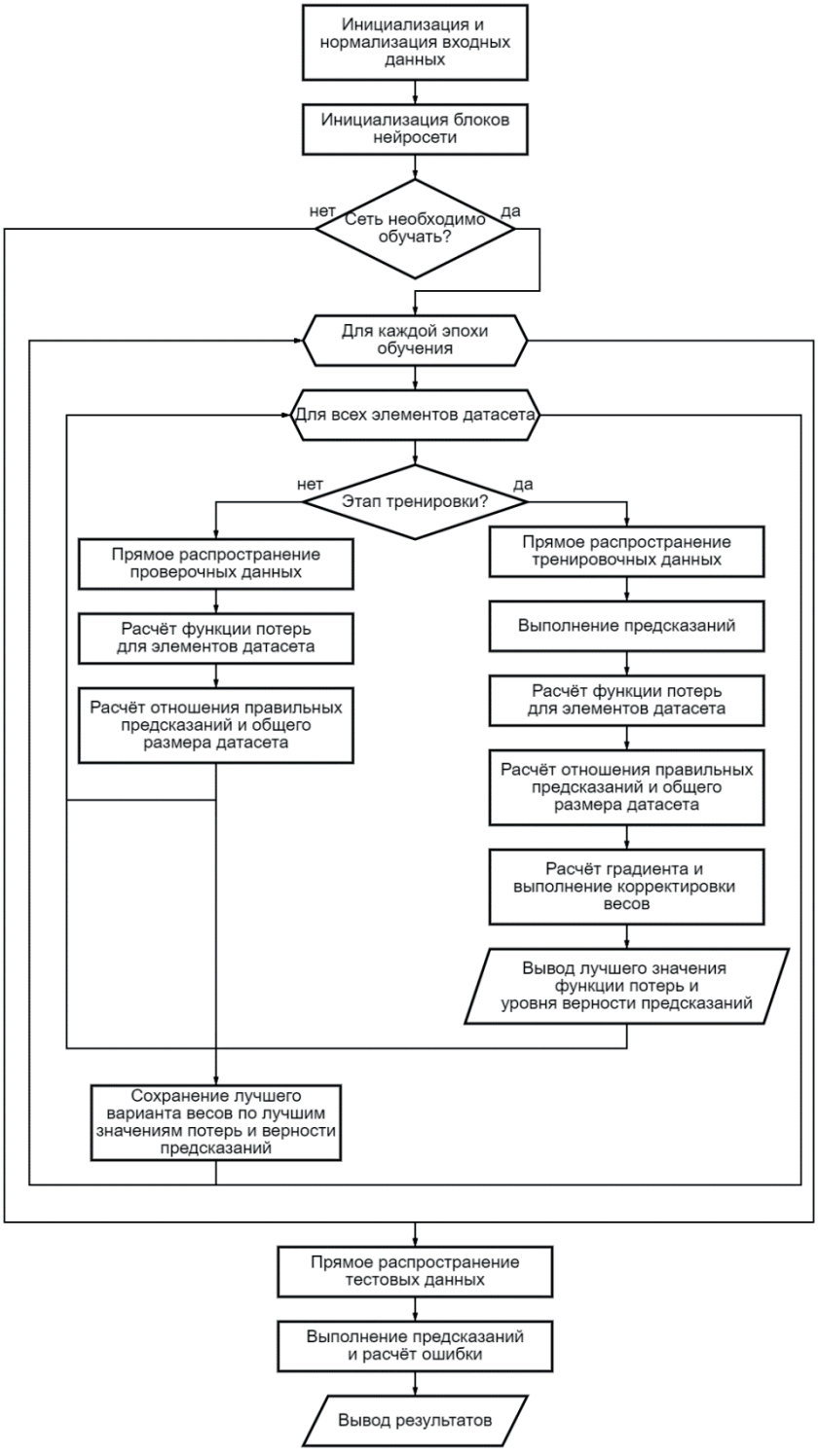


Рис. 1. Схема работы реализации сверточной нейросети

Рис. 2. Схема работы реализации генеративно-адаптивной нейросети

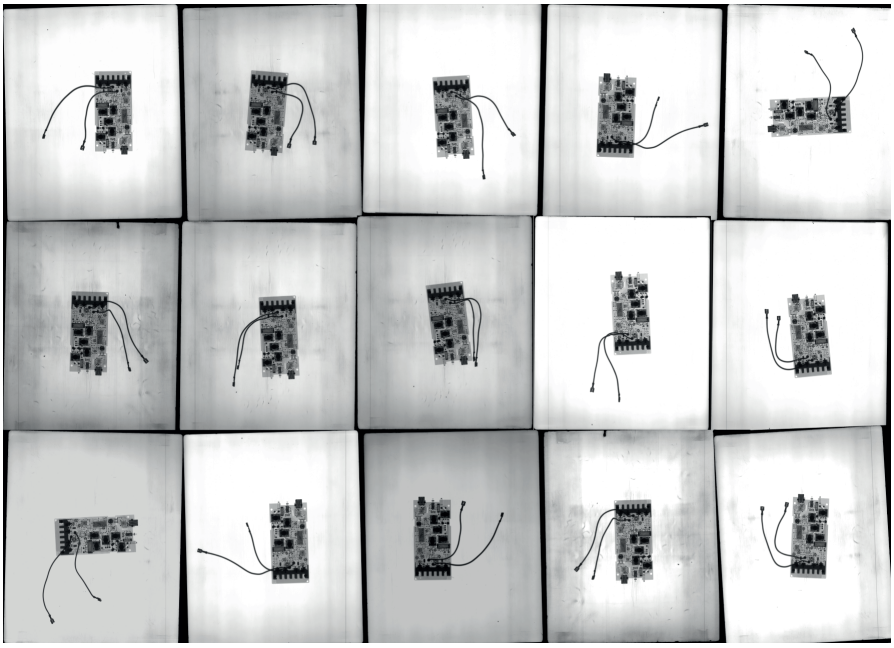


Рис. 3. Набор рентгенологических снимков плат питания

ков различных элементов ТСОИ, а разные экземпляры элементов, имеющих одинаковую модель и соответствующих своему эталону, на снимке будут иметь одинаковую структуру. Различия между изображениями будут обусловлены только качеством снимка (резкостью, яркостью, зашумленностью) и положением исследуемого элемента на нем. Это значит, что нет необходимости делать большое количество снимков элементов одной и той же модели, ведь их можно сгенерировать программным образом на основе небольшого количества оригиналов.

В качестве исходных данных для создания обучающего датасета использованы 15 плат питания одинаковой модели от источников бесперебойного питания. Данные платы были просвечены при помощи специализированного рентгенологического оборудования, после чего их снимки были отсканированы и получены изображения рентгенологических снимков (рис. 3).

При обучении нейронных сетей была использована видеокарта «NVIDIA GeForce 1050Ti» с объемом выделенной памяти графического процессора 4 Гб, который будет главным техническим ограничителем в нашей работе. Для обучения сама модель нейронной сети и обучающий датасет должны быть загружены в память видеокарты. Однако это позволит значительно увеличить скорость обучения в сравнении с CPU.

Поскольку используемые вычислительные мощности ограничены, а изображения рентгенологических снимков имеют слишком высокое разрешение (7100×8500), было принято решение тестировать разработанные программные модули на небольшом участке изображения платы с разрешением 512×512 (рис. 4).

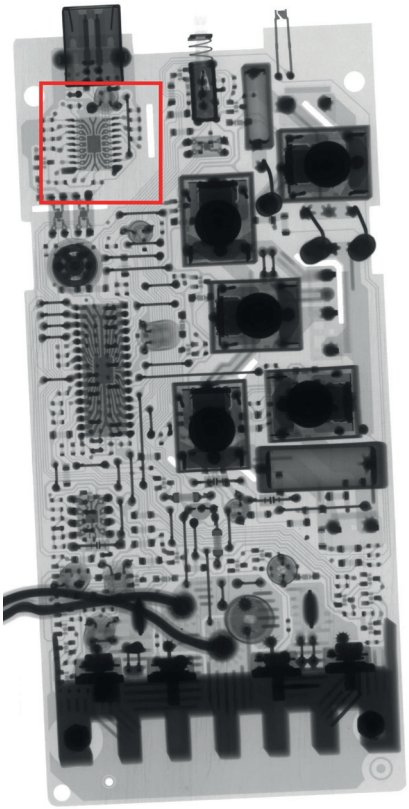


Рис. 4. Участок схемы, выбранный для формирования итогового обучающего датасета

Сверточная нейронная сеть		GAN
480 изображений		
160 изображений. Тренировочные. Соответствуют эталону.	**90** изображений. Проверочные. Соответствуют эталону.	160 изображений. Соответствуют эталону
160 изображений. Тренировочные. Модифицированные.	**90** изображений. Проверочные. Модифицированные.	

Рис. 5. Размеры обучающих датасетов

На основе участков, подобных обозначенному и полученных из 15 исходных изображений, были сформированы две базы обучающих данных, размеры которых представлены на рис. 5.

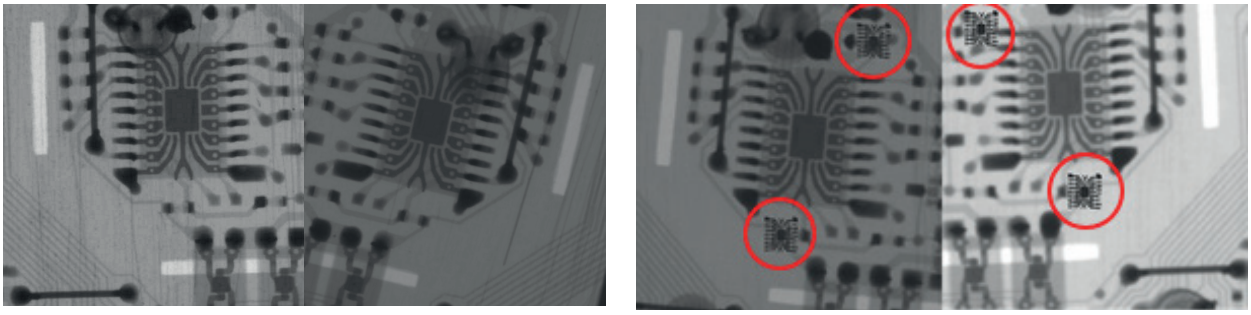
Как можно заметить, для обучения генеративно-адаптивной нейронной сети потребовалось практически в 3 раза меньше данных. Это связано с тем, что модифицированные и проверочные изображения она создает для себя сама.

Изображения, соответствующие эталону, были сгенерированы из 15 исходных изображений путем их поворота, внесения зашумления, размытия и т. д. Изображения же модифицированных плат формировались посредством внесения графической имитации закладного устройства. Примеры изображений представлены на рис. 6.

На рис. 6, *б* отчетливо видны внесенные нами модификации. Подобные присутствуют и на остальных изображениях из нашей базы, не соответствующих эталону.

В связи с тем, что размер изображений для обучающего датасета производился случайным образом, далее был произведен поиск максимального разрешения, с которым способны работать реализованные нами алгоритмы обучения нейронных сетей с учетом доступных вычислительных ресурсов. При этом на этапе инициализации и нормализации данных в программных модулях изображения подвергались операции масштабирования. По результатам экспериментов было определено, что для сверточной нейронной сети максимальным является разрешение 1024×1024, а для GAN – 128×128. При работе в этих условиях ресурсы видеокарты были задействованы практически на максимум (рис. 7, 8).

Такая сильная разница в максимальном возможном разрешении связана с более массивной моделью, а также большими потребностями в ресурсах на этапе обучения у GAN-сети. Помимо этого, в сверточной сети мы используем переобученную архитектуру сверточных слоев, которые были заранее натренированы на базе данных ImageNet и способны отлично выделять признаки на изображениях [3]. Поэтому, по сути, мы проводим обучение только пары полносвязных слоев, которые от-



а) *б)*

Рис. 6. Примеры элементов обучающего датасета: а – изображения участков, соответствующие эталонным, б – изображения участков, имитирующие наличие закладных устройств

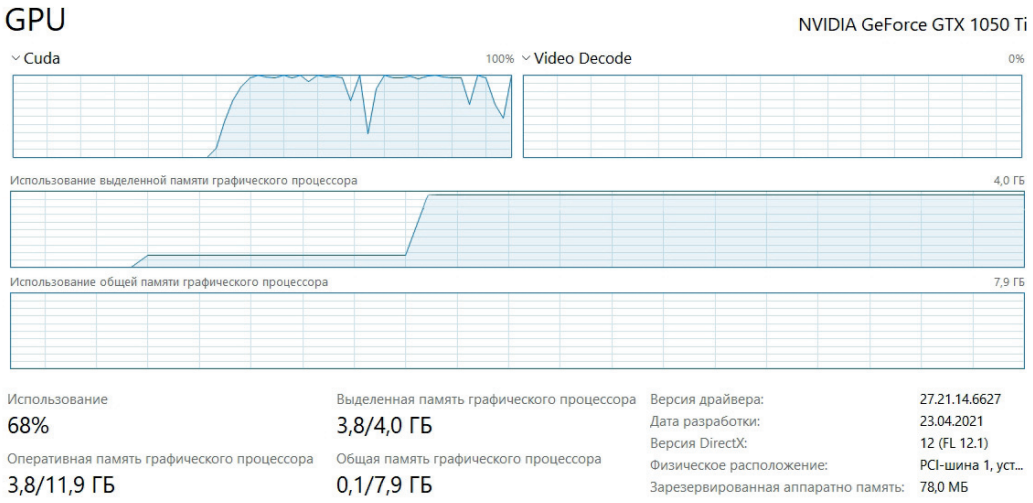


Рис. 7. Использование ресурсов при обучении GAN на изображенияхс разрешением 128×128

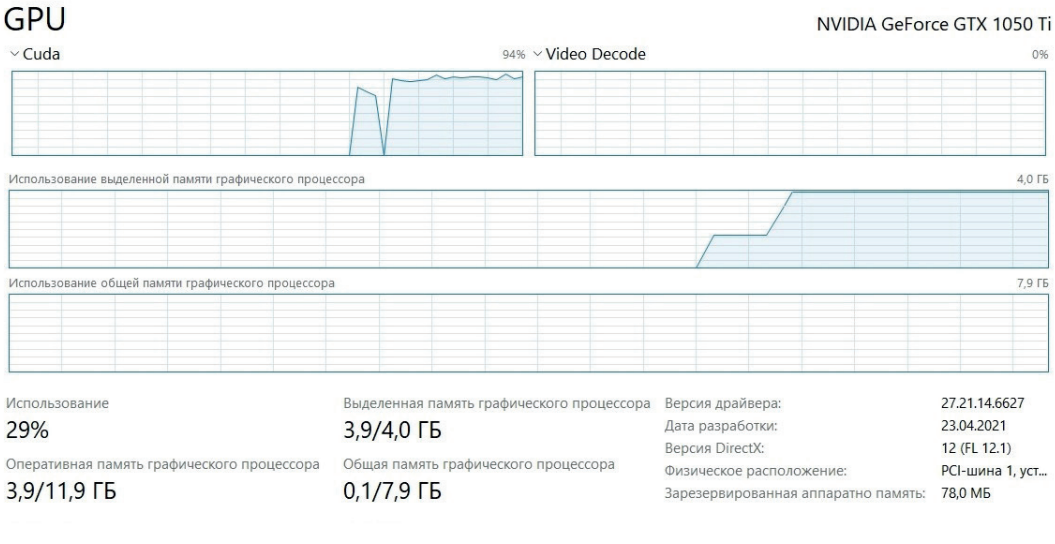


Рис. 8. Использование ресурсов при обучении сверточной нейросети на изображениях с разрешением 1024×1024

вечают непосредственно за классификацию изображения на основе полученных карт признаков. Это значительно повышает точность, скорость обучения и снижает потребности в ресурсах. Использовать подобную хитрость при реализации программного модуля для GAN-сети не удалось, из-за чего пришлось обучать с нуля все слои как генератора, так и классификатора, что привело к высоким потребностям в объеме памяти.

После определения максимального размера изображений для каждого из программных модулей было проведено обучение нейронных сетей. Экспериментальным путем было выбрано количество эпох обучения для получения лучших результатов и отсутствия эффекта переобучения. По итогу обучения были получены результаты (табл. 1).

Таблица 1

Результаты обучения выбранных моделей нейронных сетей

Характеристика	Сверточная сеть	GAN
Количество эпох	25	40
Время обучения	25 мин 16 с	10 мин 26 с
Время обучения на один пиксель изображения	1,45 мс	38,2 мс
Точность классификации	98,6 %	67,3%
Итоговое значение функции потерь	0,0235	0,3353

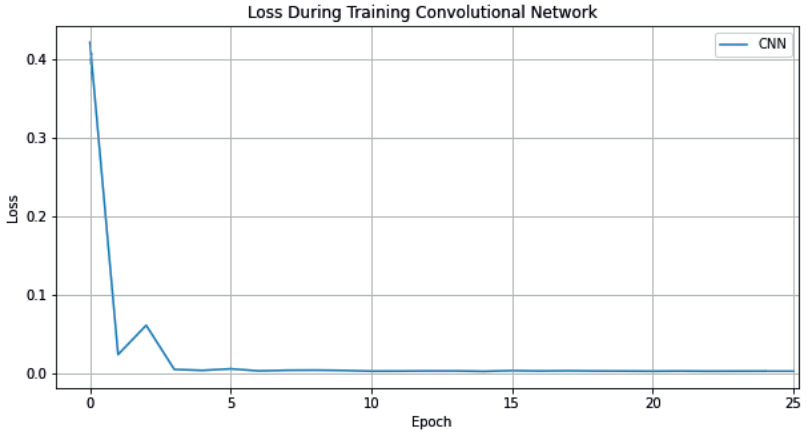


Рис. 9. Значение функции потерь при обучении сверточной нейросети

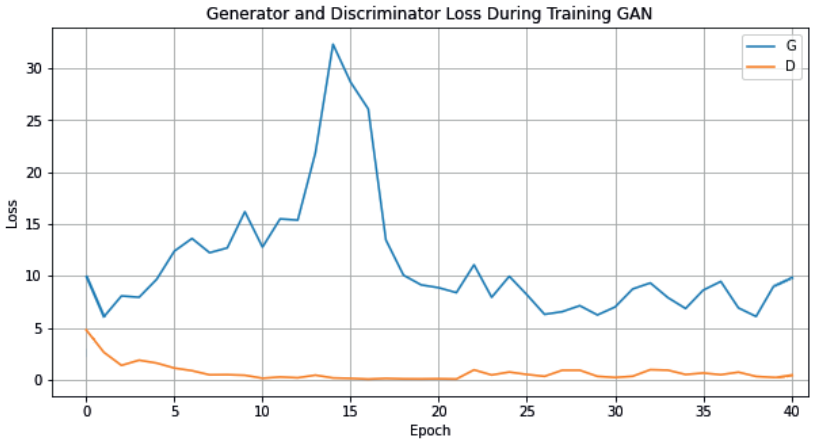


Рис. 10. Значение функции потерь генератора и дискриминатора при обучении GAN

Также были построены графики зависимости значения функции потерь от эпохи обучения, отражающие уровень обученности сети и приближенности ее результатов к желаемым (рис. 9, 10).

Из анализа рис. 9, 10 можно видеть, что для использования сверточной нейросети и метода обучения с учителем требуется достаточно трудоемкий и долгий процесс формирования обучающей выборки, данная сеть серьезно превосходит GAN по точности предсказаний, меньшим значениям функции потерь и размеру анализируемого изображения. Проблема низкой точности генеративно-адаптивной сети может быть связана с тем, что при обучении приходится соблюдать баланс между дискриминатором и генератором [4]. Генератор плохо справляется с воссозданием изображений, обладающих большим количеством мелких деталей, поэтому приходится придерживать обучение дискриминатора, из-за чего снижается итоговая точность работы сети.

После обучения моделей была проведена проверка их работы на тестовой выборке данных. В ходе проверки проводился подсчет количества правильно классифицированных изображений, и времени, затраченного на процесс классификации. Для тестирования было сгенерировано 60 изображений, половина из которых не соответствуют эталонным образцам, а графические имитации закладных устройств, нанесенные на них, отличались от тех, которые использовались в обучающей выборке. Пример таких изображений представлен на рис. 11.

Результаты, полученные по итогу тестового запуска, отражены в табл. 2.

Как и предполагалось, исходя из теоретических показателей, полученных в результате обучения, сверточная сеть показала очень хорошие результаты, допустив всего две ошибки классификации, генеративно-адаптивная сеть совершила 17 ошибок. Это составляет 1,7% и 28,4% от общего числа тестовых образцов, что мало отличается от теоретических значений точности обеих сетей.

Стоит обратить внимание на скорость классификации. Эксперт на качественную проверку

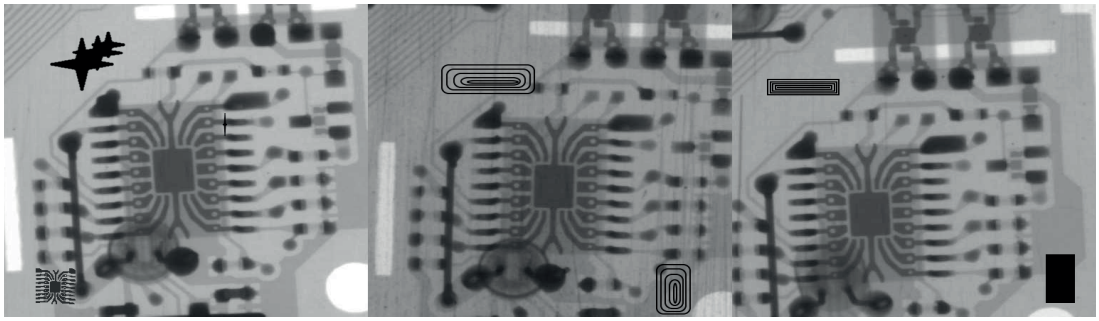


Рис. 11. Тестовые изображения с платами, не соответствующими эталонным вариантам

Таблица 2

Результаты проверки работы программных модулей

Характеристика	Сверточная сеть	GAN
Опознанные эталонные платы	30	27
Опознанные модифицированные платы	29	16
Скорость проверки	35 с 110 мс	5 с 620 мс
Расчетная скорость проверки снимков 3500×2500	7 мин 1 с	52 мин 27 с

одного снимка, аналогичного нашему, тратит в среднем около минуты. При этом данное время может увеличиваться в зависимости от уровня усталости, который также повышает вероятность ошибки. Реализованные нами программные модули справляются с работой быстрее человека, однако показатель скорости генеративно-адаптивной сети отличается от скорости работы человека незначительно и существенно ниже скорости работы сверточной нейросети. Это связано с тем, что для анализа целого снимка, его необходимо будет разбить на фрагменты, соответствующие максимальному размеру изображения, с которым способен работать тот или иной программный модуль. Для GAN необходимо будет проанализировать 560 таких фрагментов, что обесценивает высокую скорость проверки одного фрагмента.

Выводы

По итогам практического исследования выбранных архитектур нейронных сетей и алгоритмов их обучения были получены исчерпывающие результаты. Программные модули, реализованные нами, не способны работать с рентгенологическим снимком целиком. В реальном применении его необходимо было бы разбивать на фрагменты максимально допустимого для каждого из программных модулей размера и анализировать частями либо значительно увеличить вычислительные ресурсы в процессе обучения. Несмотря на это, как сверточная нейронная сеть с применением метода обучения с учителем, так и генеративно-адаптивный алгоритм на основе СНС, обучаемый с частичным применением учителя, способны показывать неплохие результаты при их использовании в качестве вспомогательного инструмента для анализа изображений при проведении специальных проверок технических средств. Оба алгоритма позволяют упростить и ускорить рабочий процесс. Однако GAN требует больше вычислительных ресурсов и времени для обучения, и ее использование имеет смысл только в случае отсутствия возможности программно сгенерировать обучающую выборку необходимого размера либо если вероятность встретить закладное устройство в проверяемом техническом средстве крайне мала.

Библиографический список

1. Специальная проверка техники (СП). URL:https://www.apsecurity.ru/services/special-inspections.html (дата обращения: 18.03.2021).

2. *Коржук В.* Исследование возможности применения моделей нейронных сетей для специальных проверок технических средств // Семьдесят четвертая Международная студенческая научная конференция ГУАП: сб. докл.: в 4 ч. Ч. 2. Технические науки. СПб.: ГУАП, 2021. 190 с.

3. *Кузнецов А.* Кластеризация изображений при использовании предобученных нейронных сетей // International journal of open information technologies. 2019. Vol. 4, № 7. P. 42–47.

4. Устойчивость обучения GAN. URL: https://habr.com/ru/post/416531/ (дата обращения: 10.10.2021).

УДК 004.728 **DOI: 10.31799/978-5-8088-1701-2-2022-2-191-195**

А. Ю. Глушенкова*
студент
В. Ю. Михайлов*
студент
А. М. Тюрликов*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОБЗОР СПОСОБОВ УМЕНЬШЕНИЯ ЗАДЕРЖКИ В КАНАЛЕ СЛУЧАЙНОГО МНОЖЕСТВЕННОГО ДОСТУПА В СОВРЕМЕННЫХ СОТОВЫХ СЕТЯХ

Рассмотрены методы уменьшения задержки передачи данных в сотовых сетях с помощью улучшения процедуры произвольного доступа путем модификации физического и канального уровней доступа к среде.
Ключевые слова: *сотовые сети, задержка передачи, процедура произвольного доступа, преамбулы.*

A. Yu. Glushenkova*
Student
V. Yu. Mikhaylov*
Student
A. M. Turlikov*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

OVERVIEW OF WAYS TO REDUCE THE DELAY IN THE RANDOM MULTIPLE ACCESS CHANNEL IN MODERN CELLULAR NETWORKS

The methods of reducing the data transmission delay in cellular networks by improving the random access procedure, by modifying the physical and channel levels of access to the environment are considered.
Keywords: *cellular networks, transmission delay, random access procedure, preamble.*

В современных сотовых сетях для подключения абонента и дальнейшей передачи данных используется процедура произвольного доступа (Random Access Channel (RACH)). Она может быть двух- или четырехэтапной. В статье будет рассмотрена четырехэтапная процедура произвольного доступа, далее будем называть ее стандартной. В статье [1] описана стандартная процедура подключения абонента к сети, рассмотрим ее (рис. 1).

Шаг 1. Передача преамбулы. Абонент случайным образом выбирает преамбулу из общего числа преамбул и отправляет ее на базовую станцию в слоте физического канала произвольного доступа (Physical Random Access Channel (PRACH)). На практике возможна ситуация, что несколько абонентов выберут одинаковую преамбулу.

Шаг 2. Ответ случайного доступа. Базовая станция отправляет ответ случайного доступа, который состоит из идентификатора обнаруженной преамбулы, информации временного опережения и начального ресурса физического канала передачи данных по восходящей линии (Physical Uplink Shared Channel (PUSCH)). Эти

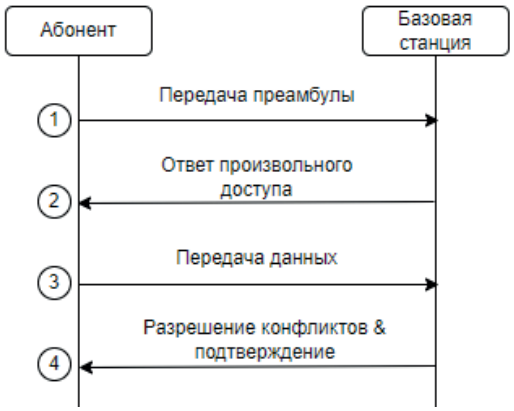


Рис. 1. Схема RACH-процедуры

данные используются для передачи данных по восходящей линии.

Шаг 3. Передача данных от абонента. Используя ресурс PUSCH, указанный в ответе произвольного доступа, абонент передает пакет данных по восходящей линии, запрос на соединение управления радиоресурсами, обновление области отслеживания или планирование.

Шаг 4. Передача сообщения о подтверждении. Если базовая станция успешно декодирует данные, переданные одним абонентом, то она отправляет обратно сообщение о подтверждении, которое содержит идентификатор абонента, полученный из декодированных данных. В обратном случае базовая станция ничего не передает, таким образом уведомляя абонента о конфликте при выборе преамбул, поскольку абонент ожидает ответа в течение некоторого интервала времени. В такой ситуации абонент повторяет процедуру подключения.

В стандартной процедуре RACH возникает ситуация, когда два или более абонента случайным образом выбирают идентичную преамбулу и передают ее в одном и том же слоте PRACH на первом этапе. Такую ситуацию мы будем называть конфликтом преамбул. Если это так, абоненты получают такое же предоставление ресурса восходящей линии связи в сообщении ответа произвольного доступа на втором этапе. Затем абоненты передают свои данные на одном и том же ресурсе PUSCH на третьем этапе, и таким образом возникает конфликт ресурсов PUSCH. Следовательно, декодирование данных может завершиться неудачно, и это позволяет базовой станции распознавать конфликт преамбулы на третьем этапе и уведомлять о конфликте преамбул соответствующих абонентов на четвертом этапе. Этот типичный метод обнаружения коллизий преамбулы очень неэффективен с точки зрения задержки произвольного доступа и использования ресурсов PUSCH. Таким образом, необходимы расширенные методы обнаружения конфликтов преамбулы и уведомления об их обнаружении.

В настоящее время ведутся поиски по улучшению стандартной процедуры на физическом уровне и путем изменения процедуры разрешения конфликта [1–4]. Ученые предлагают различные методы уменьшения задержки передачи данных. Для улучшения физического уровня необходимо изменять первый шаг стандартной процедуры, с этой целью предлагается [1] передавать абоненту дополнительную последовательность Задова – Чу, по которой базовая станция может узнать дополнительную информацию о числе абонентов. Для улучшения канального уровня требуется изменять алгоритм разрешения конфликта, предлагается использовать модифицированный древовидный алгоритм разрешения конфликта [3]. Рассмотрим подробнее указанные подходы.

Улучшение физического уровня возможно путем обнаружения конфликта преамбул на первом шаге процедуры RACH. Рассмотрим генерацию преамбул [5]:

$$z_r[n] = \exp(-j\pi rn(n+1)/N_{ZC}),$$

где N_{zc} – длина последовательности Задова – Чу (в сотовых сетях 5-го поколения N_{zc} равна 139 или 839, что соответствует преамбулам коротких и длинных форматов. В сетях 4-го поколения используются только длинные форматы); $n = 0, \ldots, N_{zc}$, r – корневое число, взаимно простое с N_{zc}.

В сотовых сетях используется 56 и 64 преамбулы для 4- и 5-го поколения соответственно. Преамбулы генерируются исходя из конфигурации системы путем циклического сдвига последовательностей Задова – Чу.

В статье [1] предложен метод обнаружения факта наличия конфликта среди абонентов, передававших преамбулу с номером i. Авторы статьи предлагают вместе с выбранной преамбулой передавать последовательность Задова – Чу, которая генерируется на основе другого корневого числа для i-й преамбулы. Абоненты передают эту последовательность с различными сдвигами. Таким образом, абоненты, выбравшие преамбулу с номером i, передают одинаковую преамбулу, но разные последовательности Задова – Чу.

На приемной стороне базовая станция вычисляет корреляцию между опорными преамбулами и опорными последовательностями Задова – Чу. Таким образом, базовая станция обнаружит только одну преамбулу и несколько последовательностей Задова – Чу. По этим данным базовая станция определяет, что произошел конфликт. Возможна ситуация, когда абоненты выберут одинаковые последовательности Задова – Чу и базовая станция не обнаружит конфликт, но ее вероятность очень мала.

Описанная схема позволяет обнаружить преамбулы уже на втором шаге процедуры RACH. В результате можно уменьшить задержку подключения и устранить потери выделяемых ресурсов PUSCH.

В статье [2] предложены методы машинного обучения для разработки новой структуры подключения, комбинирующей ортогональную и неортогональную передачу. Описываемые в [1]

подходы используются для обучения нейронной сети, чтобы точно предсказать число конфликтующих абонентов. На основе знания о числе абонентов, участвующих в конфликте, выделяется эксклюзивный ресурс PUSCH для ортогональной передачи, если конфликты отсутствуют, либо одинаковый ресурс PUSCH для всех конфликтующих абонентов для неортогональной передачи. Методы, приведенные в статье, позволяют уменьшить задержку передачи и определить количество абонентов, участвующих в конфликте.

Теперь перейдем к описанию процедур разрешения конфликта. Улучшение процедуры разрешения конфликта в числе прочего может быть основано на древовидном алгоритме, впервые предложенном Б. С. Цыбаковым и В. А. Михайловым в 1978 г. [6], а также Д. Капетанакисом в 1979 г. [7]. В данном алгоритме все абоненты следят за выходом системы и при возникновении конфликта строят так называемое дерево разрешения конфликта, в соответствии с которым принимается решение о действии в данном окне. Эти алгоритмы все еще исследуются и улучшаются, например в работе [8] рассматривается применение древовидного алгоритма вкупе с технологией физического уровня Successive Interference Cancellation. Труды [6–8] имеют скорее теоретический характер и не учитывают особенности организации случайного доступа в сотовых сетях.

В работе [3] предлагается динамический древовидный алгоритм (Dynamic Tree-Splitting (DTS)) для эффективного разрешения конфликтов случайного множественного доступа путем разделения столкнувшихся абонентов на несколько групп, чтобы уменьшить количество конкурирующих абонентов в каждом конкурирующем кадре. Конфликтом преамбулы считается ситуация, если она была отправлена более чем одним абонентом, даже если она была успешно декодирована базовой станцией на шаге 1 процедуры произвольного доступа. Поэтому предполагается, что абоненты, столкнувшиеся в шаге 3, уже считались столкнувшимися в шаге 1. Таким образом, абоненты, выбравшие одну и ту же преамбулу, направляются в определенную группу преамбул для следующей попытки доступа. Столкнувшиеся абоненты получают информацию о следующей попытке доступа в сообщении обратной связи после каждой попытки доступа. Для этой цели в данной статье используется новый тип сообщения на шаге 4, обозначаемый как «Шаг 4б». Обычно в стандартной процедуре произвольного доступа сообщение шага 4 отправляется абонентам, успешно передавшим преамбулы, информируя их о ресурсах восходящей линии связи, которые будут использоваться для их передачи данных. Однако предложенный шаг 4б является альтернативой шага 4, и в нем при конфликтах в преамбулах столкнувшимся абонентам отправляется информация о времени и группе преамбул, которые будут использоваться при следующей попытке доступа. Шаг 4б обрабатывается аналогично шагу 4 и занимает то же время обработки, обычно 5 мс в соответствии со стандартной конфигурацией RACH. Следовательно, задержка столкнувшихся абонентов не увеличивается, так как они получают сообщение обратной связи шага 4б после каждой попытки доступа, в то время как успешные абоненты получают сообщение шага 4.

Основная цель DTS – оптимизировать количество преамбул в каждой группе в соответствии с интенсивностью столкновений каждой возможности распределения абонентов по преамбулам. В англоязычной литературе для описания распределения абонентов по преамбулам используется термин Random Access Opportunity (RAO), далее в тексте для краткости будем использовать сокращение RAO. Динамическое распределение преамбул отличается от подходов в предшествующих работах со смежной темой, где количество групп преамбул определяется в каждом RAO без учета интенсивности столкновений, что определенно влияет на использование преамбул и в свою очередь на общую производительность RACH. В этой статье количество групп преамбул G в каждом RAO динамически определяется на основе количества преамбул M и ветвей b, назначенных каждой группе: $G_i = M/b_i$. Количество ветвей b_i для каждого RAO изменяется динамически на основе коэффициента столкновения CC_i i-го RAO, в котором было инициировано столкновение:

$$CC_i = \frac{\text{Количество столкнувшихся абонентов}}{\text{Количество преамбул с конфликтами}}.$$

Мы пытаемся достигнуть соответствия количества ветвей b и среднего числа абонентов, выбравших данную преамбулу. Такое соответствие позволит увеличить вероятность успеха в следующем RAO.

На рис. 2 изображен пример работы алгоритма при 6 преамбулах (M) и 15 абонентах (N), пытающихся подключиться к системе. Каждый слот на рисунке обозначает отдельную преамбулу, числа в слотах – количество абонентов, выбравших данную преамбулу. В примере на первом уровне дерева ($k = 1$) конфликты прои-

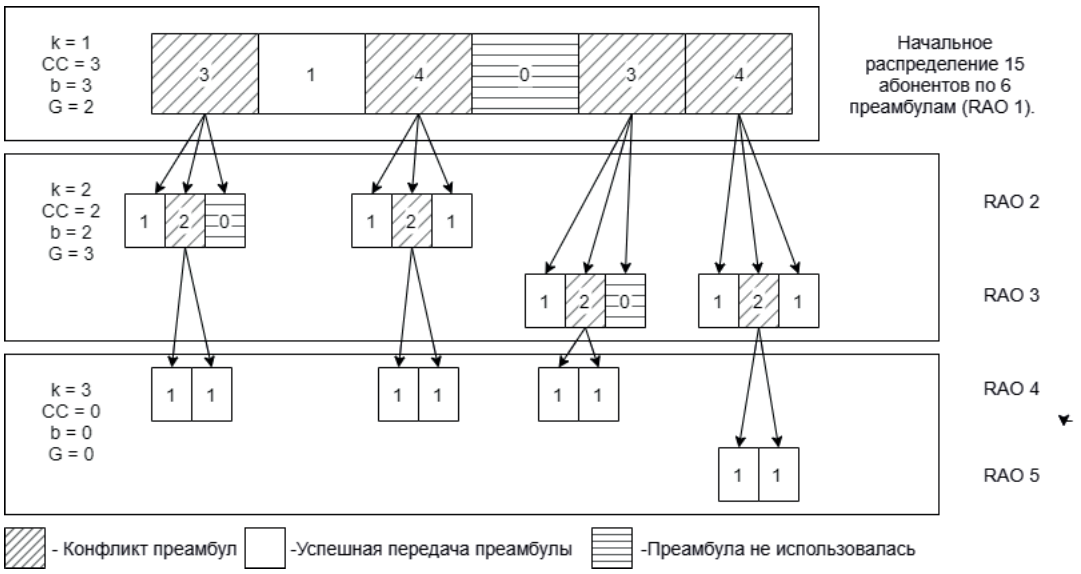


Рис. 2. Пример работы динамического древовидного алгоритма для 6 преамбул (M) и 15 абонентов (N)

зошли в четырех преамбулах, и, применяя формулу для *CC*, получим $b = \lfloor 14/4 \rfloor = 3$. Значит, каждая из столкнувшихся преамбул инициирует три ветви для следующей попытки доступа, а используя $G = 6/3 = 2$, выясняем что две группы преамбул (по три в каждой) будут использованы в следующем RAO, оставшиеся переносятся в RAO после этого. На уровне дерева $k = 2$ расчеты проводим заново и получаем $b = \lfloor 8/4 \rfloor = 2$, $G = 6/2 = 3$, т. е. три группы по две преамбулы в следующем RAO. Процесс повторяется, пока все конфликты не будут разрешены, либо k не достигнет некоторого параметра системы $k_{\max}$, и все неразрешенные попытки будут сброшены.

Представленный алгоритм позволяет увеличить использование ресурсов RACH, что в свою очередь повышает скорость передачи данных в RACH и уменьшает задержки подключения.

Заключение

Были рассмотрены два направления по улучшению физического и канального уровня процедуры произвольного доступа, в данный момент развивающиеся параллельно. Достоинство улучшения физического уровня состоит в возможности обнаружения конфликта преамбул на первом шаге, тогда базовая станция не будет лишний раз выделять ресурсы канала для передачи. Однако для улучшения физического уровня требуется изменить схему обработки принятого сигнала, что является недостатком данного подхода. Достоинство улучшения канального уровня заключается в динамическом определении количества используемых преамбул для разрешения конфликта и уменьшении задержки подключения абонентов, недостаток – в необходимости ждать 3- и 4-й шаги процедуры. В дальнейшем представляет интерес объединение этих направлений, тогда получится использовать их сильные стороны и значительно улучшить процедуру RACH.

Библиографический список

1. *Jang H. S., Kim S. M., Park H., Sung D. K.* An Early Preamble Collision Detection Scheme Based on Tagged Preambles for Cellular M2M Random Access // IEEE Transactions on Vehicular Technology. 2017. Vol. 66, № 7. P. 5974–5984.

2. *Jang H. S., Lee H.,* T. Quek Q. S., Shin H. Deep Learning-Based Cellular Random Access Framework // IEEE Transactions on Wireless Communications. 2021. Vol. 20, № 11. P. 7503–7518.

3. *Althumali H., Othman M., Noordin N. K., Hanapi Z. M.* Dynamic Tree-Splitting Algorithm for Massive Random Access of M2M Communications in IoT Networks // IEEE Systems Journal. 2021. 2 august.

4. RACH Optimization with Decision Tree Based Supervised Learning for Conditional Handover in 5G Beamformed Systems / U. Karabulut, A. Awada, I. Viering et al. 2-019. URL: https://eprint arXiv:1910.11890 (дата обращения: 14.11.2021).

5. *Chu D.* Polyphase codes with good periodic correlation properties (Corresp.) // IEEE Transactions on Information Theory. 1972. Vol. 18, № 4. P. 531–532.

6. *Цыбаков Б. С., Михайлов В. А.* Свободный синхронный доступ пакетов в широковещательный канал с обратной связью // Проблемы передачи информации. 1978. Т. 14, вып. 4. С. 32–59.

7. *Capetanakis J.* Tree algorithms for packet broadcast channels // IEEE Transactions on Information Theory. 1979. Vol. 25, № 5. P. 505–515.

8. *Stefanović Č., Deshpande Y., Gürsu M., Kellerer W.* Tree-Algorithms with Multi-Packet Reception and Successive Interference Cancellation. 2021. URL: https://eprint arXiv:2108.00906 (дата обращения: 14.11.2021).

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
Г. А. Богоявленский*
студент
Н. И. Маралов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ТИПЫ И ОБЗОР СОВРЕМЕННЫХ АНАЛИЗАТОРОВ КОДА, ПРИМЕНЯЕМЫХ ДЛЯ ПОИСКА И ОПРЕДЕЛЕНИЯ УЯЗВИМОСТЕЙ В РАЗРАБАТЫВАЕМОМ ПРОГРАММНОМ ОБЕСПЕЧЕНИИ

Рассмотрены различные типы и приведен обзор современных анализаторов кода, приведены их преимущества и недостатки.
Ключевые слова: *анализ кода, определение уязвимостей, динамический и статический анализ кода.*

V. G. Eryshov*
PhD, Tech., Associate Professor
G. A. Bogoyavlensky*
Student
N. I. Maralov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

REVIEW OF CODE ANALYZERS FOR IDENTIFYING VULNERABILITIES IN DEVELOPED SOFTWARE

Various types are considered and an overview of modern code analyzers is given, their advantages and disadvantages are given.
Keywords: *code analysis, identification of vulnerabilities, dynamic and static code analysis.*

При разработке нового программного обеспечения (ПО) вполне вероятны умышленное внедрение опасного кода, недекларированные возможности (НДВ), неумышленные ошибки разработчиков в коде [1]. При этом оставляют желать лучшего уровень доверия и степень безопасности разрабатываемого ПО.

В процессе обеспечения функциональности ПО вопрос безопасности его кода нередко отходит на второй план. Как результат вместе с дополнительными функциями, новым модулем или нетиповой надстройкой можно приобрести уязвимости и недекларированные возможности. Обнаружение подобных угроз на этапе эксплуатации системы или приложения несет для компании большие финансовые и в отдельных случаях репутационные риски [2].

Для поиска и идентификации уязвимостей кода используются различные средства, например, **анализаторы исходного кода разрабатываемого ПО**. По способу обнаружения уязвимостей они разделяются на две группы: статические (**SAST** – static application security testing) и динамические (**DAST** – dynamic application security testing).

Статический анализ исходного кода происходит без запуска программы и часто даже без ее сборки. Анализатор исследует программный код, написанный разработчиком по заданным правилам, построчно или по блокам кода. Разумеется, такой анализ не позволит обнаружить все уязвимости, появившиеся в ходе разработки, так как некоторые из них скрыты и могут быть обнаружены только в ходе выполнения программы. Главное преимущество такого подхода – *скорость* выполнения анализа: часто утилиты статического анализа интегрируются в среду разработки и могут запускаться по ходу написания кода (например, после 5 с отсут-

ствия появления нового кода или после каждого сохранения файлов).

Так же, как другие средства, используемые при разработке, утилиты статического анализа могут включаться в процесс непрерывной интеграции [3]. Таким образом можно немедленно обнаруживать основную массу уязвимостей до их возможного появления в конечном продукте. Типы уязвимостей, выявляемых статическими анализаторами [4]:

– известные уязвимости безопасности;

– утечки памяти и других ресурсов;

– нарушение алгоритма использования встроенной библиотеки;

– неумышленные ошибки в разрабатываемом коде;

– вызов процедур, не имеющих смысла;

– выявление неэффективных конструкций;

– диагностические правила общего назначения (опечатки, формирование строк, избыточные условия if-else и т. п.).

Рассмотрим некоторые системы комплексного статического анализа.

SonarQube – платформа с открытым исходным кодом для непрерывного анализа многоязычных проектов и измерения качества программного кода. Анализируемый язык необходимо указывать явным образом. Измеряет качество программного кода в соответствии с 7 показателями качества программного обеспечения, которые разработчики называют Seven Axes of Quality [5]:

– потенциальные ошибки – неточности в программе, из-за которых программа выдает ошибки;

– стиль программирования – набор структур, используемых при написании кода на конкретном языке программирования;

– тесты – набор структур, применяемых при написании кода на конкретном языке программирования;

– комментарии – набор структур, применяемых при написании кода на конкретном языке программирования;

– архитектура – структура важнейших решений об организации и использовании программной среды;

– сложность – количество неких вычислительных ресурсов (обычно время ЦП), требуемое для работы алгоритма.

Преимущества данной системы:

– поддержка широко известных языков, таких как Java, C, C++, C#, Objective-C, Swift, PHP, JavaScript, Python и др.;

– предоставление отчетов о дублировании кода, соблюдении стандартов кодирования, покрытии кода модульными тестами, возможных ошибках в коде, плотности комментариев в коде, техническом долге и др.;

– сохранение истории метрик, отрисовка графиков изменений этих метрик во времени;

– возможность интеграции с распространенными IDE, CI, внешними инструментами управления проектами;

– функционал расширяется с помощью сторонних плагинов.

AppChecker – отечественный статический анализатор кода, предназначенный для поиска дефектов в исходном коде приложений, разработанных на C/C ++, C#, Java, PHP [6].

Функционал:

– поиск дефектов кодирования с помощью постоянно обновляемой базы правил и технологии анализа потока данных;

– поиск программных закладок.

Преимущества:

– поиск свыше 100 типов дефектов кодирования;

– постоянное пополнение базы правил поиска дефектов;

– поддержка классификации CWE (Common Weakness Enumeration – общий перечень уязвимостей);

– применение технологии анализа потока данных;

– возможность проводить совместный аудит кода несколькими экспертами;

– гибкая конфигурация анализируемых проектов;

AppChecker имеет графический интерфейс, который позволяет как отслеживать количество срабатываний по конкретному дефекту к общему числу срабатываний, производя статистический анализ и отображая эту информацию в графики процентного содержания, так и демонстрировать фрагменты кода пользователю, определяя тип уязвимости.

Далее рассмотрим статические анализаторы, специализированные и разработанные под конкретные языки программирования.

Cppcheck – статический анализатор кода для языка C/C++, используемый для обнаружения ошибок, которые не выявляются компиляторами языка программирования. Основная задача анализатора – минимизация количества ложных срабатываний в процессе поиска и обнаружения ошибок.

Функционал данного анализатора не ограничивается поиском критических ошибок. Возможны добавления предупреждений для поиска проблем производительности, совместимости, подозрительных мест программы и ошибок стиля про-

граммирования [7]. При использовании Cppcheck нет требований по использованию компиляторов, IDE, заголовочных файлов, что говорит о его «неприхотливости» в плане исходных данных.

Функционал:

– обнаружение возможных утечек памяти при выполнении кода;

– обнаружение неинициализированных переменных в коде;

– поиск и обнаружение устаревших и неиспользуемых функций и процедур;

– предупреждение о неиспользуемом (устаревшем) коде;

– поиск подозрительных участков кода, содержащих возможные уязвимости и ошибки.

До начала анализа кода **анализатор** активизирует препроцессор так же, как и компилятор. Затем следует очередной этап минимизации исходного кода: удаляются все лишние отступы и пробелы, любая лексическая конструкция кода разделяется пробелом. Вычисляются все константы. Во всех участках кода ставятся блоки “{}”, в том числе если они были не проставлены. В случае присутствия объявления или присвоения переменной внутри блока “if/for/while” оно выносится снаружи этой конструкции.

SpotBugs – анализатор статического анализа кода Java, который находит признаки возможных ошибок кода и уязвимостей безопасности. О найденных дефектах кода анализатор сообщает в предупреждениях (warnings). В последней выпущенной версии анализатора содержится более 400 предупреждений четырех категорий: наиболее опасные, опасные, серьезные, требующие внимания [8].

Преимущества:

– обнаружение в коде большого типа ошибок;

– возможность анализа программ, скомпилированных для любой версии Java;

– интеграция с Ant, Maven, Gradle;

– лицензия бесплатна, имеет открытый исходный код;

– создание отчета на основе обнаруженных ошибок.

На вход анализатору отдается директория или выбранный файл с байт-кодом. Для анализа байт-кода используется библиотека **Apache BCEL**. Непосредственно проверки реализованы в наследниках класса Detector.

Golangci-lint – утилита для статического анализа кода языка Go, известная своей скоростью выполнения анализа, достигаемой за счет механизмов кэширования, многопоточности, точного конфигурирования и позволяющей ее использование непосредственно во время написания кода [9].

Преимущества:

– скорость работы;

– количество плагинов, расширяющих функционал (в том числе gosec – анализ безопасности);

– открытый исходный код;

– интеграция со всеми основными средами разработки (VS Code, Sublime Text, GoLand, GNU Emacs, Vim, Atom);

– легкое включение в механизм непрерывной интеграции;

– создание отчетов в унифицированном формате для включения в общую отчетность систем непрерывной интеграции и поставки.

Динамический анализ. В отличие от статических анализаторов, которые в своем большинстве анализируют исходный код программы (в редких случаях байт-код или машинный код), динамические исследуют программный продукт в формате «черного ящика» (т. е. без знания архитектуры, конкретных программных решений и взгляда на исходный программный код), используя для поиска уязвимостей техники, схожие с поведением потенциального злоумышленника [10].

Типы выявляемых уязвимостей, ошибок:

– межсайтовый скриптинг – разновидность атак инъекции;

– SQL-инъекция – внедрение произвольных sql-запросов во время работы программы, взаимодействующей с базами данных;

– разглашение пути – атака, целью которой является выяснение абсолютных путей к файлам программы, исполняемому файлу и прочим ресурсам на сервере;

– утечки конфиденциальной информации;

– отказ в обслуживании или DOS (Denial of Service) – доведение информационной системы до состояния, в которой ей не хватит ресурсов на обработку новых запросов пользователей;

– выполнение кода злоумышленника на удаленном хосте (сервере);

– повреждение данных (памяти);

– переполнение буфера.

Netsparker – коммерческое автоматизированное решение для сканирования и анализа веб-уязвимостей, находит уязвимости независимо от архитектуры или платформы. Сканер генерирует доказательство эксплойта, подтверждающее, что он не является ложноположительным, что улучшает автоматизацию и масштабируемость [10].

Netsparker Enterprise предназначен для предприятий, которым требуется настраиваемое решение для сложных сред.

В зависимости от варианта и потребностей заказчика Netsparker может быть реализован

как настольное программное обеспечение, управляемая служба или локальное решение.

Функционал:

– уникальный механизм анализа уязвимостей;

– интеграция с существующей средой SDLC;

– высокоуровневый анализ на основе клиентских IP-адресов, доменов верхнего и второго уровней и информации о сертификатах SSL;

– функции анализа обхода аутентификации;

– отчетность содержит информацию об уязвимости, ее влияние, пути устранения и избежания ошибки в будущем;

– автоматическое покрытие уязвимости с высокой степенью воздействия, которые вы не можете исправить немедленно.

Positive Technologies – отечественный производитель средств DAST. Популярный продукт этого производителя – **PT Application Inspector**. Обнаруживает и локализует уязвимости и НДВ. В процессе работы данный анализатор проверяет исходный код, применяя как статические, так и динамические методы анализа. По итогам анализа PT он генерирует безопасные эффективные тестовые запросы, которые либо подтверждают, либо опровергают наличие уязвимостей или НДВ [11].

Особенности:

– проведение анализа веб-приложений и большого количества систем разработки;

– отечественный, сертифицированный программный продукт;

– применение базы уязвимостей из банка данных угроз ФСТЭК, WASC и классификатора CWE.

Заключение

В статье рассмотрены основные типы анализаторов исходного кода для определения уязвимостей в разрабатываемом программном обеспечении. Приведены современные анализаторы для различных часто используемых языков программирования. Начинающие разработчики или разработчики, желающие укрепить безопасность своего продукта, могут воспользоваться данным списком для быстрого ознакомления с методами и решениями с целью анализа безопасности разрабатываемого программного обеспечения.

Библиографический список

1. Поиск уязвимостей в программах с помощью анализаторов кода. URL: http://www.codenet.ru/progr/other/code-analysers.php (дата обращения: 25.11.2021).

2. *Горошко Л.* Обзор сканеров кода: взгляд интегратора. URL: https://www.anti-malware.ru/analytics/Market_Analysis/code-scanner-view-integrator (дата обращения: 25.11.2021).

3. GitLab SAST and Java 11. URL: https://widerin.net/blog/gitlab-sast-and-java-11/ (дата обращения: 25.11.2021).

4. Статический анализ. Анализ унаследованного кода, когда исходный код утрачен: делать или не делать? Преимущества от внедрения системы. URL: https://barforme.ru/staticheskii-analiz-analiz-unasledovannogo-koda-kogda/ (дата обращения: 25.11.2021).

5. SonarQube. URL: https://ru.wikipedia.org/wiki/SonarQube (дата обращения: 25.11.2021).

6. AppChecker Cloud. URL: https://npo-echelon.ru/production/65/10920 (дата обращения: 25.11.2021).

7. Cppcheck. URL: https://ru.wikipedia.org/wiki/Cppcheck (дата обращения: 25.11.2021).

8. Инструменты, которые повышают качество кода Java. URL: https://javarush.ru/groups/posts/3208-kofe-breyk-61-instrumentih-kotorihe-povihshajut-kachestvo-koda-java-jazihk-java-i-proektih-s-ot (дата обращения: 25.11.2021).

9. Introduction. URL: https://golangci-lint.run/ (дата обращения: 25.11.2021).

10. 10 BEST Dynamic Application Security Testing (DAST) Software. URL: https://www.softwaretestinghelp.com/dynamic-application-security-testing-dast-software/ (дата обращения: 25.11.2021).

11. PT Application Inspector: Революция в анализе защищенности приложений. URL: https://unlim.group/upload/iblock/66c/66c758ed850fcd371a9de6c6cacba7c1.pdf (дата обращения: 25.11.2021).

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
А. Г. Кабанец*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ФАЗЗИНГ-ТЕСТИРОВАНИЕ. СОВРЕМЕННЫЕ СРЕДСТВА ФАЗЗИНГА

Рассмотрены понятие фаззинг-тестирования, методы фаззинга, приведена классификация фаззеров, описаны некоторые фаззеры.
Ключевые слова: *фаззеры, фаззинг-тестирование, методы фаззинга.*

V. G. Eryshov*
PhD, Tech., Associate Professor
A. G. Kabanets*
Student
**St. Petersburg State University of Aerospace instrumentation*

FUZZING TESTING. MODERN FUZZING TOOLS

The concept of fuzzing testing, fuzzing methods are considered, the classification of fuzzers is given, and some fuzzers are given.
Keywords: *fuzzers, fuzzing testing, fuzzing methods.*

В настоящее время очень важно проверять безопасность и надежность программного обеспечения и компьютерных систем. Поэтому при разработке программного обеспечения, которое бы соответствовало требованиям безопасности, часто используют фаззинг-тестирование. Фаззинг-тестирование, или фаззинг, – это процедура выявления ошибок или уязвимостей в программном обеспечении, принцип которой состоит в преднамеренном введении в программу некорректных или случайных данных для того, чтобы вызвать исключения или сбои программы.

Впервые термин «фаззинг» [1] был упомянут в 1988 г. в Университете Висконсина. На научном семинаре под руководством Бартона Миллера была представлена программа, предназначенная для тестирования устойчивости программ и приложений на базе Unix. При работе эта программа использовала случайно сгенерированные данные, которые потоком передавала как входные данные для исследуемых программ до тех пор, пока они не выходили из строя и завершали свою работу из-за ошибок. Кроме того, в приложение были встроены инструменты для обнаружения и локализации ошибок, а с помощью систематического анализа определялся тип найденных ошибок. И так как приложение можно было использовать для проверки работоспособности огромного набора программ, Бартон Миллер обнародовал исходники для дальнейшего развития проекта. Фаззинг позволяет прогнозировать наличие различных ошибок и проводить анализ для выявления того, какие входные данные привели к ошибкам.

Классификация фаззеров по разным признакам [2].

1. По наличию знаний о входных данных.

– **Black-box** – в этом случае про формат данных ничего не известно. Для black-box фаззера исследуемая программа представляет собой черный ящик, поскольку нет никаких сведений о ее внутренней структуре. Например, black-box-фаззером можно назвать программу, которая генерирует и подает на вход тестируемого объекта случайные данные, никак не связанные с этим объектом. Black-box-фаззеры могут лишь выявить мелкие простые ошибки. Поэтому существуют попытки их улучшения: black-box-фаззер, который постепенно исследует и узнает внутреннюю структуру тестируемой программы во время фаззинга, анализируя выход программы.

– **White-box** – о формате данных известно все. Для увеличения покрытия исследуемого кода фаззеры данного типа применяют про-

граммный анализ. Например, SAGE использует символическое выполнение для систематического исследования различных путей в программе. Если доступна спецификация программы, то white-box-фаззер может в процессе тестирования генерировать не совсем случайные данные, а основанные на каких-то специальных моделях, которые соответствуют спецификации данной программы, а также проверять на соответствие выход программы. White-box-фаззер – достаточно эффективное средство для обнаружения ошибок, которые скрыты глубоко в программе. Минус состоит в том, что время, которое будет затрачено на анализ спецификации или самой программы, может быть слишком долгим.

– Gray-box – известны некоторые сведения о формате данных, а не полная информация. Gray-box-фаззер использует инструментарий, а не программный анализ для сбора сведений об исследуемом объекте. В качестве примера можно привести AFL и libFuzzer, которые применяют упрощенные средства для отслеживания переходов базового блока. Данный способ фаззинга приводит к приемлемым затратам ресурсов, но в то же время оповещает фаззер об увеличении покрытия кода в процессе тестирования, благодаря этому gray-box-фаззеры стали чрезвычайно эффективными инструментами обнаружения уязвимостей.

2. По цели, которая будет подвергнута фаззингу:

– source-based, если имеется исходный код проекта;

– binary-based, если отсутствует исходный код проекта.

3. По наличию обратной реакции от тестируемого приложения:

– feedback driven, когда она есть;

– not feedback driven при ее отсутствии.

4. По операциям, которые будут совершаться над входными данными:

– генерационные, или порождающие, фаззеры. При тестировании они формируют входные данные с нуля, имитируя требующийся формат данных. В этом случае данные, подающиеся на вход программы, сгенерированы случайно, а наборы тестов не имеют связей между собой. Но так как современные приложения достаточно сложны в своем устройстве, данный тип фаззинга недостаточно эффективен на практике;

– мутационные фаззеры. При их использовании в процессе тестирования данные, которые подаются на вход тестируемого объекта, изменяются и адаптируются за счет исследования работы этого объекта и анализа выходных данных. И хотя данный вид фаззинга сложнее реализовать, он способствует увеличению объема поиска ошибок и позволяется находить ошибки, которые не были обнаружены ранее;

– комбинированные.

Некоторые методы фаззинга [3].

Метод заранее подготовленных ситуаций для тестирования. Перед тестированием и созданием самих тестов необходимо изучить примеры работы проверяемой программы, чтобы понять, какие данные в ней поддерживаются, а также какие у них подходящие значения. Затем уже формируются специальные тесты для проверки работоспособности программы. Цель тестов состоит в создании ситуаций, при которых программа даст сбой. Недостаток этого метода – неизбежные ограничения. Так как в методе заранее подготовленных ситуаций отсутствует доля случайности, то при тестировании проверяется не огромное разнообразие входных данных, а только заготовленный список.

Метод случайных данных можно применять в том случае, когда главным фактором тестирования является его скорость. В этом случае входные данные представляют собой псевдослучайные данные, при помощи которых фаззер пытается нарушить нормальную работу приложения. Обычно данный метод реализуют с помощью бесконечного цикла, внутри которого новые сгенерированные данные подаются на вход программы.

Мутационное тестирование протокола вручную можно назвать самым простым методом фаззинга. В нем отсутствует автоматизация. При использовании данного метода человек, выполняющий тестирование, вручную подает на вход проверяемой программы разные некорректные данные, пытаясь этими действиями привести к появлению ошибок в работе приложения. Такой способ фаззинг-тестирования чаще применяется для проверки работоспособности веб-приложений.

Мутационное тестирование, называемое также тестированием методом грубой силы, является фаззером, в результате работы которого используется какой-то образец файла и последовательно искажаются все его байты, слова или двойные. Объем кода, который будет протестирован, зависит от количества тестируемых файлов. И для обеспечения достаточного охвата проверенного кода необходимо использовать большое количество разнообразных образцов.

Автоматическое порождающее тестирование протокола – модернизированный метод мутационного фаззинга. Если применять этот метод тестирования, то в первую очередь необхо-

димо сделать исследование. После этого для генерации новых тестов используются не заготовленные шаблоны, а специальная грамматика, которая основана на спецификации. В результате устанавливаются части файла, остающиеся постоянными и недопустимые для изменений, и части, использующиеся в процессе тестирования как аргументы.

Существуют и более совершенные методы фаззинга. Например, тестирование, которое применяет трассировки и строит уравнения для SMT-решателей. Теоретически такой метод может способствовать покрытию самых труднодоступных частей кода.

У всех методов фаззинга имеются общие фазы тестирования, присутствующие в каждом из методов.

1. Постановка задачи. Без рассмотрения особенностей приложения будет затруднительно выбрать инструмент или технологию. Если вы тестируете разработанное внутри вашей фирмы приложение с целью анализа его безопасности, цель будет выбрана автоматически. Однако если вы ищете ошибки в приложениях сторонних производителей, нужно будет проявить гибкость. При определении цели изучите историю работы производителя программы – перечень уже выявленных в ней ошибок. Это можно сделать на таких сайтах, как SecurityFocus1 или Secunia2, где представлены найденные изъяны. Кроме выбора приложения, также может оказаться необходимо указать конкретный файл или библиотеку внутри этого приложения.

2. Определение вводимых значений. Почти все ошибки вызываются приложениями, в которых пользователям необходимо вводить данные и работать с ними без предварительной обработки или применения шаблонов проверки. Учет вариантов ввода жизненно важен для успешного тестирования.

3. Формирование некорректных данных. На данном этапе в зависимости от метода тестирования выбирается способ формирования новых входных данных. Можно использовать заранее определенные входные данные, можно менять уже ранее использованные данные или создавать постоянно меняющиеся.

4. Использование некорректных данных. В этом состоит смысл всего тестирования. На этом этапе сформированные данные для тестирования используются как входные данные для программы. В большинстве фаззеров данный этап автоматизирован.

5. Отслеживание исключений. Жизненно важный, но часто пропускаемый при фаззинге шаг – мониторинг исключений, или процесс поиска ошибок. Тестирование будет бесполезным, если не отслеживать, какие именно данные вызвали исключения.

6. Определение работоспособности. Это заключительный этап фаззинга, на котором принимается решение о возможности дальнейшего использования фрагментов, в которых произошли ошибки. И данный этап, как правило, осуществляется вручную, а для его реализации необходимы особые знания в области безопасности.

Также фаззеры бывают:

– локальные;

– удаленного доступа;

– оперативной памяти;

– интегрированные среды фаззеров.

Локальные фаззеры. В эту категорию входят фаззеры командной строки и фаззеры переменной среды, которые работают с аргументами командной строки и переменными окружения, такими как путь файла, аргумент выполнения и др. Кроме того, к данной категории относятся фаззеры формата файла. Входными данными для фаззера могут быть любые файлы любого формата, в том числе «неправильного». Задача антивирусного сканера состоит в том, чтобы както определить формат файла и начать с ним взаимодействовать: попробовать распаковать, или открыть, запустить эвристическое сканирование и т. п.

Фаззеры удаленного доступа используются для программ и приложений, связанных с сетевым интерфейсом. В современном мире практически на всех предприятиях сотрудники при работе пользуются общедоступными серверами, которые включают веб-сайты, электронную почту и др. При наличии в данных системах уязвимостей различного рода появляется возможность проведения атак на другие серверы, связанные с этими. Фаззеры данной категории можно разделить на фаззеры сетевых протоколов, фаззеры веб-приложений и фаззеры веббраузеров.

Фаззеры сетевых протоколов включают фаззеры простых протоколов и фаззеры более сложных протоколов. Первые протоколы в большинстве случаев обладают простой системой безопасности, основаны на тексте ASCII, и в них отсутствуют поля длины или контрольной суммы. Вторые же по большей части строятся на основе двоичных данных. Для распознавания таких протоколов может понадобиться кодирование.

При фаззинге веб-приложений тестер прежде всего ищет уязвимости, которые специфич-

ны для таких приложений, как SQL, XSS и т. д. Это требуется, чтобы фаззеры могли использовать HTTP и собирать ответы для дальнейшего анализа и определения наличия ошибок.

Фаззеры веб-браузеров часто используют функциональность HTML для автоматизации процесса фаззинга. Например, существует утилита для постоянной автоматической загрузки случаев для тестирования. Эта уникальная способность веб-браузеров позволяет создать полностью автоматизированный клиентский фаззер без особых сложностей и наворотов.

Фаззеры оперативной памяти. Порой во время тестирования что-то препятствует быстрому и эффективному фаззингу. Тогда может оказаться полезным фаззинг оперативной памяти. При применении одного из его вариантов необходима «заморозка», после которой делается моментальный снимок процесса и сразу же подаются некорректные данные в один из шаблонов анализа ввода. По итогу каждого случая тестирования делается новый снимок, вбрасываются новые данные. Так повторяется, пока все случаи для тестирования не закончатся. Как и любой другой способ фаззинга, фаззинг оперативной памяти обладает преимуществами и недостатками. Среди преимуществ отметим:

– скорость. Не приходится иметь дело с пропускной способностью сети, к тому же можно игнорировать тот не имеющий значения код, который исполняется между получением пакета из сети и собственно анализом; в результате тестирование улучшается;

– ярлыки. Иногда протокол использует обычные алгоритмы кодирования или сжатия либо содержит код проверки контрольной суммы. Вместо того чтобы тратить время на создание фаззера, который мог бы работать со всем этим, стоит создать фаззер внутренней памяти, который может сделать моментальный снимок после извлечения, декодирования, проверки контрольной суммы, чем сэкономит множество сил.

Недостатком фаззинга оперативной памяти можно назвать его сложность для выполнения.

Интегрированные среды фаззеров (фреймворки) используются для тестирования разнообразных программ. Это универсальный фаззер или библиотека фаззеров, которая способствует упрощению представления данных для множества типов объектов. Обычно интегрированные среды фаззеров включают библиотеку, которая производит фаззинговые строки или значения, как правило, вызывающие проблемы при анализе. Также типичен набор шаблонов для упрощения ввода и вывода в сети и на диске.

Разработчики используют различные программы для осуществления фаззинга. На данный момент на рынке актуальны следующие инструменты [4].

– **American Fuzzy LOP**. Программа предназначена для развертывания без особой настройки. При ее создании было проведено огромное количество исследований, касающихся работоспособности оптимальных фаззеров, а также результатов тестирования, которые в наибольшей степени помогают тестировщикам в их работе. Данная программа может быть использована, когда требуются минимальные затраты по времени для создания запроса и получения результатов.

– **Radamsa**. Этот фаззер предназначен для отправки примеров запросов программам, которые могут привести к неожиданным результатам. Главный его плюс – точность.

– **Honggfuzz**. Данный фаззер оптимизирован, многопоточен и ориентирован на безопасность. Он способен задействовать все системные ресурсы. В отличие от других фаззеров, при использовании которых необходимо запускать сразу несколько экземпляров, данный фаззер использует все доступные ядра процессора, чтобы ускорить сам процесс фаззинга.

– **Libfuzzer**. Является инструментом эволюционного фаззинга. Его цель – генерация более релевантных результатов, чем при использовании традиционных фаззеров.

– **OSS-Fuzz**. Данный фаззер применяется для работы с программным обеспечением с открытым исходным кодом. Цель заключается в помощи сообществу разработчиков с открытым исходным кодом при создании более безопасных приложений.

– **Sulley Fuzzing Framework**. Является как механизмом фаззинга, так и фреймворком тестирования. Главное отличие данного фаззера от других в том, что при его использовании можно безотказно работать достаточно длительное время (несколько дней), часто проверяя программу на возникновение различных нетипичных реакций при подаче неверных данных, а также фиксировать все полученные результаты. Кроме того, у Sulley имеется несколько дополнительных функций, например возможность параллельной работы. Инструмент может без дополнительных настроек определять, какие наборы тестов привели к сбою.

– **Boofuzz**. Данный фаззер является усовершенствованной версией Sulley Fuzzing Framework.

– BFuzz. Хотя этот фаззер уже активно используется, он существует в бета-версии. При использовании BFuzz в качестве исследуемого объекта выступают различные веб-страницы и веб-сайты.

– PeachTech Peach Fuzzer. Принцип работы этого фаззера состоит в возможности самостоятельного настраивания механизма фаззинга пользователем с помощью заранее сформированных тестов, разработанных для большого количества программ. Поэтому PeachTech Peach Fuzzer позволяет пользователям лично контролировать тестирование.

Заключение

В настоящее время фаззинг в основном используется как автоматизированный метод обнаружения ошибок, которыми могут воспользоваться злоумышленники, в программах и приложениях, где важным условием является безопасность. В более широком плане фаззинг применяется не для демонстрации отсутствия ошибок, а для демонстрации их наличия в программе с целью дальнейшего устранения. Фаззинг-тестирование – инструмент, который помогает выявить уязвимости программного обеспечения и предотвратить их использование в будущем.

Библиографический список

1. Фаззинг: материал из Национальной библиотеки им. Н. Э. Баумана. URL: https://ru.bmstu.wiki/Фаззинг (дата обращения: 25.011.2021).

2. DevSecOps: организация фаззинга исходного кода. URL: https://habr.com/ru/company/dsec/blog/517596/ (дата обращения: 25.011.2021).

3. *Саттон М., Грин А., Амини П.* Fuzzing: исследование уязвимостей методом грубой силы. СПб.: Символ Плюс 2009. 560 с.

4. 9 популярных инструментов для фаззинга. URL: https://cisoclub.ru/9-populyarnyh-instrumentov-dlya-fazzinga/ (дата обращения: 25.011.2021).

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
А. А. Клименко*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ТИПЫ, КЛАССЫ, ОБЗОР СОВРЕМЕННЫХ МЕЖСЕТЕВЫХ ЭКРАНОВ, СЕРТИФИЦИРОВАННЫХ ПО ТРЕБОВАНИЯМ ФСТЭК

Рассмотрены типы, классы и приведен обзор современных сертифицированных программных, программно-аппаратных межсетевых экранов.
Ключевые слова: *класс, тип, межсетевой экран, брандмауэр, фильтрация пакетов данных.*

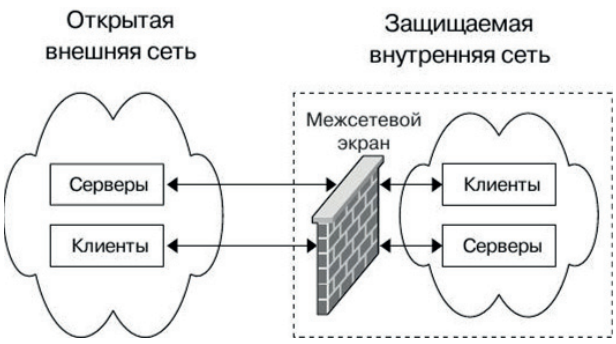
V. G. Eryshov*
PhD, Tech., Associate Professor
A. A. Klimenko*
Student
**St. Petersburg State University of Aerospace Instrumentation*

TYPES, CLASSES, OVERVIEW OF MODERN FIREWALLS, CERTIFIED ACCORDING TO FSTEC REQUIREMENTS

The types, classes and an overview of modern certified software, software and hardware firewalls are considered.
Keywords: *class, type, firewall, firewall, data packet filtering.*

Один из важнейших механизмов защиты автоматизированных систем (АС) от атак хакеров – межсетевой экран. Межсетевым экраном называют программную или программно-аппаратную систему, осуществляющую проверку входящих и выходящих потоков информации, которая может фильтровать данные, обеспечивая тем самым защиту АС [1] (рисунок).

Пакеты данных могут фильтроваться по различным параметрам протоколов разных уровней стека TCP/IP (таблица) [2].



Принцип действия межсетевых экранов

Различные критерии фильтрации данных

Уровень	Критерий
Сетевой	МЭ анализирует заголовки пакетов (IP, ICMP, протоколы маршрутизации)
Транспортный	МЭ анализирует некоторые параметры заголовков (TCP, UDP)
Прикладной	МЭ анализирует заголовки, типы команд и др. на прикладном уровне

Существуют два метода фильтрации данных межсетевым экраном:

1) назначаются пакеты, которые межсетевой экран должен заблокировать;

2) назначаются пакеты, которые межсетевой экран должен разрешить отправлять и принимать.

IP-адрес системы может быть скрыт межсетевым экраном, если будет использована трансляция сетевых адресов NAT (Network Address Translation). Режимов трансляции несколько: динамический, статический, статический с динамической выборкой адресов IP, комбинированный.

В Информационном сообщении ФСТЭК России от 28 апреля 2016 г. № 240/24/1986 «Об утверждении требований к межсетевым экранам»

[3] были выделены следующие типы межсетевых экранов:

– межсетевой экран уровня сети (тип «А»), применяемый на физической границе (периметре) информационной системы или между физическими границами сегментов информационной системы. *Межсетевые экраны типа «А» могут иметь только программно-техническое исполнение*;

– межсетевой экран уровня логических границ сети (тип «Б»), применяемый на логической границе (периметре) информационной системы или между логическими границами сегментов информационной системы. *Межсетевые экраны типа «Б» могут иметь программное или программно-техническое исполнение*;

– межсетевой экран уровня узла (тип «В»), применяемый на узле (хосте) информационной системы. *Межсетевые экраны типа «В» могут иметь только программное исполнение и устанавливаются на мобильных или стационарных технических средствах конкретного узла информационной системы*;

– межсетевой экран уровня веб-сервера (тип «Г»), применяемый на сервере, обслуживающем сайты, веб-службы и веб-приложения, или на физической границе сегмента таких серверов (сервера). *Межсетевые экраны типа «Г» могут иметь программное или программно-техническое исполнение и должны обеспечивать контроль и фильтрацию информационных потоков по протоколу передачи гипертекста, проходящих к веб-серверу и от веб-сервера*;

– межсетевой экран уровня промышленной сети (тип «Д»), применяемый в автоматизированной системе управления технологическими или производственными процессами. *Межсетевые экраны типа «Д» могут иметь программное или программно-техническое исполнение и должны обеспечивать контроль и фильтрацию промышленных протоколов передачи данных.*

Выделяют шесть классов защиты межсетевых экранов, классифицирующих требования к функциям безопасности межсетевых экранов. Самый низкий класс защиты – шестой, самый высокий – первый.

Применение классов:

6-й класс – в государственных ИС 3- и 4-го классов защищенности, в АСУ ТП 3-го класса защищенности, в ИСПДн при необходимости обеспечения 3- и 4-го уровней защищенности персональных данных;

5-й класс защиты – в государственных ИС 2-го класса защищенности, в АСУТП 2-го класса защищенности, в ИСПДн при необходимости обеспечения 2-го уровня защищенности персональных данных;

4-й класс защиты – в государственных ИС 1-го класса защищенности, в АСУТП 1-го класса защищенности, в ИСПДн при необходимости обеспечения 1-го уровня защищенности персональных данных, в ИС общего пользования 2-го класса.

Межсетевые экраны 3-, 2- и 1-го классов защиты применяются в ИС, в которых обрабатывается информация, содержащая государственную тайну.

Российский рынок МЭ сегодня представлен в Государственном реестре сертифицированных СЗИ [4] следующими производителями и современными сертифицированными по требованиям ФСТЭК средствами (в скобках представлен срок действия сертификата ФСТЭК).

ООО «Предприятие „ЭЛТЕКС“» [5]:

1) межсетевой экран ESR-20, версия ПО 1.5;

2) межсетевой экран ESR-21, версия ПО 1.5;

3) межсетевые экраны ESR-1500, ESR-1511, версия ПО 1.5;

4) межсетевой экран ESR-1000, версия ПО 1.5;

5) межсетевой экран ESR-100, версия ПО 1.5;

6) межсетевой экран ESR-200, версия ПО 1.5.

Американская транснациональная корпорация Fortinet:

1) программно-аппаратный комплекс (ПАК) FortiGate для защиты промышленной сети с встроенным межсетевым экраном;

2) программно-аппаратный комплекс FortiGate, функционирующий под управлением ПО FortiOS версии 6.X.

ООО «ИнфоВотч Арма»: программный комплекс «InfoWatch ARMA Industrial Firewall» (27.07.2021–27.07.2026).

Американская транснациональная компания Cisco, по данным сайта ФСТЭК России из Государственного реестра сертифицированных СЗИ [4].

Коммутаторы	1. Cisco VS-C6509E-SUP2T с модулем ASASM с ПО: Cisco ASA версии 9.12. 2. Cisco VS-C6509E-SUP2T с ПО Cisco IOS версии 15.5.1-SY5. 3. Cisco WS-C3850R-48U-S с ПО Cisco IOS XE версии 16.9.5. 4. Cisco IE-3000 (IE-3000-8TC; 8TC-E) с ПО Cisco IOS. 5. Cisco Nexus 7700 с ПО Cisco NX-OS версии 8.4(2). 6. Cisco WS-C3560V2-24TS с ПО Cisco IOS. 7. Cisco C9300-24T-A с ПО Cisco IOS-XE версии 16.12.02. 8. Cisco Catalyst WS-C3650-24TS-E с ПО Cisco IOS-XE версии 16.12.1.

Межсетевые экраны	1. Cisco ASA 55xx (Cisco ASA 5512, 5515, 5525, 5545, 5555, 5585) с ПО Cisco ASA версии 9.X. 2. Cisco Firepower 2100 (модели: Firepower 2110, Firepower 2120, Firepower 2130, Firepower 2140). 3. Cisco Firepower 2130 с ПО Cisco ASA версии 9.14. 4. Cisco «ASA 5525-K8» с ПО Cisco ASA версии 9.14 (14.05.2021–14.05.2026).
Маршрутизаторы	1. Cisco 2901/K9 с ПО: Cisco IOS версии 15.7.3M4b. 2. Маршрутизатор Cisco ISR4331/K9 с ПО Cisco IOS-XE версии 16.12.1a.

VMWare: программная платформа VMware NSX-T Data Center.

Smart Tech Innovations: программное обеспечение «Система сетевой безопасности Mirada».

АО «ИнфоТеКС»: программно-аппаратный комплекс ViPNet Coordinator IG 4.

Информационная внедренческая компания (ИВК): межсетевой экран «ИВК КОЛЬЧУГА-К».

Kerio Technologies & Tiny Software: межсетевой экран Kerio Control.

ООО «АМИКОН»: программно-аппаратный комплекс «ФПСУ-IP 3.X».

АО «InfoWatch (ИнфоВотч)»: программный комплекс InfoWatch Attack Killer.

Trend Micro: программное обеспечение Trend Micro Deep Security 10.

АО «ЭЛВИС-ПЛЮС»: ПАК «ЗАСТАВА-Клиент» «VPN/FW «ЗАСТАВА», в.6» (19.05.2020–19.05.2025).

DIONIS DPS: ПАК Dionis DPS (13.02.2020–13.02.2025).

АО «АЛТЭКС-СОФТ»: ПАК «Check Point Security Gateway версии R77.30» (28.01.2020–28.01.2025).

Заключение

В статье дано определение МЭ, рассмотрены его функции, правила фильтрации пакетов данных, типы, классы и перечень современных сертифицированных образцов. Сотрудники служб ИБ, определив в соответствии с требованиями руководящих документов ФСТЭК тип и класс защищенности МЭ, могут воспользоваться данным перечнем и выбрать соответствующие необходимые средства и системы межсетевого экранирования.

Библиографический список

1. Информационная безопасность. Межсетевое экранирование. URL: https://informationsecurityweb.wordpress.com/ (дата обращения: 22.11.2021).

2. Защита от сетевых атак на основе межсетевого экранирования: лекция // Интуит. Национальный открытый университет. URL: https://intuit.ru/studies/courses/940/456/lecture/10222 (дата обращения: 22.11.2021).

3. Об утверждении требований к межсетевым экранам: информ. сообщение от 28 апр. 2016 г. № 240/24/1986 // ФСТЭК России: офиц. сайт. URL: https://fstec.ru/normotvorcheskaya/informatsionnye-i-analiticheskie-materialy/1142-informatsionnoe-soobshchenie-fstek (дата обращения: 22.11.2021).

4. Государственный реестр сертифицированных средств защиты информации // ФСТЭК России: офиц. сайт. URL: https://fstec.ru/normotvorcheskaya/informatsionnye-i-analiticheskie-materialy/1142-informatsionnoe-soobshchenie-fstek (дата обращения: 22.11.2021).

5. Межсетевые экраны. URL: https://eltex-msk.ru/katalog/setevoe-oborudovanie/mezhsetevye-ekrany (дата обращения: 22.11.2021).

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
К. А. Ларионец*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОСНОВНЫЕ ЭТАПЫ, МЕТОДИКИ И СРЕДСТВА ПРОВЕДЕНИЯ ТЕСТИРОВАНИЯ НА ПРОНИКНОВЕНИЕ

Рассмотрены процесс тестирования на проникновение в информационные системы, основные методики, этапы и инструменты тестирования.
Ключевые слова: *тестирование на проникновение, пентест, «черный» ящик, «серый» ящик, «белый» ящик.*

V. G. Eryshov*
PhD, Tech., Associate Professor
K. A. Larionets*
Student
**St. Petersburg State University of Aerospace Instrumentation*

THE MAIN STAGES, TECHNIQUES AND TOOLS FOR CONDUCTING PENETRATION TESTING

The article discusses the process of penetration testing into information systems, the main methods and stages of testing, the main tools.
Keywords: *penetration test, pentest, invasion, «black» box, «grey» box, «white» box.*

Тестирование на проникновение (сокр. пентестинг, от англ. Penetration Testing) – это процедура оценки реальной защищенности информационной системы (ИС) с использованием контролируемых и максимально безопасных для инфраструктуры и бизнес-процессов атак, а также выявление и попытки эксплуатации уязвимостей. Пентест предоставляет детальную информацию о проблемах информационной безопасности (ИБ), помогает распланировать наиболее важные направления по улучшению реальной защищенности информационной системы [1].

Цель тестирования – обнаружить возможные уязвимости, которые способны привести к нарушению трех основных принципов ИБ: конфиденциальности, целостности и доступности информации путем провоцирования некорректной работы системы или же доведения до отказа в обслуживании.

По результатам пентеста делается оценка текущего уровня защищенности ИС, возможности противостоять попыткам вторжения потенциального нарушителя ИБ. В случае выявления уязвимостей в ИС обязательно составляется список рекомендаций по их устранению, график устранения, привлекаемые ресурсы. Конкретный объем и способы выполнения теста могут различаться в зависимости от потребностей организации, заказывающей оценку, а также от предложений услуг консалтинговой фирмы, проводящей тест.

Тестирование может быть сосредоточено на веб- и мобильных приложениях, сетевой инфраструктуре, беспроводных устройствах, физических офисах и многих других объектах. Пентестеры могут использовать человеческий фактор (социальная инженерия) или специально разработанный код. Все зависит от масштаба планируемого вторжения. Тестирование на проникновение предполагает два направления работ [1].

1. Внешний пентест. Проверка распространяется на все пограничные ресурсы, расположенные в демилитаризованной зоне (средства удаленного доступа, веб-сайты и др.), межсетевые экраны и другие устройства, доступ к которым имеется через публичные IP-адреса (рис. 1) [2].

Цель нарушителя – проникновение во внутреннюю сеть либо получение контроля над внешними ресурсами.

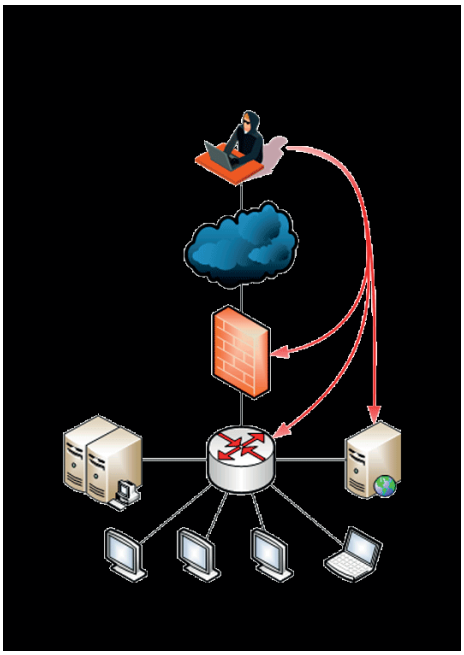


Рис. 1. Внешний пентест

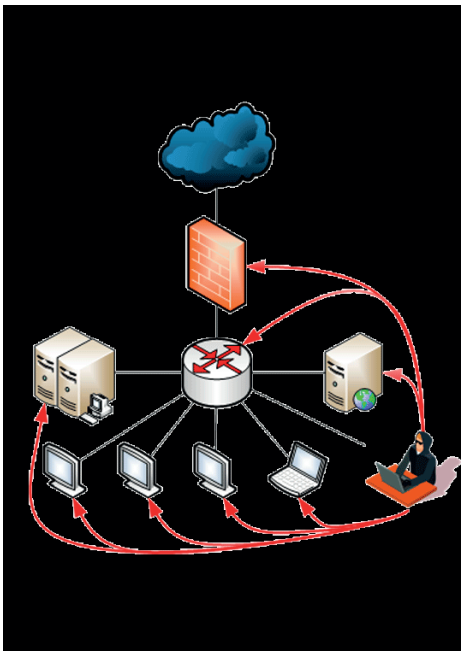


Рис. 2. Внутренний пентест

2. Внутренний пентест. Проверка распространяется на внутренние серверы, сетевое оборудование, АРМ пользователей, средства виртуализации. Выявляются всевозможные недостатки в организации сети, проверяются Wi-Fi-сети (рис. 2) [2]. Ко всему перечисленному проверку проходят те же ресурсы, что и при внешнем пентесте, но доступ осуществляется из внутренней сети (с использованием внутренних IP-адресов), т. е. нарушитель действует из сегмента локальной сети.

Цель нарушителя – контроль инфраструктуры или отдельных сервисов сети.

Основные типы тестирования

Главный критерий – осведомленность взломщика об атакуемом объекте. Степень его информированности можно систематизировать следующим образом.

1. «**Черный ящик**». Метод черного ящика предполагает имитацию действий взломщика, который не располагает сведениями о компании и ее корпоративной сети. Большая часть хакеров придерживается именно этой методики, перебирая весь доступный инструментарий для выявления и эксплуатации уязвимостей в системе защиты.

2. «**Белый ящик**». Полная противоположность «черной» версии. При использовании данного метода атакующий злоумышленник имеет все необходимые сведения: начиная с архитектуры сети и заканчивая средствами защиты. В таких ситуациях взломщик – нынешний или бывший сотрудник, имеющий учетную запись в организации.

3. «**Серый ящик**». Метод серого ящика – промежуточный вариант. Взломщик не располагает общей картиной, но знает важные аспекты. В данной ситуации взломщиком может быть сотрудник организации либо клиент, который имеет доступ к системе и минимальные сведения о сети. Либо же появляется взломщик со стороны, добывший информацию, которая поможет осуществить верный вектор атаки.

Основные этапы тестирования

Типичное тестирование состоит из четырех этапов [3].

Этап 1. Сбор информации:

1) составление карты сети;

2) определение возможных целей;

3) перечисление слабых мест в службах, работающих на этих целях.

Этап 2. Целенаправленное проникновение: проникновение в уязвимые сервисы (получение несанкционированного доступа к ним).

Этап 3. Постэксплуатация и повышение привилегий:

1) определение информации о скомпрометированных системах, которая может быть использована для дальнейшего доступа (закрепления в системе);

2) повышение привилегий до самого высокого уровня доступа в сети (до уровня администратора).

Этап 4. Документирование:

1) сбор доказательств проникновения;

2) подготовка и предоставление окончательного отчета компании-заказчику.

После того как тестовая часть вторжения завершена, пентестер посвящает оставшееся время составлению подробного отчета. Этот отчет содержит детальное описание всех способов, которыми удалось взломать сеть и обойти меры безопасности, а также предложение мер, которые компания может предпринять, чтобы закрыть выявленные бреши и гарантировать, что они больше не будут использованы кем-либо еще.

Основные современные методики тестирования

Выделяют пять самых известных методик: OSSTMM, NIST SP800-115, OWASP, ISSAF и PTES [4]. Для проведения тестирования можно использовать одну из них, однако в зависимости от конкретной организации и ее бизнес-процессов можно применить несколько.

1. OSSTMM (The Open Source Security Testing Methodology Manual). В OSSTMM подробно описаны: план тестирования, показатели для анализа уровня защищенности и советы по формированию финального отчета.

Методология предлагает пять основных направлений для тестирования операционной безопасности:

– безопасность человека. Безопасность, зависящая напрямую от психологического взаимодействия людей, либо физического;

– физическая безопасность. Элемент безопасности (чаще неэлектронный), чья работа предполагает электромеханическое или физическое воздействие;

– беспроводная связь. Защищенность беспроводных средств коммуникаций, от инфракрасных датчиков до Wi-Fi;

– телекоммуникации. Цифровые или аналоговые средства телефонной коммуникации. Это касается не только телефонии, но и передачи служебных данных по телефонным каналам;

– сети передачи данных. Внутренняя и внешняя безопасность корпоративных сетей, подключений с использованием сети Интернета и сетевого оборудования (маршрутизаторы, коммутаторы, концентраторы и др.).

Данное разделение способствует комплексной оценке уровня защищенности организации, а также помогает упростить весь процесс тестирования.

2. NIST SP800-115 (NIST Special Publications 800 Series). В подразделе стандарта опубликованы: порядок реализации тестирования, важные вопросы оценки уровня безопасности, рекомендации для анализа результатов и дальнейшей разработки мер по снижению выявленных рисков. Данная методология – одна из важных для проведения тестирования, многие профессиональные аудиторские компании придерживаются принятых ею норм. Стандарт включает:

– методы осмотра: сниффинг сети, обзор логов, документации, конфигурации системы, осуществление проверки целостности всех файлов;

– методы, использующиеся для проверки целевых уязвимостей: социальная инженерия, подбор паролей;

– оценку безопасности: анализ, оценка, а также непосредственная обработка данных;

– действия по имеющимся результатам: устранение найденных уязвимостей, указание рекомендаций для снижения рисков, составление финального отчета.

3. OWASP (Open Web Application Security Project) – онлайн-проект, который предлагает помощь по обеспечению безопасности веб-приложений. Данным проектом разработана методология для тестирования сайтов, приложений, API. OWASP выпустил несколько значимых документов, среди которых:

– OWASP Top10 – описаны широко распространенные уязвимости мобильных и веб-приложений, а также устройств интернета вещей. Угрозы упорядочены: по степени воздействия на бизнес-процессы; по сложности;

– OWASP Testing Guide – включает различные методики по тестированию защищенности веб-приложений, подкрепленные примерами из практики;

– OWASP Developer Guide – содержатся советы по проектированию безопасного кода;

– OWASP Code Review – предназначено для веб-разработчиков, а также менеджеров продуктов. В документе описаны наилучшие методы по проверке безопасности написанного кода.

Главное преимущество онлайн-проекта OWASP заключается в методологии, которая определяет тестирование на всех стадиях жизненного цикла разработки: от установления требований и проектирования до разработки, внедрения и дальнейшей поддержки.

4. ISSAF (Information System Security Assessment Framework). В документе содержатся ответы на вопросы, связанные с информационной безопасностью, изложены рекомендации

по тестированию на проникновение. Внутри документа имеется описание утилит, с помощью которых можно провести тестирование, приведены указания по использованию, а также объяснено, какие итоги можно получить в процессе тестирования.

ISSAF – достаточно подробная и сложная методология, которая может адаптироваться под любую организацию для проверки ИБ. Согласно ISSAF, все этапы тестирования тщательно документируются. Для каждого этапа описаны рекомендации по использованию конкретных инструментов.

5. PTES (Penetration Testing Methodologies and Standards). В стандарте указаны рекомендации по проведению базового тестирования, а также его расширенных вариантов, в частности для организаций с особыми требованиями к информационной безопасности. Главное преимущество стандарта PTES в том, что в нем содержится подробное описание ожиданий и целей пентеста.

Стандарт выделяет следующие этапы пентеста.

Обследование. Тестировщику организация предоставляет общую информацию об объектах инфраструктуры.

Моделирование угроз. Определяются направления и векторы атаки критически важных элементов ИТ с учетом бизнес-процессов.

Анализ уязвимостей. Выявляются и оцениваются риски, связанные с найденными уязвимостями. На данном этапе рассматриваются все уязвимости, к которым может прибегнуть злоумышленник.

Эксплуатация уязвимости. Для имитации неправомерных действий производится попытка эксплуатировать найденную уязвимость. Тестировщик пробует получить доступ к элементам информационной системы.

Составление отчета. Итог тестирования на проникновение подробно документируется вместе со всей полученной информацией о найденных уязвимостях и дальнейшими рекомендациями по их устранению.

Также в стандарте PTES содержится руководство по выполнению повторного (постэксплуатационного) тестирования, помогающего определить эффективность выявленных уязвимостей.

Существующие средства для тестирования на проникновение

В процессе тестирования на проникновение пентестер может использовать большое количество инструментов для выявления уязвимостей систем. Эти инструменты можно разделить на четыре большие группы [5].

1. Комплексные инструменты:

– Metasploit – программа, предоставляющая информацию об уязвимостях, поиск вирусных программ для систем обнаружения вторжений, тестирование атак на вычислительные системы;

– Burp Suite – платформа для тестирования безопасности веб-приложений, включающая компоненты, обеспечивающие полноценный аудит безопасности. Функции не ограничиваются поиском файловых структур, подбором паролей, модификацией запросов;

– Nessus – инструмент для автоматизации проверки и обнаружения уязвимостей в защите информационных систем;

– OpenVAS – сканер уязвимостей.

2. Сетевые сканеры:

– NMAP – утилита, предназначенная для сканирования IP-сетей с любым количеством объектов, определения состояния объектов сканируемой сети (портов и соответствующих им служб);

– MASSCAN – инструмент, позволяющий массово сканировать IP-порты; помогает определять сети огромных масштабов, например внутренние сети или Интернет.

3. Анализаторы трафика:

– TCPDUMP – сниффер, запускающийся из командной строки. С помощью этого инструмента можно узнать, какого формата пакеты проходят через сеть в данный момент. Инструмент обычно используется в учебных целях и для сетевой отладки, но утилита также позволяет выявлять попытки сканирования хоста и проводить атаки на сеть;

– Wireshark – анализатор сетевых протоколов, обеспечивающий захват пакетов в реальном времени, сортировку информации, просмотр всего содержимого, отображение нагрузки на сеть и многое другое;

– Ettercap – сниффер для анализа протоколов сети, использующий методику атаки «человек посередине»;

– Aircrack-ng – утилиты, с помощью которых можно тестировать безопасность беспроводных сетей;

– MITMPROXY – утилита командной строки, с помощью которой можно проводить отладку, оценку уровня конфиденциальности и тестирование на проникновение. Помимо этого, можно проверять, перехватывать, изменять проходящий HTTP-трафик.

4. Брутфорсеры:

– John the Ripper – инструмент, позволяющий проводить аудит паролей на предмет их сложности. Инструмент реализован таким об-

разом, что можно использовать сразу несколько вариантов атак: перебор по заданному словарю, полный перебор или же гибридный способ;

– **THC-Hydra** – инструмент для подбора паролей к сервисам;

– **HashCat** – один из самых эффективных инструментов по подбору паролей через хеш-значения.

Kali Linux предназначен для тестирования на проникновение, в нем заранее определены настройки для удобной работы с установленными инструментами.

Заключение

Тесты на проникновение – часть стандартной процедуры проверки уровня информационной безопасности организаций. Пентесты проводятся с использованием специализированных программ (подбор паролей, поиск уязвимостей, обнаружение вредоносных программ) и охватывают большое количество проверок. Типичные среди них:

– сбор информации, поиск данных о заказчике в открытых источниках;

– поиск информации о существующих ресурсах, программном обеспечении и приложениях, операционных системах;

– анализ уязвимостей и угроз ИБ, характерных для исследуемой ИС;

– эксплуатация найденных уязвимостей и угроз, имитация реальной атаки;

– формирование отчета, разработка рекомендаций по устранению найденных УБИ.

Библиографический список

1. Пентест – реальный взгляд на информационную безопасность организации. URL: https://www.securitylab.ru/blog/company/ecrs/346186.php (дата обращения: 29.11.2021).

2. Пентест – реальный взгляд на информационную безопасность организации. URL: https://www.ecrs.ru/blog/all/pentest-realnyy-vzglyad-na-informatsionnuyu-bezopasnost-organizatsii/ (дата обращения: 29.11.2021).

3. *Дорофеев А.* Профессиональное тестирование на проникновение: удел настоящих гиков-фанатов командной строки или уже нет? URL: https://habr.com/ru/company/npoechelon/blog/337776/ (дата обращения: 29.11.2021).

4. *Соколов А.* 5 методик тестирования на проникновение. URL: https://itglobal.com/ru-ru/company/blog/5-pentest-metodologies/ (дата обращения: 29.11.2021).

5. *Музалевский Ф. А.* Что такое пентест (pentest)? URL: https://rtmtech.ru/articles/chto-takoe-pentest/ (дата обращения: 29.11.2021).

УДК 003.09

В. Г. Ерышов*
кандидат технических наук, доцент
П. С. Летуновская *
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

АНАЛИЗ АЛГОРИТМОВ ИСПОЛЬЗОВАНИЯ ЦИФРОВЫХ ВОДЯНЫХ ЗНАКОВ В СЖАТЫХ ИЗОБРАЖЕНИЯХ

Рассмотрена одна из основных проблем современности цифрового мира – цифровое пиратство и способы борьбы с ним с помощью цифровых водяных знаков. Приведены результаты различных методов защиты от цифрового пиратства.
Ключевые слова: *алгоритмы, изображение, защита, цифровой водяной знак.*

V. G. Eryshov*
PhD, Tech., Associate Professor
P. S. Letunovskaya *
Student
** St. Petersburg State University of Aerospace Instrumentation*

ANALYSIS OF ALGORITHMS FOR USING DIGITAL WATERMARKS IN COMPRESSED IMAGES

Considered one of the main problems of the modern digital world – digital piracy, and ways to combat it using digital watermarks. The results of various methods of protection against digital piracy are presented.
Keywords: *algorithms, Image, Protection, Digital Watermark.*

На современном этапе развития общества безопасность цифровой информации имеет огромное значение. Одна из актуальных угроз для нее – цифровое пиратство [1]. Благодаря высоким темпам развития мультимедийных и интернет-технологий обрабатывать, передавать, получать информацию и делиться ей стало намного проще, поэтому существенно возросли случаи незаконного копирования, нарушения авторских прав, несанкционированного использования незаконно добытой информации. Согласно опросу, проведенному ВЦИОМ [2], 81% россиян считают, что нет необходимости платить деньги за просмотр или скачивание медиаресурсов.

В настоящее время наблюдается значительный интерес к цифровым водяным знакам, которые могут использоваться (внедряться) для защиты с помощью идентификации владельца цифрового информационного ресурса. Механизмы данного направления защиты настроены на встраивание скрытых меток, устойчивых к атакам различного вида. Благодаря внедрению цифрового водяного знака в авторское произведение можно доказать принадлежность произведения автору.

Цифровой водяной знак (далее – ЦВЗ) – это специальная метка, внедряемая в файл с целью контроля и защиты его использования [3]. ЦВЗ используются для защиты авторских прав, а также может служить защитой от несанкционированных использования, распространения и копирования защищаемой информации. Исходя из цели применения ЦВЗ, он может содержать различные вспомогательные данные: информацию о владельце, уникальную последовательность символов и пр.

Структурная схема использования, внедрения ЦВЗ в сжатое цифровое изображение и извлечения из него приведена на рис. 1.

В работе [4] показано, что стеганографические преобразования в частотной области контейнера намного устойчивее к его изменениям, чем алгоритмы, работающие в пространственной области. В связи с этим рассмотрим алгоритмы, которые используют для внедрения коэффициентов дискретно-косинусного преобразования (ДКП) изображения.

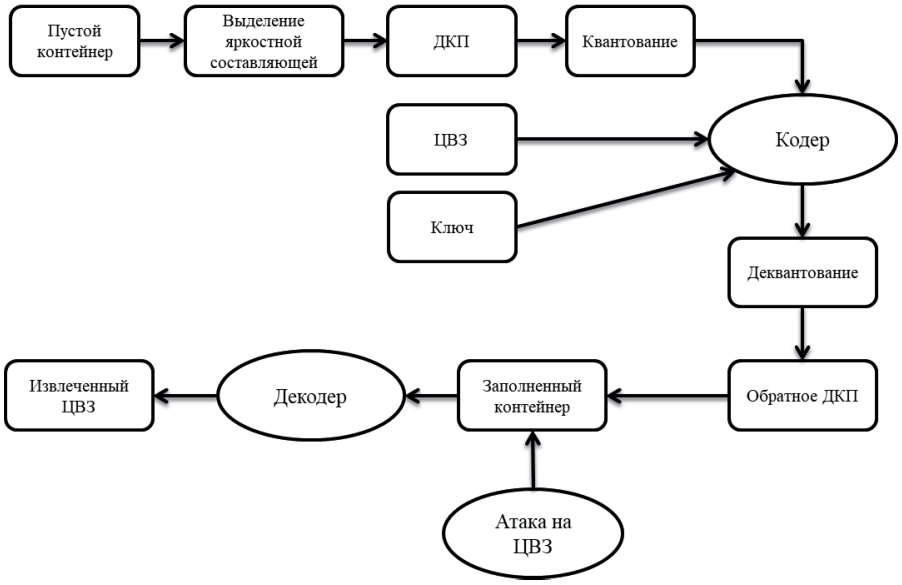


Рис. 1. Структурная схема внедрения и извлечения ЦВЗ

Алгоритм Кокса

В алгоритме Кокса ЦВЗ внедряется в файл путем изменения коэффициентов ДКП. Примем переменную $s = \{-1; +1\}$. Она будет обозначать, какой бит информации был внедрен (0 или 1). Для внедрения ЦВЗ выбирают наибольший коэффициент ДКП в блоке. Сам процесс внедрения описывается формулой

$$C_i^{'} = C_i e^{\alpha s}, \qquad (1)$$

где C_i – старое значение коэффициента дискретно-косинусного преобразования; $C_i^{'}$ – новое значение коэффициента дискретно-косинусного преобразования; α – весовой коэффициент. Чем он выше, тем больше заметность внедрения.

Для извлечения ЦВЗ из файла необходимо иметь пустой контейнер. Для извлечения бита информации находится разность между коэффициентами ДКП пустого и заполненного контейнера наибольшего значения. Этот процесс описывается формулой

$$s' = \begin{cases} 0, \text{ если } C_i^{'} - C_i e^{\alpha s} < 0 \\ 1, \text{ если } C_i^{'} - C_i e^{\alpha s} \geq 0 \end{cases}. \qquad (2)$$

Алгоритм Бенхама

В алгоритме Бенхама для внедрения в файл бита информации псевдослучайно выбираются три коэффициента среди средних частот блока ДКП.

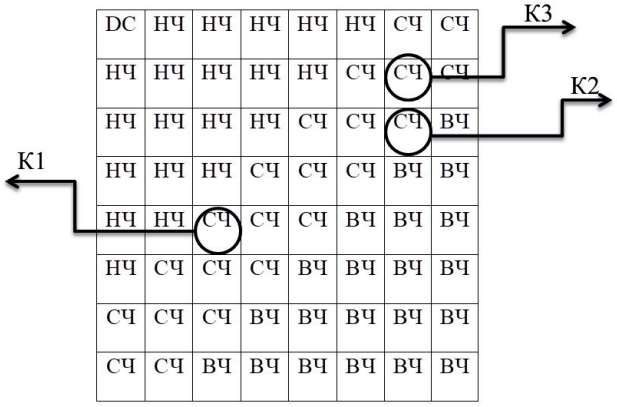


Рис. 2. Выбранные коэффициенты среди средних частот блока ДКП

Чтобы внедрить 1, третий коэффициент делают меньше минимального из первых двух на некоторое значение b, чтобы внедрить 0 – больше максимального из первых двух на некоторое значение b.

Для извлечения ЦВЗ из файла необходимо сравнить третий коэффициент с первыми двумя, и сделать соответствующие выводы о том, какой бит информации был внедрен. На рис. 2 схематично изображены три выбранных коэффициента в блоке ДКП.

Алгоритм Сентхурана и Ранатунги

Авторами [5] предложен алгоритм внедрения ЦВЗ на основе изменения таблицы квантования JPEG. Внедрение ЦВЗ в файл происходит после этапа квантования. Предложенная в этой работе таблица квантования модифици-

8	6	5	8	1	1	1	1
6	6	7	1	1	1	1	28
7	7	1	1	1	1	35	28
7	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	55	50	52	50

а

2	2	1	2	1	1	1	1
2	2	2	1	1	1	1	28
2	2	1	1	1	1	35	28
2	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	55	50	52	50

б

4	3	2	4	1	1	1	1
3	3	3	1	1	1	1	28
3	3	1	1	1	1	35	28
3	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	55	50	52	50

в

6	4	3	6	1	1	1	1
4	4	5	1	1	1	1	28
5	5	1	1	1	1	35	28
5	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	55	50	52	50

г

Рис. 3. Используемые в алгоритме таблицы квантования:
а – исходная (первая), б – вторая, в – третья, г – четвертая

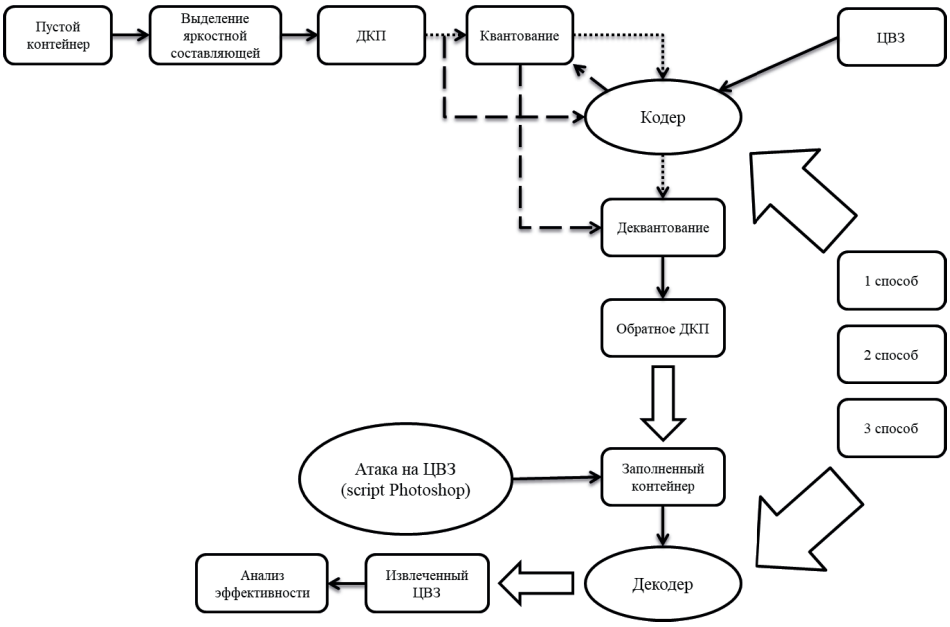


Рис. 4. Общая схема эксперимента для оценки эффективности алгоритмов использования (внедрения) ЦВЗ в цифровое изображение

руется в области низких частот путем умножения на 1/4, 1/2, 3/4. Таким образом, для проведения исследования генерируются четыре различные таблицы квантования (рис. 3).

Описание схемы эксперимента по оценке эффективности алгоритмов использования (внедрения) ЦВЗ в цифровое изображение

Общая схема эксперимента приведена на рис. 4.

В качестве примера ЦВЗ в работе используется логотип Санкт-Петербургского государственного университета аэрокосмического приборостроения. Он имеет размеры 16 на 16 пикселей и является монохромным (рис. 5).

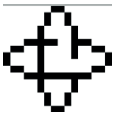


Рис. 5. Используемый ЦВЗ

Во всех рассмотренных ранее алгоритмах ключ для внедрения ЦВЗ генерируется случайным образом и представляет собой последовательность номеров блоков, в которые необходимо внедрить ЦВЗ в изображение.

Эффективность рассматриваемых алгоритмов использования ЦВЗ оценивается по двум критериям.

1. Пиковое отношение сигнала к шуму (далее – PSNR, англ. Peak Signal to Noise Ratio). Приемлемым с точки зрения качества считается значение PSNR, равное 37 дБ и больше.

2. Среднеквадратичная ошибка (далее – MSE, англ. Mean Squared Error).

Сравнительный анализ алгоритмов внедрения цифрового водяного знака

Пример внедрения ЦВЗ тремя алгоритмами представлен на рис. 6.

При внедрении ЦВЗ по алгоритму Кокса на изображении отчетливо видны артефакты, яркость которых увеличивается по мере возрастания параметра α и уровня квантования. Невооруженным взглядом заметно, какие места контейнера были подвержены изменениям. По мере увеличения параметра R хуже извлекается цифровой водяной знак. При внедрении метки по алгоритму Бенхама артефакты также присутствуют, но видны меньше, чем при внедрении алгоритмом Кокса. При внедрении ЦВЗ по алгоритму Сентхурана и Ранатунги визуальных артефактов не возникает. ЦВЗ извлекается безошибочно при использовании таблиц квантования № 1, 3, 4.

При проведении атак на заполненные контейнеры (атаки на ЦВЗ осуществлены с применением программы Photoshop):

– увеличение яркости на 20%;

– уменьшение яркости на 20%;

– размытие по Гауссу на 2%;

– добавление шума по Гауссу на 5%.

ЦВЗ в основном безошибочно извлекается из контейнеров, заполненных по алгоритму Бенхама. Это подтверждает столбчатая диаграмма, приведенная на рис. 7.

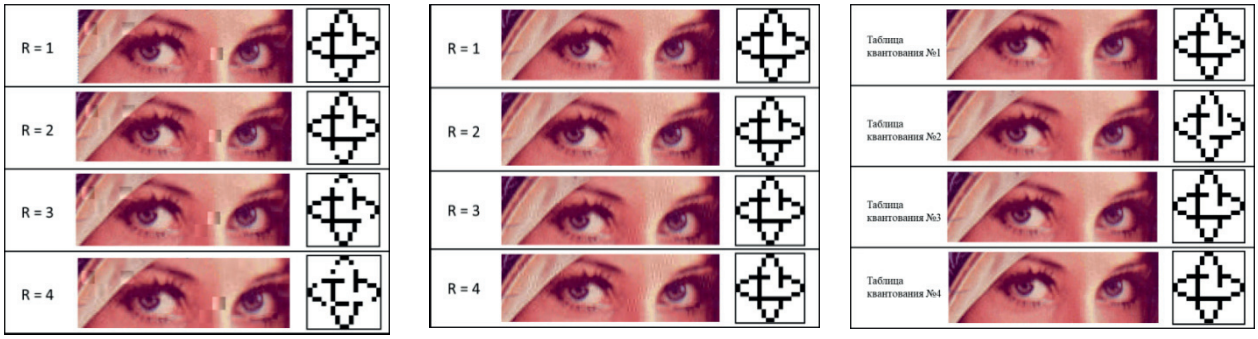


Рис. 6. Пример внедрения ЦВЗ тремя алгоритмами

Заключение

В статье был проведен анализ алгоритмов использования ЦВЗ в сжатом изображении. Результаты показали, что алгоритм Бенхама является наилучшим из рассмотренных в статье с точки зрения критериев эффективности и качества извлекаемой метки, а также более устойчив к атакам на заполненный контейнер, чем другие исследованные алгоритмы.

Библиографический список

1. *Гашицкий М. С., Козина Е. В.* Цифровое пиратство в России: проблемы и пути решений // Форум

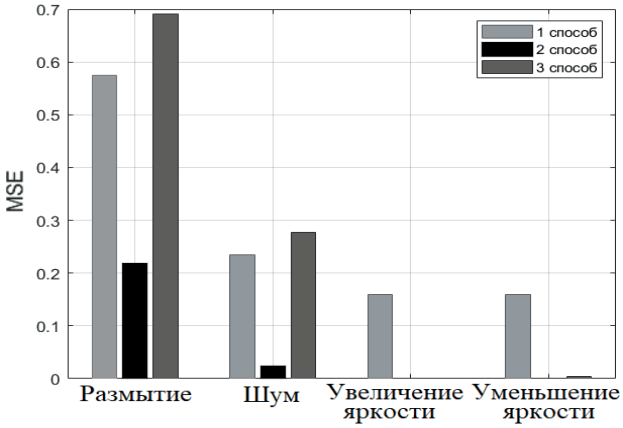


Рис. 7. Значения MSE, полученные в результате извлечения ЦВЗ после проведения серии атак на заполненные контейнеры

молодых ученых: Междунар. науч.-практ. период. сетевое изд. 2020. № 5 (45). С. 107–109.

2. Аналитический обзор // WCIOM: сетевое издание. URL: https://wciom.ru/analytical-reviews/analiticheskii-obzor/zaplati-za-podpisku-smotri-spokojno (дата обращения: 28.04.2021).

3. *Гребенников В.* Стеганография. История тайнописи. М.: Самиздат, 2019. 142 с.

4. *Cox I. J., Kilian J., Leighton F. T., Shamoon T.* Secure spread spectrum watermarking for multimedia // IEEE Transactions on Image Processing. 1997. Vol. 12, iss. 6. P. 1673–1687.

5. *Senthooran V., Ranathunga L.* An Experimental Investigation of Statistical Model based Secure Steganography for JPEG Images // Indian Journal of Science and Technology. 2017. № 10 (27). P. 1–11.

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
В. А. Минаева*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

АНАЛИЗ МЕТОДОВ ОЦЕНКИ РИСКОВ АВТОМАТИЗИРОВАННЫХ СИСТЕМ

Приведено исследование методов оценки рисков автоматизированных систем, позволяющих наиболее детально описать безопасность системы.
Ключевые слова: *методы, оценка риска, отказ, опасность, работоспособность, HAZOP, FMEA, FTA.*

V. G. Eryshov*
PhD, Tech., Associate Professor
V. A. Minaeva *
Student
**St. Petersburg State University of Aerospace Instrumentation*

ANALYSIS OF RISK ASSESSMENT METHODS FOR AUTOMATED SYSTEMS

The article studies the methods for assessing the risks of automated systems, which make it possible to describe the security of the system in the most detailed way.
Keywords: *methods, risk assessment, failure, hazard, operability, HAZOP, FMEA, FTA.*

Обеспечение безопасности на всех этапах работы автоматизированных систем (АС) – важная задача. Сложность и разветвленность этих систем не позволяет найти один способ защиты всех компонентов и процессов. Для обеспечения качественной защиты требуется понимание рисков работоспособности АС предприятия. В данной статье будет рассмотрено несколько методов оценки рисков АС, позволяющих с разных сторон изучить защищенность АС организации.

Оценка риска – это процесс, включающий последовательное выполнение ряда задач: идентификации, анализа и оценки риска [1]. Существует несколько известных методов оценки рисков. Рассмотренные в данной статье методики позволяют провести исследование опасности и работоспособности (HAZOP), проанализировать виды и последствия отказов (FMEA), а также построить дерево неисправностей (FTA).

1. HAZOP (Hazard and Operability Study) – это поиск опасностей и исследование работоспособности, мощный *метод анализа опасностей*. Использование программируемых электронных систем становится все более распространенным, следовательно, есть необходимость в методе HAZOP, который можно эффективно использовать для программного обеспечения.

HAZOP – это полуформализованная командная деятельность, которая систематически рассматривает представление системы и ее рабочих процедур с целью выявления потенциальных опасностей. Метод основан на принципе, что проблема может возникнуть только тогда, когда есть некоторое отклонение от замысла системы, представленной в рассматриваемой модели.

Процедура заключается в поиске представления (элемент за элементом) для каждого мыслимого *отклонения от нормального* функционирования системы путем использования списка *ключевых слов*. Они тщательно подобраны, чтобы побудить открытое, свободное мышление обо всех возможных отклонениях в работе системы. По мере выявления каждого отклонения команда обсуждает потенциальные *причины и последствия*, а также рекомендует соответствующие *меры по исправлению* ситуации.

Можно рассмотреть HAZOP через несколько представлений. Эти различные точки зрения помогут применить его к моделям программной инженерии (Software Engineering Models). Таким образом, формальная модель позволяет исследовать структуру HAZOP, а причинно-след-

ственная модель – интегрировать HAZOP с причинно-следственными методами обеспечения безопасности, такими как анализ отказов и их последствий (FMEA).

Формальная модель

Для описания формальной модели HAZOP вводятся два отношения: понятие, что *цель выражается через свойства,* и понятие, что *свойства вызывают управляющие слова* [2].

Отношения, допускаемые HAZOP, показаны на рис. 1. Данное представление не ограничивает развитие бессмысленных отклонений, поскольку все еще существует возможность того, что цель может ассоциироваться с неподходящими свойствами, а также ключевые слова могут быть вызваны свойствами, которые производят бессмысленные утверждения. Также возможно, что цель будет отражена через свойства, для которых не существует значимого управляющего слова или могут быть свойства, относящиеся к ключевым словам, но не выражающие никакой известной цели.

В общем случае модель можно описать следующим образом: для определенной *цели* существует набор соответствующих *свойств*, которые ассоциируются с этой целью. После того как свойства выбраны, они в свою очередь могут быть связаны с определенным набором *управляющих слов* (рис. 2).

Причинно-следственная модель

Метод исследования опасностей HAZOP рассматривается в контексте с другими методиками анализа отказов и подчеркивает семантику, лежащую в основе отказа. Рис. 3 иллюстрирует основную идею исследования [3].

Каждая авария имеет соответствующую цепочку *событий и состояний*. Одним из возможных инициирующих событий является отказ или неисправность компонента системы. Он может быть внутренним или внешним, случайным или систематическим. Такие неисправности могут вызвать ошибочное внутреннее состояние системы. Эти ошибки могут привести к тому, что система перестанет работать в соответствии со своей спецификацией и, следовательно, возникнет отказ на уровне системы. В результате отказа система может оказаться в опасном состоянии. Если опасное состояние не контролируется, это может привести к аварии. В свою очередь авария может спровоцировать катастрофу.

Механизмы анализа отказов, включая HAZOP, необходимы для того, чтобы связать

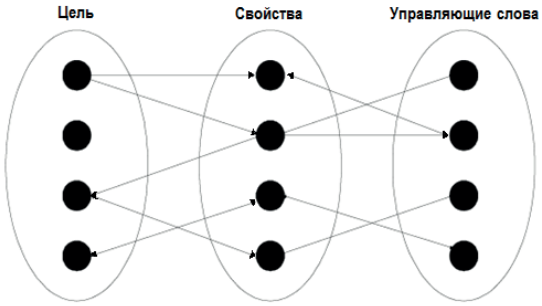


Рис. 1. Отношения HAZOP

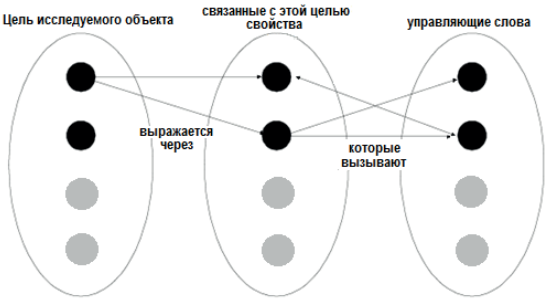


Рис. 2. Описание модели

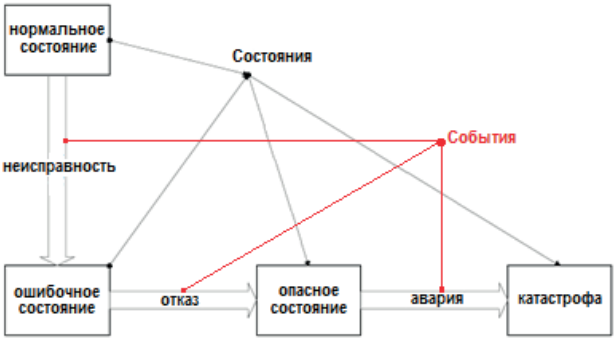


Рис. 3. От неисправности к аварии

представление системы с этой основополагающей цепочкой причинности. Анализ отказов – это в первую очередь искусственное упражнение, определяющее способы, которыми отказ в реальном мире может повлиять на поведение системы. Для того чтобы проследить путь от неисправностей к отказам или наоборот, используются следующие методы: анализ дерева неисправностей (FTA) и анализ режимов и последствий отказов (FMEA). Эти методы показывают представление зависимостей между различными компонентами системы, чтобы можно было проследить причинно-следственные связи между отказом в конкретном компоненте и отказом системы в целом. FTA и FMEA являются аб-

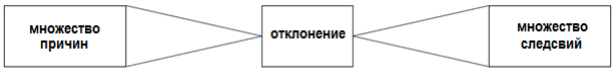


Рис. 4. Простая модель HAZOP

стракциями одной и той же базовой причинно-следственной модели распространения отказа по системе.

FTA и FMEA дополняют друг друга. В частности, FTA начинается с главного события (режим отказа на уровне системы) и определяет ряд потенциальных причин для конкретного последствия. В отличие от этого, FMEA представляет собой анализ «снизу вверх», начиная с отказа на уровне компонента и оценивая последствия.

HAZOP не вписывается в эту простую причинно-следственную модель. Отправная точка для исследования HAZOP – *отклонение от проектного замысла*. После идентификации HAZOP нацелен на определение потенциальных причин (неисправностей) и последствий (отказов на уровне системы) конкретного отклонения (рис. 4).

Представление системы

Для проведения процесса исследования по методике HAZOP необходима вся исходная информация о системе и ее параметрах и характеристиках. Процесс исследования по методу HAZOP анализирует все формы отклонений от эталонов проектов из-за каких-либо недостатков в элементах исследуемого процесса.

Исследование проводится на этапе уточнения конструкции, когда его общая схема уже создана, но можно внести некоторые несущественные изменения. При этом разрешается проведение исследований во время работы этого процесса, что, однако, будет неэффективно с финансовой точки зрения. В результате исследования должен быть сформирован отчет, содержащий: найденные отклонения, их возможные причины и последствия, разработанные рекомендации по устранению найденных проблем и оценку рисков неисправимых отклонений.

Представление системы перед командой HAZOP – ключевой вопрос, который будет контролировать эффективность процесса исследования. Неполная, непоследовательная или двусмысленная модель системы, представленная команде, часто отражает аналогичный уровень понимания разработчиками созданной ими системы. Чтобы команда могла обсуждать поведение модели при отказе, она должна уметь использовать эти многочисленные аспекты системы, видеть основную причинно-следственную цепочку от события отказа до потенциально опасного состояния.

Управляющие слова

В ходе исследования по данному методу выявляют требования к исследуемым процессу или системе, разбивают их на определенные части и осуществляют их анализ с целью обнаружения отклонений от эталонов, причин этих отклонений. Управляющие слова определяют, какие части системы или процесса и как реагируют на отклонения характеристик от эталонных значений. Ключевое слово зависит от уровня абстракции, необходимого для создания отклонения, а также от конкретного свойства, используемого для выражения цели. Например, сравнение не имеет режима отказа в области значений (выдаваемых программой), поэтому такие ключевые слова, как БОЛЬШЕ и МЕНЬШЕ неуместны. Для этого события больше подходят управляющие слова ОШИБОЧНЫЙ или НЕВЕРНЫЙ.

1. FMEA. Метод анализа видов и последствий отказов (Failure Mode Effect Analysis) применяется для обнаружения и идентификации определенных видов отказов и его последствий [4] (рис. 5). Может сопровождаться анализом критичности каждого вида отказа, оцениваемого по качественной, количественной или смешанной шкале (FMECA) [5].

Метод FMEA/FMECA активно задействован в задачах проектирования, производства и эксплуатации системы [6]. Основное использование данного метода хорошо коррелирует с процессами оптимизации конструкции и характеристик системы. Анализ может применяться как к новым процессам и системам, так и к уже существующим.

При составлении FMEA используются следующие параметры:

1) значимость (Severity) **S** – ранг значимости (тяжести) последствий отказа. Устанавливается в диапазоне от 1 до 10;

2) возникновение (Occurrence) **O** – возможность возникновения причины отказа. Устанавливается в диапазоне от 1 до 10;

Рис. 5. Модель FMEA

3) обнаружение (Detection) **D** – возможность обнаружения возникших сбоев и их причин и последствий. Варьируется в диапазоне от 1 до 10 (где 1 – проблема определяется точно, 10 – проблема не определяется);

4) приоритетное число риска (**RPN** – Risk priority number) – произведение рангов S·O·D. Является количественной оценкой риска, но не может быть использовано в качестве единой оценки потенциальных рисков, так как одно и то же значение **RPN** может быть получено в результате различных значений S, O и D;

5) матрицы рисков (Risk Matrix) – механизм оценки критичности отказа, использующий установленные пороговые значения для каждого из рангов, по которому проводится оценка [6].

В процессе анализа FMEA на начальном этапе определяется сам объект анализа. В случае когда объект анализа представляет часть исследуемой системы, то первым делом устанавливаются границы системы. После того как заданы границы исследования, разрабатывается таблица для запоминания его результатов. Если обнаруженные сбои коррелируют с опасными параметрами, то рекомендуется проведение особого контроля этих параметров.

Для каждого элемента составляется список наиболее значимых видов отказов и вероятность их возникновения. Затем для каждого вида отказа определяются все возможные последствия, которые могут произойти, и вероятность возникновения каждого из них.

Далее определяется рейтинг значимости (S) и все потенциальные причины для каждого вида отказа. С этой целью может применяться причинно-следственный анализ причин отказов (например, FTA). Для каждой причины находят рейтинг вероятности ее возникновения (O).

Для каждой причины выбираются существующие методы контроля, которые должны

Рис. 6. Модель FTA

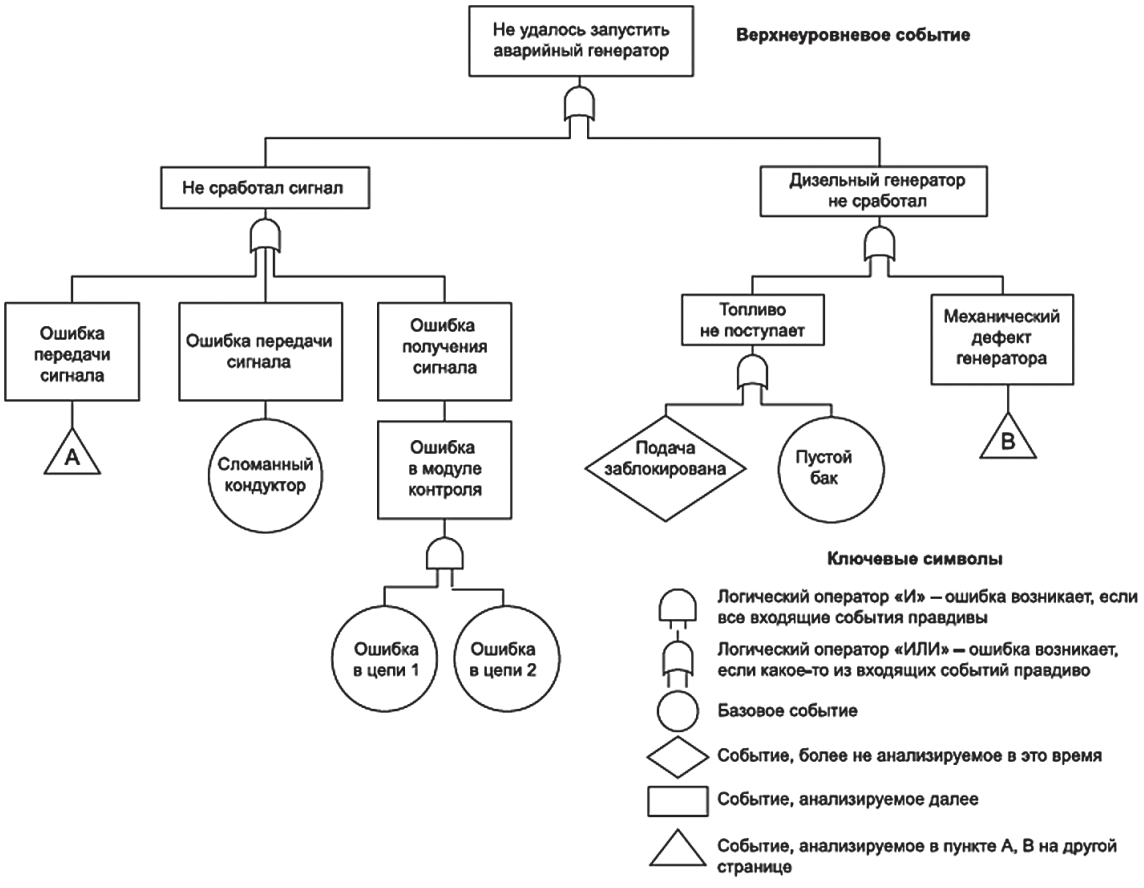


Рис. 7. Пример дерева отказов FTA

предотвращать возникновение причин и снижать вероятность того, что произойдет отказ. После этого для каждого метода контроля определяется рейтинг обнаружения (D).

С целью ранжирования возможных сбоев по их критичности находят приоритетное число риска. Разрабатывается перечень действий, которые необходимы для уменьшения опасности последствий. После внедрения разработанных мероприятий приоритетное число риска рассчитывается вновь.

2. FTA. Анализ дерева отказов (Fault Tree Analysis) – метод, обнаруживающий и идентифицирующий отказы (сбои) системы и показывающий пути их возникновения. Эти пути изображают графически в виде логической древовидной диаграммы. С помощью дерева неисправностей исследуют способы снижения или устранения потенциальных причин отказов (рис. 6) [7].

Событие, которое является вершиной дерева отказов, определяется как результат анализа видов сбоев, их последствий и критичности. Дерево отказов позволяет анализировать одно событие (отказ), поэтому для подробного описания причин отказов системы необходимо строить множество деревьев для каждого отказа (рис. 7).

Использование данного метода не гарантирует достаточности для определения надежности исследуемой системы. Метод FMEA может предусмотреть множество потенциальных негативных событий и их последствий, а FTA дает возможность установить все логические и временные соотношения, описывающие событие отказа, расположенного на вершине дерева. Таким образом, методы FTA и FNEA дополняют друг друга, что позволяет точно описать отказы и причины их возникновения в системах, где чрезвычайно высокие требования к безопасности функционирования всех процессов и компонентов [8].

Заключение

В данной статье были рассмотрены методы оценки рисков ИБ АС, определения опасности и работоспособности ИС, анализа видов отказов и их последствий, анализа дерева отказов. Каждый из этих методов может использоваться самостоятельно, имеет специфику и сферы применения. Например, HAZOP позволяет проводить систематические комплексные исследования сложных систем. Представители проекта HAZOP – это эксперты по различным направлениям деятельности, в том числе имеющие практический опыт работы в различных производствах, что предоставляет возможность наиболее полно рассматривать все потенциальные риски для всей системы и применить меры по предотвращению отказов. В совокупности с анализами FMEA и FTA, которые при совместном использовании комплексно описывают каждый потенциальный отказ системы, его последствия и причины возникновения, получается детализированный анализ безопасности автоматизированной системы, что важно во многих производственных областях.

Библиографический список

1. ГОСТ Р 58771–2019. Менеджмент риска. Технологии оценки риска. URL: https://docs.cntd.ru/document/1200170253 (дата обращения: 19.11.2021).

2. *Hebbron B. D.* Applying HAZOP to Software Engineering Models. URL: https://www.researchgate.net/publication/2648980_Applying_HAZOP_to_Software_Engineering_Models (дата обращения: 19.11.2021).

3. ГОСТ Р 27.012–2019. Надежность в технике. Анализ опасности и работоспособности (HAZOP). URL: https://docs.cntd.ru/document/1200170007 (дата обращения: 19.11.2021).

4. ГОСТ Р 51901.12–2007. Менеджмент риска. Метод анализа видов и последствий отказов. URL: https://docs.cntd.ru/document/1200062125 (дата обращения: 19.11.2021).

5. Менеджмент качества. FMEA анализ. URL: https://www.kpms.ru/Implement/Qms_FMEA.htm (дата обращения: 19.11.2021).

6. Что такое FMEA анализ? // B□hme & Weihs: офиц. сайт. URL: https://boehme-weihs.ru/resursy/statii/chto-takoe-fmea-analiz/ (дата обращения: 19.11.2021).

7. Анализ дерева отказов (Fault tree analysis (FTA) // Портал знаний. URL: http://statistica.ru/knowledge-clusters/technical-sciences/analiz-dereva-otkazov/ (дата обращения: 19.11.2021).

8. Идентификация и оценка рисков. Анализ дерева отказов (Fault tree analysis, FTA) // Лаборатория «Статистическое знание». URL: http://best-stat.ru/risk-menedzhment/identifikatsiya-i-otsenka-riskov-6-analiz-dereva-otkazov-fault-tree-analysis-fta.html (дата обращения: 19.11.2021).

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
О. А. Нестеренков*
студент
А. А. Чемоданов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОПИСАНИЕ ВЕКТОРОВ КОМПЬЮТЕРНЫХ АТАК (OWASP, CWE, CAPEC, ATT&CK, WASC, STIX/TAXII)

Приведено описание наиболее известных векторов компьютерных атак, используемых для оценки угроз безопасности информации.
Ключевые слова: *вектор атаки, угроза, уязвимость, OWASP, CWE, CAPEC, ATT&CK, WASC STIX/TAXII.*

V. G. Eryshov*
PhD, Tech., Associate Professor
O. A. Nesterenkov*
Student
A. A. Chemodanov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

DESCRIPTION OF VECTORS OF COMPUTER ATTACKS (OWASP, CWE, CAPEC, ATT&CK, WASC, STIX/TAXII)

The article contains a description of the most well-known vectors of computer attacks used to assess threats to information security.
Keywords: *attack vector, threat, vulnerability, OWASP, CWE, CAPEC, ATT&CK, WASC STIX/TAXII.*

Оценка угроз безопасности информации (УБИ) проводится в целях определения актуальных УБИ, реализация которых возможна в системах и сетях в процессе их работы. Исходными данными для оценки УБИ могут выступать банк данных угроз ФСТЭК России [1], техническое задание на создание системы или сети, договоры и соглашения предприятий, а также нормативно-правовые акты. Кроме этого, исходными данными являются векторы компьютерных атак, разработанные различными коммерческими организациями. В данной статье будут рассмотрены наиболее известные шаблоны компьютерных атак.

Вектор (шаблон) атаки – это способ, метод или средство, при помощи которого киберпреступники проникают в целевую систему – взламывают ее. *Вектор атаки* определяет путь, которым злоумышленники доставляют полезную нагрузку на компьютер [2].

I. OWASP

OWASP (Open Web Application Security Project) – это всемирная организация, которая занимается разработкой методологии в области обеспечения и повышения безопасности веб-приложений [3].

Первая версия рейтинга OWASP Top 10 появилась в 2004 г. Версия рейтинга, представленная сообществу в 2021 г., претерпела изменения (рис. 1) [4].

Рейтинг OWASP Top 10 составляется на основе данных, предоставляемых сообществом экспертов отрасли информационной безопасности (ИБ). Для каждого пункта представлены общая информация, описание, рекомендации по предотвращению, примеры соответствующих атак и полезные ссылки [5].

1. **A1:2021 Нарушение контроля доступа**

Под контролем доступа понимается наложение ограничений для аутентифицированных

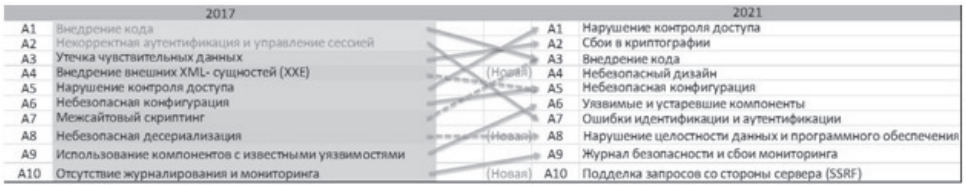


Рис. 1. OWASP Top 10, изменения 2021 г.

пользователей на выполнение действий, выходящих за рамки их уровня разрешений.

2. **A2:2021 Сбои в криптографии**

Криптографические сбои относятся к проблемам с криптографией или к отсутствию криптографии в целом.

3. **A3:2021 Внедрение кода**

Вредоносный код вставляется злоумышленником и может поставить под угрозу данные или все приложение. Наиболее распространенными атаками подобного типа являются SQL-инъекции, межсайтовый скриптинг (XSS) атаки, CCS-инъекции и др.

4. **A4:2021 Небезопасный дизайн**

Эта категория уязвимостей ориентирована на риски, связанные с недостатками дизайна и архитектуры. Как пояснил OWASP, они отличаются от рисков, связанных с недостатками в реализации.

5. **A5:2021 Небезопасная конфигурация**

Неправильная конфигурация безопасности относится к элементам управления безопасностью, которые не защищены или неправильно настроены.

6. **A6:2021 Уязвимые и устаревшие компоненты**

Эта категория ранее называлась «Использование компонентов с известными уязвимостями».

7. **A7:2021 Ошибки идентификации и аутентификации**

При некорректной аутентификации и идентификации функции приложений некорректно воплощаются в аутентификации и управления сеансами, что приводит к уязвимостям и использованию нарушителем ИБ ключей, паролей и токенов сеанса.

8. **A8:2021 Нарушение целостности данных и программного обеспечения**

Эта новая категория в списке OWASP относится к уязвимостям в обновлениях программного обеспечения, критических данных и конвейерах CI/CD, целостность которых не проверена.

9. **A9:2021 Журнал безопасности и сбои мониторинга**

Эта категория была расширена за счет включения большего количества типов сбоев. Несмотря на то, что регистрацию и мониторинг сложно протестировать, она важна, поскольку сбои могут повлиять на подотчетность, видимость.

10. **A10:2021 Подделка запросов со стороны сервера (SSRF)**

Данные проблемы возникают, когда веб-приложение не проверяет предоставленный пользователем URL-адрес при выборке удаленного ресурса.

OWASP Top 10 используется в множестве организаций, связанных с безопасностью веб-приложений.

II. CWE

CWE (Common Weakness Enumeration) – перечень уязвимостей и недостатков ИБ программного обеспечения, представленный в иерархическом виде. CWE активно поддерживается MITRE и бурно развивается при поддержке мирового сообщества специалистов ИБ. CWE необходима для стандартизации методов оценки ИБ разрабатываемых программных продуктов.

III. CAPEC

CAPEC (Common Attack Pattern Enumeration and Classification) рассматривает шаблоны атак как характеристику общих методов, применяемых при кибератаках на уязвимые элементы ИС. В CAPEC применяется похожий на CWE иерархический метод. Приведены два основных представления и несколько вспомогательных.

IV. WASC

WASC (The Web Application Security Consortium) – это всемирное сообщество, которое ранее достаточно плотно занималось созданием стандартов ИБ. В настоящее время не демонстрирует новых результатов. В WASC Threat Classification показаны недостатки и классы кибератак, которые используются для взлома веб-приложений и другой ценной информации.

V. ATT&CK

Mitre ATT&CK (Adversarial Tactics, Techniques & Common Knowledge — «тактики, техники и сведения о нарушителях ИБ») – обширная база данных организации Mitre о характеристиках тактик, техник и методов нарушения ИБ, применяемых нарушителями [6].

Данные базы Mitre составлены в виде структурированных матриц, каждая из которых построена в виде таблицы, в которой заголовки столбцов определяются как тактики кибернарушителей, а внутреннее содержимое ячеек – соответствующие им методики реализации этих атак, так называемые техники [7]. Матрицы Mitre объединяются в четыре основные совокупности: PRE-ATT&CK, Enterprise, Mobile, ATT&CK for ICS. Кроме них, в базе Mitre представлены совокупности техник, используемые существующими APT-группировками.

VI. STIX и TAXII

STIX и TAXII – стандарты, разработанные для повышения эффективности предотвращения и смягчения последствий кибератак. STIX устанавливает предмет анализа угроз, а TAXII определяет порядок распространения такой информации. STIX и TAXII являются машиночитаемыми и поэтому легко автоматизируются.

STIX (Structured Threat Information eXpression) – стандартизированный язык, разработанный MITRE и техническим комитетом OASIS Cyber Threat Intelligence для представления информации о киберугрозах.

TAXII (Trusted Automated eXchange of Intelligence Information) рассматривает основные способы обмена данными об УБИ через службы и средства обмена сообщениями. Разработан специально для обеспечения поддержки информации STIX. Три основные модели TAXII [8]:

– «Звезда» – один репозиторий информации;

– «Источник/абонент» – один источник информации;

– «Равноправные узлы» – обмен информацией между несколькими группами.

Заключение

В статье были рассмотрены известные векторы компьютерных атак. Некоторые из них, такие как OWASP Top 10, представляют собой рейтинг наиболее опасных УБИ и могут использоваться для начального этапа оценки ИБ или в качестве маркетингового хода. Многие шаблоны атак основываются на представлении CWE, так как этот перечень наиболее структурированный и полный. С помощью Mitre ATT&CK можно проанализировать весь путь осуществления угрозы, от разведки до действий злоумышленника по реализации атаки. STIX и TAXII позволяют представить данные в машиночитаемом виде и тем самым упростить анализ.

Библиографический список

1. Банк данных угроз безопасности информации // ФСТЭК России. URL: https://bdu.fstec.ru/threat (дата обращения: 08.12.2021).

2. Вектор атаки // ИТ-энциклопедия «Касперского». URL: https://encyclopedia.kaspersky.ru/glossary/attack-vector/ (дата обращения: 08.12.2021).

3. Общий обзор классификаций угроз безопасности: OWASP, CWE, CAPEC, WASC // Информационная безопасность для пользователей и специалистов: портал. URL: https://safe-surf.ru/specialists/article/5210/595970/ (дата обращения: 08.12.2021).

4. The OWASP Top 10 2021: офиц. сайт. URL: https://owasp.org/Top10/ (дата обращения: 08.12.2021).

5. OWASP top 10 2021 – the ultimate vulnerability guide // Crashtest Security: офиц. сайт. URL: https://crashtest-security.com/owasp-top-10-2021/ (дата обращения: 08.12.2021).

6. MITRE ATT&CK: офиц. сайт. URL: https://attack.mitre.org/ (дата обращения: 08.12.2021).

7. Что такое база знаний MITRE ATT&CK и зачем она нужна // Anomali: офиц. сайт. URL: https://www.anomali.com/ru/resources/what-mitre-attck-is-and-how-it-is-useful (дата обращения: 08.12.2021).

8. Что такое STIX/TAXII// Anomali: офиц. сайт. URL: https://www.anomali.com/ru/resources/what-are-stix-taxii (дата обращения: 08.12.2021).

УДК 004.421.5 **DOI: 10.31799/978-5-8088-1701-2-2022-2-226-228**

В. Г. Ерышов*
кандидат технических наук, доцент
Б. С. Шром*
магистрант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ГЕНЕРИРОВАНИЕ ПСЕВДОСЛУЧАЙНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ НА МОБИЛЬНОМ УСТРОЙСТВЕ

Рассмотрены датчики для генерации псевдослучайных последовательностей, проведено исследование последовательностей, полученных с этих датчиков, в плане их сходства со случайной последовательностью.
Ключевые слова: *криптография, шифрование, последовательность.*

V. G. Eryshov *
PhD, Tech., Associate Professor
B. S. Shrom *
Postgraduate Student
**St. Petersburg State University of Aerospace Instrumentation*

GENERATING A PSEUDO-RANDOM SEQUENCE ON A MOBILE DEVICE

In this article, we considered sensors for generating pseudo-random sequences, and also studied the sequences obtained from these sensors for their similarity to a random sequence.
Keywords: *cryptography, encryption, sequence.*

В современном мире у каждого человека есть смартфон, при помощи которого он может общаться с другими людьми или системами. Для того чтобы защитить передаваемую информацию, может потребоваться сгенерировать псевдослучайную последовательность, статистически не отличимую от случайной, которая будет выступать в качестве ключевой последовательности. Для генерации псевдослучайной последовательности можно использовать сигналы, снятые с физических датчиков устройства. В статье будет рассмотрен этот процесс.

Выбор физических датчиков

Физические датчики собирают различную информацию об окружающей среде, которую в некоторых случаях бывает достаточно сложно предугадать. Современные смартфоны обладают различными датчиками с разными параметрами, но в большинстве имеются следующие [1]:

1) камера – используется для создания фотографий и видеозаписей. Является самым сложным и комплексным датчиком в современных смартфонах. В большинстве современных смартфонов используются две и более камеры;

2) микрофон – используется для считывания звуковых колебаний. Операционная система Android позволяет приложению получать необработанный поток данных с микрофона в режиме реального времени, как и для всех датчиков далее;

3) акселерометр – измеряет ускорение в пространстве по трем осям. Служит для ориентации устройства в пространстве;

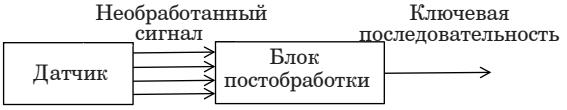
4) гироскоп – измеряет скорость вращения устройства по трем осям. Служит для ориентации устройства в пространстве;

5) датчик освещения – измеряет интенсивность света напротив экрана устройства. Служит для динамического изменения яркости экрана;

6) датчик расстояния – измеряет расстояние объектов до экрана устройства. Необходим для блокировки экрана устройства во время разговора по телефону;

7) датчик магнитных полей – измеряет напряженность магнитного поля по трем осям. Может служить для позиционирования устройства, а также для расчета азимута;

8) барометр – измеряет атмосферное давление. Может служить для определения изменения высоты устройства.



Постобработка сигнала

Микрофон, акселерометр и гироскоп позволяют получать потоковые данные наибольшей длины в единицу времени, а также дают возможность пользователю влиять на показания датчиков. Они и будут использоваться как источник случайных данных.

Создание ключевого потока

Ключевой поток должен обладать равномерным случайным распределением [2], однако сигнал, снятый с датчика, часто им не обладает. Для создания ключевой последовательности поток данных с датчика будет подвергнут постобработке (рисунок).

Смысл постобработки заключается в том, что на выходе блока постобработки оказывается менее длинная, чем на входе, последовательность бит, однако обладающая более высокой степенью равномерности. В качестве постобработки к данным от физического датчика будет применена хэш-функция SHA-256 (Secure Hash Algorithm).

Описание методов тестирования

Ключевые последовательности должны быть неотличимы от случайных последовательностей. У них не должно быть статистических изъянов, которым не подвержены случайные последовательности, а также ключевые последовательности не должны быть предсказуемыми.

Пакет тестов NIST STS

Пакет тестов NIST STS (National Institute of Standards and Technology Statistical Test Suite) был разработан в США в 1999 г. В настоящее время он состоит из 15 тестов и находится в свободном для использования доступе. Каждый из 15 тестов вычисляет статистику, выявляющую один из присущих неслучайным последовательностям дефектов [3]. За основу пакета тестов взяты идеи, описанные Д. Кнутом [4], а также тесты, включенные в пакет DIEHARD.

Алгоритм Берлекэмпа – Мэсси

В тесте Linear Complexity Test от NIST фигурирует понятие линейной сложности. Линейная сложность последовательности – это длина регистра сдвига с линейной обратной связью минимальной длины, генерирующего данную последовательность.

Алгоритм Берлекэмпа – Мэсси предназначен для поиска регистра сдвига с линейной обратной связью минимальной длины, генерирующего заданную последовательность. Данный алгоритм рассматривает входную последовательность поэлементно, начиная с первого элемента последовательности, при этом алгоритм составляет регистр сдвига с линейной обратной связью минимальной длины. При рассмотрении следующего элемента последовательности составленный регистр может изменить свою конфигурацию и длину.

Автокорреляционный тест

Тестирование последовательности данным способом позволяет выявить наличие несвойственной для случайной последовательности автокорреляции.

Имея двоичную последовательность $X = \{x_0, x_1, \ldots, X_{N-1}\}$ и последовательность

$$X' = \left\{x'_{(0+t)modN}, x'_{(1+t)modN}, \ldots, x'_{(N-1+t)modN}\right\},$$

циклически сдвинутую на t позиций, можно получить значение функции автокорреляции от t, которая показывает количество совпавших элементов исходной и циклически сдвинутой на t позиций последовательностей:

$$f(t) = \sum_{i=0}^{N-1} x_i \oplus x'_{(i+t)modN}.$$

Тогда количество не совпавших элементов можно найти, вычтя $f(t)$ из N.

Разность между количеством совпавших и несовпавших бит исходной и сдвинутой функции должно укладываться в статистическую погрешность. Формула для вычисления минимального количества числа испытаний N для достижения заданной точности ε выглядит следующим образом [5]:

$$N = \frac{9}{4\varepsilon^2}. \quad (1)$$

Из этой формулы можно получить выражение для нахождения точности ε от числа испытаний N:

$$\varepsilon = \frac{3}{2\sqrt{N}}. \quad (2)$$

Тогда для прохождения автокорреляционного теста необходимо, чтобы доля разности со-

впавших и несовпавших бит от общей длины последовательности не превышала заданный для случайной последовательности уровень точности ε. Иными словами, для прохождения автокорреляционного теста необходимо выполнение неравенства

$$\frac{|N-2f(t)|}{N} < \varepsilon,\ t \in [1;\ N) \quad \Rightarrow$$
$$\frac{|N-2f(t)|}{N} < \frac{3}{2\sqrt{N}},\ t \in [1;\ N). \qquad (3)$$

Сжатие последовательностей

Так как случайную последовательность невозможно значительно сжать без потери информации [6], то, подвергнув сжатию исследуемую последовательность, можно проверить ее статистические свойства. Для проведения процедуры сжатия был выбран архиватор WinRAR (https://www.win-rar.com/).

Результаты тестирования

Оценка скорости генерации ключевых последовательностей была проведена 5 раз. Средняя скорость генерации для микрофона равна 745 байт/с, для акселерометра – 199, а для гироскопа – 553 байт/с.

В ходе проведения тестов из пакета NIST STS для ключевых последовательностей со всех трех датчиков были получены следующие результаты: ключевая последовательность, полученная с использованием акселерометра, успешно прошла все тесты, что может свидетельствовать о ее статистической неотличимости от случайной последовательности; ключевые последовательности, полученные с использованием гироскопа и микрофона, успешно прошли большинство тестов, что также говорит об их схожести со случайными последовательностями. Однако проверить эти последовательности тестами Random Excursions Test и Random Excursions Variant Test не удалось. Это может быть связано с недостаточной длиной входных данных или с несбалансированным двоичным деревом, которое строится в ходе выполнения тестирования [7].

Полученные с помощью алгоритма Берлекэмпа – Мэсси профили линейной сложности для каждой тестируемой последовательности близки к прямой $n/2$, что свидетельствует о невозможности построения регистров сдвига с линейной обратной связью, воспроизводящих данные последовательности с длиной менее $n/2$. Данные результаты согласуются с проведенным ранее тестом на линейную сложность из пакета тестов NIST STS.

В ходе автокорреляционного теста выяснилось, что все тестируемые ключевые последовательности успешно проходят его.

В ходе попыток сжать ключевые последовательности с разных датчиков было обнаружено, что сжать ключевые последовательности не удалось ни на байт, что может свидетельствовать о статистической независимости элементов этих последовательностей. Данные результаты согласуются с проведенным ранее универсальным тестом Маурера из пакета тестов NIST STS.

Заключение

В статье были сгенерированы псевдослучайные последовательности при помощи выбранных физических датчиков. Они были протестированы с использованием тестов, входящих в пакет NIST STS и автокорреляционным тестом. Так же была предпринята попытка сжать эти последовательности с использованием архиватора WinRAR и были построены профили линейной сложности для последовательности, полученной с каждого датчика. Тесты показали, что все ключевые последовательности схожи по своим характеристикам со случайными последовательностями.

Была протестирована скорость генерации ключевых последовательностей с использованием микрофона, гироскопа и акселерометра. Выяснилось, что скорость генерации ключевого потока во всех случаях достаточно низкая, однако быстрее всего ключевой поток генерируется с использованием микрофона (750 байт в секунду).

Библиографический список

1. *Дейтел П., Дейтел Х., Уолд А.* Android для разработчиков. СПб.: Питер, 2016. 512 с.

2. Криптографическая защита информации / А. В. Яковлев, А. А. Безбогов, В. В. Родин, В. Н. Шамкин. Тамбов: Изд-во ТГТУ, 2006. 140 с.

УДК 004.056.53

В. Г. Ерышов*
кандидат технических наук, доцент
В. И. Юдина*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИНФОРМАЦИОННЫЕ РИСКИ. КЛАССИФИКАЦИЯ, ОСНОВНЫЕ ЭТАПЫ ОЦЕНКИ. МЕТОДИКИ АНАЛИЗА И УПРАВЛЕНИЯ РИСКАМИ

Рассмотрены классификация и основные этапы оценки рисков. Проанализированы методики управления рисками.
Ключевые слова: *информационные риски, классификация, методологии управления, этапы оценки рисков.*

V. G. Eryshov*
PhD, Tech., Associate Professor
V. I. Yudina*
Student
**St. Petersburg State University of Aerospace Instrumentation*

INFORMATIONAL RISKS. CLASSIFICATION, MAIN STAGES OF ASSESSMENT. RISK ANALYSIS AND MANAGEMENT TECHNIQUES

The article discusses the classification and the main stages of risk assessment. Risk management techniques are analyzed.
Keywords: *information risks, classification, management methodologies, stages of risk assessment.*

Введение

Понятие «информационный риск» нашло в наше время широкое применение, но точного определения, принятого большинством ученых и практиков, до сих пор не получило. В аспекте информационной безопасности под риском можно понимать потенциальную возможность использования уязвимостей системы для реализации угрозы, которая нанесет ущерб владельцу системы, а также нарушит одну или несколько базовых характеристик безопасности – конфиденциальность, целостность, доступность.

Классификация рисков

Анализ и управление рисками, а также их оценка невозможны без классификации рисков. Существует множество критериев, которые являются основой для классификации, выделим основные из них:

– по источнику (внешний и внутренний);

– по области применения (информационный, экономический и т. д.);

– по степени опасности;

– по продолжительности влияния;

– по объему.

Преимущественно для классификации рисков выделяют внешние и внутренние риски.

К внешним рискам можно отнести:

– техногенные;

– природно-естественные;

– социально-политические;

– финансово-экономические.

К внутренним рискам можно отнести:

– связанные с сотрудниками (человеческим фактором);

– связанные с управлением корпоративной инфраструктурой;

– технические;

– технологические.

Если со стремительным развитием технологий в сфере информационной безопасности количество рисков технического происхождения уменьшилось, то число инцидентов, в основе которых лежит человеческий фактор до сих пор велико, поэтому важно уделять должное внимание не только технической составляющей, но и подбору сотрудников и организационным аспектам.

Как ранее говорилось, риски несут за собой нарушение основных аспектов информационной безопасности – конфиденциальности, целостности и доступности. От чего же может зависеть возникновение рисков, приводящих к нарушению каждого из этих аспектов?

Под доступностью понимается возможность получения доступа к данным в предусмотренное на это время. Беспрепятственное получение необходимой информации в свою очередь зависит от работоспособности системы, ее загруженности и используемых каналов передачи, поэтому причиной возникновения группы рисков, связанной с нарушением доступности, могут быть сбои в программном обеспечении и неисправности оборудования, которые могут позволить реализовать сетевую атаку.

Под целостностью информации понимается невозможность ее изменения или удаления лицами, не имеющим на это прав. Риски, связанные с нарушением целостности, могут быть вызваны отказом программного обеспечения или оборудования, ненадежностью используемых средств разграничения доступа к данным.

Под конфиденциальностью понимается уровень защиты информации от несанкционированного доступа. Риски, связанные с несанкционированным доступом к информации, могут возникнуть в результате несовершенства алгоритмов аутентификации пользователей.

Все описанные выше группы рисков могут возникнуть не только по причине технических неполадок или несовершенства используемой информационной системы, а также в результате несоблюдения стандартов и требований, которые должны использоваться при проектировании систем.

Методики управления рисками. Этапы оценки.

Поскольку проблемы, связанные с информационными рисками, остро стоят на повестке дня уже не первое десятилетие, создано множество систем управления рисками, задача каждой из которых заключается в оценке рисков по различным параметрам, а также в дальнейшей выработке рекомендаций по их обработке и устранению.

Все известные методики управления рисками можно условно разделить на три группы:

– основанные на количественной оценке рисков;

– основанные на качественной оценке рисков;

– использующие смешанные оценки.

Количественная оценка рисков применяется в ситуациях, когда возможно сопоставить исследуемые риски с количественными значениями, выраженными в деньгах, процентах, времени и т. д. Данный способ позволяет точно и, что не менее важно, с помощью численных значений наглядно сравнить риски, он не учитывает причины их возникновения и последствия, которые они несут.

Основные этапы количественного метода оценки рисков [1].

1. Определение ценности активов в установленном количественном показателе.

2. Количественная оценка потенциального ущерба в случае реализации угроз, предшествующих возникновению рисков.

3. Вычисление вероятности каждой из угроз.

4. Определение общего потенциального ущерба по каждой угрозе за установленный промежуток времени (например, за 1 год).

5. Анализ полученных результатов.

В результате анализа по каждой из выявленных угроз необходимо принять решение о минимизации, принятии или переносе риска.

Под минимизацией понимается усиление мер по защите информации (покупка дополнительных средств защиты, проведение инструктажа персонала). При этом необходимо соотнести затраты на повышение безопасности с размером ущерба, который мог принести минимизируемый риск, и вычислить эффективность внедряемых мер.

Принятие риска означает, что работы по его устранению вестись не будут. Как правило, это применимо к угрозам с малым ущербом или маленькой вероятностью появления.

Перенос риска означает, что последствия от его возникновения перекладываются на третье лицо.

Поскольку может возникнуть ситуация, не требующая оценки конкретных качественных показателей, которые в основном служат для упорядочивания рисков, существует метод разделения рисков на допустимые и недопустимые, при этом допустимость рисков оценивается конкретным человеком, как правило главой отдела по информационной безопасности.

При качественном методе оценки рисков количественные показатели не используются. Вместо этого объектам оценки присваиваются значения в соответствии с установленной экспертной шкалой (например, одно из значений: низкий, средний, высокий; или числовое значение по десятибалльной шкале), соответствующие последствиям риска, его вероятности и уровню. Преимущество данного метода заклю-

чается в том, что он позволяет при минимальных затратах ресурсов определить максимальное количество рисков.

Основные этапы качественного метода оценки рисков [1].

1. Определение ценности активов в соответствии с установленной экспертной шкалой.

2. Вычисление вероятности возникновения угрозы, предшествующей каждому из рисков.

3. Выявление уровня успешной реализации угрозы в условиях текущего состояния информационной безопасности.

4. Вычисление уровня риска на основе ценности актива, с которым он связан, и вероятности предшествующей ему угрозы.

5. Анализ полученных по каждой угрозе и риску данных.

При качественном методе оценки рисков, как правило, оперируют понятием «приемлемый уровень риска», и задача проведенной оценки заключается в том, чтобы на основании полученных в результате анализа данных предложить меры повышения защиты информационный безопасности, позволяющие понизить уровень риска до приемлемого.

Несмотря на то что качественный подход позволяет проанализировать причину возникновения и последствия рисков, шкала, используемая при оценке, субъективна, и могут возникнуть трудности на этапе сравнения показателей.

Обзор известных методик

Одна из популярных методик, основанных на качественном подходе, – методика CRAMM (CCTA Risk Analysis and Management Method), разработанная Службой безопасности Великобритании в 1985 г.

Процесс управления рисками по данной методике состоит из следующих этапов [2].

1. Инициирование. Опрашиваются стейкхолдеры, и на основании полученных данных формируется описание рассматриваемой области, ее границы и состав вовлеченных лиц.

2. Идентификация и оценка ИТ-активов. Выделяются три вида активов: данные, ПО и физические активы. Для каждого из активов оценивается критичность, а также последствия от нарушения его целостности, конфиденциальности и доступности.

3. Оценка угроз и уязвимостей. Данный этап выполняется только для наиболее критичных ИТ-активов.

4. Вычисление риска по формуле

$$\text{Риск} = \text{Р (реализации)} \times \text{Ущерб}.$$

При этом вероятность реализации риска вычисляется по формуле

$$\text{Р (реализации)} = \text{Р (угрозы)} \times \text{Р (уязвимости)}.$$

5. Управление риском. Определяется набор мер, необходимых для обеспечения информационной безопасности.

Данная методика признается международными институтами и не раз доказывала на практике свою эффективность. Но она обладает рядом недостатков:

– сложный и трудоемкий процесс сбора данных;

– большие затраты ресурсов на этапе анализа рисков и управления ими;

– невозможность оценить риски в денежном эквиваленте.

Еще одна известная методология – COBIT for Risk, разработана ассоциацией ISACA (Information Systems Audit and Control Association) в 2013 г. Методология рассматривает риски информационной безопасности применительно к рискам основной деятельности организации, описывает подходы к реализации функции управления рисками информационной безопасности в организации и к процессам качественного анализа рисков информационной безопасности и управления ими [2].

В основе указанного подхода лежит построение сценариев, описывающих появление рисков. Методология содержит более ста сценариев, охватывающих разные категории воздействия (геополитика, окружающая среда, управление жизненным циклом ИС и т. д.). Для каждого сценария описываются тип угрозы, тип ее источника, событие, приводящее к возникновению угрозы, а также типы активов, на которые она влияет. В результате анализа каждого риска выдвигается одно из решений: избегание, принятие, передача или снижение риска.

Недостаток методологии – необходимость вовлечения большого количества заинтересованных лиц и сбора большого объема данных, используемых для анализа рассматриваемой области, а также, аналогично методике CRAMM, данный подход не позволяет оценить риски в денежном эквиваленте, что может потребоваться для обоснования внедряемых мер защиты.

Заключение

В статье были проанализированы методологии управления рисками, рассмотрены основные этапы качественной и количественной

оценки рисков, а также проведен обзор двух известных методологий: CRAMM и COBIT for Risk. Каждый из методов обладает достоинствами и недостатками, поэтому наиболее эффективно будет их совместное применение на практике, что позволит обеспечить комплексный подход к решению проблем управления рисками.

Библиографический список

1. Методика оценки рисков информационной безопасности. Вопросы управления рисками информационной безопасности // Leally. URL: https://leally.ru/download-soft/metodika-ocenki-riskov-informacionnoi-bezopasnosti-voprosy/ (дата обращения: 24.11.2021).

2. *Коротнев К.* Методики управления рисками информационной безопасности и их оценки (часть 1). 2020 // Безопасность пользователей в сети Интернет: портал. URL: https://safe-surf.ru/specialists/article/5193/587932/ (дата обращения: 24.11.2021).

УДК 004.05 **DOI: 10.31799/978-5-8088-1701-2-2022-2-233-235**

Н. В. Ерышов*
студент
В. С. Коломойцев*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРОГНОЗИРОВАНИЮ РИСКОВ НА ОСНОВАНИИ ТРЕБОВАНИЙ ГОСТ Р 59339–2021

Предложены методические указания по прогнозированию рисков. Методика описывает типовые действия в процессе управления рисками для системы на основании требований ГОСТ Р 59339–2021. Показан перечень элементов системы, подлежащих прогнозированию рисков. Определен порядок их прогнозирования. Рассмотрены некоторые из ключевых процессов управления рисками.
Ключевые слова: *риск, угроза, система, защита информации, системные процессы, управление рисками.*

N. V. Eryshov*
Student
V. S. Kolomoitcev*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

METHODOLOGICAL GUIDELINES FOR RISK FORECASTING BASED ON THE REQUIREMENTS OF GOST R 59339–2021

The paper offers methodological guidelines for risk forecasting. The methodology describes typical actions for the system in the risk management process based on the requirements of GOST R 59339–2021. The list of system elements subject to risk forecasting is shown. The order of risk forecasting is defined. Some of the key risk management processes are considered.
Keywords: *risk, threat, system, information protection, system processes, risk management.*

Проблема защиты информации в процессе управления рисками для системы актуальна в современных реалиях. Развитие технологий привело к тому, что любое производство вне зависимости от отрасли, будь то оказание услуг или любая иная сфера деятельности человека, подвергается всесторонним опасностям [1]. Для защиты от наиболее вероятных угроз могут применяться, например, экспертные системы оценки рисков.

Экспертная система оценки рисков – это система, которая на вход получает информацию об объекте защиты, анализирует ее и выдает рекомендации или готовую к исполнению инструкцию [2]. С ее помощью можно построить эффективную модель защиты информации в процессе управления рисками.

Государственный стандарт (ГОСТ) – это один из основных, главных стандартов в Российской Федерации. Нормативно-технический документ регламентирует комплекс норм, требований и правил к объекту стандартизации. Такие стандарты применимы как на нормы, так и на материальные предметы. На основании ГОСТ Р 59339–2021 можно проанализировать риски для построения модели защиты информации [3]. В нем описываются положения системного анализа для процесса управления рисками для системы применительно к вопросам защиты информации в системах различных областей приложения.

Информация о рисках может быть использована для решения проблем обеспечения безопасности информации в организации [4]. На основании рекомендаций, полученных в результате работы программы, компетентный сотрудник вправе принять решение о включении риска в модель защиты информации или исключении из нее, важности защиты объектов от этой угрозы и необходимости защиты с разной интенсивностью. Результат работы программы – это рекомендация, основанная на требованиях ГОСТ Р 59339–2021, т. е. действия, не обязательные к исполнению, но основанные на по-

лученных от пользователя данных об объектах информатизации в организации (наличие и количество серверов, автоматизированных рабочих мест, маршрутизаторов и коммутаторов и т. д.) [4, 5].

Методические указания по прогнозированию рисков определяют типовые действия в процессе управления рисками для системы [6]. При этом риски характеризуют прогнозными вероятностными значениями в сопоставлении с возможными оценками ущерба.

Основная цель прогнозирования рисков – установление степени вероятного нарушения требований по защите информации и/или нарушения надежности реализации исследуемых системных процессов (процессов соглашения, процессов организационного обеспечения проекта, процессов технического управления, технических процессов) с учетом требований по защите информации на заданный период прогноза. Прогнозирование рисков осуществляется в интересах решения определенных задач системного анализа при планировании и реализации системных процессов, обосновании эффективных предупреждающих мер по снижению рисков или их удержанию в допустимых пределах.

Применительно к конкретной системе для прогнозирования рисков нарушения требований по защите информации в процессе управления рисками для системы определению подлежат:

– состав заинтересованных сторон, имеющих интерес к этой системе;

– состав входных результатов и выполняемых действий процесса управления рисками для системы и используемые для этого активы;

– перечень потенциальных угроз и возможных сценариев возникновения и развития угроз для выходных результатов и выполняемых действий процесса управления рисков для системы;

– иные объекты, используемые в прогнозировании рисков при необходимости того, насколько организация процесса управления рисками способна обеспечить возможности по его выполнению в заданной среде применения системы.

Для выполнения необходимых работ системной инженерии, связанных с прогнозированием рисков, используют методы и модели, устанавливают ограничения на допустимые риски, разрабатывают необходимые методики системного анализа. Например, процесс управления риском заключаются в следующем:

– качества системы – в ненарушении надежности реализации процесса выполнения необходимых условий с завершением всех предпринимаемых действий процесса управления качеством системы;

– планирования проекта – в выполнении необходимых условий с завершением всех предпринимаемых действий процесса планирования проекта, соблюдением сроков выполнения необходимых действий процесса;

– управления информационной системы – в ненарушении надежности реализации процесса обеспечения необходимой надежности представления используемой информации, обеспечении необходимой своевременности представления используемой информации, обеспечении полноты оперативного отражения в системе новых объектов и явлений;

– определения потребностей и требований заинтересованной стороны – в ненарушении надежности реализации процесса выполнения необходимых условий анализа бизнеса или назначения системы;

– определения системных требований – в защите от реализации угроз безопасности информации, направленных на нарушения функционирования системы при защите системы от вредоносного программного обеспечения, использовании системы обнаружения вторжения и пресечении попыток проникновения в операционную среду.

Полный список методик системного анализа приведен в таблице [3].

Порядок прогнозирования рисков.

1. Определяют моделируемые системы и устанавливают анализируемые объекты.

2. Устанавливают конкретные цели прогнозирования.

3. Формируют перечень возможных угроз.

4. Выбирают расчетные показатели и подходящие математические модели и методы.

5. Разрабатывают входные данные, необходимые для проведения расчетов.

Результаты прогнозирования рисков должны быть удобны для обработки заказчиком функционирования системы и/или аналитиков моделируемой системы [6]. Результаты представляются в виде гистограмм, графиков, таблиц или в ином виде, позволяющем анализировать зависимости рисков от изменения значений исходных данных при решении задач системного анализа. Результаты расчетов подлежат использованию для управления рисками при выполнении работ системной инженерии [5–7].

После демонстрации результатов прогнозирования исполнителю необходимо представить возможные способы уменьшения рисков, кото-

Перечень методик системного анализа

Системный процесс	Стандарт
Процессы приобретения и поставки продукции и услуг для системы	ГОСТ Р 59329–2021. Прил. Д
Процесс управления моделью жизненного цикла системы	ГОСТ Р 59330–2021. Прил. Д
Процесс управления инфраструктурой системы	ГОСТ Р 59331–2021. Прил. Е
Процесс управления портфелем проектов	ГОСТ Р 59332–2021. Прил. Д
Процесс управления человеческими ресурсами системы	ГОСТ Р 59333–2021. Прил. Е
Процесс управления качеством системы	ГОСТ Р 59334–2021. Прил. Д
Процесс управления знаниями о системе	ГОСТ Р 59335–2021. Прил. Е
Процесс планирования проекта	ГОСТ Р 59336–2021. Прил. Д
Процесс оценки и контроля проекта	ГОСТ Р 59337–2021. Прил. Д
Процесс управления решениями	ГОСТ Р 59338 – 2021. Прил. Е
Процесс управления рисками для системы	ГОСТ Р 59339–2021. Прил. В (данная таблица)
Процесс управления конфигурацией системы	ГОСТ Р 59340–2021. Прил. Д
Процесс управления информацией системы	ГОСТ Р 59341–2021. Прил. Е
Процесс измерений системы	ГОСТ Р 59342–2021. Прил. Д
Процесс гарантии качества для системы	ГОСТ Р 59343–2021. Прил. Е
Процесс анализа бизнеса или назначения системы	ГОСТ Р 59344–2021. Прил. Д
Процесс определения потребностей и требований заинтересованной стороны для системы	ГОСТ Р 59345–2021. Прил. Е
Процесс определения системных требований	ГОСТ Р 59346–2021. Прил. Ж
Процесс определения архитектуры системы	ГОСТ Р 59347–2021. Прил. Е
Процесс определения проекта	ГОСТ Р 59348–2021. Прил. Д
Процесс системного анализа	ГОСТ Р 59349–2021. Прил. Е
Процесс реализации системы	ГОСТ Р 59350–2021. Прил. Д
Процесс комплексирования системы	ГОСТ Р 59351–2021. Прил. Д
Процесс верификации системы	ГОСТ Р 59352–2021. Прил. Д
Процесс передачи системы	ГОСТ Р 59353–2021. Прил. Д
Процесс аттестации системы	ГОСТ Р 59354–2021. Прил. Д
Процесс функционирования системы	ГОСТ Р 59355–2021. Прил. Е
Процесс сопровождения системы	ГОСТ Р 59356–2021. Прил. Е
Процесс изъятия и списания системы	ГОСТ Р 59357–2021. Прил. Д

рые могут быть количественно обоснованы с применением рекомендуемых методов и моделей, представляют собой механизмы непосредственного управления рисками при реализации каждого из системных процессов.

Библиографический список

1. *Петренко С. А., Симонов С. В.* Управление информационными рисками. Экономически оправданная безопасность. М.: АйТи; ДМК Пресс, 2004. 384 с.

2. *Плетнев П. В., Белов В. М.* Методика оценки рисков информационной безопасности на предприятиях малого и среднего бизнеса // Доклады Томского государственного университета систем управления и радиоэлектроники. 2012. № 1–2 (25). С. 83–86.

3. ГОСТ Р 59339–2021. Системная инженерия. Защита информации в процессе управления рисками для системы. URL: https://docs.cntd.ru/document/1200179457 (дата обращения: 24.11.2021).

4. *Kolomoitcev V., Kolomoytseva K.* Monitoring the effectiveness of the use of means of protection in information protection systems // 2021 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF) (2021). St. Petersburg, 2021. P. 1–5.

5. О подходе к оценке информационной безопасности предприятия / А. Р. Айдинян, О. Л. Цветкова, И. Р. Кикоть, А. В. Казанцев, В. В. Каплун // Системный анализ, управление и обработка информации: сб. тр. V Междунар. науч. семинара, п. Дивноморское, 2–6 окт. Ростов н/Д: ДГТУ, 2014. С. 109–111.

6. *Шарапов А. В.* Проблема определения понятия информационных рисков // Безопасность информационных технологий. 2010. № 2. С. 44–48.

7. *Баранова Е. К., Зубровский Г. Б.* Управление инцидентами информационной безопасности. Проблемы информационной безопасности // Проблемы информационной безопасности: тр. I Междунар. науч.-практ. конф. / Крым. федер. ун-т им. В. И. Вернадского. Симферополь, 2015. С. 27–33.

УДК 004.056.2 **DOI: 10.31799/978-5-8088-1701-2-2022-2-236-241**

Д. А. Зыков*
студент
В. В. Комашинский*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИССЛЕДОВАНИЕ СТАТИСТИЧЕСКИХ МЕТОДОВ ДЛЯ ПОВЫШЕНИЯ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Проанализирована эффективность применения статистических методов для повышения защищенности информационных систем. Рассмотрены метод бутстрэпа, методы множественного сравнения и методы множественного ранжирования, метод кластеризации kmeans и метод классификации k-ближайших соседей, метод случайного леса.
Ключевые слова: *средства зашиты информационных систем, статистические методы, защищенность информационных систем.*

D. A. Zykov*
Student
V. V. Komashinsky*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

RESEARCH OF STATISTICAL METHODS FOR INCREASING THE SECURITY OF INFORMATION SYSTEMS

The effectiveness of the use of statistical methods to improve the security of information systems has been analyzed. The following methods were considered: bootstrap method, multiple comparison and multiple ranking methods, k-means clustering method and k-nearest neighbors classification method, random forest method.
Keywords: *information security systems, statistical methods, security of information systems.*

В настоящее время одно из перспективных направлений в информационной безопасности – повышение защищенности информационных систем при помощи применения интеллектуальных информационных технологий.

В данной работе рассмотрены следующие статистические методы и проанализирована эффективность их применения для повышения защищенности информационных систем:

1) метод бутстрэпа, позволяющий оценивать многие статистики сложных распределений;

2) методы множественного ранжирования и методы множественного сравнения;

3) метод кластеризации k-means и метод классификации k-ближайших соседей (k-NN);

4) метод случайного леса.

Метод Бутстрэпа

Бутстрэп (англ. bootstrap) в статистике – практический компьютерный метод исследования распределения статистик вероятностных распределений, основанный на многократной генерации выборок методом Монте-Карло на базе имеющейся выборки. Позволяет просто и быстро оценивать самые разные статистики (доверительные интервалы, дисперсию, корреляцию и т. д.) для сложных моделей.

В основе идеи бутстрепа лежит оценка структурных характеристик генеральной совокупности на основе перевыборки (resampling) из выборки. Иными словами, перевыборка по отношению к выборке рассматривается как выборка по отношению к генеральной совокупности.

Алгоритм работы метода.

1. Из генеральной совокупности формируется случайная выборка из N(t) наблюдений (например, если требуется определить среднюю сумму чека посетителя супермаркета, будем оценивать ее на основе выборки из 1000 клиентов).

2. К выборке применяется случайная перевыборка с возвратом (псевдовыборка) того же объема, но в которую некоторые наблюдения могут попасть несколько раз, а другие не по-

пасть совсем. Например, если выборка содержала 5 значений (1, 2, 3, 4, 5), то результатом перевыборки может быть (2, 2, 4, 5, 5). Затем вычисляется ее среднее.

3. Процедура перевыборки повторяется достаточно много раз (несколько десятков, сотен или даже тысяч), и для каждого случая вычисляется среднее.

4. Из полученного набора средних значений вычисляется среднее и рассматривается как среднее всей генеральной совокупности.

Важнейшие преимущества бутстрепа:

– простота реализации;

– отсутствие необходимости гипотез о параметрах распределения данных;

– возможность оценивания многих статистических характеристик (среднего, дисперсии, стандартного отклонения, доверительных интервалов, квантилей, коэффициентов корреляции и др.).

К недостаткам метода можно отнести использование малореалистичного предположения о независимости перевыборок и значительные вычислительные затраты при их многократном построении.

Метод оказывается особенно полезным, когда теоретическое распределение данных неизвестно или объем выборки мал для прямой статистической оценки. Может быть использован в построении онтологии информационной безопасности [1].

Большое преимущество бутстрэпа в приложениях ИБ – его простота. Это простой способ получения оценок стандартных ошибок и доверительных интервалов для сложных оценок распределения, таких как процентильные точки, пропорции, отношение шансов и коэффициенты корреляции. Бутстрэп также является подходящим способом контроля и проверки стабильности результатов. Хотя для большинства задач невозможно узнать истинный доверительный интервал, бутстрэп асимптотически более точен, чем стандартные интервалы, полученные с использованием выборочной дисперсии и предположений о нормальности. Бутстрэппинг также оказывается удобным методом, который позволяет избежать затрат на повторение эксперимента для получения других групп выборочных данных.

Хотя бутстрэппинг (при некоторых условиях) асимптотически согласован, он не дает общих гарантий на выходной выборке. Результат может зависеть от репрезентативной выборки. Кажущаяся простота может скрывать тот факт, что при проведении бутстрэп-анализа делаются важные предположения (например, независимость выборок), тогда как они были бы более формально сформулированы в других подходах. Кроме того, бутстрэппинг может занять много времени.

Метод бутстрэппинга может быть применен в анализе статистики атак, уязвимостей ИБ, построении онтологии ИБ в информационной системе. Его наиболее целесообразно использовать при наличии малых объемов данных (выборок) и при неизвестном теоретическом распределении данных.

Методы множественного ранжирования и методы множественного сравнения

Множественные сравнения возникают, когда необходимо на одной и той же выборке параллельно проверить ряд статистических гипотез. В частности, критерий Стьюдента может быть использован для проверки гипотезы о различии средних только для двух групп. Если план исследования большего числа групп, совершенно недопустимо просто сравнивать их попарно. Для корректного решения этой задачи можно воспользоваться, например, дисперсионным анализом.

Однако дисперсионный анализ позволяет проверить лишь гипотезу о равенстве всех сравниваемых средних. Но, если гипотеза не подтверждается, нельзя узнать, какая именно группа отличалась от других. Это позволяют сделать методы множественного сравнения, которые в свою очередь также бывают параметрические и непараметрические.

Эти методы дают возможность провести множественные сравнения так, чтобы вероятность хотя бы одного неверного заключения оставалась на первоначальном выбранном уровне значимости, например 5%.

Параметрические критерии:

– критерий Стьюдента для множественных сравнений;

– критерий Ньюмана – Кейлса;

– критерий Тьюки;

– критерий Шеффе;

– критерий Даннета.

Непараметрические критерии:

– критерий Краскела – Уоллиса;

– медианный критерий.

Основные параметрические критерии для множественного сравнения независимых групп могут после некоторых модификаций применяться для установления различий и в повторных измерениях, если дисперсионный анализ выявил наличие таких различий.

На основе методов множественного ранжирования и методов множественного сравнения

могут быть построены модели предсказания и распознавания инсайдерских угроз информационной безопасности.

В работе [2] рассмотрено использование различных алгоритмов множественного сравнения для предсказания инсайдерских угроз.

Оценка алгоритма предсказания и нахождения угроз основана на двух условиях:

1) оценка теоретической надежности алгоритма;

2) оценка предвзятости модели.

Для ранжирования алгоритмов в соответствии с двумя упомянутыми условиями может быть использован двусторонний дисперсионный анализ по рангам, чтобы проверить нулевую гипотезу об отсутствии статистически значимых различий в производительности алгоритмов. После применения двухфакторного дисперсионного анализа по критерию рангов используются односторонние «сравнения групп или условий с контролем» для выполнения попарных тестов на основе метода множественного сравнения, чтобы проверить нулевую гипотезу об отсутствии существенной разницы между двумя алгоритмами.

Согласно предложенным оценкам алгоритмов и статистическим тестам, на основе методов множественного ранжирования и множественных сравнения были построены следующие результирующие оценки [2].

1. Оценка теоретической надежности алгоритмов (рис. 1).

2. Оценка простоты алгоритма (рис. 2).

Методы множественного ранжирования и методы множественного сравнения могут быть использованы в сфере ИБ для нахождения наиболее эффективных алгоритмов, позволяющих выполнить поставленную задачу, что позволяет повысить защищенность информационной системы путем выбора наиболее целесообразных и эффективных средств защиты информации.

Метод кластеризации k-means и метод классификации ближайших соседей (k-NN)

Метод k-средних используется для кластеризации данных на основе алгоритма разбиения векторного пространства на заранее определенное число кластеров k.

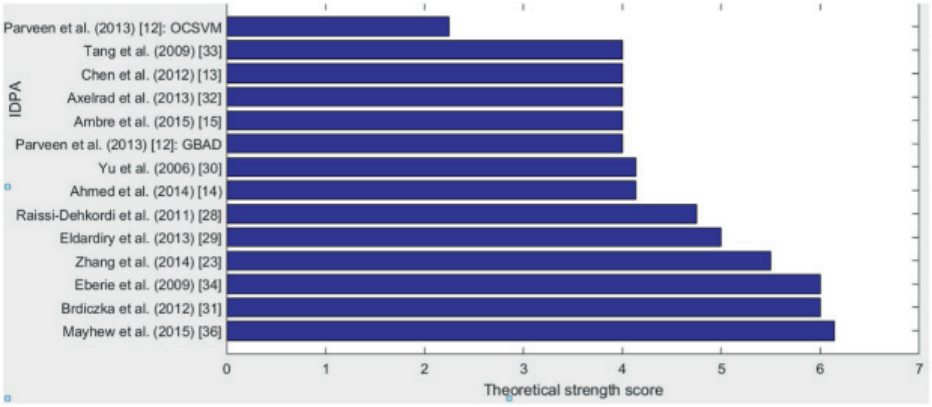


Рис. 1. Теоретическая надежность алгоритмов

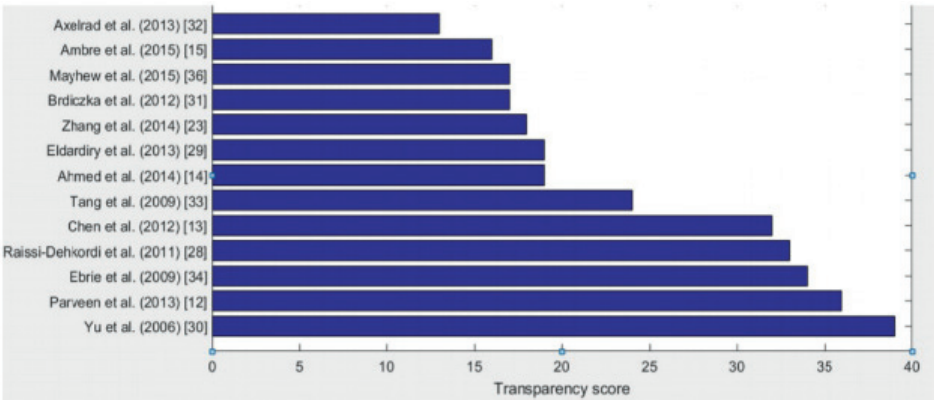


Рис. 2. Оценка простоты реализации алгоритмов

Преимущества алгоритма – скорость и простота реализации. К недостаткам можно отнести неопределенность выбора начальных центров кластеров, а также то, что число кластеров должно быть задано изначально, что может потребовать некоторой априорной информации об исходных данных.

Существуют методы кластеризации, которые можно рассматривать как происходящие от k-средних. Например, в методе k-медиан (k-medoids) для вычисления центроидов используется не среднее, а медиана, что делает алгоритм более устойчивым к аномальным значениям в данных.

Метод k-ближайших соседей используется для решения задачи классификации. Он относит объекты к классу, которому принадлежит большинство из k его ближайших соседей в многомерном пространстве признаков. Это один из простейших алгоритмов обучения классификационных моделей.

В процессе обучения алгоритм запоминает все векторы признаков и соответствующие им метки классов. При работе с реальными данными, т. е. наблюдениями, метки класса которых неизвестны, вычисляется расстояние между вектором нового наблюдения и ранее запомненными. Затем выбирается k ближайших к нему векторов, и новый объект относится к классу, которому принадлежит большинство из них.

Несмотря на относительную алгоритмическую простоту, метод показывает хорошие результаты. Главный его недостаток – высокая вычислительная трудоемкость, которая увеличивается квадратично с ростом числа обучающих примеров.

В работе [3] рассмотрен алгоритм установления сетевой безопасности на основе алгоритма кластеризации k-средних в исследованиях интеллектуального анализа данных. Алгоритм интеллектуального анализа данных состоит из алгоритма кластерного анализа, корреляционного анализа и алгоритма классификации.

Кластерный анализ – распространенный метод интеллектуального анализа данных, который может быть использован для выявления неконтролируемых аномалий и решить проблемы, существующие в традиционных методах. Кластерный анализ может стать основой для создания системы обнаружения вторжений.

Система обнаружения вторжений в основном предназначена для того, чтобы различать нормальное и аномальное поведение, а затем принимать соответствующие меры. Наиболее сложной задачей оказывается определение разницы между аномальными и нормальными данными. По алгоритму кластеризации группа позволяет обобщить данные и найти точки соприкосновения, а затем выделить различие. Таким образом, применение неконтролируемого алгоритма кластеризации в области обнаружения аномалий дает возможность повысить эффективность обнаружения вторжений.

В большинстве приложений нет возможности использовать категорированные данные, и, следовательно, нельзя точно определить нормальное или аномальное состояние. В таких случаях, как правило, используется пороговое значение, позволяющее категорировать данные.

Исходя из результатов, представленных в статье [3], можно заключить что методы k-средних и k-ближайших соседей могут быть эффективно использованы в интеллектуальных системах обнаружения вторжений и повысить защищенность информационных систем.

Методы k-средних и k-ближайших соседей целесообразны для применения в практических средствах защиты, так как отличаются простотой реализации и высокой вычислительной эффективностью.

Метод случайного леса

Случайный лес (англ. Random forest) – алгоритм машинного обучения, предложенный Лео Брейманом и Адель Катлер, заключающийся в использовании комитета (ансамбля) решающих деревьев. Алгоритм сочетает две основные идеи: метод бэггинга Бреймана и метод случайных подпространств, предложенный Тин Кам Хо. Алгоритм применяется для задач классификации, регрессии и кластеризации. Основная идея состоит в использовании большого ансамбля решающих деревьев, каждое из которых само по себе дает очень невысокое качество классификации, но за счет их большого количества результат получается хорошим.

Достоинства метода.

– Способность эффективно обрабатывать данные с большим числом признаков и классов.

– Нечувствительность к масштабированию (и вообще к любым монотонным преобразованиям) значений признаков.

– Одинаково хорошо обрабатываются как непрерывные, так и дискретные признаки. Существуют методы построения деревьев по данным с пропущенными значениями признаков.

– Существуют методы оценивания значимости отдельных признаков в модели.

– Внутренняя оценка способности модели к обобщению (тест по неотобранным образцам out-of-bag).

– Высокая параллелизуемость и масштабируемость.

Недостаток метода – большой размер получающихся моделей. Требуется $O(K)$ памяти для хранения модели, где K – число деревьев.

Внедрение машинного обучения для поддержки информационной безопасности – неизбежная тенденция из-за постоянного увеличения сетевого трафика и изощренности атак. Алгоритмы машинного обучения с успехом используются во все большем количестве областей, включая обработку изображений, распознавание речи и текста, маркетинг в социальных сетях, и в последнее время в кибербезопасности. Современные системы обнаружения вторжений в сеть (NIDS) все больше обогащаются машинным обучением и алгоритмами глубокого обучения.

В работе [4] предложен новый подход к усилению защиты детекторов, основанный на машинном обучении. Исследование сфокусировано на алгоритме случайного леса из-за его доказанной эффективности для обнаружения вторжений; однако недавние исследования также подчеркивают его уязвимость перед некоторыми типами вредоносного ПО. Решение, предложенное в [4], основано на наблюдении, что существующие детекторы машинного обучения полагаются на чрезмерно жесткие критерии классификации: они обычно обучаются с помощью меток классов, которые разделяют выборки на несвязанные категории, где каждая выборка может быть вредоносной или доброкачественной. Подобный подход не может работать в киберпространстве, где каждый образец может иметь более расплывчатые атрибуты. По этой причине была использована идея введения некоторой степени гибкости в набор обучающих данных с помощью вероятностных меток.

Предложенный подход был проверен с помощью большого количества экспериментов, выполненных на наборе общедоступных и классифицированных трассировок трафика, содержащих более 20 млн сетевых потоков с доброкачественными и вредоносными образцами различных семейств вредоносных программ [4]. Эти наборы данных отражают сетевое поведение средних и крупных предприятий и представляют подходящие настройки для реалистичной оценки.

Реалистичный сценарий, в котором предложенный детектор может быть успешно применен, представлен на рис. 3, где показана сеть большого предприятия с множеством внутренних хостов и пограничным маршрутизатором, подключенным к экспортеру сетевого потока. Сгенерированные потоки проверяются системой обнаружения сетевых вторжений, основанной на машинном обучении, которая направлена на выявление вредоносных действий путем использования алгоритма случайного леса. Мы предполагаем, что злоумышленник уже закрепился во внутренней сети, скомпрометировав одну или несколько машин и развернув вредоносное ПО, которое взаимодействует с инфраструктурой управления и контроля.

Обширная кампания экспериментальных оценок демонстрирует эффективность предложенного метода, который дает двукратное преимущество перед современным: в сценариях, подверженных злонамеренным манипуляциям с входными данными, он повышает уровень обнаружения до 250%; в сценариях, которые не подвержены атакам противника, он достигает такой же или более высокой точности, чем существующие методы.

Результаты исследований, предложенные в статье [4], показывают, что на данный момент метод случайного леса имеет огромную практическую важность в интеллектуальных системах защиты информации, так как позволяет обрабатывать большие массивы признаков с наивысшей вычислительной эффективностью. Также существуют способы улучшения данного алгоритма, способные увеличить его эффективность в сфере информационной безопасности.

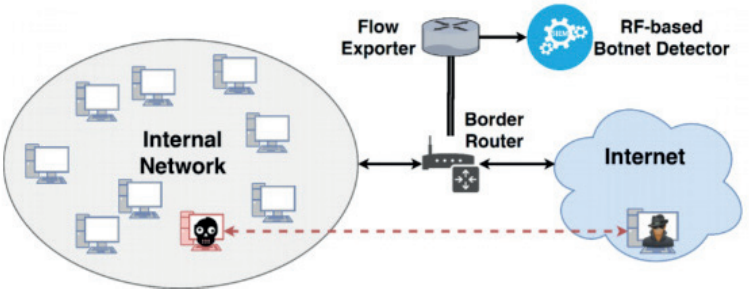


Рис. 3. Схема сети предприятия

На данный момент метод случайного леса – один из самых широко используемых алгоритмов машинного обучения при построении средств защиты информационных систем. Средства защиты, использующие его, помогают значительно повысить защищенность информационных систем.

Были рассмотрены следующие статистические методы:

1) метод бутстрэпа;

2) методы множественного ранжирования и методы множественного сравнения;

3) метод кластеризации k-means и метод классификации k-ближайших соседей (k-NN);

4) метод случайного леса.

Наиболее перспективными, позволяющими значительно повысить защищенность информационных систем являются методы случайного леса и метод кластеризации k-means. В настоящее время они широко используются в построении систем обнаружения вторжений, отличаются высокой вычислительной эффективностью и адаптивностью, простотой реализации. Методы бутстрэпа и множественного сравнения могут быть применены лишь в частных случаях, таких как аналитика статистики угроз информационной безопасности.

Библиографический список

1. *Wali A., Chun S., Geller J.* A Bootstrapping Approach for Developing a Cyber-security Ontology Using Textbook Index Terms // Availability, Reliability and Security. 2018. P. 569–576.

2. *Gheyas I., Abdallah A.* Detection and prediction of insider threats to cyber security: a systematic literature review and meta-analysis // Big Data Analytics. 2019. № 1.

3. *Bu C.* Network Security Based on K-Means Clustering Algorithm in Data Mining Research // Conference: 8th International Conference on Social Network, Communication and Education. 2018.

4. Hardening Random Forest Cyber Detectors Against Adversarial Attacks / G. Apruzzese, M. Andreolini, M. Colajanni, M. Marchetti // IEEE Transactions on emerging topics in computational intelligence. 2019.

УДК 621.391 **DOI: 10.31799/978-5-8088-1701-2-2022-2-242-245**

М. Н. Исаева*
ассистент
А. А. Овчинников*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОЦЕНКА КОРРЕКТИРУЮЩЕЙ СПОСОБНОСТИ НИЗКОПЛОТНОСТНЫХ КОДОВ ДЛЯ МНОГОКРАТНЫХ ПАКЕТОВ ОШИБОК

Рассматривается возможность исправления однократных и многократных пакетов ошибок с использованием низкоплотностных кодов. Были проведены эксперименты для различных параметров блочно-перестановочной конструкции низкоплотностного кода. Представлены результаты, демонстрирующие возможность исправления однократных и многократных пакетов ошибок рассматриваемыми кодами.
Ключевые слова: *низкоплотностные коды, корректирующая способность кода, пакеты ошибок.*

Работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации, соглашение № FSRF-2020-0004 «Научные основы построения архитектур и систем связи бортовых информационно-вычислительных комплексов нового поколения для авиационных, космических систем и беспилотных транспортных средств».

M. N. Isaeva*
Assistant
A. A. Ovchinnikov*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

EVALUATION OF THE CORRECTING ABILITY OF LOW-DENSITY CODES FOR MULTIPLE ERROR BURSTS

The possibility of correcting single and multiple error packets using low-density codes discussed in this paper. Experiments were carried out for various parameters of the block-permutation constructions of the low-density code. The results demonstrating the possibility of correcting single and multiple error packets by the considered codes are presented.
Keywords: *Low–density codes (LDPC-codes), maximum burst error correction, multiple error bursts.*

Введение

В настоящее время по каналам связи циркулируют огромные потоки информации, которые могут быть подвержены ошибкам, возникающим при передаче по сетям. Характеристикам каналов свойственно изменяться, причем в течение одного сеанса связи, при этом они могут меняться быстро и медленно. При медленном изменении ошибки локализованы на определенных позициях передаваемых символов, при быстром – распределены более равномерно по позициям. Более адекватной, по сравнению с предположением о независимости ошибок, как, например, в двоичном симметричном канале (ДСК), является модель с чередованием «хорошего» состояния канала, в котором ошибки возникают редко, и «плохого», в котором они случаются часто.

В реальных системах используются более простые модели, предполагающие независимость ошибок, однако из теории информации известно, что учет особенностей группирования ошибок потенциально приводит к увеличению возможной скорости передачи.

Типичное описание шума в канале с памятью – так называемый пакет ошибок. В традиционной теории помехоустойчивого кодирования коды, исправляющие пакеты ошибок, исследованы мало. Чаще всего при построении таких кодов делается предположение о том, что на позициях кодового слова пакет ошибок может быть только один, что неадекватно передаче по каналу с быстро меняющимися характеристиками. В данной статье рассматривается возможность исправления как однократных, так и многократных пакетов ошибок с исполь-

зованием низкоплотностных кодов, наиболее широко представленных в стандартах современной связи.

Описание модели ошибок

Рассмотрим аддитивную модель, в которой по каналу связи передается некоторый (n, k)-код (двоичный). Будем считать, что из канала связи получена двоичная последовательность y из n символов, влияние шума на передаваемые данные можно описать с помощью аддитивной модели: $y = x + e$, где e – вектор ошибки, x – входная последовательность канала.

Если ошибки независимы (рис. 1), то корректирующая способность кода оценивается как максимальное количество ошибок, исправляемая на кодовой длине, и характеризуется минимальным расстоянием d_0, где $d_0 = 2t + 1$, t – число исправляемых ошибок. Для случайного линейного кода нахождение минимального расстояния является NP-трудной задачей [1]. Если ошибки группированы, то могут быть разные подходы задания модели ошибок.

Например, одиночный пакет (рис. 2, *а*) ошибок может быть задан как подмножество позиций в принятом слове, в котором первая и последняя позиции являются ненулевыми элементами и эти элементы расположены друг от друга не далее, чем на b позиций, внутри пакета количество ненулевых элементов не учитывается.

Двукратный пакет длиной b (рис. 2, *б*) – это такой пакет, в котором все ненулевые элементы

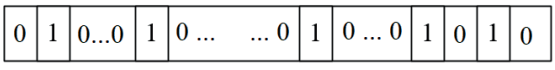


Рис. 1. Вектор с независимыми ошибками

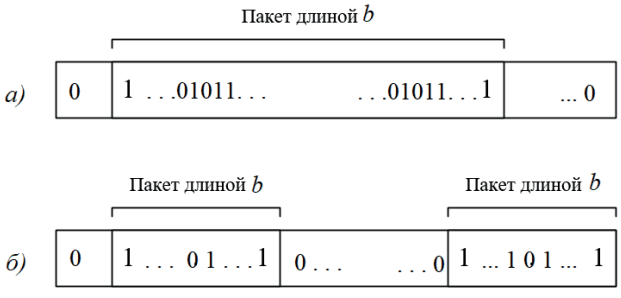


Рис. 2. Примеры задания модели ошибок: а) одиночный пакет ошибок длиной b; б) двойной пакет ошибок длиной b

вектора ошибок сгруппированы на двух подмножествах позиций, каждый из которых содержит не более b позиций.

Низкоплотностные коды

Низкоплотностные коды или коды с малой плотностью проверок на четность (low-density parity-check – LDPC) предложены Р. Галлагером в 1962 г. [2–3]. Задаются с помощью проверочной матрицы **H**, которая является разреженной. В современных стандартах связи 802.11 (Wi-Fi) [4], 5G [5] для задания кодов используют блочно-перестановочные конструкции (БПК) [6]:

$$\mathbf{H} = \begin{bmatrix} \mathbf{C}^{t_{11}} & \mathbf{C}^{t_{12}} & \dots & \mathbf{C}^{t_{1\rho}} \\ \mathbf{C}^{t_{21}} & \mathbf{C}^{t_{22}} & \dots & \mathbf{C}^{t_{2\rho}} \\ \dots & \dots & \dots & \dots \\ \mathbf{C}^{t_{\gamma 1}} & \mathbf{C}^{t_{\gamma 2}} & \dots & \mathbf{C}^{t_{\gamma\rho}} \end{bmatrix},$$

где $\mathbf{C}^{t_{ij}}$ – блоки, представляющие собой матрицы циклической перестановки в степени t_{ij} размером $m \times m$:

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & 0 & \dots & 1 & 0 \end{bmatrix},$$

а базовую матрицу в общем виде можно представить так:

$$\mathbf{H_{base}} = \begin{bmatrix} t_{11} & \dots & t_{1\rho} \\ \vdots & \ddots & \vdots \\ t_{\gamma 1} & \dots & t_{\gamma\rho} \end{bmatrix}.$$

Таким образом, если размер базовой матрицы ($\gamma \times \rho$), то размер, соответствующий матрице **H**: $r = \gamma \times m$, а $n = \rho \times m$, где $k = n - r$, m – размер блока.

Процедура определения длины пакета

В [7] рассмотрена корректирующая способность кодов в каналах с памятью. Если два различных вектора дают одинаковый синдром, то они лежат в одном смежном классе. В этом случае столбцы матрицы **H***, полученной, как представлено на рис. 3, линейно зависимы, соответственно, не могут быть исправлены. С учетом

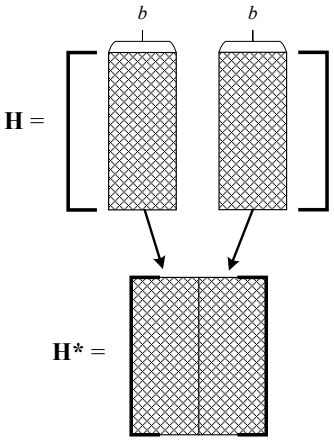


Рис. 3. Получение матрицы H для одиночного пакета ошибок при определении максимальной длины исправляемого пакета*

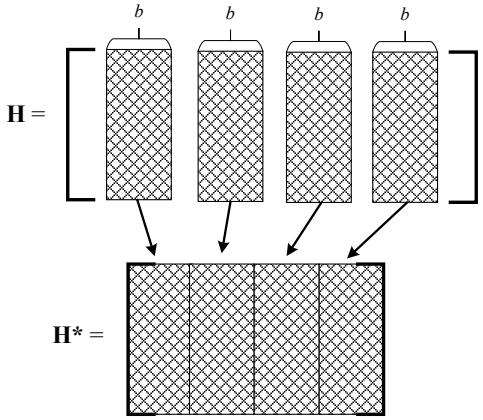


Рис. 4. Получение матрицы H для двойного пакета ошибок при определении максимальной длины исправляемого пакета (для многократного – аналогично)*

этого в блочно-перестановочной матрице с размером блока m пакеты длиной m не могут быть исправлены и для определения длины исправляемого пакета необходимо рассматривать различные значения $b = m - 1, m - 2, \ldots, 1$, до тех пор, пока **H*** не будет иметь полный ранг. Сложность определения длины исправляемого пакета является полиномиальной.

Для того чтобы определить длину многократного пакета, воспользуемся аналогичным принципом (рис. 4). Соответственно будут рассматриваться многократные пакеты ошибок (по 2, 3 пакета и т. д.) одинаковой длины $b = m - 1, m - 2, \ldots, 1$. Необходимо также учитывать расположение каждого пакета относительно друг друга и размер получившейся матрицы **H***.

Результаты экспериментов

Для определения длины исправляемых пакетов с помощью блочно-перестановочных конструкций были проведены вычислительные эксперименты. Были рассмотрены коды с различными параметрами (табл. 1). В качестве результатов приведена максимальная длина одиночного, двойного и тройного пакетов b_1, b_2, b_3, а также частота получения конкретного значения длины пакета в ходе экспериментов F_1, F_2, F_3.

Для рассмотренных блочно-перестановочных конструкций размер одиночного пакета сопоставим с размером блока. При этом с высокой вероятностью (~0,5) можно построить случайные блочно-перестановочные коды с длиной двукратных исправляемых пакетов в пределах 0,4–0,5m от размера блока. Для троекратных исправляемых пакетов в пределах 0,15–0,25m с такой же вероятностью, как для двукратных.

Таблица 1

Параметры кодов и результаты экспериментов

$m = 15\ \gamma = 3\ \rho = 6$						$m = 15\ \gamma = 4\ \rho = 8$					
b_1	F_1	b_2	F_2	b_3	F_3	b_1	F_1	b_2	F_2	b_3	F_3
12	0,7	0	0,1	0	0,2	12	0,2	6	0,2	3	0,3
14	0,3	4	0,2	1	0,1	14	0,8	7	0,8	4	0,7
		5	0,5	2	0,7						
		6	0,5								
$m = 16\ \gamma = 3\ \rho = 6$						$m = 16\ \gamma = 4\ \rho = 8$					
b_1	F_1	b_2	F_2	b_3	F_3	b_1	F_1	b_2	F_2	b_3	F_3
12	0,3	2	0,1	0	0,3	12	0,6	4	0,7	1	0,1
14	0,7	3	0,1	1	0,2	14	0,4	6	0,3	2	0,3
		4	0,1	2	0,1					3	0,1
		5	0,2	3	0,4					4	0,5
		6	0,5								
$m = 19\ \gamma = 3\ \rho = 6$						$m = 19\ \gamma = 4\ \rho = 8$					
b_1	F_1	b_2	F_2	b_3	F_3	b_1	F_1	b_2	F_2	b_3	F_3
18	1,0	0	0,1	0	0,1	18	1,0	9	1,0	6	1,0
		5	0,2	2	0,2						
		6	0,1	3	0,5						
		7	0,2	4	0,2						
		8	0,4								

Заключение

В статье была рассмотрена возможность исправления как однократных пакетов, так и многократных для низкоплотностных кодов с различными параметрами. Результаты показали, что чем больше исправляемых пакетов мы рассматриваем, тем меньше максимальная длина

одного исправляемого пакета, однако встречаются коды, в которых нет возможности исправления многократных пакетов.

Библиографический список

1. *Vardy A.* The intractability of computing the minimum distance of a code // IEEE Transactions on Information Theory. 1997. Vol. 43, № 6. P. 1757–1766.

2. *Gallager R. G.* Low-Density Parity-Check Codes // IRE Transactions on Information Theory. 1962. Vol. 8, № 1. P. 21–28.

3. *Gallager R. G.* Low Density Parity Check Codes. Cambridge, MA: MIT Press, 1963. 90 p.

4. IEEE 802.11n/d1.0 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. Mar. 2006. URL: https://www.techstreet.com/standards/ieee-802-11n-2009?product_id=1643007 (дата обращения: 01.12.2021).

5. Chairman's Notes of Agenda Item 7.1.5 Channel Coding and Modulation, document R1-1613710, 3GPP, Nov. 2016. URL: https://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_87/Docs/R1-1613710.zip (дата обращения: 01.12.2021)

6. *Козлов А. В., Крук Е. А., Овчинников А. А.* Подход к построению блочно-перестановочных кодов с малой плотностью проверок на четность // Изв. вузов. Приборостроение. 2013. Т. 56, №8. С. 9–14.

7. *Вересова А. М., Овчинников А. А.* Об одном алгоритме декодирования пакетов для блочно-перестановочных низкоплотностных кодов // Волновая электроника и инфокоммуникационные системы: XXII Междунар. науч. конф.: сб. ст. 2019. Ч. 1. С. 154–159.

УДК 004.75 **DOI: 10.31799/978-5-8088-1701-2-2022-2-246-249**

В. С. Канаров*
студент
Д. Д. Галкин*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ДЕЦЕНТРАЛИЗОВАННАЯ СИСТЕМА УМНОГО ДОМА НА ОСНОВЕ ЯЧЕИСТОЙ ТОПОЛОГИИ СЕТИ

Предлагается архитектура распределенного хранения сценариев автоматизаций устройств умного дома, основанного на ячеистой топологии сети.
Ключевые слова: *сеть, интернет вещей, ячеистая топология сети, распределенная система, умный дом, беспроводная сенсорная сеть.*

V. S. Kanarov*
Student
D. D. Galkin
Student
**St. Petersburg State University of Aerospace Instrumentation*

DECENTRALIZED SMART HOME SYSTEM BASED ON MESH NETWORK TOPOLOGY

The article proposes an architecture for distributed storage of information about the interactions of smart home devices based on a mesh network topology.
Keywords: *network, internet of things, mesh network topology, distributed system, smart home, wireless sensor network.*

В современном мире можно найти множество решений построения умного дома (системы автоматизированных домашних устройств) [1] на базе Wi-Fi, такие как Яндекс-устройства, Xiaomi, SonOff [2]. Они просты и легко применяются в квартирах или небольших домах, где радиус работы роутера позволяет делать это без особых проблем. Но бывают случаи, когда роутер не может покрыть всю необходимую зону работы, например частный дом. Для обеспечения работы системы в подобных условиях используют ретрансляторы. Система перестает быть простой, кроме того, возникают расходы на дополнительные устройства. Для решения этой проблемы некоторые компании начали использовать mesh-сети [3]. В таких сетях каждое устройство, кроме своей основной функции, дополнительно является ретранслятором, что позволяет расширять область работы системы посредством увеличения количества подключенных устройств. Например, работа Aqara основана на технологии ZigBee [4]. Но даже после того как устройства получили некоторую децентрализованность, они оставили за собой архитектуру централизованной системы (рис. 1). В них всегда есть главное устройство, которое принимает решение, если это необходимо.

На рис. 1 используются следующие обозначения: 0 – центр умного дома (далее используется обозначение «шлюз»), вся система настраивается через него, 1–8 – устройства. Черными стрелками обозначены подключения устройств

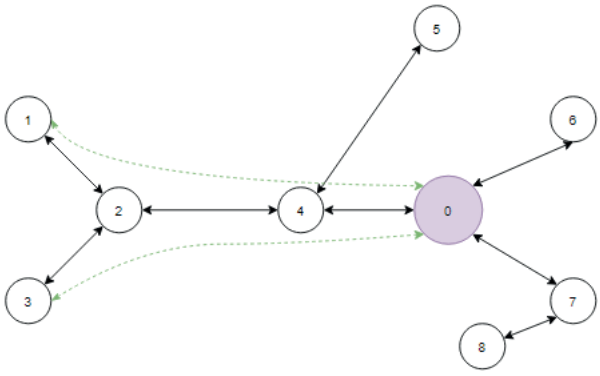


Рис. 1. Пример централизованной системы с ячеистой топологией

между собой по беспроводной сети, зелеными – потоки данных.

Предположим, что устройство 1 – контроллер умной лампы (далее лампа), а устройство 3 – умный выключатель. В классической схеме если у выключателя появилась новая информация, например на него нажали, он передает ее шлюзу, который в свою очередь просматривает в базе данных необходимую реакцию для этого действия и отправляет на лампу команду включения. Шлюз считался безотказной частью системы, и если устройство не смогло подключиться к нему, то и информация об изменении системы в целом устройству бы не поступила.

В децентрализованной системе устройства, к которым привязаны автоматизации, часто находятся в непосредственной близости друг от друга и с высокой долей вероятности они взаимодействуют напрямую. Тем не менее они вынуждены отправлять информацию до шлюза, а затем изменять свое состояние по его команде. Это работает в идеальной модели, но не всегда система может быть построена идеально, где связь с каждым из устройств надежна. Предположим, что на 4-е устройство постоянно поступает информация, в связи с чем оно нагружено и не может передавать информацию моментально (рис. 2).

Тогда простая автоматизация включения лампы по нажатию на клавишу может занять существенное время для пользователя или вовсе сделать решение нецелесообразным. Кроме того, сигнал от устройства может вовсе пропасть, если устройство вышло из строя, либо его отключили (рис. 3).

В таком случае пользователь полностью потеряет контроль в некоторой области. Для решения этой проблемы предлагается следующее: распределить информацию об автоматизациях между устройствами. Тогда для отдельной группы устройств будет не так важно состояние системы в целом, а клиент получит необходимую надежность (рис. 4).

Выделим описанные выше свойства в систему допущений.

1. Система основана на ячеистой топологии сети (mesh-сеть).

2. Шлюз считается безотказным устройством, но остальные устройства могут выйти из строя.

3. Не все устройства имеют возможность напрямую связаться с шлюзом.

4. Устройства локализованы в нескольких областях и имеют прямой канал связи между собой.

5. Устройства, участвующие в одном сценарии автоматизации, чаще всего принадлежат одной области.

Разберем конкретный пример.

На рис. 5 устройство 0 – шлюз, устройства 3 и 5 – первый и второй выключатель соответственно, устройство 1 – контроллер умной лампочки (далее лампочка), а устройство 6 – умный замок. Также имеются две автоматизации: из-

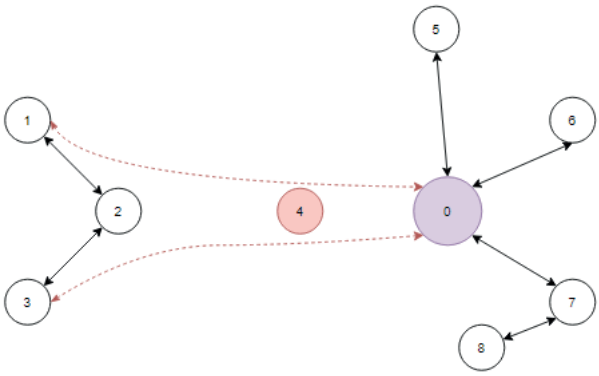


Рис. 3. Потеря контроля при отключенном 4-м устройстве

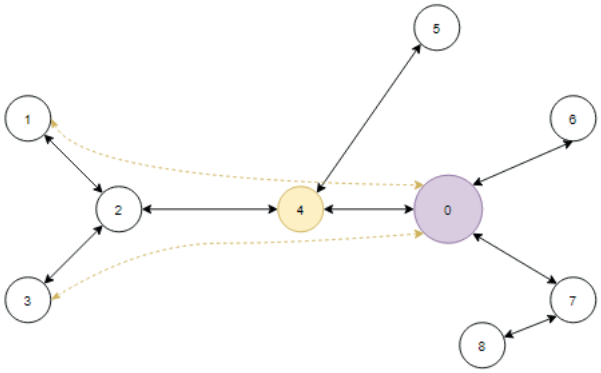


Рис. 2. Перегрузка 4-го устройства

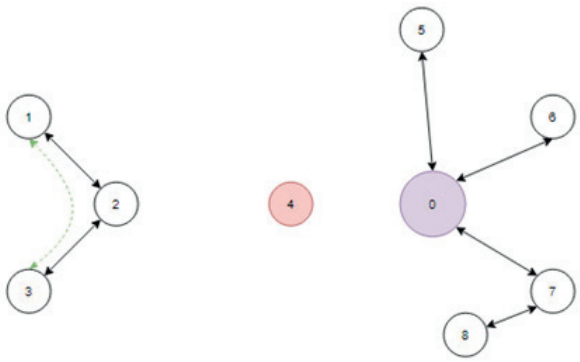


Рис. 4. Децентрализованная система при отключенном 4-м устройстве

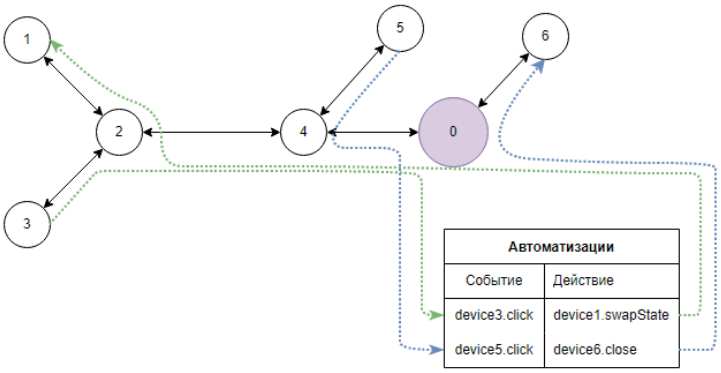


Рис. 5. Пример работы децентрализованной системы на основе ячеистой топологии с использованием классической архитектурой

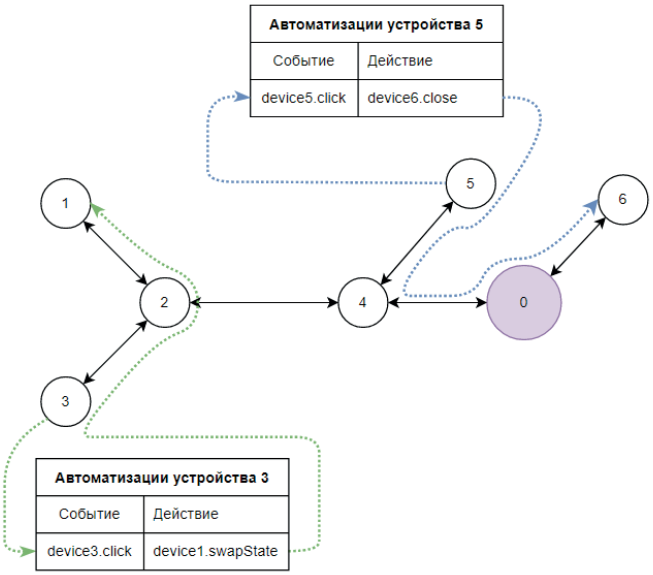


Рис. 6. Пример работы децентрализованной системы на основе ячеистой топологии с использованием предложенной архитектурой

менение состояние лампочки по нажатию на первый выключатель и закрытие двери при нажатии на второй выключатель.

Если пользователь нажмет на первый выключатель, то пакет информации с действием будет отправлен на шлюз, проходя через 2-е и 4-е устройство пакет дополнительно нагрузит сеть в целом. Затем шлюз проверит в базе данных, какие действия необходимо сделать при нажатии на выключатель, если действие было найдено, соответствующая команда будет отправлена на необходимое устройство, в данном случае устройство 1. Данная архитектура рассчитана на традиционное построение архитектуры сети, когда все устройства подключены к единому устройству-роутеру и все команды проходят через него (в дальнейшем такую архитектуру будем называть базовой). Можно заметить, что базовая архитектура не учитывает тот факт, что устройства 1–3 находятся в отдельной ветви mesh сети и при отключении устройства 4 могут самоорганизоваться в отдельную подсеть.

Цель предложенной архитектуры – использование возможности сети mesh к самоорганизации [5, 6] для обеспечения работы сценариев автоматизации без доступа к шлюзу. Рассмотрим на приведенном примере.

Для того чтобы первый выключатель мог отправлять команды на лампочку даже при отсутствии связи со шлюзом, ему необходимо обладать информацией о том, в каких сценариях автоматизации он участвует. Если распределить все сценарии автоматизации между устройствами, получим схему, представленную на рис. 6.

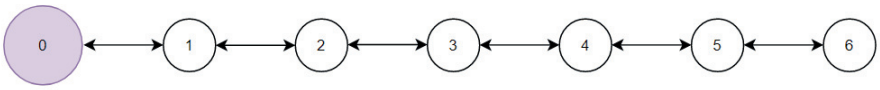


Рис. 7. Пример децентрализованной системы при последовательном подключении устройств

Таблица 1

Сравнение базовой и предложенной архитектур

Критерий	Предложенная архитектура	Базовая архитектура
Хранение сценариев автоматизации	Сценарии необходимо хранить непосредственно на устройствах, которые этот сценарий выполняют	Сценарии хранятся на шлюзе, который считается безотказным элементом системы
Нагрузка сети	Команды сценариев отправляются по кратчайшему пути до устройства-исполнителя	Все команды проходят полный путь от устройства-активатора до шлюза и от шлюза до устройства-исполнителя
Возможна работа сценариев без связи со шлюзом	Возможна, если устройства, участвующие в сценарии, имеют соединение	Нет
Алгоритм добавления или удаления сценария	Сравнительно сложный алгоритм взаимодействия и хранения данных	Простая запись в общей таблице автоматизаций

В данной системе связь первого выключателя со шлюзом необязательна для того, чтобы при нажатии на него включилась или выключилась лампочка. Этот вариант системы может быть особо актуален в тех случаях, когда устройства локализованы в нескольких местах и не имеют надежной связи друг с другом либо когда система выходит вытянутой и передача изменения до шлюза занимает длительное время и нагружает множество устройств, как, например, на рис. 7.

Взамен возможности пользоваться устройствами при отключении от шлюза на устройствах необходимо выделять место под хранение сценариев автоматизаций.

Из табл. 1 следует, что предложенная архитектура имеет как преимущества, так и недостатки. Преимущество состоит в том, что нагрузка на сеть уменьшается и мы получаем возможность работы устройств без соединения со шлюзом, а недостаток – что алгоритм изменения сценариев сложнее и данные необходимо хранить в памяти устройств, которая ограничена. Но на практике число сценариев не превышает количества возможных действий устройства, т. е. примерно равно 2–4 сценариям на одно устройство, и эта информация легко помещается в современных умных устройствах.

В статье была предложена архитектура децентрализованной системы хранения сценариев для умного дома и на качественном уровне проведено сравнение с использующейся в настоящее время централизованной архитектурой с одним главным управляющим устройством. На основе выполненных теоретических исследований в дальнейшем будут проведены реализация, практические исследования и сравнение данных архитектур на количественном уровне.

Библиографический список

1. Умный дом. URL: https://www.intelvision.ru/blog/what-is-smarthome (дата обращения: 06.12.2021).

2. Готовые решения систем умного дома. URL: https://videokontroldoma.ru/gotovye-komplekty-umnyj-dom (дата обращения: 06.12.2021).

3. Mesh network. URL: https://docs.espressif.com/projects/esp-idf/en/latest/esp32/

api-guides/esp-wifi-mesh.html?highlight=mesh (дата обращения: 05.12.2021).

4. *Farahani S.* ZigBee Wireless Networks and Transceivers. 1st ed. Newnes, 2008. 360 p.

5. *Greengard S.* The Internet of Things (The MIT Press Essential Knowledge series). Massachusett: The Mit Press, 2015. 232 p.

6. *Dargie W., Poellabauer C.* Fundamentals of Wireless Sensor Networks: Theory and Practice. John Wiley & Sons, Ltd., 2010. 336 p.

УДК 51-74 **DOI: 10.31799/978-5-8088-1701-2-2022-2-250-252**

Д. К. Ким*
кандидат физико-математических наук, ассоциированный профессор
Г. Д. Георгиев**
кандидат технических наук, исследователь
**Университет Нархоз, Алматы, Казахстан*
***Центр энергетических решений, Варна, Болгария*

ЗАДАЧА ОПТИМИЗАЦИИ НАПРЯЖЕНИЯ ТРАНСФОРМАТОРА ДЛЯ СЕТИ С РАСПРЕДЕЛЕННОЙ ГЕНЕРАЦИЕЙ

Мы изучаем проблему выбора оптимального значения напряжения трансформатора в электрической сети с распределенными источниками энергии. Колебания потребления энергии вместе с изменчивостью солнечной генерации в таких сетях оказывают значительное влияние на профиль напряжения по сравнению с традиционными сетями. Мы рассматриваем стохастическое поведение напряжений узлов как значения случайных величин и применяем известные вероятностные методы для их анализа. В работе исследуется задача оптимального выбора значения трансформатора, при котором максимизируется вероятность нахождения напряжений в интервале допустимых значений.
Ключевые слова: *математическая модель, распределенные источники энергии, солнечная генерация, стохастическая оптимизация.*

D. K. Kim*
PhD, Phys.-Math., Associate Professor
G. D. Georgiev**
PhD, Tech., Researcher
**Narxoz University, Almaty, Kazakhstan*
***Center for energy solutions, Varna, Bulgaria*

OPTIMIZATION PROBLEM FOR TRANSFORMER TAP SETTING IN THE GRID WITH DISTRIBUTED GENERATION

We study the problem of selecting the optimal transformer tap setting in a grid with distributed energy sources. Fluctuations in energy consumption together with the variability of solar generation in such networks have a significant impact on the voltage profile compared to traditional networks. We consider the stochastic behavior of node voltages as values of random variables and apply known probabilistic methods to analyze them. The work investigates the problem of optimal choice of transformer value, at which the probability that voltages belong to an interval of permitted values is maximized.
Keywords: *mathematical model, distributed energy sources, solar power generation, stochastic optimization.*

Введение

Распределенные источники энергии (РИЭ), под которыми в данной работе будем понимать фотоэлектрические модули (ФМ) на крышах зданий, становятся все более распространенными во многих странах [1]. Традиционно поток энергии, вырабатываемой крупными генерирующими станциями, шел от более высоких к более низким уровням напряжений через распределительные сети. С внедрением распределенной генерации потоки энергии становятся двунаправленными, что может привести к значительным проблемам с напряжением и безопасностью сети [2].

В традиционной ситуации (до внедрения РИЭ) напряжение в сети постепенно падает по мере удаления от трансформатора и его минимальные значения наблюдаются в самых удаленных узлах. Тогда возможна ситуация, когда напряжения в узлах сети низкого напряжения (НН) опустятся ниже допустимого значения (нижний предел напряжения). Чтобы не допустить такого случая, устанавливают напряжение выше номинального в начале сети (на трансформаторе).

Генерация РИЭ в сети снижает нагрузку узлов и при достаточно больших значениях, например в середине солнечного дня, может превысить потребление узла, так что часть ее тока поступит в сеть. Таким образом, при умеренном вводе РИЭ нагрузка сети частично снижается и происходит улучшение качества напряжения.

При высокой суммарной мощности распределенной генерации токи в сети могут иметь обратное направление в сторону трансформатора. Тогда напряжения в узлах поднимаются и могут превысить допустимый уровень напряжения (верхний предел напряжения). Если рассматривать вечернее время, когда нет солнечной генерации, то мы имеем традиционный случай – ток течет от трансформатора по направлению к узлам сети и напряжение снижается по мере удаления от трансформатора. Суточные профили нагрузок для сети НН без ФМ и с ФМ показаны на рис. 1 и 2. Тогда установка повышенного напряжения в начале сети, чтобы повысить напряжение в вечернее время, может привести к ухудшению ситуации в дневное время. В Европейском союзе принят стандарт EN50160, согласно которому среднеквадратичное значение напряжения должно находиться в диапазоне ±10% от номинального напряжения в течение не менее 95% времени. Тогда возникает задача выбора оптимального значения напряжения в начале линии, при котором были бы сведены к минимуму ситуации как со слишком низким, так и со слишком высоким напряжением.

В новых условиях необходима информация о состоянии сети [3]. Важно отметить, что в распределительных сетях низкого напряжения (НН) количество доступных измерений в реальном времени обычно очень ограничено [3].

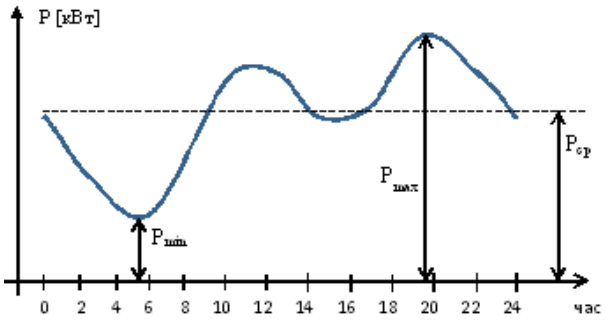


Рис. 1. Суточный профиль нагрузки P без РИЭ

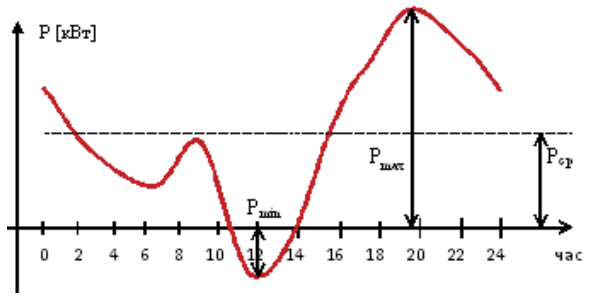


Рис. 2. Суточный профиль нагрузки P с РИЭ

Поэтому в качестве дополнительных данных используются псевдоизмерения с большими погрешностями. Это означает, что оцененное состояние сети будет содержать много неопределенностей.

В данной работе мы предполагаем, что потребляемая суточная мощность отдельно взятого потребителя в фиксированный момент времени есть случайная величина со своей вероятностной характеристикой. Тогда выбор оптимального напряжения в начале сети может быть сведен к максимизации вероятности, что напряжения узлов сети принадлежат области допустимых значений.

Мы рассматриваем распределительную сеть НН со случайными токами, которые зависят от ФМ и нагрузки потребителей. Показываем, как вероятностный подход позволяет изучать напряжения узлов сети и исследуем пример распределительной сети с линейной топологией. Полученные результаты позволяют лучше понять поведение сети и ее свойства.

Математическая модель

Для дальнейших выводов нам понадобится математическая модель сети НН. Математическая модель сети высокого напряжения может быть найдена в [4]. Мы рассмотрим сеть с m+1 узлом и напряжениями в них:

$$U_0, U_1, \ldots, U_m.$$

Положим, что напряжение U_0 в начальном узле сети может быть выбрано произвольным образом. Определим вектор напряжений, за исключением начального узла,

$$\mathbf{U} = \left(U_1, \ldots, U_m\right)^t,$$

вектор токов

$$\mathbf{I} = \left(I_1 \ \ldots, I_m\right)^t$$

и активные сопротивления между узлами сети

$$R_1, R_2, \ldots, R_m.$$

Мы предполагаем, что можем пренебречь реактивными параметрами сети. Зависимость вектора напряжений от вектора токов может быть записана в виде

$$\mathbf{U} = \mathbf{V} + \mathbf{Z} \times \mathbf{I},$$

где $\mathbf{Z}$ – матрица узловых сопротивлений, $\mathbf{V}$ – вектор известных значений. Мы предполагаем, что токи могут быть представлены в следующем виде:

$$I_i = -d_i \times SI + J_i, \quad i = 1,2,\ldots m,$$

где SI – ток от ФМ в сеть; d_i – известные коэффициенты, которые характеризуют технические характеристики солнечных панелей (площадь покрытия, угол наклона и т. п.) узла I; J_i – ток от узла к потребителю.

Положим, что J_i – независимые, одинаково распределенные случайные величины с известным вероятностным распределением. Тогда задачу выбора оптимального значения U_0 можно сформулировать в следующем виде: найти такое U_0, при котором вероятность, что напряжения для любого момента времени суток принадлежат интервалу от 0,9 до 1,1 номинального напряжения $U_{\text{пот}}$ максимальна.

Далее под SI будем понимать максимальное значение тока от солнечной панели в сеть. Обозначим через J_i^d случайные величины во время максимальной солнечной радиации и J_i^u – случайные величины во время максимальной нагрузки, так называемый вечерний пик. Также обозначим через U_i^d напряжение во время максимальной солнечной радиации и U_i^u – напряжение во время вечернего пика. Пусть $a = 0,9U_{\text{пот}}$, $b = 1,1U_{\text{пот}}$ и

$$\left[a,b\right]^m = \left[a,b\right]\times\left[a,b\right]\times\ldots\times\left[a,b\right].$$

Тогда будем искать такое значение U_0, при котором

$$\min\left\{\mathbf{P}\left(\mathbf{U}^d \in \left[a,b\right]^m\right), \mathbf{P}\left(\mathbf{U}^u \in \left[a,b\right]^m\right)\right\} \to \max.$$

Пример

Пусть токи J_i^d как независимые случайные величины имеют нормальное распределение со средним 3 А и стандартным отклонением 1, а токи J_i^u – нормальное распределение со средним 5 А и стандартным отклонением 2. Положим, что $U_{\text{пот}}$=230 В и сеть состоит из $m = 30$ узлов. Для краткости положим, что $d_i = d$ и $R_i = R$, где $d = 1$ и $R = 0,01$ Ом. Тогда оптимальное значение $U_0^* = 239,54$ В и при этом начальном напряжении вероятность того, что все напряжения узлов сети будут принадлежать допустимому интервалу значений, равна 0,96.

Заключение

Ввод распределенных источников энергии в распределительную сеть, например в виде фотоэлектрических модулей на крышах зданий, приводит к серьезной трансформации модели ее развития. Пассивные прежде потребительские узлы превращаются в просьюмеров с возможностью генерации и накопления энергии. Распределительная сеть становится активным элементом энергосистемы с разнонаправленными потоками мощностей и вариациями напряжений, определяемыми нагрузками и генерацией РИЭ. Данная работа посвящена одной из основных проблем массового ввода РИЭ – ухудшению качества напряжения в узлах сети.

Мы рассматриваем математическую модель для сети НН с распределенной генерацией. Токи и напряжения на каждом узле изменяются случайным образом с течением времени, и мы рассматриваем неизвестные параметры как нормально распределенные случайные величины. Такой подход позволяет применять вероятностный аппарат для анализа напряжений в сетях НН. В частности, мы изучили выбор напряжения трансформатора в начале сети, при котором достигается максимальная вероятность нахождения напряжений узлов в области допустимых значений.

Библиографический список

1. *Eisenreich M., Balzer G., Backes J., Maurer B.* Integration of pv systems into low voltage networks using standard load profiles // Proc. International Conference on Renewable Energies and Power Quality. Granada, Spain, 2010.

2. *Kenneth A. P., Folly K.* Voltage rise issue with high penetration of grid connected pv // IFAC Proceedings. 2014. Vol. 47. P. 4959–4966.

3. *Brinkmann B., Negnevitsky M.* A probabilistic approach to observability of distribution networks // IEEE Transactions on power systems. 2016. Vol. 32. P. 1169–1178.

4. *Бостанбеков К. А., Георгиев Г. Д., Ким Д. К.* Математическая модель функционирования энергосистемы с солнечной электрической станцией в структуре генерации // Обработка, передача и защита информации в компьютерных системах '21: Междунар. науч. конф. СПб., 2021. С. 132–137.

УДК 612.087.1:57.087.1 **DOI: 10.31799/978-5-8088-1701-2-2022-2-253-255**

В. С. Коломойцев*
кандидат технических наук, доцент
В. Г. Ерышов*
кандидат технических наук, доцент
А. И. Альмухамедов*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИДЕНТИФИКАЦИЯ ПО РИСУНКУ ВЕН ЛАДОНИ КАК ПЕРСПЕКТИВНЫЙ МЕТОД БИОМЕТРИЧЕСКОЙ ИДЕНТИФИКАЦИИ

Рассмотрены современные методы биометрической идентификации личности, проведен обзор метода идентификации по рисунку вен ладони, а также сравнение данного метода с распространенными методами идентификации.
Ключевые слова: *биометрическая идентификация, метод идентификации, рисунок вен ладони, вероятность ложного допуска, вероятность ложного недопуска.*

V. S. Kolomoitsev*
PhD, Tech., Associate Professor
V. G. Eryshov*
PhD, Tech., Associate Professor
A. I. Almukhamedov*
Student
**St. Petersburg State University of Aerospace Instrumentation*

IDENTIFICATION BY THE PATTERN OF THE VEINS OF THE PALM AS A PROMISING METHOD OF BIOMETRIC IDENTIFICATION

The article discusses modern methods of biometric identification of a person, reviews the method of identification based on the pattern of veins in the palm, and also compares this method with common methods of identification.
Keywords: *biometric identification, identification method, palm vein pattern, false acceptance rate, false rejection rate.*

В общем случае биометрическая идентификация (БИ) [1] определяется как процесс сравнения предоставленного идентификатора, являющегося биометрическим параметром, со всеми записями из базы зарегистрированных пользователей. К настоящему времени в мире имеется ряд широко используемых методов БИ [2], разрабатываются и исследуются перспективные. Целью данного исследования будет выявление перспективного метода БИ, не уступающего распространенным.

Сфера биометрических технологий демонстрирует стремительный рост, и в связи с этим появляется критически важная проблема – определение общих стандартов надежности биометрических систем защиты.

Для разных методов биометрической идентификации используются следующие параметры оценки [3]:

– вероятность ложного допуска (ВЛД) – доля транзакций верификаций самозванца, которые будут ошибочно приняты;

– вероятность ложного недопуска (ВЛНД) – доля транзакций верификации подлинного лица, которые будут ошибочно отвергнуты;

– чувствительность к влиянию внешних факторов;

– устойчивость к подделке;

– возможность бесконтактной идентификации;

– скорость идентификации;

– стоимость.

Методы биометрической идентификации различаются прежде всего по используемым биометрическим идентификаторам, где биометрический идентификатор – это биометрический уникальный признак объекта, по которому объект можно однозначно идентифициро-

вать. В данный момент выделяют четыре самых распространенных метода БИ [2]:

– по отпечатку пальца;
– по трехмерному изображению лица;
– по радужной оболочке глаза;
– по сетчатке глаза.

Около половины всех систем идентификации используют идентификацию по отпечатку пальца. Данные методы имеют совершенно разные параметры, от чего может зависеть область их применения.

Перспективные методы БИ используют следующие биометрические параметры:

– запах тела;
– тон сердца;
– дезоксирибонуклеиновая кислота (ДНК);
– эмоциональное состояние и мимика;
– рисунок вен ладони.

Однако большинство методов нуждается в исследованиях и дальнейших разработках [4]. Ключевой недостаток – достаточно низкая точность, не способная конкурировать с распространенными методами биометрической идентификации, в случае с ДНК-идентификацией – ее крайне низкая скорость. Однако идентификация по рисунку вен ладони лишена данных недостатков.

В основе метода БИ по рисунку вен ладони (РВЛ) лежит свойство крови поглощать излучение в ближнем ИК-диапазоне. Данный эффект может зарегистрировать любая современная камера. Устройство сканера предполагает наличие ИК-подсветки, камеры и ИК-Фильтра. При этом на получаемых монохромных изображениях можно различить рисунок вен.

Сегодня лидер в области идентификации по рисунку вен – японская компания Fujitsu. Данная организация обладает самой обширной базой изображений ладоней, снятых с помощью сканера – более 150000, что представляет огромные возможности для машинного обучения. Однако она была собрана для коммерческих целей, что делает невозможным ее использование в открытом доступе.

Рисунок вен обладает хорошей биометрической характеристикой, так как он не изменяется с течением времени, уникален у каждого человека, на него не влияют внешние дефекты кожи [5]. Можно заметить, что данный метод имеет некоторое сходство с методом сканирования сетчатки глаза, поскольку за объект сравнения также берется рисунок кровеносных сосудов. К тому же сканирование сетчатки глаза на данный момент – один из самых надежных биометрических методов. Однако васкулярное сканирование лишено основного недостатка систем сканирования сетчатки – негативного психологического фактора. К тому же глаза более подвержены болезням, влияющим на рисунок сосудов, например катаракте.

Таким образом, проведем сравнение распространенных методов БИ с выбранным перспективным методом БИ по основным параметрам (табл. 1) [6]. Как можно увидеть, метод идентификации по РВЛ по параметрам устойчивости к подделке и стоимости имеет аналогичные показатели, что и идентификация по радужке и сетчатке, при этом также обладая высокой скоростью и возможностью бесконтактной идентификации.

Таблица 1

Сравнительный анализ параметров для различных методов БИ

Метод БИ	Чувствительность к влиянию внешних факторов	Устойчивость к подделке	Возможность бесконтактной идентификации
По отпечатку пальца	Высокая	Низкая	Безуспешна
По трехмерному изображению лица	Низкая	Средняя	На среднем расстоянии
По радужной оболочке глаза	Средняя	Высокая	На большом расстоянии
По сетчатке	Высокая	Высокая	Безуспешна
По рисунку вен ладони	Средняя	Высокая	На небольшом расстоянии

Идентификация по отпечатку пальца и трехмерному изображению лица – наиболее распространенные на настоящий момент методы.

Переходя к показателям ВЛД и ВЛНД (табл. 2), можно также заметить, что метод идентификации по РВЛ обладает показателями, приближенными к методам по сетчатке и по радужке, в то время как методы идентификации по отпечатку пальца и трехмерному изображению лица показывают худший результат.

Таблица 2

Параметры ВЛД и ВЛНД для различных методов БИ, %

Метод БИ	ВЛД	ВЛНД
По отпечатку пальца	0,001	0,6
По трехмерному изображению лица	0,0005	0,1
По радужной оболочке глаза	0,00001	0,016
По сетчатке	0,0001	0,4
По рисунку вен ладони	0,0008	0,01

Наиболее важными для пользователя являются параметры скорости работы и стоимости

Таблица 3

Скорость идентификации и стоимость для различных методов БИ

Метод БИ	Скорость идентификации	Стоимость
По отпечатку пальца	Высокая	Низкая
По трехмерному изображению лица	Средняя	Средняя
По радужной оболочке глаза	Высокая	Высокая
По сетчатке	Низкая	Высокая
По рисунку вен ладони	Высокая	Высокая

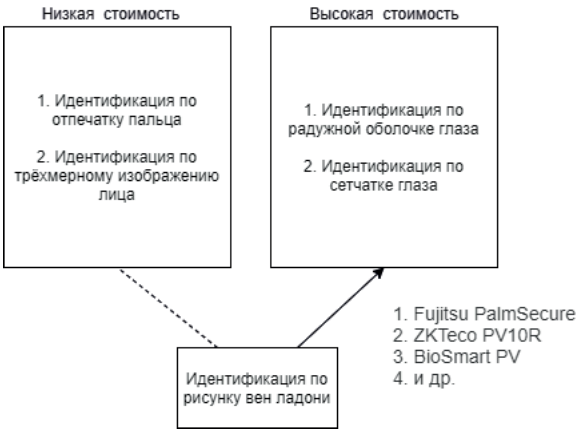


Рис. 1. Распределение методов БИ по стоимости сканеров

сканера. Эти параметры для каждого метода рассмотрены в табл. 3.

Таким образом, распространенные методы идентификации подразделяются на группы, ключевым параметром в которых оказывается стоимость (рис. 1).

Наиболее распространены методы идентификации по отпечатку и трехмерному изображению лица из первой группы. Они обладают высокими показателями ВЛД и ВЛНД, имеют низкую устойчивость к подделке, однако и достаточно низкую стоимость.

Во второй группе низкие параметры ВЛД и ВЛНД, а высокую устойчивость к подделке компенсирует достаточно высокая стоимость. Технологии, реализующие данные методы, сегодня более дорогостоящие, чем технологии, основанные на распространенных методах идентификации: по трехмерному изображению лица и по отпечатку пальца. Это обусловлено применением дорогостоящих материалов и элементов при разработке сканирующих устройств.

На данный момент имеющиеся системы можно отнести во вторую группу как достаточно дорогостоящие, но обладающие высокой точностью и низкими показателями ВЛД и ВЛНД. Однако имеется возможность путем изменения конструкционных особенностей добиться результатов, которые позволили бы отнести данный метод к первой группе.

Выводы

Системы идентификации по РВЛ не уступают современным системам, использующим распространенные методы БИ. Целью дальнейшего исследования будет разработка сканера РВЛ, сохраняющего прежние характеристики при понижении стоимости разработки и повышении показателей ВЛД и ВЛНД до уровня методов из группы с низкой стоимостью.

Библиографический список

1. Руководство по биометрии / Р. М. Болл, Дж. Х. Коннел, Ш. Панканти [и др.]. М.: Техносфера, 2007. 21 с.

2. Исследование российского рынка биометрических технологий 2018–2022 гг. // JSON.TV: информ.-аналит. портал. URL: https://json.tv/ict_telecom_analytics_view/issledovanie-rossiyskogo-rynka-biometricheskih-tehnologiy-2018-2022-gg-20181130015609 (дата обращения: 14.04.2021).

3. ГОСТ Р ИСО/МЭК 19795-1–2007. Автоматическая идентификация. Идентификация биометрическая. Эксплуатационные испытания и протоколы испытаний в биометрии. М.: Стандартинформ, 2019. 6 с.

4. Перспективные методы биометрической аутентификации и идентификации / В. М. Колешко, Е. А. Воробей, П. М. Азизов [и др.]. Минск: БНТУ, 2009. 94 с.

5. *Tome P., Marcel S.* Palm Vein Database and Experimental Framework for Reproducible Research // 2015 International Conference of the Biometrics Special Interest Group (BIOSIG). 2015. С. 1–7.

6. *Моржаков В. А.* Современные биометрические методы идентификации // Безопасность. Достоверность. Информация. 2009. № 2. С. 44–48.

УДК 004.7 **DOI: 10.31799/978-5-8088-1701-2-2022-2-256-260**

В. В. Комашинский*
кандидат технических наук, доцент
А. О. Скрылёв*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ОБ ИНТЕЛЛЕКТУАЛЬНОМ ОБНАРУЖЕНИИ АНОМАЛЬНОГО ПОВЕДЕНИЯ ПОЛЬЗОВАТЕЛЯ В СЕТИ

Рассматривается технология анализа модели поведения пользователя в задаче обнаружения аномального поведения. Предлагается алгоритм машинного обучения, основанный на нейронной сети, позволяющий дать оценку поведению пользователя в соответствии с его моделью действий.
Ключевые слова: *системы обнаружения вторжений, аномальное поведение пользователя, модель поведения пользователя, нейронные сети.*

V. V. Komashinsky
PhD, Tech., Associate Professor
A. O. Skrylev
Student
**St. Petersburg State University of Aerospace Instrumentation*

ABOUT INTELLIGENT DETECTION OF ABNORMAL USER BEHAVIOR ON THE NETWORK

The article describes the technology of analysis of the user behavior model in the problem of detecting abnormal behavior. A machine learning algorithm based on a neural network is proposed, which makes it possible to assess user behavior in accordance with his action model.
Keywords: *intrusion detection systems, abnormal user behavior, user behavior model, neural networks.*

Введение

Один из методов обеспечения безопасности информационных систем сегодня – анализ аномальной активности. У аномалий существуют разные причины происхождения, которые могут быть связаны как с деятельностью злоумышленников, невнимательностью легитимных пользователей, так и неисправностью аппаратного и программного обеспечения.

Обнаружение аномальной деятельности

Для выявления аномалий могут быть использованы системы обнаружения вторжений. Относительно новое и актуальное решение – технология анализа поведения пользователей и сущностей – User and Entity Behavior Analytics (UEBA/UBA). Наиболее распространенной реализацией указанной технологии выступают системы от таких вендоров, как IBM, Exabeam, ArcSight, Forcepoint, Splunk, Securonix, Varonis. Все эти системы активно используются в различных направлениях (рис. 1) [1].

Преимущество методов защиты, основанных на обнаружении аномалий, состоит в возможности выявления даже новых видов атак, поскольку их сигнатуры еще неизвестны. Такой подход подразумевает сравнение поведения пользователя за некоторый период его работы с моделью нормальной деятельности [2]. Задача – разработка метода с использованием искусственной нейронной сети, который позволит детектировать отклонения в поведении пользователя. Варианты реализации данного подхода представлены обширной классификацией методов обнаружения (рис. 2).

Среди указанных методов есть статистический и кластерный анализы, байесовские сети и др. В качестве метода реализации используется нейронная сеть. Целесообразность выбора данного подхода обусловлена предварительно проведенным анализом. Ее достоинства – гибкость, быстродействие, возможность

Рис. 1. Актуальные направления использования решений UEBA/UBA

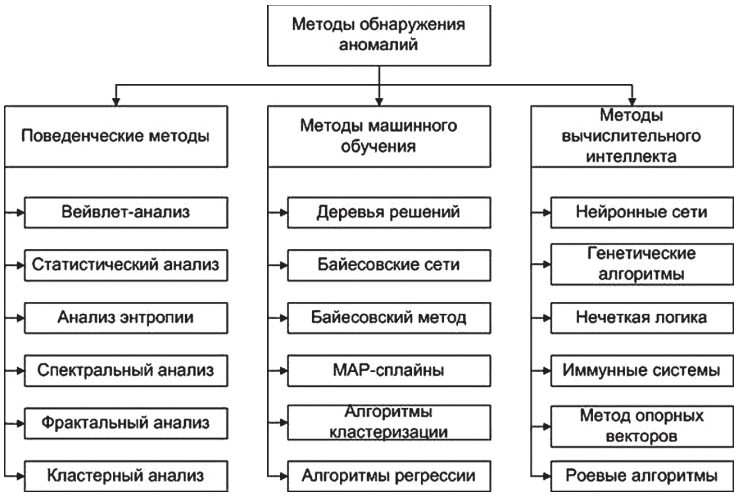


Рис. 2. Методы подхода к обнаружению аномального поведения

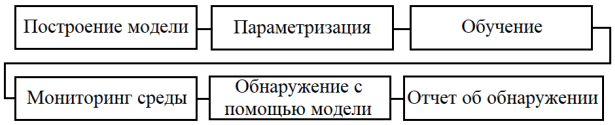


Рис. 3. Схема обнаружения аномального поведения

прогнозирования [3]. Общий принцип работы систем обнаружения аномального поведения с помощью нейронных сетей представлен на рис. 3 [4].

Сначала строится модель нормального поведения пользователя. Затем она переводится в параметры, которые будут подаваться на вход нейронной сети, после чего сеть будет обучаться на основе этой модели. В результате нейронная сеть будет способна определить, есть ли в поведении пользователя какие-либо отклонения и сигнализировать об этом.

Оценка поведения пользователей

Некоторые варианты реализации оценки поведения пользователя [5].

– Обработка таких параметров работы пользователя, как: время начала и окончания работы, используемые сетевые ресурсы, типы операций и т. д. Данные параметры подаются на вход нейронной сети. В качестве выхода сети выступает один нейрон, выдающий коэффициент нормальности: 0 – нормальное поведение, 1 – аномальное.

– Детектор атак, определяющий поведение пользователя, основываясь на количестве исполнения различных команд в течение заданного интервала времени. В данной модели учитывается только количество команд и не учитывается их последовательность.

– Аномальное поведение пользователя выявляется на основе последовательности выполня-

емых от его имени команд. Суть метода заключается в том, что на основе выполняемых ЭВМ команд пользователя осуществляется прогноз следующей команды. По количеству отклонений за заданный промежуток времени делается вывод о наличии аномалии. Таким образом, задача идентификации в данном подходе сведена к прогнозированию временного ряда. Команды берутся из системных журналов ОС, кодируются и подаются на вход искусственной нейронной сети.

Особенности реализации обнаружения аномального поведения

Предлагается использовать метод обнаружения аномалий, основанный на анализе вектора, характеризующего поведение пользователя. Вектор представляет собой модель нормального поведения пользователя в информационной системе. Информационная система представляет собой сеть компьютеров, имеющих доступ в Интернет. В свою очередь рассматривается конкретный узел – компьютер, на котором хранится защищаемая информация. Информация представляет собой как сведения, отнесенные к коммерческой тайне, так и текущие результаты работы пользователя. В качестве пользователя, модель которого будет строиться, был выбран сотрудник маркетингового отдела.

Модель сотрудника можно описать с помощью следующих характеристик.

Режим работы:

– T_start – время начала рабочего дня;

– T_end – время окончания рабочего дня;

– Day – дни, когда сотрудник работает.

Особенности работы.

– Тип событий и их количество:

– N_mail – часто обращается к почтовым сервисам;

– N_office – пользуется в основном программами Microsoft Office;

– N_social – активно обращается к социальным сетям;

– N_related_apps – пользуется специальным ПО, которое компания предоставляет сотрудникам данного отдела;

– N_related_dirs – обращается к папкам, относящимся только к своей сфере деятельности (папки с маркетинговыми стратегиями развития, базами данных клиентов, текстами для рассылок и т. д.).

В качестве параметров в данную модель подставляются данные из журналов мониторируемых программ. Формируется вектор с данными параметрами, который представляется в виде числовых значений, подаваемых на вход нейронной сети. Так как используется метод обучения с учителем, данному вектору ставится в соответствие метка 0. Также формируется вектор с отклонениями в параметрах, который будет представлять аномальное поведение. Из векторов с нормальным и аномальным поведением формируется база данных, с помощью которой будет обучаться нейросеть.

При реализации метода была использована библиотека TensorFlow.

В качестве нейросети была использована Sequential, модель которой представляет собой линейную совокупность слоев:

model = Sequential().

По архитектуре сеть представляет собой многослойный персептрон (Multilayer Perceptron, MLP) (рис. 4). Нейросеть состоит из нескольких слоев нейронов, где каждый нейрон соединен со всеми нейронами последующего слоя [6].

Выбор типа нейронной сети обусловлен тем, что данная конфигурация нейронной сети подходит для бинарной классификации. Происходит разделение на нормальное и аномальное поведение.

На каждом слое используется метод dense для полного соединения слоев друг с другом. Функция активации слоев – ReLu, так как она подходит для сетей, на выходе которых нет неограниченной области определения. ReLu возвращает значение x, если x положительно, и 0 в противном случае:

$$A=\max(0, x).$$

В последнем слое используется функция sigmoid, поскольку итоговое значение слоя представляет одну из двух категорий классификации:

$$A = \frac{1}{1+e^{-x}}.$$

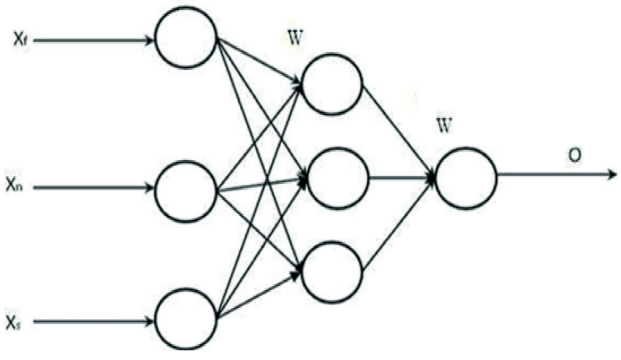


Рис. 4. Схема MLP

В итоге нейронная сеть состоит из четырех слоев – входного, двух скрытых и выходного:

model.add(Dense(20, input_dim=8, activation='relu')) – входной

model.add(Dense(10, activation='relu')) – скрытый

model.add(Dense(5, activation='relu')) – скрытый

model.add(Dense(1, activation='sigmoid')) – выходной.

Функция потерь – binary cross entropy, также используется потому, что на выходе должно быть получено два значения:

$$BCE(pt, pp) = -\frac{1}{N}(pt * \log(pp) + (1 - pt) * \log(1 - pp)).$$

model.compile(loss="binary_crossentropy", optimizer="adam",metrics=['accuracy'])

history = model.fit(X, Y, validation_split=0.15, epochs = 300, batch_size=10)

Полученные результаты

В результате проектирования нейронной сети получились зависимости. График потерь (рис. 5) показывает, насколько нейросеть ошибается. Значение функции потерь необходимо свести к минимуму. На графике видно, что уже на первых эпохах наблюдается значительное снижение потерь. График точности (рис. 6) демонстрирует, как точно нейросеть может определить аномальное поведение. Функция потерь составляет 43,54% на тренировочной выборке и

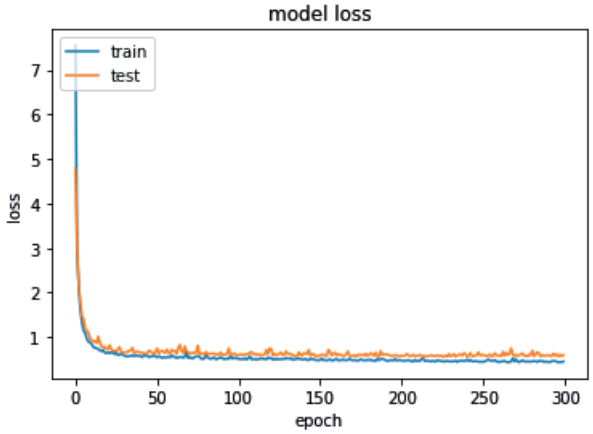


Рис. 5. График потерь

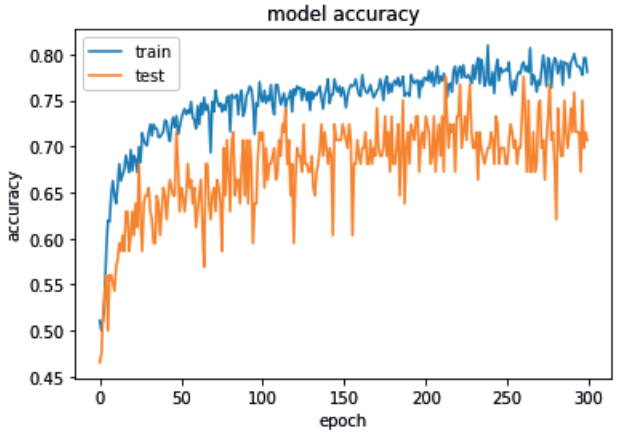


Рис. 6. График точности

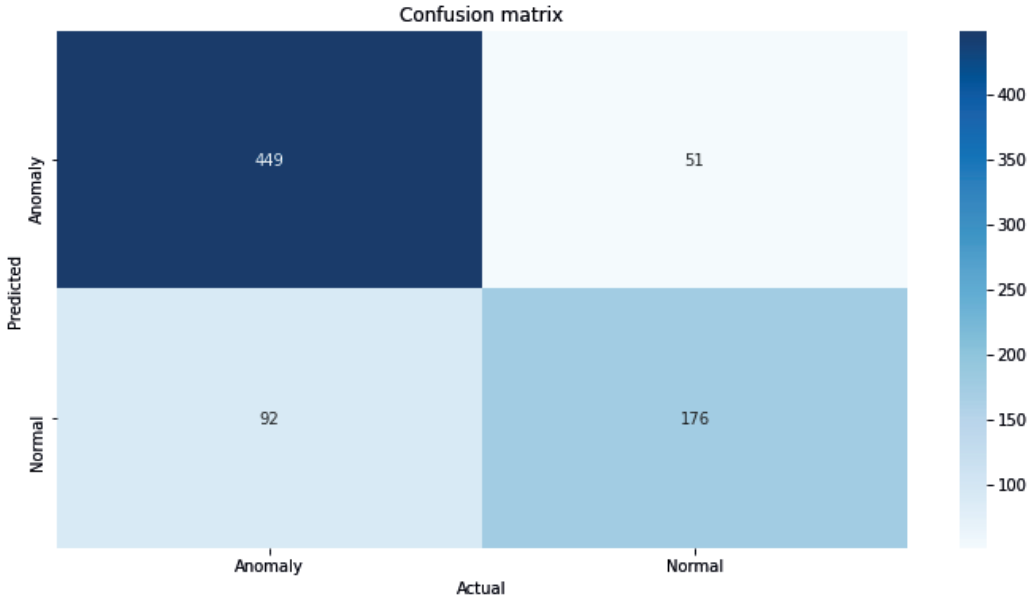


Рис. 7. Матрица ошибок 1-го и 2-го рода

56,77% – на тестовой, а точность – 78,52% для тестовой и 68,87% – для тренировочной.

В результате тестов была построена матрица ошибок (рис. 7), которая показывает те случаи, где нейросеть допускала ошибки. Из 541 аномального случая было верно определено 449, а из 227 случаев с нормальным поведением – 176.

Заключение

Реализованная модель обнаружения аномального поведения может быть улучшена путем расширения используемой в реализации базы данных. При этом надо понимать, что в процессе работы нейронной сети не исключены случаи ложного срабатывания. Поэтому каждый случай обнаружения аномального поведения у пользователя обязан быть рассмотрен администратором безопасности, эксплуатирующим соответствующую систему обнаружений.

Библиографический список

1. *Саенков П. А.* Использование методов и алгоритмов анализа данных в мобильной UEBA/DSS-системе для решения задач информационной безопасности // Известия ТулГУ. Технические науки. 2019. Вып. 12. С. 585–588.

2. *Браницкий А. А., Котенко И. В.* Анализ и классификация методов обнаружения сетевых атак // Тр.СПИИРАН. 2016. Вып. 45. С. 207–244.

3. *Абрамов Е. С., Аникеев М. В., Макаревич О. Б.* Подготовка данных для использования в обучении и тестировании нейросетей при обнаружении сетевых атак // Известия ТРТУ. 2003. № 4 (33). С. 204–206.

4. *Зуев В. Н., Ефимов А. Ю.* Нейросетевой поведенческий анализ действий пользователя в целях обнаружения вторжений уровня узла // Программные продукты и системы. 2019. № 2. С. 268–272.

5. *Гафаров Ф. М., Галимянов А. Ф.* Искусственные нейронные сети и их приложения: учеб. пособие. Казань: Изд-во Казан. ун-та, 2018. 121 с.

6. *Катасев А. С., Катасева Д. В., Кирпичников А. П.* Нейросетевая диагностика аномальной сетевой активности. Казань: КНИТУ, 2015. 164 с.

УДК 004.056.2

А. С. Лучкин*
студент
В. В. Комашинский*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

НЕЙРОСЕТЕВЫЕ МЕТОДЫ И АЛГОРИТМЫ ДЛЯ ПОВЫШЕНИЯ ЗАЩИЩЕННОСТИ ИНФОРМАЦИОННЫХ СИСТЕМ

Рассмотрены различные подходы к обучению нейросетевых структур для применения нейросетевых методов и алгоритмов в целях повышения защищенности информационных систем путем подбора наиболее подходящих нейросетевых структур, способных по своим характеристикам минимизировать различные типы угроз для информационных систем.
Ключевые слова: *нейросетевые методы и алгоритмы, обучение нейросетевых структур, методы обучения нейросетевых структур, защита информационных систем.*

A. S. Luchkin*
Student
V. V. Komashinsky*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

NEURAL NETWORK METHODS AND ALGORITHMS TO IMPROVE THE SECURITY OF INFORMATION SYSTEMS

Various approaches to training neural network structures for neural network methods and algorithms for applying methods of protecting information systems by selecting the most appropriate neural network structures are considered.
Keywords: *neural network methods and algorithms, training of neural network structures, methods of training neural network structures, protection of information systems.*

Введение

В настоящее время технологии машинного обучения получают все большее распространение во многих сферах жизни человека. Они используются для анализа больших данных, оценки изображений и других задач. Данная технология помогает находить закономерности на больших объемах данных, что позволяет рассматривать их как инструмент для анализа и повышения защищенности информационных систем.

Общие подходы к обучению

Обучение – это базовый компонент, необходимый для создания интеллекта. Люди черпают свой интеллект в способности мозга учиться на собственном опыте и использовать его, чтобы справиться при столкновении с существующими и новыми ситуациями. Воспроизведение человеческого интеллекта в машинах и компьютерах – цель методов искусственного интеллекта.

Искусственные нейронные сети (ИНС) – это модели, разработанные для имитации способности к обучению человеческого мозга. Как и у людей, обучение, проверка и тестирование являются важными компонентами в создании таких вычислительных моделей. Искусственные нейронные сети обучаются на основе некоторых наборов данных (которые могут быть помечены или немаркированы) и вычисляют корректировку для своих внутренних переменных посредством моделирования.

Основываясь на правилах обучения и методах обучения, обучение в ИНС можно разделить на контролируемое, подкрепляющее и обучение без учителя.

В контролируемом обучении, как следует из названия, искусственная нейронная сеть находится под наблюдением учителя, который использует свои знания среды для обучения сети

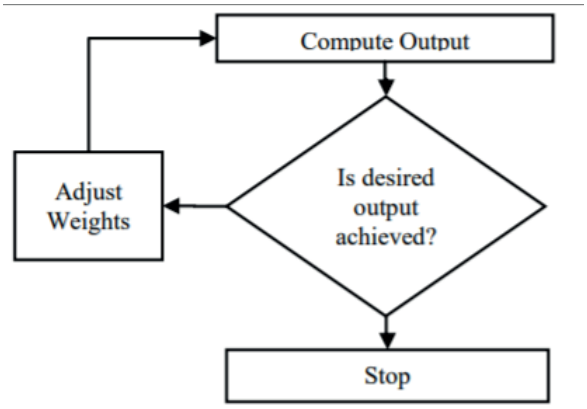


Рис. 1. Схема подконтрольного обучения

с помеченными наборами данных. Следовательно, искусственные нейронные сети учатся, получая входные данные и искомые метки (ожидаемое предсказание), затем строят на основе входа предсказание. После этого посредством использования предсказания и меток вычисляется ошибка, которая используется для корректировки внутренних параметров системы. Тонкая настройка сети продолжается до набора весов, которые минимизируют несоответствие между выводом и желаемым выводом. На рис. 1 показана блок-схема, которая демонстрирует контролируемое обучение в ИНС. Результатом обучения с учителем может быть классификатор или предсказатель, но применение этого метода ограничено, так как учитель может обладать не всеми возможными входными данными и метками, что не позволяет покрыть все случаи.

Обучение без учителя используется, когда невозможно разметить наборы данных идентификаторами классов (метками). Такое положение возникает в ситуациях, когда нет знаний об окружающей среде или стоимость их приобретения слишком высока. При обучении без учителя ИНС не находится под наблюдением. Вместо этого модель обучается на немаркированных наборах данных и пытается найти закономерности между ними. В этой ситуации ИНС учится классифицировать данные, используя расстояние между кластерами.

Методы и алгоритмы искусственных нейронных сетей без учителя

В данном разделе приведен обзор алгоритмов нейронных сетей без учителя.

Самоорганизующиеся карты

Самоорганизующиеся карты – основной тип искусственных нейронных сетей, которые основаны на методе обучения без учителя и используют сходство между данными.

Самоорганизующиеся карты являются биологически вдохновленными топографически вычислительными картами, которые учатся за счет самоорганизации своих нейронов. Эти карты состоят из входных и выходных нейронов без скрытых слоев, что приводит к конкурентному обучению, процессу, в котором все выходные нейроны конкурируют друг с другом. Победитель таких соревнований запускается и обозначается как нейрон-победитель.

Для вычисления результатов выходных нейронов используется метрика, например евклидово расстояние. Обычно евклидово расстояние между нейроном и вектором признаков текущей выборки подается на вход. Выбранные (победившие) нейроны из группы нейронов, которые расположены в узлах решетки, настроены на различные входные шаблоны (структуры), организуются и формируют топографическую карту решетчатой структуры.

Алгоритм самоорганизации карт включает пять этапов:

1) инициализация;

2) выборка;

3) поиск нейрона-победителя, чей весовой вектор наилучшим образом соответствует входному вектору;

4) обновление веса победившего нейрона.

Данный процесс повторяется до тех пор, пока не будет достигнута неизменность весов [1].

Сеть Кохонена (рис. 2) – разновидность самоорганизующихся карт.

Самостоятельная организация карт в основном применяется в искусственных нейронных сетях для кластеризации и сопоставления. Функции, похожие на мозг, являются наиболее подходящими для применения в области исследовательских данных [1].

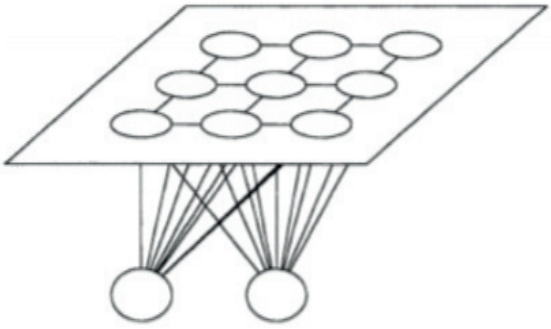


Рис. 2. Сеть Кохонена

Метод «Автокодировщик»

Автокодировщик – архитектура ИНС, которая позволяет производить обучение без учителя с использованием метода обратного распространения ошибок. Общая структура автокодировщика – это сеть прямого распространения, в которой нет обратных связей. Количество нейронов в выходном слое равно их количеству во входном слое (рис. 3).

Основной принцип работы и обучения сети автокодировщика – получить на выходном слое отклик, наиболее близкий к входному. Для того чтобы избежать прямой связи выхода с входом без обработки данных, необходимо чтобы промежуточный слой содержал меньше активных узлов, чем входной слой. Эти ограничения позволяют нейросети искать обобщения и корреляцию в поступающих на вход данных, выполнять их сжатие. Таким образом, нейросеть обучается выделять общие признаки в входных данных, которые записываются в узлах сети.

Два основных практических применения автоэнкодеров для визуализации данных:

– сглаживание шума;

– снижение размерности.

Автоэнкодеры обучаются автоматически на примерах данных. Это означает, что легко натренировать части алгоритма, которые будут затем хорошо работать на конкретном типе ввода и не будут требовать применения новой техники, а только соответствующие данные для обучения. Автокодеры обучены как сохранять информацию, так и придавать новым представлениям разные свойства. Для этого используют разные типы автоэнкодеров.

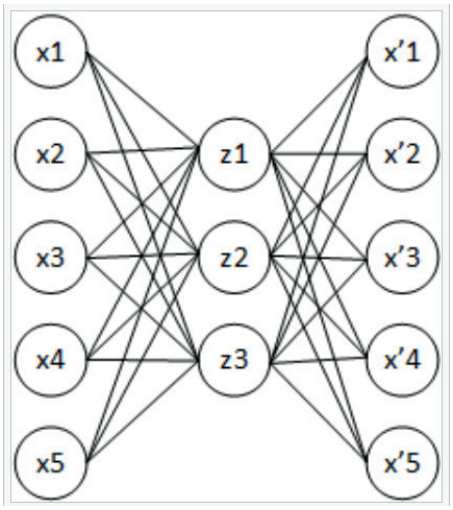


Рис. 3. Схема автодекодера

Применение алгоритмов обучения без учителя в информационной безопасности

Применение алгоритмов без учителя описано в [2]. Данный тип обучения может использоваться в качестве детектора вредоносного ПО.

Есть несколько способов обнаружить вредоносное ПО. Например, были разработаны несколько вариантов классификаторов, обученных на исходных кодах приложения для системы Android. При этом была достигнута точность предсказания 96,76% [2].

Были протестированы несколько конфигураций, и двухслойный вариант автокодировщика оказался наиболее успешным. Эти результаты лучше случайного леса, наивного байесовского метода, логистической регрессии и машины опорных векторов (SVM). Pascanu et al. [2] разработали метод обнаружения вредоносных программ, который использует обучения без учителя в сочетании с многослойным персептроном (MLP) и логистической регрессией для классификации.

Указанный тип обучения возможно использовать для уменьшения количества данных об инцидентах ИБ, так как его методы подходят для разбиения на классы входных данных. Эта особенность позволит предварительно разбить исходную выборку на классы, а затем использовать ее для обучения моделей с учителем или дальнейшего анализа. Для этой задачи могут применяться самоорганизующиеся карты.

Кроме вышеперечисленных задач, автодекодеры используются для защиты от DGA. DGA – это часто используемые вредоносные инструменты, генерирующие большое количество доменных имен, которые могут использоваться для сложных коммуникаций. Большое количество разнородных доменов затрудняют блокировку вредоносных доменов с помощью стандартных методов, таких как черный список. Классификатор, построенный по принципу автодекодера, с вероятностью 70% определяет вредоносные домены [2]. Также автодекодеры позволяют проводить аутентификацию пользователя на основе его особенностей.

Выводы по разделу

Методы обучения без учителя позволяют обучать модели в ситуации, когда невозможно достоверно разбить входные данные на классы. Модели, использующие данный метод обучения, могут использоваться в задачах классификации и предсказания, что позволяет приме-

нять их, например, в задачах поиска вредоносного ПО и разбиения исходных данных на классы для дальнейшего их анализа.

Методы и алгоритмы искусственных нейронных сетей с учителем

В данном разделе приведен обзор алгоритмов нейронных сетей с учителем.

Общее описание

Обучение с учителем – один из способов машинного обучения, в ходе которого испытуемая система принудительно обучается с помощью примеров входных данных и ожидаемого выхода. Нейронная сеть позволяет воссоздать зависимость между входными и выходными данными на основе обучающей выборки. Данная зависимость в теории позволяет аппроксимировать данные из реальности и на основе полученной закономерности строить некие предсказания с некоторой точностью. Для измерения точности ответов, так же как и в обучении на примерах, может вводиться функционал качества.

В качестве модели возможно использовать нейронные сети. Пример нейронной сети приведен на рис. 4.

Метод коррекции ошибки

Метод коррекции ошибок – метод машинного обучения, при использовании которого веса связей не изменяются до тех пор, пока отклик модели остается правильным. Как только сеть выдает неверный отклик, веса изменяются на единицу со знаком, противоположным ошибке предсказания.

Модификации метода

В теореме сходимости рассматриваются различные виды этого метода, доказано, что любой из них позволяет получить схождение при решении любой задачи классификации.

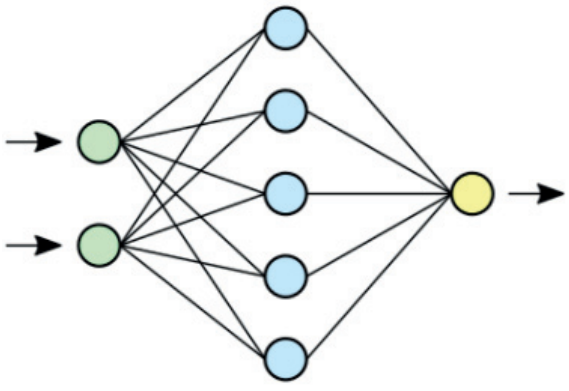


Рис. 4. Схема классической нейронной сети

Метод коррекции ошибок без квантования

Если реакция на данные S_i правильная, то никакого изменения в весах не вводится, но при появлении неверного отклика к весу каждого активного А-элемента прибавляется величина $\gamma = p_i \Delta x_i$, где Δx_i – число единиц подкрепления, выбирается так, чтобы величина сигнала превышала порог θ, а $p_i = +1$ при условии $if S_i^+$ или $p_i = -1$ при условии $if S_i^-$. При этом S_i^+ – отклик, принадлежащий положительному классу, а S_i^- – стимул, принадлежащий отрицательному классу.

Метод коррекции ошибок с квантованием

Отличается от метода коррекции ошибок без квантования только тем, что $\Delta x_i = 1$, т. е. равно одной единице подкрепления.

Данный метод и метод коррекции ошибок без квантования одинаковы по скорости достижения решения в общем случае и более эффективны по сравнению с методами коррекции ошибок со случайным знаком или случайными возмущениями.

Метод коррекции ошибок со случайным знаком подкрепления

Знак подкрепления γ выбирается случайно независимо от отклика нейронной сети и с равной вероятностью может быть положительным или отрицательным. Но, так же как и в базовом методе, если нейронная сеть дает правильную реакцию, то подкрепление равно нулю.

Метод коррекции ошибок со случайными возмущениями

Величина и знак γ для каждой связи в системе выбираются отдельно и независимо в соответствии с некоторым распределением вероятностей. Это метод приводит к самой медленной сходимости, по сравнению с описанными модификациями.

Метод обратного распространения ошибки

Метод обратного распространения ошибки— метод вычисления градиента, который используется при обновлении весов многослойной нейронной сети. Основная идея метода состоит в распространении сигналов ошибки от выходов сети к ее входам в направлении, обратном

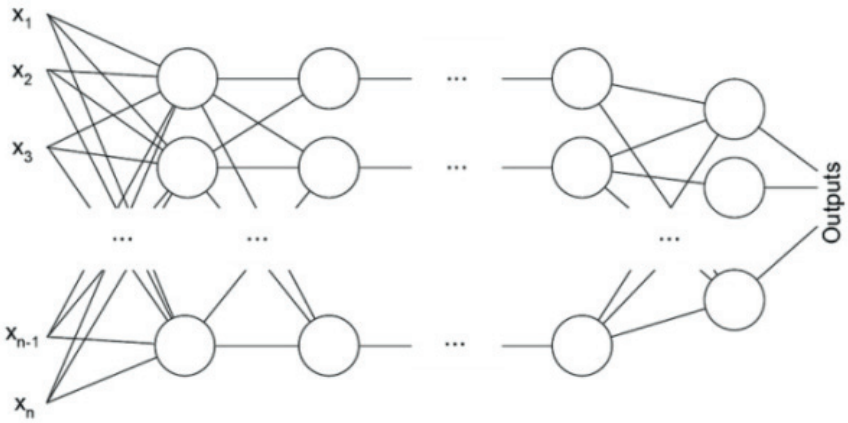


Рис. 5. Схема обратного распространения ошибок

прямому распространению сигналов в обычном режиме работы (рис. 5).

Метод опорных векторов

Метод опорных векторов (SVM) – набор схожих алгоритмов обучения с учителем, использующихся для задач классификации и регрессионного анализа (рис. 6). Основная идея метода – перевод исходных векторов в пространство более высокой размерности и поиск разделяющей гиперплоскости с наибольшим зазором в этом пространстве. Две параллельные гиперплоскости строятся по обеим сторонам гиперплоскости, разделяющей классы. Разделяющей гиперплоскостью будет гиперплоскость, создающая наибольшее расстояние до двух параллельных гиперплоскостей. Алгоритм основан на допущении, что чем больше разница или расстояние между этими параллельными гиперплоскостями, тем меньше будет средняя ошибка классификатора.

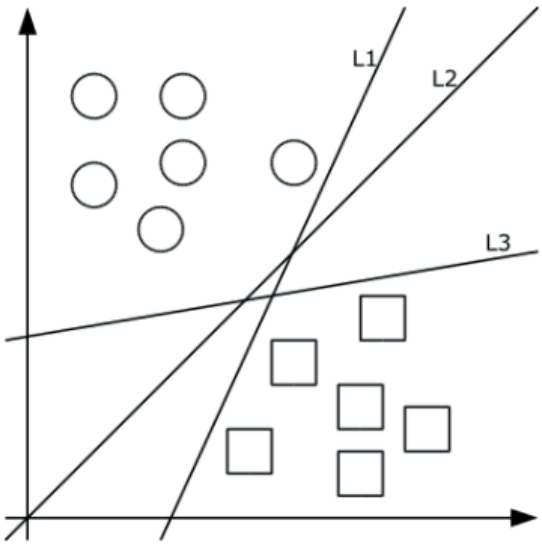


Рис. 6. Графическое представление метода опорных векторов

Результирующий алгоритм крайне похож на алгоритм линейной классификации с той лишь разницей, что каждое скалярное произведение заменяется нелинейной функцией ядра (скалярным произведением в пространстве с большей размерностью). В этом пространстве уже может существовать оптимальная разделяющая гиперплоскость. Так как размерность получаемого пространства может быть больше размерности исходного, то преобразование, сопоставляющее скалярные произведения, будет нелинейным, а значит функция, соответствующая в исходном пространстве оптимальной разделяющей гиперплоскости, будет также нелинейной.

Если исходное пространство имеет достаточно высокую размерность, то выборка может быть линейно разделимой.

Наиболее распространенные ядра.

1. Полиномиальное (однородное).
2. Полиномиальное (неоднородное).
3. Радиальная базисная функция.
4. Радиальная базисная функция Гаусса.
5. Сигмоид.

Применение методов обучения с учителем в информационной безопасности

В контексте информационной безопасности методы обучения с учителем имеют достаточно широкий список применения. С их помощью возможно решать следующие задачи.

1. Обнаружение вредоносного ПО.

Примером реализации этой задачи может служить реализованный Сакси и Берлин детектор вредоносного ПО, точность которого, по их утверждениям, составляет 90–95%. Для реализации они использовали четырехуровневую нейронную сеть (вводной слой, два скрытых слоя и выходной слой) с использованием параметрической ректифицированной линейной или сигмоидной активации функции.

2. Классфикации.

3. Атаки на скачивание.

Используя недостатки плагинов, злоумышленник может перенаправлять пользователей с часто используемых веб-сайтов на веб-сайты, где код эксплойта вынуждает пользователей скачать и запустить вредоносное ПО. Сеть, построенная на анализе url-ресурса, позволила с вероятностью близкой к 90% обнаруживать переходы на данные ресурсы и блокировать их. Это позволяет экономить ресурсы других систем защиты.

4. Идентификация спама.

5. Обнаружение сетевых вторжений [3]. Для этого сети обучают на большом объеме исторических данных об угрозах.

Выводы по разделу

Использование нейронных сетей с учителем позволяет решать большой спектр задач. Существует несколько методов обучения и их модификаций. Главная особенность данного класса методов – необходимость наличия обучающей (размеченной) выборки, что иногда становится сложной задачей.

Кортикоморфные нейронные сети

В данном разделе будут рассмотрены кортикоморфные нейронные сети.

Описание

Кортикоморфные, подобные коре мозга, искусственные нейронные сети были разработаны для обеспечения быстрого обучения без необходимости предъявления нейросетям датасетов из тысяч и миллионов примеров. Данный тип сети относится к классу биоморфных моделей (рис. 7).

Кортикоморфные сети строятся как ламинарные (слоистые) структуры, состоящие из вертикально ориентированных колонок соединенных между собой искусственных нейронов с четкими функциональными ролями слоев и ядер. Иерархически соединенные между собой

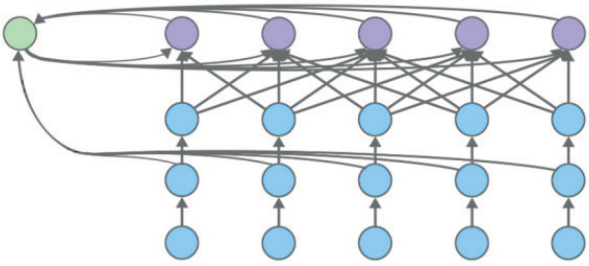


Рис. 7. Структура кортикоморфной сети

колонки нейронов образуют функциональные поля, обеспечивающие обработку данных различного уровня сложности в искусственном кортексе. В рамках кортикоморфных нейросетей обеспечивается запоминание объектов, их припоминание с учетом введенного ассоциативного основания (логика), самостоятельное продолжающееся ассоциирование, ветвление ассоциаций, торможение неправильного, забывание несущественного, различные виды системной реконсолидации памяти и другие когнитивные функции. Важный элемент кортикоморфных нейросетей – кодирующие семантические объекты афферентно-инвариантные нейроны (модели пирамидальных клеток коры, их иногда называют «нейронами бабушки»), способные активироваться определенной структурой входов своего рецептивного поля, а также нейроны новизны и другие типы нейронов.

Особенность биологических моделей нейронов – большая требовательность к вычислительным ресурсам по сравнению с классическими нейронными сетями.

Применение в информационной безопасности

Применение биоморфных нейронных сетей, к которым относится кортикоморфная, возможно в области распознавания аудиовизуального контента. Данные модели решают следующие задачи ИБ.

1. Анализ изображения на предмет аномалий.

2. Анализ видеоряда.

3. Авторизация пользователей с помощью их отличительных особенностей, например голоса.

Заключение

В работе были рассмотрены подходы к обучению нейронных сетей, а именно обучение без учителя, обучение с учителем и кортикоморф-

ные нейронные сети. Каждый имеет сильные и слабые стороны, области применения.

Также было описано несколько прикладных задач информационной безопасности, к которым возможно применить нейронные сети для повышения защищенности информационных систем, что при корректной настройке позволит снизить число ложных срабатываний средств защиты, а также уменьшить нагрузку на сотрудников.

Для решения задач классификации лучше всего подходят методы обучения без учителя. Это позволяет использовать нейросети в задачах обнаружения вредоносного ПО, защиты от DGA и аутентификации пользователей. Методы обучения с учителем хорошо себя показывают в задачах обнаружения сетевых вторжений, идентификации (например идентификации спама) и защиты от атаки на скачивание. Недостатком является необходимость иметь изначально маркированную тренировочную выборку, что может быть осложнено в некоторых случаях. Кортикоморфные нейронные сети неплохо справляются с задачами анализа аудиовизуального контента, что позволит использовать их при анализе изображений, видеоконтента, а также аутентификации пользователей по некотором признакам.

Библиографический список

1. *Fabiyi S. D.* A Review of Unsupervised Artificial Neural Networks with Applications. URL: https://www.researchgate.net/publication/331133951_A_Review_of_Unsupervised_Artificial_Neural_Networks_with_Applications (дата обращения: 07.12.2021).

2. *Berman D. S., Buczak A. L., Chavis J. S., Corbett C. L.* A Survey of Deep Learning Methods for Cyber Security. 2019. URL: https://www.mdpi.com/2078-2489/10/4/122/ (дата обращения: 07.12.2021).

3. *Lee J., Kim J., Kim I., Han K.* Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles. 2019. URL: https://ieeexplore.ieee.org/document/8896978 (дата обращения: 07.12.2021).

УДК 004.056.5 **DOI: 10.31799/978-5-8088-1701-2-2022-2-268-273**

П. П. Недошивин*
студент
В. В. Комашинский*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

МЕТОДЫ ТЕОРИИ ИГР ДЛЯ ОБЕСПЕЧЕНИЯ БЕЗОПАСНОСТИ СВЯЗИ НА КОММУТИРУЕМЫХ СИСТЕМАХ

Рассмотрены существующие подходы к применению методов теории игр для анализа поведения злоумышленника и работы системы защиты при атаке на автономные коммутируемые системы, например такие как транспортные средства. Данная проблема исследована в приложении к курсу «Защищенные информационные системы» по специальности магистратуры 10.04.01 «Информационная безопасность».
Ключевые слова: *узел шанса, информационный набор, политики контроля в системе защиты, типы атак на коммуникации.*

P. P. Nedoshivin*
Student
V. V. Komashinsky*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

GAME THEORY METHODS TO ENSURE THE SECURITY OF COMMUNICATION ON SWITCHED SYSTEMS

The existing approaches to the application of game theory methods for analyzing the behavior of an attacker and the operation of a protection system during an attack on autonomous switched systems, such as vehicles, are considered. This problem was investigated in the Appendix to the course «Secure Information Systems» for master's direction 10.04.01 «Information security».
Keywords: *node of chance, information set, control policies in the protection system, types of attacks on communications.*

Подключенные и автономные транспортные средства стали обширной и многообещающей областью исследований за последние два десятилетия. Как тесно связанная тема автомобильные взводы завоевывают свою репутацию за счет обеспечения комфорта вождения/пассажиров, повышения энергоэффективности, снижения загрязнения, а также увеличения пропускной способности. Концепция взвода включает группу транспортных средств, движущихся тесно связанными друг с другом из пункта отправления в пункт назначения как единое целое. Член взвода получает информацию о динамике других транспортных средств и о маневрах через сеть связи между транспортными средствами (V2V) для расчета команд управления соответствующим образом и поддержания устойчивости взвода, т. е. для поддержания небольшого расстояния между транспортными средствами и относительной скорости [1].

Однако такие реализации связи V2V также открывают новые векторы атак, которые увеличивают уязвимость системы безопасности и делают взводы транспортных средств привлекательной целью для киберфизических атак. Злоумышленники могут удаленно внедрить несколько фальсифицированных узлов транспортных средств во взвод, что позволяет им публиковать тщательно разработанные сообщения радиомаяка, чтобы получить привилегию на дорогу или вызвать заторы на дорогах и даже серьезные столкновения. Настоятельно необходимо устранить риски для безопасности, вызванные такими атаками на основе обмена данными.

Рассматривается иерархическая продольная структура управления с контроллером верхнего уровня и контроллером нижнего уровня, как показано на рис. 1. Контроллер верхнего уровня использует информацию о других транспорт-

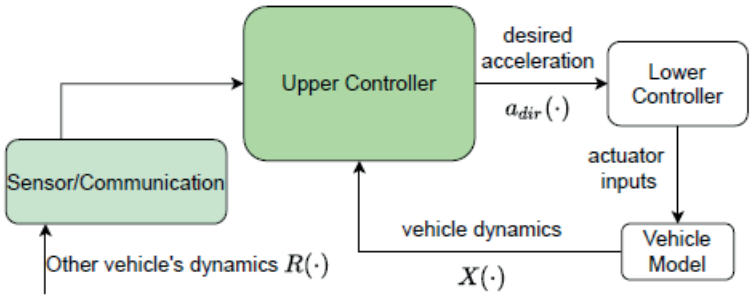


Рис. 1. Устройство системы продольного регулирования с точки зрения машины во взводе

ных средствах, полученную через датчики или связь, и внутреннюю динамику транспортного средства для вычисления желаемого ускорения a_{dir} для каждого автомобиля. Контроллер нижнего уровня генерирует входные сигналы исполнительного механизма (например, команды дроссельной заслонки и/или торможения) для отслеживания желаемого ускорения [2].

Более детально рассматривается взвод из $2 \le N < \infty$ ТС. Состояние i-го транспортного средства определяется как $\left[x_i(t), v_i(t)\right]^T$, где $x_i(t) \in \mathbb{R}$ и $v_i(t) \in \mathbb{R}$ – положение и скорость транспортного средства i. Следует сосредоточиться на контроллере верхнего уровня, в котором каждое транспортное средство следует динамике второго порядка:

$$\dot{x}_i(t) = v_i(t),$$
$$\dot{v}_i(t) = u_i(t),$$

где $u_i(t)$ – управляющий вход (ускорение) в систему. Для упрощения записи время t в дальнейшем опускается. Политики контроля, рассматриваемые в системе защиты, могут принимать любую из следующих двух форм.

1. Кооперативный адаптивный круиз-контроль (CACC)

Управляющий вход в CACC задается

$$u_i = \sum_{j \in \mathcal{N}_i} \alpha_{ij}\left(x_i - x_j + L_{ij}\right) + \sum_{j \in \mathcal{N}_i} \beta_{ij}\left(v_i - v_j\right) + \sum_{j \in \mathcal{N}_i} \gamma_{ij} a_j,$$

где набор N_i содержит транспортные средства, которые связываются с транспортным средством i (т. е. соседи транспортного средства i), а a_j – ускорение транспортного средства j. Здесь $\alpha_{ij} \in \mathbb{R}$, $\beta_{ij} \in \mathbb{R}$ и $\gamma_{ij} \in \mathbb{R}$ – коэффициенты усиления регулятора. Таким образом, i-е транспортное средство регулирует желаемое ускорение, чтобы координировать свою скорость с соседями и поддерживать свое относительное положение относительно желаемого (или целевого) расстояния между транспортными средствами L_{ij}. Выбирается желаемое расстояние $L_{ij} = L\Delta_{i,j}$, где $\Delta_{i,j} \in \mathbb{N}$ – количество транспортных средств (прыжков) между транспортными средствами i и j, а L постоянно и единообразно для всех машин взвода. Информация о динамике других транспортных средств, о положении, скорости и ускорении $\left(x_j, v_j, a_j\right)$ получена посредством беспроводной связи через сеть V2V, которая может быть реализована, например, в форме 5G или автомобильной специализированной сети [3].

2. Адаптивный круиз-контроль (ACC)

В этом контроллере управляющий вход представлен как

$$u_i = \sum_{j \in \mathcal{N}_i} \alpha_{ij}\left(x_i - x_j + L_{ij}\right) + \sum_{j \in \mathcal{N}_i} \beta_{ij}\left(v_i - v_j\right),$$

где набор $j \in \mathcal{N}_i$ содержит соседей транспортного средства i, которые обнаруживаются бортовыми датчиками дальности. Как уже указано, $\alpha_{ij} \in \mathbb{R}$ и $\beta_{ij} \in \mathbb{R}$ – коэффициенты усиления управления. Политика управления ACC использует только относительное положение и скорость в качестве обратной связи, чтобы генерировать желаемое ускорение для поддержания заданного расстояния между транспортными средствами L_{ij} и относительной скорости. По сравнению с CACC, информация о динамике других транспортных средств получается только с помощью сенсорных измерений, которые являются надежными, в отличие от сообщений связи [4].

В общем случае система обратной связи с двойным интегратором (1) для автомобиля i может быть представлена так:

$$\dot{z} = Az + BR,$$

где $z=\left[x_i(t),v_i(t)\right]^T$ – вектор состояния транспортного средства i (для простоты используются неправильные обозначения), а $R=\left[\left[x_j-L_{ij}\right]\left[v_j\right]\left[a_j\right]\right]^T$, $\forall j\in\mathcal{N}_i$ – внешний входной вектор, который состоит из динамики других транспортных средств, где $\left[.\right]$ представляет собой вектор-строку соответствующего размера. Матрица A будет иметь следующую форму (применяется политика управления CACC (2)):

$$A_{CACC}=\begin{pmatrix}0 & 1\\ k_1 & k_2\end{pmatrix}=\begin{pmatrix}0 & 1\\ \sum_{j\in\mathcal{N}_i}\alpha_{ij} & \sum_{j\in\mathcal{N}_i}\beta_{ij}\end{pmatrix},$$

$$A_{ACC}=\begin{pmatrix}0 & 1\\ k_3 & k_4\end{pmatrix}=\begin{pmatrix}0 & 1\\ \sum_{j\in\mathcal{N}_i}\alpha_{ij} & \sum_{j\in\mathcal{N}_i}\beta_{ij}\end{pmatrix},$$

где элементы k состоят из соответствующих комбинаций параметров α и β, как в (2) и (3). Аналогично матрица B принимает следующий вид:

$$B_{CACC}=\begin{pmatrix}\left[0\right] & \left[0\right] & \left[0\right]\\ \left[-\alpha_{ij}\right] & \left[-\beta_{ij}\right] & \left[\gamma_{ij}\right]\end{pmatrix},\forall j\in\mathcal{N}_i,$$

$$B_{ACC}=\begin{pmatrix}\left[0\right] & \left[0\right] & \left[0\right]\\ \left[-\alpha_{ij}\right] & \left[-\beta_{ij}\right] & \left[0\right]\end{pmatrix},\forall j\in\mathcal{N}_i,$$

где $\left[.\right]$ представляет собой вектор-строку подходящего размера.

Предположение. Чтобы упростить анализ, принимается конкретная настройка CACC, которая основана на топологии информационного потока, «следующим за предшественником-лидером». В частности, каждое транспортное средство получает передаваемую информацию о местоположении, скорости и ускорении только от командира взвода и движущегося впереди его транспортного средства, что эквивалентно означает $\mathcal{N}_i=\{1,i-1\}$ в (2). Предполагается, что каждое транспортное средство во взводе оснащено только радиолокационным датчиком в передней части транспортного средства, который измеряет положение и скорость своего предшественника (т. е. $\mathcal{N}_i=\{i-1\}$ в (3)). Предполагается, что командир взвода управляется водителем-человеком, на которого не влияют атаки связи или шум датчиков. Шум связи и шум датчиков игнорируются из-за их низкого влияния на безопасность системы по сравнению с преднамеренными атаками [5].

Рассматривается особый тип коммуникационной атаки, а именно атака фальсификации сообщения. Непрерывно отслеживая сеть связи, злоумышленник может изменять содержимое полученных сообщений и впоследствии вставлять их обратно в сеть. Наличие такого типа атаки может вызвать нестабильность во взводе транспортных средств или даже столкновения.

Пусть U представляет собой набор транспортных средств, подвергшихся атаке. Состояние затронутого транспортного средства изменяется как

$$\dot{x}_i(t)=v_i(t),$$
$$\dot{v}_i(t)=u_i(t)+\xi(t),i\in\mathcal{U},$$

где $\xi(t)$ – преднамеренные модификации передаваемых сообщений. В отличие от шума, который лишь умеренно снижает производительность системы, противник может нацеливаться на конкретных членов взвода и проводить скрытные, адаптивные и агрессивные атаки, что ставит под угрозу безопасность взвода [6].

Предположение 1. Кибератаки, связанные с коммуникациями, не изменяют физических свойств отдельных транспортных средств, что означает защиту целостности датчиков и контроллеров транспортных средств.

Предположение 1 проясняет внимание к атакам на основе коммуникации в этой статье. По сравнению с атаками на исполнительные механизмы и датчики, атаки с фальсификацией сообщений не изменяют напрямую целевые физические системы, а достигают злонамеренных результатов за счет модификации входных данных системы. Хотя контроллер ACC имеет аналогичную структуру обратной связи (т. е. матричную структуру) с CACC, у них есть разные механизмы для получения другой информации о динамике транспортного средства (через бортовые датчики). Невосприимчивость к атаке фальсификации сообщений и тот факт, что физическая система попрежнему надежна при такой атаке, означает, что контроллер ACC является подходящим дополнительным контроллером в такой враждебной среде. Однако напомним, что ACC не может гарантировать устойчивость строки, что ограничивает ее пригодность для длительного использования во взводе.

Учитывая преимущества и ограничения контроллеров ACC и CACC, предлагается использовать ACC в качестве вторичного контроллера, работающего в качестве резервного, когда сеть связи вызывает подозрения. Таким образом, преимущества обоих контроллеров будут сохранены, а эффекты, вызванные кибератаками, сведены к минимуму [7].

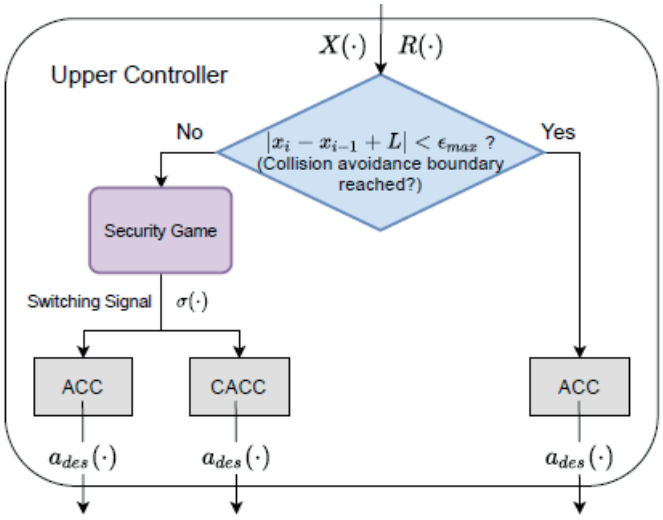


Рис. 2. Структура верхнего контроллера

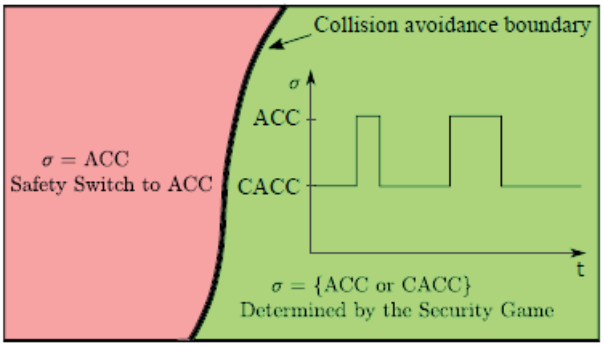


Рис. 3. Представление коммутируемой системы в пространстве состояний

Общая структура улучшенного контроллера верхнего уровня показана на рис. 2, который дает представление об обнаружении атак и смягчении их последствий. Оба контроллера образуют переключаемую систему, решение о переключении которой определяется новым теоретико-игровым анализом взаимодействий атакующего и защитника. Специальное ограничение состояния дополнительно повышает безопасность от ограниченных, но агрессивных модификаций сообщений, в которых решение ограничено, но нарушает практические ограничения ($\varepsilon_{\max}$), которые представляют собой транспортные средства, близкие к аварии. Результирующее пространство состояний визуализировано на рис. 3 [8].

Формально возмущение комбинированной атаки коммутируемой системы можно записать в матричной форме:

$$\dot{z} = A_p z + B_p R + M_p \xi, p \in \mathcal{P},$$
$$p = \begin{cases} ACC, & \text{если} \left| x_i - x_{i-1} + L \right| < \varepsilon_{\max} \\ \sigma(t), & \text{иначе} \end{cases}, \quad (8)$$

где P – индексный набор доступных подсистем, а $\mathcal{P} = \{CACC, ACC\}$, состоящий в данном случае из двух систем управления, $\sigma(t) : [0, \infty) \to \mathcal{P}$ представляет собой кусочно-непрерывную постоянную функцию, созданную на основе игры безопасности, которая определяет индекс активной системы в каждый момент времени, ξ – эффекты атаки, $\{(A_p, B_p, M_p)\}_{p \in \mathcal{P}}$ – набор троек матриц состояний с подходящими размерами для различных систем управления, а ограничение состояния $\left| x_i - x_{i-1} + L \right| < \varepsilon_{\max}$ представляет собой упомянутую поверхность переключения, которая определяет границу предотвращения столкновений на рис. 2 и 3. $M_{ACC} = 0$ на основании предположения 1.

Предлагается игра по защите кибербезопасности без взаимодействия, в которую играют Злоумышленник, детектор аномалий и Защитник, чтобы управлять процессом переключения контроллера в интерактивном режиме. Под атакующим и Защитником подразумевается соответственно Злоумышленник и устройство, которое генерирует сигналы переключения. Действия обоих игроков и отчеты об обнаружении представлены в виде ребер, а результирующие состояния – в виде узлов в дереве игры (рис. 4). Игра начинается с того, что Злоумышленник выбирает, атаковать взвод транспортных средств или нет, представлен крайними левыми ветвями. Если Злоумышленник решает атаковать, он выполняет атаку фальсификации сообщения, как моделируется в (7). Понятие «детектор» носит очень общий характер, чтобы убедиться, что игровая структура подходит для различных типов подходов к обнаружению аномалий, с пониманием того, что процесс проектирования для всех детекторов не может полностью предвидеть все сложные реальные ситуации и модели атак, ведущие к определенной вероятности ошибки [9].

Определение 1 (узел «Шанс»). Узел «Шанс» можно рассматривать как фиктивного игрока, который выполняет действия в соответствии с распределением вероятностей.

Используются случайные узлы для моделирования неопределенности результатов обнаружения, как показано на рис. 4. Должен существовать постоянный процесс обновления этих

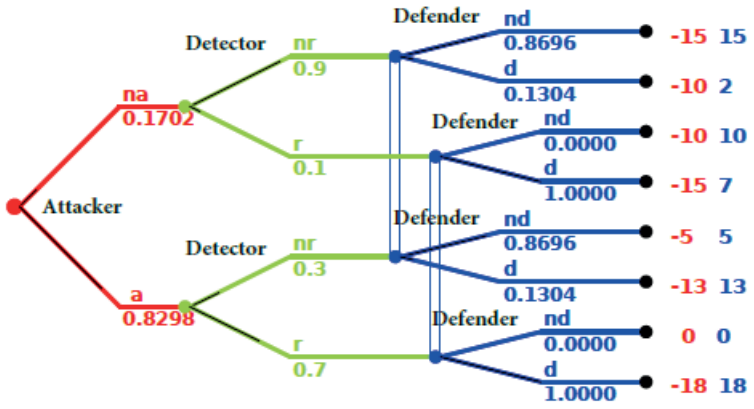


Рис. 4. Пример игровой модели: действия атакующего показаны слева, результаты обнаружения – по центру, решения о переключении контроллера защитника – справа

предварительных убеждений на основе результатов обнаружения в реальном развертывании.

Определение 2 (информационный набор). Информационный набор игрока i представляет собой набор узлов игрока i, которые i не может различить.

Как только Защитник получает результаты обнаружения, не ясно, была проведена фактическая атака или нет. Эта уникальная ситуация моделируется информационными наборами, показанными на рис. 4 пунктирными линиями. На основании отчетов об обнаружении для Защитника существует два набора информации: один указывает на атаку, а другой – на ее отсутствие. Это означает, что Защитник должен учитывать последствия как фактического нападения, так и его отсутствия, когда детектор сообщил об атаке [10].

Наконец, после рассмотрения рациональных действий Злоумышленника и вероятности ошибок обнаружения Защитник решает, понижать ли уровень контроллера CACC до контроллера ACC или оставаться с контроллером CACC, например в случае, когда об атаке сообщается с низкой, но ненулевой вероятностью. Схема реконфигурации двухрежимной системы управления использует решения Защитника в игре в качестве сигнала переключения σ переключаемой системы управления. Как показано зеленым цветом на рис. 3, решение игры активирует одну из подсистем (т. е. ACC или CACC), которая в свою очередь генерирует новые значения состояния для другой игры, прежде чем потребуется следующее переключение.

Формально игра моделируется с помощью:

– поля действия атакующего $\mathcal{A}^A := \{a:$ начать атаку, $na:$ не атаковать $\}$;

– узлов вероятности (детектор аномалий) $C := \{r:$ сообщить об атаке, $nr:$ не сообщать об атаке $\}$;

– области действия Защитника $\mathcal{A}^D := \{d:$ переключиться на ACC, $nd:$ переключиться на CACC $\}$.

Профиль стратегии моделируется как $\langle a,c,d \rangle$ для $a \in A$, $c \in C$ и $d \in D$. Значения полезности для атакующего и Защитника обозначаются как $\left[\left(R_1^A, R_1^D\right),\ldots,\left(R_8^A, R_8^D\right)\right]$, которые могут быть выбраны для отражения компромиссов и рисков безопасности конкретного взвода транспортных средств.

Один из примеров игры показан графически, как на рис. 4. Утилиты каждого профиля стратегии выделены красным для атакующего и синим для Защитника на листьях дерева игры. Пример детектора аномалий: в 90% случаев правильно сообщает неопасные данные при отсутствии атаки; он выдает ложные срабатывания в оставшиеся 10% времени. Когда атака действительно произошла, этот детектор правильно сообщает об этом в 70% случаев и пропускает оставшиеся 30% времени. Эти значения должны быть откалиброваны на основе развернутого детектора аномалий в различных параметрах настройки. Используется популярный решатель игр с открытым исходным кодом Gambit, чтобы найти равновесные решения по Нэшу. Численно вычисляется уникальная смешанная стратегия. Распределение вероятностей действий каждого игрока показано под краями фигурки. Например, злоумышленник с указанными утилитами будет атаковать с вероятностью 82,98%, и если детектор сообщает об отсутствии обнаруженной атаки, Защитник

все равно выберет переход на контроллер ACC на основе датчика с 13,04% вероятности отреагировать на это высокое намерение атаки и несовершенные результаты обнаружения [11].

Хотя связь V2V дает взводам транспортных средств повышенную стабильность работы, в результате высокий уровень связи и открытости может привлекать киберфизические атаки на основе связи. Следовательно, была исследована схема реконфигурации контроллера для смягчения последствий атаки и, таким образом, повышения безопасности системы в злонамеренной среде. Это первая попытка использовать игры безопасности для управления процессом переключения в контексте коммутируемых систем, где были исследованы взаимодействия между интеллектуальным злоумышленником и защитником, обладающим несовершенным детектором. Два общих контроллера CACC и ACC для взводов автономных транспортных средств были тщательно проанализированы. Поверхность скольжения, основанная на предварительном ограничении состояния, подчеркивает неадекватность общих определений устойчивости в этом контексте и дополнительно гарантирует безопасность системы. Получено ограничение на минимальное время задержки, чтобы гарантировать стабильность цепочки коммутируемой системы в благоприятных условиях. Хотя представленный подход эффективен для смягчения атак в краткосрочной перспективе, мы не можем гарантировать стабильность строки в состязательной среде, что, возможно, потребует новых схем управления, помимо CACC или ACC. Следовательно, интересная открытая проблема состоит в том, чтобы установить, может ли сигнал переключения достичь стабильности цепочки при атаках на основе обмена данными (например, найти вероятные гарантии стабильности цепочки на основе характеристик несовершенного детектора аномалий). Более того, дополнительные исследования моделирования могут подтвердить полезность и надежность, если принять во внимание низкоуровневые модели контроллера и транспортных средств [12].

Библиографический список

1. Securing connected & autonomous vehicles: Challenges posed by adversarial machine learning and the way forward / A. Qayyum, M. Usama, J. Qadir, A. Al-Fuqaha // IEEE Communications Surveys & Tutorials. 2020. Vol. 22, № 2. P. 998–1026.

2. Effects of colluding sybil nodes in message falsification attacks for vehicular platooning / F. Boeira, M. P. Barcellos, E. P. de Freitas et al. // 2017 IEEE Vehicular Networking Conference (VNC). P. 53–60.

3. *Sumra I. A., Hasbullah H. B., AbManan J.-l. B.* Attacks on security goals (confidentiality, integrity, availability) in vanet: a survey // Vehicular Ad-Hoc Networks for Smart Cities. 2015. P. 51–61.

4. Privacy in inter-vehicular networks: Why simple pseudonym change is not enough / B. Wiedersheim, Z. Ma, F. Kargl, P. Papadimitratos // 2010 Seventh international conference on wireless ondemand network systems and services (WONS). 2010. P. 176–183.

5. Distributed secure platoon control of connected vehicles subject to dos attack: theory and application / D. Zhang, Y.-P. Shen, S.-Q. Zhou et al. // IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020.

6. *Feng S., Ishii H.* Dynamic quantized leader-follower consensus under denial-of-service attacks // 59th IEEE Conference on Decision and Control (CDC). 2020. P. 488–493.

7. *Merco R., Biron Z. A., Pisu P.* Replay attack detection in a platoon of connected vehicles with cooperative adaptive cruise control // Annual American Control Conference (ACC). 2018. P. 5582–5587.

8. *Keijzer T., Ferrari R. M.* A sliding mode observer approach for attack detection and estimation in autonomous vehicle platoons using event triggered communication // 58th Conference on Decision and Control (CDC). 2019. P. 5742–5747.

9. *Dadras S., Dadras S., Winstead C.* Identification of the attacker in cyber-physical systems with an application to vehicular platooning in adversarial environment // Annual American Control Conference (ACC). 2018. P. 5560–5567.

10. Tree-based intelligent intrusion detection system in internet of vehicles / L. Yang, A. Moubayed, I. Hamieh, A. Shami // 2019 IEEE Global Communications Conference (GLOBECOM). P. 1–6.

11. *Alotibi F., Abdelhakim M.* Anomaly detection for cooperative adaptive cruise control in autonomous vehicles using statistical learning and kinematic model // IEEE Transactions on Intelligent Transportation Systems, 2020.

12. Strategic mitigation against wireless attacks on autonomous platoons / G. Sun, T. Alpcan, B. I. P. Rubinstein, S. Camtepe // Machine Learning and Knowledge Discovery in Databases: Applied Data Science Track, ECML-PKDD. 2021. P. 69–84.

УДК 004.7 **DOI: 10.31799/978-5-8088-1701-2-2022-2-274-276**

О. А. Нестеренков*
студент
А. М. Тюрликов*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ИДЕНТИФИКАЦИЯ И АУТЕНТИФИКАЦИЯ В СИСТЕМАХ СЛУЧАЙНОГО МНОЖЕСТВЕННОГО ДОСТУПА С БОЛЬШИМ ЧИСЛОМ УСТРОЙСТВ

Рассмотрена задача идентификации и аутентификации устройств в системах Интернета вещей, где количество передающих устройств огромно, а данные имеют небольшой размер. Последовательно изучается алгоритм аутентификации сообщений на базовой станции, предлагается система передачи, исследуется влияние использования возможности аутентификации на параметры передачи.
Ключевые слова: *случайный множественный доступ, АЛОХА.*

O. A. Nesterenkov*
Student
A. M. Turlikov*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

IDENTIFICATION AND AUTHENTICATION IN MASSIVE RANDOM ACCESS PROTOCOLS WITH A LARGE NUMBER OF DEVICES

In this article, we consider the problem of identification and authentication in Internet of Things systems with a large number of transmitting devices and packets of finite and relatively short length. We consistently describe the message's authentication algorithm, introduce 2 massive random access systems and find how authentication feature effects on the transmission parameters.
Keywords: *massive random access, ALOHA.*

Введение

В современном мире активно развивается концепция Интернета вещей, одна из задач которой – межсетевое взаимодействие устройств. Для ее решения необходимо рассмотреть дополнительные требования, накладываемые на систему связи: подключение огромного числа устройств к базовой станции, редкие передачи, небольшой размер передаваемых данных. Именно эти требования не позволяют использовать стандартные подходы к построению систем множественного доступа.

Сейчас активно разрабатываются новые системы передачи, одна из которых – система без идентификации источника [1]. Ее особенность – отсутствие поля адреса источника в передаваемом пакете данных. Такое ограничение появилось из-за сценариев с бесконечным количеством устройств ($M \to \infty$). Передача идентификационного поля размером $\sim \log_2 M$ вместе с пакетом конечной длины вступает в противоречие с требованием о пакетах небольшого размера.

Работа построена следующим образом: в первой части вводится модель системы без идентификации источника и на ее примере рассматривается алгоритм идентификации и аутентификации, предложенный в [2]. Во второй вводится модель передачи, основанная на алгоритме АЛОХА, и исследуется влияние предложенного алгоритма идентификации на параметры передачи.

Система без идентификации источника

Модель системы базируется на следующих допущениях [2].

Допущение 1. Все сообщения имеют одинаковую длину D. Все время передачи по каналу разбито на окна, длительность одного окна принята за единицу времени. В начале каждого окна базовая станция отправляет квитанцию b

(псевдослучайное число или последовательность) всем абонентам для синхронизации. Абоненты могут начать передачу только в начале окна.

Допущение 2. В начале каждого окна выбираются K пользователей из M, готовых к передаче. Абоненты формируют свои сообщения и отправляют в канал.

Допущение 3. Декодеру известно число пользователей K, передававших в текущем окне, и он всегда декодирует полученные сообщения без ошибок.

Алгоритм идентификации и аутентификации для данной системы выглядит следующим образом:

1) у каждого пользователя имеется уникальный секретный ключ k_i, который известен только ему и базовой станции;

2) все пользователи, выбранные для передачи, формируют message authentication code (MAC), основываясь на передаваемых данных d_i, секретном ключе k_i и квитанции b:

$$m_i = f(d_i, k_i, b);$$

3) MAC добавляется к передаваемому сообщению, и полученный пакет отправляется в канал;

4) после получения всех сообщений базовая станция разделяет их на пары $[d_i, m_i]$;

5) для каждого сообщения базовая станция генерирует MAC, используя переданные данные d_i, ключ k_j из множества ключей всех пользователей, $j \in \{1...M\}$, и квитанции b, и сравнивает его с m_i;

6) если количество совпадений строго равно 1, можно выдвинуть предположение, что данное сообщение было отправлено пользователем j;

7) если количество совпадений не равно 1, нельзя точно определить отправителя.

В работе [2] проведен анализ системы на предмет криптографических ошибок, однако если рассматривать данную модель передачи для сценариев Интернета вещей, то можно заметить следующий недостаток: в ней для декодера заранее установлено число одновременно передающих абонентов. Если сообщения от устройств будут поступать редко, то работа системы станет неэффективной из-за длительного ожидания определенного числа переданных пакетов.

Система на базе алгоритма АЛОХА

Модель базируется на следующих допущениях [3].

Допущение 1. Совпадает с допущением 1 модели системы без идентификации источника.

Допущение 2. В окне возможны три события:

– «Конфликт» – событие, когда в одном окне одновременно передают два или более абонента. Предполагается, что при наложении сигналов сообщения полностью искажаются и не могут быть правильно декодированы;

– «Успех» – событие, когда в одном окне передает один абонент. В этом случае полученное сообщение успешно декодируется;

– «Пусто» – событие, когда в одном окне ни один из абонентов не передавал.

Допущение 3. В канале отсутствуют шумы. Абоненты наблюдают выход канала и достоверно определяют, какое событие произошло в окне.

Допущение 4. В системе имеется M абонентов. На вход системы поступает пуассоновский входной поток с интенсивностью λ. Интенсивность входного потока у каждого абонента одинакова и равна λ / M.

Алгоритм аутентификации в данной системе будет использоваться тот же, что и в системе без идентификации источника.

Анализ системы

Введем термин «критическая интенсивность». Под ней будем понимать значение интенсивности входного потока, после превышения которого система перестает работать стабильно (количество сообщений, вышедших из системы, становится меньше количества сообщений, вошедших в нее).

Так как система рассматривается для сценария Интернета вещей с огромным числом абонентов, имитационное моделирование не представляется возможным из-за длительных вычислений. Поэтому для поиска размера MAC, при котором достигается наименьшая задержка для данной системы, при заданном числе абонентов был проведен следующий расчет.

1. Выбираются возможные размеры идентификационного поля. Для обеспечения качества передачи размер MAC не должен быть большим (при $L = D$ значение критической интенсивности уменьшается в 2 раза), поэтому их можно перебрать.

2. Для заданного числа абонентов M рассчитывается значение критической интенсивности:

$$\lambda_{\text{кр}_L} = \lambda_{\text{кр}} f(D, L, M),$$

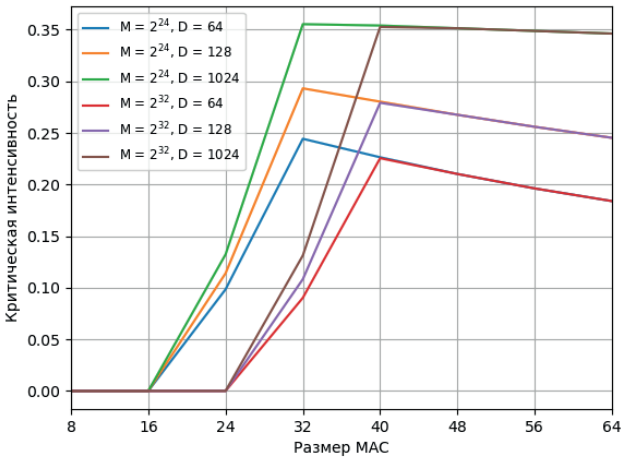


Рис. 7. График критической интенсивности для разных размеров MAC

где $\lambda_{кр}$ – значение критической интенсивности для системы без MAC.

На рис. 1 представлен график зависимости критической интенсивности от длины MAC. Можно заметить, что при $M = 2^{24}$ для достижения наименьшей средней задержки передачи в данной системе следует использовать значение MAC = 32 бит. Однако при использовании классического метода идентификации размер данного поля составил бы 24 бита. Можно сделать вывод, что использование предложенной модели в сценариях Интернета вещей для достижения наименьшей задержки может быть непозволительным при передаче даже с оптимальной длиной MAC.

Заключение

В работе был изучен алгоритм аутентификации и идентификации, предложенный для систем без идентификации источника. Также было проведено имитационное моделирование для системы, основанное на алгоритме АЛОХА, с использованием данного алгоритма и исследовано влияние возможности аутентификации на параметры передачи. Дальнейшим направлением исследований станет попытка отказаться от недостатков систем без идентификации источника и выявление способов более эффективного использования канала в данных системах.

Библиографический список

1. *Polyanskiy Y.* A perspective on massive random-access // IEEE Int. Symp. Inf. Theory (ISIT). 2017. June. P. 2523–2527.

2. How to Identify and Authenticate Users in Massive Unsourced Random Access / R. Kotaba, A. E. Kalør, P. Popovski [et al.] // CoRR. 2021. Vol. abs/2104.10576.

3. *Бурков А. А., Тюрликов А. М.* Основы построения инфокоммуникационных систем и сетей: лаборат. практикум. СПб.: ГУАП 2018. 50 с.

УДК 004.056.55 **DOI: 10.31799/978-5-8088-1701-2-2022-2-277-281**

А. А. Овчинников*
кандидат технических наук, доцент
А. М. Вересова*
аспирант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

ВЛИЯНИЕ УМЕНЬШЕНИЯ ИНФОРМАЦИОННОЙ ИЗБЫТОЧНОСТИ НА ПАРАМЕТРЫ КРИПТОСИСТЕМЫ НА ОСНОВЕ КОДОВ, ИСПРАВЛЯЮЩИХ ПАКЕТЫ ОШИБОК

При анализе и разработке криптосистем с открытым ключом большое внимание уделяется оптимизации размера ключей и стойкости. Однако информационная избыточность в таких системах также является важной характеристикой. В статье рассматривается вопрос оценки параметров системы, основанной на декодировании пакетов ошибок, при уменьшении избыточности.
Ключевые слова: *информационная избыточность, кодовые криптосистемы, низкоплотностные коды.*

A. A. Ovchinnikov*
PhD, Tech., Associate Professor
A. M. Veresova*
PhD Student
**St. Petersburg State University of Aerospace Instrumentation*

THE EFFECT OF REDUNDANCY REDUCING ON THE PARAMETERS OF THE CRYPTOSYSTEM BASED ON BURST-CORRECTING CODES

When analyzing and developing public-key cryptosystems, much attention is paid to optimizing the size of keys and the cryptocomplexity of the system. However, information redundancy in such systems is also an important characteristic. The issue of evaluating parameters of the system based on decoding error bursts while reducing redundancy is discussed.
Keywords: *redundancy, code-based cryptosystems, low-density parity-check codes.*

Криптосистемы с открытым ключом делятся на детерминированные и вероятностные. С точки зрения теоретической стойкости предпочтительны системы, использующие обращение к датчику случайных чисел. В частности, это может применяться при шифровании одинаковых сообщений, а также для противодействия некоторым разновидностям атак. При этом известно [1], что вероятностные схемы, поскольку отображают одно и тоже сообщение в различные шифртексты, не могут быть взаимно однозначной функцией без внесения в шифртекст информационной избыточности. Величина этой избыточности в разных криптосистемах может быть различной. Например, при использовании системы RSA [2] с 1024-битным модулем для шифрования 128-битного симметричного ключа размер шифртекста в 8 раз превышает размер исходного сообщения. Однако при использовании системы RSA для шифрования сообщений произвольной длины (превышающей размер ключа RSA) избыточность может быть сравнительно небольшой.

Наиболее распространенные асимметричные алгоритмы шифрования основаны на теоретико-числовых задачах, например таких как целочисленная факторизация в RSA. Подобные задачи не имеют известного полиномиального алгоритма решения, но не относятся к классу NP-трудных задач [3], что делает возможным нахождение таких алгоритмов в будущем. Для некоторых задач уже существуют полиномиальные квантовые алгоритмы решения, поэтому все более актуальной становится проблема построения систем, устойчивых к квантовым вычислениям. Среди множества трудных задач, на основе которых возможно построение постквантовых криптосистем, важное место занимает декодирование линейного кода [4].

Кодовые криптосистемы

Первая из кодовых криптосистем была предложена Р. Мак-Элисом в 1978 г. [5], через несколько лет после опубликования работы У. Диффи и М. Хеллмана о криптографии с открытым ключом [6]. Система Мак-Элиса основана на трудной задаче декодирования линейного кода без видимой структуры. Секретным ключом в этой системе выступают матрица $\mathbf{G}$ – порождающая $(k \times n)$-матрица линейного кода, исправляющего t ошибок – и случайные квадратные матрицы $\mathbf{S}$ и $\mathbf{P}$, а открытым – матрица $\mathbf{G}' = \mathbf{SGP}$, являющаяся порождающей матрицей линейного кода с такой же скоростью и минимальным расстоянием, как и код, задаваемый матрицей $\mathbf{G}$.

Шифрование выполняется путем наложения случайного вектора ошибок весом t на кодовое слово:

$$\mathbf{x} = \mathbf{mG}' + \mathbf{e} = \mathbf{c} + \mathbf{e}.$$

Для дешифрования необходимо вычислить

$$\mathbf{x}' = \mathbf{xP}^{-1} = (\mathbf{mSGP} + \mathbf{e}) \cdot \mathbf{P}^{-1} = \mathbf{mSG} + \mathbf{eP}^{-1},$$

а затем применить известную простую процедуру декодирования для получения $\mathbf{c}'$, из которого находится исходное сообщение $\mathbf{m} = \mathbf{c}' \cdot \mathbf{S}^{-1}$.

Таким образом, стойкость системы Мак-Элиса основана на трудности задачи исправления t ошибок в коде с неизвестной структурой. Классический вариант системы основан на коде Гоппы размером 524×1024, т. е. со скоростью 1/2. Система обладает высокой скоростью преобразований, так как используется простая двоичная арифметика, однако недостатком системы считается большая длина ключей. Кроме того, в системе Мак-Элиса на длинный вектор накладывается малое число ошибок: при классических параметрах в сообщении из 1024 бит искажаются лишь 50 символов. Информационная избыточность в системе вне зависимости от длины исходного сообщения составляет примерно половину шифртекста. Поэтому в ряде трудов анализировалось использование системы Мак-Элиса при работе со скоростью больше, чем 0,5 [7].

С учетом недостатков системы Мак-Элиса рассматриваются два основных направления развития кодовой криптографии. Первое связано с уменьшением размера открытого ключа в системе Мак-Элиса за счет использования специальных кодов Гоппы либо поиска альтернативных кодов. Пример такого подхода описан в работах М. Балди [7, 8], где используются квазициклические низкоплотностные коды (QC-LDPC). Второе направление связано с использованием более трудной математической задачи по сравнению с исправлением фиксированного количества ошибок.

Криптосистема с исправлением пакетов ошибок

В работе [9] была предложена концепция системы, основанной на задаче полного декодирования. Подход состоит в том, что маскирующему преобразованию подвергается не только порождающая матрица кода, но и множество исправляемых этим кодом ошибок. В [10] в качестве секретного ключа для такой системы предлагается использовать код, исправляющий пакеты ошибок, а в качестве маскирующего преобразования – матрицу, структура которой показана на рис. 1. Заштрихованная область матрицы обозначает позиции, содержащие случайные биты, а остальные позиции занимают нулевые элементы. Такая матрица задает отображение векторов, представляющих собой пакеты ошибок длиной x, в пакеты длиной b.

Данное преобразование позволяет рассмотреть следующий вариант криптосистемы. Выбирается матрица $\mathbf{G}$, задающая (n, k)-код, для которого известна эффективная процедура исправления единичных пакетов ошибок длиной b, составляющих множество E. Матрица $\mathbf{M}_1$, имеющая структуру, представленную на рис. 1, используется для отображения множества E пакетов ошибок длиной b во множество $\tilde{E}$ пакетов длиной x: для любого $\tilde{\mathbf{e}} \in \tilde{E}$ вектор $\tilde{\mathbf{e}}\mathbf{M}_1 = \mathbf{e} \in E$. Затем выбирается случайная невырожденная матрица $\mathbf{M}_2$ и вычисляется $\mathbf{M} = \mathbf{M}_1\mathbf{M}_2$ и $\mathbf{G}' = \mathbf{GM}_2$. Публичный ключ составляют матрицы $\mathbf{G}'$, $\mathbf{M}$ и параметр x в качестве описания $\tilde{E}$, а секретный – матрицы $\mathbf{G}$, $\mathbf{M}_1$ и $\mathbf{M}_2$.

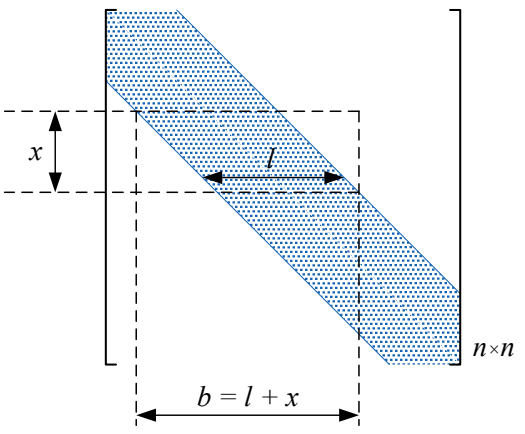


Рис. 1. Маскирующая матрица

Для шифрования сообщения $\mathbf{m}$ необходимо выбрать случайный вектор $\tilde{\mathbf{e}} \in \tilde{E}$ и вычислить $\mathbf{e}' = \tilde{\mathbf{e}}\mathbf{M}$, после чего вычислить шифртекст

$$\mathbf{x} = \mathbf{m}\mathbf{G}_\text{›} + \mathbf{e}'.$$

Для дешифрования находится $\mathbf{x}'$:

$$\mathbf{x}' = \mathbf{x}\mathbf{M}_2^{-1} = \left(\mathbf{m}\mathbf{G}\mathbf{M}_2 + \tilde{\mathbf{e}}\mathbf{M}_1\mathbf{M}_2\right)\mathbf{M}_2^{-1} = \mathbf{m}\mathbf{G} + \mathbf{e},$$

к которому затем применяется эффективная процедура декодирования для получения исходного сообщения $\mathbf{m}$.

Конкретный выбор кодов и алгоритмов декодирования не описан в статье [10], но можно выделить ключевые свойства, важные для системы:

– возможность исправления пакетов ошибок;

– вариативная скорость кода;

– компактное представление ключей.

Выбор кода

В [10] предлагается использовать низкоплотностные коды, так как они обладают хорошей корректирующей способностью при исправлении пакетов ошибок. Классические алгоритмы декодирования рассматривают пакеты как независимые ошибки, но из-за разреженности проверочной матрицы низкоплотностного кода пакет затрагивает малое количество проверочных символов, что позволяет успешно декодировать такие ошибки с некоторой вероятностью.

Кроме того, в работе [11] был предложен алгоритм для гарантированного исправления пакетов ошибок в пределах корректирующей способности блочно-перестановочных низкоплотностных кодов. Блочно-перестановочная конструкция состоит из $(m \times m)$-блоков, представляющих собой степени матрицы циклической перестановки или нулевые матрицы (в случае нерегулярных кодов). Такая конструкция может храниться и обрабатываться в виде базовой $(\gamma \times \rho)$-матрицы, состоящей из значений степеней перестановок блоков. Для регулярных блочно-перестановочных кодов длина пакета, который код может исправить, ограничена сверху размером блока m. Процедура для точного определения корректирующей способности кода основана на вычислении ранга некоторых подматриц и имеет полиномиальную сложность. Эксперименты показывают, что, когда размер блока m является простым числом, с высокой вероятностью корректирующая способность кода максимальна и равна $m - 1$. Для декодирования низкоплотностного кода, как при использовании классических алгоритмов, так и для алгоритма декодирования пакетов, необходимо, чтобы в проверочной матрице отсутствовали циклы длиной 4.

Параметры криптосистемы исходят из скорости 1/2. Однако в работе [7] показано, что для системы Мак-Элиса наилучшую стойкость обеспечивают коды со скоростью 2/3. Блочно-перестановочная конструкция также позволяет варьировать скорость генерируемого кода. В табл. 1 представлена вероятность того, что случайно сгенерированная проверочная матрица блочно-перестановочного кода со скоростью 2/3 без циклов длиной 4 будет обладать максимальной корректирующей способностью. Можно заметить, что вероятность успешной генерации близка к единице и растет с увеличением размеров блоков и их количества.

Таблица 1

Вероятность генерации матрицы H с максимальной корректирующей способностью

Размерность матрицы $\gamma \times \rho$	Размер блока m	Вероятность
3×9	31	0,958
	67	0,978
	127	0,980
4×12	31	1,000
	67	1,000
	127	1,000

Оценка параметров системы

Для проведения оценки параметров криптосистемы можно ориентироваться на уровень стойкости, равный 2^{128}. Для того чтобы описанная система обладала такой стойкостью, сложность перебора по следующим множествам должна быть не менее, чем 2^{128}:

– множество секретных матриц $\mathbf{H}$;

– множество E';

– множество $\tilde{E}$.

Перебор по множеству секретных матриц соответствует перебору по множеству базовых матриц размерностью $\gamma \times \rho$ с элементами из диапазона $\{0, \ldots, m - 1\}$. Число таких возможных матриц – $m^{\gamma\rho}$. Пусть базовая матрица имеет размерность 3×9. Из $m^{27} = 2^{128}$ соответствующий размер блока будет равен $m = 2^5 = 32$. Для большей вероятности получения кода с максимально возможной длиной исправляемого пакета выберем ближайшее простое число: $m = 31$. Для любых больших размерностей блока сложность перебора будет увеличиваться. Выбор размера блока должен согласовываться с длиной исправляемого пакета, которую требуется обеспе-

чить, а приведенная оценка может рассматриваться как нижняя граница для m.

Множество E_x составляют векторы $\mathbf{e}' = \tilde{\mathbf{e}}\mathbf{M}$, где $\mathbf{M} = \mathbf{M}_1\mathbf{M}_2$. Хотя матрица $\mathbf{M}_1$ имеет заданный вид (см. рис. 1), а вектор $\tilde{\mathbf{e}}$ является пакетом ошибок длиной x, матрица $\mathbf{M}_2$ – случайная, поэтому вектор $\mathbf{e}_x$ представляет собой случайный вектор с равномерным вероятностным распределением, т. е. с ожидаемым весом $n/2$. Это делает невозможным как перебор по таким векторам, так и взлом системы путем решения задачи декодирования: никакой код, в том числе код $\mathbf{G}_x$, не может исправлять векторы ошибок такого веса.

Множество $\tilde{E}$ представляет собой множество всех возможных векторов с пакетами ошибок длиной x. Можно считать, что количество векторов с фиксированным расположением пакета равно 2^x (хотя следует на позициях пакета выбирать подвектор веса, близкого к $x / 2$, и количество таких векторов будет меньше указанного). Пакет может быть расположен на любой из $n - x + 1$ позиций, тогда сложность перебора по $\tilde{E}$ задается соотношением

$$(n - x + 1)2^x = 2^{128}.$$

При использовании блока размером $m = 31$ длина исправляемого пакета $b = 30 = l + x$, где l – ширина диагонали в матрице $\mathbf{M}_1$ на рис. 1. При таких параметрах имеем $n = m\rho = 31 \cdot 9 = 279$ и $(280 - x)2^x = 2^{128}$, откуда $x \approx 120$, что невозможно при $m = 31$. При использовании блока с размером $m = 127$ ($b = 126$) при аналогичных вычислениях будут получены $n = 1143$, $x = 118$ и $l = 8$. Данные параметры соответствуют матрице $\mathbf{M}_1$ размерностью 1143×1143, содержащей диагональ шириной $l = 8$. Использование такой матрицы может быть небезопасным, поэтому стоит рассмотреть возможность расширения диагонали.

Пусть, например, $l = 50$. Тогда длина исправляемого секретным кодом пакета должна быть $b = 120 + 50 = 170$ и, следовательно, размер блока $m \geq 171$. Это задает следующие параметры системы:

$$\gamma = 3, \rho = 9, l = 50, b = 170, m = 171,$$

$$n = 1539, x = 120.$$

Тогда сложность перебора пакетов длиной x оценивается как $(1539 - 120)2^{120} = 2^{130,5}$, что соответствует требуемому уровню стойкости. Для хранения секретной матрицы $\mathbf{H}$ достаточно всего $3 \cdot 9 \cdot \lceil \log_2 171 \rceil = 216$ бит. При этом размер открытой матрицы $\mathbf{G}$ составляет 1026×1539.

Таким образом, стойкость рассмотренной системы зависит от сложности перебора по пакетам длиной x, т. е. по множеству $\tilde{E}$, а также от возможности атак на структуру матрицы $\mathbf{M}_1$, что регулируется значением l. Эти параметры определяют значения b и m, при которых перебор по матрицам $\mathbf{H}$ нереализуем.

В табл. 2 приведено сравнение параметров k и n, определяющих размер ключей при скорости кода 2 / 3, количества накладываемых ошибок t и стойкости для системы Мак-Элиса, системы Балди на основе QC-LDPC и предлагаемой системы на основе пакетов ошибок. Приведенные параметры показывают, что предлагаемая криптосистема позволяет достичь требуемый уровень стойкости с использованием гораздо меньшего размера кода, чем другие системы, несмотря на понижение стойкости при уменьшении информационной избыточности.

Таблица 2

Сравнение параметров кодовых криптосистем

Система	R	k	n	t	Стойкость	Атака
Мак-Элиса	1/2	524	1024	50	2^{53}	декодирование t ошибок
	2/3	684	1024	34	2^{54}	
		2752	4096	112	2^{178}	
Балди	2/3	8192	12288	27	2^{54}	декодирование t ошибок
		24576	36864	81	2^{135}	
На основе пакетов	1/2	704	1408	704	2^{135}	перебор по $\tilde{E}$
	2/3	1026	1539	1026	2^{130}	
		2056	3084	2056	2^{211}	

Заключение

В статье рассмотрен вопрос оценки параметров кодовой криптосистемы на основе декодирования пакетов ошибок при уменьшении информационной избыточности. Показано, что при уменьшении избыточности стойкость понижается, но для описанной системы требуемый уровень стойкости достигается при увеличении размера матрицы в 1,6 раза, что обеспечивает код меньшей длины, чем в альтернативных криптосистемах.

Библиографический список

1. *Mao W.* Modern cryptography: theory and practice. Upper Saddle River, NJ: Prentice Hall, 2004. 707 p.

2. *Menezes A. J., Van Oorschot P. C., Vanstone S. A.* Handbook of applied cryptography: CRC Press series on discrete mathematics and its applications. Boca Raton: CRC Press, 1997. 780 p.

3. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. 416 с.

4. *Berlekamp E., McEliece R., van Tilborg H.* On the inherent intractability of certain coding problems (Corresp.) // IEEE Transactions on Information Theory. 1978. Vol. 24, № 3. P. 384–386.

5. *McEliece R. J.* A Public-Key Cryptosystem Based On Algebraic Coding Theory // Deep Space Network Progress Report 42–44. 1978. Vol. 44. P. 114–116.

6. *Diffie W., Hellman M.* New directions in cryptography // IEEE Transactions on Information Theory. 1976. Vol. 22, № 6. P. 644–654.

7. *Baldi M.* QC-LDPC Code-Based Cryptography: Springer Briefs in Electrical and Computer Engineering. Cham: Springer Intern. Publ., 2014. 120 p.

8. *Baldi M., Bianchi M., Chiaraluce F.* Security and complexity of the McEliece cryptosystem based on QC-LDPC codes // IET Information Security. 2013. Vol. 7, № 3. P. 212–220.

9. *Krouk E.* A New Public-Key Cryptosystem // Proceedings of the Sixth Swedish-Russian International Workshop on Information Theory. Moelle, Sweden, 1993. P. 285–286.

10. *Krouk E., Ovchinnikov A.* Code-Based Public-Key Cryptosystem Based on Bursts-Correcting Codes // XIII Advanced International Conference on Telecommunications (AICT 2017), June 25–29. Venice, Italy, 2017. P. 93–95.

11. *Veresova A. M., Ovchinnikov A. A.* About One Algorithm for Correcting Bursts Using Block-Permutation LDPC-Codes // 2019 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). St. Petersburg, Russia, 2019. P. 1–4.

УДК 004.07 **DOI: 10.31799/978-5-8088-1701-2-2022-2-282-286**

Е. Д. Пойманова*
кандидат технических наук
П. С. Летуновская*
магистрант
Б. С. Шром*
магистрант
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

АРХИТЕКТУРА МНОГОУРОВНЕВОЙ СИСТЕМЫ ХРАНЕНИЯ ДАННЫХ

Произведен обзор правового аспекта Федерального закона от 6 июля 2016 г. № 374-ФЗ («закона Яровой») и его последствий для рынка отечественных систем хранения данных. Предложена архитектура многоуровневой системы хранения данных, включающая механизмы по распределению и миграции данных внутри системы.
Ключевые слова: *системы хранения данных, «закон Яровой», СОРМ.*

E. D. Poymanova*
PhD, Tech.
P. S. Letunovskaya*
Postgraduate Student
B. S. Shrom*
Postgraduate Student
**St. Petersburg State University of Aerospace Instrumentation*

ARCHITECTURE OF A MULTI-LEVEL DATA STORAGE SYSTEM

This article provides an overview of the legal aspect of Federal Law № 374 of July 6, 2016 (the «Yarovaya Law») and its implications for the market of domestic data storage systems. The architecture of a multi-level data storage system is proposed, which includes mechanisms for the distribution and migration of data within the system.
Keywords: *storage systems, the «Yarovaya Law», SORM.*

Введение

Сегодня Российская Федерация озабочена формированием собственной IT-инфраструктуры. Принимаются меры по поддержке отечественных производителей электроники и программного обеспечения. Одна из поддерживаемых отраслей – рынок отечественных систем хранения данных. Такие системы имеют в составе как программную, так и аппаратную части. Спрос на эти системы создается за счет принятия законов, обязывающих операторов данных использовать именно отечественную продукцию. Одновременно с этим отечественный рынок пока не в силах покрыть внутренние потребности.

В данной статье рассмотрена актуальность создания новых систем хранения данных, проведен анализ рынка отечественных систем хранения данных, предложена архитектура многоуровневой системы хранения данных и поставлены задачи, которые необходимо выполнить в процессе ее реализации.

Актуальность проблемы

Согласно вступившему в силу Федеральному закону от 6 июля 2016 г. № 374-ФЗ с поправками, вышедшими в 2019 г., операторы сотовой связи и компании интернет-провайдеров обязаны хранить данные своих абонентов шесть месяцев, включая телефонные переписки и переговоры. Также закон обязывает в течение трех лет хранить метаданные – информацию о том, кому, когда звонил или пересылал файлы пользователь [1]. Цель поправок, называемых «законом Яровой», – противодействие терроризму, экстремистской деятельности, усиление общественной безопасности.

Важнейшим нововведением стали обязательства для операторов связи и интернет-компаний хранить на территории России все передаваемые пользователями данные и передавать по запросу эту информацию органам власти. Кроме того, операторам связи в России предъявляются требования согласования пла-

на мероприятий по внедрению СОРМ (система технических средств для обеспечения функций оперативно-розыскных мероприятий). СОРМ – программно-аппаратный комплекс (ПАК), который предназначен для проведения оперативно-розыскных мероприятий (ОРМ) в российских сетях связи. Устанавливать СОРМ обязаны все операторы и интернет-провайдеры независимо от размера сети.

Основные нормативные документы, которые регламентируют СОРМ, – Федеральный закон «О связи» от 07.07.2003 г. № 126-ФЗ, приказы Минкомсвязи (приказ Министерства связи № 2339 от 9 августа 2000 г., приказ Министерства информационных технологий и связи Российской Федерации от 16.01.2008 г. № 6 «Об утверждении Требований к сетям электросвязи для проведения оперативно-розыскных мероприятий») и постановления правительства РФ (от 27 августа 2005 г. № 538, от 31 июля 2014 г. № 743). ФСБ использует комплекс СОРМ для борьбы с терроризмом, экономическими и другими преступлениями. Операторов, которые не устанавливают СОРМ, лишают лицензии. Основная задача СОРМ – обеспечение безопасности государства и его граждан, что достигается выборочным контролем прослушиваемой информации.

Есть несколько видов СОРМ. СОРМ-1 необходима для прослушивания телефонных разговоров. Регулируется приказами Минкомсвязи России от 19.11.2012 г. № 268 (фиксированная телефония) и от 12.12.2016 г. № 645 (сотовая связь). СОРМ-2 – система для слежения за российскими пользователями Интернета. Регулируется приказом Минкомсвязи России от 16.04.2014 г. № 83. СОРМ-3 используется для долгосрочного хранения и оперативного доступа к детальной информации об абонентах оператора или провайдера и оказанных услугах связи, причем не только в реальном времени, но и за определенный период (до трех лет). Для накопления таких объемов данных применяются большие системы хранения [2]. Регулируется приказом Минкомсвязи России от 29.10.2018 г. № 573.

После принятия «поправок Яровой» последовала реакция операторов сотовой связи, которые заявили, что у них нет серверов необходимой емкости для хранения информации такого объема. Разработкой и продажей сертифицированного отечественного СОРМ-2 занимаются девять организаций: VAS Experts, «МФИ Софт», «Норси-Транс», «Сигнатек», «ТехАргос», «Основа Лаб», «Сигналтек», «Специальные технологии», «Национальные технологии». Половина из перечисленных производителей, кроме «Сигналтек», Vas Experts, «Норси-Транс» и «Специальных технологий», входят группу компаний «Цитадель». СОРМ-3 на данный момент не сертифицирован ни у одной компании. Минкомсвязи осведомлено о том, что сертифицированного оборудования, необходимого для «закона Яровой», в России пока нет [3].

Очевидно, что внедрение таких систем, как СОРМ, требует от операторов связи значительных вложений. При этом согласно Постановлению Правительства «Об установлении запрета на допуск отдельных видов товаров, происходящих из иностранных государств, и внесении изменений в некоторые акты правительства», с 1 января 2021 г. стоимость иностранных комплектующих в отечественных системах хранения данных (СХД) не должна превышать 35% от общей стоимости системы [4]. Следовательно, необходимы новые отечественные разработки в сфере хранения и обработки больших объемов информации.

Перечисленные факты делают актуальным создание новых отечественных программных и аппаратных решений в области хранения данных. Очевидно, что наладить их производство за столь короткий промежуток времени невозможно, поэтому данная проблема еще долго будет сохранять актуальность.

Обзор решений

Требование постановления Правительства РФ № 1746 от 21.12.2019 г. «Об установлении запрета на допуск отдельных видов товаров, происходящих из иностранных государств…» о непревышении порога стоимости иностранных комплектующих в 25% от стоимости конечного изделия с 1 января 2022 г. сильно сокращает выбор оборудования и программного обеспечения сотовым операторам и интернет-провайдерам [4]. В настоящий момент производством систем хранения данных занимаются компании «Норси-Транс», «Национальные технологии» (принадлежит «Ростеху»), группа компаний «Цитадель», ООО «Орион» и пр. [5].

Один из предлагаемых продуктов – система хранения данных «Купол» от компании «Национальные технологии», который, однако, работает на процессоре производства компании IBM на открытой архитектуре OpenPOWER [6]. Как отмечалось, применение зарубежных разработок в составе СХД возможно, однако их стоимость не должна превышать 25% уже с 1 января 2022 г. В системах хранения данных наибольшая доля стоимости комплектующих при-

ходится на запоминающие, а не вычислительные ресурсы, при этом своего производства, например, твердотельных накопителей в России на данный момент нет [7].

Другие примеры отечественных систем: СХД «Яхонт» от компании «Норси-Транс», аппаратно-программный комплекс «ЭЛКОМ-НТ СОРМ/ИС» от группы компаний «РТК НТ», ИС СОРМ «Январь». Сравнение данных отечественных продуктов приведено в таблице.

Сравнение отечественных СХД

СХД	Объем хранимых данных	Масштабируемость	Применение нескольких типов носителей информации
Купол	От 1000 ТИБ до 18816 ТБ	Да	Нет
Яхонт	22,5–75,0 Тб	Да	Нет
ЭЛКОМ-НТ СОРМ/ИС	Нет данных	Нет данных	Нет данных
Январь	Нет данных	Да	Нет

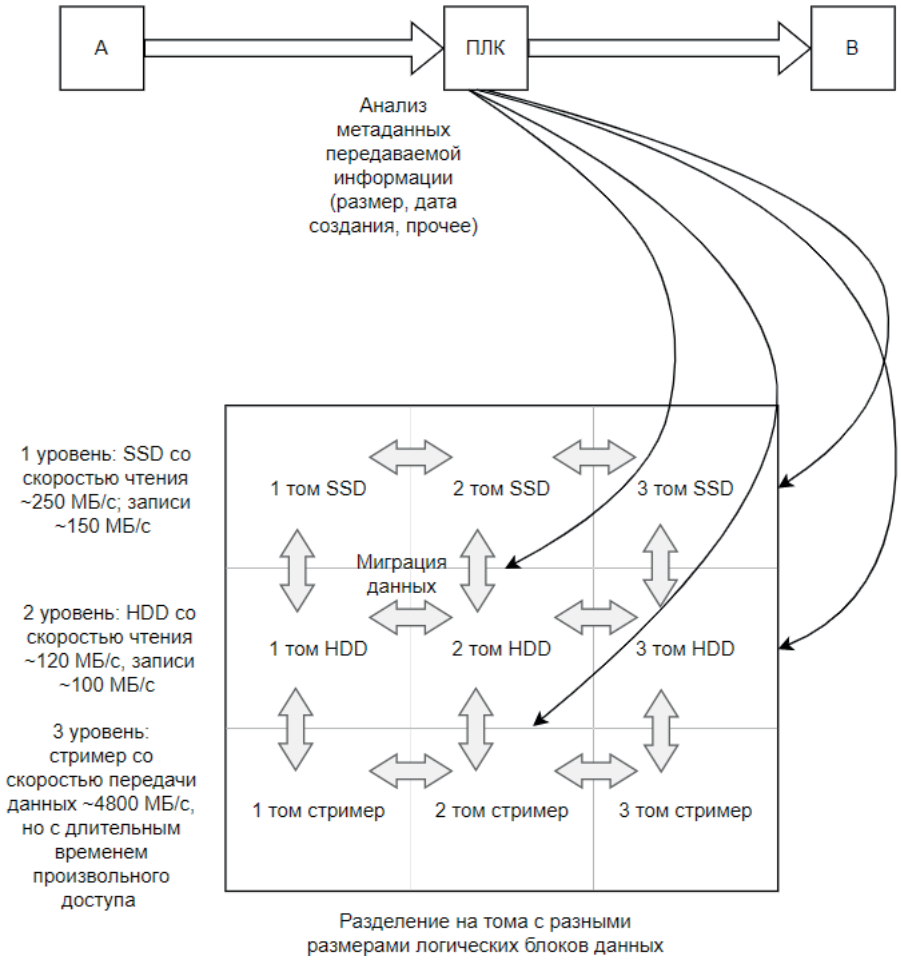
Как видно из таблицы, на данный момент ни в одной из рассмотренных СХД не используется многоуровневость: хранение данных на нескольких типах носителей. В качестве носителей используются HDD-накопители.

Схема предлагаемой СХД

Предлагается создание многоуровневой системы хранения данных, схема которой приведена на рисунке.

На рисунке изображен канал связи между узлами А и В. В этот канал связи встраивается программируемый логический контроллер (ПЛК). ПЛК производит анализ метаданных передаваемой между узлами информации и производит запись в нужный уровень и том СХД.

СХД состоит из нескольких уровней запоминающих устройств. На первом располагается запоминающее устройство с наибольшей скоростью чтения/записи, но с наибольшей стоимостью хранения на единицу информации, а на



Многоуровневая система хранения данных

последнем – устройство с наименьшей скоростью чтения/записи, но с наименьшей стоимостью хранения на единицу информации. Уровни СХД позволят увеличить скорость чтения/записи для актуальных данных, при этом незначительно увеличив соотношение стоимости на единицу хранимой информации, так как чем глубже уровень носителя, тем дешевле обходится хранение одной единицы информации и тем быстрее эту информацию можно будет получить из СХД в силу особенностей различных типов используемых накопителей.

При этом каждый уровень разбивается на тома с различными размерами логических блоков, что при определенных условиях должно положительно сказаться на скорости чтения и записи и обеспечить более полное заполнение носителя полезной информацией.

В предлагаемой схеме СХД предусмотрены два механизма управления потоками данных. Первый – распределение данных по уровням и томам при их первоначальной записи в СХД. Предполагается, что при первоначальной записи информации о ней будут собраны метаданные, в зависимости от которых будет приниматься решение о расположении информации в СХД. Второй механизм – миграция данных внутри СХД для возможности держать часто запрашиваемую информацию на уровнях с высокой скоростью чтения/записи и низким временем задержки к получению информации. Для получения информации с HDD сначала необходимо раскрутить жесткий диск, на что может потребоваться 2–5 с. Для получения информации со стримера сначала необходимо подвести к считывающей головке необходимый участок магнитной ленты, на что может потребоваться еще больше времени. Для получения информации с SSD-накопителя уже не требуется совершать каких-либо подготовительных действий – достаточно отправить на SSD сигнал с запросом, после чего микроконтроллер внутри SSD сразу сможет приступить к считыванию и передаче запрашиваемой информации. Кроме того, как показано на рисунке, носители различаются скоростью потоковой записи и потокового чтения, что также было учтено в предлагаемой схеме СХД.

Возможно реализовать систему для оценки заполняемости СХД и прогнозирования ее своевременного расширения на основе модели, представленной в статье [8].

Постановка задачи

К предложенной схеме СХД возникает ряд вопросов о деталях ее реализации.

1. Как будет происходить оценка поступающей в СХД информации при ее записи и миграции: с использованием алгоритмических методов или с использованием методов машинного обучения?

2. Файлы, поступающие на запись в СХД, имеют такие метаданные, как: размер, дата и время создания, дата и время поступления на запись, формат и пр. Какие метаданные учитывать для принятия решения о записи файла в конкретный уровень и том СХД?

3. Для обеспечения работоспособности системы необходимо продумать архитектуру СХД с точки зрения комплектующих. Нужно обеспечить такие интерфейсы связи нескольких носителей информации между собой, чтобы скорость передачи данных при миграции не ограничивалась пропускной способностью этих интерфейсов. Также необходимо выбрать архитектуру процессора, который будет принимать решения о распределении поступающей в СХД информации.

4. Для обеспечения надежности хранения данных в СХД применяют технологию RAID (Redundant Array of Independent Disks). Существуют стандартизированные модели размещения дисков в таких системах, называемые уровнями RAID. Однако технология RAID не предназначена для миграции данных внутри носителей физического хранилища данных. Следовательно, необходимо оценить возможность использования технологии RAID для предлагаемой схемы СХД.

5. Для построения прогноза заполняемости СХД можно выделить три разных подхода: статистические методы оценки, оценка при помощи машинного обучения и комбинированная оценка. Последний подход предполагает использование статистических моделей (например, модель ARIMA) с параметрами, вычисленными при помощи нейронных сетей и машинного обучения. Необходимо определить наиболее подходящий подход к составлению прогноза.

6. Возможна ситуация, когда объем трафика в единицу времени будет больше, чем скорость обработки и записи данных в СХД. Для записи данных без потерь необходимо предусмотреть буфер, в который будет поступать необработанная информация. Следовательно, встает задача о конфигурации такого буфера.

Заключение

В статье рассмотрены факторы, повысившие актуальность создания отечественных систем хранения данных. Проанализированы до-

ступные на рынке отечественные СХД. На основании этого была предложена модель многоуровневой системы хранения данных, а также механизмы по распределению и миграции данных внутри системы. Была поставлена задача по дальнейшим действиям в исследовании и разработке предложенной системы хранения данных.

Библиографический список

1. О внесении изменений в Федеральный закон «О противодействии терроризму» и отдельные законодательные акты Российской Федерации в части установления дополнительных мер противодействия терроризму и обеспечения общественной безопасности: Федер. закон от 06.07.2016 г. № 374-ФЗ. URL: http://www.consultant.ru/document/cons_doc_LAW_201078/ (дата обращения: 01.12.2021).

2. Как устроен СОРМ? URL: https://www.rspectr.com/articles/515/kak-ustroen-sorm. (дата обращения: 01.12.2021).

3. Достаточного количества оборудования для исполнения «закона Яровой» нет даже за рубежом, сообщили в МЭР. URL: https://www.newsru.com/russia/14jul2016/zakonyarovoy.html. (дата обращения: 01.12.2021).

4. Об установлении запрета на допуск отдельных видов товаров, происходящих из иностранных государств, и внесении изменений в некоторые акты Правительства Российской Федерации: Постановление Правительства РФ от 21 дек. 2019 г. № 1746. URL: https://base.garant.ru/73332186/ (дата обращения: 01.12.2021 г.)

5. 2018: Список производителей СХД для исполнения «закона Яровой». URL: https://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:%D0%A1%D0%A5%D0%94_(%D1%80%D1%8B%D0%BD%D0%BE%D0%BA_%D0%A0%D0%BE%D1%81%D1%81%D0%B8%D0%B8)#2018:_.D0.A1.D0.BF.D0.B8.D1.81.D0.BE.D0.BA_.D0.BF.D1.80.D0.BE.D0.B8.D0.B7.D0.B2.D0.BE.D0.B4.D0.B8.D1.82.D0.B5.D0.BB.D0.B5.D0.B9_.D0.A1.D0.A5.D0.94_.D0.B4.D0.BB.D1.8F_.D0.B8.D1.81.D0.BF.D0.BE.D0.BB.D0.BD.D0.B5.D0.BD.D0.B8.D1.8F_.C2.AB.D0.B7.D0.B0.D0.BA.D0.BE.D0.BD.D0.B0_.D0.AF.D1.80.D0.BE.D0.B2.D0.BE.D0.B9.C2.BB (дата обращения: 01.12.2021).

6. Система хранения данных Купол. URL: https://www.q-pol.ru/pdf/datasheet_Q_POL.pdf (дата обращения: 01.12.2021).

7. «ВымпелКом» объявил закрытый конкурс для исполнения «закона Яровой». URL: https://www.rbc.ru/technology_and_media/28/02/2020/5e57f01c9a7947d305f9ee77 (дата обращения: 01.12.2021).

8. *Пойманова Е. Д., Татарникова Т. М.* Модель прогноза дифференцированного наращивания емкости систем хранения данных // Волновая электроника и инфокоммуникационные системы: сб. ст. XXIII Междунар. науч. конф. СПб., 2020. С. 314–318.

УДК 004.07 **DOI: 10.31799/978-5-8088-1701-2-2022-2-287-290**

Е. Д. Пойманова*
кандидат технических наук
К. А. Хмелевский*
студент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

АВТОМАТИЗАЦИЯ РАСПРЕДЕЛЕНИЯ ФАЙЛОВ ПО НОСИТЕЛЯМ ДАННЫХ ПРИ ЗАПИСИ

Предложен механизм распределения данных внутри физического хранилища системы хранения при их непосредственной записи, основанный на принципе анализа метаданных.
Ключевые слова: *системы хранения данных, носители, метаданные, контроллер.*

E. D. Poymanova*
PhD, Tech.
K. A. Khmelevskiy*
Student
**St. Petersburg State University of Aerospace Instrumentation*

AUTOMATION OF THE FILES DISTRIBUTION ON DATA MEDIA DURING RECORDING

The article proposes a mechanism for distributing data within the physical storage of the storage system when they are directly recorded, based on the principle of metadata analysis.
Keywords: *storage systems, storage medium, metadata, controller.*

Введение

Общий рост количества информации способствует развитию сферы информационных технологий, в том числе технологий хранения и обработки информации. От специалистов в данной области требуется создание новых решений, позволяющих, в частности, эффективно управлять уже имеющимися ресурсами, такими как, например, носители в системах хранения данных.

Актуальность

В эпоху информатизации пользователи регулярно сталкиваются с проблемой такого способа хранения данных, который мог бы позволить рациональное использование пространства хранилища без потери скорости взаимодействия с ним [1, 2].

Решением проблемы являются системы хранения данных (СХД), которые используются в центрах обработки данных. Эти системы предназначены для безопасного и отказоустойчивого хранения обрабатываемых данных с возможностями быстрого восстановления доступа к данным в случае сбоя в работе системы. Важнейшие их характеристики – скорость операций чтения и записи, надежность и безопасность хранения информации. Учитывая особенности применяемого в разработке СХД оборудования – накопителей и контроллеров, имеется возможность проектировать гибкую систему для использования в рамках поставленной заказчиком задачи [2].

Носители данных, использующиеся в СХД, можно разделить на три группы в зависимости от используемого типа памяти.

1. **Магнитные диски** (HDD (англ. *Hard Disk Drive*)). Наиболее распространены для хранения информации за счет надежности, возможности восстановления данных, а также небольшой стоимости. Однако в современных условиях существенным недостатком становится то, что скорость передачи информации ограничивается скоростью вращения шпинделя [3]. Магнитные диски используются в RAID-массивах (англ. Redundant Array of Independent Disks «избыточный массив независимых дисков») – технологии, которая объединяет не-

Таблица 1

Сравнение носителей данных разных типов

Характеристика	Магнитная лента [5]	Жесткий диск [6, 7]	Флэш-накопитель [8, 9]
Максимальная скорость чтения (устойчивая, без сжатия данных)	до 400 Мб/с (Quantum LTO 9)	до 524 Мб/с (Seagate Exos 2X14)	до 7500 Мб/с (PNY XLR8 CS3140)
Максимальная скорость записи (устойчивая, без сжатия данных)	до 400 Мб/с (Quantum LTO 9)	до 524 Мб/с (Seagate Exos 2X14)	до 6850 Мб/с (PNY XLR8 CS3140)
Средняя скорость чтения и записи (устойчивая, без сжатия данных)	160 МБ/с	200 Мб/с	1500 Мб/с
Максимальный объем	45 Тб (в сжатом формате) (Quantum LTO 9)	20 Тб (WD Gold Enterprise Class SATA Hard Drive)	100 Тб (ExaDrive DC – 100 TB)

сколько дисков в логический элемент для повышения производительности [4].

2. **Магнитная лента** (англ. *Streamer*). Благодаря высокой плотности записи это один из самых дешевых способов хранения информации, если рассматривать стоимость 1 ТБ данных. Важное достоинство данного типа памяти – большая надежность по сравнению с хранением на диске. Существенный недостаток – низкая скорость произвольного доступа к данным из-за последовательного доступа. Хранение на магнитной ленте широко используется крупнейшими IT-корпорациями [3].

3. **Флэш** (SSD (англ. *Solid-State Drive*)). Такого рода накопители отличаются высокой скоростью работы. Для сравнения: скорость чтения и записи может превосходить обычный жесткий диск в несколько раз. Флэш-память более устойчива к физическим повреждениям. При этом в силу конструктивных особенностей успешное восстановление данных с выработавшего свой ресурс или поврежденного накопителя маловероятно: в одной плате впаяны контроллер и чипы памяти, соответственно при скачке напряжения пострадать может не один компонент, а все сразу [3].

В табл. 1 приведено сравнение типов носителей.

Цели и задачи исследования

Цель исследования – разработка хранилища данных, позволяющего эффективно хранить данные, что означает возможность экономно расходовать ресурсы хранилища при сохранении номинальной скорости чтения/записи каждого носителя.

Для достижения поставленной цели сформулированы следующие задачи.

1. Провести анализ работы существующих систем хранения данных, включая анализ принципов организации и работы современных СХД и видов носителей, которые в них используются.

2. Разработать способ распределения файлов с использованием внешних носителей и одноплатного компьютера, позволяющий на основе анализа метаданных файлов, поступающих в хранилище, распределять файлы среди носителей в соответствии с заданными параметрами. Для этого необходимо выполнить следующее.

2.1. Создать принципиальную схему работы СХД.

2.2. Использовать разбиение хранилища файлов на тома и логические уровни СХД.

2.3. Описать процесс получения метаданных файлов.

2.4. Использовать метаданные для принятия решения о записи файла на определенный носитель.

3. Поставить эксперимент, целью которого будет выявление экономии дискового пространства при помощи распределения данных в СХД на основе анализа метаданных файлов.

Схема работы хранилища данных

На рис. 1 представлена предлагаемая схема работы хранилища данных.

Хранилище является многоуровневым, каждый уровень разбит на тома в зависимости от размера хранимых файлов. Контроллер принимает решение о записи файла в определенный том хранилища на основе анализа его метаданных (анализируется размер и дата создания).

Правило распределения файлов по дате основывается на предположении, что чем раньше был создан файл, тем реже к нему будет обращаться пользователь:

1) наиболее частый доступ – файл создан меньше 3 мес. назад;

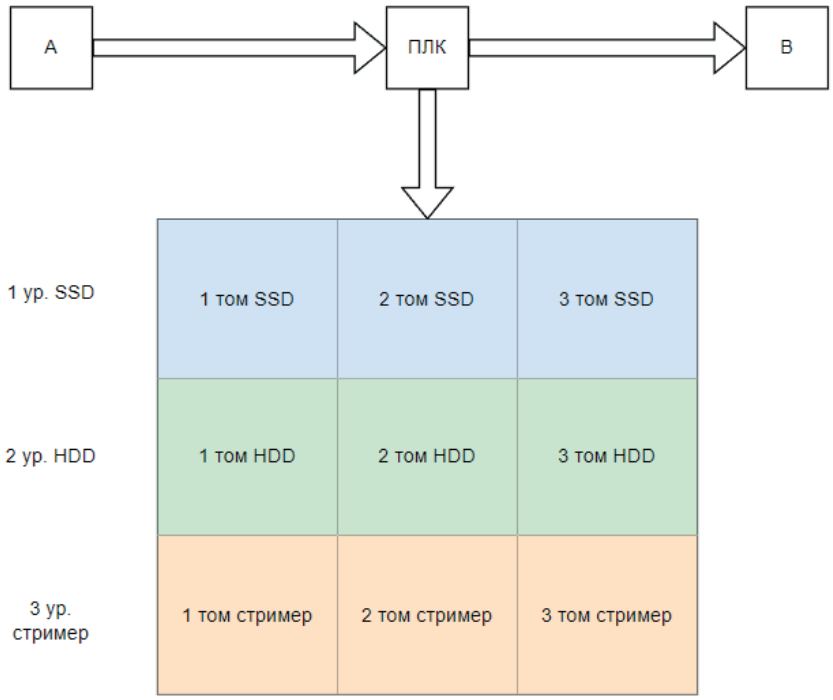


Рис. 1. Схема работы хранилища данных

2) стандартный доступ – файл создан меньше года назад;

3) наименьший доступ – год и более.

Всего используются три логических уровня.

– Первый, где в качестве накопителя используется SSD, предназначен для файлов, к которым предполагается наиболее частое обращение.

– Второй, где в качестве накопителя используется HDD, предназначен для файлов, к которым предполагается стандартное по частоте обращение.

– Третий, где в качестве накопителя используется Streamer, предназначен для файлов, к которым предполагается наименьшее по частоте обращение.

Каждый из уровней разделен на тома (логические разделы). Каждый раздел отведен под файлы определенного размера:

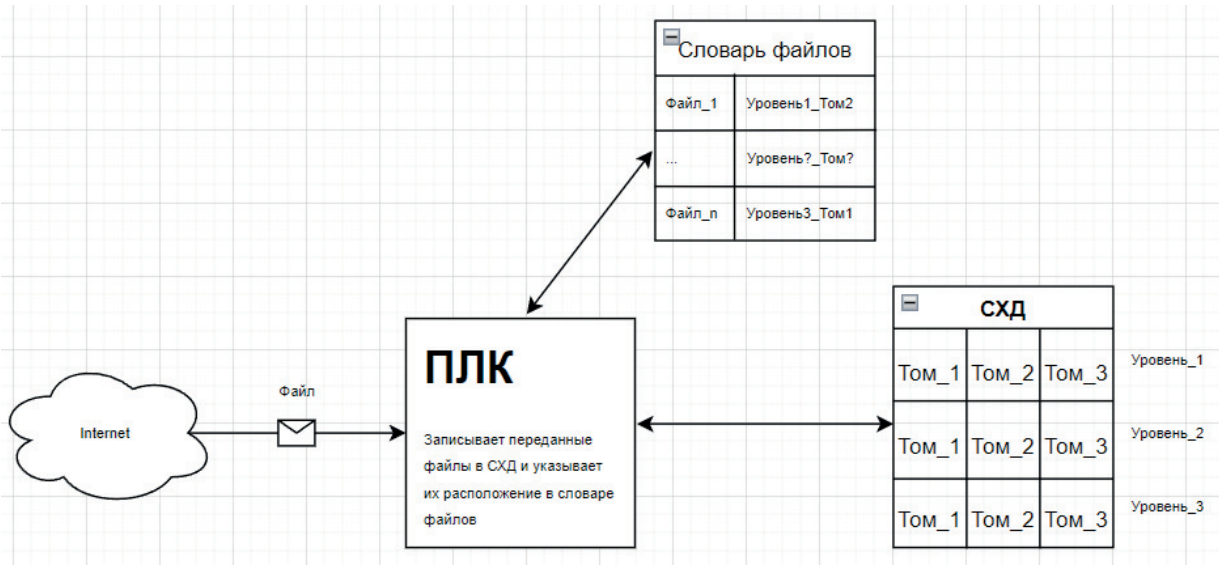


Рис. 2. Схема работы ПЛК со «словарем файлов»

– файлы большого размера ([размер_файла в Mb] >= 10 Mb) располагаются в томе № 1;

– файлы среднего размера (10Mb > [размер_ файла в Mb] >= 2Mb) располагаются в томе № 2;

– файлы наименьшего размера ([размер_ файла в Mb] < 2Mb) располагаются в томе № 3.

Распределение и анализ файлов в СХД

Распределение файлов по уровням и томам на основе анализа их метаданных входит в задачу контроллера. Реализовать работу СХД в целом предполагается с помощью языка программирования Python, так как для него существует множество библиотек для удобной работы с файловыми системами (например, pathlib или fuseрy).

Внутри одноплатного компьютера предполагается работа rest-api-сервера, который обеспечит передачу файлов от пользователя в СХД. На самом сервере будет располагаться сервис анализа метаданных файла.

Изначально при первичной отправке файла в СХД сервис определит уровень и том в зависимости от даты создания файла и его веса и запишет его в выбранное место. При этом сервис запоминает расположение и создает счетчик обращений, записывая это в специальный «словарь файлов» (рис. 2).

Для каждого тома будет изменен кластер в зависимости от предназначаемых для хранения по размеру в нем файлов, это поможет достичь дополнительной экономии памяти.

Заключение

Предлагаемый программно-аппаратный комплекс позволит решить спектр задач управления хранением данных. Одна из них – осуществление функции записи и первичного распределения файлов по уровням и томам в СХД. Это позволит обеспечить заданное время доступа к ячейкам физического хранилища (носителям) в такой многопользовательской системе, как система хранения данных.

Таким образом, совокупность указанных решений позволит управлять хранением данных с целью максимального использования имеющихся в СХД ресурсов.

Библиографический список

1. *Пойманова Е. Д.* Модели управления ресурсами систем хранения данных. URL: https://etu.ru/assets/files/nauka/dissertacii/2020/pojmanova/avtoreferat_pojmanova.pdf (дата обращения: 09.12.2021).

2. *Татарникова Т. М., Пойманова Е. Д.* Система управления хранилищем данных // Информационные системы и технологии в моделировании и управлении: сб. тр. V Междунар. науч.-практ. конф., Ялта, 20–22 мая 2020 г. / отв. ред. К. А. Маковейчук. Ялта: Ариал, 2020. С. 96–100.

3. *Скотт М.* Модернизация и ремонт ПК = Upgrading and Repairing PCs. 19-е изд. М.: Вильямс, 2011.

4. RAID Basics. Oracle. URL: docs.oracle.com/cd/E19487-01/817-3337-18/appa_raid_basic.html (дата обращения: 09.12.2021).

5. LTO-9. URL: https://www.quantum.com/en/products/tape-storage/lto-9/ (дата обращения: 09.12.2021).

6. Seagate Exos 2X14. URL: https://www.seagate.com/ru/ru/support/internal-hard-drives/enterprise-hard-drives/exos-2X14/#specs (дата обращения: 09.12.2021).

7. WD Gold Enterprise Class SATA Hard Drive. URL: https://www.westerndigital.com/ru-ru/products/internal-drives/wd-gold-sata-hdd#WD201KRYZ (дата обращения: 09.12.2021).

8. Обзор NVMe-накопителя PNY XLR8 CS3140: Phison E18 против Samsung 980 PRO и WD Black SN850. URL: https://3dnews.ru/1039156/obzor-pny-xlr8-cs3140 (дата обращения: 09.12.2021).

9. World's Highest Capacity and Most Efficient SSDs. URL: https://nimbusdata.com/products/exadrive/ (дата обращения: 09.12.2021).

УДК 004.62 **DOI: 10.31799/978-5-8088-1701-2-2022-2-291-293**

И. Д. Попов
ассистент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ОБ АВТОМАТИЧЕСКОМ ПРЕОБРАЗОВАНИИ МНОГОУРОВНЕВОЙ ДРЕВОВИДНОЙ СТРУКТУРЫ НАБОРА ДАННЫХ KYOTO 2006+ В БАЗУ ДАННЫХ

Приведено описание структуры набора данных Kyoto 2006+. Предложена программа на языке python для автоматического преобразования набора данных в базу данных. Приведены данные тестирования программы.
Ключевые слова: *обучение нейронной сети, набор данных, kyoto, база данных, python.*

I. D. Popov
Assistant
St. Petersburg State University of Aerospace Instrumentation

ABOUT AUTOMATIC CONVERSION OF KYOTO'S MULTILEVEL TREE STRUCTURE TO A DATABASE

A description of the structure of the Kyoto 2006+ dataset is provided. A python program for automatic transformation of a dataset into a database is proposed. The data of testing the program are given.
Keywords: *neural network training, dataset, kyoto, database, python.*

Введение

Для обучения искусственных нейронных сетей используются специальные наборы данных – датасеты. В области обнаружения сетевых атак существует, например, Kyoto 2006+ [1]. Этот датасет отличается от других достаточно популярных наборов, в частности, тем, что основан на реальных данных [2, 3].

Структура набора данных Kyoto 2006+, представленная на веб-сайте [4], неудобна для использования в качестве исходных данных, так как имеет многоуровневый древовидный вид, поэтому задачей становится преобразование имеющихся данных, например, в базу данных.

Описание структуры каталога набора данных Kyoto 2006+

Данные в датасете представлены за несколько лет, и на веб-сайте [4] имеется каталог в следующем виде (рис. 1):

– в корне каталога находится список годов, за которые имеются данные;

– у каждого года есть список месяцев, каждый из которых представляет собой архив файлов с выборкой;

– в некоторых месяцах указаны дни, которые отсутствуют в «датасете»;

– данные каждого дня расположены в отдельном файле в виде последовательности строк, формат которых описан в [1];

– загрузить можно (т. е. присутствует активная ссылка) конкретный год, месяц года, но в любом случае загружается архив.

Обрабатывать или загружать такие данные с помощью библиотек для работы с ней-

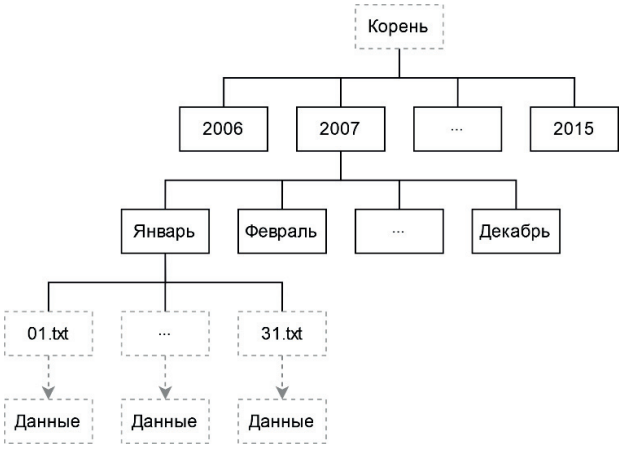


Рис. 1. Структура каталога набора данных Kyoto 2006+

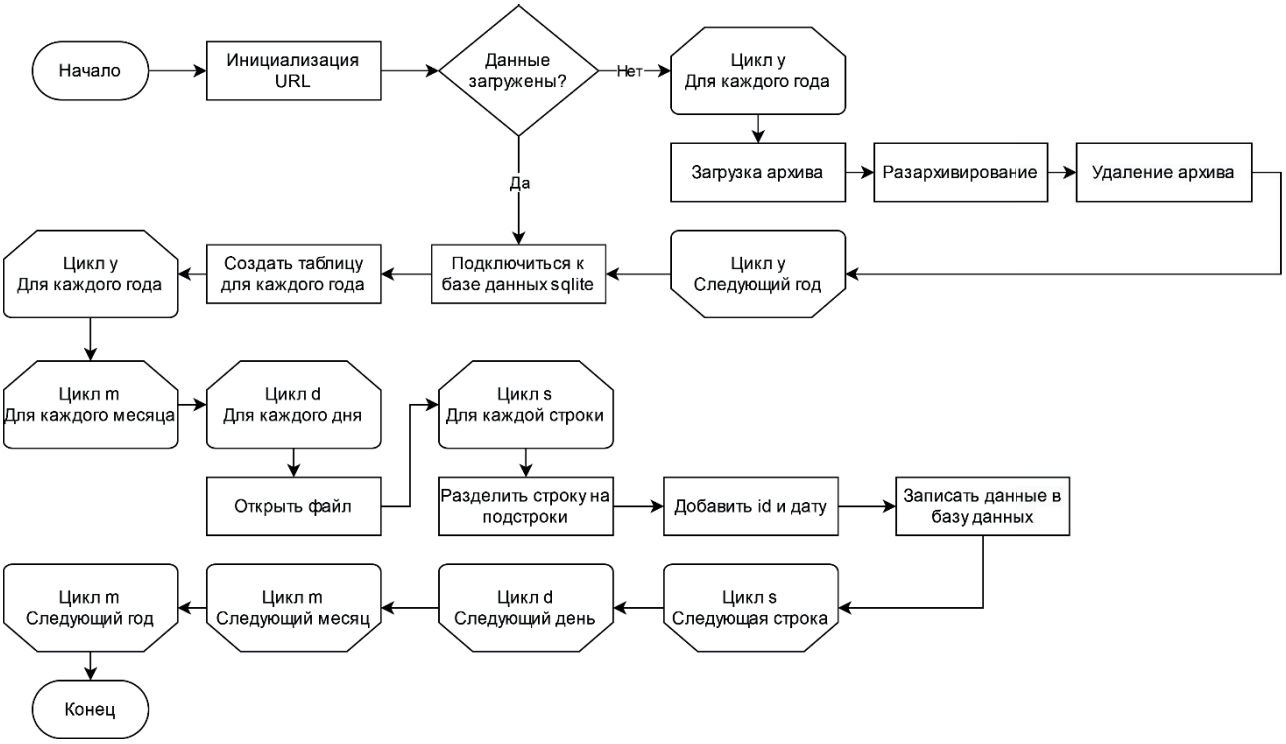


Рис. 2. Блок-схема алгоритма

ронными сетями неудобно из-за многоуровневой вложенности, поэтому имеет смысл объединить все данные в один большой файл или по крайней мере разделить на несколько файлов (например, по годам), а информацию о годе, месяце и дне включить в отдельное поле в файле.

В итоге все данные будут находится в одном файле, а данные о дате при необходимости можно извлечь из соответствующих полей.

Алгоритм загрузки данных и формирования базы данных

Алгоритм включает последовательность действий:

– загрузка архива (если папка с данными еще отсутствует, после разархивирования она появляется), его разархивирование и удаление архива;

– подключение к базе данных SQLite;

– для каждого года создается таблица в базе данных, если она еще не создана;

– для каждого месяца в каждом году считывается файл с данными за конкретный день, откуда эти данные отправляются базу данных, предварительно добавляется поле с датой, чтобы избавиться от иерархичной структуры.

Описание алгоритма приведено в виде блок-схемы на рис. 2.

Реализация алгоритма

Алгоритм реализован на языке программирования Python. Программа состоит из одного файла и последовательно выполняет описанные ранее действия. Исходный код программы представлен в [5], где также доступна его загрузка и приведено краткое руководство пользователя.

Примерный объем архивных данных, указанный на веб-сайте, составляет 20 Гб, а после разархивирования – около 100 Гб. При обычной скорости интернет-соединения в 100 Мб/с время загрузки и разархивирования данных составило около 1,5 ч. Время формирования базы данных из загруженных файлов – около 5 ч.

Тестирование программы производилось на современном оборудовании. При загрузке архивов производительность упиралась в скорость скачивания (точнее отдачу сервера), а при разархивировании – в скорость записи на HDD, поэтому необходимо выполнять данные процессы параллельно для ускорения.

Заключение

Результатом работы стала программа на языке Python, позволяющая загрузить набор данных, представленных в многоуровневой древовидной структуре, а также база данных

в виде одного файла, получаемая в результате работы данной программы. Базу данных удобнее использовать в сторонних программах или библиотеках (фреймворках). Для ускорения процесса необходимо реализовать многопоточную версию алгоритма (отдельные потоки для загрузки данных и записи в базу данных), так как основные ограничения – скорость интернет-соединения и скорость записи HDD.

Полученную программу несложно модифицировать для получения csv-файла (таблица), которые также часто применяются во многих библиотеках (фреймворках) для работы с нейронными сетями, в которых выбираются только необходимые поля (поля с датой, получаемые после работы программы, можно пропустить).

Библиографический список

1. Traffic Data from Kyoto University's Honeypots. URL: http://www.takakura.com/Kyoto_data/ (дата обращения: 12.11.2021).

2. KDD Cup 1999 Data // UCI KDD Archive. URL: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html (дата обращения: 12.11.2020).

3. NSL-KDD dataset // University of New Brunswick. URL: https://www.unb.ca/cic/datasets/nsl.html (дата обращения: 12.11.2020).

4. New version data. URL: http://www.takakura.com/Kyoto_data/new_data201704/ (дата обращения: 12.11.2021).

5. Kyoto 2006+ to SQLite // GitHub. URL: https://github.com/iIyaPopov/kyoto2006-sqlite (дата обращения: 03.12.2021).

УДК 004.056.055 **DOI: 10.31799/978-5-8088-1701-2-2022-2-294-297**

В. А. Рындюк
кандидат технических наук, доцент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ИСПОЛЬЗОВАНИЕ МЕТОДОВ НЕЧЕТКОЙ ЛОГИКИ В РЕШЕНИИ ЗАДАЧ ЗАЩИТЫ ИНФОРМАЦИИ

Разработана методика нечеткого оценивания класса защищенной информационной системы с учетом качественно определенных параметров. Проблема рассмотрена в приложении к изучению дисциплины «Основы информационной безопасности» для бакалавров направления 10.03.01 «Информационная безопасность».
Ключевые слова: *нечеткая логика, класс защищенной информационной системы, экспертные оценки, нечеткие переменные.*

V. A. Ryndyuk
PhD, Tech., Associate Professor
St. Petersburg State University of Aerospace Instrumentation

THE USE OF FUZZY LOGIC METHODS IN SOLVING INFORMATION SECURITY PROBLEMS

A methodology for fuzzy assessment of the class of a protected information system, taking into account qualitatively defined parameters, has been developed. The problem is considered in the Appendix to the study of the discipline «Information security basics» for bachelor's direction 10.03.01 «Information security».
Keywords: *fuzzy logic, class of a protected information system, expert assessments, fuzzy variables.*

Анализ современных законодательных актов РФ, в частности ФЗ № 149 «Об информации, информационных технологиях и о защите информации» от 27.07.2006 г., других нормативных документов по защите информации (ЗИ), а также нормативно-правовой базы по созданию защищенных автоматизированных систем, говорит о том, что в настоящее время требования к защите государственных информационных ресурсов неуклонно возрастают. Так как задачи защиты информации в автоматизированных системах напрямую связаны с влиянием человеческого фактора на процесс защиты, то считаем важным при проведении экспертиз и оценке требуемых параметров защиты учитывать не только количественно, но и качественно (нечетко) определенные параметры. При обращении с такими величинами обычно применяются методы теории нечетких множеств (НМ), оперирующие такими понятиями, как нечеткие множества, лингвистические переменные (ЛП) и др.

Анализируя современные стандарты информационной безопасности (ИБ) в плане защиты информационных систем (ИС), мы обнаружили, что в «Требованиях о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах» (приказ ФСТЭК России № 27 от 15.02.2017 г.) [1] присутствуют нечетко определенные параметры при определении класса защищаемой информационной системы. А это в процессе практического применения данного стандарта для реализации экспертиз приводит к необходимости разработки и использования соответствующих методов и методологий оценивания таких параметров.

Опираясь на современные теоретические разработки в области оценки качественных параметров при проведении экспертных оценок [2–5], считаем актуальным применение принципов нечеткой логики в практическом использовании современных стандартов РФ для защиты информации.

Наша цель – разработка метода нечеткой оценки экспертами класса защищаемой информационной системы [1] с учетом качественно определенных параметров. Для ее достижения необходимо решить несколько задач.

1. Анализ «Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информаци-

онных системах» (приказ ФСТЭК России № 27 от 15.02.2017 г.), далее по тексту – «Требования».

2. Выявление нечетко детерминированных параметров при определении класса защищаемой информационной системы.

3. Создание методики организации экспертной оценки класса ИС.

4. Подробное описание методики оценивания класса ИС.

При обеспечении защищенности информации, не составляющей государственную тайну, в государственных ИС необходимо определение класса ИС, поскольку он учитывается на всех стадиях ее жизненного цикла: формирования требований к ЗИ, содержащейся в ИС; разработки системы ЗИ (СЗИ) ИС; аттестации ИС по требованиям ЗИ; обеспечения ЗИ при работе аттестованной ИС и при выводе ее из эксплуатации. В зависимости от класса защищенности ИС составляют перечень специальных мер ЗИ, в том числе обеспечивающих необходимые уровни защиты ПДн.

Требование к классу защищенности включается в техническое задание на создание ИС и/или техническое задание (частное техническое задание) на создание СЗИ ИС.

В процессе определения класса ИС [1] учитывается значимость обрабатываемой в ней информации, а также масштаб ИС (федеральный, региональный, объектовый). Уровень значимости информации (УЗ) определяется степенью возможного ущерба для обладателя информации / заказчика и/или оператора от нарушения конфиденциальности (неправомерные доступ, копирование, предоставление или распространение), целостности (неправомерные уничтожение или модифицирование) или доступности (неправомерное блокирование) информации.

Экспертные оценки применяются при определении степени возможного ущерба от нарушения основных характеристик информационной безопасности.

На основе методов, используемых в теории и практике обеспечения информационной безопасности в ИС, а также в теории НМ, и на основе методологии [2] предлагается **методика оценки класса ИС.**

Она подразумевает наличие следующих этапов.

1. Выделение определенного свойства ИБ в зависимости от вида оцениваемой информации.

2. Выбор метода и на основании мнений одного или группы экспертов формирование нечетких эталонов лингвистических термов для каждого из оцениваемых аспектов выделенных свойств ИБ (определяют эталоны для обозначения степени ущерба от нарушения аспектов конфиденциальности (4 шт.), целостности (2 шт.) и доступности (1 шт.)).

3. Выбор экспертами (одним или группой) метода построения нечетких переменных (НП) для оцениваемых аспектов.

4. Сравнение полученных значений с эталонными и на основании этого определение степени возможного ущерба для информации.

5. Определение уровня значимости информации.

6. Установление класса ИС на основе полученных результатов.

Рассмотрим вопросы организации экспертиз на примере оценки класса ИС. Введем необходимые обозначения.

Приведем алгоритм оценки класса ИС, используя обозначения:

– y – показатель оцениваемого вида информации, $y = \{1, 2, 3, \ldots n\}$. Тогда при $y = 1$ оценке подлежит служебная тайна, при $y = 2$ – коммерческая, при $y = 3$ – нотариальная и т. п. в зависимости от оцениваемого предприятия. Принимая во внимание, что экспертизе будут повергаться различные виды информации (служебная, нотариальная тайна и др.), логично предположить, что от этого будет зависеть количество оцениваемых аспектов свойств ИБ;

– i – показатель оцениваемого свойства ИБ, $i = \{1, 2, 3\}$. Тогда при $i = 1$ оценке подлежит конфиденцияльность, при $i = 2$ – целостность, при $i = 3$ – доступность информации;

– j – количество нечетких оцениваемых аспектов для i-го свойства ИБ информации: $j = \overline{1,J}$, где $J = 4$ – максимальное количество нечетких аспектов;

– m_j – количество элементов оцениваемых аспектов, задается экспертами.

Согласно [1], уровень значимости информации (УЗ) определяется степенью возможного ущерба СтУ(i, j) для обладателя информации/заказчика и/или оператора.

Оцениваются следующие аспекты.

1. СтУ(1) – степень возможного ущерба от нарушения конфиденциальности (неправомерный доступ – СтУ(1, 1), неправомерное копирование – СтУ(1, 2), неправомерное предоставление – СтУ(1, 3) или распространение информации – СтУ(1, 4)).

2. СтУ(2) – степень возможного ущерба от нарушения целостности (неправомерные уничтожение – СтУ(2, 1) либо модифицирование – СтУ(2, 2)).

3. СтУ(3) – от нарушения доступности информации (неправомерное блокирование – СтУ(3, 1)).

Согласно [1], возможный ущерб может быть высоким, средним или низким.

СтУ принимает значение высокой, если в результате нарушения хотя бы одного из свойств безопасности информации возможны существенные негативные последствия в какой-либо (социальной, политической, международной, экономической, финансовой и др.) области деятельности и/или ИС и/или оператор не могут выполнять все свои функции.

СтУ будет средней, если в результате нарушения одного из свойств ИБ возможны умеренные негативные последствия в какой-либо области деятельности и/или ИС и/или оператор не могут выполнять хотя бы одну из своих функций.

СтУ принимает значение низкой, если в результате нарушения одного из свойств безопасности информации возможны незначительные негативные последствия в какой-либо области деятельности и/или ИС и/или оператор могут выполнять возложенные свои функции, но недостаточно эффективно или только с привлечением дополнительных сил и средств.

Значение СтУ позволяет потом определить УЗ информации (который может принимать значения: высокий – УЗ 1, средний –УЗ 2 и низкий –УЗ 3) и в итоге – класс защищенности ИС (К1, К2 или К3).

Подробно опишем методику оценивания.

1. Определение вида оцениваемой информации ограниченного доступа. Причем, согласно [1], в ИС могут обрабатываться два и более видов информации. УЗ определятся для каждого из них отдельно.

Пусть всех видов информации N. Для оценки, в зависимости от ИС, эксперту предоставляется $n \le N$ видов информации.

2. Определение для каждого выбранного i-го свойства ИБ максимального количества элементов $k_{ij\max}$ для каждого из j оцениваемых аспектов свойств ИБ СтУ(i, j).

В случае нечеткой оценки СтУ нарушения конфиденциальности количество нечетких оценок аспектов равно 4, а $i = 1$. Здесь и далее под нечеткими оценками СтУ будем понимать рызмытые, нечетко определенные оценки СтУ, относительно которых при оценках эксперта используют методы НМ.

3. Определение интервалов $[\underline{k}_{ijm}, \overline{k}_{ijm}]$ для каждого из аспектов СтУ$(1,j)$, т. е. интервалов, задающих количество m элементов, которые, по мнению эксперта, должны оцениваться для каждого из j аспектов.

4. Формирование лингвистических термов. Например, для определения лингвистических переменных «неправомерный доступ», «неправомерное копирование», «неправомерное предоставление», «неправомерное распространение информации» нужно для каждого из СтУ $i,j.m$ оцениваемых аспектов обозначить базовые терм-множества $T_{jm} = \{T_s\}$ $(s = \overline{1,L}$, где L – количество термов-эталонов). Терм-множества каждой из этих ЛП, например, при $L = 3$, можно задать термами с названиями: низкий (Н), средний (С) и высокий (В), которые представляют собой названия числовых НП. После этого нужно построить их функции принадлежности [6], заранее определившись с их видом и способом формирования.

5. Формирование экспертом нечетких переменных соответствующих видов.

6. Сравнение эталонных значений и результатов экспертной оценки, получение оценочных коэффициентов от сравнения [7]. Определение степени ущерба СтУ.

7. На основе полученных данных степени ущерба определяем уровень значимости информации и далее с учетом известного масштаба системы (федеральный, региональный или объектовый) – класс защищенности ИС.

Таким образом, разработана методика определения класса защищенной информационной системы с учетом качественно определенных параметров. Методика может иметь практическое применение в экспертных оценках, что повысит качество оценивания автоматизированных ИС.

Библиографический список

1. Об утверждении Требований о защите информации, не составляющей государственную тайну, содержащейся в государственных информационных системах: приказ Федер. службы по техн. и экспорт. контролю от 11 февр. 2013 г. № 17. URL: http://base.garant.ru/70391358 (дата обращения: 20.11.2019).

2. *Корченко А. Г., Рындюк В. А.* Экспертиза ТЗИ для программно-управляемых АТС на основе нечетких множеств // АВИА-2003: матер. V Междунар. науч.-техн. конф. Т. 2. Киев, 2003. С. 22.45–22.49.

3. *Корченко А. Г.* Построение систем защит информации на нечетких множествах. Теория и практические решения. Киев: МК-Пресс, 2006. 320 с.

4. *Рындюк В. А.* Нечеткая логика и защита информации // Университетские чтения-2009, посвящ. 70-летию ПГЛУ, 12–13 янв. 2009 г. Пятигорск, 2009. С. 268–273.

5. Метод n-кратного инкрементирования числа термов лингвистических переменных в задачах анализа и оценивания рисков / А. Г. Корченко, Б. С. Ах-

метов, С. В. Казмирчук, М. Н. Жекамбаева // Безопасность информации. 2015. Т. 21, № 2. С. 191–200.

6. *Корченко А. Г., Рындюк В. А.* Исследование методов формирования функций принадлежности на основе количественных парных сравнений // Захист інформації. 2003. № 3. С. 10–17.

7. Исследование методов сравнения нечетких чисел / В. В. Душеба, В. Г. Потапов, А. Г. Корченко, В. А. Рындюк // Збірник наук. праць Інституту проблем моделювання в енергетиці. Вип. 20. Львів: ПП «Системи, технології, інформаційні послуги». 2003. С. 12–21.

УДК 621.391 **DOI: 10.31799/978-5-8088-1701-2-2022-2-298-300**

Ф. А. Таубин*
доктор технических наук, профессор
А. Н. Трофимов*
кандидат технических наук, доцент
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

КАСКАДНОЕ КОДИРОВАНИЕ С ВНУТРЕННИМ МНОГОУРОВНЕВЫМ КОДОМ ДЛЯ FLASH-ПАМЯТИ – ОБЩАЯ СХЕМА

Рассматривается общий подход к построению каскадной схемы кодирования для NAND-flash-памяти на основе кодов Рида – Соломона и многоуровневого кода на основе суперпозиции произвольных компонентных двоичных кодов.
Ключевые слова: *flash-память, каскадные конструкции, код Рида – Соломона, целочисленные многомерные решетки, многоуровневый код.*

F. A. Taubin*
Dr. Sc., Tech., Professor
A. N. Trofimov*
PhD, Tech., Associate Professor
**St. Petersburg State University of Aerospace Instrumentation*

CONCATENATED CODING WITH INNER MULTIULEVEL CODE FOR FLASH MEMORY – GENERAL SCHEME

We consider a general approach to design a concatenated coding scheme for NAND flash memory based on Reed-Solomon codes and a multilevel code based on superposition of arbitrary binary component codes.
Keywords: *flash memory, concatenated constructions, Reed-Solomon code, multidimensional integer lattices, multilevel code.*

Введение

NAND-flash-память – в настоящее время доминирующий вид энергонезависимой памяти благодаря постоянно растущей плотности хранения, высокоскоростному доступу к данным и неуклонному снижению стоимости [1]. Сегодня доступны чипы памяти, позволяющие хранить до 4 бит в ячейке памяти [1, 2], однако стандартом де-факто, как правило, является flash-память с четырьмя ранжированными по уровню состояниями, что позволяет хранить 2 бита в ячейке. Вместе с тем повышение плотности хранения данных сопровождается снижением количества электронов (величины заряда) в плавающем затворе ячейки памяти и усилением влияния негативных факторов, искажающих значения уровней заряда в процессе записи/считывания. В результате характеристики основных параметров многоуровневой flash-памяти – вероятность ошибки, выносливость и долговечность хранения – часто оказываются неприемлемыми (как по отдельности, так и в совокупности). Например, исходная вероятность ошибки в четырехуровневой flash-памяти даже при умеренных значениях числа циклов перезаписи и времени хранения составляет порядка $10^{-4}...10^{-3}$ и более [3], тогда как требуемая вероятность ошибки лежит в диапазоне $10^{-12}...10^{-16}$. Стандартным решением, позволяющим улучшить потребительские характеристики многоуровневой flash-памяти, является введение помехоустойчивого кодирования [4, 5].

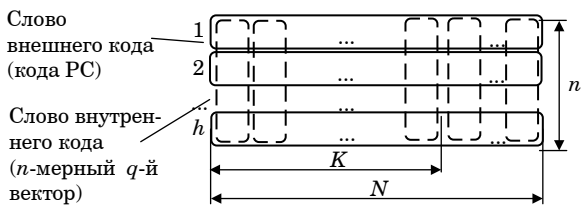
Один из возможных эффективных подходов к организации введения помехоустойчивого кодирования связан с использованием каскадных конструкций [6–8]. В работе [8] был предложен и проанализирован ряд каскадных конструкций для многоуровневой NAND-flash-памяти, базирующихся на использовании конечных подмножеств целочисленных многомерных решеток в качестве внутреннего кода и кодов Рида – Соломона во внешней ступени каскадной схемы. Расширение перечня приемлемых конструкций внутреннего кода возможно посредством построения внутреннего кода как

многоуровневого кода на основе суперпозиции произвольных компонентных двоичных кодов. На основе этого подхода в данной работе представлен перечень новых каскадных конструкций, полученных исходя из рационального сочетания дистанционных характеристик, плотности записи и сложности декодирования.

Схема каскадного кодирования

В рассматриваемой схеме каскадного кодирования в качестве внешнего кода выбран расширенный (удлиненный на один символ) (N, K) код Рида – Соломона над полем F_{2^s}, $2^s = N$. Внутренний код, обозначим его B_0, представляет собой множество последовательностей, состоящих из n символов над алфавитом $A = \{0,1,\ldots,q-1\}$, где q – число уровней записи ячейки flash-памяти. Элементы алфавита A связаны с входными уровнями (уровнями записи) ячейки памяти посредством взаимно однозначного отображения I множества A на множество $\{x_0, x_1,\ldots, x_{q-1}\}$ вида $I(i) = x_i$, $0 \le i \le q-1$. Иными словами, алфавит A представляет собой множество индексов уровней записи q. Будем полагать далее, что $m = \log_2 q$ целое число, т. е. ячейка памяти может хранить m бит данных. Для исключения пакетирования ошибок на внешней ступени декодирования кодовые символы внешнего кода (при необходимости) подвергаются блоковому перемежению (рисунок), а именно: h последовательных слов кода Рида – Соломона записываются в прямоугольную таблицу, содержащую h строк, после чего последовательно считываются по столбцам; каждый столбец этой таблицы отображается в один символ внутреннего кода.

Параметр h определяется объемом внутреннего кода, обозначим его $|B_0|$, и размером алфавита кода Рида – Соломона, совпадающим с его длиной N, в соответствии со следующим соотношением: $N^h = |B_0|$. При $h = 1$ один символ внешнего кода отображается в одно слово внутреннего кода, т. е. наблюдается обычная каскадная схема кодирования. Обозначим через R и R_0 скорости внешнего и внутреннего кода соответственно; очевидно, что $R = K/N$, $R_0 = \log_2|B_0|/n$. Общая скорость каскадного кода $R_{\text{общ}}$, определяющая значение плотности записи в ячейку flash-памяти, есть произведение R и R_0, т. е. $R_{\text{общ}} = RR_0 = K\log_2|B_0|/(Nn)$ бит/ячейка.



Структура каскадного кода с перемежением между ступенями

Будем полагать, что внутренний код B_0 определяется следующей кодовой формулой:

$$B_0 = \psi(C_0) + 2\psi(C_1) + \ldots + 2^{L-1}\psi(C_{L-1}) + \\ +2^L\psi\left(F_2^n\right) + 2^{L+1}\psi\left(F_2^n\right) + \ldots + 2^{m-1}\psi\left(F_2^n\right), \quad (1)$$

где $C_0, C_1,\ldots,C_{L-1}$ – двоичные коды, $\psi : F_2^n \to Z^n$ есть естественное вложение прямого произведения n экземпляров поля F_2 в n-мерную целочисленную решетку Z^n. Из (1) следует, что рассматриваемый внутренний код B_0 представляет собой L-уровневый каскадный код с L компонентными двоичными кодами. Пусть компонентные двоичные коды $C_0, C_1,\ldots, C_{L-1}$ в (1) являются линейными кодами с порождающими матрицами $\mathbf{G}_0, \mathbf{G}_1,\ldots, \mathbf{G}_{L-1}$ соответственно и пусть порождающая матрица $\mathbf{G}_j$ имеет размер $k_j \times n$, $0 \le j \le L-1$. Тогда внутренний код B_0 может быть представлен в следующем виде:

$$B_0 = \sum_{j=0}^{L-1} 2^j\psi\left(\left\{\mathbf{a}_l^{(j)}\mathbf{G}_j \,\middle|\, 1 \le l \le \exp_2\left(k_j\right)\right\}\right) + \\ + \sum_{j=L}^{m-1} 2^j\psi\left(\left\{\mathbf{a}_l^{(j)} \,\middle|\, 1 \le l \le \exp_2\left(n\right)\right\}\right), \quad (2)$$

где $\left\{\mathbf{a}_l^{(j)} \,\middle|\, 1 \le l \le \exp_2(k_j)\right\}$ – совокупность двоичных векторов размера k_j, $\left\{\mathbf{a}_l^{(j)} \,\middle|\, 1 \le l \le \exp_2(n)\right\}$ – совокупность двоичных векторов размера n. Объем внутреннего многоуровневого кода, как следует из (1) и (2), есть произведение объемов компонентных кодов $C_0, C_1,\ldots, C_{L-1}$ и подмножества B_1, т. е.

$$\left|B_0\right| = \exp_2\left((m-L)n + \sum_{j=0}^{L-1} k_j\right),$$

так что скорость внутреннего многоуровневого кода

$$R_0 = (m-L) + \sum_{j=0}^{L-1} k_j / n.$$

Представление (2) позволяет в явном виде указать весьма простую процедуру кодирования для внутреннего многоуровневого кода B_0. Двоичный блок $\mathbf{u}$, кодируемый кодом B_0, состо-

ит, очевидно, из $\sum_{j=0}^{L-1} k_j + (m-L)n$ символов.

Представим блок $\mathbf{u}$ в виде набора m подблоков, т. е. $\mathbf{u} = (\mathbf{u}_0, \mathbf{u}_1, ..., \mathbf{u}_{L-1}, \mathbf{u}_L, ..., \mathbf{u}_{m-1})$, где подблок $\mathbf{u}_j$, $0 \le j \le L-1$, имеет длину k_j, а подблоки $\mathbf{u}_L, ..., \mathbf{u}_{m-1}$ имеют длину n. Положим, что кодовое слово $\mathbf{b}$ внутреннего кода B_0, соответствующее входному кодируемому блоку $\mathbf{u}$, имеет вид

$$\mathbf{b} = \sum_{j=0}^{L-1} 2^j \psi(\mathbf{u}_j \mathbf{G}_j) + \sum_{j=L}^{m-1} 2^j \psi(\mathbf{u}_j), \qquad (3)$$

где $\mathbf{G}_j$ – порождающая матрица линейного двоичного (n, k_j) кода C_j, $0 \le j \le L-1$. Нетрудно видеть, что совокупность из

$$\exp_2\left(\sum_{j=0}^{L-1} k_j + (m-L)n\right)$$

n-мерных векторов над алфавитом A, порождаемых согласно (3), совпадает с множеством слов внутреннего кода B_0, задаваемого представлением (2).

Библиографический список

1. *Aritome S.* NAND Flash Memory Technologies. Hoboken, NJ, USA: Wiley, 2016. 432 p.

2. A 5.6 MB/s 64 Gb 4 b/Cell NAND flash memory in 43 nm CMOS / N. Shibata, T. Nakano, M. Ogawa et al. // Proc. IEEE Int. Solid-State Circuits Conf. 2009. P. 246–247.

3. Bit error rate in NAND flash memories / N. Mielke, T. Marquart, N. Wu et al. // Proceedings of IEEE International Reliability Physics Symposium, Phoenix. 2008. P. 9–19.

4. *Micheloni R., Marelli A., Ravasio R.* Error Correction Codes for Non-Volatile Memories. Italy: Springer Science & Business Media. 2008. 338 p.

5. *Dolecek L., Cassuto Y.* Channel coding for nonvolatile memory technologies: Theoretical advances and practical considerations // Proc. of the IEEE. 2017. Vol. 105(9). P. 1705–1724.

6. *Freudenberger J., Kaiser U., Spinner J.* Concatenated code constructions for error correction in non-volatile memories // Int. Symposium on Signals Systems and Electronics (ISSSE). 2012. P. 1–6.

7. *Kurkoski B. M.* Coded modulation using lattices and Reed-Solomon codes, with applications to flash memories // IEEE Transactions on Selected Areas in Communications. 2014. Vol. 32. № 5. P. 900–908.

8. *Таубин Ф. А., Трофимов А. Н.* Каскадное кодирование на основе многомерных решеток и кодов Рида-Соломона для многоуровневой флэш-памяти // Труды СПИИРАН. 2018. Вып. 2 (57). С. 75–103.

УДК 621.391 **DOI: 10.31799/978-5-8088-1701-2-2022-2-301-304**

А. Н. Трофимов*
кандидат технических наук, доцент
Ф. А. Таубин*
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*

КАСКАДНОЕ КОДИРОВАНИЕ С ВНУТРЕННИМ МНОГОУРОВНЕВЫМ КОДОМ ДЛЯ FLASH-ПАМЯТИ – ПРИМЕРЫ КОНСТРУКЦИЙ

Приводятся примеры конструкций, реализующих каскадную конструкцию с внутренним многоуровневым кодом. Рассматривается процедура декодирования внутреннего кода по максимуму правдоподобия с использованием решетчатой диаграммы. Оценивается помехоустойчивость одной из представленных конструкций.
Ключевые слова: *flash-память, каскадные конструкции, код Рида – Соломона, целочисленные многомерные решетки, многоуровневый код.*

A. N. Trofimov*
PhD, Tech., Associate Professor
F. A. Taubin*
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*

CONCATENATED CODING WITH INNER MULTIULEVEL CODE FOR FLASH MEMORY – EXAMPLES

Some examples of concatenated code constructions with inner multilevel code are given. The maximum likelihood decoding based on a trellis representation of the inner code is considered. The code scheme performance is evaluated.
Keywords: *flash memory, concatenated constructions, Reed-Solomon code, multidimensional integer lattices, multilevel code.*

Выбор внутреннего многоуровневого кода

Общая схема рассматриваемого каскадного кодирования с внутренним многоуровневым кодом представлена ранее. В ней квадрат минимального евклидова расстояния внутреннего кода B_0, задается представлением

$$\delta_0^2 = \min\begin{Bmatrix} 4^0 d_H(C_0), 4^1 d_H(C_1), \ldots, \\ 4^{L-1} d_H(C_{L-1}), 4^L \end{Bmatrix},$$

где $d_H(C_j)$ – минимальное расстояние кода C_j, $0 \le j \le L-1$. Отметим, что для рассматриваемой далее модели многоуровневой flash-памяти евклидово расстояние лишь косвенным образом определяет вероятность ошибки декодирования внутреннего кода. Тем не менее эта характеристика оказывается в достаточной степени адекватной при конструировании подходящих каскадных схем кодирования для многоуровневой flash-памяти.

Ограничимся далее конкретным рассмотрением двухуровневых ($L = 2$) конструкций внутреннего кода. При выборе подходящих (для рассматриваемой ситуации) компонентных кодов C_0 и C_1 внутреннего кода B_0 будем руководствоваться следующими соображениями. Во-первых, обеспечение максимально возможного значения квадрата минимального расстояния $\delta_0^2 = 16$ (и ближайшего возможного к нему значения $\delta_0^2 = 12$) сопряжено с использованием достаточно длинных кодов (n порядка 30 и более) с неприемлемо высокой сложностью декодирования и довольно малой плотностью записи. Учитывая, что ближайшее возможное к 16 и 12 значение квадрата минимального расстояния есть 8, получаем, что в качестве кода C_1 целесообразно использовать код с проверкой на четность. Во-вторых, желательно выбирать код C_0 таким образом, чтобы он: а) был лучшим среди известных коротких кодов с заданными параметрами n и k_0 (или близок к лучшему), б) имел достаточно простое описание в виде решетча-

Таблица 1

Параметры каскадных конструкций с внутренним двухуровневым кодом B_0 на основе циклически усеченного сверточного кода C_0 и кода проверки на четность C_1

Коды C_0 и C_1	Объем внутреннего кода $\lvert B_0\rvert$	Квадрат минимального расстояния δ_0^2	Возможные значения h и длин кода Рида – Соломона N	Плотность записи $R_{\text{общ}}$, бит/ячейка
$C_0 = (8,4,4)$ $C_1 = (8,7,2)$	2^{11}	4	$N = 2048$, $h = 1$	1,375 R
$C_0 = (8,2,5)$ $C_1 = (8,7,2)$	2^9	5	$N = 8$, $h = 3$; $N = 512$, $h = 1$	1,125 R
$C_0 = (12,4,6)$ $C_1 = (12,11,2)$	2^{15}	6	$N = 8$, $h = 5$; $N = 32$, $h = 3$	1,25 R
$C_0 = (12,3,6)$ $C_1 = (12,11,2)$	2^{14}	6	$N = 128$, $h = 2$	1,167 R
$C_0 = (18,9,6)$ $C_1 = (18,17,2)$	2^{26}	6	$N = 8192$, $h = 2$	1,444 R
$C_0 = (21,7,8)$ $C_1 = (21,20,2)$	2^{27}	8	$N = 8$, $h = 9$; $N = 512$, $h = 3$	1,286 R

той диаграммы, в) позволял использовать отлаженные и хорошо апробированные на практике аппаратные реализации декодера. С учетом указанных соображений будем полагать, что в качестве кода C_0 используется циклически усеченный сверточный код (tail-biting convolutional code, TBCC). Указанный класс конструкций внутреннего кода будем называть конструкциями TB/SPC. В табл. 1 [1] приведены параметры ряда каскадных конструкций для четырехуровневой flash-**памяти** ($q = 4$) на основе внутреннего TB/SPC кода, имеющего длину $n \le 24$ и скорость $R_0 \ge 1{,}125$.

Сопоставление приведенных в табл. 1 вариантов и внутренних кодов, построенных на основе решеток Барнса – Уолла, представленных в [2], показывает, что в рамках обменных соотношений между плотностью записи, минимальным расстоянием и сложностью декодирования представленная двухуровневая конструкция позволяет получить существенно более широкий выбор вариантов.

Декодирование внутреннего кода

Внутренний код B_0 можно рассматривать как прямую сумму взвешенных (с весами 2^0, $2^1,\ldots, 2^{L-1}$) ψ-вложений кодовых слов двоичных кодов C_0, $C_1,\ldots, C_{L-1}$ и подмножества

$$B_1 = 2^L \psi(F_2^n) + 2^{L+1}\psi(F_2^n) + \ldots + 2^{m-1}\psi(F_2^n).$$

Это означает в свою очередь, что внутренний код может быть представлен в виде bit-level trellis решетчатой диаграммы, состоящей из n ярусов. Такая диаграмма, обозначим ее Ω, может быть представлена в виде прямого произведения bit-level решетчатых диаграмм кодов C_0, $C_1,\ldots, C_{L-1}$. Важным достоинством предложенной двухуровневой TB/SPC конструкции является сравнительная простота формирования решетчатой диаграммы Ω. В этом случае решетчатая диаграмма Ω для каждого из приведенных вариантов внутреннего кода B_0 может быть получена с помощью небольшой модификации решетчатой диаграммы циклически усеченного сверточного кода C_0 со скоростью k_0/n и кодовым ограничением ν_0. При этом: а) длина новой решетчатой диаграммы Ω не меняется и составляет k_0 секций, б) число начальных и конечных состояний не меняется и составляет 2^{ν_0}, число промежуточных состояний возрастает вдвое – до величины $S_\Omega = 2^{\nu_0+1}$, в) смежные состояния решетчатой диаграммы Ω соединяются $2^{n/k_0-1}$ параллельными ребрами. Такая структура решетчатой диаграммы Ω позволяет использовать для декодирования внутреннего кода B_0 процедуры (с небольшой модификацией), разработанные для декодирования циклически усеченных сверточных кодов. Обозначим через X_i, $1 \le i \le S_\Omega/2$, совокупность путей (подрешетку) в решетчатой диаграмме Ω, исходящих из состояния с номером i и заканчивающихся в состоянии с номером i. Очевидно, решетчатая диаграмма Ω может быть представлена как объединение $\bigcup_{i=1}^{S_\Omega/2} X_i$ подрешеток с идентичными начальными и конечными состояниями. С учетом такого представления декодирование по максимуму правдоподобия может быть реализовано как последовательный (по всем подрешеткам $\{X_i,\ 1 \le i \le S_\Omega/2\}$) поиск пути с минимальным суммарным весом (вес каждого ребра решетчатой диаграммы назначается в соответ-

ствии с правилом, приведенным в работе [3]). Указанная процедура декодирования имеет наиболее простую логическую структуру, однако высокая сложность реализации может стать в ряде ситуаций серьезным ограничительным фактором. Вместе с тем для представленных в табл. 1 вариантов внутреннего кодирования количество последовательно просматриваемых подрешеток лежит в пределах от 4 до 16, что вполне приемлемо.

Краткое описание модели многоуровневой flash-памяти и оценка характеристик надежности хранения данных

В изучаемой модели блок flash-памяти рассматривается как множество независимых ячеек и полагается, что физический носитель может пониматься как стационарный канал без заметной тенденции к пакетированию ошибок. При описании упрощенной математической модели одиночной ячейки flash-памяти мы используем модель из [4, 5]. Входные уровни каждой ячейки принимают некоторые фиксированные значения $x_0,x_1,\ldots,x_{q-1}$, а выходные значения представляют собой случайные величины, описываемые условными функциями плотности вероятности (ф. п. в.) $p_{y|x}(y|x)$, $-\infty < y < \infty$, $x = x_0,x_1,\ldots,x_{q-1}$, которые могут быть заданы как

$$p_{y|x}(y \mid x_i) = \frac{1}{\sqrt{2\pi}\sigma(x_i)} \exp\left(-\frac{(y-x_i)^2}{2\sigma^2(x_i)} \right).$$

Такое описание называется также моделью с гауссовским шумом, стандартное отклонение которого $\sigma(x_i)$ зависит от входного значения x_i [1], или ID-AGN моделью (input-dependent additive Gaussian noise). Важная особенность модели ячейки многоуровневой flash-памяти в том, что с ростом числа циклов перезаписи и времени хранения значения x_i (кроме x_0) уменьшаются, а значения $\sigma(x_i)$ (кроме $\sigma(x_0)$) увеличиваются, что соответствует ухудшению канала. Зависимость параметров x_i и $\sigma(x_i)$, $i = 0,1,\ldots, q-1$, числа циклов перезаписи и времени хранения выражается сложным образом и здесь не приводится. Детали описания этой зависимости могут быть найдены в [4, 5]. Распространенным примером служит значение $q = 4$, хотя эта модель допускает обобщение на большее число входных уровней.

Оценка помехоустойчивости предложенных каскадных кодовых конструкций включает два этапа. На первом с использованием новой техники точного вычисления аддитивной границы, представленной в работе [6], оценивается вероятность ошибки декодирования по максимуму правдоподобия слова внутреннего кода. На втором этапе вычисляется вероятность ошибки декодирования слова внешнего кода (кода Рида – Соломона) с использованием результатов, полученных на первом этапе. В качестве примера рассмотрим зависимость вероятности ошибки на бит от числа циклов перезаписи для кодовой конструкции TB/SPC во внутренней ступени и кода Рида – Соломона (512,502) во внешней ступени. Это код обеспечивает плотность записи 1,261 бит/ячейка. В работе [2] мы представили и дали анализ для каскадного кода, построенного с использованием подмножества решетки Λ_{16} на внутренней ступени.

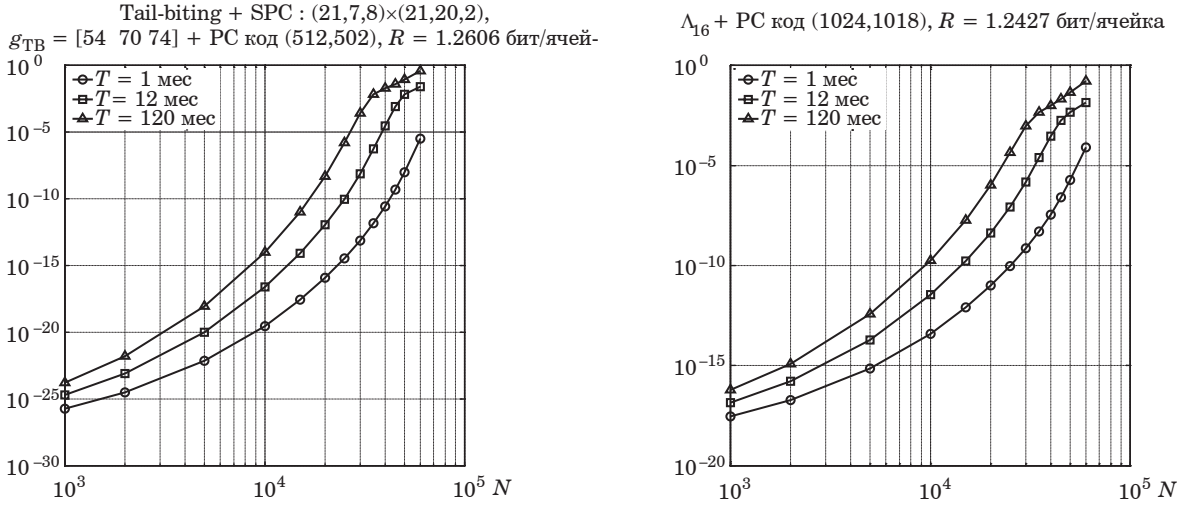


Рис. 1. Вероятность ошибки на бит для каскадных кодов с использованием TB/SPC (слева) и кода с аналогичной плотностью записи из [2] (справа) в зависимости от числа циклов перезаписи

Таблица 2

Количество циклов перезаписи (по сравнению с результатами [3])

Параметры внутреннего и внешнего кода	Плотность записи	$P_b=10^{-10}$			$P_b=10^{-15}$		
		T = 1 мес.	T = 12 мес.	T = 120 мес.	T = 1 мес.	T = 12 мес.	T = 120 мес.
Λ_{16}, (1024,1018), t = 3, [4]	1,243	$2{,}4\cdot10^4$	$1{,}4\cdot10^4$	10^4	$5\cdot10^3$	$3\cdot10^3$	$2\cdot10^3$
TB(21,7,8)/SPC(21,20,2), (512,502), t = 5	1,261	$4\cdot10^4$	$2{,}4\cdot10^4$	$1{,}6\cdot10^4$	$2{,}1\cdot10^4$	$1{,}3\cdot10^4$	$8\cdot10^3$
Увеличение, %	1,4	67	71	60	320	333	300

пени и кода Рида – Соломона (1024,1018), обеспечивающего примерно такое же значение плотности записи. Данные для этих примеров приведены на рис. 1 и в табл. 2.

Эти и другие результаты, полученные для кодов, построенных с использованием конструкции TB/SPC, но не представленные здесь, показывают, что эта схема может обеспечить внушительный выигрыш (до трех раз и более) по числу циклов перезаписи при равной плотности записи и надежности хранения данных.

Библиографический список

1. *Таубин Ф. А., Трофимов А. Н.* Каскадное кодирование с внутренним двухуровневым TB/SPC кодом для многоуровневой flash-памяти // Волновая электроника и инфокоммуникационные системы: XXIII Междунар. науч. конф.: сб. ст. Ч. 1. СПб., 2020. С. 354–361.

2. *Таубин Ф. А., Трофимов А. Н.* Каскадное кодирование для многоуровневой флэш-памяти с исправлением ошибок малой кратности во внешней ступени // Труды СПИИРАН. 2019. Вып. 18 (5). С. 1149–1181.

3. *Таубин Ф. А., Трофимов А. Н.* Каскадное кодирование на основе многомерных решеток и кодов Рида – Соломона для многоуровневой флэш-памяти // Труды СПИИРАН. 2018. Вып. 2 (57). С. 75–103.

4. Error correction codes and signal processing in flash memory / X. Wang, G. Dong, L. Pan, R. Zhou // Flash Memories / ed. Igor Stievano. URL: http://www.intechopen.com/books/flash-memories/error-correction-codes-and-signal-processing-in-flash-memory (дата обращения: 26.10.2014).

5. Estimating information – theoretical NAND flash memory storage capacity and its implication to memory system design space exploration / G. Dong, Y. Pan, N. Xie et al. // IEEE Trans. Very Large Scale Integration (VLSI) Systems. 2012. Vol. 20, № 9. P. 1705–1714.

6. *Трофимов А. Н., Таубин Ф. А.* Вычисление аддитивной границы вероятности ошибки декодирования с использованием характеристических функций // Информационно-управляющие системы. 2021. Вып. 4. С. 71–85.

ИНФОРМАЦИОННО-СЕТЕВЫЕ ТЕХНОЛОГИИ

УДК 621.396:681.323 **DOI: 10.31799/978-5-8088-1701-2-2022-2-305-307**

С. И. Зиатдинов
доктор технических наук, профессор
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ИССЛЕДОВАНИЕ ВЛИЯНИЯ РАССОГЛАСОВАНИЯ КОЭФФИЦИЕНТОВ ПЕРЕДАЧИ КАНАЛОВ КОГЕРЕНТНОЙ СИСТЕМЫ НА ОЦЕНКУ ПАРАМЕТРОВ КОМПЛЕКСНОГО СИГНАЛА

Рассматривается вопрос измерения параметров комплексного сигнала когерентной системой, выполненной в виде двух квадратурных каналов. Представлены выражения для оценки амплитуды и частоты флюктуирующего узкополосного комплексного сигнала при наличии ошибок настройки коэффициентов передачи квадратурных каналов. Произведено исследование влияния неизбежных на практике отличий коэффициентов передачи каналов на ошибки измерения основных параметров комплексного сигнала. Показано, что ошибки настройки коэффициентов передачи каналов когерентной системы обработки приводят к заметным погрешностям при измерении амплитуды и частоты узкополосного комплексного сигнала.
Ключевые слова: *ошибки настройки, комплексный сигнал, измерение параметров.*

S. I. Ziatdinov
Dr. Sc., Tech., Professor
St. Petersburg State University of Aerospace Instrumentation

STUDY OF THE INFLUENCE OF THE DISCONNECTION OF THE TRANSMISSION COEFFICIENTS OF THE CHANNELS OF THE COHERENT SYSTEM ON THE ESTIMATION OF THE PARAMETERS OF THE COMPLEX SIGNAL

The question of measuring the parameters of a complex signal by a coherent system made in the form of two quadrature channels is considered. Expressions for estimating the amplitude and frequency of a fluctuating narrow-band complex signal in the presence of errors in adjusting the transmission coefficients of the quadrature channels are presented. The study of the influence of the inevitable on practice differences in the transmission coefficients of the channels on the measurement errors of the main parameters of the complex signal. It is shown that errors in tuning the transmission coefficients of the channels of a coherent processing system lead to noticeable errors in measuring the amplitude and frequency of a narrow-band complex signal.
Keywords: *tuning errors, complex signal, parameter measurement.*

Введение

В когерентных системах обработки с целью упрощения аппаратной реализации устройств используется преобразование обрабатываемых сигналов на видеочастоту на базу двух квадратурных каналов, на выходе которых формируются вещественная и мнимая составляющие комплексного сигнала. Рассмотренные в известной литературе методы обработки комплексных сигналов основаны на безошибочном преобразовании входных высокочастотных радиосигналов на видеочастоту. Однако реализовать на практике абсолютно идентичные каналы обработки не представляется возможным. Каналы могут иметь различные коэффициенты передачи, а используемые для преобразования опорные напряжения могут иметь фазовый сдвиг, отличный от 90°. Исследование влияния шибок настройки коэффициентов передачи квадратурных каналов когерентной системы обработки на точность измерения амплитуды и частоты узкополосных комплексных сигналов составляет основное содержание статьи.

Модель комплексного сигнала

С учетом ошибок настройки коэффициентов передачи квадратурных каналов их выходные сигналы представим следующей парой комплексно сопряженных составляющих [1, 2]:

$$x(t) = U(t)\cos[\omega_0 t + \varphi_0(t)] = U(t)\cos\varphi(t),$$

$$y(t) = kU(t)\sin[\omega_0 t + \varphi_0(t)] = kU(t)\sin\varphi(t),$$

где $U(t)$ и $\varphi_0(t)$ – флюктуирующие амплитуда и начальная фаза сопряженных сигналов; ω_0 – средняя частота; $k = 1 + \Delta k$; Δk – отклонение коэффициентов передачи квадратурных каналов; $\varphi(t) = \omega_0 t + \varphi_0(t)$.

При точной настройке квадратурных каналов необходимо положить $\Delta k = 0$. На практике $\Delta k \neq 0$.

В дальнейшем рассмотрим влияние ошибок настройки квадратурных каналов Δk на оценку амплитуды $U(t)$ и частоты $\omega(t) = d\varphi(t)/dt$ рассматриваемого комплексного сигнала.

Влияние отклонения коэффициентов передачи квадратурных каналов на оценку амплитуды комплексного сигнала

В рассматриваемом случае, когда $\Delta k \neq 0$, сопряженные сигналы принимают вид

$$x(t) = U(t)\cos[\omega_0 t + \varphi_0(t)],$$

$$y(t) = kU(t)\sin[\omega_0 t + \varphi_0(t)].$$

Квадрат амплитуды комплексного сигнала $z(t) = u_x(\mathrm{t}) + j\,u_y(\mathrm{t})$ может быть найден из соотношения

$$U_z{}^2(t) = u_x{}^2(t) + u_y{}^2(t) = U^2(t)\{\cos[\omega_0 t + \varphi_0(t)]\}^2 + k^2U^2(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2.$$

С учетом того, что $k = 1 + \Delta k$, данное выражение принимает вид

$$U_z{}^2(t) = U^2(t)\{\cos[\omega_0 t + \varphi_0(t)]\}^2 + U^2(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2 + 2\Delta kU^2(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2 + \Delta k^2U^2(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2 = U^2(t)\{1 + (2\Delta k + \Delta k^2)\{\sin[\omega_0 t + \varphi_0(t)]\}^2\}.$$

Воспользуемся известным соотношением $\sin^2 x = 0{,}5(1 - \cos 2x)$. Тогда полученное выражение можно записать следующим образом:

$$U_z{}^2(t) = U^2(t)\{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}.$$

Полученное соотношение показывает, что появление амплитудного рассогласования Δk квадратурных каналов приводит к ошибке оценки амплитуды комплексного сигнала. Величина возникающей ошибки оценки амплитуды имеет вид

$$\Delta U(t) = U(t) - U_z(t) = U(t)\{1 - \{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}^{0{,}5}.$$

Поскольку реально $\Delta k << 1$, то с учетом разложения квадратного корня в биномиальный ряд данное выражение можно упростить [3, 4]:

$$\Delta U(t) = U(t) - U_z(t) = U(t)\{1 - \{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}^{0{,}5} \approx \approx \Delta kU(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2.$$

Тогда максимальная ошибка $\Delta U_{\max} \approx \Delta kU(t)$ пропорциональна величине амплитудного рассогласования каналов Δk. Полученные соотношения показывают, что ошибка оценки амплитуды комплексного сигнала, вызванная амплитудным рассогласованием квадратурных каналов, является не только флюктуирующей функцией времени, но и содержит колебательную составляющую с частотой сигнала. При точной настройке квадратурных каналов, когда $\Delta k = 0$, ошибка оценки амплитуды равна нулю.

Влияние отклонения коэффициентов передачи квадратурных каналов на оценку амплитуды комплексного сигнала

Согласно [2], оценка частоты сигнала производится с помощью следующего соотношения:

$$\omega_z(t) = \{u_x(t)[du_y(t)/dt] - u_y(t)[du_x(t)/dt]\}/[u^2{}_x(t) + u^2{}_y(t)]\}.$$

С учетом ранее принятой модели комплексного сигнала производные в данном выражении имеют вид

$$du_x(t)/dt = [dU(t)/dt]\cos[\omega_0 t + \varphi_0(t)] - U(t)[\omega_0 + d\varphi_0(t)/dt\,]\sin[\omega_0 t + \varphi_0(t)],$$

$$du_y(t)/dt = k\,[dU(t)/dt]\sin[\omega_0 t + \varphi_0(t)] - U(t)[\omega_0 + d\varphi_0(t)/dt\,]\cos[\omega_0 t + \varphi_0(t)].$$

Знаменатель в выражении для частоты определяется соотношением

$$U^2(t)\{1 - \{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}.$$

Тогда выражение для частоты принимает вид

$$\omega_z(t) = \{kU(t)\sin[\omega_0 t + \varphi_0(t)][dU(t)/dt]\cos[\omega_0 t + \varphi_0(t)] - U(t)[\omega_0 + d\varphi_0(t)dt]\sin[\omega_0 t + + \varphi_0(t)]\}/U^2(t)\{1 - \{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}.$$

После несложных преобразований данное соотношение приводится к форме

$$\omega_z(t) = (1 + \Delta k)[\omega_0 + d\varphi_0(t)dt]/\{1 - \{1 + 0{,}5(2\Delta k + \Delta k^2)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}.$$

С учетом того, что $\Delta k << 1$, можно записать

$$\omega_z(t) = (1 + \Delta k)[\omega_0 + d\varphi_0(t)dt]/\{1 - \{1 + \Delta k)\times \times\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}.$$

Полученное соотношение показывает, что при появлении амплитудного рассогласования Δk квадратурных каналов возникает ошибка оценки частоты комплексного сигнала. Величина появившейся ошибки определяется соотношением

$$\Delta\omega(t) = \omega_0 + d\varphi_0(t)dt - \omega_z(t) = [\omega_0 + d\varphi_0(t)dt]\ ]/ /\{1 - \{1 + \Delta k)\{1 - \cos\{2[\omega_0 t + \varphi_0(t)]\}\}\}\}.$$

При $\Delta k << 1$ можно записать, что $\Delta\omega_{max} \approx \Delta k\omega(t)$.

Из полученных результатов следует, что максимальное значение ошибки измерения частоты пропорционально величине амплитудного рассогласования каналов. При точной настройке каналов, когда $\Delta k = 0$, ошибка оценки частоты равна нулю.

Библиографический список

1. *Бакулев П. А.* Радиолокационные системы. М.: Радиотехника, 2004. 319 с.

2. *Тихонов В. И.* Статистический анализ и синтез радиотехнических устройств и систем. М.: Радио и связь, 1991. 608 с.

3. *Пискунов Н. С.* Дифференциальное и интегральное исчисления. Т. 2. М.: Наука, 1965. 310 с.

4. *Цыпкин А. Г.* Справочник по математике. М.: Наука. 1988. 431 с.

УДК 621.396:681.323 **DOI: 10.31799/978-5-8088-1701-2-2022-2-308-310**

С. И. Зиатдинов
доктор технических наук, профессор
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ВЛИЯНИЕ ФАЗОВОГО РАССОГЛАСОВАНИЯ КВАДРАТУРНЫХ КАНАЛОВ КОГЕРЕНТНОЙ СИСТЕМЫ НА ОЦЕНКУ АМПЛИТУДЫ И ЧАСТОТЫ КОМПЛЕКСНОГО СИГНАЛА

Исследуется вопрос измерения амплитуды и частоты комплексного сигнала когерентной системой, выполненной в виде двух квадратурных каналов. Получены выражения для оценки амплитуды и частоты флюктуирующего узкополосного комплексного сигнала при фазовом рассогласовании квадратурных каналов. Произведено исследование влияния неизбежного на практике отличия от 90° опорных напряжений каналов на ошибки измерения основных параметров комплексного сигнала. Показано, что фазовое рассогласование квадратурных каналов когерентной системы обработки приводит к заметным погрешностям при измерении амплитуды и частоты узкополосного комплексного сигнала.
Ключевые слова: *ошибки настройки, комплексный сигнал, измерение параметров.*

S. I. Ziatdinov
Dr. Sc., Tech., Professor
St. Petersburg State University of Aerospace Instrumentation

THE INFLUENCE OF THE PHASE MATCHING OF THE QUALITY CHANNELS OF THE COHERENT SYSTEM ON THE ESTIMATION OF THE AMPLITUDE AND FREQUENCY OF THE COMPLEX SIGNAL

The problem of measuring the amplitude and frequency of a complex signal by a coherent system made in the form of two quadrature channels is investigated. Expressions are obtained for estimating the amplitude and frequency of a fluctuating narrow-band complex signal with phase mismatch of quadrature channels. The study of the influence of the inevitable on practice difference from ninety degrees of the reference voltages of the channels on the measurement errors of the main parameters of the complex signal has been carried out. It is shown that the phase mismatch of the quadrature channels of a coherent processing system leads to noticeable errors in measuring the amplitude and frequency of a narrow-band complex signal.
Keywords: *tuning errors, complex signal, parameter measurem.*

Введение

В когерентных системах обработки с целью упрощения аппаратной реализации устройств используется преобразование обрабатываемых сигналов на видеочастоту на базу двух квадратурных каналов, на выходе которых формируются вещественная и мнимая составляющие комплексного сигнала. Рассмотренные в известной литературе методы обработки комплексных сигналов основаны на безошибочном преобразовании входных высокочастотных радиосигналов на видеочастоту. Однако реализовать на практике абсолютно идентичные каналы обработки не представляется возможным. Каналы могут иметь различные коэффициенты передачи, а используемые для преобразования опорные напряжения могут иметь фазовый сдвиг, отличный от 90°. Исследование влияния фазового рассогласования квадратурных каналов когерентной системы обработки на точность измерения амплитуды и частоты узкополосных комплексных сигналов составляет основное содержание статьи.

Модель комплексного сигнала

С учетом возможного фазового рассогласования квадратурных каналов их выходные сигналы представим следующей парой комплексно сопряженных составляющих [1, 2]:

$$x(t) = U(t)\cos[\omega_0 t + \varphi_0(t)] = U(t)\cos\varphi(t),$$

$$y(t) = U(t)\sin[\omega_0 t + \varphi_0(t) + \Delta\varphi] = U(t)\sin\varphi(t),$$

где $U(t)$ и $\varphi_0(t)$ – флюктуирующие амплитуда и начальная фаза сопряженных сигналов; ω_0 –

средняя частота; $\Delta\varphi$ – фазовое рассогласование квадратурных каналов; $\varphi(t) = \omega_0 t + \varphi_0(t) + \Delta\varphi$.

При точной настройке квадратурных каналов необходимо положить $\Delta\varphi = 0$. На практике $\Delta\varphi \neq 0$.

В дальнейшем рассмотрим влияние фазового рассогласования $\Delta\varphi$ квадратурных каналов на оценку амплитуды $U(t)$ и частоты $\omega(t) = d\varphi(t)/dt$ рассматриваемого комплексного сигнала.

Влияние фазового рассогласования квадратурных каналов на оценку амплитуды комплексного сигнала

Квадрат амплитуды комплексного сигнала $z(t) = u_x(t) + ju_y(t)$ может быть найден из соотношения

$$U_z^2(t) = u_x^2(t) + u_y^2(t) = U^2(t)\{\cos[\omega_0 t + \varphi_0(t)]\}^2 + \\ + U^2(t)\{\sin[\omega_0 t + \varphi_0(t) + \Delta\varphi]\}.$$

Воспользуемся известным тригонометрическим выражением [3]:

$$\sin[\omega_0 t + \varphi_0(t) + \Delta\varphi] = \sin[\omega_0 t + \varphi_0(t)]\cos\Delta\varphi + \\ \cos[\omega_0 t + \varphi_0(t)]\sin\Delta\varphi.$$

Тогда после несложных преобразований данное выражение приобретает вид

$$U_z^2(t) = U^2(t)\{\cos[\omega_0 t + \varphi_0(t)]\}^2(1 + \sin^2\Delta\varphi)\} + \\ + U^2(t)\{\sin[\omega_0 t + \varphi_0(t)]\}^2\cos^2\Delta\varphi + \\ + U^2(t)\sin\{2[\omega_0 t + \varphi_0(t)]\}\sin\Delta\varphi\cos\Delta\varphi.$$

При малом фазовом рассогласовании $\Delta\varphi$ $\cos\Delta\varphi \approx 1$ и $\sin\Delta\varphi \approx \Delta\varphi$ [4]. В результате после несложных выкладок выражение для амплитуды комплексного сигнала можно записать следующим образом:

$$U_z^2(t) = U^2(t)\{1 + \Delta\varphi^2\{\cos[\omega_0 t + \varphi_0(t)]\}^2 + \\ + \Delta\varphi\,(t)\sin\{2[\omega_0 t + \varphi_0(t)]\}\} = \\ = U^2(t)\{\,1 + 0{,}5\Delta\varphi^2 + (\Delta\varphi - 0{,}5\Delta\varphi^2)\sin\{2[\omega_0 t + \\ + \varphi_0(t)]\}\}.$$

С учетом малости величины фазового рассогласования $\Delta\varphi$ полученное выражение можно дополнительно упростить:

$$U_z^2(t) = U^2(t)\{\,1 + \Delta\varphi)\sin\{2[\omega_0 t + \varphi_0(t)]\}\} \text{ или}$$

$$U_z(t) = U(t)\{1 + \Delta\varphi)\sin\{2[\omega_0 t + \varphi_0(t)]\}\}^{0,5}.$$

Величина ошибки оценки амплитуды комплексного сигнала, вызванная фазовым рассогласованием квадратурных каналов, составит

$$\Delta U(t) = U(t) - U_z(t) = U(t)(1 - \{1 + \Delta\varphi)\sin\{2[\omega_0 t + \\ + \varphi_0(t)]\}\}^{0,5}\}.$$

При $\Delta\varphi << 1$

$$\Delta U(t) = 0{,}5U(t)\Delta\varphi\sin\{2[\omega_0 t + \varphi_0(t)]\}.$$

Тогда максимальную ошибку измерения амплитуды можно определить соотношением

$$\Delta U_{\max} \approx 0{,}5\Delta\varphi U(t).$$

В результате видно, что максимальная ошибка измерения амплитуды комплексного сигнала пропорциональна величине фазового рассогласования каналов.

Влияние фазового рассогласования квадратурных каналов на оценку частоты комплексного сигнала

Согласно [2], оценка частоты сигнала производиться с помощью следующего соотношения:

$$\omega_z(t) = \{u_x(t)[du_y(t)/dt] - u_y(t)[du_x(t)/dt]\}/[u^2{}_x(t) + \\ + u^2{}_y(t)]\}.$$

В данном соотношении производные имеют вид

$$du_x(t)/dt = [dU(t)/dt]\cos\varphi(t)] - U(t)[d\varphi(t)/dt]\sin\varphi(t),$$

$$du_y(t)/dt = [dU(t)/dt]\sin[\varphi(t) + \Delta\varphi] - \\ - U(t)d\varphi(t)/dt]\cos[\varphi(t) + \Delta\varphi],$$

где $\varphi(t) = \omega_0 t + \varphi_0(t)$.

После подстановки данных выражений в формулу для частоты, опуская несложные промежуточные преобразования, получим

$$\omega_z(t) = \omega(t)\{\cos\Delta\varphi/\{1 + \Delta\varphi)\sin\{2[\omega_0 t + \varphi_0(t)]\}\},$$

где $\omega(t) = \omega_0 + d\varphi_0(t)/dt$.

Из полученного соотношения видно, что при отсутствии фазового рассогласования квадратурных каналов ошибка измерения частоты комплексного сигнала равняется нулю. При $\Delta\varphi \neq 0$ величина ошибки измерения частоты

$$\Delta\omega(t) = \omega(t) - \omega_z(t) = \omega(t)\{1 - \cos\Delta\varphi/\{1 + \Delta\varphi)\times \\ \times\sin\{2[\omega_0 t + \varphi_0(t)]\}\}.$$

При $\Delta\varphi << 1$ максимальное значение ошибки измерения частоты $\omega(t)$ составит $\Delta\omega_{\max} \approx \approx 0{,}5\Delta\varphi^2\omega(t)$.

Таким образом, максимальное значение ошибки измерения частоты пропорционально квадрату фазового рассогласования каналов.

Заключение

Наличие фазовых рассогласований квадратурных каналов когерентной системы обработки сигналов приводит к ошибкам измерения как

амплитуды, так и частоты узкополосного комплексного сигнала, содержащим колебательную составляющую с частотой обрабатываемого сигнала. Величина максимальной ошибки измерения амплитуды комплексного сигнала, вызванная фазовым рассогласованием квадратурных каналов, пропорциональна фазовому отклонению опорных напряжений. В то же время максимальная величина ошибки измерения частоты комплексного сигнала пропорциональна квадрату значения фазового рассогласования.

Библиографический список

1. *Бакулев П. А.* Радиолокационные системы. М.: Радиотехника, 2004. 319 с.

2. *Тихонов В. И.* Статистический анализ и синтез радиотехнических устройств и систем. М.: Радио и связь, 1991. 608 с.

3. *Пискунов Н. С.* Дифференциальное и интегральное исчисления. Т. 2. М.: Наука, 1965. 310 с.

4. *Цыпкин А. Г.* Справочник по математике. М.: Наука. 1988. 431 с.

УДК 519.254 **DOI: 10.31799/978-5-8088-1701-2-2022-2-311-316**

Д. М. Клионский*
кандидат технических наук, доцент
В. В. Геппенер**
доктор технических наук, профессор
**Санкт-Петербургский государственный университет аэрокосмического приборостроения*
***Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»*

ОБРАБОТКА ИЗМЕРИТЕЛЬНЫХ ДАННЫХ ВО ВРЕМЕННОЙ И ЧАСТОТНОЙ ОБЛАСТЯХ С ИСПОЛЬЗОВАНИЕМ ПАРАМЕТРИЧЕСКИХ МОДЕЛЕЙ ВРЕМЕННЫХ РЯДОВ

Рассматриваются измерительные данные различной природы (вибрационные, медленно меняющиеся) и способы их обработки во временной и частотной областях. Данные, исследуемые в работе, получены по результатам испытаний объектов ракетно-космической техники, которые имеют важное значение для дальнейшего пуска изделия в эксплуатацию. Рассматриваются алгоритмы анализа вибрационных сигналов во временной и частотной областях.
Ключевые слова: *измерительные данные, вибрационные данные, медленно меняющиеся параметры, испытания объектов РКТ, временная область, частотная область.*

D. M. Klionskiy*
PhD, Tech., Associate Professor
V. V. Geppener**
Dr. Sc., Tech., Professor
**St. Petersburg State University of Aerospace Instrumentation*
***St. Petersburg Electrotechnical University «LETI»*

PROCESSING OF MEASUREMENT DATA IN TIME AND FREQUENCY DOMAINS USING PARAMETRIC MODELS OF TIME SERIES

The paper discusses measurement data of different types (vibrational slow-changing) and the ways of their processing in the time and frequency domains. The data are obtained as a result of tests of space-rocket objects and have further significance for further exploitation of the object. We also consider algorithms for vibrational signal processing in time and frequency domains.
Keywords: *measurement data, vibrational data, slow-changing signals, tests of space-rocket objects, time domain, frequency domain.*

Испытания объектов ракетно-космической техники (объектов РКТ, ракет, ракетно-космических комплексов) – важный этап, который сопровождает процесс подготовки к дальнейшей эксплуатации, самостоятельное практическое значение имеют испытания [1–4]. Подготовка к проведению испытания занимает определенное время и направлена на то, чтобы оно прошло успешно, а по результатам была получена необходимая информация. В дальнейшем эта информация используется специалистами-обработчиками для анализа результатов испытаний и принятия решений о вводе того или иного изделия в эксплуатацию. Практически каждый процесс проведения испытания имеет индивидуальные особенности и ряд отличительных процедур, кроме того, испытания сопровождаются собственным набором внешних условий, которые также учитываются и в дальнейшем размещаются в протоколе проведения испытаний. Программа, которая ориентирована на испытания, называется *циклограммой*, ей целесообразно следовать в процессе проведения испытаний и в дальнейшем при выполнении анализа результатов. После анализа результатов испытания изучаются во временной и частотной областях математические параметры полученных сигналов и данных, а также различного рода характеристики, которые рассчитываются по полученным сигналам. Объект РКТ оснащен, как правило, большим набором датчиков, число которых может варьироваться

(например, от 10 до 30). Датчики подразделяются на датчики вибраций, температуры, давления и т. д. Каждый из них осуществляет непрерывный посыл сигналов, и эти сигналы можно условно разделить на две группы: медленно меняющиеся сигналы/параметры (ММП) и быстро меняющиеся сигналы/параметры (БМП) [5]. При необходимости на наземном приемном пункте сигналы можно сохранить в базе для того, чтобы проанализировать в дальнейшем и сравнить с результатами других испытаний.

Обработка сигналов (в большинстве своем это сигналы, полученные по телеметрическим каналам связи) может проводиться как ручным методом, так и автоматизированными математическими методами, которые на настоящий момент включают такую обширную группу, как временные методы (основанные на статистическом анализе, прогнозировании данных, применении параметрических моделей и др.). Также довольно широко и с хорошей эффективностью могут применяться частотные методы, в том числе классический спектральный анализ (Фурье-периодограмма, модифицированные периодограммы, Фурье-периодограмма с окном, Фурье-спектрограмма), параметрический спектральный анализ: методы, основанные на моделях авторегрессии (модель с линейно независимыми последовательными по времени состояниями), скользящего среднего, авторегрессии скользящего среднего и др., и обширная группа частотно-временных методов, которые представляют полученный сигнал на частотно-временной плоскости. Это означает, что сигнал рассматривается в системе координат «время–частота», а информация об энергии сигнала закодирована цветом. В качестве примера широко применяемого на практике частотно-временного распределения можно привести распределение Вигнера – Вилля, Фурье-спектрограммы, вейвлет-спектрограммы и др.

Рассмотрим схему обработки БМП (быстро меняющихся параметров). Исходными данными служит входной телеметрический поток БМП, который работает с датчиками нескольких типов, в том числе с особым типом акустических датчиков. Поскольку процесс проведения испытаний может сопровождаться различными сбоями и помехами извне, необходима отбраковка этих сбоев, а также заполнение разрывов в данных. Разрывы могут заполняться полиномиальными методами, на основе регрессии и пр. Кроме того, предварительная обработка также включает выделение одиночных сигналов из группового телеметрического сигнала, после чего выбирается тип обработки. Может вестись обработка одного телеметрического канала (предполагается, что передача сигнала производится на расстоянии по телеметрическому каналу связи, в связи с чем предпринимается автоматизированная обработка одного канала или совместная обработка нескольких телеметрических каналов). Среди возможных ее типов может быть обработка во временной, частотной, частотно-временной области, а также различные статистические методы, ручная обработка результатов измерений с последующей интерпретацией специалистами-анализаторами, полуавтоматическая обработка, допусковый контроль параметров измерений и т. д.

При одновременном рассмотрении сигналов от нескольких объектов РКТ целесообразно осуществлять их классификацию, для чего также могут использоваться различные алгоритмы [6, 7] (кластер-анализ – автоматическая классификация, например алгоритмы k-средних, EM-алгоритм, иерархическая классификация и т. д.). Это помогает выполнять их группировку по различным признакам (совокупности признаков). При этом среди наиболее часто используемых классификаций в области РКТ следует выделить *вибрационные процессы* (стационарные и нестационарные), виброударные, акустические и переходные процессы. Они различаются по длительности, типу вибрационного процесса (переходный, ударный, установившийся вибрационный), стационарности (стационарный/нестационарный) и диапазону концентрации энергии в частотной области – вблизи нулевой частоты, полосовая структура, в высокочастотном диапазоне.

На рис. 1 представлена схема обработки быстро меняющихся процессов. Они широко характеризуются спектральной плотностью в частотной области. При этом довольно часто встречаются *многокомпонентные вибрационные сигналы*, в частности многокомпонентные вибрационные сигналы, которые обладают набором аддитивных компонент различной структуры и различных частотных свойств. Компоненты отличаются разномасштабностью (различным частотным разрешением), т. е. каждая компонента имеет собственное частотное разрешение и во многих случаях отдельный физический смысл. Немаловажно то, что их можно разделить на стационарные и нестационарные процессы, при этом нестационарность может быть относительно среднего значения (тренд среднего), дисперсии или других характеристик.

На рис. 2 приведены примеры вычисления спектральной плотности БМП на основе модели

Рис. 1. Схема обработки БМП

с линейно независимыми последовательными по времени состояниями. Наряду с моделью авторегрессии (АР-модель), также хорошо применяются полиномиальная модель, модель на основе полигармонического сигнала. АР-модель имеет ряд преимуществ при ее применении к исходным данным. Во-первых, спектральные оценки являются более гладкими и менее осциллирующими, что удобнее для восприятия соответствующих графиков. Классическая Фурье-периодограмма и ее модификации, такие как периодограмма Даньелла, Бартлетта, Уэлча, периодограмма с окном, часто дают довольно заметную степень изрезанности (как следствие несостоятельности отдельных спектральных оценок, наличия эффекта растекания спектра и пр.). Спектральные оценки на основе АР-модели обеспечивают возможность различения

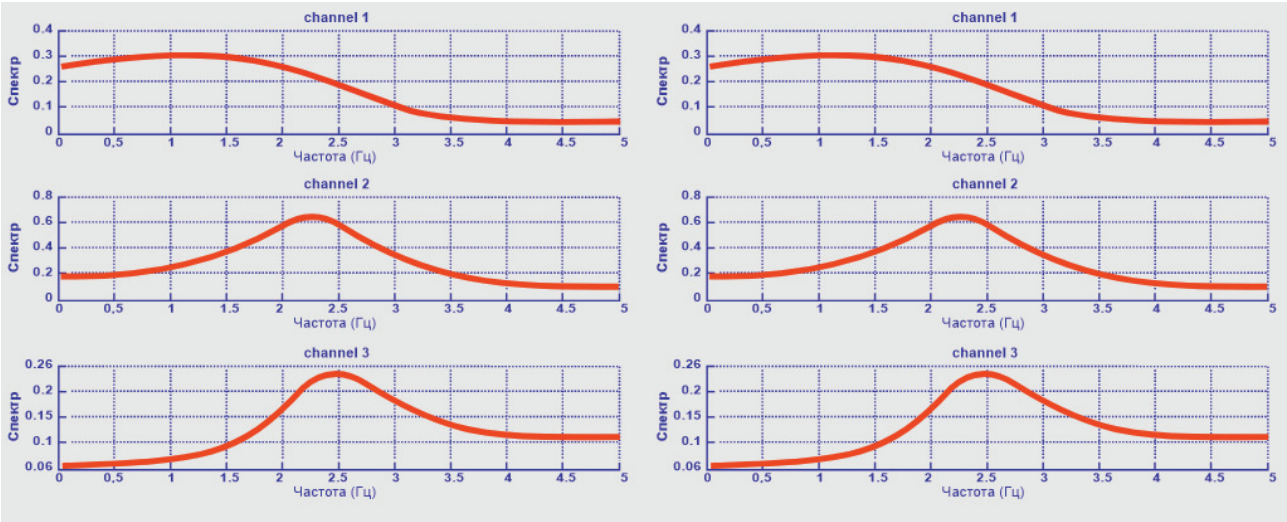


Рис. 2. Спектральная плотность БМП (трехканальный сигнал), рассчитанная на основе АР-модели для случая отсутствия аномалий (слева) и их наличия (справа). Во втором случае используется робастная процедура

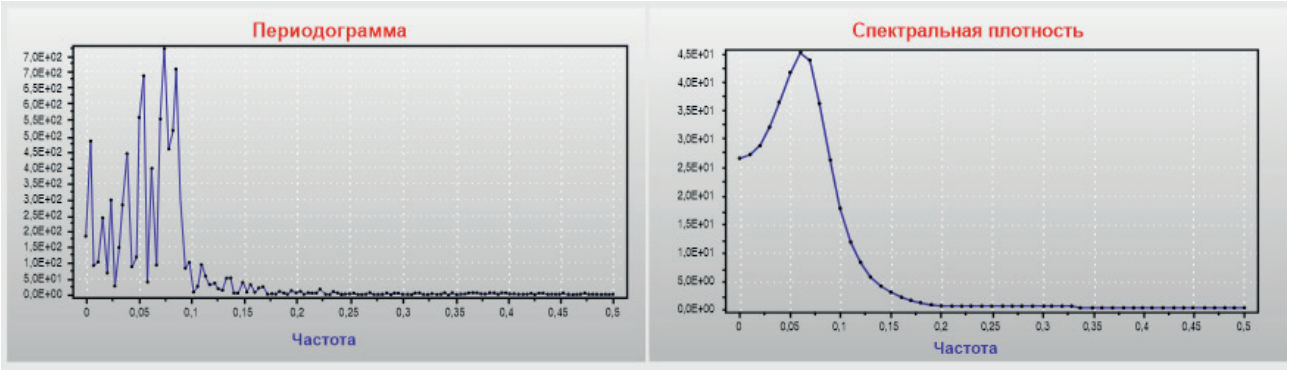


Рис. 3. Спектральная плотность, рассчитанная на основе Фурье-периодограммы (слева) и АР-модели (справа)

пиков с более высоким качеством, а также нет необходимости уменьшать влияние эффекта растекания спектра, который может давать дополнительные побочные спектральные составляющие. Модели авторегрессии скользящего среднего (АРСС) и скользящего среднего (СС) более трудоемки с точки зрения вычислительных затрат и проверки их адекватности исходным данным, при этом также дают хорошее качество при вычислении оценок спектральной плотности. Робастные процедуры направлены на то, чтобы дополнительно повысить качество оценки спектральной плотности мощности. В сигналах могут присутствовать аномалии, вызванные различными сбоями в работе объектов в ходе проведения испытаний.

На рис. 3 показаны примеры вычисления периодограммы исходного вибрационного сигнала на основе преобразования Фурье (слева) и его спектральная плотность (справа), полученная путем применения сглаженной спектральной оценки на основе АР-модели. Из графиков видно, что АР-модель обеспечила более высокую степень сглаженности и при этом сохранился диапазон концентрации энергии.

Разработанный нами алгоритм оценивания спектральной плотности вибрационных процессов основан на применении вейвлет-сглаживания [8] и Фурье-периодограммы и позволяет оценить СПМ квазистационарных установившихся вибрационных процессов и энергию механических вибраций в полосах частот. Часто применяются полуоктавные, октавные или третьоктавные полосы, для чего представление может осуществляться в логарифмическом масштабе. Данный алгоритм является альтернативным по отношению к классическим методам Фурье-анализа, а также параметрическим методам. Сама Фурье-периодограмма может выступать в качестве исходной информации для дальнейшего улучшенного оценивания спектральной плотности мощности, при этом используется свойство четности классической спектральной оценки. Такого рода ситуация, когда исходная периодограмма выступает в качестве априорной информации для оценивания параметров вибрационного сигнала, позволяет передавать заведомо больший объем информации. Немаловажно, что вейвлет-преобразование имеет апробированные быстрые вычислительные алгоритмы (например, быстрый алгоритм Малла), что существенно ускоряет вычисления при больших объемах данных (до нескольких сотен тысяч или миллионов отсчетов).

Исходно имеем установившийся вибрационный процесс $s_{\text{УВП}}(n)$ длины N. При этом длина должна быть целой степенью двойки, что необходимо для эффективного применения быстрого вейвлет-преобразования в дальнейшем. Иначе сигнал дополняется необходимым количеством нулей или же выполняется его экстраполяция. Установившиеся вибрационные процессы являются квазистационарными при рассмотрении реализации полностью либо стационарными на отдельных участках.

Вначале вычисляется Фурье-периодограмма $W_N(k)$ установившегося вибрационного процесса. При достаточно больших значениях N справедливо следующее математическое соотношение [9, 10]:

$$W_N(k) = S(k)u(k), \qquad k = 0,1,...,N, \qquad (1)$$

где k – номер спектрального отсчета, $W_N(k)$ – вычисленная Фурье-периодограмма, $S(k)$ – искомая СПМ, $u(k)$ – случайная составляющая. При $k = 1,...,N-1$ величина $u(k)$ имеет одностороннее экспоненциальное распределение с параметром 1, а при $k = 0$ и $k = N$ величина $u(k)$ имеет распределение χ^2 с одной степенью свободы.

Данное соотношение можно записать иначе путем логарифмирования обеих частей и, та-

ким образом, преобразования его в более удобную для дальнейшей работы аддитивную форму:

$$\ln W_N(k) = \ln S(k) + \ln u(k), \quad k = 0,1,...,N. \quad (2)$$

Дальнейшие математические преобразования связаны с использованием математического ожидания случайной составляющей и введением новой величины $\varepsilon(k)$ со статистическим распределением:

$$\ln W_N(k) = \ln S(k) + \varepsilon(k) + E[\ln u(k)], \quad k = 0,1,...,N, \quad (3)$$

где

$$\varepsilon(k) = \ln u(k) - E[\ln u(k)], \quad k = 0,1,...,N. \quad (4)$$

Величины $\varepsilon(0)$ и $\varepsilon(N)$ имеют распределение, совпадающее с распределением величины $\varepsilon(k)$ при $0 < k < N$, при больших N значениями $\varepsilon(0)$ и $\varepsilon(N)$ можно пренебречь.

Выполненные преобразования позволяют записать выражение в виде

$$\ln W_N(k) + \gamma = \ln S(k) + \varepsilon(k), \quad k = 0,1,...,N, \quad (5)$$

где $\varepsilon(k)$ – случайная величина с нулевым средним значением, γ – константа Эйлера.

Имея оценку логарифмической периодограммы $\ln W_N(k)$, можно вычислить ее вейвлет-коэффициенты, например с помощью дискретного вейвлет-преобразования:

$$b_j(m) = \sum_{k=0}^{N-1} (\ln W_N(k) + \gamma) w_j((k - 2^i m)_{\bmod N}), \quad (6)$$

$$u_j(m) = \sum_{k=0}^{N-1} \ln S(k) w_j((k - 2^i m)_{\bmod N}), \quad (7)$$

$$y_j(m) = \sum_{k=0}^{N-1} \varepsilon(k) w_j((k - 2^i m)_{\bmod N}), \quad (8)$$

где величины $w_j(k)$ используются для обозначения базисного вейвлета на масштабе j, m – параметр сдвига, $y_j(m)$ – вейвлет-коэффициенты случайной величины $\varepsilon(k)$.

Поскольку вейвлет-преобразование обладает свойством линейности, можно записать:

$$b_j(m) = u_j(m) + y_j(m). \quad (9)$$

Дальнейшее сглаживание вейвлет-коэффициентов (по аналогии с задачей очистки сигналов от шума на основе вейвлетов) выполняется с помощью жесткой пороговой обработки. Данную модификацию можно в общем виде записать следующим образом:

$$\tilde{b}_j(m) = \begin{cases} b_j(m), & |b_j(m)| > \rho_j \\ 0, & |b_j(m)| \le \rho_j \end{cases}, \quad (10)$$

где $\tilde{b}_j(m)$ – модифицированные вейвлет-коэффициенты после проведения жесткой пороговой обработки, ρ_j – пороговые значения.

В случае отсутствия локальных особенностей в виде резонансных пиков целесообразно применение мягкой пороговой обработки:

$$\tilde{b}_j(m) = \begin{cases} b_j(m) - \rho_j, & b_j(m) > \rho_j \\ 0, & -\rho_j < b_j(m) \le \rho_j \\ b_j(m) + \rho_j, & b_j(m) \le -\rho_j \end{cases}. \quad (11)$$

Пороги ρ_j зависят от номера уровня вейвлет-разложения:

$$\rho_j = \alpha_j \ln \frac{N}{2}. \quad (12)$$

Коэффициенты α_j табулированы для ряда вейвлет-базисов (койфлеты, вейвлеты Добеши, симлеты). Жесткая пороговая обработка гораздо лучше подходит в случае наличия резонансных пиков, в то время как мягкая пороговая обработка более целесообразна в том случае, если спектральная плотность не имеет подобных особенностей (является гладкой функцией). Жесткая пороговая обработка вейвлет-коэффициентов хорошо подходит для многокомпонентного вибрационного сигнала в частотной области в сравнении с другими видами пороговой обработки.

Далее применяется обратное дискретное преобразование Фурье к модифицированным вейвлет-коэффициентам. В результате формируется оценка модифицированной логарифмической периодограммы.

После проведения сглаживания оценка искомой СПМ установившегося вибрационного процесса $S_{\text{УВП}}(n)$ определяется по формуле

$$\hat{S}(k) = e^{\ln \widetilde{W}_N(k) + \gamma}. \quad (13)$$

Этот результат и есть сглаженная спектральная оценка, и в дальнейшем его можно применять для детального анализа структуры СПМ с целью изучения структуры и различных спектральных особенностей, например областей локализации энергии и пр.

Библиографический список

1. Теоретические основы испытаний и экспериментальная отработка сложных технических систем: учеб. пособие / Л. Н. Александровская [и др.]. М.: Логос, 2003. 736 с.

2. Элементы теории испытаний и контроля технических систем / В. И. Городецкий [и др.]; под ред. Р. М. Юсупова. Л.: Энергия, 1978. 192 с.

3. Испытания жидкостных ракетных двигателей / А. Е. Жуковский [и др.]; под ред. проф. В. Я. Левина. М.: Машиностроение, 1981. 201 с.

4. *Луарсабов К. А., Пронь Л. В., Сердюк А. В.* Летные испытания жидкостных ракетных двигателей. М.: Машиностроение, 1977. 192 с.

5. Современная телеметрия в теории и на практике. Учебный курс / А. В. Назаров [и др.]. – СПб.: Наука и техника, 2007. 672 с.

6. *Жукова Н. А., Тристанов А. Б.* Использование алгоритмов кластеризации и классификации для обработки телеметрической информации // Научная сессия МИФИ-2007, г. Москва, 22–26 янв. 2007 г.: сб. науч. тр. М., 2007. С. 184–185.

7. Анализ состояния сложных динамических объектов с использованием алгоритмов интеллектуальной обработки данных / А. В. Васильев [и др.] // Технологии Microsoft в теории и практике программирования: тез. докл. Междунар. научн.-техн. конф., Спб., 13–14 марта 2007 г. СПб., 2007. С. 88–90.

8. *Чуи К.* Введение в вейвлеты: пер. с англ. М.: Мир, 2001. 412 с.

9. *Клионский Д. М., Орешко Н. И., Геппенер В. В.* Оценивание спектральной плотности телеметрических данных на основе теории // Цифровая обработка сигналов и ее применения – DSPA'2011: тез. докл. 13-й Междунар. конф., г. Москва, 30 марта – 1 апр. 2011 г. М., 2011. С. 123–127.

10. *Бриллинджер Д.* Временные ряды. Обработка данных и теория: пер. с англ. М.: Мир. 1980. 536 с.

УДК 621.394.147 **DOI: 10.31799/978-5-8088-1701-2-2022-2-317-324**

В. А. Кузнецов
кандидат технических наук, доцент
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ВЕКТОРИЗАЦИЯ АЛГОРИТМА ДЕКОДИРОВАНИЯ БЛОКОВОГО ТУРБОКОДА ПРОИЗВЕДЕНИЯ ПУТЕМ ПОЛНОГО ПЕРЕБОРА КОДОВЫХ СЛОВ

При достаточно высоком количестве алгоритмов помехоустойчивого кодирования в современных комплексах связи часто используются компромиссные между быстродействием и корректирующей способностью решения. Сегодня разработка быстрых алгоритмов помехоустойчивого кодирования – задача значительно более сложная, чем оптимизация уже существующих с использованием более мощных вычислительных ресурсов. Объект исследования – блоковые турбокоды произведения. Предмет исследования – реализация алгоритма полного перебора кодовых слов для вычисления информации для расчета критерия правдоподобия одной кодовой строки. Исследована возможность ускорения времени работы декодера на основе полного перебора кодовых слов. В качестве примера при построении блокового кода выбран расширенный код Хэмминга (16,11). Цель работы – разработка оптимизированных алгоритмов декодирования блоковых турбокодов, адаптированных к архитектуре векторных команд. Использованы векторные команды для процессоров Intel AVX (Advanced Vector Extensions) версий 2.0 и выше. Инструкции AVX2.0 на данный момент присутствуют на широкодоступных недорогих версиях процессоров Intel в отличие от более новых инструкций версии AVX512. Для исследуемых и разработанных алгоритмов представлены оценка времени и ошибок декодирования. Также приведен анализ используемых векторных инструкций.
Ключевые слова: *блоковые турбокоды, таблица кодовых слов, код Хэмминга, векторные команды.*

V. A. Kuznetsov
PhD, Tech., Associate Processor
St. Petersburg State University of Aerospace Instrumentation

TURBO CODES BRUTE-FORCE-DECODING OPTIMIZATION USING VECTOR INSTRUCTIONS

In spite of high number of techniques for decoding error-correcting codes, in modern communication systems conventionally used solutions based on the compromise between speed and corrective ability. Today, the development of fast error-correcting coding algorithms is a much more complicated task than the optimization of existing ones using more powerful computing resources. The object of the research are the product codes built using linear block codes. The subject of research is brute-force-algorithm of estimation of the log-likelihood ratio (LLR) of the binary decisions given by the decoder. The reduction capabilities of time required for the decoding based on a complete search in codewords table is investigated. As an example, the extended Hamming code (16, 11) was considered. The aim of the work presented in the paper is the development of optimized decoding algorithms for block turbo codes adapted to the architecture of vector instructions. In this work, vector instructions AVX 2.0, supporting by Intel processors, were used. AVX 2.0 Instructions are currently present on widely available, not expensive Intel processors, unlike the AVX 512 instructions. For all algorithms, an estimate of the time and decoding errors is presented. An analysis of using vector instructions is also presented.
Keywords: *block turbo codes, codewords table, Hamming code, vector instructions.*

Введение

Векторизация вычислительных процессов – активно развивающееся направление во многих отраслях промышленности. Большое количество библиотек обработки данных реализованы с использованием расширенного набора команд процессора. Для методов, имеющих широкий спектр применения, например преобразование Фурье, оптимизация алгоритма и адаптация его к определенной конфигурации вычислительного оборудования – задача в большинстве случаев решенная. Для некоторых методов применение векторных команд требует значительной проработки как самого алгоритма, так и структуры входных данных, в других случаях решение осложнено тем, что предлагаемые векторные операции процессора не ориентированы на решение поставленных задач.

Блоковые турбокоды произведения являются помехоустойчивыми кодами с корректирующей способностью, зависящей от комбинации используемых базовых кодов. Они строятся на основе блоковых кодов, для которых могут применяться алгоритмы декодирования как с жестким, так и с мягким решением [1, 2].

В статье рассматривается пример декодирования строки блокового турбокода произведения с мягким решением [1]. За основу взята система обозначений и схема декодирования алгоритма Чейза – Пиндиаха [3–6], в которой процедура поиска весовых коэффициентов (дополнительной информации) реализована путем полного перебора таблицы кодовых слов. Тем самым обеспечивается максимум правдоподобия, в то время как оригинальные алгоритмы [3–5] не позволяют его получить, но требуют значительно меньше вычислительных ресурсов. В качестве базового кода выбран расширенный код Хэмминга (16,11).

Декодирование турбокодов

Каждое измерение блока турбокода рассматривается как множество кодовых слов, сформированных с использованием выбранного блокового кода. В работе исследуются блоковые коды, описываемые следующим набором параметров: длины составляющих кодов символами n, числа информационных символов этих кодов k, минимальное расстояние Хэмминга между кодовыми словами d. Кодирование происходит путем формирования проверочных символов, добавляемых в конец или начало результирующего кодового слова.

При построении турбокодов произведения используется любая комбинация блоковых кодов, а один кодовый блок удобно представлять в виде многомерной структуры.

Для вычисления надежности найденного кодового слова в одном измерении для каждого кодового слова итерации в алгоритме Пиндиаха [3–5] требуется осуществить поиск двух кодовых слов: мягкого решения D и конкурирующего слова C из допустимых для данного кода. Мягкое решение D соответствует максимуму скалярного произведения или минимуму евклидова расстояния от принятой последовательности R_i. Конкурирующее слово – наближайшее к принятой последовательности, которое отличается в текущей битовой позиции от мягкого решения D.

Результатом для каждого входного слова в итерации (полуитерации) является кодовое слово, такое что

$$r'_j = \frac{\left(\left|R-C\right|^2 - \left|R-D\right|^2\right)}{4} d_j, \qquad (1)$$

где j – позиция символа. Вес (дополнительная информация) символа согласно [3]:

$$w_j = \frac{\left(\left|R-C\right|^2 - \left|R-D\right|^2\right)}{4} d_j - r_j.$$

При определении расстояний или скалярных произведений наиболее трудоемкая задача – перебор кодовых слов с целью нахождения наближайшего слова и конкурирующих слов для каждой позиции бита. Само конкурирующее слово для вычисления (1) не требуется, необходимо определение расстояния до наближайшего слова, содержащего в текущей позиции отличный символ; другими словами, для вычисления (1) достаточно найти два значения для каждой позиции: минимум расстояния, если +1 в текущей позиции $\max_j^{+1}$, и минимум расстояния, если –1 в текущей позиции $\max_j^{-1}$, которые будут соответствовать $\left|R-C\right|^2$ и $\left|R-D\right|^2$ в зависимости от символа в текущей позиции.

Декодирование принятой последовательности в [3–5] осуществляется по алгоритму Чейза. Согласно [1], вычисление мягкого решения D может быть произведено тремя способами, в дальнейшем именуемыми первый, второй и третий алгоритмы Чейза в соответствии с нумерацией, принятой в опорной статье. Общий принцип мягкого декодирования Чейза заключается в установлении возможных кодовых слов на основе определенного правила и нахождение наближайшего слова к принятой последовательности.

Первый алгоритм Чейза предполагает набор кодовых слов для перебора, построенный при всех возможных вариантах расположения $(d-1)$ ошибок.

Второй алгоритм Чейза предполагает набор кодовых слов для перебора, построенный при всех вариациях значений символов в некоторых L наименее надежных позициях, называемых слабыми позициями и имеющих наименьшее значение по модулю, бит четности в рассмотрении не участвует.

Очевидно, что алгоритм полного перебора кодовых слов применим только при небольшом размере таблиц кодовых слов, так, согласно табл. 1, для кода Хэмминга (32, 26) потребуется

Таблица 1

Количество слов для перебора в различных алгоритмах

Код	Полный перебор КС	1-й алгоритм Чейза	2-й алгоритм Чейза, $L = 3$	2-й алгоритм Чейза, $L = 8$	2-й алгоритм Чейза, $L = k - 1$	Сокр. перебор, вес 4	Сокр. перебор, вес 4,6
Код Хэмминга (16, 11)	2048	697	8	256	1024	141	589
Код Хэмминга (32, 26)	67108864	41448	8	256	33554432	1241	28993

хранить 67108864 слов по 32 бит, что соответствует 268 Мб.

Таблица кодовых слов расширенного кода Хэмминга (16, 11) состоит из $2^{11} = 2048$ элементов. Для каждого кодового слова кода Хэмминга (16, 11) в ней есть одно слово с весом 0, 140 кодовых слов с весом 4, 448 – с весом 6, 870 – с весом 8, 448 – с весом 10, 448 – с весом 12 и одно кодовое слово с весом 16.

Количество кодовых слов для перебора можно сократить, исходя из того факта, что конкурирующие слова имеют ограниченное расстояние от входного кодового вектора. Тем самым достаточно рассматривать только некоторые из них. Для этого требуется произвести сортировку подтаблиц кодовых слов. Ключом для выбора подтаблицы является результат жесткого декодирования входного вектора. Использование большого количества подтаблиц приводит к значительному увеличению объема памяти, требующегося для хранения таблицы декодера.

Количество слов для перебора в первом алгоритме Чейза будет определяться из количества расположений в векторе d–1 ошибок. Для кода Хэмминга (16, 11) 1, 2 и 3 ошибок, что может быть вычислено как сумма сочетаний 16 по 1, 16 по 2 и 16 по 3, что равно 697.

Количество слов для перебора во втором алгоритме Чейза может быть вычислено как 2^L.

Обзор векторных команд AVX2.0

Для работы с векторными командами использована библиотека интринсик-функция immintrin.h, позволяющая применять векторные команды и регистры аналогично базовым функциям и типам языка С.

Осуществляется обработка 32 битных чисел float, один регистр AVX версии 2.0 имеет размер 256 бит и содержит 8 чисел float. В библиотеке immintrin.h тип регистра, состоящего из 8 чисел float обозначается __m256 (половина этого регистра __m128), для указания операций, связанных с этим типом, введено окончание команды _ps. Для использования доступны 32 регистра.

При реализации алгоритма декодирования применялись функции библиотеки immintrin.h, приведенные в табл. 2. Предварительно осуществлено тестирование каждой функции, и,

Таблица 2

Команды библиотеки immintrin.h [7], используемые для реализации алгоритма декодирования

Команда	Прототип интринсик-функции	Описание команды
Инициализация и работа с памятью		
vmovupsymm, m256	__m256 _mm256_loadu_ps (float const * mem_addr)	Загрузка 256 бит из памяти по указанному адресу в регистр назначения
vmovaps m256, ymm	void _mm256_store_ps (float * mem_addr, __m256 a)	Запись 256 бит из указанного регистра в память по указанному адресу
vxorpsymm, ymm, ymm	__m256 _mm256_setzero_ps (void)	Возвращает регистр с нулевыми значениями
Логические операции		
vxorpsymm, ymm, ymm	__m256 _mm256_xor_ps (__m256 a, __m256 b)	Операция поразрядного XOR для пар значений во входных регистрах и сохранение результата в возвращаемый регистр
vandpsymm, ymm, ymm	__m256 _mm256_and_ps (__m256 a, __m256 b)	Операция поразрядного AND для пар значений во входных регистрах и сохранение результата в возвращаемый регистр

Окончание табл. 2

Команда	Прототип интринсик-функции	Описание команды
vpslldymm, ymm, imm8	__m256i _mm256_slli_epi32 (__m256i a, int imm8)	Логический сдвиг на указанное количество бит значений регистра и сохранение результата в возвращаемый регистр. Использовалось приведение типов при передаче данных из регистров типа __m256
Арифметические операции		
vmulpsymm, ymm, ymm	__m256 _mm256_mul_ps (__m256 a, __m256 b)	Попарное умножение значений из входных регистров и сохранение результата в возвращаемый регистр
vaddpsymm, ymm, ymm	__m256 _mm256_add_ps (__m256 a, __m256 b)	Попарное сложение значений из входных регистров и сохранение результата в возвращаемый регистр
vhaddpsymm, ymm, ymm	__m256 _mm256_hadd_ps (__m256 a, __m256 b)	Горизонтальное сложение пар значений из входных регистров и сохранение результата в возвращаемый регистр. Порядок записи в возвращаемый регистр: a_0+a_1, a_2+a_3, b_0+b_1, b_2+b_3, a_4+a_5, a_6+a_7, b_4+b_5, b_6+b_7. Для суммирования всех элементов в одном регистре требуются три операции hadd, при этом необходимо добавлять новые или нулевые данные во второй регистр и менять порядок следования элементов во входных регистрах
vmaxpsymm, ymm, ymm	__m256 _mm256_max_ps (__m256 a, __m256 b)	Попарное сравнение значений из входных регистров и сохранение наибольшего в возвращаемый регистр
Перестановки и работа с данными		
vperm2f128 ymm, ymm, ymm, imm8	__m256 _mm256_permute2f128_ps (__m256 a, __m256 b, int imm8)	Смешивание блоков по 4 элемента из входных регистров в возвращаемом регистре по правилу, заданному двумя 4-битными значениями, хранящимися в третьем параметре
vblendvpsymm, ymm, ymm, ymm	__m256 _mm256_blendv_ps (__m256 a, __m256 b, __m256 mask)	Смешивание элементов из входных регистров в возвращаемом регистре по правилу, заданному в знаковых битах третьего регистра
vblendpsymm, ymm, ymm, imm8	__m256 _mm256_blend_ps (__m256 a, __m256 b, constint imm8)	Смешивание элементов из входных регистров в возвращаемом регистре по правилу, заданному в статической битовой маске
vbroadcastssymm, xmm	__m256 _mm256_broadcastss_ps (__m128 a)	Множественная запись значения из нижнего разряда указанного регистра во все позиции возвращаемого регистра
	__m256i _mm256_srli_si256 (__m256, int imm8)	Байтовый сдвиг внутри регистра
Преобразование типов		
–	__m128 _mm256_castps256_ps128 (__m256 a)	Преобразование типа регистра. Не является командой. Используется для возможности применить функцию _mm_broadcastss_ps

*Тип int imm8 представляет собой статическую целочисленную маску размером 32 бит.

если возможно, осуществлялось сравнение выполняемой процедуры с аналогом без использования векторных команд.

Некоторые интринсик-функции, например _mm256_broadcastss_ps и _mm_broadcastss_ps, являются эмуляцией одной и той же команды для разных типов, некоторые представлены не для всех типов, что усложняет подбор функций и требует, в сравнении с ассемблерным кодом, лишних преобразований.

Согласно алгоритму вычисления, представленному ниже, наиболее частой операцией оказывается горизонтальное суммирование элементов. Горизонтальное суммирование всех элементов двух регистров с помощью функции _mm256_hadd_ps требует дополнительных опе-

раций и может быть представлено следующим образом:

```
avxREG2 = _mm256_hadd_ps( avxREG2 , avxREG1 );
avxREG1 = _mm256_permute2f128_ps( avxREG2, avxZEROREG, 0x20 );
avxREG2 = _mm256_permute2f128_ps( avxREG2, avxZEROREG, 0x31 );
avxREG2 = _mm256_hadd_ps( avxREG1 , avxREG2 );
avxREG2 = _mm256_hadd_ps( avxREG2 , avxZEROREG );
avxREG2 = _mm256_hadd_ps( avxREG2 , avxZEROREG ),
```

где регистр нулей avxZEROREG может быть заменен функцией _mm256_setzero_ps(). Очевидно, что три последние операции складывают добавочные нули, в связи с этим в алгоритме 2 используется последовательное добавление кодовых слов из таблицы в регистры для горизонтального суммирования, а в алгоритме 3 горизонтальное суммирование не применяется.

Описание алгоритма полного перебора кодовых слов с использованием AVX2.0

Алгоритм вычисления дополнительной информации с полным перебором кодовых слов (в дальнейшем алгоритм 1) состоит из следующих операций.

1. Загрузка входного кодового вектора (float) в AVX-регистры. Всего на одну последовательность требуется $n\%n_{reg}$ регистров и операций _mm256_loadu_ps. Возможно сохранение дополнительно инвертированного кодового вектора для ускорения последующих операций.

2. Цикл из $N_{cw} / 32$ итераций.

1.1. Загрузка части таблицы кодовых слов декодера, сформированных поразрядно в $n\%n_{reg}$ регистров AVX. Требуется $n\%n_{reg}$ регистров и операций _mm256_loadu_ps. Каждая группа операций загружает в регистр 32 кодовых слова.

1.2. Цикл из 32 итераций.

1.2.1. Инверсия по маске знаковых битов входного кодового вектора. $n\%n_{reg}$ операций _mm256_blendv_ps.

1.2.2. Перестановка частей регистров для дальнейшего горизонтального суммирования. $n\%n_{reg}$ операций _mm256_permute2f128_ps.

1.2.3. Горизонтальное суммирование для вычисления скалярного произведения. Четыре операции _mm256_hadd_ps. Для конвейерного суммирования: каждое кодовое слово требует две операции hadd, кроме: –1 первого и +3 последних полупустых had.

1.2.4. Запись первого элемента в 8 позиций регистра. Одна операция _mm256_broadcastss_ps.

1.2.5. Обнуление нулевой позиции для конвейерного суммирования.

1.2.6. Заполнение регистров для сравнения с максимумом для нуля, $n\%n_{reg}$ операций _mm256_blendv_ps.

1.2.7. Сравнения с максимальными значениями для нуля в каждой позиции, $n\%n_{reg}$ операций _mm256_max_ps.

1.2.8. Заполнение регистров для сравнения с максимумом для единицы, $n\%n_{reg}$ операций _mm256_blendv_ps.

1.2.9. Сравнения с максимальными значениями для единицы в каждой позиции, $n\%n_{reg}$ операций _mm256_max_ps.

1.2.10. Логический сдвиг кодовых слов таблицы декодера, перенос следующего кодового слова в знаковые разряды, $n\%n_{reg}$ операций _mm256_slli_epi32.

2. Вычисление дополнительной информации на основе максимальных значений для нуля и единицы в каждой позиции.

Количество необходимых операций для алгоритма 1 представлено в табл. 3.

Таблица 3

Количество операций для алгоритма 1

Операция	Общее количество операций	Количество операций для расширенного кода Хэмминга (16, 11)
_mm256_loadu_ps	$n\%n_{reg} (1 + N_{cw} / 32)$	130
_mm256_blendv_ps	$3\ n\%n_{reg} N_{cw}$	12288
_mm256_permute2f128_ps	$n\%n_{reg} N_{cw}$	4096
_mm256_hadd_ps	$4\ n\%n_{reg} N_{cw}$	16384
_mm256_broadcastss_ps	N_{cw}	2048
_mm256_max_ps	$2\ n\%n_{reg} N_{cw}$	8192
_mm256_slli_epi32	$n\%n_{reg} N_{cw}$	4096

Таблица 4

Пример упорядоченного блока 32 таблицы кодовых слов в регистрах

Регистр	Элемент	Содержимое	Элемент	Содержимое
AVX Регистр 256	$PCK_{CW1}[0]$	11111111111111111111111111111111	$TRP_{CW1}[0]$	11111111111111111111111011100000
	$PCK_{CW1}[1]$	11111111111111111111111111111111	$TRP_{CW1}[1]$	11111111110011001111111011010011
	$PCK_{CW1}[2]$	11111111111111111111111111111111	$TRP_{CW1}[2]$	11111111101110001111111010100111
	$PCK_{CW1}[3]$	11111111111111111111111111111111	$TRP_{CW1}[3]$	11111111100010111111111010010100
	$PCK_{CW1}[4]$	11111111111111111111111111111111	$TRP_{CW1}[4]$	11111111011100011111111001101110
	$PCK_{CW1}[5]$	11111111111111111111111111111111	$TRP_{CW1}[5]$	11111111010000101111111001011101
	$PCK_{CW1}[6]$	11111111111111110000000000000000	$TRP_{CW1}[6]$	11111111001101101111111000101001
	$PCK_{CW1}[7]$	11111111000000001111111100000000	$TRP_{CW1}[7]$	11111111000001011111111000011010
AVX Регистр 256	$PCK_{CW2}[0]$	11110000111100001111000011110000	$TRP_{CW2}[0]$	11111101111000111111110011111100
	$PCK_{CW2}[1]$	11001100110011001100110011001100	$TRP_{CW2}[1]$	11111101110100001111110011001111
	$PCK_{CW2}[2]$	10101010101010101010101010101010	$TRP_{CW2}[2]$	11111101101001001111110010111011
	$PCK_{CW2}[3]$	10101010010101010101010110101010	$TRP_{CW2}[3]$	11111101100101111111110010001000
	$PCK_{CW2}[4]$	11110000000011110000111111110000	$TRP_{CW2}[4]$	11111101011011011111110001110010
	$PCK_{CW2}[5]$	11000011001111000011110011000011	$TRP_{CW2}[5]$	11111101010111011111110001000001
	$PCK_{CW2}[6]$	10010110011010011001011001101001	$TRP_{CW2}[6]$	11111101001010101111110000110101
	$PCK_{CW2}[7]$	10011001011001101001100101100110	$TRP_{CW2}[7]$	11111101000110011111110000000110

Для декодирования одной строки полуитерации турбокода на основе кода Хэмминга (16, 11) задействовано 12 регистров, всего потребуется 47234 операций. Из табл. 3 видно, что наибольшее количество операций требуется для горизонтального суммирования.

Количество горизонтальных суммирований может быть сокращено до $(n\%n_{reg}N_{cw}+2)$, но в этом случае будет производиться $(n\%n_{reg}N_{cw})$ обнулений старшего числа конвейера суммы и $(2\ n\%n_{reg}N_{cw})$ копирований регистров перед каждым сдвигом таблицы кодовых слов, что потребует $2\ n\%n_{reg}$ дополнительных регистров (в дальнейшем алгоритм 2). Копирование регистров может быть заменено $(2\ n\%n_{reg}N_{cw})$ временными логическими сдвигами без необходимости задействовать дополнительные регистры. При одинаковом количестве операций такой подход позволил сократить затраченное время.

Для устранения большого количества дополнительных операций, связанных с горизонтальным суммированием, вычисление скалярного произведения может быть осуществлено с использованием команды vaddps (в дальнейшем алгоритм 3). Данные в блоках таблицы кодовых слов были «транспонированы» таким образом, что два кодовых слова находятся каждый в половине одного 256-битного регистра и вычисление скалярного произведения организовано для 8 кодовых слов одновременно. Пример хранения блока таблицы из 32 кодовых слов для алгоритмов 1, 2 и 3 представлен в табл. 4. Выделено одно и то же кодовое слово. Для алгоритма 3 необходимо использовать обе таблицы PCK и TRP, так как для сравнения с минимумом требуется таблица PCK-представлений кодовых слов.

Так как за одну операцию обрабатывается 8 бит кодовых слов, а в коде Хэмминга 11 информационных символов, то в таблице всех кодовых слов можно упорядочить кодовые слова таким образом, что первые 8 бит идущих подряд кодовых слов будут одинаковы, таблицы кодовых слов можно сжать, при этом сократив количество операций загрузки данных в регистры. Отброшенные избыточные биты представлены в табл. 5.

Для каждого кодового слова в таблице кодовых слов существует инвертированное кодовое слово, для поиска максимального значения достаточно сравнить абсолютное значение вычисленного скалярного произведения.

Изменение 8-го бита инвертирует все проверочные биты (так как была для этого бита использована строка в порождающей матрице: 1111), из чего следует, что можно рассматривать попарно блоки по 8 кодовых слов, инвертируя суммы последних 5 бит. По этой причине можно сократить содержимое TRP_{CW1} еще вдвое.

Размер таблицы из 1024 CW составляет 32,768 Кб. Сжатая таблица PCK занимает 3,2 Кб памяти. Для алгоритма 3 требуется также транспонированное TRP-представление 9,2 Кб.

Таблица 5

Исключение избыточных бит из таблицы кодовых слов

Регистр	Элемент	Содержимое	Элемент	Содержимое В TRP_{CW1} представлена половина бит
AVX Регистр 256	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	11111111 11111111 11111110 11100000
	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	..... 11001100 11010011
	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	..... 10111000 10100111
	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	..... 10001011 10010100
	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	..... 01110001 01101110
	PCK_{CW1}	1..... 1..... 1..... 1.....	TRP_{CW1}	..... 01000010 01011101
	PCK_{CW1}	1..... 1..... 0..... 0.....	TRP_{CW1}	..... 00110110 00101001
	PCK_{CW1}	1..... 0..... 1..... 0.....	TRP_{CW1}	..... 00000101 00011010

Таблица 6

Результаты тестирования декодирования одной строки блока турбокода для расширенного кода Хэмминга (16, 11, 4)

Параметр	Полный перебор без AVX2.0	Алгоритм 1	Алгоритм 2	Алгоритм 3
T, тактов	3120348	245348	205136	173856
Доля от полного перебора без AVX, %	100	7,9	6,5	5,6

Ускорение процедуры вычисления суммы фактически в 51,4 раза (5 вместо 32 операций сложения, также с учетом работы инструкций по 8 операций сложения), ускорение процедуры сравнения в 16 раз.

Результаты тестирования алгоритмов

Для тестирования алгоритмов применяется разница между значениями счетчика тактов процессора, полученного командой rdtsc. Использовалась выборка размерностью 100, в которой отброшены 25 максимальных значений. Сравнительные результаты тестирования алгоритмов 1, 2, 3 и алгоритма полного перебора без использования векторных инструкций (поэлементные операции) представлены в табл. 6.

Заключение

Полученные результаты на примере алгоритма полного перебора демонстрируют возможность значительного уменьшения операции декодирования турбокодов. Для алгоритмов, не использующих таблицы кодовых слов, помимо представленных подходов, требуется выполнение операции декодирования для каждого кодового слова с умножением на вектор ошибок. Наиболее быстро операция декодирования осуществляется при использовании таблиц синдромов [8]: вычисление синдрома путем матричного умножения и исправление вектором ошибок, определяемым из таблицы синдромов. Первый алгоритм Чейза может использовать фиксированную таблицу векторов ошибок, в таком случае можно рассматривать хранение таблицы векторов ошибок аналогично таблице кодовых слов алгоритма 1. Тогда первый алгоритм Чейза будет содержать следующие дополнительные операции для каждого пробного кодового слова.

Для каждого из N_{cw} кодовых слов:

1) умножение демодулированного входного кодового слова на вектор ошибок, $n\%n_{reg}$ операций _mm256_xor_ps;

2) вычисление синдрома;

3) умножение на вектор ошибок таблицы синдромов, $n\%n_{reg}$ операций _mm256_xor_ps, _mm256_loadu_ps.

Второй алгоритм Чейза требует также осуществления операций поиска слабых позиций, формирования или поиска таблицы векторов ошибок, для варианта с множеством предварительно сформированных подтаблиц.

Библиографический список

1. *Chase D.* A class of algorithms for decoding block codes with channel measurement information // IEEE Trans. Inform. Theory. 1972. IT-18. P. 170–182.

2. *Hamming R. W.* Error detecting and error correcting codes // The Bell System Technical Journal. 1950. Vol. 29, № 2. P. 147–160.

3. *Pyndiah R. M.* Near Optimum Decoding of Product Codes: Block Turbo Codes // IEEE Trans. Commun. 1998. Vol. 46. P. 1003–1010.

4. Near optimum decoding of product codes / R. Pyndiah, A. Glavieux, A. Picart, S. Jacq // Proc. of GLOBECOM. 1994. Vol. 1. P. 339–343.

5. *Goalic A., Pyndiah R.* Real-time turbo decoding of product codes on a digital signal processor // GLOBECOM 97. IEEE Global Telecommunications Conference. Conference Record. 1997. P. 267–270.

6. *Зайцев Г. В., Лутков А. Н.* Анализ эффективности турбодекодирования кодов произведения при различных вариантах обмена информацией между итерациями // Цифровая обработка сигналов. 2016. № 2. С. 3–8.

7. Руководство по инструкциям эмулятора. URL: https://software.intel.com/sites/landingpage/IntrinsicsGuide/ (дата обращения: 10.12.2021).

8. *Морелос-Сарагоса Р.* Искусство помехоустойчивого кодирования. Методы, алгоритмы, применение. М.: Техносфера, 2005. 320 с.

УДК 528-88 **DOI: 10.31799/978-5-8088-1701-2-2022-2-325-329**

В. А. Миклуш
старший преподаватель
Санкт-Петербургский государственный университет аэрокосмического приборостроения

ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНАЯ СИСТЕМА ДИСТАНЦИОННОГО ЗОНДИРОВАНИЯ АКВАТОРИИ ПОРТА

Рассмотрена структура комплексной измерительной системы дистанционного мониторинга. Она включает активные и пассивные дистанционные системы наблюдения за акваторией. Предложена структурная схема, объединяющая источники информации, центральный процессор и лицо, принимающее решения. Рассмотрена возможность применения системы управления движением судов (СУДС) для экологического мониторинга морской поверхности.
Ключевые слова: *экологический мониторинг, информационно-измерительные системы, радиолокационные подсистемы.*

V. A. Miklush
Senior Lecturer
St. Petersburg State University of Aerospace Instrumentation

INFORMATION-MEASURING SYSTEM FOR REMOTE SENSING OF THE PORT WATER AREA

The structure of an integrated measuring system for remote monitoring is considered. It includes active and passive remote monitoring systems for the water area. A structural diagram is proposed that combines: sources of information, a central processor and a decision-maker. The possibility of using a vessel traffic control system (VTCS) for environmental monitoring of the sea surface is considered.
Keywords: *environmental monitoring, information-measuring systems, radar subsystems.*

Введение

Проблемы исследования морей и океанов, рационального использования их ресурсов имеют в настоящее время большое значение. На первое место встает вопрос защиты морской среды от антропогенного загрязнения. Для эффективного решения поставленной задачи необходим комплексный подход, использующий современные информационно-измерительные технологии выявления и распознавания аномалий физических полей океана, возникающих вследствие загрязнения. Сегодня при мониторинге морской поверхности стали актуальными дистанционные методы исследования Земли (ДМИЗ). Они основаны на анализе структуры сформированных в результате отражения от морской поверхности электромагнитных волн от природных или искусственных образований сигналов.

При решении экологических проблем используют глобальные, региональные и объектовые системы мониторинга акваторий и прибрежных зон. Системам объектового мониторинга морской акватории придается большое значение, хотя их зона ответственности имеет значительно меньшие размеры, чем региональных, но сегодня важна еще и оперативность получения информации об экологическом состоянии морской акватории в условиях возникновения аварийных ситуаций при разливах нефти, нефтепродуктов и других экологически опасных веществ [1].

Поскольку в акваториях крупных морских портов осуществляется наиболее интенсивное движение судов, то остро встает проблема обеспечения экологического контроля акватории порта, для решения которой необходимо организовать службы, отвечающие за обеспечение экологического контроля акватории порта и оценку загрязненности водной среды.

При проектировании комплексной измерительной системы дистанционного мониторинга акватории порта необходимо:

– провести анализ особенностей морской акватории (географическое расположение, береговой рельеф, особенности течений, температуры и т. п.);

– определить круг задач, решаемых системой мониторинга;

– провести оценку возможности решения поставленных задач на основе существующей современной материально-технической базы;

– определить состав перспективных средств наблюдения и контроля за морской акваторией.

Важное условие решения задач, поставленных перед многосенсорной системой мониторинга акваторий порта, – возможность получения оперативной, достоверной информации о состоянии объекта наблюдения. Анализ этой информации позволяет оценить состояние акватории и принять меры для предотвращения чрезвычайных ситуаций, например разливов нефти и нефтепродуктов на водную поверхность, их растекание по водной поверхности. На систему мониторинга акваторий порта возложено решение следующих задач [2, 3]:

– сбор и хранение данных о состоянии морской акватории;

– обработка и классификация информации о состоянии морской акватории для дальнейшей оценки ее текущего состояния;

– предоставление доступа к информации, необходимой для принятия оперативных управляющих решений по ликвидации ЧС в реальном (квазиреальном) масштабе времени;

– анализ и прогноз развития ситуации;

– архивация информации, создание банка данных и моделей ситуаций с определением численных критериев ЧС;

– оценка степени риска возникновения ЧС.

Для решения поставленных задач в системе экологического мониторинга используется множество датчиков различной физической природы. Однако увеличение числа различных датчиков и точек пространства, в которых ведется наблюдение, значительно удорожает систему мониторинга акватории порта. Вследствие этого перед разработчиками встает задача оптимизации структуры системы экологического мониторинга.

Источники информации в комплексной системе мониторинга

Информационные системы, являющиеся источниками информации для системы экологического мониторинга порта, представлены на рис. 1. В состав комплексной системы экологического мониторинга акватории порта могут входить как активные, так и пассивные дистанционные системы наблюдения за акваторией порта. Эти системы представляют наибольший интерес для осуществления контроля за морской поверхностью с точки зрения оперативного получения информации.

Рис. 1. Структурная схема источников информации

Как видно из рис. 1, для экологического мониторинга морской поверхности возможно применение как активных, так и пассивных методов зондирования морской поверхности акватории. К активным подсистемам относятся оптические (лидары), радиолокационные и гидроакустические подсистемы [4].

Лидарные подсистемы относятся к оптическим системам зондирования, которые используют импульсные или непрерывные световые потоки при зондировании морской поверхности.

Радиолокационные подсистемы (РЛП) используются для дистанционного мониторинга морской поверхности в зоне ответственности и являются всепогодными. Их задачи:

– оценка основных характеристик и параметров морской поверхности (сила волнения, направление и скорость течений на водной поверхности);

– обнаружение и оценка основных параметров исследуемых объектов;

– обнаружение аномалий, загрязняющих выбросов (разливы нефти и нефтепродуктов) на водной поверхности,

– определение координат загрязнения, их размеров;

– контроль и прогнозирование динамики загрязнений.

РЛП состоит из сети радиолокационных станций (РЛС) сантиметрового и миллиметро-

вого диапазонов [5]. Для целей обнаружения загрязнения нефтью и нефтепродуктами наиболее информативны РЛС миллиметрового диапазона волн, но их существенный недостаток состоит в том, что в условиях дождя они слепнут. В связи с этим в РЛП необходимо использовать дублирующие РЛС сантиметрового диапазона (λ от 1 до 10 см). Это обусловлено сохранением их работоспособности при интенсивных осадках.

Гидроакустические подсистемы используются для изучения глубинного строения толщи воды и дна, определения и локации мест скопления пролитых нефтепродуктов.

Метеостанция применяется для получения оперативной информации о метеорологических параметрах и характеристиках атмосферы (температура, сила и направление ветра и т. д.) в зоне наблюдения с целью точного выявления возможного загрязнения и его распространения.

При совместной обработке данных в системе мониторинга анализируются: оптические изображения (телекамеры); тепловые портреты акватории (инфракрасные телекамеры), портреты акватории (лазерные локаторы); радиолокационные изображения. Для представления заключения о наличии либо отсутствии загрязнения, о количественном и качественном составе загрязняющих веществ проводится анализ данных, полученных от используемых подсистем. Заключение передается службам предотвращения и ликвидации загрязнений. Также в системе мониторинга используется обратная связь для корректировки работы первичных информационных подсистем.

Структура комплексной системы мониторинга акватории

Структурная схема связи комплексной системы экологического мониторинга морской акватории показана на рис. 2.

Каждая подсистема имеет собственный локальный процессор, предназначенный для управления подсистемой и обработки первичной информации, получаемой от нее. Работой локальных процессоров управляет центральный процессор. В круг его задач входят осуществление совместной обработки данных (СОД) от различных подсистем и анализ: радиолокационных и оптических изображений; тепловых портретов акватории; результатов поверхностно-частотно-временной обработки сейсмоакустических сигналов и объемно-частотно-временной обработки гидроакустических сигналов.

Обнаружение нефтяных разливов и контроль за их распространением в комплексной системе экологического мониторинга морской поверхности осуществляются автоматическими средствами контроля и с помощью оператора. Оператор является лицом, принимающим решение (ЛПР) о наличии (отсутствии) загряз-

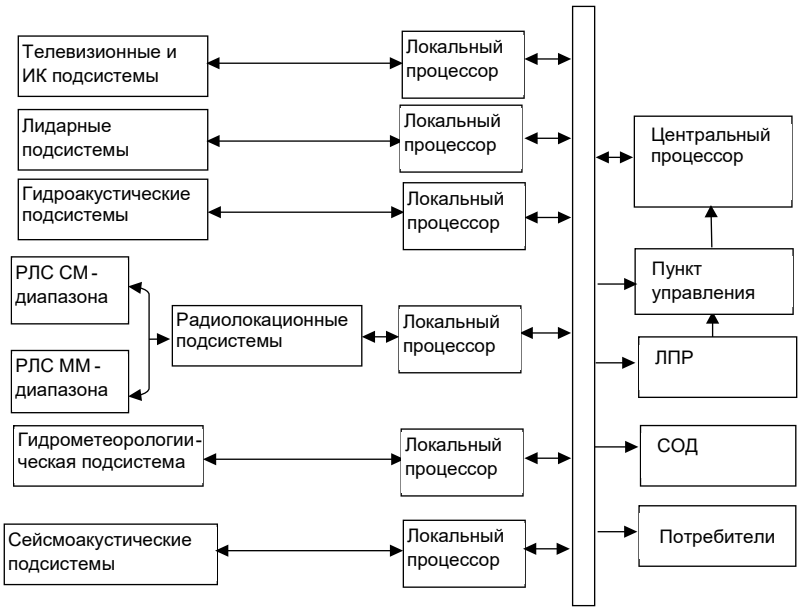


Рис. 2. Структурная схема связи комплексной системы мониторинга морской поверхности

нения. Решение выносится на основе анализа полученных данных от радиолокационных, телевизионных и других подсистем, расположенных в зоне ответственности системы мониторинга. Важность поставленной задачи и высокая ответственность за принимаемые решения приводит к тому, что наряду с высокоэффективными техническими средствами необходимо привлечение ЛПР [6].

Применения СУДС в экологическом мониторинге акватории порта

Система мониторинга акватории порта является многосенсорной, следовательно, с экономической точки зрения затратной. Главную роль в системе обнаружения нефтяных пятен на морской поверхности играют методы дистанционного мониторинга, в основе которых лежат радиолокационные средства наблюдения за морской поверхностью. Поэтому для уменьшения стоимости системы мониторинга акватории порта целесообразно рассмотреть возможность использования уже существующих в порту радиолокационных и других подсистем [7]. Например, такой как система управления движением судов (СУДС). Привлекательность подобной реализации определяется не только экономическими факторами, но и тем, что существующие радиолокаторы управления движением судов обладают высокой азимутальной разрешающей способностью.

Система мониторинга и телеуправления СУДС отвечает за межсистемную интеграцию данных, сопровождение цели по нескольким РЛС, осуществление прогноза движения цели, за интеграцию АИС и РЛС информации, обработку радиолокационных данных, запись и воспроизведение данных, объявление тревоги. Кроме этого, имеется ряд дополнительных функций, которые обеспечиваются за счет опциальных дополнительных модулей и приложений.

Контролируемые подсистемы СУДС [8]:

– РЛС;

– контроллеры видеокамер;

– радиопеленгаторы;

– станции АИС;

– метеостанции;

– телекоммуникационное оборудование;

– компьютеры (чип мониторинга материнской платы);

– источники бесперебойного питания (UPS);

– контроллеры параметров систем жизнеобеспечения (температура, влажность);

– контроль доступа и системы пожарной сигнализации и пожаротушения;

– контроль статуса всех объектов СУДС;

– детализированная информация о всех тревогах и событиях;

– дружественный графический интерфейс;

– возможность подключения фактически любого устройства, поддерживающего SNMP-протокол.

Зона действия СУДС должна быть перекрыта рабочими зонами основных технических средств. Рабочие зоны определяются для каждого вида основных технических средств СУДС посредством натурных испытаний. Акватории морских портов должны быть перекрыты рабочими зонами РЛС.

Основной источник определения загрязнения акватории порта – радиолокационные станции, которые есть в СУДС, при этом они обеспечивают полное перекрытие акватории порта, имеют дублирующие подсистемы, работают в двух диапазонах волн, чаще всего 3 и 10 см, каждая из станций имеет свой радарный процессор для первичной обработки информации. При этом в центральный процессор передается также первичная радиолокационная информация. В СУДС уже осуществлена интеграция данных от различных источников, а также предусмотрена возможность отображения первичной радиолокационной информации поверх электронной карты. Для использования СУДС в мониторинге акватории порта необходимо добавить еще один локальный процессор, который будет обрабатывать радиолокационную информацию уже с точки зрения обнаружения нефтяных разливов. Для этого необходимо разработать алгоритмы обработки полученной радиолокационной информации и реализовать по ним расчеты в локальном процессоре.

Заключение

Достоинствами рассматриваемой реализации мониторинга акватории порта с использованием СУДС являются, во-первых, экономическая целесообразность, которая обусловлена использованием уже существующих в порту радиолокационных систем и постов, входящих в ее состав; во-вторых, оперативность и непрерывность получения информации; в-третьих – высокая азимутальная разрешающая способность существующих радиолокаторов СУДС. Энергетический потенциал используемых РЛС достаточен для обнаружения неоднородностей, вызван-

ных загрязнением морской поверхности, поскольку размеры зоны контроля не требуют большой дальности действия от используемых РЛС. В связи с этим для решения задачи экологического контроля водной поверхности целесообразно использование РЛС системы управления движением судов (СУДС), которыми оснащаются порты.

Библиографический список

1. *Михайлов В. В.* Системы метеорологического, экологического и аэрокосмического мониторинга. М.: Радиотехника, 2015. 184 с.

2. *Миклуш В. А., Сикарев И. А., Татарникова Т. М.* Организация экологического мониторинга акватории порта посредством обработки помехозащищенного сигнала системы управления движением судов // Проблемы информационной безопасности. Компьютерные системы. 2020. № 4. С. 72–78.

3. Результаты исследований в области дистанционных методов обнаружения нефтяных загрязнений на водной поверхности, проводимых в РГГМУ / П. П. Бескид, П. Ю. Богданов, В. А. Миклуш [и др.] // Гидрометеорология и экология. 2020. № 60. С. 371–391.

4. Радиолокационные системы / В. В. Ахияров, С. И. Нефедов, А. И. Николаев [и др.]. М.: Изд-во МГТУ им. Н. Э. Баумана, 2016. 349 с.

5. *Crowly J. L., Demazeau Y.* Principles and techniques for sensor data fusion // Signal Processing. 1993. № 32. P. 5–27.

6. *Миклуш В. А., Татарникова Т. М., Палкин И. И.* Решение задачи экологического мониторинга акватории порта с помощью распределенной системы датчиков // Известия высших учебных заведений. Приборостроение. 2021. Т. 64. № 5. С. 404–411.

7. *Бескид П. П., Шишкин А. Д.* Об опыте проведения экологического мониторинга состояния морской поверхности радиолокационными средствами // Безопасность жизнедеятельности. 2011. № 2 (122). С. 20–24.

8. Об утверждении требований к радиолокационным системам управления движением судов, объектам инфраструктуры морского порта, необходимым для функционирования глобальной морской системы связи при бедствии и для обеспечения безопасности, объектам и средствам автоматической информационной системы, службе контроля судоходства и управления судоходством: приказ Министерства транспорта РФ от 23 июля 2015 г. № 226. URL: https://base.garant.ru/71236210/ (дата обращения: 24.11.2021).

СВЕДЕНИЯ ОБ АВТОРАХ

Аграновский Андрей Владимирович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – современные информационные системы и технологии, информатизация общества
a_agranovskii@mail.ru

Акопян Белла Кареновна
аспирант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровая обработка сигналов и изображений, современная биомедицинская электроника, математическое моделирование с использованием сред компьютерной алгебры
akopyan.bella@yandex.ru

Альмухамедов Алексей Игоревич
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность, биометрическая идентификация
aleksei.alm@gmail.com

Андреев Андрей Александрович
студент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – вычислительные методы, теория чисел
andre001155@yandex.ru

Антипова Алена Александровна
магистрант кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – визуальные новеллы, изобразительное искусство, интерактивная 2D- и 3D-графика
alyona_ant5@mail.ru

Артемьев Илья Сергеевич
студент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – 3D-моделирование и анимация персонажей, интерактивная 3D-графика
warframe9800@gmail.com

Афанасьева Виктория Игоревна
студент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – компьютерное зрение, обучение нейронных сетей, сегментация изображений, моделирование в среде MATLAB
victoria_afanaseva@mail.ru

Бакин Евгений Александрович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровая обработка сигналов, статистический анализ данных, теория принятия решений, компьютерное моделирование
jenyb@mail.ru

Балонин Николай Алексеевич
доктор технических наук, профессор кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – динамические системы, матрицы Адамара, Критские матрицы, интернет-робототехника, интернет-книги
corbendfs@mail.ru

Балонин Юрий Николаевич
инженер кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – вычислительные методы, теория чисел, моделирование с использованием сред компьютерной математики
tomaball@mail.ru

Богоявленский Глеб Анатольевич
студент кафедры безопасности информационных систем, стажер-программист БФТ. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, технологии защиты информации, криптография
glebik555@bk.ru

Боженко Виктория Вячеславовна
ассистент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, искусственный интеллект, машинное обучение
vibozhenko@yandex.ru

Борисовская Анна Владимировна
ассистент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы передачи данных
borisovskaya@k36.org

Вересова Алина Максимовна
ассистент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория информации и кодирования, низкоплотностные коды, постквантовая криптография
amveresova@gmail.com

Виноградова Екатерина Петровна
старший преподаватель кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – компьютерная обработка изображений, корреляционный анализ, математическое моделирование с использованием сред компьютерной алгебры
kate_v@rambler.ru

Витвинов Валерий Константинович
магистрант кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – технологии передачи данных в системах с низким энергопотреблением, LoRa, мобильная разработка
vitvinov-98@mail.ru

Вихров Владислав Владимирович
магистрант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – статистический анализ данных, data science, data engineering
vlad.v090@gmail.com

Волкова Анастасия Сергеевна
студент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – интеллектуальные сети электроснабжения, моделирование в среде Simulink
volkova2305@bk.ru

Воронов Андрей Владимирович
кандидат технических наук, доцент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность, технические средства защиты информации
voron-a@inbox.ru

Галкин Дмитрий Денисович
студент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – беспроводные технологии, интернет вещей
guapower@yandex.ru

Георгиев Георги Димитров
кандидат технических наук. Центр энергетических решений, Варна, Болгария. Область научных интересов – моделирование энергосистем
georgidg@gmail.com

Геппенер Владимир Владимирович
доктор технических наук, профессор кафедры математического обеспечения и применения ЭВМ. Санкт-Петербургский государственный электротехнический университет «ЛЭТИ». Область научных интересов – обработка телеметрии и виброизмерений, анализ данных, прогнозирование
geppener@mail.ru

Глушенкова Алина Юрьевна
студент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы передачи данных, множественный доступ
aln.g99@mail.ru

Гордеев Александр Владимирович
доктор технических наук, профессор кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – операционные системы, параллельные вычислительные системы, администрирование информационных систем
ff2avg@mail.ru

Горелик Денис Вадимович
аспирант кафедры вычислительных систем и программирования. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – системы динамического распараллеливания в кластерных системах
den5509@mail.ru

Григорьева Наталья Никифоровна
старший преподаватель кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – моделирование процессов в технических системах
gera_nn@mail.ru

Гурнов Константин Борисович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровая обработка сигналов и изображений различной физической природы, машинное обучение, статистический анализ данных
kocta4212@mail.ru

Давидович Борис Владимирович
магистрант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – микропроцессорные и микроконтроллерные устройства, биотехнические системы, датчики физических параметров, анализ данных
davidovichborisvladimir@yandex.ru

Ельцова Анастасия Денисовна
студент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование, мониторинг технического состояния систем очистки сточных вод
eltsova_ad@mail.ru

Ерышов Вадим Георгиевич
кандидат технических наук, доцент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность
eryshov@mail.ru

Ерышов Никита Вадимович
студент кафедры безопасности информационных систем, администратор отдела цифровой трансформации и сервисов управления цифрового развития. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов: информационная безопасность, цифровые технологии
env1701@mail.ru

Жаринов Олег Олегович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – современная электроника, цифровая обработка сигналов, математическое моделирование с использованием сред компьютерной алгебры
zharinov73@inbox.ru

Загураева Мария Викторовна
ассистент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные системы и технологии, бизнес-аналитика
m.v.zaguraeva@gmail.com

Зиатдинов Сергей Ильич
доктор технических наук, профессор кафедры информационно-сетевых технологий. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – схемотехника телекоммуникационных устройств, информационные системы, оптимизация
zsi@k53.guap.ru

Зулкашев Руслан Саматович
студент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – сверточные нейронные сети, генеративные модели
rusik.zulkashev@gmail.com

Зыков Дмитрий Александрович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – распределенные системы, системы хранения данных
zykov.d11@mail.ru

Иванова Мария Станиславовна
аспирант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование
ivanovamariya_94@mail.ru

Исаева Мария Николаевна
ассистент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – стеганография, теория кодирования, методы обеспечения информационной безопасности
imn@guap.ru

Исаков Виктор Иванович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – моделирование сложных бортовых технических систем
dept41@aanet.ru

Кабанец Анастасия Григорьевна
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – криптография, технологии защиты информации
kabanets333@gmail.com

Канаров Вячеслав Сергеевич
студент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – беспроводные технологии, интернет вещей
v.kanarov@gmail.com

Килимник Вячеслав Александрович
кандидат технических наук, начальник научно-исследовательского отдела биотехнических проблем (НИО БП). Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – микромощная электроника, имплантируемые системы, микропроцессорная техника, многоканальная функциональная электростимуляция, биотехнические системы, учебно-исследовательские комплексы по биофизике
kil-aanet@ya.ru

Ким Дмитрий Константинович
кандидат физико-математических наук, ассоциированный профессор. Университет Нархоз, Алматы, Казахстан. Область научных интересов – вероятностно-статистические методы и их приложения
kdk26@mail.ru

Клименко Анастасия Александровна
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы передачи данных, эллиптические кривые, моделирование и анализ полученных данных
klimenkoanastasia412@gmail.com

Клионский Дмитрий Михайлович
кандидат технических наук, доцент кафедры математического обеспечения и применения ЭВМ и информационных систем. Санкт-Петербургский государственный электротехнический университет «ЛЭТИ». Область научных интересов – математическое и компьютерное моделирование, обработка сигналов и изображений, Matlab, обработка телеметрии и виброизмерений, анализ данных, прогнозирование
kdm1986@gmail.com

Клюканов Виталий Константинович
студент кафедры прикладной математики. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, искусственный интеллект, машинное обучение
klukanovv@mail.ru

Ключарев Александр Анатольевич
кандидат технических наук, доцент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – имитационное моделирование, обработка экспериментальных данных, системы реального времени
ak@aanet.ru

Коломойцев Владимир Сергеевич
кандидат технических наук, доцент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – нормативно-правовая безопасность; построение защищенных инфокоммуникационных систем; оценка эффективности инфокоммуникационных систем
dek-s-kornis@yandex.ru

Комашинский Владимир Владимирович
кандидат технических наук, доцент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – анализ защищенности компьютерных систем, защищенные информационные системы
gvladkom@gmail.com

Коржук Владислав Сергеевич
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность, нейросетевые технологии
14korzhuk@mail.ru

Кочин Дмитрий Александрович
аспирант кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – геоинформационные системы, экспресс-анализ случайных процессов
dimakohin@yandex.ru

Кузнецов Виталий Александрович
кандидат технических наук, доцент кафедры информационно-сетевых технологий. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровая обработка сигналов, компьютерное зрение, трехмерное моделирование
k.avk-c@mail.ru

Ларионец Кирилл Андреевич
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность
sirexword@gmail.com

Летуновская Полина Сергеевна
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность, системы хранения данных
ponita@mail.ru

Лучкин Арсений Сергеевич
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, разработка защищенных веб-приложений
luchkin.arseniy@gmail.com

Майн Екатерина Евгеньевна
ассистент кафедры вычислительных систем и сетей, аспирант. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – виртуальная и дополненная реальность, компьютерная графика, цифровые миры
emajn3dsma@gmail.com

Маралов Никита Игоревич
студент кафедры безопасности информационных систем, инженер-программист V4Scale. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – системы непрерывной поставки и интеграции программного обеспечения, защита информации, информационные технологии, криптография
blacknikboard@gmail.com

Медведев Олег Стефанович
доктор медицинских наук, заведующий кафедрой фармакологии факультета фундаментальной медицины. Московский государственный университет имени М. В. Ломоносова. Область научных интересов – биотехнические системы, учебно-исследовательские биотехнические комплексы, применение современной техники в лабораториях для экспериментов на животных
oleg.omedvedev@gmail.com

Миклуш Виктория Александровна
старший преподаватель кафедры информационно-сетевых технологий. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – экологический мониторинг морской поверхности, радиолокация
miklush-v@yandex.ru

Минаева Виолетта Андреевна
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – защита информации, оценка рисков автоматизированных систем
veta2000@icloud.com

Мирошниченко Никита Игоревич
студент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – интеллектуальные сети электроснабжения, моделирование в среде Simulink
nikitos_mir.1997@mail.ru

Михайлов Валентин Юрьевич
студент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы передачи данных, множественный доступ
valyamih1@gmail.com

Недошивин Павел Петрович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория игр, постквантовая криптография
ppnedoshivin98@mail.ru

Ненашев Вадим Александрович
кандидат технических наук, доцент кафедры вычислительных систем и сетей, руководитель лаборатории интеллектуальных технологий и моделирования сложных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – обработка и передача информации, пространственно-распределенные системы, компьютерное моделирование, системы навигации и радиолокации, статистический анализ, интеллектуальные технологии
nenashev@guap.ru

Нестеренков Олег Александрович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы передачи данных
olegnest20@gmail.com

Никитин Александр Васильевич
кандидат технических наук, доцент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровые реальности, виртуальная и дополненная реальности, дополненная виртуальность и различные области их применения
guap.nike@mail.ru

Овчинников Андрей Анатольевич
кандидат технических наук, заведующий кафедрой безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория информации и кодирования, низкоплотностные коды, постквантовая криптография
mldoc@guap.ru

Павлов Владислав Станиславович
доктор технических наук, профессор кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – пространственно-временная обработка информационных сигналов; синтез, анализ и моделирование помехоустойчивых систем управления
w14z@yandex.ru

Пойманова Екатерина Дмитриевна
доцент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные процессы, системы хранения данных, анализ данных
e.d.poymanova@gmail.com

Поляк Марк Дмитриевич
старший преподаватель кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование, синергетические системы нелинейного управления, теория случайных процессов, искусственные нейронные сети
markpolyak@gmail.com

Попов Илья Дмитриевич
ассистент кафедры информационной безопасности. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – компьютерные сети, анализ сетевого трафика
asp.i.popov@k36.org

Путилова Надежда Владимировна
старший преподаватель кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – автоматизация и управление образованием, автоматизированные информационные системы, системы управления базой данных
N_V_P_hex@mail.ru

Раскопина Анастасия Сергеевна
студент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – интеллектуальные сети электроснабжения, моделирование в среде Simulink
raskopina.nastia@yandex.ru

Решетникова Нина Николаевна
кандидат технических наук, доцент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – интерактивная 3D-графика, моделирование и анимация персонажей, виртуальная и дополненная реальность
reni_07@list.ru

Рогачев Сергей Александрович
старший преподаватель кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – геоинформационные системы, анализ данных, исследования Земли из космоса, использование аэрокосмической информации, машинное обучение, искусственные нейронные сети
rogachev.seal@gmail.com

Русанов Максим Витальевич
студент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое и компьютерное моделирование, компьютерная графика
hatrue.max@gmail.com

Рындюк Виктория Александровна
кандидат технических наук, доцент кафедры безопасности информационных систем. Санкт-Петербургский университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность
Vika012001@mail.ru

Сергеев Александр Михайлович
кандидат технических наук, доцент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровая обработка информации, сетевые технологии, системы медицинского назначения
aleks.asklab@gmail.com

Сергеев Михаил Борисович
доктор технических наук, заведующий кафедрой вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория разрядных вычислений, методы проектирования спецпроцессоров для систем контроля и управления
mbse@mail.ru

Скобцов Юрий Александрович
доктор технических наук, профессор кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – искусственный интеллект, эволюционные вычисления, моделирование и тестирование цифровых устройств
ya_skobtsov@list.ru

Скрылёв Андрей Олегович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – машинное обучение и нейронные сети в области защиты информации
andreyskrylev1@gmail.com

Сорокин Александр Васильевич
старший преподаватель кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – интеллектуальные сети электроснабжения, моделирование в среде Simulink
aleksandr.sorokin.v@gmail.com

Таубин Феликс Александрович
профессор кафедры аэрокосмических компьютерных и программных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – цифровые системы связи, методы помехоустойчивого кодирования, широкополосные системы, беспроводные сети
ftaubin@yahoo.com

Толмачев Сергей Геннадьевич
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – корреляционный анализ, теория кодирования
tsg17@yandex.ru

Трофимов Андрей Николаевич
доцент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория передачи дискретных сообщений, теория информации, теория кодирования
andrei.trofimov@vu.spb.ru

Турнецкая Елена Леонидовна
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – современные информационные системы и технологии, информатизация общества
turnetskaya@mail.ru

Тюринова Виолетта Александровна
магистрант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование сложных систем, обработка информации в условиях неопределенности, разработка программного обеспечения для имитационного моделирования
viletteee@yandex.ru

Тюрликов Андрей Михайлович
доктор технических наук, профессор, директор института информационных систем и защиты информации, заведующий кафедрой инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – теория связи, теория информации
turlikov@k36.org

Фаттахова Мария Владимировна
доцент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование, исследование операций, теория игр, моделирование программных систем
mvfa@yandex.ru

Фоменкова Анастасия Алексеевна
ассистент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – моделирование технологических процессов
a.a.fomenkova@mail.ru

Хмелевский Кирилл Александрович
студент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии, системы хранения и передачи данных, множественный доступ
k.khmelevskyy@gmail.com

Чекменёва Анна Александровна
магистрант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – микропроцессорные устройства, датчики газов, датчики физических параметров, цифровая и аналоговая обработка сигналов
chekmeneva.anna.1999@mail.ru

Чемоданов Артём Александрович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность
artem_chemodanov2000@mail.ru

Шамрицкая Дарья Сергеевна
магистрант кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область

научных интересов – микропроцессорные устройства, датчики физических параметров, биотехнические системы
dashacomka@gmail.com

Шепета Александр Павлович
доктор технических наук, профессор кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – обработка информации в условиях априорной неопределенности, математическое моделирование стохастических процессов и полей
shepeta@aanet.ru

Шепета Дмитрий Александрович
кандидат технических наук, доцент кафедры вычислительных систем и сетей. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – моделирование сложных бортовых технических систем
dima@shepeta.com

Шром Богдан Станиславович
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационная безопасность, системы хранения данных, разработка программного обеспечения
tuhlomon@gmail.com

Щеголева Александра Андреевна
студент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – математическое моделирование, синергетические системы нелинейного управления
aleksandrasheg@yandex.ru

Щёкин Сергей Валерьевич
кандидат технических наук, доцент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – информационные технологии
svs@aanet.ru

Юдина Виктория Игоревна
студент кафедры безопасности информационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – оценка рисков информационной безопасности, изучение методик анализа и управления
udinaviktoriia@gmail.com

Юрченко Анна Евгеньевна
студент кафедры компьютерных технологий и программной инженерии. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – программирование, разработка прикладных программных продуктов, дистанционное зондирование Земли
ann20001010@mail.ru

Яковлев Александр Викторович
кандидат технических наук, доцент кафедры проблемно-ориентированных вычислительных комплексов. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – машинное обучение, анализ видео- и аудиоматериалов, обработка сигналов, оценка состояния человека
sven-7@mail.ru

Янковский Никита Андреевич
ассистент кафедры инфокоммуникационных систем. Санкт-Петербургский государственный университет аэрокосмического приборостроения. Область научных интересов – системы интернета вещей, передача малых данных
yannik98@yandex.ru

СОДЕРЖАНИЕ

Научное издание

ОБРАБОТКА, ПЕРЕДАЧА И ЗАЩИТА ИНФОРМАЦИИ В КОМПЬЮТЕРНЫХ СИСТЕМАХ ‘22

Вторая Международная научная конференция 11–15 апреля 2022 г.

Сборник докладов

Ответственные за выпуск:
кандидат технических наук, доцент *В. А. Ненашев;*
А. М. Вересова

Редакторская подготовка *С. В. Денисовой*
Компьютерная верстка *А. Н. Колешко*

Подписано к печати 15.03.2022. Дата выхода в свет: 30.03.2022. Формат 60x84 1/8.
Усл. печ. л. 39,6. Уч.-изд. л. 41,1. Тираж 150 экз. Заказ № 95.

Редакционно-издательский центр ГУАП
190000, г. Санкт-Петербург, ул. Б. Морская, д. 67, лит. А

Распространяется бесплатно